

Máster en Ingeniería de Sistemas y Control



Trabajo Fin de Máster

Reconocimiento de una plataforma para aproximación de vehículos no tripulados mediante técnicas de Visión por Computador

Alumno: José Antonio García Pulido
Director: Gonzalo Pajares Martinsanz

Curso Académico 2013/2014 – Convocatoria de Junio

Máster en Ingeniería de Sistemas y Control



Área de Visión por Computador

Trabajo Fin de Máster

Reconocimiento de una plataforma para aproximación de vehículos no tripulados mediante técnicas de Visión por Computador



Autorización

Se autoriza a la Universidad Complutense de Madrid (UCM) y a la Universidad Nacional de Educación a Distancia (UNED) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o el prototipo desarrollado.

Firmado:

Resumen

Este Trabajo Fin de Máster, que está materializado a través de este documento que constituye la memoria del mismo, tiene como objetivo fundamental tratar de resolver un problema real inherente a la aproximación de vehículos no tripulados mediante el uso de técnicas de **Visión por Computador**. Por consiguiente, este proyecto se encuentra totalmente enmarcado de una manera intrínseca en el área de la Visión por Computador, pero aplicada a un entorno muy real y concreto, que está en claro auge, como son los Vehículos Aéreos no Tripulados ó UAVs (de su acrónimo en inglés *Unmanned Aerial Vehicle*).

En la actualidad existe una clara tendencia generalizada para la automatización de determinadas tareas, como pueden ser la gestión de incendios, vigilancia marítima, el análisis de plantaciones, etcétera, mediante el uso de UAVs que lo llevan a cabo en forma de misiones. En este sentido, uno de los procesos clave que hay que abordar a la hora de finalizar una misión para que el UAV pueda ser reutilizado, es gestionar su aproximación para garantizar que se lleva a cabo un aterrizaje seguro y preciso en el área que queda delimitada por su estación base a través de una plataforma de aproximación

Cabe reseñar, que la plataforma de aproximación que será objeto de estudio en este trabajo, tiene dibujada sobre su superficie una imagen peculiar que atiende a una morfología específica que sirve para tratar de garantizar la imposibilidad de que aparezca de forma natural en un entorno agreste. Dicha figura, será capturada como parte de una imagen a través de la cámara que el UAV alberga en su interior, para ser analizada posteriormente mediante el uso de técnicas de Visión por Computador como son la binarización, segmentación, etiquetado y reconocimiento de regiones, etc. Tras este proceso, el vehículo autónomo deberá discernir si se trata realmente de su estación base, y si es así, llevar a cabo la aproximación, y por ende el susodicho aterrizaje o ataque (si se trata de un vehículo marino).

Palabras Clave

Visión por computador, Imágenes de exterior, Binarización, Espacios de color, Etiquetado, Segmentación, Búsqueda de regiones, Estación base, Imágenes aéreas, Plataforma de Aproximación, Eficiencia computacional, Aproximación, Vehículo autónomo, Reconocimiento de patrones.



A todas aquellas personas que han entregado su vida a la ciencia, ya que sin su trabajo previo nada de esto hubiera sido posible.

Agradecimientos

En primer lugar, me gustaría agradecer a Gonzalo, mi director de proyecto, por toda la ayuda prestada y por darme la oportunidad de poder colaborar y aportar mi granito de arena en un proyecto real como éste. Esto ha hecho que mi motivación e interés en el TFM hayan crecido exponencialmente hasta alcanzar unos resultados inimaginables antes de comenzar mi andadura en este Máster.

A Rebeca, por la enorme paciencia que ha tenido durante todos estos años, y a la cantidad de sacrificios que ha realizado para que yo alcanzara esta gran meta. Su ayuda noble e incondicional, me ha dado la estabilidad emocional necesaria para superar este gran reto, a pesar de que su vida personal se ha visto lastrada por las ingentes cantidades de tiempo que he dedicado a este trabajo. Claro ejemplo ha sido la ayuda que me ha prestado durante el proceso de adquisición de imágenes, para sujetar el papel A3 y que no volara entre imagen e imagen, etcétera.

Finalmente, y no menos importante, quisiera agradecer también de corazón, a mis padres y hermano el apoyo y la confianza que me han proporcionado durante toda mi vida. Sin su ayuda, sus consejos, y su esfuerzo, siempre tratándome de orientar hacia el buen camino, nunca hubiera recibido una educación que me permitiera abordar un trabajo de este tipo.

ÍNDICE de Contenidos

Autorización	4
Resumen	5
Palabras Clave	6
Agradecimientos.....	8
Listado de Acrónimos	13
Capítulo 1: Descripción del problema y objetivo	14
1.1 – Introducción	14
1.2 – Descripción del Problema	15
1.2.1 – Problema con la adquisición	16
1.3 – Objetivos	17
1.4 – Organización del Trabajo.....	17
Capítulo 2: Estado del Arte	18
2.1 – Visión General	18
2.1.1 – Casos de Estudio	18
2.1.2 – Patentes	22
2.2 – Adquisición de imágenes.....	25
2.2.1 – Cámara	25
2.2.2 – Conversores A/D.....	25
2.2.3 – Unidad de Procesamiento.....	27
2.2.4 – Condiciones de Adquisición de Imágenes	27
Capítulo 3: Solución Propuesta	29
3.1 – Visión por Computador	29
3.1.1 – Imagen digital	29
3.1.2 – Análisis preliminar de la figura de aproximación.....	30
3.2 – Diagramas	31
3.2.1 – Diagrama de bloques (granularidad gruesa)	31
3.2.2 – Diagrama de flujo (granularidad fina)	31
3.2.3 – Depuración visual del proceso	36
3.3 – Transformación espacial.....	37
3.3.1 – Escalado	37
3.3.2 – Rotación	38
3.4 – Espacios de Color.....	39
3.4.1 – La luz, el color y la visión	39
3.4.2 – RGB	40
3.4.3 – HSV	41
3.4.4 – Conversión RGB a HSV	41
3.5 – Umbralización.....	42
3.5.1 – Estrategias de Umbralización.....	42
3.5.2 – Descripción Formal.....	43
3.5.3 – Tipos de umbralización	43
3.5.4 – Método de Otsu	44
3.5.5 – Decisión adoptada.....	45
3.6 – Histograma	47
3.7 – Binarización.....	49
3.8 – Segmentación.....	49
3.8.1 – Vecindad entre píxeles.....	50
3.8.2 – Conectividad	51
3.8.3 – Etiquetado de componentes conexas	52
3.9 – Descripción de Regiones	54
3.9.1 – Propiedades métricas	55

3.9.2 – Propiedades topológicas.....	57
3.9.3 – Momentos Geométricos	58
3.10 – Descripción de líneas y contornos.....	60
3.10.1 – Transformada de Hough.....	60
3.11 – Geometría y Distancia.....	62
3.11.1 – Elipse	62
3.11.2 – Circunferencia.....	65
3.11.4 – Distancia Euclídea	67
Capítulo 4: Problemas y Decisiones Adoptadas.....	68
4.1 – Problemas de Umbralización / Binarización	68
4.2 – Problemas Geométricos	70
4.3 – Problemas de Orientación.....	76
4.4 – Problemas de Eficiencia Computacional	77
Capítulo 5: Resultados y Pruebas	89
5.1 – Análisis Comparativos Efectuados	89
5.2 – Pruebas, Datos y Resultados	90
5.3 – Detalle de las imágenes procesadas	91
5.4 – Comentarios de las imágenes identificadas.....	97
5.5 – Síntesis y Resumen de Resultados.....	99
5.6 – Gráficas de Efectividad Por Imagen	102
5.7 – Gráficas de Efectividad Por Test	103
5.8 – Evaluación del Sistema de Probabilidad Final	104
5.9 – Resumen y Conclusiones.....	105
Capítulo 6: Conclusiones y Trabajo Futuro	107
6.1 – Conclusiones del Análisis.....	107
6.2 – Conclusiones de Resultados.....	110
6.3 – Líneas de Actuación Futura.....	112
Referencias y Herramientas Empleadas	115
ANEXO I – Tabla de Imágenes (A.I)	119
ANEXO II – Tabla de Resultados (A.II)	126
ANEXO III – Tabla de Datos (A.III)	147
ANEXO IV – Código fuente	168

ÍNDICE de Figuras

Figura 1-1 – Plataforma de aproximación con figura geométrica	15
Figura 1-2 – Adquisición (a) Problema adquisición 1; (b) Problema adquisición 2..	16
Figura 2-1 – Plataforma de aproximación de Guili Xu	19
Figura 2-2 – Plataforma de aproximación de Sven Lange	19
Figura 2-3 – Plataforma de aproximación de Srikanth Saripalli	20
Figura 2-4 – Plataforma de aproximación de Cory S. Sharp.....	20
Figura 2-5 – Método de aproximación y aterrizaje de Andrea Cesetti.....	21
Figura 2-6 – Plataforma aterrizaje automático (patente US20090306840)	23
Figura 2-7 – Tablero pseudo-aleatorio de la patente US5525883A	25
Figura 2-8 – Sensor CCD y Conversores A/D	26
Figura 3-1 – Orientación final y etiquetado de las regiones de la figura.....	30
Figura 3-2 – Diagrama de bloques de la solución con granularidad gruesa	31
Figura 3-3 – Diagrama de flujo con la solución propuesta de granularidad fina	32
Figura 3-4 – Depuración visual - Imagen Original (a) Original; (b) Binaria; (c) Regiones; (d) Binaria Limpia Elipses Identificadas.....	36
Figura 3-5 – Depuración visual - Imagen Orientada (a) Original; (b) Binaria; (c) Regiones; (d) Binaria Limpia Elipses y Círculos Identificados	36
Figura 3-6 – Espacios de Color (a) Cubo RGB; (b) Cono HSV.....	41
Figura 3-7 – Histograma (a) Figura geométrica plataforma; (b) Histograma.....	48
Figura 3-8 – Binarización (a) Imagen Original; (b) Imagen Binarizada.....	49
Figura 3-9 – Adyacencia (a) Adyacencia por frontera; (b) Adyacencia por esquina	50
Figura 3-10 – Vecindad (a) Vecindad-4; (b) Vecindad-8	51
Figura 3-11 – Conectividad (a) Arreglo Píxel; (b) Conectividad-8; (c) Conectividad mezclada.....	52
Figura 3-12 – Segmentación (a) Imagen binarizada; (b) Etiquetado componentes	54
Figura 3-13 – Transformada de Hough	61
Figura 3-14 – Elipse (a) Relación con el Cono; (b) Elementos principales	63
Figura 4-1 – Diagrama de flujo explicativo de la función ObtenerElispses	72
Figura 4-2 – Diagrama de flujo funciones IdentificarC1_C2 IdentificarC3_C4	75
Figura 4-3 – Propiedad orientación de la región	77
Figura 5-1 – Calculo propiedades imágenes durante adquisición	99
Figura 5-2 – Gráfica (a) Imagen Distancia; (b) Imagen Inclinación	102
Figura 5-3 – Gráfica (a) Imagen Luminosidad; (b) Tiempo de Ejecución	102
Figura 5-4 – Gráfica (a) Test Distancia; (b) Test Inclinación.....	103
Figura 5-5 – Gráfica (a) Test Luminosidad; (b) Tiempo de Ejecución por Test.....	103
Figura 5-6 – Gráfica de evaluación del resultado (probabilidad final)	105
Figura 6-1 – Características geométricas y umbrales de detección	109

ÍNDICE de Tablas

Tabla 1-1 – Listado de Acrónimos	13
Tabla 4-1 – Estudio Computacional Identificación Círculos – Ordenación Burbuja ..	83
Tabla 4-2 – Resumen Rendimiento Estrategias – v23.6 Burbuja	83
Tabla 4-3 – Estudio Computacional Identificación Círculos – v24.5 Quick Sort	85
Tabla 4-4 – Comparativa Rendimiento Estrategias – Ordenación Quick Sort	85
Tabla 4-5 – Supuesto número de ciclos por tipo de operación.....	86
Tabla 5-1 – Listado de Análisis Parciales Comparativos.....	90
Tabla 5-2 – Configuración empleada en el detalle de imágenes procesadas.....	91
Tabla 5-3 – Detalle de Procesado IMG_A043.jpg.....	92
Tabla 5-4 – Detalle de Procesado IMG_A044.jpg.....	92
Tabla 5-5 – Detalle de Procesado IMG_A045.jpg.....	93
Tabla 5-6 – Detalle de Procesado IMG_A065.jpg.....	93
Tabla 5-7 – Detalle de Procesado IMG_A068.jpg.....	94
Tabla 5-8 – Detalle de Procesado IMG_A066.jpg.....	94
Tabla 5-9 – Detalle de Procesado IMG_A067.jpg.....	95
Tabla 5-10 – Detalle de Procesado IMG_A069.jpg	95
Tabla 5-11 – Detalle de Procesado IMG_A034.jpg	96
Tabla 5-12 – Detalle de Procesado IMG_A038.jpg	96
Tabla 5-13 – Matriz de síntesis de resultados. Distancia e Inclinación	100
Tabla 5-14 – Matriz de síntesis de resultados. Luminosidad.....	100
Tabla 5-15 – Matriz extendida de resultados finales	101
Tabla 5-16 – Matriz de evaluación del resultado de probabilidad final	104
Tabla A.I – Tabla de Imágenes	125
Tabla A.II – Tabla de Resultados	146
Tabla A.III – Tabla de Datos	167

Listado de Acrónimos

A continuación se muestra una tabla ordenada alfabéticamente en la que se recogen algunos de los principales acrónimos usados durante la elaboración de esta memoria:

SIGLAS	IDIOMA	DESCRIPCIÓN
APC	Español	Análisis Parcial Comparativo
DSP	Inglés	Digital Signal Processor
HSV	Inglés	Hue, Saturation, Value
HSB	Inglés	Hue, Saturation, Brightness
PIC	Inglés	Peripheral Interface Controller
RGB	Inglés	Red, Green, Blue
TFM	Español	Trabajo Final de Máster
UAV	Inglés	Unmanned Aerial Vehicle
UMV	Inglés	Unmanned Marine Vehicle, UMV
UCM	Español	Universidad Complutense de Madrid
ALPR	Inglés	Automatic License Plate Recognition

Tabla 1-1 – Listado de Acrónimos

Capítulo 1: Descripción del problema y objetivo

A través de este capítulo se expone una descripción exhaustiva del problema, así como los principales objetivos que se persiguen mediante la realización de este Trabajo Fin de Máster.

1.1 – Introducción

Tal y como ya se ha comentado en una de las secciones preliminares en la que se plasmó el resumen de esta memoria, el eje de actuación de este trabajo gira en torno a un problema real y concreto que existe con la maniobra de aproximación de los vehículos no tripulados, tanto aéreos como marinos. Esta problemática, será tratada de forma más exhaustiva en el apartado *1.2 – Descripción del problema* que forma parte de esta memoria.

Gracias a los avances tecnológicos que se han producido en las últimas décadas en campos como la computación, la robótica, las redes, los sistemas empotrados, o la visión por computador (disciplinas que curiosamente dan nombre a muchas de las asignaturas impartidas en este máster), se ha hecho factible el uso de robots o vehículos no tripulados para llevar a cabo actividades que hasta hace poco tiempo se desempeñaban de forma exclusiva por los seres humanos. Cabe reseñar, que muchas de estas máquinas, que podemos adjetivar como autónomas, se basan en técnicas que se encuentran intrínsecamente relacionadas con el ámbito de la inteligencia artificial, las cuales por su idiosincrasia requieren de un elevado número de operaciones aritméticas que deben ser ejecutadas en tiempo real. Esta disciplina resulta clave para que una máquina pueda tomar una decisión en base a un aprendizaje o una clasificación previa.

A pesar de que muchos de los algoritmos y técnicas empleadas en el ámbito de la inteligencia (como pueden ser las redes neuronales, los algoritmos genéticos, los sistemas expertos, los algoritmos de clasificación, lógica *fuzzy*, redes bayesianas, etcétera) han sido desarrolladas hace ya unas cuantas décadas, la tecnología de aquel entonces no se encontraba lo suficientemente desarrollada como para lograr el pragmatismo anhelado, es decir, llevarlos a la práctica en márgenes de tiempo aceptables. Por este motivo, es probable que la tendencia al alza en lo que se refiere al uso de este tipo de máquinas autónomas se deba en gran medida a los siguientes motivos que se exponen a continuación:

- Desarrollo de procesadores más potentes que permiten ejecutar cantidades ingentes de operaciones aritméticas (enteras y en coma flotante) que hacen que sea factible emplear algoritmos de inteligencia artificial en entornos productivos y de tiempo real. Cabe reseñar, que una gran parte de estos algoritmos fueron esbozados en la asignatura de sistemas inteligentes del máster.
- Reducción de los componentes electrónicos que hacen factible el desarrollo de sistemas empotrados (como pueden ser los PICs o los DSPs que han sido objeto de estudio en la asignatura de sistemas empotrados de este mismo máster) de propósito específico, que son necesarios para llevar a cabo el control de determinadas partes de un robot o vehículos autónomo.
- La revolución en la conversión analógico / digital y el descubrimiento del sensor CCD, que han propiciado que el proceso de adquisición de imágenes

se pueda llevar a cabo con fotografías digitales sobre las que se pueden aplicar técnicas de visión por ordenador. Dichas técnicas, que fueron vistas en la asignatura de visión por computador, sirven para clasificar imágenes, analizar formas, localizar regiones de interés, etc. Son por tanto la base formal sobre la que se sustenta este trabajo.

1.2 – Descripción del Problema

En el marco referencial temporal en el que se circunscribe este proyecto (primer semestre del año 2014), la Universidad Complutense de Madrid está llevando a cabo un proyecto de investigación en el ámbito de los vehículos marinos y aéreos no tripulados. Cabe reseñar, que una de las problemáticas detectadas hasta la fecha y que todavía no había sido objeto profundo de estudio, es la relacionada con la maniobra de aproximación que permita a un vehículo autónomo llegar hasta su estación base de una forma segura, rápida y precisa.

A pesar de que el sistema de navegación empleado por los vehículos autónomos se fundamenta esencialmente en un sistema satelital de posicionamiento global (GPS), se requiere el uso de una plataforma de aproximación que sirva para conocer la orientación y posición relativa entre el vehículo y su base, para así poder llegar hasta el lugar exacto donde deba aterrizar / atracar. De esta forma, el receptor GPS permite al vehículo acercarse hasta los alrededores en los que se encuentra ubicada la plataforma de aproximación, y la propia plataforma es la que le sirve para poder orientarse y determinar el lugar exacto en el que tiene que posicionarse. Para lograr este objetivo, la plataforma se vale de una figura geométrica patentada que se encuentra dibujada sobre la superficie de la misma. Esta figura, que ha sido facilitada por el personal docente de la UCM -concretamente por el director de este TFM-, es la siguiente que aparece reflejada a continuación:

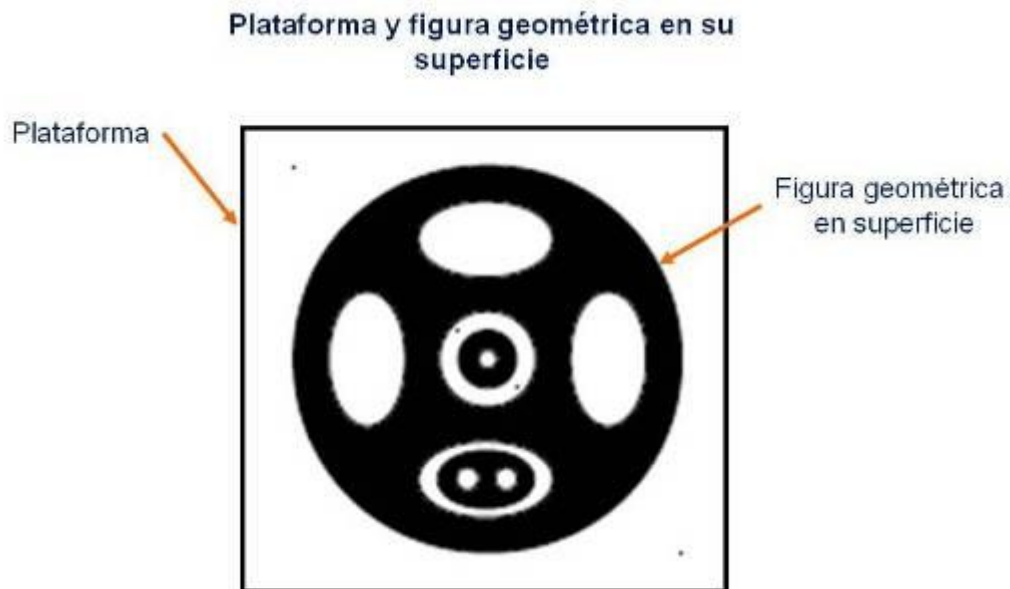


Figura 1-1 – Plataforma de aproximación con figura geométrica

A partir de un análisis previo que se ha hecho sobre la figura geométrica que se acaba de mostrar, pueden postularse las siguientes premisas que se indican a continuación:

1. La morfología de la figura atiende a una estructura que sirve para garantizar la imposibilidad de que un entorno agreste y natural (un ecosistema de la naturaleza), genere aleatoriamente un elemento que por su idiosincrasia atienda a unas características morfológicas análogas. Esto es de vital importancia cuando se trabaja con imágenes de exterior, porque permite garantizar la exclusividad y unicidad de la imagen, en tanto en cuanto, se presupone que no existen elementos naturales que atiendan a tal morfología.
2. El dibujo está formado por un compendio de figuras geométricas (círculos y elipses eminentemente) combinadas de tal forma que sirven para tratar de garantizar que si una parte queda ciega al llevar a cabo la adquisición de la imagen por el motivo que fuere, todavía sigan existiendo suficientes formas peculiares que permitan su identificación. Esto ha permitido desarrollar una técnica basada en probabilidades que se sustenta a través del análisis individualizado y colectivo de cada una de las subfiguras.

1.2.1 – Problema con la adquisición

A partir de la segunda premisa que se acaba de postular, resulta factible intuir la dificultad inherente que conlleva un proceso de adquisición de imágenes (aéreas o marinas) en un entorno exterior, donde resulta difícil controlar parámetros tan relevantes como el nivel de luminosidad, el grado de inclinación sobre la imagen, la distancia, etcétera.

Si a esto le añadimos que la autonomía del vehículo es limitada, y que por tanto se dispone de un tiempo determinado (en un entorno aéreo aún es más importante ya que el UAV puede caer al vacío) para tomar una imagen de la plataforma que tenga un nivel de calidad suficientemente bueno como para identificar la figura mediante el uso de técnicas de visión por computador, el problema no tiene una solución trivial.

Por consiguiente, el objetivo de este subapartado no es otro, que tratar de poner de manifiesto dos imágenes reales, que han presentado problemas durante la fase de adquisición, debido a que se introducen puntos ciegos que dificultan el proceso de análisis e identificación de la imagen:



(a)



(b)

Figura 1-2 – Adquisición (a) Problema adquisición 1; (b) Problema adquisición 2

Tal y como se puede observar, ambas imágenes presentan serios problemas de calidad como consecuencia de reflejos imprevistos de la luz sobre la superficie de la plataforma que se fotografió de forma manual.

1.3 – Objetivos

El objetivo fundamental que se persigue con este trabajo, es desarrollar un algoritmo mediante el uso de técnicas de visión por computador, que sirva para identificar en el mayor número de situaciones posible, la figura que hay dibujada sobre la superficie de la plataforma de aproximación, para hacerle saber al vehículo autónomo que esa es su estación base y que por tanto tiene que llegar hasta ella.

Para alcanzar este objetivo, resulta fundamental extraer un número ingente de imágenes en diferentes perspectivas y con diferentes condiciones lumínicas, que traten de reproducir fielmente situaciones reales que pueden suceder y a las que hay que hacerle frente. El nivel de eficacia del algoritmo depende del grado de versatilidad alcanzado ante diversas situaciones que se puedan dar en un entorno de exteriores.

Además, la eficiencia computacional también juega un papel fundamental, ya que en un entorno productivo de tiempo real donde la autonomía operativa del aparato juega un aspecto crítico, dar respuesta en tiempos aceptables es una obligación. Por este motivo, se ha ahondado bastante en este sentido, ya que si la eficiencia es buena y se reducen considerablemente los tiempos de ejecución, se pueden realizar muchas pruebas con diferentes configuraciones (escalado, umbralización, etcétera) antes de tener que llevar a cabo una nueva maniobra para afrontar un proceso de adquisición adicional.

1.4 – Organización del Trabajo

El trabajo que se presenta en esta memoria, se organiza esencialmente en torno a seis capítulos principales. En el capítulo 1, como se puede observar, se ha llevado a cabo una breve introducción, una descripción del problema y una determinación de objetivos que se persiguen con el mismo. En el capítulo 2, se hace un estudio del estado del arte en relación a los principales métodos y patentes que se han llevado a cabo para resolver problemas similares al que se propone en esta memoria.

En el capítulo 3, se presenta la solución propuesta de una manera formal, así como una serie de diagramas y una descripción exhaustiva de los principales conceptos y técnicas empleadas durante su desarrollo. El capítulo 4 juega un papel clave, ya que se dan a conocer cuáles han sido los principales problemas a los que hubo que hacer frente, así como las decisiones adoptadas para poder solucionarlos.; es por tanto un cuaderno de bitácora en el que se detallan los principales inconvenientes con los que nos hemos encontrado.

Por medio del capítulo 5, se muestran los resultados obtenidos como consecuencia de aplicar el algoritmo propuesto a fotografías reales, además de mostrar una serie de gráficas de resultados e imágenes que han sido depuradas por nuestro sistema, a fin de demostrar bajo qué tipo de condiciones heterogéneas es capaz de trabajar el sistema. Finalmente en el capítulo 6, se lleva a cabo una síntesis que sirve para recoger las principales conclusiones obtenidas y las líneas de actuación futura. Por último, en los anexos se muestran las tablas que contienen los resultados de las pruebas, así como todo el código fuente de MATLAB que ha sido desarrollado.

Capítulo 2: Estado del Arte

Por medio de este capítulo se hace una breve descripción en relación a los principales métodos y patentes que se han llevado a cabo para resolver problemas similares al que se propone en esta memoria.

2.1 – Visión General

El aterrizaje de un vehículo aéreo, resulta ser en general, una de las fases más críticas que se producen en el transcurso de un vuelo. En los vehículos tripulados el piloto se vale, además de la instrumentación de a bordo y su experiencia previa, de los sentidos del oído (equilibrio) y de la vista para manejar la máquina. Además, cabe resaltar, que en la última fase de la maniobra es cuando resulta necesario utilizar toda la información para obtener un alto grado de precisión que garantice el posado y la parada del vehículo sin que éste y el piloto sufran daño alguno.

De manera similar, el atraque de un buque en un muelle necesita de la información visual del piloto para su realización. Por ello, en los vehículos autónomos el uso de sistemas de visión artificial ha sido una elección natural para intentar dotar al vehículo de uno de los sentidos usados por los humanos en la misma tarea. Pero disponer de un sistema que capte información visual del entorno no es suficiente para asegurar el éxito de la tarea, y resulta necesario que la información visual pueda ser usada a un ritmo y con una precisión suficiente.

Después de haber llevado a cabo un proceso exhaustivo de búsqueda en el estado del arte de este tipo de plataformas de aproximación, se ha observado que existen diversas formas y tecnologías ya patentadas que tratan de establecer una solución al problema descrito en el apartado 1.2 – Descripción del problema de este mismo documento. Muchas de ellas se basan en el uso de mecanismos que captan la radiación del espectro visible, y otras, que se basan en la utilización de dispositivos que captan la radiación infrarroja debida a la radiación térmica.

No obstante, en cualquiera de ambos casos, resulta necesario extraer información de las imágenes proporcionadas por las cámaras para tener cierto grado de conocimiento de la posición y orientación relativa del vehículo con respecto al punto de aterrizaje / despegue o atraque / partida. En este sentido, cabe reseñar, que nuestra técnica se fundamenta en el uso de la primera opción, ya que al utilizar una cámara que captura imágenes con radiación visible se reducen los costes puesto que éstas son más baratas que las que usan la radiación infrarroja (por ejemplo).

2.1.1 – Casos de Estudio

A partir de la búsqueda efectuada en las diferentes fuentes de información bibliográficas disponibles, incluyendo Internet, se ponen de manifiesto los ejemplos siguientes que se basan esencialmente en el uso de figuras geométricas. Dichas técnicas se sintetizan a continuación:

- Xu y col. [4]: Se basa en el uso de una figura geométrica que tiene una estructura morfológica en forma de T que está situada sobre la superficie de una plataforma, de modo que mediante la usanza de técnicas de procesamiento de imágenes, se determina el ángulo que proporciona la orientación relativa de la figura y el UAV. Utilizan la radiación infrarroja para conseguir un gran contraste de temperatura entre la figura usada para

identificar el lugar del aterrizaje y el entorno. Este método utiliza la forma real de la meta cooperativa en el suelo y el principio de la desaparición de la línea, generándose un error que es lo suficientemente pequeño cuando el UAV está cerca de la pista de aterrizaje, lo que permite cumplir con los requisitos básicos de aterrizaje autónomo del UAV. Las técnicas de visión por computador empleadas son: reconocimiento de patrones usando momentos invariantes afines, segmentación y extracción de bordes.

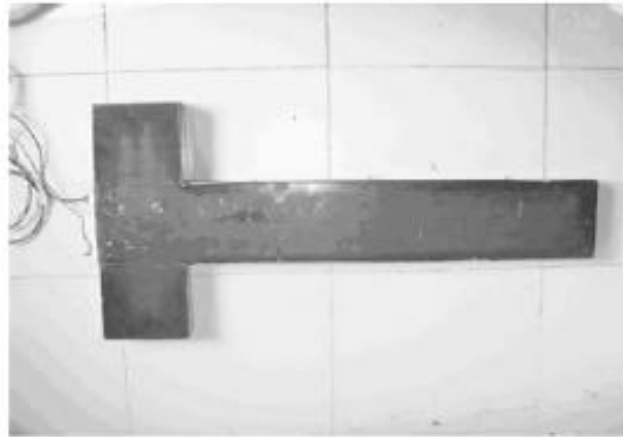


Figura 2-1 – Plataforma de aproximación de Guili Xu

- Lange y col. [5]: Se basa en una figura consistente en varios anillos blancos concéntricos sobre un fondo negro, de tal modo que cada anillo posee una anchura específica para hacer posible su identificación y diferenciación de los demás. Esta diferencia entre el radio externo e interno del anillo aumenta según nos alejamos del centro de la figura. De este modo, se permite que, aún cuando no todos los anillos están dentro del campo visual, la parte de la plataforma visible siga siendo identificable (este caso se puede dar cuando el vehículo aéreo se encuentra a baja altitud). Mediante el procesamiento de la imagen de la figura 2-2 se determina la altura a la que se encuentra la cámara. El método desarrollado asume que el UAV siempre estará paralelo al suelo, por lo que no se preocupa de determinar los ángulos de inclinación / orientación a partir de la figura.



Figura 2-2 – Plataforma de aproximación de Sven Lange

- Saripalli y col. [6]: Al igual que las dos técnicas presentadas anteriormente, este método se fundamenta en el uso de una plataforma que también tiene serigrafiada en su superficie una figura geométrica. Se trata de un dibujo con forma de H que se encuentra rotulado de color blanco. Se corresponde por tanto, con la típica marca que se encuentra dibujada sobre la superficie de los helipuertos. De la figura captada por la cámara se extrae la posición del centro de gravedad de la imagen en un plano cartesiano bidimensional OXY. También cabe reseñar, que a partir de la imagen se obtiene el ángulo de orientación de la figura con respecto al UAV. Para extraer la información de la imagen se aplican las siguientes técnicas de visión por computador: umbralización, filtrado, segmentación y etiquetado de componentes conexas.



Figura 2-3 – Plataforma de aproximación de Srikanth Saripalli

- Sharp y col. [7]: Este método se fundamenta en el uso de una figura que está construida a partir de seis cuadrados blancos, de los cuales cinco de ellos tienen el mismo tamaño. Estos cuadrados están incluidos en otro cuadrado de mayor tamaño y de color negro, que a su vez se encuentra incluido en otro cuadrado aún mayor que es de color blanco. Esta figura es adquirida desde el UAV a través de una cámara digital que lleva incorporada, para acto seguido, ser procesada mediante el uso de las siguientes técnicas de visión por computador entre las que podemos destacar: la umbralización, la segmentación, etiquetado de componentes conexas y detección de puntos de interés (esquinas). A partir del análisis se halla la posición y orientación relativa de la figura y el vehículo autónomo (UAV), si bien el procedimiento de procesado de información es palmariamente más complejo que el que constituye el objeto de la invención.

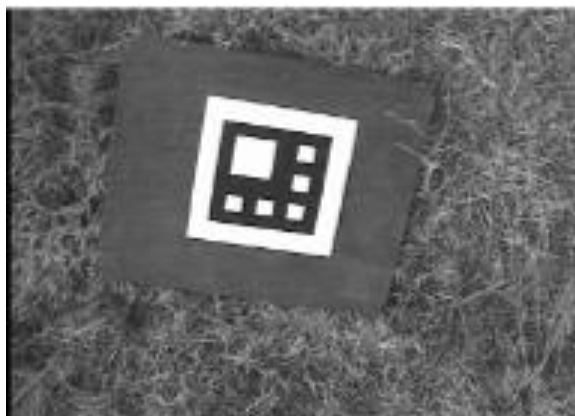


Figura 2-4 – Plataforma de aproximación de Cory S. Sharp

Tal y como se puede observar, todas estas técnicas tienen varios aspectos en común, ya que en todos los casos se utiliza una figura geométrica dibujada sobre la superficie de una plataforma de aproximación. Además, dicha figura es captada por el vehículo autónomo mediante un proceso de adquisición que genera una imagen, que sirve de entrada para el proceso de análisis e identificación. Además, en todos los casos, dicho proceso se realiza mediante el uso de técnicas de visión por computador. En este sentido, si se disminuye el nivel de granularidad y por tanto nos centramos en aspectos más técnicos y concretos de la visión por computador, observaremos que en casi todos los ejemplos se emplean las siguientes técnicas: la umbralización, la segmentación y el etiquetado.

Sin embargo, es conveniente poner de manifiesto, que también existen métodos en los que no se usa una figura geométrica artificial como base para llevar a cabo el proceso de aproximación. Un claro ejemplo de este tipo de metodología la podemos observar en el método propuesto por el científico italiano *Andrea Cesetti* en el año 2009, en el que plantea una técnica de navegación y aterrizaje para UAVs en la que se emplean marcas o señales naturales tomadas a partir de una imagen aérea cuya adquisición es competencia del propio UAV:

- Cesetti y col. [8]: Este método se basa en el análisis de una imagen aérea que ha sido adquirida por el vehículo autónomo. A partir de dicha imagen aérea, y mediante el uso de técnicas de visión por computador, se obtienen varios puntos de referencia en el entorno para una posible identificación de un lugar para el aterrizaje. El sistema trata de encontrar marcas naturales en la escena de exteriores mediante el uso de algoritmos basados en un análisis de flujo óptico, a partir del cual la imagen se divide en cuadros más pequeños a los que se le aplica una técnica conocida como SIFT (Scale Invariant Feature Transform). Por consiguiente, no intenta hacer uso de patrones preestablecidos, sino que pretende hacer una búsqueda por comparación de determinados puntos que concuerdan con lugares que se encuentran cargados de antemano en una base de datos convencional. Este método requiere que la misión se lleve a cabo en un entorno parcialmente conocido, y los resultados arrojan que es un método robusto a las oclusiones y a las variaciones de luz.

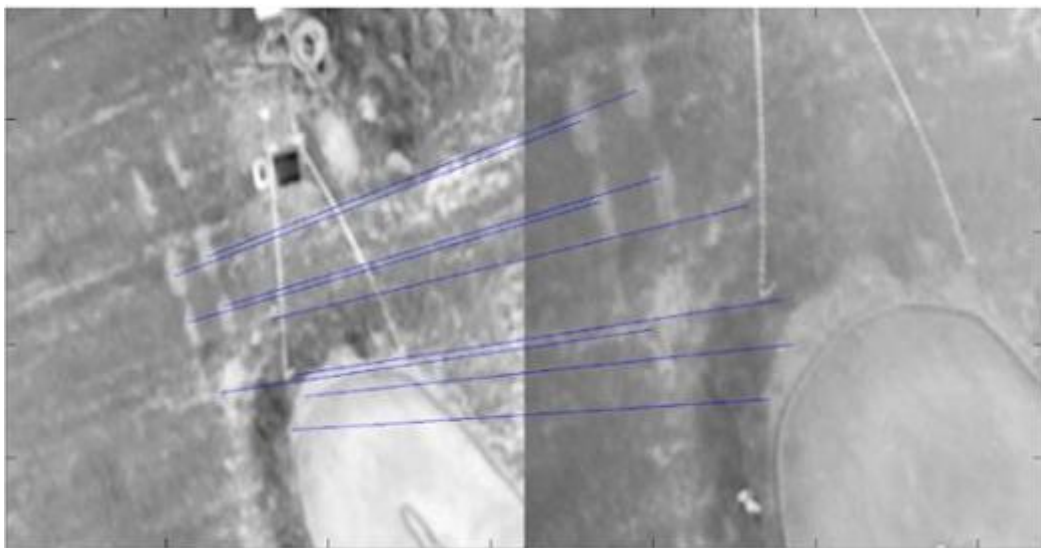


Figura 2-5 – Método de aproximación y aterrizaje de Andrea Cesetti.

A través de la imagen puede observarse, cómo el sistema detecta una coincidencia entre dos frames utilizando un punto de referencia natural para la localización y el

seguimiento. Finalmente, con el fin de aportar un caso de estudio real relacionado con el ámbito marino, y así dispensar de un mayor grado de heterogeneidad al estudio del estado del arte, se pone de manifiesto el siguiente método propuesto por un miembro del *IEEE Oceanic Engineering Society*:

- Lee y col. [9]: Presentan un método que describe un sistema de guiado automático para el atraque de un buque en el que se utilizan dos marcas de lona con forma rectangular que están pintadas de distintos colores (una verde y otra azul). El objetivo fundamental de ambas figuras es tratar de definir una línea de guía para el atraque del barco. Las marcas, que están situadas sobre una recta, y que se encuentran separadas en longitud y en altura, sirven para determinar la distancia del vehículo a las marcas y el rumbo respecto a la línea de guía. Para poder captar dichas marcas, el barco cuenta con una cámara situada en la proa, que sirve para llevar a cabo el proceso de adquisición de las imágenes. Una vez finalizada la parte relativa a la adquisición de fotos, las imágenes resultantes se procesan del siguiente modo:
 1. Se emplea un esquema de reconocimiento de colores para identificar las componentes relativas al tono (H) y la saturación (S).
 2. Ambas componentes se procesan utilizando un esquema de filtrado de ruido para detectar con precisión la ubicación de los dos objetivos.
 3. Se calculan los centros de gravedad de las dos marcas y la distancia de separación entre ambas para poder hallar cuál es la desviación correspondiente al ángulo del rumbo.
 4. El ángulo se suministra a un control de modelo interno (IMC) en forma de señal de error, para poder generar comandos de control apropiados que sirvan para dirigir el barco hacia la línea de guía.

2.1.2 – Patentes

Por último, y para acabar con este apartado correspondiente al segundo capítulo de esta memoria, se ha decidido poner de manifiesto, algunas de las patentes que se emplean en este campo:

- a. CN101109640 [10]: Esta patente presentada en julio de 2006, se basa en el uso de una cámara que sirve para extraer características de la pista de aterrizaje, y a partir de dicha información, y mediante el uso de un análisis iterativo, encontrar el modo de llevar a cabo el aterrizaje. La patente tiene dos partes: un sistema software que proporciona un análisis de visión por computador, la fusión de la información y una estimación del estado; y una disposición hardware que comprende la característica de la pista, el subsistema de sensores de a bordo, y un subsistema de información de la fusión. Considera dos formas geométricas en la pista de aterrizaje, un rombo y un triángulo, situados de forma longitudinal a lo largo de la pista. Usan la información extraída del análisis de la figura con la información de otros sensores con el fin de obtener mayor precisión en la posición relativa con respecto a la pista de aterrizaje. Las figuras se encuentra separadas, y no están pensadas para vehículos de despegue y aterrizaje vertical.
- b. US20080215204 [11]: Esta patente, que fue presentada en diciembre del año 2007, provee métodos, aparatos y sistemas de visión sintética mejorada

y de fusión de datos de sensores múltiples, que sirven para tratar de mejorar la capacidad operativa de los vehículos aéreos no tripulados (UAVs). Básicamente, lo que se plantea es un sistema de comunicación para permitir el intercambio de información entre el UAV y un centro de control. El intercambio de información, se fundamenta de manera esencial en el uso de un sistema de visión por computador que se encarga de captar y analizar los datos del entorno sobre el que se lleva a cabo la misión.

- c. [US20090306840 \[12\]](#): Esta patente, que fue presentada en abril de 2009, propone un sistema automático de visión por computador que permita el aterrizaje de vehículos aéreos no tripulados. Se toma como eje central del método una figura en forma de cruz que contiene un rectángulo en su eje mayor. A partir de la imagen captada por el vehículo en pleno vuelo, se determinan los ángulos de orientación, o guiñada, y la elevación del UAV con respecto al eje central de la figura. Por consiguiente, el flujo lógico del proceso es el siguiente que se sintetiza a continuación:

1. Se captura una imagen en la dirección de movimiento del UAV.
2. Se analiza la imagen para determinar si es un posible objetivo.
3. Se evalúa el objetivo para determinar el ángulo de planeo y elevación.
4. Se corrige la altura, posición y orientación del UAV.

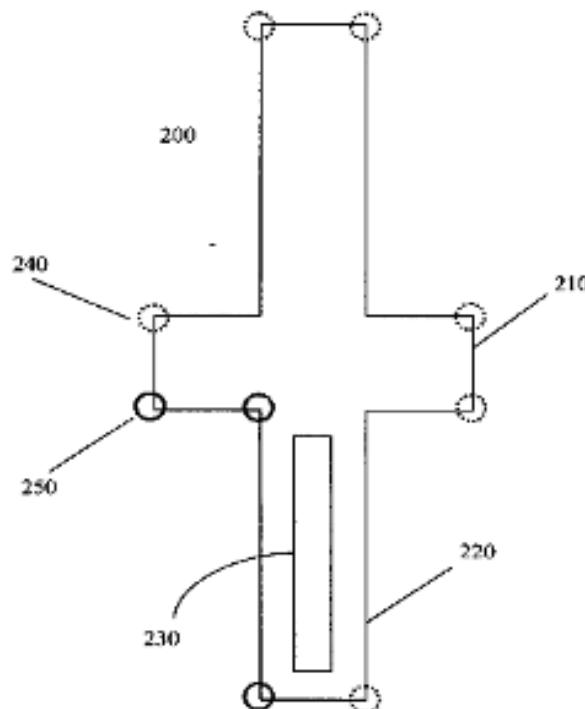


Figura 2-6 – Plataforma aterrizaje automático (patente US20090306840)

- d. [US20070093945 \[13\]](#): Esta patente norteamericana, cuya fecha de presentación data de octubre del año 2006, proporciona un sistema de procesamiento para vehículos aéreos no tripulados, que combina la tecnología de visión por computador con un método de compensación de movimiento. Por tal motivo, el sistema emplea dos tipos de sensores

contrapuestos. Por un lado, emplea como mínimo un sensor de visión que sirve para captar la información del entorno, y por otro, un sensor de movimiento que se encargue de registrar los principales datos cinemáticos del UAV. Internamente, el sistema combina las entradas de ambos sensores con el fin de aumentar las capacidades de estimación basada en la visión por computador, tanto en el ámbito de la navegación, como en el de la estabilización. Dicha patente, proporciona un sistema capaz de explorar los entornos desconocidos hasta la fecha, siendo para sobrevivir a ambientes desordenados o con gran variedad de obstáculos. A modo de resumen podemos señalar, que la patente describe una estructura organizativa de elementos, que le confieren a un UAV dotado de un sistema de visión, la capacidad de navegar basándose al menos en parte, en datos de imágenes corregidas con información proporcionada por otros sensores tradicionales, lo que lo hace apto para adaptarse a entornos dinámicos.

- e. KR1020070058957 [14]: Patente Coreana cuya fecha de publicación data de junio del año 2008, y que sirve para proporcionar un método de navegación automático para un UAV a partir de un símbolo situado en una zona con una red de recuperación. Dicho sistema reconoce la imagen, mide la distancia mediante un puntero láser, y dirige la información a un puesto de control externo. El sistema pasa por la siguientes etapas que se describen de forma esquemática a continuación:
1. Fotografía en tiempo real mediante el uso de una cámara frontal incorporada en el UAV; el símbolo de aterrizaje se encuentra ubicado en una red de recuperación.
 2. Tras el proceso de adquisición se lleva a cabo un proceso de análisis de la imagen mediante el uso de técnicas de visión por computador que son perpetradas por un procesador de imagen.
 3. Mediante el uso de un puntero láser se efectúa una medición en tiempo real de la distancia que existe entre el símbolo de aterrizaje y el vehículo aéreo no tripulado.
 4. Se transmite en tiempo real toda la información relativa al símbolo y a la distancia existente entre éste y el UAV, así como los comandos de control necesarios para llevar a cabo el aterrizaje.
- f. US5525883A [15]: Esta patente norteamericana, que fue presentada en julio del año 1994, describe un método que permite determinar la orientación y la posición de un vehículo autónomo mediante el uso de puntos de referencia. Dichos puntos de referencia son tableros de ajedrez que contienen células rectangulares de colores contrastantes (blanco y negro) dispuestos de tal forma que permiten crear un patrón pseudo-aleatorio. El vehículo autónomo lleva incorporado un sistema de reconocimiento basado en una cámara y un ordenador, que mediante el uso de técnicas de visión por computador y un mapa simbólico, que está almacenado en la unidad de reconocimiento y medición, sirve para describir la posición, la orientación y las dimensiones de cada patrón pseudo-aleatorio del tablero. La unidad de reconocimiento y medición procesa la imagen del tablero con patrones pseudo-aleatorios para identificar el punto de referencia en el que se muestra el patrón, y así tratar de calcular la posición y la orientación del vehículo autónomo. En la imagen correspondiente a la figura 2-7, se muestra una imagen del tablero.

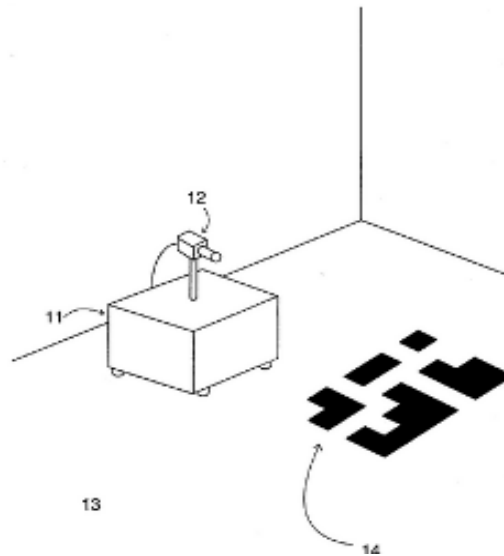


Figura 2-7 – Tablero pseudo-aleatorio de la patente US5525883A

2.2 – Adquisición de imágenes

A pesar de que la parte que más nos interesa en esta memoria es la relativa al software, y por ende la relacionada con los algoritmos propios de la visión por computador, resulta conveniente hacer una breve mención en lo que concierne a los principales elementos hardware que intervienen en un proceso de adquisición de imágenes. Dado que los resultados obtenidos dependen en gran medida de las condiciones en las que se lleve a cabo el proceso de adquisición de imágenes, se ha estimado oportuno hacer una descripción exhaustiva de los principales parámetros empleados a la hora de obtener las imágenes.

2.2.1 – Cámara

En una cámara, uno de los elementos más importantes que entra en juego a la hora de obtener imágenes, es el relacionado con el conjunto de lentes que de forma común se conocen como óptica. Este elemento engloba mecanismos tan relevantes como el obturador, el diafragma y el objetivo, que influyen en conceptos como pueden ser la distancia focal o la profundidad de campo. Sin embargo, en las cámaras digitales, a mayores de la óptica, hay que tener en cuenta otro elemento de diferente naturaleza, pero sumamente importante como es el sensor CCD.

El sensor CCD, se trata de una unidad electrónica que se expone a la luz durante un tiempo determinado (denominado tiempo de integración), tras el cual los fotones que han sido almacenados en cada píxel son transferidos de forma ordenada a una etapa de salida. Por lo tanto, nos permite evaluar de forma cuantitativa la luz que ha incidido sobre cada uno de sus píxeles, puesto que a cada uno de ellos se les hace corresponder una tensión eléctrica proporcional a la cantidad de fotones que han sido almacenados.

2.2.2 – Conversores A/D

Posteriormente, entra en juego el conversor A/D que se encargará de traducir las tensiones eléctricas correspondientes a cada píxel, en un valor digital en código

binario, una vez se haya realizado el barrido de lectura analógico contra el CCD. El parámetro más importante del convertor A/D es el número de bits que posee, es decir, cada tensión eléctrica será convertida en un valor entre cero y $2^n - 1$. Lo que significa, que a mayor número de bits, con mayor precisión se puede cuantificar el número de fotones que han incidido en un píxel.

- **Resolución:** En el ámbito del video, la resolución del convertor hace alusión a la profundidad del color, es decir, a la cantidad de bits que se van a utilizar para representar el color de cada píxel. Lo habitual es emplear uno por cada canal empleado (RGB normalmente). Por ejemplo, un convertor de 8 bits representa $2^8 = 256$ niveles de luminosidad, lo que quiere decir que si tuviéramos 3 (uno para canal en RGB) podríamos procesar imágenes de 24 bits de profundidad de color.
- **Frecuencia de muestreo:** Este parámetro es muy relevante, ya que define cuál es el número de muestras por unidad de tiempo que se toman de una señal continua para producir una señal discreta. Como unidad de frecuencia que es, utilizaremos Hz (ciclos por segundo) para expresar su valor, a partir del cual podremos saber cuál es la cantidad de muestras por segundo que resultan necesarias para hacer el muestreo (*sampling*).

El funcionamiento del CCD, en conjunción con los convertores A/D queda ilustrado por medio de la imagen mostrada en la figura 2-8:

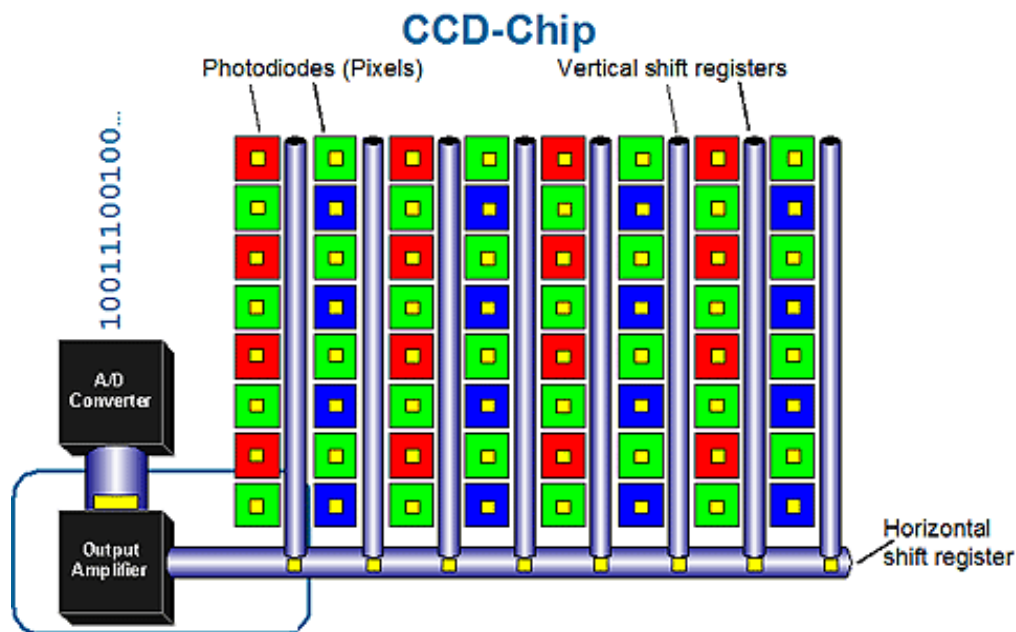


Figura 2-8 – Sensor CCD y Convertores A/D

Fuente: <http://www.xatakafoto.com/camaras/sensores-con-tecnologia-ccd-vs-cmos>

Cabe reseñar, que los convertores A/D pueden estar integrados dentro de la propia circuitería de la cámara, o encontrarse externamente. Para la interconexión suelen emplearse buses estándar como pueden ser: I²C, EMAC, UPP, etcétera, y para el control y el barrido que provoca el avance de los sucesivos píxeles que constituyen la matriz CCD, puede ser controlado por un DSP, aunque lo común sea realizarlo a través de un FPGA durante el procesamiento de las mismas.

2.2.3 – Unidad de Procesamiento

En lo que concierne al procesamiento de las imágenes, puede valorarse el uso de un DSP de propósito general o específico para video (en este último caso los DSP de la serie *DaVinci* del fabricante *Texas Instrument* pueden ser una buena elección) o emplear un ordenador de propósito general (PC de sobremesa, servidor, etcétera). En el caso de emplear un DSP (*Digital Signal Processor*) tendremos la ventaja de que su coste suele ser bastante inferior al de un equipo de propósito general, y que su hardware está diseñado para realizar operaciones numéricas a alta velocidad, lo que lo hace ser muy competitivo en temas de edición y compresión de video. Sea como fuere, el dimensionamiento del hardware estará vinculado al algoritmo que finalmente se emplee para acometer la identificación de la plataforma. En este sentido, una vez implementada la solución, habría que estudiar entre otras cosas:

- Tamaño de las imágenes a procesar y tipo de aritmética a utilizar: el tamaño de las imágenes juegan un papel fundamental, ya que a mayor resolución, mayor número de operaciones a considerar. También es necesario concretar si el tipo de operaciones predominante será en coma fija o flotante.
- Tipo de funciones a utilizar: por ejemplo, si se empleara la transformada del coseno discreta ó DCT, que está basada en la transformada de Fourier, habría que analizar el coste asociado a las operaciones MAC (producto de dos números que se añade a un acumulador; un ejemplo de operación MAC es la siguiente: $a \leftarrow a + (b * c)$).
- Numero de instrucciones y operaciones: otro aspecto muy interesante para valorar el hardware es el número de instrucciones por segundo (MIPS) y el número de operaciones en coma flotante (FLOPS). Este tema se describe brevemente en el estudio de eficiencia computacional que se muestra más adelante en el apartado *4.4 – Problema de Eficiencia Computacional*.

2.2.4 – Condiciones de Adquisición de Imágenes

En un entorno real, el proceso de adquisición de imágenes debe llevarse a cabo con una cámara que se encuentre ubicada en la parte exterior del vehículo autónomo que sea objeto de estudio en dicho momento. Sin embargo, por problemas técnicos y de localización sobre todo (el laboratorio de la universidad está en otra ciudad distinta de la que reside el autor de este TFM), no ha sido posible hacer las pruebas con un vehículo autónomo real.

Por consiguiente, en todo lo que respecta al proceso de adquisición de imágenes para este trabajo, deben tenerse en cuenta las siguientes premisas que se indican a continuación, sin perjuicio de que los detalles más profundos en relación a todas las imágenes que se han utilizadas para testear el sistema. Dicha información se encuentra recogida en la tabla *A.I* que está en el *Anexo I – Tabla de Imágenes*:

- La figura geométrica que es objeto de estudio ha sido proporcionada electrónicamente por el personal docente de la UCM. No obstante, con el fin de materializar la imagen y poder realizar fotografías de la misma tratando de emular un entorno real, se ha impreso dicha imagen en un papel del tipo A3 (297 mm × 420 m), que ha servido para emular las funciones de la plataforma de aproximación.
- La profundidad de color que se ha empleado es de 3 bytes (24 bits), es decir, 8 bits por cada canal RGB y el formato de compresión de todas las imágenes

está basado en el estándar JPEG. Además, con el fin de poder averiguar cuál es el comportamiento del software dependiendo de la calidad de la cámara empleada, se ha decidido obtener las imágenes de entrada mediante el uso de dos dispositivos diferentes:

- a. Cámara integrada del teléfono móvil *Samsung Galaxy Ace GT-S5830*.
 - b. Cámara *Canon PowerShot A 590 IS*.
- En lo que se refiere a la distancia entre el objetivo de la cámara y la figura geométrica, se ha decidido utilizando diferentes valores con el fin de buscar condiciones de toda índole (principalmente difíciles). Para más información, se puede consultar la sección de resultados y el detalle de cada imagen que aparece reflejada en la tabla de Resultados (A.II) que forma parte del *Anexo II* del presente documento.
 - En lo referente al ángulo de inclinación con respecto al eje de ordenadas o eje OY (perpendicular al suelo que suele estar en paralelo al eje de abscisas o eje OX), que sigue la línea recta imaginaria que va desde el objetivo de la cámara hasta la figura, se han empleado diversos tipos de inclinaciones para provocar reflejos que introduzcan partes ciegas en las imágenes. Un ejemplo de ello, se ha podido observar por medio de la imagen 1-2 (b).
 - El tamaño de las imágenes de entrada juega un papel esencial, ya que a mayor distancia (menor valor en el ratio área figura / área total) más nivel de resolución necesita el software. Dado que las cámaras que suelen montar estos tipos de aparatos no poseen una resolución muy elevada, se decidió obtener las imágenes con una resolución de 4 Mpx, es decir, con unas dimensiones de 2560 x 1944. No obstante, el algoritmo implementa una funcionalidad de escalado, que es totalmente parametrizable a través de la sección de configuración del sistema.

Capítulo 3: Solución Propuesta

Por medio de este capítulo se presenta una exposición formal de la solución propuesta, así como una descripción exhaustiva de los principales conceptos y técnicas empleadas en la elaboración de este Trabajo Final de Máster.

3.1 – Visión por Computador

Para llevar a cabo la solución a este problema, se han empleado técnicas y modelos propios del área de la visión por computador, por lo que resulta conveniente antes de entrar en materia, hacer una breve descripción e introducción acerca de qué es en realidad la visión por computador, en qué consiste, y qué características tiene. Pues bien, a pesar de que existen variedad de definiciones e interpretaciones que tratan de describir en qué consiste realmente la visión por computador, se puede decir que es el conjunto de técnicas que permiten diseñar e implementar algoritmos que una vez codificados y procesados por un ordenador, permiten a éste llegar a entender y comprender un escena.

La gran mayoría de las técnicas empleadas provienen del campo del Procesamiento de Señales y la Inteligencia Artificial, e incluyen actividades relacionadas con la: umbralización, segmentación, descripción, clasificación, etc. Tal y como se acaba de comentar, se debe tener en consideración que la visión por computador incluye una gran diversidad de técnicas, algunas de ellas relacionadas intrínsecamente con el procesamiento de imágenes y la extracción de patrones; otras más significativas tratan de conseguir una descripción geométrica útil, es decir, extraer la forma, el perímetro, y el área para realizar modelos geométricos; otras están íntimamente ligadas con el análisis de los colores y texturas, etcétera.

3.1.1 – Imagen digital

Un aspecto que tienen en común todas estas técnicas, es que toman como materia prima o punto de partida para la interpretación una *imagen digital*. Pues bien, una imagen digital se puede entender por un lado, como una representación discreta en donde hay valores en puntos que están regularmente muestreados, y también, como una representación cuantificada donde cada valor es representado por un número entero. Las imágenes constituyen la materia prima de una ciencia que se esfuerza en emular las capacidades perceptivas del hombre.

Formalmente, una imagen digital monocromática se puede representar como una función de intensidad luminosa bidimensional $f(x,y)$, donde el valor $f(x,y)$ nos da la intensidad luminosa de la imagen en el punto (x,y) , teniendo en cuenta que dicho nivel de intensidad es directamente proporcional al nivel de gris (brillo) de la imagen en ese punto. Esta función ha sido discretizada en ambas coordenadas espaciales (x,y) y en el nivel de gris. Se considera que el nivel de brillo mínimo equivale al negro, el máximo al blanco, y que los valores intermedios son distintos niveles de gris que varían del negro al blanco. Dos de sus características básicas son las siguientes:

- **Resolución:** número de puntos por unidad de distancia que tiene la imagen, de tal forma que, a mayor número de puntos por centímetro mayor resolución y más definida estará la imagen. Su cálculo viene determinado

por la multiplicación del número de filas por el de columnas, teniendo en cuenta que cada elemento $f(x,y)$ que compone la imagen se denomina *píxel*.

- Profundidad de color: también denominada como resolución en intensidad, hace referencia al número de bits usados para medir la intensidad una vez cuantizada la imagen, es decir, el número de niveles de gris o número de colores que empleamos para representar la imagen.

3.1.2 – Análisis preliminar de la figura de aproximación

Antes de continuar con los aspectos más técnicos de la solución propuesta, es el momento de realizar un breve análisis preliminar de la figura, que sirva además como punto de partida para etiquetar cada una de las subfiguras que forman parte de la imagen que se encuentra serigrafada sobre la plataforma de aproximación. Este proceso de etiquetado es muy importante, ya que la nomenclatura que se muestra en la figura 3-1, es la que se va a emplear cuando se desee hacer una alusión específica a cada uno de los cuatro círculos (C1, C2, C3 y C4) y las tres elipses (E1, E2 y E3) que forman parte de la plataforma de aproximación:

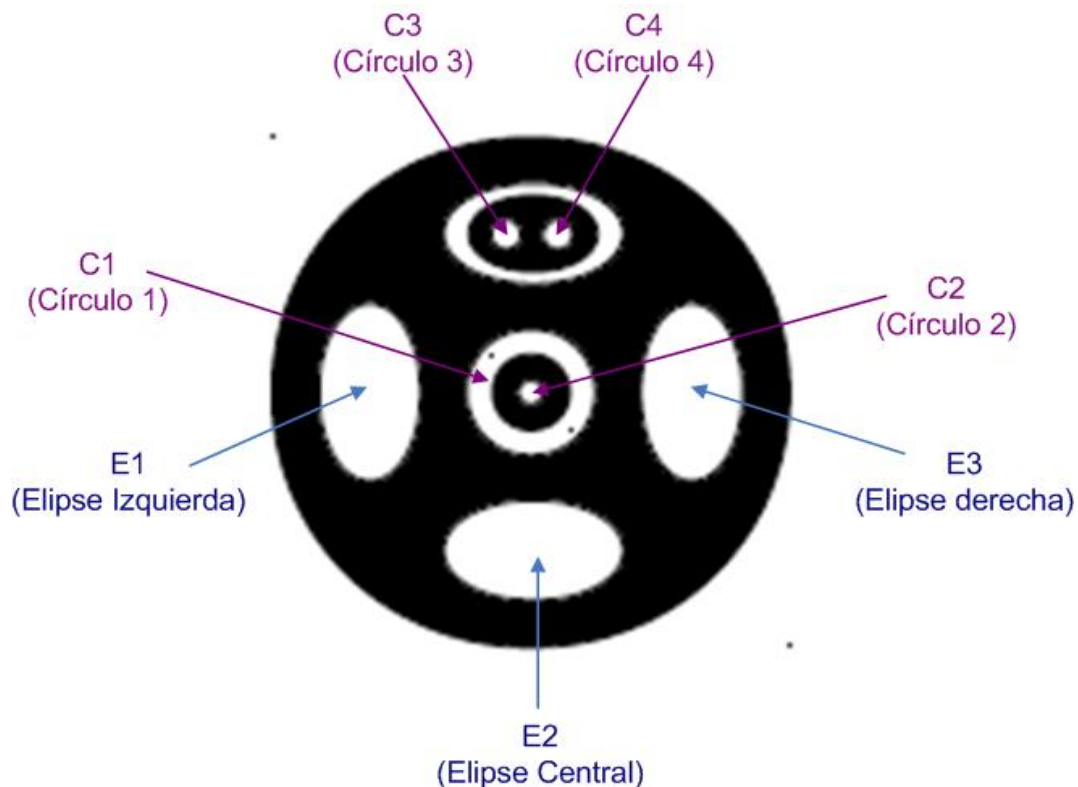


Figura 3-1 – Orientación final y etiquetado de las regiones de la figura

Otro aspecto muy relevante que se debe tener siempre en cuenta, es el relativo a la orientación de la figura. En este sentido hay que reseñar, que la orientación que se muestra en la figura 3-1, es la que se utilizará finalmente para llevar a cabo el proceso de identificación y reconocimiento. Esto quiere decir, que internamente el sistema hará los cálculos necesarios para averiguar cuál es la orientación original de la plataforma respecto a la figura 3-1. A partir de dicha información, el sistema girará la imagen tanto grados como sea necesario, para que el reconocimiento de la figura se realice siempre internamente con la orientación que muestra en la figura 3-1. Por lo tanto, esta imagen sirve como punto de referencia en este aspecto.

3.2 – Diagramas

Por medio del siguiente subapartado, se mostrarán los dos diagramas realizados, para tratar de explicar de una forma gráfica y lo más descriptiva posible, las principales áreas de actuación y decisiones de diseño que se han llevado a cabo a la hora de implementar la solución propuesta.

3.2.1 – Diagrama de bloques (granularidad gruesa)

En primer lugar, se ha decidido realizar un diagrama de bloques a través del cual se expone la secuencia de pasos que se ha seguido en la solución del problema, pero desde una perspectiva de alto nivel, es decir, con un nivel de granularidad grueso. Por consiguiente, a la hora de elaborar este diagrama, cabe reseñar, que se ha efectuado un proceso de abstracción para poner de manifiesto aquellos procesos más relevantes que se encontrarían dentro de la parte más elevada de la jerarquía de descomposición de tareas en una metodología WBS (*Work Breakdown Structure*), que se aplica en el ámbito de gestión y dirección de proyectos:

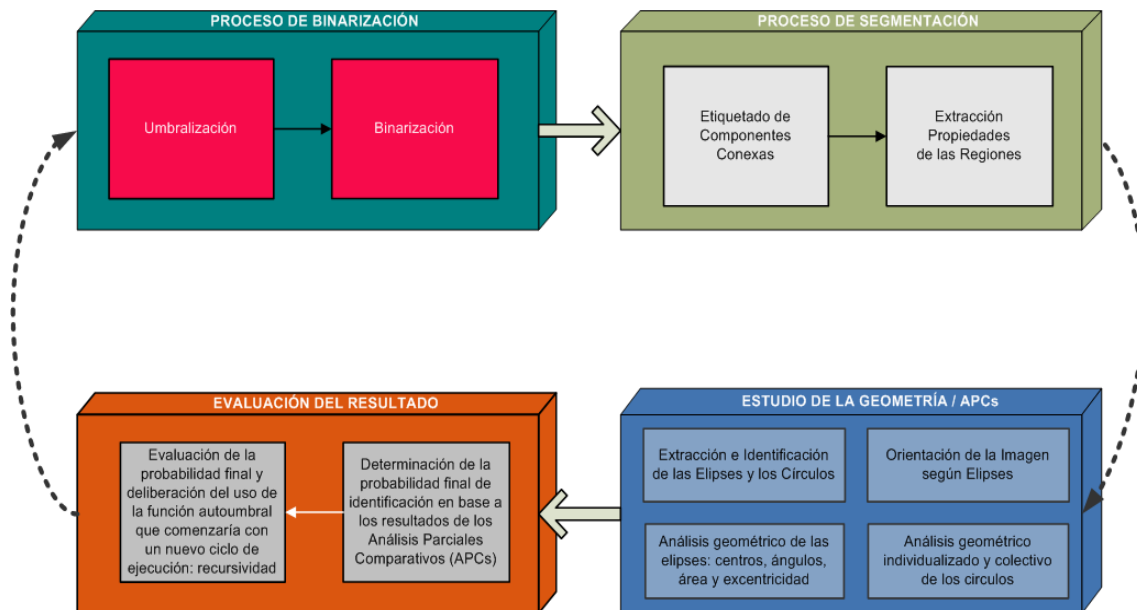


Figura 3-2 – Diagrama de bloques de la solución con granularidad gruesa

A partir del diagrama, se puede observar cómo la imagen original es binarizada y posteriormente segmentada para obtener las diferentes regiones entre las que se tienen que encontrar las elipses que forman parte de la figura que hay dibujada sobre la plataforma de aproximación. Posteriormente, se estudia con detalle el conjunto de la geometría de las elipses (ángulos, centros, etcétera), y dependiendo del número de aciertos se asigna una probabilidad final de identificación.

3.2.2 – Diagrama de flujo (granularidad fina)

Para proporcionar una explicación más clara y elocuente, que al mismo tiempo posea un mayor nivel de detalle técnico, se ha decidido hacer uso de un diagrama de flujo (*flowchart*, en inglés) en el que aparecen los principales procesos técnicos que se han llevado a cabo en la implementación del algoritmo. Dado su detalle, el nivel de granularidad es fino, tal y como se puede observar en la figura 3-3:

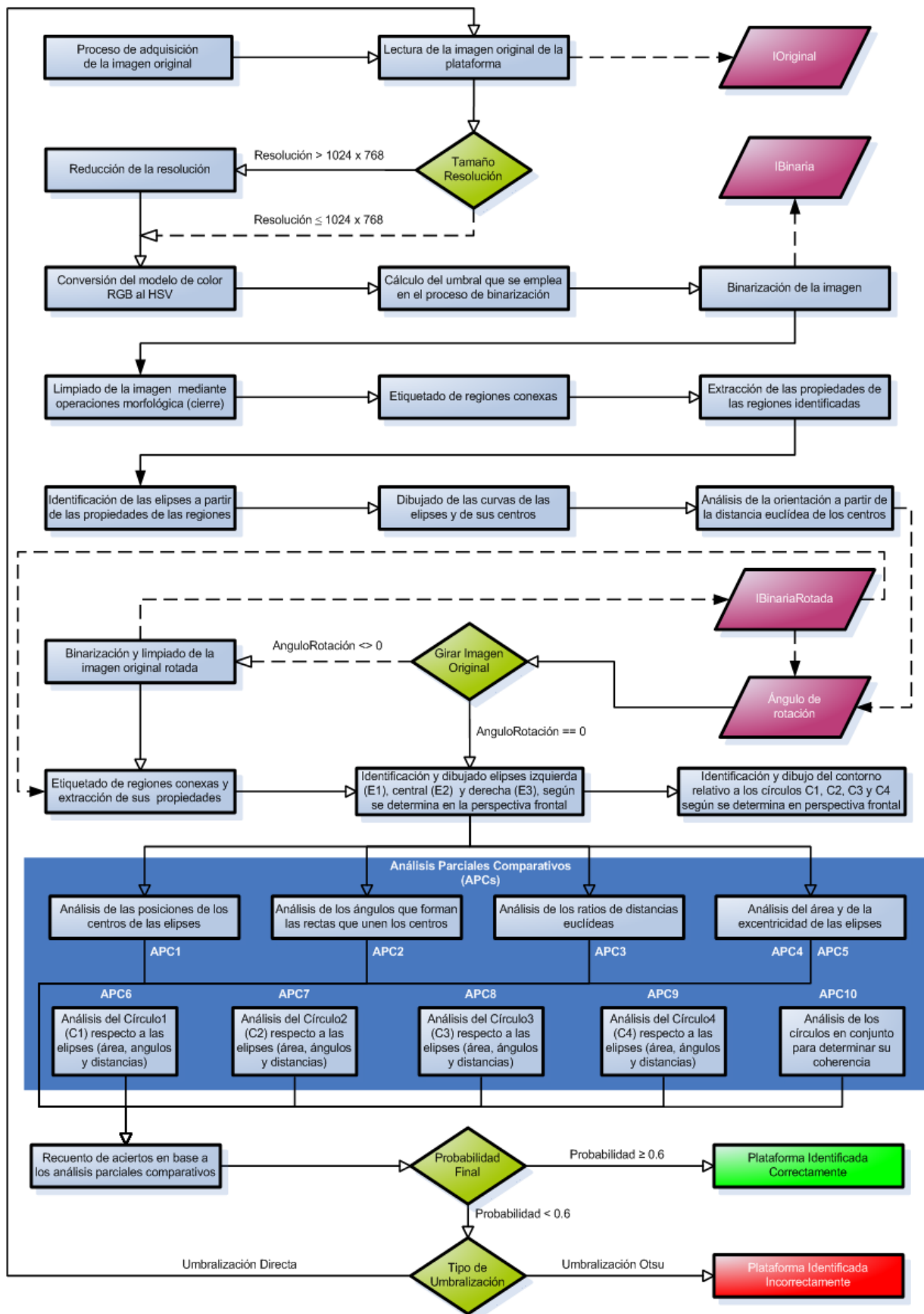


Figura 3-3 – Diagrama de flujo con la solución propuesta de granularidad fina

A pesar de que en el diagrama se refleja de forma exhaustiva cada uno de los procesos que se van ejecutando para llegar a la solución propuesta, se ha decidido incluir una breve descripción textual con el fin de dotar de mayor claridad a este documento que constituye una parte clave de la memoria descriptiva de este TFM:

1. Una vez el UAV se encuentre a punto de finalizar la misión y por ende, se esté aproximando a la base gracias a la información proporcionada por su GPS interno, se capta una imagen de la figura dibujada sobre la plataforma de aproximación, inicialmente ubicada en el suelo. Esta imagen se captará en pleno vuelo desde la cámara digital que lleva incorporada el vehículo autónomo, a una altura y con un grado de inclinación determinados por la unidad de control.
2. Si la resolución con la que se lleva a cabo el proceso de adquisición resulta demasiado elevada inicialmente, se realiza un proceso de escalado para reducir el coste computacional asociado a cada una de las operaciones que hay que efectuar. La idea es obtener imágenes con resolución media en caso de que se obtengan a una altura elevada (en torno a 4 mega píxeles), pero inicialmente el objetivo es tratar de llevar a cabo la identificación con resoluciones bajas (en torno a 1 mega píxel) mediante procesos de escalado.
3. Se lee la imagen que está representada en un espacio de color RGB, y se convierte a formato HSV. Acto seguido se procede a su binarización a partir del método de umbralización y del umbral que se hayan definido en la sección de configuración del sistema. Se utiliza la umbralización directa y la umbralización dinámica a través de una implementación del método Otsu [29,41], además de emplear un sistema autónomo basado en la recursividad que permite probar con las dos técnicas en el caso de que la primera no proporcione los resultados esperados.
4. Con la finalidad de tratar de eliminar pequeñas impurezas en el interior de la imagen, se realiza un proceso de limpieza de ésta mediante el uso del operador morfológico de apertura (*open*, en inglés), que servirá de base para tratar de eliminar todas aquellas regiones pequeñas y espúreas. Su explicación y detalles teóricos se ponen de manifiesto más adelante dentro de este mismo capítulo.
5. Segmentación de la imagen a partir del etiquetado de regiones conexas, que permitirá extraer información de cada una de las áreas en que se encuentra dividida la imagen. El proceso de segmentación, al ser una de las actividades más relevantes y trascendentales que se realizan dentro del marco de la solución propuesta, también será objeto de estudio desde un punto de vista formal y teórico.
6. A partir de las propiedades de las regiones (concretamente se utiliza el área, la excentricidad, la orientación, las posiciones X e Y de los centros y el número de Euler) se identifican las tres elipses de la figura que tienen el área pintada de blanco. Para ello, se usan umbrales de filtrado preliminares basados en el área y la excentricidad, así como el número de Euler, que es un descriptor de regiones que resulta invariante frente a traslaciones, rotaciones y cambios de escala, y nos permite de forma sencilla discriminar entre ciertas clases de objetos. Se sabe de antemano que su valor será 1 para las elipses y -2, -1, 0 y 1 para el caso de los círculos. En la detección de elipses se emplea un indicador, que se genera a partir de sumar el valor absoluto del ratio de proporciones de distancias, con la diferencia de los valores absolutos de los ángulos de las orientaciones de las elipses E1 y E3.

7. A partir de la información anterior, y utilizando las ecuaciones paramétricas de la elipse, se procede a dibujar el perímetro de las tres elipses, así como el centro de cada una de ellas sobre la imagen binaria que se ha generado en el paso 3. Estas ecuaciones paramétricas también se reutilizarán en el dibujo del contorno de los círculos cuando el grado de inclinación con el que se tome la imagen sea demasiado elevado, y por consiguiente el círculo sea visto como una elipse y no se pueda emplear la ecuación general de la circunferencia.
8. Se determina cuál es la elipse central (E2), ya que se sabe de antemano por el análisis preliminar de la figura, que la distancia existente entre las elipses laterales es mayor que la distancia de la elipse central a cualquiera de las otras dos elipses. Esto a simple vista no es fácil de discernir, pero tras haber realizado un análisis geométrico exhaustivo de la figura recogido a través de la ilustración 6-1, quedó patente que las distancias son distintas.
9. A partir de la orientación de la elipse E2 -que ha sido identificada en el paso anterior-, se determina el ángulo con el que se debe rotar la imagen original para que quede siempre orientada como se muestra en la figura 3-1 en la que se propone la nomenclatura para cada una de las figuras que serán objeto de estudio en el reconocimiento de la figura. Esta nomenclatura estándar es importante, ya que será usada de forma continuada para hacer alusión a las subfiguras que forman parte del dibujo.
10. Una vez se ha rotado la imagen original quedando con la orientación que se muestra en la figura 3-1, se vuelve a escalar (si procede), binarizar, limpiar, segmentar y extraer las propiedades de las regiones para volver a identificar las elipses. Esto es, se vuelven a ejecutar los pasos 2, 3, 4, 5, 6, 7 y 8, pero esta vez sobre la imagen original ya rotada y vista tal y como aparece en las figuras 3-1 ó 6-1.
11. A partir de las propiedades de las regiones obtenidas en el paso anterior (posiciones de los centros esencialmente), se procede a la identificación de las elipses E1, E2 y E3, izquierda, central y derecha respectivamente según la perspectiva de la figura 3-1. Esta identificación va acompañada de un proceso que permite dibujar el contorno a partir de la ecuación paramétrica de la elipse. El proceso de identificación es relativamente complejo, y se basa en un análisis de orientación y distancia entre las elipses candidatas tomadas de tres en tres (el número total de comparaciones se corresponde con el total de combinaciones sin repetición) que finalmente es almacenado en un vector que resulta ordenado de forma ascendente al acabar el proceso. En el análisis se obtiene la diferencia de orientación y distancia en términos de valor absoluto en relación a la estimación previa reflejada en la figura 6-1.
12. Una vez identificadas las elipses, se procede a la identificación de los círculos 1, 2, 3 y 4 según se determina en la figura 3-1. El proceso de identificación resulta similar al de las elipses, aunque se subdivide en dos fases (la identificación de C1-C2 por un lado, y C3-C4 por otro). Una vez identificados los círculos, se procede al dibujado de su contorno utilizando para ello un umbral basado en la excentricidad. De este modo, cuando el grado de inclinación con el que se toma la imagen respecto al suelo es elevado, y por tanto el círculo es visto como una elipse, se utilizan las ecuaciones paramétricas de ésta. Sin embargo, cuando la perspectiva de la imagen no está inclinada, y por consiguiente no se supera dicho umbral, se emplea la

expresión propia del círculo basada en el radio, el coseno (para el eje X) y el seno (para el eje Y).

13. Análisis geométrico de las elipses en su conjunto en base a los siguientes aspectos que se enumeran a continuación: análisis de las posiciones de los centros, análisis del ángulo que forman las rectas que unen los centros, análisis de la proporcionalidad de distancias, análisis de la similitud del área y de la excentricidad de las elipses. Esto es, se llevan a cabo, los Análisis Parciales Comparativos (APCs): 1, 2, 3 y 4.
14. Análisis geométrico individualizado de cada uno de los cuatro círculos respecto a la posición y geometría de las elipses identificadas anteriormente. En dicho análisis se verifica que el ángulo, la distancia, y el ratio del área respecto a las elipses se encuentre dentro del umbral definido en los rangos determinados a partir del análisis geométrico previo y los umbrales de error hallados a partir de los juegos de pruebas realizados. Se ejecutan por tanto, los Análisis Parciales Comparativos (APCs): 6, 7, 8 y 9.
15. Análisis de los círculos en su conjunto: APC 10 para determinar la coherencia geométrica que se desprende a partir del análisis geométrico que se pone de manifiesto en la figura 6-1. En dicho análisis se comprueba que la relación entre las distancias de los centros de los círculos y el ángulo que forman dichas rectas (las que unen de forma imaginaria los centros de los círculos) se encuentren dentro de los umbrales y estimaciones preestablecidas.
16. A partir de los resultados que se obtienen en cada uno de los análisis enumerados en los pasos 13, 14 y 15, se determina el nivel de probabilidad final, utilizando para ello una media aritmética ya que se proporciona el mismo peso a cada uno de los Análisis Parciales Comparativos (APCs). En este sentido, cabe reseñar, que el número total de criterios que se emplean para hallar la media son 10.
17. Determinación final del resultado. Si la probabilidad es mayor o igual a 0.6 (estipulada en la variable *Configuracion.ProbabilidadMinimaDeteccionFigura*), entonces se da por identificada la figura; en caso contrario se da por no reconocida y se comienza de nuevo el proceso (otra iteración gracias al uso de la recursividad) con otro tipo de técnica de umbralización siempre y cuando se encuentre activada la función de *autoumbral* definida a través de la sección de configuración del sistema.

Cabe reseñar, que en la solución propuesta existe una gran cantidad de parámetros que son configurables desde la sección de configuración del algoritmo. Entre ellos, destaca las opciones de visualización, la ruta de la imagen, el nivel de resolución, el uso de umbral automático, el número de elipses a trabajar, y todos los umbrales de detección: ángulos, distancias, porcentaje de diferencia en la excentricidad, en el área, etcétera. Algunos de ellos quedan determinados a través de la figura 6-1 en la que se muestra un análisis geométrico de la figura dibujada sobre la plataforma de aproximación.

3.2.3 – Depuración visual del proceso

Finalmente, se muestra una secuencia cronológica de pantallas que sirven para ilustrar las diferentes fases por las que va pasando el algoritmo para poder realizar la identificación de la figura. Inicialmente, se procesa la imagen original, se hace un análisis y se dictamina el ángulo de giro para llevar a cabo la rotación que permita orientar la imagen; una vez orientada, se procede con el reconocimiento.

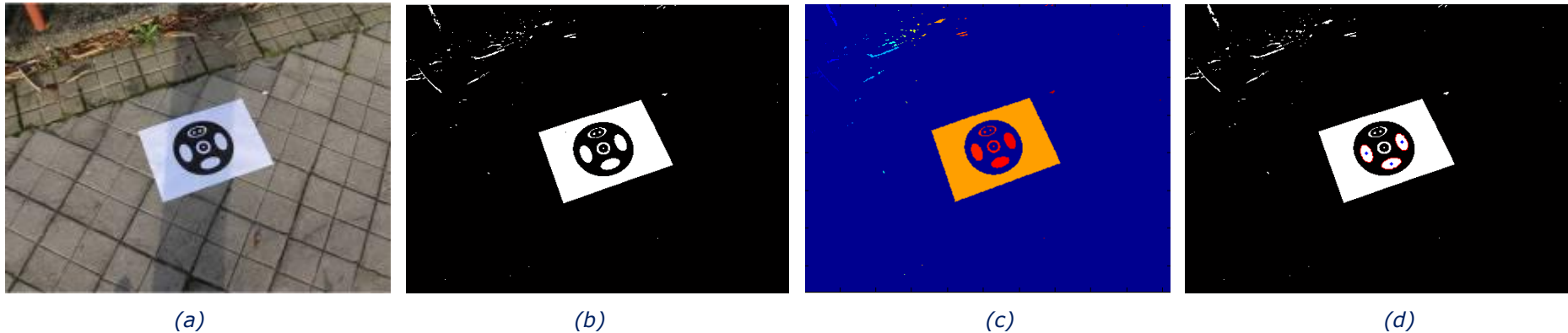


Figura 3-4 – Depuración visual - Imagen Original (a) Original; (b) Binaria; (c) Regiones; (d) Binaria Limpia Elipses Identificadas

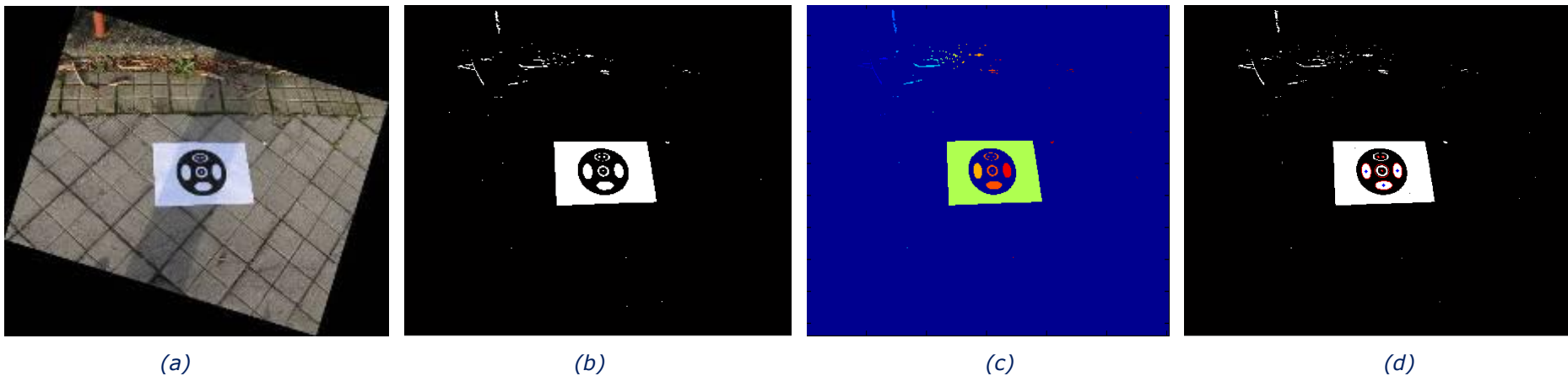


Figura 3-5 – Depuración visual - Imagen Orientada (a) Original; (b) Binaria; (c) Regiones; (d) Binaria Limpia Elipses y Círculos Identificados

3.3 – Transformación espacial

Por medio del siguiente aparatado, se llevará a cabo una breve descripción teórica, de dos transformaciones espaciales que se han empleado en la solución propuesta: el *escalado*: para trabajar con una resolución que permita reducir los tiempos de ejecución; y la *rotación*, para orientar la imagen de tal forma que siempre se realice el análisis geométrico de la figura utilizando una misma perspectiva, algo que nos facilitará su estudio al poder usar valores constantes sin necesidad de tener que hacer uso del cálculo diferencial.

3.3.1 – Escalado

En un proyecto como el nuestro, que requiere la identificación de una imagen en tiempo real, el coste computacional juega un papel muy importante, sobre todo si se tiene en cuenta que pueden ser necesarios varios intentos, y que la autonomía de los vehículos autónomos es limitada. Además, cabe reseñar, que las operaciones aritméticas que aumentan el porcentaje de uso de la CPU a cotas muy elevadas durante tiempos prolongados, requieren un consumo energético mucho mayor que aquellas que hacen uso más liviano del mismo. Por lo tanto, el proceso de escalado tiene como objetivo reducir la resolución de la imagen original para disminuir el tiempo de computación necesario para llevar a cabo la identificación de la figura.

En este sentido, cabe reseñar, que gran parte de las operaciones que se hacen en el área de visión por computador tienen un coste asintótico ó computacional $O(n^2)$ cuadrático, ya que suele ser necesario el uso de dos bucles anidados para recorrer las filas y las columnas de las matrices que representan formalmente las imágenes. Por lo tanto, trabajar con resoluciones elevadas requiere un mayor número de operaciones y por ende un coste computacional que a veces es difícil de asumir en entornos de tiempo real. Por este motivo, en nuestro proyecto se ha decidido llevar a cabo una comprobación de la resolución de la imagen, y si ésta excede un determinado tamaño que será totalmente configurable a través de una variable de configuración, se llevará a cabo una operación de escalado para reducir el tamaño, y por ende, el tiempo de computación.

La transformación geométrica espacial basada en el escalado, es probablemente una de las técnicas más empleadas en el ámbito de la informática gráfica asistida por computador, ya que cada vez que un usuario redimensiona una imagen con su ordenador, implícitamente se produce una operación de escalado sobre ésta. A partir del algebra matricial y las matrices de transformación geométricas, es posible aplicar una operación de escalado sobre un punto P de coordenadas (x,y) , usando un punto fijo P_0 de coordenadas (x_0, y_0) , lo que implica multiplicar por sendos factores $(s_x$ y $s_y)$ las distancias horizontal, d_x , y vertical, d_y , entre P_0 y P , con lo que se obtendrá el nuevo punto P' de coordenadas (x', y') . En el caso de que se esté trabajando en un plano bidimensional (2D), la expresión algebraica que permite su cálculo se determina a través de la ecuación 3.1:

$$P' = S \cdot P; \quad P' = \begin{pmatrix} x' \\ y' \end{pmatrix}, P = \begin{pmatrix} x \\ y \end{pmatrix} S = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix}, \quad \text{con: } \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} \cdot \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \quad (3.1)$$

Los factores de escala, s_x y s_y , pueden tomar cualquier valor positivo. Factores de escala superiores a 1 producen un alejamiento en la dirección horizontal o vertical según el factor implicado de P , con respecto al origen de coordenadas. Por contra, factores de escala inferiores a 1 producen un acercamiento en dirección horizontal o

vertical según el caso, de P con respecto al origen. Obviamente, factores de escala unitarios no modifican la posición sobre el eje correspondiente del punto P . No obstante, en caso de trabajar en un espacio tridimensional, la expresión matricial para el escalado de un punto P , de coordenadas (x, y, z) , a partir del origen de coordenadas es la siguiente, teniendo en cuenta que cuando los tres factores de escalado s_x, s_y, s_z son idénticos, hablamos de escalado homogéneo y, si no lo son, de escalado no homogéneo:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \text{ es decir, } P' = S(s_x, s_y, s_z) \cdot P \quad (3.2)$$

En la expresión anterior, se considera que el punto coincide con el origen de coordenadas y que los parámetros s_x, s_y y s_z son números positivos que especifican los valores de escalado con respecto a cada uno de los tres ejes coordenados. Tal y como se puede observar, la matriz resultante es de 4 dimensiones. En este sentido, cabe reseñar, que la dimensión adicional proviene de un subterfugio para expresar las traslaciones como productos por matrices. Esto se debe fundamentalmente a que la multiplicación de un vector por una matriz corresponde a la aplicación de una función lineal, pero la operación de traslación no lo es, ya que en la traslación por un vector t , el origen no se aplica al origen (valga la redundancia), sino al punto $O + t$. Por consiguiente, para expresar estas traslaciones mediante productos por matrices, se introduce una cuarta coordenada en los puntos con la siguiente convención que queda reflejada en la ecuación 3.3:

$$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x X \\ s_y Y \\ s_z Z \\ 1 \end{bmatrix} \quad (3.3)$$

Así, a cada punto del espacio le hacemos corresponder toda una recta por el origen en el espacio de cuatro dimensiones. De esta forma, se consigue expresar la traslación como una operación de multiplicación por una matriz. Esta matriz la representaremos como $T = (\vec{t})$

Finalmente, hay que reseñar, que hasta el momento se ha abordado el escalado de un punto pero siempre a partir del origen de coordenadas. Sin embargo, esto no tiene por qué ser siempre así, ya que puede interesarnos hacerlo desde otro punto fijo cualquiera. Para ello, se debe proceder de la siguiente manera:

1. Aplicar una traslación al objeto y al punto fijo, de forma que este último coincida con el origen de coordenadas.
2. Escalar el objeto a partir del origen de coordenadas.
3. Deshacer la traslación inicial, de forma que el punto fijo vuelva a su posición original.

3.3.2 – Rotación

En la solución propuesta, la rotación de la imagen original sirve para definir una orientación que permita hacer el análisis geométrico de la figura siempre desde la

misma perspectiva, de tal manera, que la recta imaginaria que interconecta las elipses laterales (según la figura 3-1) quede alineada con el eje de abscisas. Hacer el análisis geométrico utilizando una orientación predeterminada facilita su estudio aunque es más costoso desde un punto de vista computacional.

En este sentido, cabe reseñar que para determinar el ángulo de giro, es necesario procesar la imagen original realizando una binarización y segmentado que permita extraer las propiedades de las regiones para averiguar cuál es la orientación de la elipse central (E2 según la perspectiva de la figura 3-1) respecto al eje X o eje de abscisas. De esta forma, se consigue que el análisis siempre sea perpetrado desde una misma orientación.

Una vez aclarado este aspecto, a continuación se explica teóricamente como se lleva a cabo la transformación espacial basada en la rotación, ya que en el ámbito de la visión por computador, surge en múltiples ocasiones la necesidad de aplicar transformaciones geométricas a un objeto determinado por sus vértices. Así, por ejemplo, dado un poliedro en 3D definido por sus vértices, puede resultar factible estar interesado en tratar de hallar las nuevas coordenadas de dichos vértices tras aplicar una combinación de traslaciones, rotaciones y escalados.

A partir del algebra matricial y las matrices de transformación geométricas, es posible aplicar una rotación sobre un punto P de coordenadas (x,y) , de tal forma que la posición de éste se modifica siguiendo una trayectoria circular en el plano OXY , hasta convertirse en el punto P' de coordenadas (x', y') . En este sentido, hay que tener en cuenta, que para generar una rotación, se debe especificar un ángulo de rotación θ y la posición (x_r, y_r) del punto de rotación o punto pivote, P_r , a partir del cual el punto P es rotado. Dado que las imágenes con las que se trabaja se pueden representar tanto en dos, como en tres dimensiones, se ha estimado oportuno mostrar ambas matrices de transformación geométricas. Para imágenes en dos dimensiones la matriz es la siguiente:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad \text{donde } R = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3.4)$$

Sin embargo, si trabajáramos en un espacio tridimensional, entonces la matriz de rotación resultante tendrá el siguiente aspecto que se determina por medio de la ecuación 3.5:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad \text{donde } R = \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.5)$$

3.4 – Espacios de Color

En lo que concierne al espacio de color a utilizar, y a los diferentes formatos de representación existentes en la actualidad, deben tenerse en cuenta las siguientes consideraciones que se analizan a continuación.

3.4.1 – La luz, el color y la visión

En primer lugar es necesario tener en consideración, que desde el punto de vista de la física, se piensa en la luz como la propagación de energía electromagnética en forma de ondas o como haces de partículas (fotones). A modo de ejemplo, se debe tener en cuenta, que el ojo humano es capaz de percibir ondas luminosas entre los 380 nanómetros de longitud de onda (el violeta) y los 760 nanómetros de longitud de onda (el rojo). Sin embargo, en lo que concierne a la visión por computador, y por ende a la representación de gráficos por ordenador, el modelo de ondas será suficiente para explicar el comportamiento de la luz, su naturaleza, y la percepción que tiene el ser humano de los diferentes tipos de radiaciones electromagnéticas.

Nuestros ojos actúan a modo de pequeñas cámaras fotográficas, con la córnea y los cristalinicos (ubicados en la parte anterior) haciendo de objetivo, y la retina haciendo la función de película. En la retina existen diferentes receptores especializados que transforman la luz incidente en impulsos nerviosos, que el sistema nervioso y el cerebro eventualmente, descodifican en las imágenes que percibimos finalmente. Estos receptores en la retina son de dos tipos bien diferenciados: *bastones* y *conos*, así llamados por su forma. Los primeros son sensibles a la intensidad de la luz, sin que importe la longitud de onda; esencialmente, forman una imagen en blanco y negro (escala de grises) de lo que vemos. Son mucho más predominantes que los conos, se estima que su existencia cubre un rango que va desde los setenta y cinco hasta los ciento cincuenta millones en cada ojo, pero también cubren un área que es mucho más extensa.

Los segundos, los conos, añaden el componente de color. Normalmente, solamente hay entre seis y siete millones en cada ojo, y se encuentran ubicados muy cerca del centro de la retina, que es el lugar donde se formará la imagen de aquello que miramos directamente. Existen tres tipos de células foto-receptoras, cada uno con una curva de respuesta espectral diferente, por lo que en teoría todos los colores pueden especificarse mediante tres números que se corresponden con las salidas de dichos foto-receptores.

3.4.2 – RGB

Por consiguiente, un color se caracteriza por una distribución espectral que permite averiguar la intensidad con la que participa cada frecuencia del espectro. De esta forma, surge la idea natural de simbolizar un color como una terna de números representando intensidades; esto es **rojo** (en inglés, red), **verde** (en inglés, green) y **azul** (en inglés, blue) de manera que surge la representación RGB. De esta forma, el modelo RGB utiliza tres números reales comprendidos entre 0 y 1, donde naturalmente el 0 indica ausencia de luz, y el 1, por el contrario, revela una luz de intensidad máxima para dicho color.

Este modelo, que se clasifica como aditivo (cada color se obtiene añadiendo ciertas cantidades de luz de cada uno de los tres colores base), es el modelo básico más empleado en pantallas y proyectores. Por lo tanto, se debe estudiar cómo convertir el modelo al del dispositivo teniendo en cuenta que la pantalla tiene que ser capaz de refrescarse rápidamente para evitar el parpadeo. Normalmente, se suele utilizar sólo la parte entera, con lo cual el intervalo $[0,1]$ se representa mediante los enteros entre 0 y 2^n-1 , donde n designa el número de bits destinados a representar la intensidad en aquel color.

Esto introduce lo que denominamos como un error de discretización (generado al representar una cantidad continua con un rango entero), que será poco apreciable si n es lo suficientemente grande, pero sí que lo será si es pequeño. De hecho, en el sistema RGB, el *error máximo de discretización* puede presentarse a través de la

expresión: $1 / (2(2^n-1))$). La expresión resulta correcta, ya que si se utiliza un solo bit para representar el color, el error máximo es de 0.5 (si quisiéramos representar el valor continuo 0.4999 utilizaríamos el 0 y si quisiéramos representar el 0.050001 emplearíamos el 1).

3.4.3 – HSV

El modelo de color HSV (también llamado HSB), es un sistema de colores diseñado para ser intuitivo, ya que se acerca a nuestra percepción subjetiva del color, permitiendo de esta manera a un usuario, especificar un color fácilmente. En este modelo, tenemos las siguientes componentes:

- Parámetro (H): expresado en grados sexagesimales, mide la clase de color que se trata. A los 0° (o a los 360°) encontramos el rojo, y variando H de 0° a 360° , pasamos por el amarillo, verde, cian, azul, magenta, y volvemos al rojo.
- Parámetro (S): hace referencia a la saturación del color y va de 0 a 100%. También a veces se llama la "pureza". Cuanto menor sea la saturación de un color, mayor nivel de gris tendrá y más atenuado parecerá.
- Parámetro (V): también denominado brillo (B), hace referencia al valor de las gamas de color, y va desde el 0% hasta el 100%. Se trata de una transformación no lineal del espacio de color RGB.

Tal y como se puede observar, la componente de luminosidad está separada en un canal propio, lo que tiene ventajas para ciertas aplicaciones de procesamiento de imágenes.

3.4.4 – Conversión RGB a HSV

En nuestro trabajo, dado que las cámaras fotográficas representan el color en el formato RGB, y a nosotros nos interesará más trabajar con el formato HSV con el fin de utilizar la imagen de intensidad que resulta necesaria para la binarización, será necesario llevar a cabo una conversión entre ambos formatos. Para entender el algoritmo de conversión, a continuación se muestra la representación visual de ambos formatos:

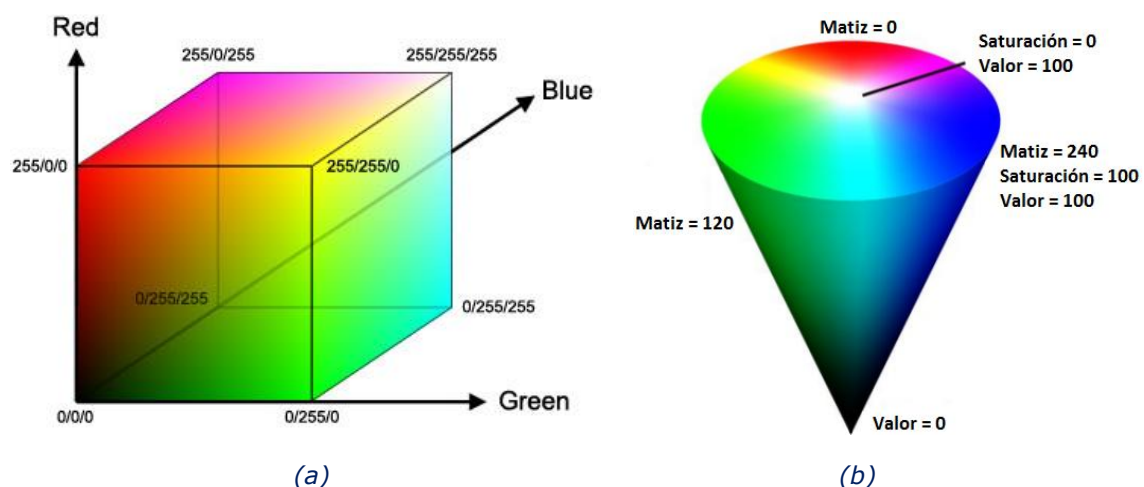


Figura 3-6 – Espacios de Color (a) Cubo RGB; (b) Cono HSV

Si se observa el cubo de colores RGB de manera que la diagonal que une el $(0, 0, 0)$ (negro) y el $(1, 1, 1)$ (blanco) se encuentre alineado con la dirección de visión, con el blanco más cerca del observador, veremos un hexágono con los colores rojo, amarillo, verde, cian, azul y magenta en los vértices. La proyección en un plano nos da la base del cono HSV. Si en lugar de empezar con todo el cubo de color RGB, empezamos con un subcubo $[0, c]^3$, donde $0 \leq c < 1$, se obtiene un hexágono como el anterior, pero más pequeño. Su proyección en un plano corresponde a una sección del cono para $B = c$. Se puede entonces, realizar la conversión del modelo RGB al HSV mediante un algoritmo que convierte unos valores de R, G y B a otros de H, S y V que representen el mismo color.

3.5 – Umbralización

A pesar de que los procesos de umbralización y binarización están estrechamente relacionados y vinculados a la hora de llevar a cabo el proceso de segmentación de una imagen, pueden ser considerados como dos conceptos distintos dentro del área de la visión por computador, y por tal motivo se ha estimado oportuno que debían ser analizados por separado. El proceso de umbralización de imágenes, conocido como *thresholding* en inglés, es un problema sustancial en el análisis automatizado de imágenes, ya que muchas de las aplicaciones de procesamiento de imágenes que se realizan en el área de la visión por computador, requieren por norma general trabajar con imágenes binarias, es decir, imágenes en la que sus píxeles únicamente atienden a 2^1 tipos de representaciones posibles, es decir, 0 y 1, que hacen alusión al color negro y blanco respectivamente.

En este sentido cabe reseñar, que al elegir un valor de intensidad particular, -de ahí que se pasara al modelo HSV porque la intensidad está separada en un canal propio definido como "umbral"-, las imágenes se pueden segmentar estableciendo aquellos píxeles cuya intensidad original está por encima del umbral, como "píxeles blancos", y el establecimiento de los otros píxeles como "píxeles negros." Este proceso, es una de las formas naturales que surgen cuando se desea extraer en una imagen el fondo del resto de los elementos que la forman. En nuestro caso concreto, se tiene que extraer el fondo de la imagen (entorno en el que se ubique la plataforma de aproximación: una pradera, la cima de una montaña, etcétera) de los elementos geométricos que componen la figura. De esta manera, una forma obvia de extraer los objetos del fondo es seleccionar un umbral T que haga esta discriminación. Después, cualquier punto (x,y) para el que $f(x,y) > T$, será considerado un punto de la figura, mientras que cualquier otro punto que no cumpla dicha condición, será estimado como fondo.

3.5.1 – Estrategias de Umbralización

Por consiguiente, la umbralización es uno de los métodos más triviales que se pueden esgrimir cuando se requiere segmentar una imagen de manera automática mediante un proceso asistido por ordenador. En este sentido, se debe tener en consideración, que en el momento de determinar este umbral, es posible emplear dos estrategias claramente diferenciadas:

- Umbral estático: prefijar de antemano un umbral que se establece de forma manual, y que por consiguiente no variará a no ser que el propio usuario lo cambie expresamente a través de las opciones de configuración del sistema. Su principal problema es que no se adapta a las diferentes situaciones que se pueden dar a la hora de analizar imágenes distintas, y normalmente su asignación se formaliza mediante un proceso de ensayo / error. Un umbral

estático suele ocasionar problemas cuando varía el contraste durante el proceso de adquisición de imágenes.

- Umbral dinámico: consiste en utilizar un método de umbralización que se adapte a las imágenes, de tal forma que su valor varía en función del tipo de imagen que se vaya a procesar. Un ejemplo de este tipo de umbralización, y que además se encuentra bastante extendido, es el método de *Otsu* (1979) [29,41]. Esta técnica se utiliza para tratar de averiguar un umbral óptimo de forma automática sin necesidad de que se tenga que facilitar información previa de la imagen. Además una de sus principales ventajas radica en que se considera apto para imágenes que contengan ruido, histogramas planos o problemas de iluminación.

3.5.2 – Descripción Formal

De manera formal, podemos definir la umbralización (*thresholding* en inglés) como el proceso que permite clasificar los píxeles de la función $f(x,y)$ atendiendo a una operación que implica pruebas en una función T de la forma siguiente:

$$T = T[x, y, p(x, y), f(x,y)] \quad (3.6)$$

Donde $p(x,y)$ denota una propiedad local en el punto x, y , mientras que $f(x,y)$ determina el nivel de gris en ese punto. De tal forma, una imagen $g(x,y)$ a la que se le haya aplicado el proceso de umbralización se define formalmente de la siguiente manera:

$$g(x,y) = \begin{cases} 0, & f(x,y) > T \\ 1, & f(x,y) \leq T \end{cases} \quad (3.7)$$

En un proceso de umbralización clásico, aquellos puntos, es decir, píxeles, que estén marcados como 1 se corresponden a objetos, mientras que los que se hayan clasificado como 0 son elementos que forman parte del fondo o viceversa.

3.5.3 – Tipos de umbralización

Si se toma como premisa la función de binarización previa, cuando T depende de manera exclusiva de $f(x,y)$, estamos hablando de un tipo de umbral global, pero si depende tanto de $f(x,y)$ como de $p(x,y)$, en este caso ya se trata de un umbral local. Por consiguiente, se puede esgrimir la siguiente clasificación:

- Umbral global: se da cuando en la ecuación de T sólo influye $f(x,y)$, por lo que se suele utilizar en imágenes donde existe una clara diferencia entre objeto / entorno, y cuando la iluminación es uniforme. Los métodos que se basan en esta metodología, tratan de encontrar un umbral que pueda ser aplicable a toda la imagen. De esta forma, la segmentación se lleva a cabo mediante un barrido de la imagen píxel por píxel, y el etiquetado de cada píxel depende de forma exclusiva del valor de T , con el cual se realiza el pertinente proceso comparativo. De esta manera, el éxito o fracaso del método depende de manera exclusiva de la partición del histograma. Un claro exponente de este tipo de umbral, es el ya mencionado de *Otsu*.
- Umbral local: sucede cuando la ecuación de T depende de $f(x,y)$, pero también depende de $p(x,y)$. Se puede utilizar cuando hay niveles de iluminación por franjas, y por tanto el umbral a manejar es distinto en cada

zona iluminada por cada franja diferente. Los métodos que se basan en el tipo de umbral local, obtienen el valor del umbral para cada píxel usando el valor de los píxeles vecinos. En general producen un resultado mejor que el de los umbrales globales, incluso en aquellos entornos donde la iluminación es variable. Por contra, sus altas necesidades de memoria y sus grandes requerimientos de procesamiento de datos, hacen que no siempre se puedan implementar en entornos de tiempo real. Dos métodos basados en el umbral local son el de *Niblack* (1986) [30] y *Sauvola* y *Pietikäinen* (2000) [42].

- Umbral dinámico: se produce cuando la ecuación de T depende de los tres términos, es decir, $f(x,y)$, $p(x,y)$ y las propias coordenadas (x,y) . Un ejemplo de este tipo de umbral se daría cuando se emplea sobre iluminación arbitraria que puede provocar en la imagen efectos, como sombras, reflejos, etcétera. Por lo tanto, los umbrales que se empleen deberán adaptarse a las condiciones especiales que se produzcan en cada una de estas zonas que tienen efectos indeseados.

3.5.4 – Método de Otsu

Con el fin de obtener un método complementario al tipo de umbralización directa, que es utilizada de forma expresa en la solución propuesta, se ha estimado oportuno emplear como alternativa el método de Otsu [29,41], ya que en aquellos casos en los que la estrategia basada en umbralización directa no devuelve los resultados esperados, el método de Otsu se ha comportado de una forma bastante acertada. Existen diversas imágenes dentro de las analizadas, que para un mismo escalado, el método por umbralización directa no ha sido capaz de retornar una imagen binaria con la suficiente calidad que sirviera de base a la hora de realizar una segmentación que extrajera las regiones de forma correcta. Sin embargo, curiosamente ante este tipo de imágenes y escalados en los que la umbralización directa fallaba, el método Otsu ha sido un complemento perfecto, ya que ha permitido obtener una imagen binaria lo suficientemente buena como para acometer el proceso de segmentación de forma adecuada.

Por lo tanto, el método de Otsu se utiliza como parte del proceso de umbralización en la solución propuesta, con el fin de dar soporte a aquellas situaciones en las que la umbralización directa no se comporta adecuadamente. En este sentido hay que reseñar, que la mayoría de las técnicas de umbralización están basadas en estadísticas sobre el histograma unidimensional de una imagen, aunque también existen otras que utilizan la matriz de co-ocurrencia de dicha imagen. Para localizar el umbral se puede optar tanto, por usar procedimientos paramétricos, como no paramétricos.

- Procedimientos paramétricos: la distribución de los niveles de gris de una clase de objeto es la que se utiliza para poder encontrar cuál es el valor del umbral adecuado para dicha imagen.
- Procedimientos no paramétricos: los umbrales se obtienen de forma óptima de acuerdo a algún criterio, como puede ser aquel en el que se minimice la suma de las varianzas entre clases, que es el usado por el método de Otsu.

Por lo tanto, el método de Otsu utiliza una medida de la dispersión de valores como es la varianza, que permite obtener la dispersión de los niveles de gris que hay en el interior de la imagen. Para ello, el método de Otsu considera que el histograma es *bimodal* (formado por la suma de dos densidades de probabilidad) e itera a lo largo de éste calculando para cada valor estimado posible de T [0, 255], la varianza

de pesos dentro de cada clase. El umbral óptimo se logra cuando la varianza entre clases entregue un valor mínimo. Si lo trasladamos al caso que nos ocupa, resulta obvio que en una imagen binaria, sólo hay dos clases: una clase representada por el valor 0, y la otra por el valor 1.

Dada una imagen con L niveles de intensidad y asumiendo que el umbral buscado es T , las probabilidades acumuladas de cada clase hasta T , y desde T hasta L , son:

$$\omega_1(t) = \sum_{z=1}^T P(z); \quad \omega_2(t) = \sum_{z=T+1}^L P(z) \quad (3.8)$$

A continuación, ya estamos en disposición de calcular los promedios y las varianzas de cada una de las clases tal y como se indica a continuación:

$$\begin{aligned} \mu_1(t) &= \sum_{z=1}^T zP(z); & \sigma_1^2(t) &= \sum_{z=1}^T (z - \mu_1(t))^2 \frac{P(z)}{\omega_1(t)} \\ \mu_2(t) &= \sum_{z=T+1}^L zP(z); & \sigma_2^2(t) &= \sum_{z=T+1}^L (z - \mu_2(t))^2 \frac{P(z)}{\omega_2(t)} \end{aligned} \quad (3.9)$$

Finalmente, ya se puede obtener la ecuación que permite determinar la varianza ponderada entre clases tal y como se muestra por medio de la siguiente expresión:

$$\sigma_w^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t) \quad (3.10)$$

En el método de Otsu, se elige el umbral T correspondiente al nivel de intensidad que proporcione la mínima varianza ponderada definida en la expresión que se acaba de mostrar. Una vez se ha determinado la expresión formal que se utiliza para hallar el valor del umbral, podemos resaltar sus principales ventajas:

- Buena respuesta frente a la mayoría de situaciones típicas del mundo real (imágenes ruidosas, con histogramas planos, mal iluminadas...).
- Es dinámico y no precisa de supervisión humana, por lo que no requiere un preprocesamiento de la imagen u otro tipo de información adicional.

Por el contrario, su principal desventaja es que a medida que el número de clases en la imagen aumenta, el método necesita mucho más tiempo para seleccionar un umbral multinivel adecuado. No obstante, en imágenes binarias como la que nos atañe para nuestro proceso de segmentación, únicamente hay dos clases, y por ende, los tiempos de ejecución serán adecuados.

3.5.5 – Decisión adoptada

Finalmente, la decisión adoptada en lo que se refiere al proceso de umbralización, ha consistido en emplear las dos estrategias comentadas con anterioridad dentro del algoritmo propuesto cuyo código fuente se proporciona en el Anexo IV. De este modo, se implementará tanto la umbralización estática como la umbralización dinámica. En lo que se refiere a la umbralización estática, el sistema utilizará un valor por defecto, que será 0.8, aunque puede ser cambiado manualmente a través de una variable de configuración del sistema llamada *Configuracion.Umbral*. En lo que concierne al tipo de umbralización dinámica, ésta se llevará a la práctica por medio de una implementación del método de Otsu que incorpora una función llamada *graythresh*, que forma parte de la *Image Processing Toolbox* de MATLAB.

En este sentido cabe destacar, que el primer parámetro que se le envía a la función *PlataformaUAV.m*, que es la que recoge toda la funcionalidad del algoritmo, sirve para indicar explícitamente el tipo de umbralización que el usuario pretende utilizar; los posibles valores son [*directo, otsu, automatico*]. No obstante, en caso de que no se indique el tipo de umbralización, el sistema utilizará el valor *directo* por defecto, o el que se establezca en la variable *Configuracion.TipoUmbral* que se encuentra en la sección inicial del programa. El hecho de que por defecto se asigne el valor *directo* a la variable que se acaba de mencionar, se debe a que a partir de las pruebas y resultados que se han puesto de manifiesto a través del apartado 5.2 - *Pruebas, Datos y Resultados*, y los *Anexos I, II y III*, es factible concluir, que la estrategia de umbralización directa ha tenido una tasa de aciertos mucho mayor, que la estrategia de umbralización dinámica basada en el método de Otsu.

De esta manera, para una misma imagen se pueden emplear los dos métodos de umbralización, de tal modo que si el primero falla, todavía resulta posible utilizar el segundo antes de tener que generar una nueva instantánea a través de un proceso de adquisición. No se debe olvidar, que hay imágenes que para un tipo de escalado concreto se segmentan bien con el método directo y mal con el de Otsu, y viceversa. Aunque también hay imágenes que se segmentan adecuadamente con las dos estrategias, y raras ocasiones en las que ambas fallen. Por tal motivo, se ha implementado una opción automática, que utiliza en primera instancia la estrategia de umbralización directa, y si ésta llega a generar una probabilidad inferior a 0.6 (establecida en la variable *Configuracion.ProbabilidadMinimaAutoumbralizacion*), o excepciones que hayan sido controladas durante el proceso de análisis, se llama a la función *GestionAutoUmbral*, que se encarga de comprobar si resulta necesario llamar de manera recursiva a la función principal *PlataformaUAV.m*, para probar la binarización con el método Otsu.

En el caso que no se haya empleado como primer parámetro en la función la opción automática pero se encuentre activa la opción *Configuracion.Autoumbralizacion*, el sistema tratará de emplear la estrategia de umbralización que no fuera usada inicialmente, en el caso que la probabilidad sea inferior a 0.6, o se produzca una excepción controlada en tiempo de ejecución. El objetivo del sistema, es tratar de evitar a toda costa tener que realizar un nuevo proceso de adquisición, por lo que se debe "jugar" con los diversos tipos de umbralizaciones existentes. En este sentido hay que reseñar, que se ha trabajado muy a fondo el tema relativo a la eficiencia computacional (tal y como se detalla en el apartado 4.3 - *Problemas de Eficiencia Computacional*), con el fin de reducir el tiempo de ejecución al máximo, y así poder hacer el mayor número de pruebas por unidad de tiempo para tratar de disminuir tanto como sea posible el número de veces en los que se debe repetir el proceso de adquisición. El objetivo es probar los dos umbrales en caso de que falle el primero, dentro de un intervalo de tiempo de ejecución razonable.

Finalmente hay que reseñar, que la variable *Configuracion.Autoumbralizacion* sirve para controlar el final de la recursividad en el caso de que el valor devuelto por la función principal durante el proceso de análisis e identificación de la plataforma, nunca supere el valor que se haya asignado a la siguiente variable establecida en la sección de configuración: *Configuracion.ProbabilidadMinimaAutoumbralizacion*. De esta manera, cuando se lleve a cabo la recursividad, el cuarto parámetro que puede recibir la función principal *PlataformaUAV.m*, que se asignará a la variable llamada *Configuracion.Autoumbralizacion*, se pondrá con valor *false* con el fin de detener la auto llamada en el momento en el que se evalúen las condiciones que se chequean nada más empezar la función *GestionAutoUmbral*. Si no se recurriera a este tipo de estrategia, en el caso de que una imagen no pueda ser identificada con un nivel de probabilidad mayor a 0.6 y ya fueran utilizadas las dos estrategias de umbralización, entraría en un bucle infinito del que el sistema no podría salir por sí sólo.

3.6 – Histograma

A pesar de que el término histograma no aparece reflejado en el diagrama de flujo presentado en el apartado 3.2.2 – *Diagrama de flujo*, al no ser un proceso propiamente dicho, se ha estimado conveniente realizar una breve exposición acerca de una de las herramientas de uso más extendido, y que mayor valor añadido aporta al área de la visión por computador (e incluso al de la propia fotografía como disciplina). No obstante, es necesario poner de manifiesto, que el histograma como tal no ha sido utilizado de manera explícita en la solución propuesta, ya que el sistema autónomo de umbralización empleará las dos técnicas de umbralización comentadas anteriormente, gracias al uso de un proceso recursivo que se dispara en función a la evaluación de unas condiciones previas en las que no se realiza un análisis del histograma. Esto tiene sus ventajas y sus inconvenientes, pero dejando al margen este tema, exponer el uso y las aplicaciones del histograma, es casi una obligación cuando se trata de resolver un problema relacionado con el área de la visión por computador.

Una vez aclarados estos aspectos preliminares que conciernen a la aplicación del histograma en la solución propuesta, se está en disposición de aseverar que, a grandes rasgos el histograma de una imagen no es más que una herramienta estadística basada en una representación cartesiana que permite dibujar en una gráfica la frecuencia con que aparecen valores dentro de una serie, de manera que cada línea o barra vertical indica la frecuencia con la que una variable toma un valor concreto dentro de su dominio de aplicación. Permite por tanto, relacionar los valores de la imagen con su frecuencia de aparición, es decir, indica el número de veces que aparece el valor de una variable (color, brillo, etcétera) en una imagen dada. A partir de tal información, se pueden obtener distintos tipos de aplicaciones prácticas del histograma con utilidades muy diversas, como puede ser: realización de estimaciones acerca de la superficie que ocupa un determinado objeto en la imagen, identificar zonas diferentes de una imagen, etcétera.

En el caso de la visión por computador, el histograma resulta útil porque permite plasmar un resumen estadístico del recuento de los píxeles de una imagen según su distribución, lo cual lo convierte en una herramienta muy versátil, ya que permite extraer información relevante acerca de la imagen. Se puede considerar por tanto, como una función discreta que representa el número de píxeles en la imagen en función de los niveles de intensidad, g . La probabilidad $P(g)$ de ocurrencia de un determinado nivel g , se define tal y como se muestra en la siguiente ecuación, donde M es el número de píxeles en la imagen, y $N(g)$ es el número de píxeles en el nivel de intensidad g . Como con cualquier distribución de probabilidad, todos los valores de $P(g)$ son menores o iguales que 1, y la suma de todos los valores $P(g)$ es igual a la unidad.

$$P(g) = \frac{N(g)}{M} \quad (3.11)$$

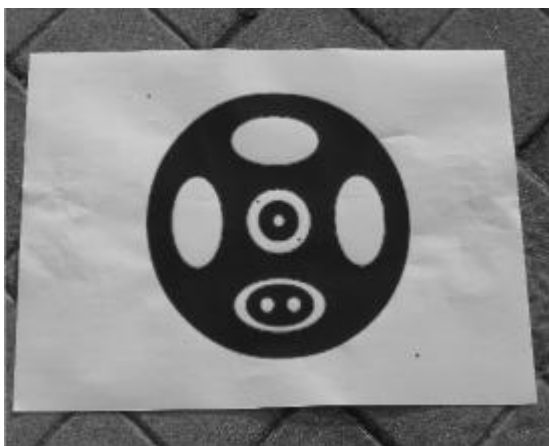
En tanto en cuanto nos atengamos al punto de vista de la variable que sea objeto de estudio, una misma imagen puede tener diversos histogramas. No obstante, en este sentido cabe reseñar, que existen determinados patrones de histogramas que sirven para tratar de generalizar determinadas condiciones que se dan en una imagen. Así, si atenemos al nivel de brillo de los píxeles (la variable suele estar en el eje horizontal y la frecuencia en el vertical) como variable objeto de estudio, y tomamos como punto de partida una muestra lo suficientemente amplia, podremos concluir que por regla general se considerará que una imagen tiene un buen contraste -entendiendo por contraste la diferencia de brillo entre las zonas más claras y más oscuras en una imagen-, si su histograma se extiende ocupando casi

todo el rango de tonos. También será factible observar, que aquellas imágenes que sean de carácter nocturno, tendrán su histograma sensiblemente desplazado hacia el lado izquierdo. Si se atiende al número de modos del que está compuesto un histograma, y se considera que un histograma está formado por la suma de superposición de modos, se está hablando de un histograma bimodal. Dichos modos se pueden definir mediante sendas funciones de densidad de probabilidad, por ejemplo, si se considera que un histograma está formado por la suma de dos densidades de probabilidad (gaussianas por ejemplo), la función global que aproxima el histograma viene dada por la siguiente expresión:

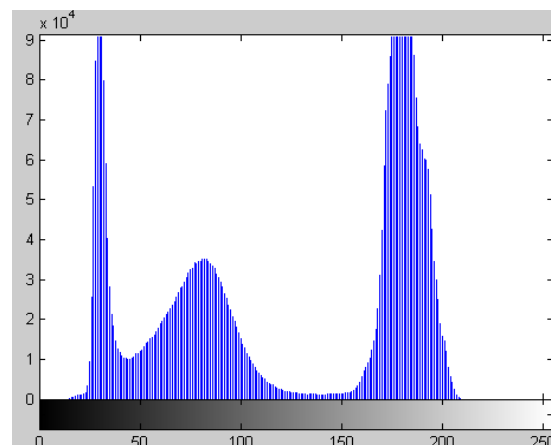
$$p(z) = P_1 p_1(z) + P_2 p_2(z) \quad (3.12)$$

Dado que el histograma sirve para realizar una estimación de la distribución de probabilidad de los niveles de intensidad, no resulta incongruente pensar, que el histograma de una imagen es una herramienta frecuentemente empleada en el análisis automático de imágenes, y por ende en el área de la visión por computador. Por lo tanto, en el caso que nos ocupa, es una herramienta que juega un papel muy importante en el proceso de binarización de una imagen, ya que la clave está en encontrar un nivel de gris o umbral que permita diferenciar la figura geométrica de la plataforma, del resto de elementos que aparecen en la escena. Este nivel puede llegar a determinarse a partir del análisis del histograma. Por ejemplo, se sabe que la iluminación no constante perturba el histograma y puede hacer que los objetos detectables mediante binarización dejen de serlo.

Sin embargo a veces no proporciona información significativa, ya que el hecho de que exista un valle en el histograma, no es indicativo seguro que se pueda discriminar el fondo de la figura. Una vez se ha realizado una explicación objetiva y exhaustiva acerca del concepto de histograma, es el momento de exponer un ejemplo real aplicado a la figura geométrica que se encuentra dibujada sobre la plataforma de aproximación del UAV. La figura que se presenta a continuación, es el resultado de una imagen real que ha sido convertida a una en escala de grises mediante el uso de la función `rgb2gray` que proporciona la *Image Processing Toolbox* de MATLAB:



(a)



(b)

Figura 3-7 – Histograma (a) Figura geométrica plataforma; (b) Histograma

En el caso de la umbralización global, donde el proceso de segmentación se realiza mediante un barrido píxel por píxel, el éxito del proceso de umbralización depende por completo de la partición del histograma. En este caso, el umbral T se puede hallar encontrando el mínimo $N-1$ entre los N picos consecutivos del histograma.

3.7 – Binarización

La binarización de imágenes es un proceso básico que se emplea en muchas de las aplicaciones del procesamiento digital de imágenes. El objetivo fundamental de esta técnica, es obtener una imagen que sólo será presentada por dos tonos de color, por lo general blanco y negro. El cimiento en el que se basa este proceso es sencillo, ya que únicamente se debe hallar el tono de color que se debe otorgar a cada píxel que sea mayor a un determinado umbral (valor límite), dado que el resto de píxeles tendrán por defecto el otro tono de color. De ahí, que el proceso de binarización y umbralización se encuentren íntimamente ligados, tal y como se comentaba con anterioridad. Por consiguiente, la binarización de una imagen consiste en comparar los niveles de grises presentes en la imagen con un valor (umbral) predeterminado. Si el nivel de gris es menor que el umbral calculado, se le asigna al píxel de la imagen binarizada el valor 0 (negro), y si es mayor, se le asigna un 1 (blanco).

De esta forma se obtiene una imagen en blanco y negro representada con una profundidad de color de 2^1 bits. Por lo tanto, en este proceso, la determinación del umbral es la parte más peliaguda y de mayor dificultad, sobre todo cuando se lleva a cabo en un entorno como el que concierne a este proyecto (imágenes de exterior), donde las condiciones son cambiantes. Desde el punto de vista del área de la visión por computador, la binarización tiene sentido cuando es posible distinguir entre objetos y fondo en base al umbral de los niveles de gris. De hecho, el objetivo que se plantea en este proyecto, es tratar de aislar la figura geométrica del resto de elementos que aparecen en la escena con el fin de etiquetar las regiones, y por ende, analizar sus propiedades para ver si coinciden con las figuras geométricas que en su conjunto componen la imagen (círculos y elipses mayormente). De esta forma, el objetivo que se persigue es hallar una imagen binaria similar a la que se presenta a continuación:

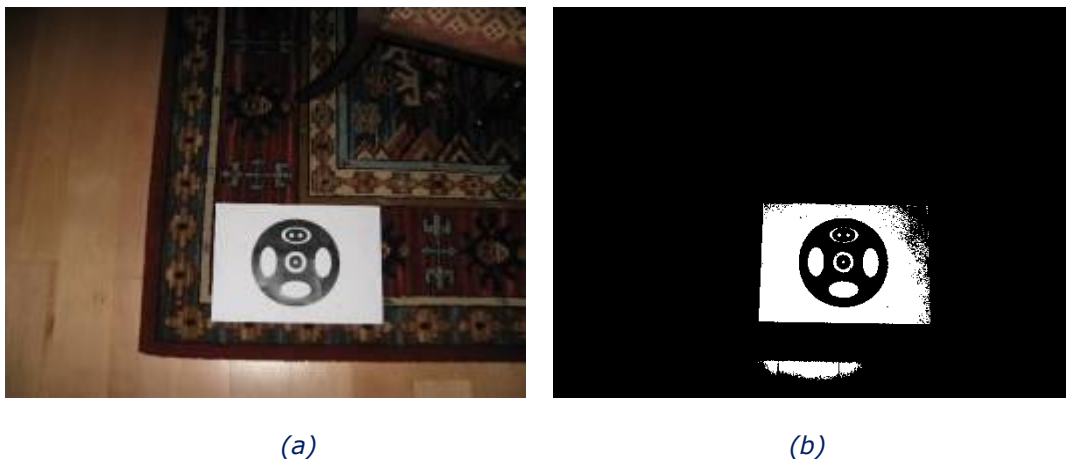


Figura 3-8 – Binarización (a) Imagen Original; (b) Imagen Binarizada

3.8 – Segmentación

El proceso de etiquetado de componentes conexas, técnica principal empleada en la tarea de segmentación, ha resultado ser una de las actividades más importantes en la solución propuesta, ya que permite segmentar la imagen binarizada en una serie de regiones, que serán estudiadas posteriormente en base a diferentes propiedades para poder discernir, si estamos o no ante nuestra plataforma de aproximación. Por

consiguiente, a partir del etiquetado de componentes conexas, y por ende de la segmentación, surge la materia prima necesaria para estudiar las diversas figuras geométricas que forman parte de la plataforma, como pueden ser las tres elipses cuya área está dibujada de color blanco, que servirán como base para el estudio geométrico que permita dictaminar si la imagen original tomada durante el proceso de adquisición, contiene la figura pintada sobre la superficie de la plataforma.

El proceso de segmentación permite diseccionar una imagen en una serie de partes o regiones, de tal modo que compartan una serie de características análogas en lo que se refiere a sus atributos de carácter más básico. Uno de los más usados es la *amplitud*, aunque los *contornos* y la *textura* también deben ser considerados ya que ha de tenerse en cuenta, que la forma de un objeto puede ser descrita en términos de sus bordes o de la región que ocupa. En ambos casos, la detección de contornos suele ser un proceso troncal a la hora de llevar a cabo la segmentación. En este sentido, a continuación se realiza una breve descripción teórica de los conceptos y técnicas más relevantes que se deben tener en cuenta en un proceso de este tipo.

3.8.1 – Vecindad entre píxeles

Los píxeles que rodean directamente un píxel p de coordenadas (x,y) son definidos como los vecinos de p . Por lo tanto, podemos decir que un píxel p en un plano cartesiano bidimensional (x,y) , tiene cuatro vecinos verticales y cuatro vecinos horizontales, cuyas coordenadas vendrán dadas por la siguiente expresión:

$$N_4(p) = (x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1) \quad (3.13)$$

Por otro lado, las vecindades diagonales también son cuatro, y vienen determinadas por la siguiente expresión:

$$N_D(p) = (x + 1, y + 1), (x + 1, y - 1), (x - 1, y + 1), (x - 1, y - 1) \quad (3.14)$$

En el área de la visión por computador, un concepto muy ligado al de vecindad es el de adyacencia, tanto, que de hecho se utiliza para poder discernir si dos píxeles son vecinos o no. En este sentido se puede aseverar, que dos píxeles son adyacentes entre sí, sí y solo sí, tienen en común una de sus fronteras o cuando menos una de sus esquinas. Por lo tanto, resulta factible tener píxeles adyacentes por frontera o por esquina, tal y como se puede observar en la siguiente imagen:



Figura 3-9 – Adyacencia (a) Adyacencia por frontera; (b) Adyacencia por esquina

Tomando como eje de actuación la adyacencia, podemos afirmar que dos píxeles se consideran vecinos si cumplen con la definición de adyacencia, lo que nos lleva a afirmar que existen dos tipos de relación de vecindad:

- Vecinos directos: comparten una de sus fronteras; adyacencia por frontera.
- Vecinos indirectos: sólo se tocan en una esquina; adyacencia por esquina.

Una vez que ya se sabe cuáles son los píxeles vecinos de un píxel p_0 , es momento de definir formalmente el concepto de vecindad, que será realmente una submatriz M_{KL} llamada V_p de tamaño $K \times L$, en donde K y L son números enteros e impares que forman parte de la matriz de la imagen I_{MN} . La expresión formal es la siguiente:

$$V_p = \{p : p \in M_{KL}\}; M_{KL} \subset I_{MN}; K = L = 3, 5, 9 \quad (3.15)$$

El tamaño de la matriz de vecindad, depende fundamentalmente del tipo de técnica que se vaya a emplear, por lo que podemos encontrar vecindades de 4×4 , 5×5 y hasta 9×9 . Con el fin de dotar de dos ejemplos prácticos, a continuación se adjuntan dos imágenes en la que aparece en primer lugar una vecindad de 4 (formada esencialmente por vecinos directos), y una vecindad de 8 (constituido por vecinos directos e indirectos):

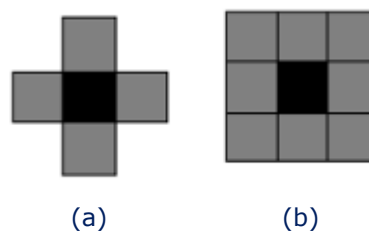


Figura 3-10 – Vecindad (a) Vecindad-4; (b) Vecindad-8

3.8.2 – Conectividad

En este sentido, resulta muy importante el concepto de *conectividad*, ya que es uno de los elementos fundamentales para la detección de objetos separados en una imagen con componentes conectados. En lo que concierne al ámbito de la visión por computador, el estudio de la conectividad se suele aplicar a imágenes binarias, y dicho concepto hace alusión a conjuntos de píxeles accesibles mediante una relación de adyacencia, que puede ser aplicada tanto a los píxeles que constituyen los objetos (*foreground*) como a los que forman parte del fondo (*background*).

Desde un punto de vista pragmático, es posible afirmar, que la conectividad se emplea para definir los límites en las regiones de componentes que forman parte de una imagen. No obstante, para determinar su conectividad, no basta sólo con hacer el estudio de su adyacencia en un sentido específico (si posee 4 vecindades), sino que resulta necesario también observar el valor $f(x,y)$, es decir, el nivel de gris o de intensidad luminosa. Por consiguiente, en una imagen binaria podemos tener una vecindad de 4, pero se consideran conectados si tienen el mismo valor (ó 0 ó 1). Por lo tanto, se debe satisfacer también el criterio de *similitud* en lo que al valor de cada píxel se refiere.

Los niveles de gris que denotaremos como G , utilizados a la hora de definir la conectividad pueden ser muy diversos. Si se trabaja con una imagen binaria –algo que suele ser lo habitual como sucede en nuestro caso–, tenemos que $G = \{1\}$. En este sentido, se puede afirmar que existen tres tipos de conectividad distintos:

- Conectividad-4: dos píxeles, a y b con un nivel de gris G , estarán conectados sí y sólo si a pertenece a $N_4(p)$.
- Conectividad-8: dos píxeles, a y b con un nivel de gris G , estarán conectados sí y sólo si a pertenece a $N_8(p)$.

- Conectividad mezclada: dos píxeles, a y b con un nivel de gris G , estarán conectados si se cumple alguna de las siguientes condiciones:
 - $a \in N_4(p)$
 - $a \in N_D(p) \wedge (N_4(p) \cap N_D(p)) = \emptyset$

La conectividad mezclada, es una alteración de la conectividad-8, que surge para eliminar las conexiones multi-trayectoria, teniendo en cuenta que la trayectoria de un píxel p con coordenadas (x,y) , a otro píxel z con coordenadas (s,t) , es una secuencia de distintos píxeles en donde $(x_0, y_0) = (x,y) \wedge (x_n, y_n) = (s,t)$

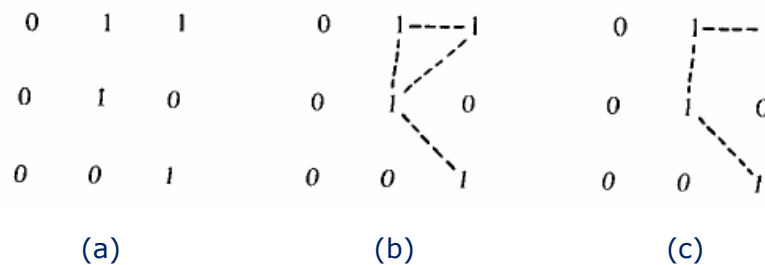


Figura 3-11 – Conectividad (a) Arreglo Píxel; (b) Conectividad-8; (c) Conectividad mezclada

Una vez se han determinado los conceptos más básicos y elementales sobre los que se fundamentan todas las técnicas de detección de regiones, pasamos a describir brevemente como se lleva a cabo en líneas generales el etiquetado de componentes conectados.

3.8.3 – Etiquetado de componentes conexas

A la hora de perpetrar el proceso de etiquetado de componentes conexas, se parte del hecho de que la imagen está binarizada, y que el objetivo final es obtener una región. Normalmente, el etiquetado de componentes conexas, al igual que muchas técnicas del área de la visión por computador, se suele llevar a cabo dentro de un bucle anidado con coste asintótico $O(n^2)$. Este bucle -al igual que sucede cuando se aplica un filtro mediante *convolución* a partir de una matriz llamada *kernel*-, sirve para procesar la imagen haciendo un barrido cuya orientación va de izquierda a derecha y de arriba a abajo, en el que se asumen una determinada conectividad.

Si tomamos como punto de partida la orientación del barrido anterior, y que p hace alusión a un píxel objeto del proceso en cualquier instante de barrido, y u y l son respectivamente los vecinos arriba e izquierda de p , se garantiza que los píxeles u y l ya habrán sido encontrados y etiquetados en el caso de que su valor sea 1, si se supone una conectividad-5. En este aspecto, los procedimientos que se pueden realizar según el tipo de conectividad, son los siguientes:

- Conectividad-4: Si $p = 0$ se mueve a la siguiente posición del barrido (un píxel a la derecha, o el primer píxel que hay a la izquierda en la fila de abajo, en caso de haber llegado al final de una fila). Si es $p = 1$, entonces debemos analizar el valor de sus vecinos u y l ; si ambos valen 0, se asigna una etiqueta a p ; si alguno de los dos vale 1, se asigna otra etiqueta a p ; si ambos valen 1 y tienen la misma etiqueta, se asigna dicha etiqueta a p especificando que son equivalentes.

Al finalizar el proceso, todos los píxeles con valor 1 habrán sido marcados, pero sólo algunos de ellos habrán sido especificados como equivalentes. Por último, se asignan todos los pares que sean equivalentes en pares equivalentes de clase, se proporciona una etiqueta diferente a cada clase, y se lleva a cabo un segundo barrido reemplazando su etiqueta por la de la clase de equivalencia.

- Conectividad-8: El algoritmo de conectividad-8 es similar al de conectividad-4, pero en este caso también deben tenerse en cuenta las dos vecindades diagonales superiores de p , que denominaremos j y k , y que por consiguiente también serán examinadas. Si $p = 0$, se mueve el puntero a la siguiente posición. Sin embargo, si $p = 1$ entonces hay que analizar las cuatro vecindades, es decir, u, l, j, k .

Si las cuatro vecindades son 0, se asigna una nueva etiqueta a p ; si sólo una de ellas es 1 entonces se asigna su etiqueta a p ; si dos vecindades son 1, se asigna una de las etiquetas a p y se anotan las equivalencias oportunas. Una vez finalizado el barrido de la imagen, se lleva a cabo el algoritmo descrito en la conectividad-4.

En la actualidad existen diversos algoritmos para el etiquetado de componentes conexas. Estos algoritmos, se fundamentan en el concepto de conectividad (que como se ha visto básicamente es conectividad-4 y conectividad-8) y procesan una fila de la imagen en cada instante asignando nuevas etiquetas al primer píxel de cada componente, e intentan propagar la etiqueta de un píxel a sus vecinos. Si tomamos como base a Pajares y Cruz (2007) [1], podremos observar a través de los capítulos 7.3.1 – *Un algoritmo iterativo*, y 7.3.2 – *Un algoritmo clásico*, el análisis exhaustivo de dos de los algoritmos empleados en el proceso de etiquetado de componentes conexas:

- Algoritmo iterativo: consta de un paso de inicialización, más una secuencia de propagación de etiquetas de arriba-abajo, seguida por una propagación de etiquetas de abajo-arriba de forma iterativa hasta que no haya cambios de etiquetas. Está formado por tres funciones básicas: *NuevaEtiqueta*: genera una nueva etiqueta con valor entero; *Vecinos*: devuelve el conjunto de vecinos ya etiquetados de un determinado píxel en su misma línea o en la línea previa; *Etiquetas*: cuando se le pasa como parámetro un conjunto de píxeles ya etiquetados devuelve el conjunto de sus etiquetas.
- Algoritmo clásico: Se denomina de esta manera, porque está basado en el algoritmo de componentes conexas para grafos, descrito por *Rosendfeld y Pfaltz* (1966) [31]. Se realiza en dos pasos y consta de una tabla de equivalencias que se necesita mantener. En el primer paso se realiza la propagación de etiquetas, y en el segundo se asigna a cada píxel la etiqueta de la clase de equivalencia del paso anterior. Su funcionamiento es análogo al algoritmo que se describió en este mismo apartado en relación a la conectividad-4 y conectividad-8.

Todos estos algoritmos se basan en el concepto de componente conexa, que se puede definir como: *el conjunto de píxeles que tienen valor binario 1, y que están conectados entre sí por un camino o conjunto de píxeles que tienen también valor binario 1; a dichos píxeles se les asigna la misma etiqueta identificativa, que ha de ser única para todos los píxeles que constituyen una región y que por consiguiente sirve como identificador único*. El objetivo final es dividir la imagen original en una serie de regiones numeradas, tal y como se muestra en la figura 3-12:

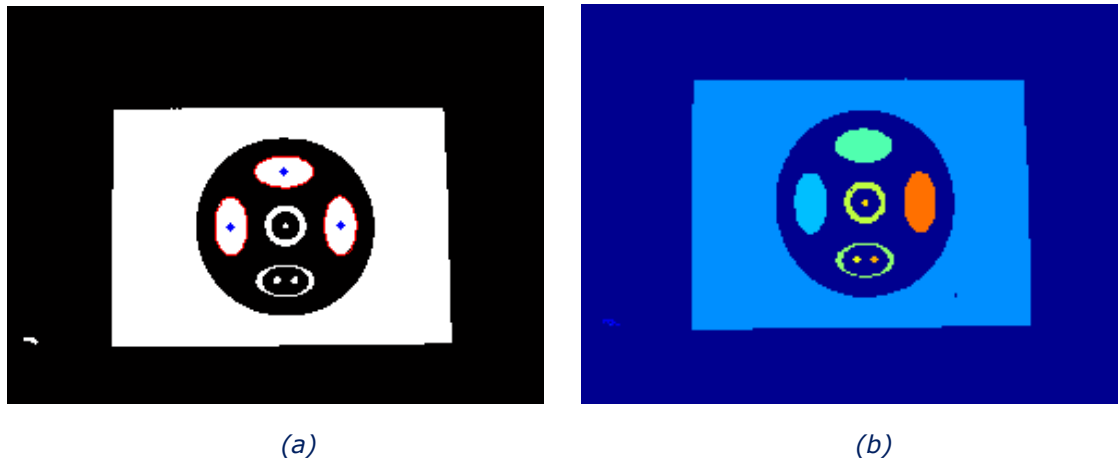


Figura 3-12 – Segmentación (a) Imagen binarizada; (b) Etiquetado componentes

En la figura 3-12 (a), puede observarse el resultado de binarizar la imagen original de la plataforma de aproximación una vez se han detectado previamente las elipses gracias al proceso de etiquetado de regiones conexas, cuyo resultado final se puede observar en la imagen (b). En este sentido, cabe destacar, que todo el proceso de dibujado de las elipses y de sus correspondientes centros, se debe esencialmente al resultado que genera el proceso de etiquetado de componentes conexas, ya que en realidad lo que se ha hecho es emplear una forma paramétrica de la ecuación de la elipse, para trazar su contorno sobre la imagen a partir de las propiedades que se obtienen de cada una de las regiones. Las propiedades utilizadas en la señalización del perímetro obtenidas con la función *regionprops* de la Image Toolbox de MATLAB son: orientación, centroEjeX, centroEjeY, longitudEjeMayor y longitudEjeMenor.

Finalmente, cabe reseñar, que cada una de las regiones que aparecen coloreadas en la imagen (b) correspondiente a la figura 3-12, tienen asociado un identificador único interno que identifica de forma unívoca cada una de las regiones dentro del sistema. Cada color hace alusión a un número de región distinto, que aunque en la imagen no se puede observar, en MATLAB sí que se puede ver si se pasa el ratón por encima de cada una de las regiones al haber usado la opción *impixelinfo*. Así tal y como se puede observar, las tres elipses tienen tres colores distintos (azul, verde y naranja) que se corresponderían con las regiones 6, 9 y 15 respectivamente.

3.9 – Descripción de Regiones

Una vez se ha realizado el proceso de segmentación, se obtendrán una serie de regiones en las que, si todo ha salido conforme a lo esperado, deberán encontrarse las diversas figuras geométricas (elipses y círculos eminentemente) que constituyen la plataforma de aproximación. Por consiguiente podemos postular, que el ámbito de descripción de regiones, es una disciplina de la visión por computador que tiene como objetivo básico, el estudio de descriptores que permitan extraer información de una región con el objetivo de describirla y por consiguiente reconocerla.

Gracias a las cualidades inherentes al cerebro humano, es factible que se tienda a trivializar un problema tan complejo (enfocándolo desde el punto de vista de una máquina) como es el relativo a la descripción e identificación de figuras y formas. En este sentido, para un ser humano, resulta básico observar e identificar miles de objetos y figuras, y dictaminar en tiempo real de qué elemento se trata, ya que es

un aprendizaje que el cerebro va adquiriendo progresivamente desde que se nace. Sin embargo, para una máquina, el proceso no resulta baladí en ninguno de los aspectos, y obliga a recurrir a las matemáticas, ya que es la mejor herramienta de la que se dispone para describir de una forma no ambigua el mundo que nos rodea.

Desde un punto de vista formal, una región se puede describir por la forma de su frontera, o por sus características internas. De acuerdo a los criterios de *Duda y col.* (2000) [32], la descripción de regiones trata de describir la forma de una figura, que ha sido extraída del plano de la imagen. Para ello, dependiendo del criterio o del punto de vista que se desee utilizar, una región puede describirse de distintas maneras, lo que se traduce en el uso de propiedades que atienden a criterios dispares.

3.9.1 – Propiedades métricas

Aquí se estudiará la descripción de las regiones, con el propósito de llegar a su identificación o caracterización. En primera instancia, se hablará acerca de las propiedades de las figuras que dependen de una métrica con el fin de poder obtener descriptores geométricos, que sirvan de base para tratar de identificar la figura que se encuentra plasmada sobre la plataforma de aproximación. En este sentido, cabe subrayar, que las métricas pueden ser consideradas como generalizaciones de la distancia euclídea, por lo que su valor estará vinculado al plano donde se encuentre implementada la figura, y variará dinámicamente a medida que ésta lo haga. Esto quiere decir, que si el plano de la figura se distorsiona, el valor de la métrica se alterará como consecuencia de esta relación de dependencia existente.

Esta característica provoca, que inicialmente los descriptores que se encuentran basados en propiedades métricas, puedan ser considerados como no aptos para tratar de llevar a cabo la identificación y reconocimiento de nuestra figura. Sobre todo, si se trata de diseñar un sistema flexible y que resulte competente a la hora de interactuar con diversas imágenes de la figura que atiendan a una gran variedad de perspectivas heterogéneas. Esta pluralidad en las perspectivas, que en realidad son transformaciones del plano, surgen como consecuencia de haber llevado a cabo el proceso de adquisición en un entorno no controlado, es decir, en una situación que implica usar diferentes distancias, niveles de zoom, y ángulos de inclinación por la idiosincrasia en el funcionamiento del UAV. Esto es todo lo contrario a lo que sucede, por ejemplo, en el proceso de adquisición de un sistema de reconocimiento de matrículas de vehículos (APLR), donde la luminosidad, la distancia y el ángulo de inclinación suele ser uniformes y están dentro de unos valores muy acotados.

Sin embargo, si se tiene en cuenta desde un inicio que las propiedades métricas que se encuentran basadas en distancias, ya sea de forma directa o indirecta (como en el caso del área, el paralelismo de curva, etcétera), pueden distorsionarse por proyecciones topológicas, y que su valor se modificará como consecuencia de las transformaciones en el plano, es posible tratar de idear una estrategia cimentada en *estimaciones, umbrales y ratios* (relaciones entre regiones), que aún basándose en descriptores geométricos típicos de las propiedades métricas, sea válida para reconocer la figura en un entorno dinámico y que atienda a diversas perspectivas. No obstante, para poder definir con exactitud los umbrales, estimaciones y ratios, es necesario realizar un estudio geométrico exhaustivo de la figura, tal y como ha quedado patente a través de la figura 6-1 que se expone más adelante.

En este sentido, algunas de las propiedades métricas más troncales y relevantes que han sido empleadas para estudiar las relaciones entre las figuras que forman

parte de la plataforma (elipses y círculos eminentemente), son las siguientes que se describen brevemente a continuación:

1. **Área:** establece el número de píxeles contenidos dentro de la frontera, por lo que su valor está radicalmente vinculado a la resolución de la fotografía. Suponiendo el mismo *ratio figura/imagen*, donde la figura es el elemento del que se quiere calcular el área, y la imagen es el plano donde está contenida la figura (su área se obtendría multiplicando el total de filas por columnas), el área será mayor cuanto más resolución tenga la imagen.
2. **Perímetro:** es la longitud de la frontera y aunque algunas veces se usa como descriptor, su aplicación más habitual se designa de cara a la obtención de la *compatibilidad* de una región que se define como P^2 / A . Cabe reseñar, que éste es un valor sin dimensiones que es mínimo para una región en forma de disco. La expresión para obtener el perímetro es la siguiente:

$$P = \sum_i \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (3.16)$$

3. **Centro de Gravedad:** es un punto cartesiano unívoco de representación de la región que se calcula a partir de los píxeles de la imagen. Como regla general, se utiliza la media aritmética para su cálculo, puesto que se le otorga el mismo peso específico a cada uno de los píxeles:

$$\bar{x} = \frac{1}{A} \sum_i x_i, \quad \bar{y} = \frac{1}{A} \sum_i y_i \quad (3.17)$$

Como se puede observar a partir de la ecuación 3.17, se le ha dado el mismo peso a todos los píxeles que intervienen, aunque de vez en cuando debe tenerse en mente, que puede resultar provechoso hacer uso de la intensidad en cada punto, lo que provocaría el siguiente cambio respecto a la expresión anterior:

$$\bar{x} = \frac{\sum_i f(x_i, y_i) x_i}{\sum_i f(x_i, y_i)}, \quad \bar{y} = \frac{\sum_i f(x_i, y_i) y_i}{\sum_i f(x_i, y_i)} \quad (3.18)$$

4. **Excentricidad:** la excentricidad de una figura puede definirse como el cociente que resulta de dividir el eje menor y el eje mayor de una región, los cuales se definen en términos de su frontera. Suele ser un descriptor muy útil por sí sólo, tal y como ha sido el caso en la solución propuesta, o como base para calcular la orientación de una figura.
5. **Orientación:** la orientación de una figura es una métrica muy importante, y en nuestro caso ha sido utilizada para poder determinar el ángulo de giro necesario para que la perspectiva sea la misma que aparece reflejada en la figura 3-1. Su valor se suele determinar a partir de los ejes mayor y menor de la región, aunque también se puede ajustar una recta que sirva como base para determinar la orientación de la misma en relación al ángulo que ésta forma. Dicho ángulo puede obtenerse de la siguiente manera:

$$S_x = \sum x_i, \quad S_y = \sum y_i, \quad S_{xx} = \sum x_i^2, \quad S_{yy} = \sum y_i^2, \quad S_{xy} = \sum x_i y_i$$

$$M_{xx} = S_{xx} - \frac{S_x^2}{A}, \quad M_{yy} = S_{yy} - \frac{S_y^2}{A}, \quad M_{xy} = S_{xy} - \frac{S_x S_y}{A} \quad (3.19)$$

Si se toman como premisas las expresiones que acaban de ser expuestas, la orientación viene dada por la siguiente ecuación que está cimentada en la función trigonométrica inversa a la tangente, es decir, la *arcotangente*:

$$\phi = \arctan = \left\{ \frac{M_{xx} - M_{yy} + \sqrt{(M_{xx} - M_{yy})^2 + 4M_{xy}^2}}{2M_{xy}} \right\} \quad (3.20)$$

3.9.2 – Propiedades topológicas

Como ya se ha comentado con anterioridad, dependiendo del punto de vista que se emplee, una región puede ser descrita de diversas maneras. Por un lado puede ser vista o comprendida como un conjunto de puntos conectados entre sí, ya que si se parte de un punto de la región es posible llegar a otro punto de la misma sin abandonarla. Por otro lado, también puede ser descrita por el número de huecos que presenta, aunque en ambos casos estaríamos hablando de las propiedades topológicas de una región.

Una de las ventajas que ofrecen las propiedades topológicas, es que se caracterizan por ser invariantes a ciertas deformaciones en el plano de la imagen, lo que quiere decir que no se pueden emplear bajo concepto alguno, nociones basadas en la distancia (ya sea directa o indirectamente), puesto que ésta es una característica intrínseca de las propiedades métricas, las cuales no resultan invariantes ante dichas distorsiones, conocidas formalmente como *homeomorfismo*. Algunas de las propiedades topológicas más relevantes son las siguientes:

1. Componente Conexa: una componente conexa de un conjunto, se puede definir como un subconjunto de dimensión máxima tal que dos de sus puntos cualesquiera, pueden unirse por una curva continua y sin rupturas perteneciente enteramente al subconjunto.
2. Número de Euler: este número, que resulta invariante frente a traslaciones, rotaciones y cambios de escala, permite de forma sencilla discriminar entre ciertas clases de objetos. Se define como: $E = C - H$, donde C es el número de componentes conexas y H el número de huecos de la imagen.
3. Número de Huecos: en visión por computador, un hueco es una región de la imagen que está completamente encerrada por una componente conexa. Formalmente, el número de huecos en la figura es uno menos que el número de componentes conexas en el complemento de la figura.

Los descriptores basados en propiedades topológicas encuentran su ámbito de aplicación como filtros preliminares, lo que quiere decir, que frecuentemente se encuentran acompañados de otros descriptores más precisos que permitan un nivel de refinamiento más elevado en base a un análisis de granularidad fina. Por ejemplo, en el reconocimiento de caracteres, si el número de Euler no coincide con un determinado patrón, el carácter ya no será comparado.

En el caso que nos ocupa, se ha utilizado el número de Euler como base para filtrar regiones antes de llevar a cabo un análisis basado en las propiedades métricas que están cimentadas en las distancias euclídeas que hay entre los centros de gravedad de las regiones. Concretamente, se ha utilizado primero en el procesado relativo a la identificación de elipses, y acto seguido en el de la identificación de círculos, de tal forma que si no se cumplen con los valores esperados en las disyunciones

lógicas comparativas (1 en el caso de las *elipses* y -2, -1, 0 y 1 en el caso de los *círculos*), ya no se considera procedente que sea necesario realizar los estudios basados en las propiedades métricas (distancias, áreas y ángulos que forman los centros de gravedad de las regiones). De esta manera, se consigue un algoritmo más eficiente, ya que se puede reducir el número de casos en los que se debe calcular la distancia euclídea, una operación que tiene coste aritmético alto para la ALU (Arithmetic Logic Unit).

3.9.3 – Momentos Geométricos

En primer lugar es necesario acentuar, que la técnica descrita por medio de este subapartado, ha sido utilizada implícitamente por la función *regionprops* que forma parte de la *Image Processing Toolbox* de MATLAB en el proceso de orientación de la imagen descrito a través del apartado 4-2 – *Problemas de Orientación*, que forma parte de este mismo documento. Este proceso básicamente lo que permite, es rotar la imagen tantos grados como sea necesario, para realizar el proceso de evaluación e identificación de las figuras desde la perspectiva que se muestra en la figura 3-1. De esta manera, el análisis se lleva a cabo utilizando siempre la misma orientación, lo que permite trabajar con valores fijos en vez de diferenciales, permitiendo dotar así de mayor claridad al establecimiento de estimaciones y umbrales.

En el ámbito de la visión por computador, un *momento* se puede definir como una media ponderada de las intensidades de los píxeles de una imagen. Son, por tanto, propiedades numéricas que se pueden obtener de una determinada región en la que se tienen en cuenta todos los píxeles de la misma, y no sólo los bordes. Suelen ser empleadas en el ámbito de la visión por computador porque proporcionan valores interesantes y atractivos que pueden ser usados incluso para llevar a cabo la umbralización de una imagen. Siguiendo este razonamiento, tendríamos que el momento más simple que se puede obtener, es el que surge como consecuencia de realizar la suma de todos los valores de los píxeles de una imagen, que de hecho se denomina momento simple de orden 0.

Una vez aclarados estos aspectos preliminares, con la ayuda de Pajares y Cruz (2007) [1], a continuación se describen muy brevemente algunos de los tipos de momentos invariantes más empleados actualmente:

1. Momento de Orden Simple: Si tomamos $f(x,y)$ como la intensidad del punto (x,y) en una región. El momento de orden $(p + q)$ para la región, se define de la siguiente manera:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x,y) \quad (3.21)$$

2. Momento Central de Orden: Permiten reconocer figuras dentro de una imagen independientemente de su posicionamiento. El momento central de orden $(p + q)$ viene dado por la siguiente expresión:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x,y) \text{ en donde } x = \frac{m_{10}}{m_{00}} \quad y = \frac{m_{01}}{m_{00}} \quad (3.22)$$

A partir de esta expresión se obtienen los momentos centrales normalizados de orden $(p + q)$, que servirán de base para hallar los momentos invariantes de *Hu*:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \text{ donde } \gamma = \frac{p+q}{2} + 1 \text{ para } (p+q) = 2,3,\dots \quad (3.23)$$

3. Momentos Invariantes de Hu: Los siete momentos invariantes de Hu , denominados así por haber sido propuestos por el propio Hu (1962) [33], constituyen una herramienta muy versátil y útil en el ámbito de la visión por computador, porque resultan ser invariantes a cambios en la traslación, rotación y escala. Son un conjunto de momentos normalizados con respecto al momento de orden 0, por lo que se pueden hallar usando los momentos centrales normalizados de órdenes 2 y 3. :

$$\begin{aligned}
 \phi_1 &= \eta_{20} + \eta_{02} \\
 \phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
 \phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
 \phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
 \phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
 \phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
 \phi_7 &= (3\eta_{21} - \eta_{03})^2(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
 \end{aligned} \tag{3.24}$$

Sin embargo, para que todos los momentos contribuyan por igual en la función de discriminación, y por tanto se encuentren dentro del mismo orden de magnitud, a veces es necesario proceder a una normalización:

$$\phi_n = \text{absIn}(\text{abs}(\phi_n)) \tag{3.25}$$

4. Momentos Invariantes Afines: Se derivan a partir de la teoría de invariantes algebraicos, y son invariantes bajo una transformación general afín. La demostración y su obtención se puede encontrar en *Flusser y Suk* (1993) [34]. Sin embargo, con el fin de aportar un mayor grado de síntesis al apartado, se ha decidido que no era útil mostrar explícitamente las cuatro expresiones formales que componen el conjunto de los cuatro momentos afines. Una mayor profundidad del tema se encuentra también en *Pajares y Cruz* (2007) [1], donde se muestran tales expresiones formales.

A continuación, se comentan las principales utilidades de los momentos de orden más comunes:

1. Momento Simple de Orden 0: Es la suma de todos los valores de los píxeles, lo que quiere decir que representa el área de la figura en imágenes binarias y la superficie en imágenes en escala de grises.
2. Momento Simple de Orden 1: Se utiliza para calcular el centroide (centro de masas) de una figura, el cual está representado por el momento de orden 0 y 1, es decir, $X = M(1,0) / M(0,0)$; $Y = M(0,1) / M(0,0)$
3. Momentos Centrales de Orden 2: Forman las componentes del tensor de inercia o matriz de rotación, y sirven para calcular el ángulo de rotación de la figura alrededor de su centro de masas, y la excentricidad de la misma.
4. Momentos Centrales de Orden 3: Desempeñan un papel muy importante ya que sirven de base para poder calcular los momentos invariantes que son inalterables frente a traslaciones, rotaciones y cambios de escala.

Tal y como se puede observar, los momentos centrales de segundo orden juegan un papel muy relevante en la solución propuesta, puesto que tanto el ángulo de

orientación como la excentricidad, son dos de las propiedades que han sido usadas ampliamente tanto en el análisis como en la identificación de cada una de los círculos y elipses que forman parte de la figura objeto de estudio de la plataforma de aproximación del UAV. Se debe poner de manifiesto, tal y como se indica a través del apartado 4.2 – *Problemas de Orientación*, que la función *regionprops* de la *Image Processing Toolbox* de *MATLAB*, emplea los momentos de segundo orden a la hora de determinar la orientación de cada una de las regiones en que se haya segmentado la imagen original. Esto quiere decir, que devuelve el ángulo en grados en el intervalo $[-90, 90]$ que hay entre el eje de abscisas *OX* y el eje mayor de la elipse, que tiene los mismos segundos momentos que la región, tal y como se pone de manifiesto a través de la documentación oficial que *MATLAB* pone a disposición de la comunidad de usuarios / desarrolladores.

Tal y como se indicaba en la parte introductoria correspondiente a este subapartado, los momentos pueden ser utilizados incluso como herramientas para umbralización de imágenes, tal y como sucede con la técnica de *umbralización por preservación del momento* (Moment Preserving Thresholding) en la que se eligen valores de modo que los momentos de nivel de gris de una imagen de entrada se conservan en la imagen de salida. El objetivo de la transformación basada en la preservación del momento, es agrupar las diferentes intensidades de los píxeles en un número de clases, de forma que, los valores de cada clase puedan ser representados por un único nivel de gris. A continuación, la ecuación 3.26 muestra una expresión a partir de la cual es factible determinar los momentos en la escala de grises a partir del histograma de una imagen *f*, en donde *n* hace alusión al número de píxeles de la imagen, y *h(i)* representa el número de píxeles con niveles de gris iguales a *i*:

$$m_k = \frac{1}{n} \sum_{i=0}^{255} h(i) \cdot i^k \quad (3.26)$$

3.10 – Descripción de líneas y contornos

El área de visión por computador, ofrece diversas técnicas para tratar el problema relativo a la descripción de líneas y contornos. Normalmente todos estos métodos parten de la base de que existen ciertas partes de las imágenes que han sido aisladas como entidades significativas. Por consiguiente, se supone que una parte de la imagen original, es decir, algún subconjunto de la misma al que se le suele denominar como *figura*, ha sido extraída con anterioridad como resultado de algún proceso previo que puede estar vinculado a la extracción de puntos de interés, extracción de regiones, etcétera. El tratamiento será aplicable a figuras compuestas de puntos discretos, figuras compuestas de líneas, y figuras compuestas de áreas.

Una vez aclarado este aspecto preliminar, y a pesar de que tanto la descripción de líneas como la de contornos no ha sido utilizada de forma explícita en la solución, se ha estimado que resulta oportuno describir en esta memoria como mínimo una técnica que sirve para distinguir las líneas y contornos de una imagen, ya que éste es posiblemente uno de los problemas más recurrentes dentro del área de la visión por computador. Por tal motivo, a continuación se expone la transformada de Hough, que aunque no ha sido utilizada explícitamente en la solución de este TFM, se describe por ser un icono en la resolución de este tipo de problemática.

3.10.1 – Transformada de Hough

La Transformada de Hough es una técnica ampliamente usada en el área de la visión por computador, para realizar un reconocimiento de patrones que permita

distinguir determinadas formas geométricas dentro de una imagen. Esta técnica, que fue patentada inicialmente por *Paul Hough* (1962) [35], sólo se aplicaba a la detección de rectas en una imagen. Sin embargo, dicha patente fue perfeccionada posteriormente por los científicos *Duda y Hart* (1972) [36], para que pudiera ser apta para identificar cualquier elemento de la imagen que lograra ser descrito de forma matemática a través de una serie de parámetros. De ahí, que se acuñara el concepto de *Transformada de Hough Generalizada*. En el caso de tratar con rectas, los parámetros son (ρ, θ) , pero si se trabaja con circunferencias, entonces éstos son (x, y, r) , es decir, el centro y radio de la circunferencia. Permite representar contornos tan complejos como se desee, en tanto en cuanto, éstos puedan ser parametrizables. La transformada de Hough permite descubrir los puntos que forman parte de los bordes mediante la transformación de los puntos fronterizos en el espacio paramétrico donde se convierten en líneas.

Pues bien, si suponemos que para n puntos de una imagen se desea encontrar aquellos subconjuntos de esos puntos que residan en las líneas rectas, una de las posibles soluciones, trata de encontrar inicialmente, todas las líneas determinadas por cada par de puntos, y a continuación encontrar todos los subconjuntos de puntos que están cerca de dichas líneas. Sin embargo, esta solución que resulta bastante trivial, es privativa desde el punto de vista de la eficiencia computacional, puesto que implica encontrar $n(n-1)/2$ líneas y a continuación realizar $(n)(n-1)/2$ comparaciones de cada punto de todas las líneas. La primera operación tendrá prácticamente un coste asintótico cuadrático $O(n^2)$, mientras que la segunda casi será cúbico $O(n^3)$.

Considerando un punto (x_i, y_i) del plano Euclídeo, una recta puede ser representada con la ecuación explícita de la recta que tiene la forma $y_i = ax_i + b$, donde m es la pendiente de la recta -tangente del ángulo que la recta forma con el eje OX -, y n es la ordenada en el origen -coordenada y del punto en el que la recta corta al eje OY -. A través de (x_i, y_i) pasan un número infinito de líneas, pero todas ellas satisfacen la ecuación $y_i = ax_i + b$ para diversos valores de a y b . Sin embargo, escribiendo esta ecuación en forma de espacio de parámetro $b = -x_i a + y_i$ en el plano OAB , se obtiene la ecuación de una única línea para un par determinado (x_i, y_i) . Además, un segundo punto (x_j, y_j) , también tiene una línea en el espacio parámetro asociado con él, y esta línea corta a la línea asociada con (x_i, y_i) en (a', b') , donde a' es la pendiente, y b' la ordenada en el origen de la línea que contiene a (x_i, y_i) y (x_j, y_j) en el espacio OXY . De hecho, todos los puntos contenidos en esta línea tienen líneas en el espacio parámetro que cortan a (a', b') tal y como se observa a continuación:

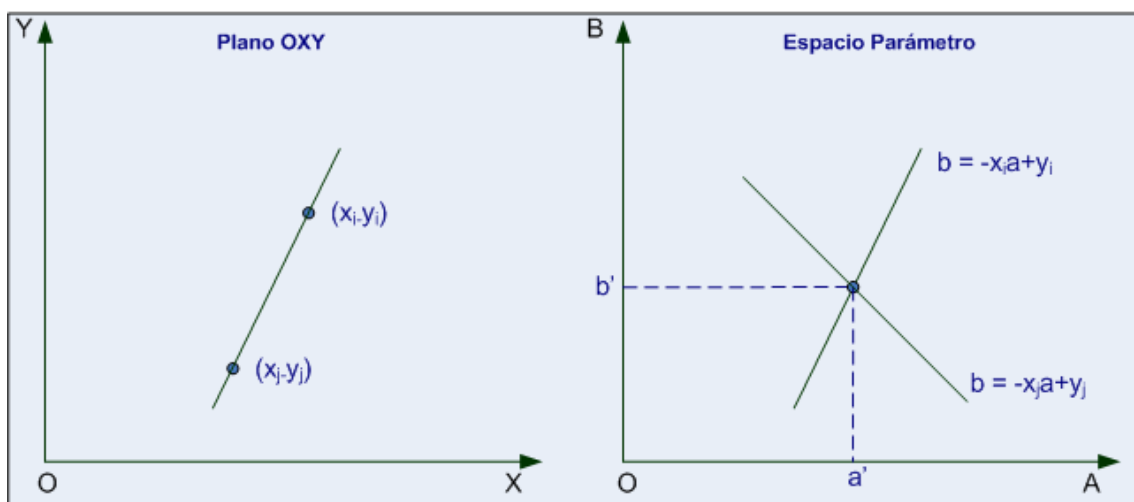


Figura 3-13 – Transformada de Hough

Una de las características intrínsecas del cálculo de la Transformada de Hough, surge de la subdivisión del espacio parámetro OAB en las denominadas celdas de acumulación donde $(a_{máx}, a_{mín})$ y $(b_{máx}, b_{mín})$, son los rangos esperados de los valores de la pendiente y la ordenada. La célula de coordenadas (i, j) , con valor de acumulador $A(i, j)$, corresponde al cuadrado asociado con las coordenadas del espacio de parámetro (a_i, b_j) . Inicialmente estas celdas toman el valor cero, aunque después, para cada punto (x_k, y_k) del plano imagen, se fija el parámetro a , igual a cada uno de los valores permitidos de subdivisión sobre el eje OXY , y se resuelve para el b correspondiente utilizando la ecuación $b = -ax_k + y_k$. Las b resultantes se redondean al valor más próximo permitido del eje OX . Si una elección de a_p resulta ser la solución, b_q se fija $A(p, q) = A(p, q) + 1$. Al finalizar el procedimiento un valor de M en $A(i, j)$, corresponde a M puntos del plano OXY situados en la línea $y = a_i x + b_j$.

Por consiguiente, el modo de funcionamiento es estadístico, y de acuerdo a los puntos que se tengan, se debe averiguar las posibles líneas en las que puede estar ubicado el punto. Esto se logra por medio de una operación que se aplicada a cada línea en un rango determinado, lo que quiere decir, que el número de subdivisiones que se hagan en el plano ab juega un papel primordial. Si al evaluar la ecuación de la recta para cada punto de la imagen (x_k, y_k) ésta se cumple, entonces se incrementa en uno el número de votos de la celda, teniendo en cuenta que un número de votos elevado indica que el punto pertenece a la recta.

Sin embargo, un problema que aparece al utilizar la ecuación $y = ax + b$ para representar una línea, es que tanto la pendiente como la ordenada en el origen, tienden al infinito cuando la línea se acerca a la vertical. Por ello, una forma de evitar esto, es utilizar la representación polar de la recta, la cual se representa a partir de los parámetros ρ y θ , donde ρ es la distancia entre la línea y el origen, y θ será el ángulo que forma el vector que va desde el origen al punto más cercano. A través de la parametrización, la ecuación de la recta puede ser escrita del siguiente modo: $\rho = x \cos \theta + y \sin \theta$. Para poder emplear dicha expresión, hay que realizar una transformación de las coordenadas cartesianas (x, y) que definen los píxeles de la imagen, a las coordenadas polares asociadas al espacio de parámetros (ρ, θ) .

3.11 – Geometría y Distancia

En este apartado se realiza un breve estudio acerca de los principales patrones geométricos que se han utilizado como eje fundamental para poder efectuar la identificación de la plataforma de aterrizaje. Por otro lado, en lo que a la vertiente trigonométrica que corresponde, se ha estimado que sería oportuno describir en qué consiste realmente la distancia euclídea, y cuál ha sido su ámbito de aplicación en la solución propuesta.

3.11.1 – Elipse

En el ámbito de la visión por computador, la detección de elipses es considerada como una de las actividades más importante en lo que al reconocimiento de patrones se refiere. De hecho, durante las dos últimas décadas (1990-2010), los investigadores han desarrollado diversos enfoques para la detección de elipses, los cuales pueden ser divididos en dos grandes grupos: *clustering* (Yin, 1998 [37]) (Dave, 1992 [38]) y *optimización* (Fitzgibbon y col. 1999 [39]), (Zhang y Liu, 2004 [40]). En la solución propuesta, esta figura geométrica desempeña un papel decisivo, porque el dibujo de la plataforma de aterrizaje, tiene impresas varias

elipses que será necesario identificar convenientemente de cara al reconocimiento final de la plataforma. La detección pues de las elipses E1, E2, y E3 es fundamental.

Sin embargo, el objetivo de este apartado, no es tanto proporcionar un enfoque riguroso acerca de cuáles han sido los fundamentos teóricos empleados durante la identificación de las tres elipses con fondo blanco que componen la figura, sino dar a conocer las principales características de la elipse como entidad desde un punto de vista geométrico. A medida que se vayan describiendo tales características, se irá comentando cual ha sido la utilidad y la relevancia que ha desempeñado cada una de ellas, en lo que al proceso de reconocimiento se refiere.

Pues bien, una elipse se puede definir como una figura formada por una línea curva, cerrada y plana, que constituye un lugar geométrico de todos los puntos de un plano, tales que la suma de las distancias a otros dos puntos fijos llamados focos, es una constante positiva ($2a$, donde a es la medida del semieje mayor de la elipse). Esto se traduce en que, dado un punto P situado en la curva de la elipse, y dos focos F y F' , la suma de distancias de P a F y F' es constante. Desde un punto de vista descriptivo, una elipse resulta similar a un círculo achatado que se puede obtener como resultado de cortar la superficie de un cono por un plano oblicuo al eje de simetría, tal y como se muestra en la imagen 3-14 (a) que se muestra a continuación:

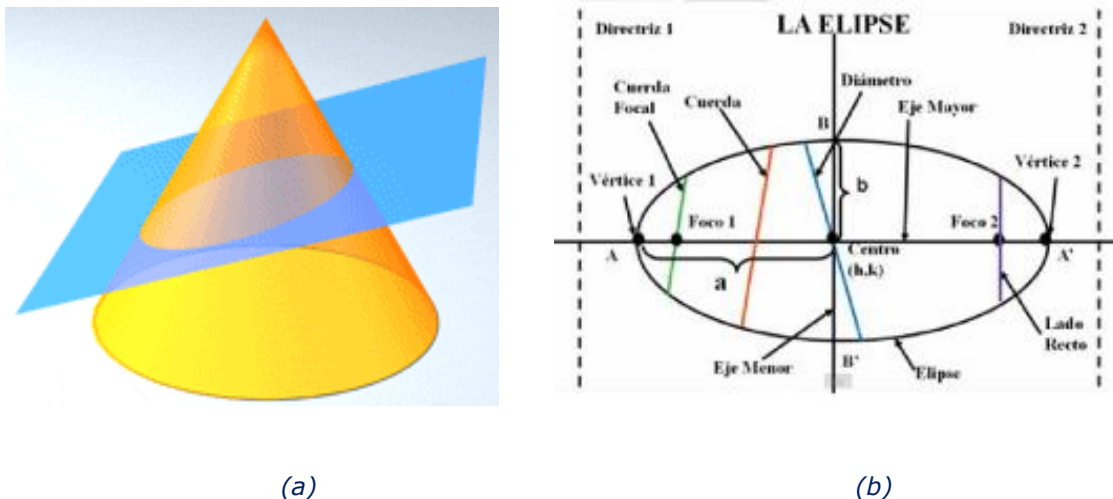


Figura 3-14 – Elipse (a) Relación con el Cono; (b) Elementos principales

A partir de la imagen 3-13 (b) que contiene los elementos principales de una elipse, es posible realizar una descripción formal de cada uno de estos elementos, así como de otro tipo de atributos usados en el análisis de la plataforma de aproximación, tales como: la excentricidad, el área y sus principales ecuaciones:

1. **Área:** Hace alusión a la superficie que existe en el interior de una elipse, y su cálculo viene determinado por la siguiente expresión matemática:

$$A = \pi \cdot a \cdot b \quad \text{siendo } a, b \text{ los semiejes de la elipse} \quad (3.27)$$

El área de la elipse juega un factor importante en el reconocimiento de la plataforma, ya que es uno de los atributos usados en el filtrado preliminar, y en el proceso de asignación de probabilidades para hacer evaluación de la figura. Si las tres elipses con fondo blanco (E1, E2, y E3) tienen un área que está dentro del intervalo definido (dicho intervalo se obtiene a partir de los

valores de las elipses y un umbral de error basado en porcentaje), entonces se da por validado este análisis parcial comparativo (denominado APC4), ya que quiere decir que pasa el examen de analogía en lo concerniente al área.

2. Excentricidad: La excentricidad ε (épsilon) de una elipse es la razón entre su semidistancia focal (longitud del segmento que parte del centro de la elipse y acaba en uno de sus focos), denominada por la letra c , y su semieje mayor. Su valor se calcula por medio de cualquiera de las expresiones que hay planteadas en la ecuación 3.28:

$$\varepsilon = \frac{c}{a}, \text{ y dado que } c = \sqrt{a^2 - b^2}, \text{ entonces } c = \sqrt{1 - \left(\frac{b}{a}\right)^2}$$

$$\left\{ \begin{array}{l} \varepsilon = \frac{c}{a} \\ c = \sqrt{1 - \left(\frac{b}{a}\right)^2} \end{array} \right. \text{ en donde } (0 \leq \varepsilon \leq 1) \quad (3.28)$$

La excentricidad de la elipse, al igual que el área, también juega un papel relevante en el reconocimiento de la plataforma de aproximación, dado que es el descriptor geométrico empleado en un análisis parcial comparativo (denominado APC5) que es conmensurable, y por ende evaluable, de cara a la asignación final de probabilidad. A través de esta prueba, se chequea que la excentricidad de las tres elipses (E1, E2, y E3) que forman parte de la figura objeto de reconocimiento, se encuentren dentro del intervalo definido por sus valores y un umbral de error basado en porcentaje. Además, dado que un círculo en una perspectiva con un ángulo elevado de inclinación, es visto como si fuera una elipse, también se utiliza la excentricidad como clasificador para decidir qué tipo de ecuación emplear (paramétricas de la elipse o la propia de la circunferencia) en el dibujado del contorno de los círculos.

3. Puntos y elementos principales: A continuación se hace una descripción de los principales puntos y elementos de una elipse desde un punto de vista geométrico:
- a. Focos: son dos puntos equidistantes del centro denominados según la figura 3-14 (b) como F y F' . Se encuentran emplazados en el eje mayor y tienen una cualidad muy importante, ya que la suma de las distancias desde cualquier punto P situado en la curva de la elipse a los dos focos es constante, e igual a la longitud del diámetro mayor definida por la siguiente expresión, en donde a representa el semieje mayor de la elipse:

$$d(P, F) + d(P, F') = 2a \approx \overline{PF} + \overline{PF'} = 2a \quad (3.29)$$

- b. Eje mayor y eje menor: el eje mayor y el eje menor de una elipse, se corresponden con los segmentos que unen los puntos A y A' , y B y B' respectivamente. El eje mayor tiene un valor de $2a$, y equivale al resultado de la suma de las distancias de cualquier punto a los focos, la cual es constante. El eje menor tiene un valor de $2b$. Los dos ejes son perpendiculares entre sí, y se cortan en el centro de la elipse, también comúnmente denominado como centro de simetría. En las dos expresiones que se presentan por medio de la ecuación 3.30, a y

b se corresponden con los valores de los semiejes mayor y menor respectivamente.

$$\text{EjeMayor} = \overline{AA'} = 2 \cdot a \quad \text{EjeMenor} = \overline{BB'} = 2 \cdot b \quad (3.30)$$

Los valores de los ejes de una elipse resultan muy relevantes para la solución propuesta, ya que son la fuente de datos principal para determinar cuál es la orientación de una elipse respecto al eje OX. De hecho, el número de grados que es necesario rotar la imagen, para que los criterios de análisis puedan ser procesados siempre con la orientación que se muestra en la figura 3-1, se obtiene a partir de la orientación de las tres elipses (E1, E2, y E3), una vez éstas han sido previamente reconocidas. Dicha orientación es fruto de operar con los valores asociados a la longitud del eje mayor y menor.

- c. Distancia y semidistancia focal: la distancia entre los focos FF' se denomina distancia focal y tiene un valor de $2c$, que coincide con la expresión $2\varepsilon a$, siendo ε la excentricidad de la elipse. Por el contrario, la semidistancia focal, es la longitud del segmento que parte del centro de la elipse y acaba en uno de sus focos; se identifica con la letra c . Su valor es justo la mitad de la distancia focal.

$$\text{DistanciaFocal} = \overline{FF'} = 2 \cdot c = 2 \cdot \varepsilon \cdot a; \quad \text{SemidistanciaFocal} = c \quad (3.31)$$

4. Ecuaciones de la elipse: Como es habitual en geometría, existe más de una expresión matemática para representar el contorno de una misma figura. No obstante, dado que todas estas ecuaciones son bastante conocidas, se ha estimado oportuno no incluirlas como parte de esta memoria. Sin embargo, sí que se debe poner de manifiesto, la utilización explícita de las *ecuaciones paramétricas de la elipse* durante el proceso de identificación de la figura. Su ámbito de aplicación se circunscribe a dibujar el contorno de las elipses (E1, E2, y E3) y los círculos (C1, C2, C3, y C4), siempre que en este último caso la excentricidad del círculo sea superior al valor establecido en la variable de configuración, *Configuracion.UmbralDibujoCirculo*, que por defecto es de 0.4. En caso contrario, se emplea la ecuación paramétrica de la circunferencia. Dicho esto, pasamos a mostrar las ecuaciones paramétricas de la elipse (ecuación 3.32), que han sido utilizadas durante el proceso de identificación y reconocimiento de la figura:

$$\begin{cases} x = x_0 \cdot a \cdot \cos(\alpha) \\ y = y_0 \cdot b \cdot \sin(\alpha) \end{cases} \quad \text{en donde } \alpha \in (0, 2\pi] \quad (3.32)$$

3.11.2 – Circunferencia

A pesar de que la circunferencia es una figura ampliamente conocida, y tanto sus propiedades como ecuaciones básicas han sido divulgadas considerablemente a lo largo de los tiempos porque son objeto de estudio desde la educación primaria, no por ello se iba a dejar de reflejar el hecho de que tales propiedades y expresiones han sido utilizadas en la solución propuesta. No se debe olvidar, que la figura que constituye parte de la plataforma de aproximación, está formada por diversos círculos que deben ser identificados como acción previa al reconocimiento final de la plataforma. De hecho, la detección de los círculos (C1, C2, C3, y C4) tiene un peso del 50% respecto a la probabilidad total de reconocimiento de la plataforma.

La circunferencia y el círculo configuran no sólo unas características de equilibrio y regularidad modélica, sino también de máxima economía. Así, de todas las figuras planas equiláteras y equiángulas con igual perímetro, la circunferencia es la que encierra una superficie de mayor área. En este sentido, una circunferencia de centro en un punto fijo O , y radio r , representa el lugar geométrico de los puntos del plano cuya distancia a O es r , es decir, representa el lugar geométrico de los puntos de un plano que equidistan de otro punto fijo y coplanario llamado centro en una cantidad constante llamada radio. Este concepto, no debe confundirse con el del círculo, ya que un círculo de centro en un punto O y radio r , es el lugar geométrico de los puntos del plano cuya distancia a O es menor que r . Un círculo es por tanto, el subconjunto del plano limitado por una circunferencia.

1. Área: el área, al igual que sucede con la excentricidad, son descriptores que han sido empleados explícitamente durante el proceso de reconocimiento de la figura. Tal y como se puede deducir a partir de la explicación anterior, la circunferencia hace referencia únicamente al perímetro de una figura circular, lo que quiere decir que sólo posee longitud. Por consiguiente, esto quiere decir, que el área de la circunferencia está asociada íntimamente al concepto de círculo. El área del círculo delimitado por una circunferencia se determina a través de la siguiente expresión:

$$A = \pi \cdot r^2 \quad (3.33)$$

2. Excentricidad: en un contexto purista, el concepto de excentricidad no tiene cabida en las figuras circulares, o si la tiene es con valor cero, ya que una figura circular puede ser considerada como una elipse de excentricidad nula, o una elipse cuyos semiejes son iguales, o los focos coinciden. Sin embargo, dada la naturaleza intrínseca que circunscribe la problemática que rodea al proceso de adquisición de imágenes en pleno vuelo, en lo que a la cantidad de perspectivas posibles se refiere, la excentricidad del círculo juega un factor muy importante que se debe tener en cuenta durante el proceso de identificación y reconocimiento de los círculos.
3. Ecuaciones de la circunferencia: al igual que sucedió en el caso de la elipse, las ecuaciones de la circunferencia son expresiones claramente conocidas que no necesitan ser citadas explícitamente en este documento. No obstante, como consecuencia de haber formado parte de la solución, debe hacerse alusión explícita a la *ecuación paramétrica de la circunferencia*, que se parametriza en base a las funciones trigonométricas del seno y coseno.

$$\begin{cases} x = x_0 + r \cdot \cos(\alpha) \\ y = y_0 + r \cdot \sin(\alpha) \end{cases} \text{ en donde } \alpha \in (0, 2\pi] ; y (x_0, y_0) \text{ define el centro} \quad (3.34)$$

Esta ecuación juega un papel importante en la solución propuesta, ya que es la expresión empleada a la hora de dibujar el contorno de los cuatro círculos que se estudian para reconocer la figura, es decir, es la expresión que se emplea para dibujar el perímetro de C_1 , C_2 , C_3 , y C_4 . Sin embargo, debe tenerse en mente, que si el plano de la imagen se encuentra inclinado como consecuencia de tener girada la cámara respecto a la horizontal del suelo en el proceso de adquisición, la excentricidad aumentará y el círculo será visto como una elipse.

Por tal motivo, la ecuación paramétrica de la circunferencia se utiliza cuando la excentricidad de los círculos es inferior a un umbral determinado en la

sección de configuración del programa (0.4 por defecto). En caso que sea superior, se ha decidido utilizar la ecuación paramétrica de la elipse que se estudió con anterioridad. La variable que gestiona este umbral se denomina *Configuracion.UmbralDibujoCirculo*, y su valor por defecto es de 0.4.

3.11.4 – Distancia Euclídea

A modo introductorio se puede afirmar, que en el ámbito de la inteligencia artificial, y más concretamente en los algoritmos de clasificación (ampliamente usados en el área de visión por computador), existe un tipo de medida que se puede denominar como *medida de disimilitud, desemejanza o distancia*, dado que sirve para medir la distancia que hay entre dos objetos de forma que, cuanto mayor sea su valor, más diferentes son los objetos y menor probabilidad de que los métodos de clasificación los pongan en el mismo grupo (clase). Algunas de las distancias más utilizadas en este ámbito son: la distancia euclídea, la distancia euclídea al cuadrado, la distancia métrica de Chebychev, la distancia de Manhattan, la distancia de Minkowsk, etc.

Sin embargo, el caso que nos ocupa en este TFM, está relacionado con la distancia que existe entre dos puntos del espacio euclídeo representado normalmente a través de un plano cartesiano bidimensional, que en términos coloquiales se conoce como distancia ordinaria, y que se reduce a calcular el valor de la línea recta que une dichos puntos. En términos más formales, esta métrica se conoce como *distancia euclídea*. A pesar de que la distancia euclídea es la medida de distancia más usada en un espacio bidimensional, se debe poner de manifiesto, que resulta muy cara desde el tratamiento computacional de imágenes, ya que tiene un coste aritmético muy alto, sobre todo si se trabaja con números elevados.

Por consiguiente, se suelen emplear otras aproximaciones más sencillas desde un punto de vista computacional, como puede ser la distancia *City Block*, o la del *Tablero de Ajedrez*, ya que a pesar de que no responden al concepto real de distancia, proporcionan una aproximación significativamente buena con un coste computacional mucho menor que el de la distancia euclídea. En este sentido hay que reseñar, que el algoritmo ha sido diseñado para que se puedan emplear de una forma sencilla diferentes tipos de distancias. Para ello únicamente hay que codificar el tipo de distancia que se desee usar dentro de la función *CalcularDistancia*, utilizar un identificador de asociación, y asignarlo en la variable de configuración llamada *Configuracion.TipoDistancia*, que por defecto tiene el valor 0, que significa distancia euclídea.

Capítulo 4: Problemas y Decisiones Adoptadas

En este capítulo se describen los principales problemas a los que se ha tenido que hacer frente, así como la manera en que han sido abordados, y las principales decisiones adoptadas para resolverlos.

4.1 – Problemas de Umbralización / Binarización

A partir de las técnicas descritas en el capítulo anterior, es factible afirmar, que el objetivo fundamental que se persigue, es tratar de obtener una imagen binaria con una calidad lo suficientemente alta como para que sólo aparezca en ella el contorno de nuestra figura geométrica. Así, al hacer el etiquetado de regiones, sólo deberían aparecer las regiones propias en las que se descompone nuestra figura, de manera que a partir de sus propiedades se pueda discriminar si estamos o no ante nuestra plataforma de aproximación. Sin embargo, por desgracia, en algunos casos el proceso de umbralización no funciona adecuadamente, y esto provoca que no se genere una imagen binaria adecuada, lo que influye negativamente en el proceso de segmentación, y por ende en el reconocimiento de la plataforma.

A partir de las diversas pruebas efectuadas con el objetivo de refinar el algoritmo, se ha observado que uno de los principales problemas a los que se ha tenido que hacer frente, es el relativo a la elección del método de umbralización adecuado que sirva de base para decidir a partir de qué intensidad en la escala de grises se clasifican los dos píxeles entre las dos *clases* en que se divide una imagen binaria: la clase 0, que representa el negro, y la clase 1, que representa el blanco. Además, también es importante, que la imagen binaria resultante contenga el menor número de "impurezas" o áreas no deseadas, es decir, partes de la imagen original que no se corresponden con regiones que son propias de la figura geométrica que está dibujada sobre la plataforma de aterrizaje.

Por consiguiente, uno de los problemas más importantes y complejos a los que hubo que hacer frente, es el relacionado con la umbralización, ya que la conversión al modelo HSV desde el espacio de color RGB para obtener la imagen de intensidad resulta trivial con las funciones de alto nivel existentes en la *Image Processing Toolbox* de MATLAB. Sin embargo, dado que el proceso de adquisición de imágenes en un entorno de exteriores no está controlado, la determinación del umbral resulta bastante compleja cuando la figura geométrica ocupa áreas extremas (muy grande o demasiado pequeña), o cuando los niveles de grises se superponen lo que resulta una aproximación a una distribución unimodal.

La estrategia que se ha empleado de cara a la elección de un umbral adecuado, que sirva de base para obtener la imagen binaria que constituirá la materia prima para el proceso de segmentación, contempla los siguientes aspectos:

1. Implementar dos estrategias de umbralización bien diferenciadas que cubran la mayor parte de las situaciones a las que hay que hacer frente. Así, de esta manera, se ha utilizado tanto la umbralización directa, como la umbralización dinámica, esta última materializada a través de una implementación basada en el método de Otsu. Cada una de esas opciones puede ser utilizada enviando como segundo parámetro la opción: '*directa*' u '*otsu*', o a través de la configuración del programa (los valores son los mismos que los que se emplean a través del uso de parámetros).

2. Desarrollar un sistema automático de umbralización que esté basado en la recursividad de la función principal *PlataformaUAV.m*, y que emplee las dos estrategias que se basan en la umbralización directa y el método de Otsu si es necesario. El sistema se pone en marcha si se cumple alguna de las condiciones siguientes: si al utilizar la estrategia principal, que será la *directa* por tener mayor tasa de acierto en las diferentes pruebas realizadas, se produce un error o una excepción controlada durante el proceso de segmentación; ó se obtiene una probabilidad final que sea inferior a 0.6 (que se establece en *Configuracion.ProbabilidadMinimaAutoumbralizacion*).

A la hora de desarrollar este sistema autónomo, se valoró la posibilidad de incorporar una funcionalidad a la rutina *GestionAutoUmbral*, que se encargara de generar el histograma de la imagen, para posteriormente hacer un análisis de la distribución de la densidad de probabilidad del mismo, con el fin de decidir en base a esta información, cuál de las dos técnicas de umbralización se debería aplicar desde un inicio. Cabe reseñar, que esta técnica no tiene por qué ser excluyente a la estrategia basada en la umbralización automática comentada anteriormente (en el punto 2), sino que puede ser complementaria de ésta. Visto desde este punto de vista, la ventaja principal que aporta la nueva funcionalidad es que permite elegir de una manera dinámica y para cada situación concreta, cuál de las dos estrategias de umbralización tiene más probabilidad de éxito, sin tener que decidir de manera inicial, es decir, en tiempo de compilación (a través de la sección de configuración), cuál de las dos actuará como primaria, y cuál lo hará como secundaria, en el caso de que la primera falle.

Lo mismo se puede extrapolar en lo que se refiere a la elección del valor del umbral (almacenado en la variable *Configuracion.Umbral*) cuando se emplee la estrategia de umbralización directa. En este sentido, esta nueva funcionalidad basada en el análisis del histograma, aporta como ventaja una reducción en el número de casos en los que se tenga que poner en marcha el sistema automático de umbralización, ya que se supone que habrá una mayor tasa de acierto con la estrategia primaria, y con la elección del umbral en el caso de que se use la umbralización directa. Sin embargo, como contraprestación a esta ventaja, se tendrá que hacer frente al coste computacional que supone el tener que generar y analizar el histograma para cada imagen que vaya a ser procesada e identificada, lo que puede provocar que incluso la primera ventaja no sea lo suficientemente buena como para hacer frente al coste computacional de la desventaja. Por este motivo, y dado que la umbralización directa presenta una gran tasa de acierto en relación a las pruebas realizadas, se ha estimado que no resulta conveniente incluir esta nueva funcionalidad como parte de la solución.

Como justificación razonada hay que decir, que si se parte de la premisa, que el tiempo y el esfuerzo humano que requiere tanto la implementación del código como el análisis del estudio computacional, pueden no verse materializados como parte del trabajo final al no haber indicios sólidos que proporcionen ventajas, lo más sensato es concentrar los recursos en mejoras más factibles o líneas de actuación que sean visibles o apreciables en esta memoria final. De esta manera, se ha decidido aunar esfuerzos en dos áreas muy concretas. Por un lado, aumentar la eficiencia computacional con el fin de reducir todo lo que se pueda el tiempo de ejecución, algo fundamental ya que permitiría probar en tiempos razonables con distintas estrategias de umbralización. Por otro lado, aumentar la dedicación a todas las actividades necesarias para la realización de pruebas: generar imágenes, analizar condiciones heterogéneas, etcétera, ya que entre otros aspectos permiten corregir errores, aumentar la estabilidad del sistema, y escudriñar en nuevas líneas de mejora.

4.2 – Problemas Geométricos

Sin embargo, el problema relacionado con la umbralización no ha sido el único de los grandes inconvenientes a los que ha habido que hacer frente en este proyecto, ya que la otra gran vicisitud que hubo que abordar ha sido la relacionada con la geometría analítica. En realidad, el problema se reduce a determinar si cada una de las regiones que forman parte de la imagen binarizada, se corresponde con cada una de las figuras que constituyen la plataforma de aproximación. Para resolver este problema se tuvieron en cuenta las siguientes premisas:

- El eje fundamental de actuación de cara a la clasificación y reconocimiento de la figura que hay dibujada sobre la superficie de la plataforma de aproximación del UAV, ha consistido esencialmente en tratar de identificar si en las regiones etiquetadas que aparecen como consecuencia de haber llevado a cabo el proceso de segmentación, se encuentran las tres elipses y los cuatro círculos de fondo totalmente blancos que forman parte de la figura geométrica original, que finalmente fueron identificadas como: E1 (izquierda), E2 (central), E3 (derecha), C1 (grande centro), C2 (pequeño centro), C3 (arriba izquierda) y C4 (arriba derecha), tal y como se recoge en la figura 3-1. La identificación parcial y colectiva de cada uno de estos elementos sirve para otorgar una probabilidad de identificación final.
- En el caso de que se desea emplear el área como descriptor geométrico, y por consiguiente se pretenda determinar su valor para una figura concreta a través de técnicas de visión por computador, debe tenerse en cuenta que ésta depende en gran medida de dos factores: la resolución, y el porcentaje que representa la figura con respecto al total de la imagen. La resolución obviamente depende de la configuración de la cámara, y de si se realizan operaciones de escalado o no, mientras que el segundo factor, se encuentra totalmente vinculado al nivel de zoom y a la distancia a la que se encuentre el objeto de la lente del objetivo de la cámara. Este factor depende por tanto, de las características con las que se lleve a cabo el proceso de adquisición de las imágenes en pleno vuelo.
- A pesar de que el descriptor geométrico basado en la excentricidad ó ϵ (épsilon) resulta ser más estable ante los dos factores que inciden en el área, su principal problema se encuentra cuando se trabaja con perspectivas que tienen niveles de inclinación acusados. En una imagen aérea, el ángulo de inclinación viene determinado por las condiciones de adquisición, es decir, el ángulo que se forma entre el eje OY (perpendicular al eje de abscisas o eje OX que suele ser paralelo al suelo donde se encuentra posada la plataforma de aproximación), y la línea recta imaginaria que une el objetivo de la cámara y la figura. En este sentido, cabe reseñar, que el nivel de inclinación juega un papel tan importante, que un círculo visto desde lo alto con un nivel de inclinación elevado, se proyecta en la imagen como una elipse, y por tanto es identificado como tal. De hecho, durante la fase de pruebas del algoritmo, el círculo C1 que tiene una excentricidad muy cercana a 0 en la figura 3-1 o 6-1, cuando el grado de inclinación es elevado (IMG_A043.jpg) presentaba valores de excentricidad superiores a 0.6 (recordamos que el rango va de 0 a 1).
- Tomando como base la premisa anterior, tenemos que otro de los problemas es el relativo al control de la inclinación de la cámara con relación al suelo cuando se lleva a cabo el proceso de adquisición. Como ya se ha dicho anteriormente, una perspectiva inclinada en la imagen, provoca que los

círculos sean percibidos como elipses, y por ende que sean detectados y dibujados como tales. Con el fin de solucionar este problema, a la hora de dibujar el contorno de los círculos, se decidió utilizar un umbral basado en la excentricidad para determinar qué tipo de expresión matemática se tiene que emplear. De esta manera, cuando la excentricidad supera un umbral determinado (0.4), se utilizan las ecuaciones paramétricas de la elipse para dibujar el contorno del círculo ya que es visto como una elipse. Por contra, si es inferior a dicho umbral, se emplea la expresión propia del círculo basada en el radio, el coseno (para el eje OX), y el seno (para el eje OY).

- Por los motivos comentados anteriormente, resulta obvio afirmar, que no es factible utilizar constantes o valores fijos a la hora de identificar las diversas subfiguras de las que se compone la plataforma de aproximación (elipses y círculos eminentemente). Sin embargo, después de analizar los resultados obtenidos en la fase de pruebas, sí que se ha considerado oportuno -por temas relacionados con la efectividad y el rendimiento básicamente-, utilizar constantes para constituir rangos de filtrado en lo que a la excentricidad y al área se refiere. No obstante, debe tenerse en mente, que el rango que se constituya a partir de dichos valores, debe ser lo suficientemente amplio como para que el sistema se comporte de manera adecuada a pesar de que el nivel de resolución, o que el ratio figura / imagen (vinculado al zoom y a la distancia respecto al objetivo), sean manifiestamente dispares.
- La aproximación del vehículo autónomo a la plataforma puede consumarse desde cualquier dirección, lo que obliga a tener en cuenta este factor cuando se vaya a realizar el análisis geométrico. Por este motivo, y dado que se ha diseñado un algoritmo que emplea un sistema automático de orientación para realizar el reconocimiento utilizando siempre una misma orientación (la que aparece en la figura 3-1), resulta imprescindible que el sistema, una vez hayan sido etiquetado todas las regiones, sea capaz de identificar la elipse E2 (elipse central) con respecto a las demás. A partir de la orientación de dicha elipse se determina el nivel de rotación necesario para que la imagen siempre quede orientada tal y como se muestra en las figura 3-1 y 6-1. De esta manera, se facilitan las labores de reconocimiento, sobre todo en lo que al análisis de los ángulos se refiere.
- A la hora de llevar a cabo el etiquetado de componentes conexas, aunque se utilizan filtros en relación al área y la excentricidad, no estamos exentos de que surjan más de tres regiones en el proceso de extracción de elipses. Esta discriminación ha constituido, por tanto, otro de los principales problemas a los que se tuvo que hacer frente. Dado que el proceso de análisis se basa inicialmente en las tres elipses que figuran en la imagen 6-1, es necesario hacer descartes para quedarnos únicamente con tres de ellas en caso de que surjan más regiones candidatas, algo que suele ser lo habitual. Para ello, se decidió implementar una rutina dentro del algoritmo, que permite realizar comparaciones en subconjuntos de tres regiones, por lo que el número total de comparaciones vendrá dado por el número de formas en que se puede seleccionar un subconjunto de k elementos a partir de un conjunto dado con n elementos. Este número viene dado por la expresión matemática que permite determinar el número total de *combinaciones sin repetición*:

$$\frac{n!}{r \cdot (n-r)!} \quad (4.1)$$

Básicamente, se hacen comparaciones atendiendo al área, la excentricidad, la orientación, y las distancias euclídeas, de tal forma, que se elegirá aquel

subconjunto de 3 regiones que más se aproxime a la determinación geométrica de nuestra figura según los umbrales y estimaciones que se han asignado en la sección de configuración. Para ello, se ha implementado un algoritmo que calcula el sumatorio de las diferencias que hay en relación a las proporciones de las distancias entre las elipses, con las estimaciones asignadas en la sección de configuración, que finalmente se añaden al diferencial obtenido de restar los valores absolutos de las orientaciones de las elipses E1 y E3. Finalmente, toda esta información es almacenada en el campo *Ratio* del *array Distancia*, que es ordenado posteriormente mediante un método de ordenación (*quick sort* en nuestro caso) utilizando como criterio de orden la diferencia que se acaba de calcular. Su funcionamiento queda ilustrado con la siguiente figura:

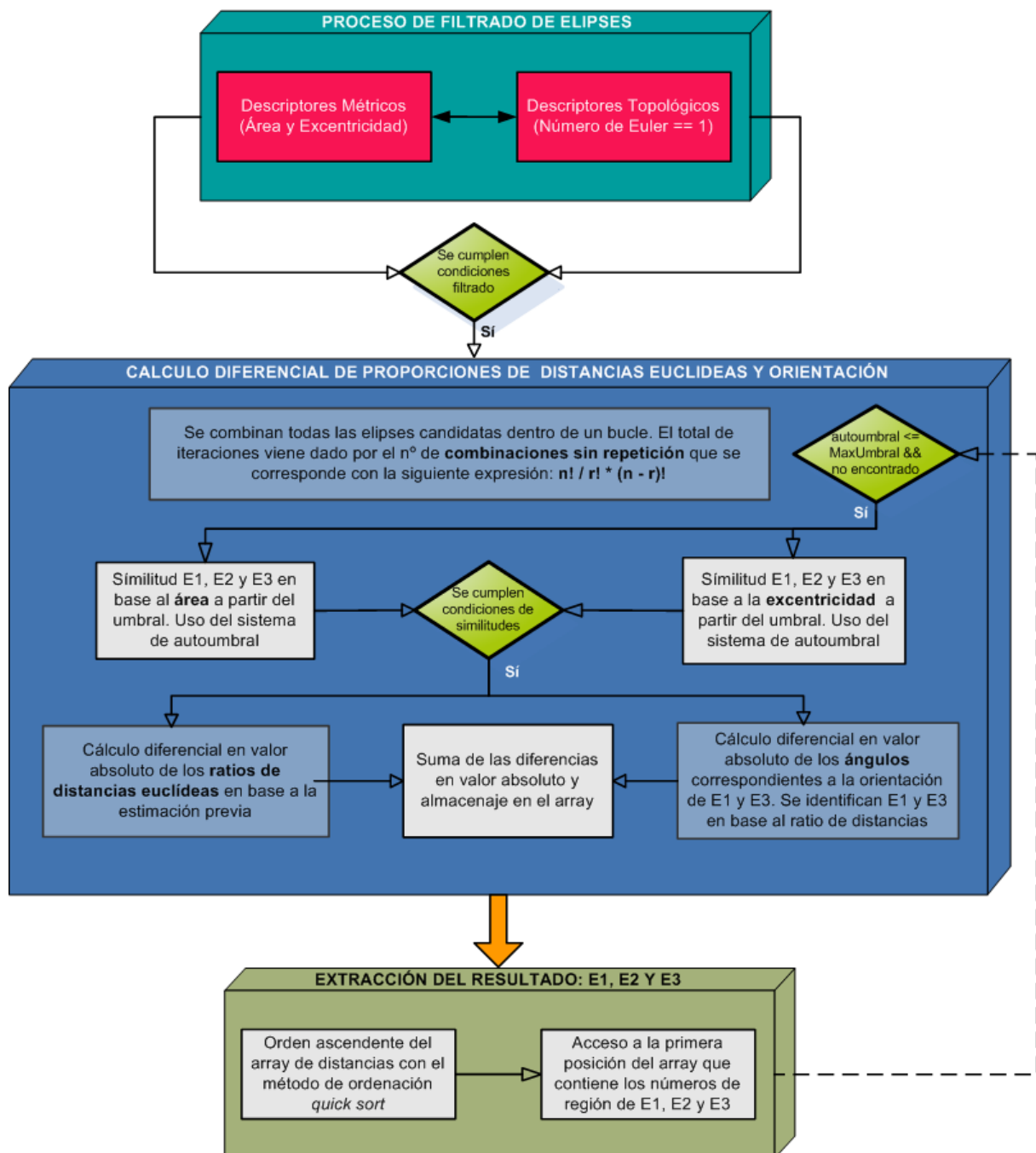


Figura 4-1 – Diagrama de flujo explicativo de la función *ObtenerElipses*

Tal y como se desprende del análisis de la figura 4-1, para que la región sea tomada en consideración, y por tanto se calcule la distancia, tiene que pasar el filtro de analogía (establecido a partir de sus umbrales correspondientes) en lo que a la excentricidad y área se refiere, ya que las elipses E1, E2, y E3 que se tienen que detectar, son prácticamente iguales en este aspecto. Una vez hayan sido procesadas todas las regiones que están almacenadas en el vector, finalmente se eliminan aquellas posiciones del vector cuyo índice sea superior a tres, ya que si se tiene en cuenta que el *array* ha sido ordenado previamente, las regiones deseadas (elipses) serán aquellas que cumplen con las propiedades iniciales basadas en el área, en el número de Euler y en la excentricidad; el resto se desecha.

En relación al algoritmo de identificación de elipses que se acaba de describir, cabe reseñar que se añadió una nueva funcionalidad a posteriori que permite *autoincrementar los umbrales* de forma autónoma en un 1% en el caso de que no se detecten las elipses necesarias por haber empleado un umbral de detección demasiado bajo. Esta funcionalidad fue añadida cuando se observó por medio de un juego de pruebas, que en algunas situaciones determinadas no se cumplía similitud alguna en relación al área y la excentricidad para todas las combinaciones del vector. De esta manera aumenta la probabilidad de encontrar regiones válidas, aunque existan ciertas desviaciones respecto a los umbrales previstos.

En lo que concierne a esta nueva funcionalidad que se acaba de comentar, hay que poner de manifiesto, que se decidió modificar la rutina de la función para que se autoincremente el umbral relativo al área y la excentricidad en un 1% hasta que se cumpla alguna similitud, aumentando así la probabilidad de identificación de elipses. No obstante, este autoincremento del umbral se encuentra totalmente gestionado por una variable de control que se define en la sección de configuración del programa, que permite que al sistema no entrar en un bucle infinito en el caso de que no exista analogía alguna. Esta variable se ha denominado *Configuracion.MaximoValorAutoUmbralElipses* y su valor por defecto se ha establecido a 70.

- Un problema análogo al que se acaba de señalar, surgió también cuando se trataban de identificar los cuatro círculos que son usados durante el proceso de reconocimiento de la plataforma. De este modo, al igual que sucedió con las elipses, fue necesario realizar una función que se encargara de analizar las propiedades de cada región, para decidir cuál de ellas tenía más analogía con el círculo en cuestión, estableciéndose así los vínculos región / círculo. Esto resulta fundamental, dado que en el análisis y en el reconocimiento se utilizan las características geométricas particulares de cada círculo. En este sentido, el problema fue resuelto de la siguiente manera:
 - a. Se recorren todas las regiones con el fin de elaborar una lista con las posibles regiones candidatas que superen los criterios de filtrado que han sido definidos en la sección de configuración del programa. Estos criterios atenderán esencialmente al área, la excentricidad, y el valor del número de Euler, que debe coincidir con alguno de los elementos del siguiente conjunto: $\{-2, -1, 0, 1\}$. En primer lugar se identifican los círculos C1 y C2 mediante el uso de la función *IdentificarC1_C2*. En este sentido se debe tener en cuenta, que en primera instancia hay que encontrar el círculo C1, ya que para la identificación de C2 se utiliza un ratio basado en el área cuyo denominador emplea el valor del área de C1.

- b. El primer objetivo de la función *IdentificarC1_C2* es encontrar C1. Para ello, se calcula la distancia euclídea de cada región con el punto central de la figura (hallado en relación a la mitad de la distancia euclídea que hay entre las elipses E1 y E3), así como la proporción que existe entre el área de la región y el área media de las tres elipses identificadas con anterioridad. Toda esta información se almacena en un array que es ordenado ascendentemente mediante el método de ordenación *quick sort*. Aquella región con menor distancia, y que su área dividida entre el área media de las tres elipses quede dentro del rango definido en la configuración (su valor se define en la figura 6-1), será considerada como el círculo C1. Acto seguido, se tratará de identificar el círculo C2 de manera análoga utilizando para ello la misma estructura de datos que fue ordenada con anterioridad. Sin embargo, ahora el ratio relativo al área se establece con el área del círculo C1, y no con la media de las elipses.
- c. Posteriormente se identifican los círculos C3 y C4 mediante el uso de la función *IdentificarC3_C4*. Para ello, se calcula la distancia existente entre el centro de cada región y el punto medio que equidista de las elipses E1 y E3, cuyo valor sirve de base para calcular el ratio con la distancia entre las elipses E1 y E3. Acto seguido se resta este valor, con el ratio que relaciona la distancia que hay entre E1-E3, y C3-C4 con el punto medio de las elipses E1 y E3 (el ratio *I4* y *I5* que aparece en la figura 6-1), obteniéndose la diferencia en términos de valor absoluto. Esta diferencia se almacena en el *array* siempre y cuando el ratio que se constituye a partir del área de C1, esté dentro de los umbrales que se hayan estimado en la sección de configuración. En caso contrario, se borra el elemento del *array*, porque éste va a ser ordenado finalmente de manera ascendente y así se reduce el tiempo de ejecución. Por consiguiente, una vez ordenado el *array*, quiere decir que las dos primeras posiciones del vector se corresponderán con las dos regiones que tengan una menor distancia, y que el ratio que forma su área con el del círculo C1 está dentro de los umbrales definidos en la configuración. De esta manera, dichas regiones serán consideradas como C3 y C4. Finalmente, para discriminar cuál de las dos regiones se corresponde con C3 y C4, se calcula el ángulo que forma cada una de ellas con el círculo C1 respecto al eje de abscisas OX, de manera que aquel que tenga un mayor valor será considerado C3, y el que lo tenga menor será considerado C4.
- d. Al igual que sucedió en el proceso de detección de elipses, mediante el uso de diferentes pruebas, se detectaron determinadas situaciones en las que los umbrales provocaban que ninguna región cumpliera con los rangos establecidos en las estructuras condicionales. Para solventar el problema, en vez de redimensionar los rangos, se decidió que era más conveniente reescribir el código de las rutinas para que en el caso de que ninguna región cumpla las condiciones, el umbral se autoincremente en un 1%. Si únicamente se redefine el umbral en la sección de configuración, el problema se puede volver a repetir con otra imagen que tenga unas condiciones excepcionales y quede fuera del rango establecido a partir del umbral. En este sentido hay que reseñar, que para controlar que el sistema no entre en bucle infinito, el autoincremento se lleva a cabo siempre que no se supere el valor indicado en la variable *Configuracion.MaximoValorAutoUmbralCirculos*. Dicha funcionalidad ha sido añadida tanto en el reconocimiento de C1 y C2 (función *IdentificarC1_C2*), como en el proceso de selección de

C3 y C4 (función *IdentificarC3_C4*). Este proceso queda plasmado de forma visual a través de la figura 4-2 que se muestra a continuación:

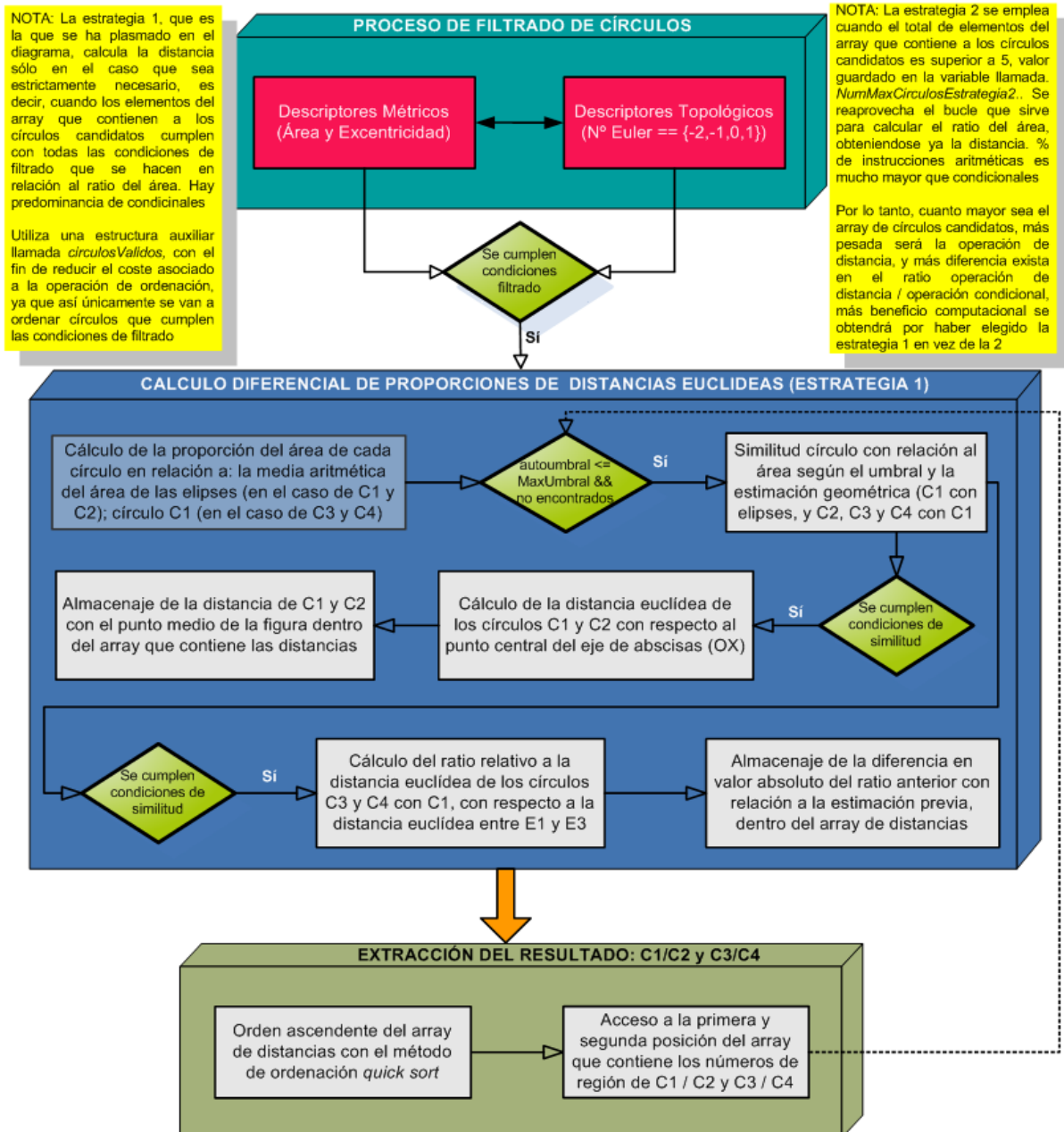


Figura 4-2 – Diagrama de flujo funciones *IdentificarC1_C2* *IdentificarC3_C4*

- Probablemente, otro de los problemas con mayor impacto sobre el resultado final y que más tiempo ha consumido, ha sido el relativo a la determinación de los umbrales en la sección de configuración del programa, ya que como se ha podido observar en el párrafo anterior, su valor es de vital importancia para establecer relaciones. En este sentido cabe reseñar, que el programa desarrollado íntegramente en MATLAB ha sido concebido para que pueda ser totalmente parametrizable desde la sección de configuración de la función principal. La asignación de umbrales y estimaciones, juega un papel esencial para definir los rangos, y por ende, para poder determinar qué valor es considerado como apto o inadecuado. Su asignación se cimentó en base a 3 líneas de actuación:

- a. A través del análisis geométrico que se pone de manifiesto a través de la figura 6-1, teniendo siempre en mente que este análisis está hecho desde una *perspectiva de planta* en la que no hay inclinación, es decir el ángulo de inclinación respecto al eje OX es de 0° . Sin embargo, se desea que el sistema de reconocimiento se comporte de forma adecuada cuando existan niveles de inclinación elevados, lo que requiere establecer umbrales que lo hagan flexible para poder dar soporte a nuevas perspectivas.
- b. A partir de un número elevado de pruebas que sirvieron de base para poder calibrar el sistema según los resultados que se iban obteniendo, es decir, utilizando una estrategia empírica basada en el paradigma ensayo-error. Para ello se realizaron exhaustivos procesos de adquisición de fotografías con el fin de obtener un gran número de imágenes heterogéneas en base a diferentes distancias, inclinaciones, resoluciones, etcétera, ya que el objetivo final del trabajo, es que la imagen sea identificada ante el mayor número de condiciones posible.
- c. Esta funcionalidad es una de las más importantes. Para llevar a cabo esta línea de actuación, se dotó al sistema de una serie de rutinas, que permiten exportar a ficheros CSV toda la información depurada del programa que aparece reflejada en las tablas de resultados y datos que aparecen en los Anexos II, y III respectivamente. El uso de esta información ha sido de vital importancia a la hora de depurar el sistema, y así tratar de "afinar" lo mejor posible la parametrización del software, esto es, definir valores de umbrales y estimaciones. Este aspecto se considera clave para conseguir tasas de aciertos elevadas cubriendo el mayor número de situaciones posibles. La asignación de umbrales se basó totalmente en esta línea de actuación, que tiene como objetivo aumentar la versatilidad del software.

4.3 – Problemas de Orientación

Con el fin de facilitar el proceso de identificación y análisis de las regiones (sobre todo en lo que a los ángulos se refiere), se optó en la etapa de diseño, por llevar a cabo una orientación de la imagen, de tal forma que el sistema sea capaz de identificar de forma autónoma, cuántos grados tiene que rotar para que el proceso de identificación se lleve a cabo siempre en una misma perspectiva, que es la que aparece reflejada en las figuras 3-1 y 6-1. Esto facilita el proceso de identificación, y permite tomar como referencia el eje de abscisas OX, sin necesidad de tener que hacer cálculos diferenciales sobre los ángulos, lo que aumenta la interdependencia entre los diversos elementos que constituyen la figura.

El proceso de orientación se fundamenta en el uso de la propiedad *orientation* que posee cada una de las regiones, una vez se consuma el proceso de segmentación y descripción de regiones. Esta propiedad que es retornada por la función *regionprops* que forma parte de la *Image Processing Toolbox* de MATLAB, devuelve el ángulo en grados en el intervalo $[-90, 90]$ que hay entre el eje de abscisas OX y el eje mayor de la elipse, que tiene los mismos segundos momentos que la región. El concepto se ilustra de forma gráfica mediante la figura 4-3 para una mejor comprensión por parte del lector. En dicha figura se pueden observar los siguientes elementos: ejes de la elipse dibujados con línea continua de color azul; focos de la elipse dibujados

por dos puntos rojos; y el eje de abscisas o eje OX (o una recta paralela en su defecto) señalada por una imagen discontinua de color azul.

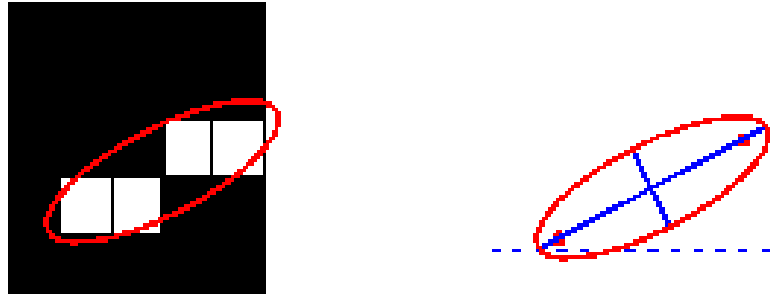


Figura 4-3 – Propiedad orientación de la región

A partir de la imagen se deduce que el ángulo que retorna la propiedad orientación de cada una de las regiones, será el que existe entre las líneas azules (la continua y discontinua). Sin embargo, la función retorna un valor escalar en el intervalo $[-90, 90]$ y no en el intervalo $[-180, 180]$ que es el que se necesita para poder orientar la imagen de forma completa y poder determinar a cuál de los cuatro cuadrantes pertenece el ángulo que existe en los 2π radianes o 360° que tiene la circunferencia completa. Para solucionar este problema se plantean dos posibles opciones. Por un lado, dado que se sabe la orientación, las coordenadas del centro, y la longitud del eje mayor y menor, se pueden utilizar las funciones trigonométricas coseno y seno para obtener las coordenadas X e Y respectivamente. A partir de las coordenadas, se puede calcular el vector director, y posteriormente hallar el ángulo que dicho vector forma con el vector director $\vec{v}(1, 0)$ que identifica al eje OX.

Otra posible opción, que ha sido por la que se ha optado desde un inicio, consiste en utilizar directamente la propiedad orientación de cada una de las regiones que devuelve la función *regionprops*. Sin embargo, tanto la opción anterior como ésta, requiere el uso (en determinados casos y circunstancias) de hasta dos rotaciones. La primera rotación, sirve para orientar la figura en el intervalo $[-90, 90]$, es decir, sirve para dejar la figura tal y como aparece en la imagen 3-1 ó viceversa (la figura 3-1 girada 180°), lo que nos asegura que quede orientada en una de esas dos posiciones. La segunda rotación, que es opcional, girará la imagen 0° ó 180° para certificar que la perspectiva siempre sea igual a la que aparece en la figura 6-1. En este sentido, es importante tener en cuenta, que hay que utilizar una estructura condicional que gestione la segunda rotación, y la lleve a cabo sólo en el caso de que sea estrictamente necesario. Inicialmente, se había obviado este control, por lo que siempre se llevaban a cabo dos rotaciones aunque la última fuera de 0° , lo que incurría siempre en un coste computacional extra que en la mayor parte de los casos no era necesario asumir. Este control es muy importante, ya que reduce considerablemente el tiempo de ejecución, dado que en la mayoría de las ocasiones esta segunda rotación no resulta necesaria.

4.4 – Problemas de Eficiencia Computacional

Otro de los problemas que hubo que afrontar, ha sido el relativo al rendimiento y a la *eficiencia computacional*, ya que si el software desarrollado llegara a implantarse en un entorno de producción, el tiempo de ejecución no puede demorarse en exceso, algo que sucedía hasta la fecha y que por tanto ha sido objeto de mejora. No se debe olvidar, que la batería de un UAV tiene una capacidad finita, y además se supone que cuando éste se aproxima a la base para aterrizar, significará en la

mayoría de las ocasiones que su carga puede estar próxima a agotarse. Además, un uso muy alto de CPU durante cantidades prolongadas de tiempo, contribuiría a que el tiempo de autonomía se redujera drásticamente por culpa de tener que abastecer las necesidades energéticas de la CPU. Cabe reseñar en este sentido, que este es uno de los típicos problemas con el que se encuentra un ingeniero cuando trabaja con máquinas que deben tomar decisiones en entornos de tiempo real, o en intervalos de tiempo muy reducidos. En este aspecto la línea de actuación realizada una vez el código se encontraba en un estado avanzado, consistió en hacer una revisión total del mismo analizando los tiempos de ejecución por módulos, para ver cuáles eran los procesos que proporcionaban un mayor margen de mejora.

Gracias a este profundo y exhaustivo proceso de revisión computacional, se detectó el inconveniente con la orientación señalada anteriormente, así como otra serie de anomalías computacionales que afectaban al rendimiento de varias de las funciones que sirven para identificar los círculos y las elipses. En general, lo que se hizo en este aspecto, ha consistido básicamente en analizar el código de una manera muy minuciosa con la finalidad de añadir una serie de estructuras condicionales tanto en bucles como en otras partes del código, con el fin de controlar, y por tanto disminuir, las situaciones en las que se debe calcular la distancia euclídea. Si se analiza con detalle la estrategia que se lleva a cabo en los procesos de selección de los círculos C1, C2, C3, y C4, se puede observar claramente cómo uno de los procesos básicos y fundamentales que más coste computacional requiere, es el relativo a la ordenación de los *arrays*, que con el método de ordenación *quick sort*, tiene un coste asintótico $O(n * \log n)$. Esta operación, que resulta inevitable, ha sido objeto de estudio ya que se usa tanto en la función *ObtenerElipses*, como en las funciones *IdentificarC1_C2* e *IdentificarC3_C4*, dado que éstas toman como región válida aquella que tenga el valor más pequeño en el campo distancia, cuyo cálculo es el que se ha tratado de reducir tanto como sea posible, sobre todo si es de tipo euclídea. No obstante, la distancia será tomada en consideración, siempre y cuando se cumpla con el filtro relativo al área estipulado mediante los umbrales y la estimación previa. Teniendo en cuenta las premisas que se acaban de postular, se tuvieron en consideración las siguientes propuestas o estrategias de identificación:

1. Estrategia de identificación 1: Esta opción tiene como objetivo básico, calcular la distancia sólo en el caso que sea estrictamente necesario, es decir, cuando los elementos del *array* que contienen a los círculos candidatos cumplen con todas las condiciones de filtrado que se hacen en relación al ratio del área. En el caso que cumpla con la condición de filtrado, el elemento es asignado a un *array* llamado *circulosValidos* que contendrá sólo aquellos elementos que cumplen con las condiciones estipuladas. Si no cumple, no se introduce en dicha estructura, y por tanto se descarta. Cabe reseñar que se utiliza una estructura auxiliar con el fin de reducir el coste asociado a la operación de ordenación, ya que de esta forma únicamente se ordenarán los círculos que cumplen las condiciones de filtrado. Tanto en la búsqueda de C2, como en la función *IdentificarC3_C4*, se reaprovechan los cálculos de distancia que hayan sido realizados previamente.
2. Estrategia de identificación 2: Esta estrategia reaprovecha el bucle que recorre todos los círculos candidatos para calcular el ratio del área, obteniendo ya de paso la distancia. Posteriormente, se ordena el *array*, y el primer elemento que cumpla con el filtrado en relación al ratio del área, ya se considera como el círculo buscado. Esto quiere decir, que el porcentaje de instrucciones aritméticas empleadas durante la ejecución de esta estrategia es mucho mayor que el de condicionales. Por lo tanto, a priori, esta opción resulta apta cuando el número de círculos del *array*

no es muy elevado, y las operaciones aritméticas relativas al cálculo de la distancia son pocas y sencillas, es decir, que no hay mucha diferencia en ciclos con una instrucción condicional. Por último hay que reseñar, que en esta estrategia, al igual que en la anterior, la función denominada *IdentificarC3_C4*, reutiliza los cálculos relativos a la distancia que se hayan realizado durante la ejecución de la función *IdentificarC1_C2*.

3. Estrategia de identificación 3: Esta estrategia es muy similar a la primera, ya que tiene como objetivo calcular la distancia sólo en el caso en que se cumpla con las condiciones de filtrado que se hacen en relación al ratio del área. Sin embargo, a diferencia de la primera opción, esta estrategia no se apoya en el uso de una estructura de datos auxiliar en la que se almacenan los elementos que hayan cumplido con las condiciones, sino que se procede directamente al borrado del elemento que no cumple con las condiciones de filtrado. Para ello se apoya en una función llamada *EliminarElementoArray*, que se encarga de suprimir el elemento y de redimensionar el *array* enviado como parámetro, que en este caso será la estructura de datos denominada *Círculos*, que contiene el conjunto de todos los círculos candidatos que cumplen con los descriptores basados en las propiedades métricas y topológicas previas.

Antes de mostrar los datos relativos al estudio de eficiencia computacional donde se analizan las dos primeras estrategias, ya se pone de manifiesto que la estrategia de identificación 3 es la que peor rendimiento ofrece con mucha diferencia. El número de operaciones condicionales y de movimiento de datos que se han de realizar para buscar cada elemento dentro del *array* y redimensionarlo, son tan elevadas, que la hacen completamente ineficiente. Cada vez que un elemento cumpla con las condiciones de filtrado, debe hacerse una búsqueda con coste asintótico lineal $O(n)$ y un redimensionamiento del *array*. Por este motivo, y con el fin de dotar de mayor claridad y síntesis a la memoria, los datos derivados de esta estrategia ya no se muestran en las tablas que contienen el estudio.

Si se emplea un ratio que permita establecer una relación proporcional entre el número de operaciones de distancia y el número de operaciones de tipo condicional, se observará que para la estrategia 1 el valor proporcionado por este indicador será notablemente menor que en la estrategia 2. Esto se traduce en que la estrategia 1 resulta idónea cuando: el cálculo de la distancia requiera de varias operaciones aritméticas que impliquen muchos ciclos de reloj, tal y como sucede en el caso de la distancia euclídea; o que la longitud del *array* de círculos candidatos sea de gran tamaño, puesto que permitirá ahorrar gran parte de las operaciones condicionales y de movimiento de datos que son necesarias para poder llevar a cabo la ordenación de la estructura de datos. De hecho, si se toma como área de estudio la rama de la *arquitectura de computadores*, se puede observar de forma clara, cómo por norma general, el número de ciclos que un procesador necesita para ejecutar la operación condicional resulta sensiblemente inferior al que se precisa en operaciones de tipo aritméticas que requieren del uso de la ALU (*Arithmetic Logic Unit*). En el caso de la distancia euclídea, las operaciones aritméticas son múltiples ya que: se tienen que restar dos números y elevar la diferencia, sumar estos resultados parciales, y finalmente calcular la raíz cuadrada de todo ello.

El número de ciclos concretos que requiere una operación determinada depende de varios factores, entre los que destacan: arquitectura del procesador, uso de salto retardado, uso de segmentación (*pipelining*), etcétera. Esta última se corresponde con una técnica que usa cortocircuitos internos que comunican las salidas con las entradas de las diferentes fases (FETCH1, FETCH2, ALU1, ALU2, etcétera) para poder enviar y recibir los resultados sin necesidad de tener que esperar un ciclo

entero (o más) hasta que el dato esté disponible. Esto se conoce como conflictos de datos y/o conflictos estructurales, también llamados R-STALL, W-STALL y S-STALL. Si se supone que el CPI (*Cycles Per Instruction*) medio de un procesador con una arquitectura x86 para una instrucción de tipo condicional / salto es de 1 ciclo, que la suma / sustracción son 3 ciclos, y que la multiplicación / división / raíz cuadrada requieren de 4 ciclos, quiere decir que el número estimado de ciclos que precisa una operación de cálculo de distancia euclídea ronda los 21 ciclos de reloj; su justificación es como sigue: 3 ciclos resta coordenada X, +3 ciclos resta coordenada Y + 4 ciclos cuadrado coordenada X + 4 ciclos cuadrado coordenada Y, + 3 ciclos suma coordenadas X e Y, + 4 ciclos raíz cuadrada del resultado anterior.

A mayores de las actuaciones que se acaban de comentar, hay que reseñar además la implementación de una nueva modificación que redundaba en una mejora que tiene que ver con la eficiencia computacional del algoritmo. Esta actuación, que se considera de gran relevancia por la drástica reducción en el número de ciclos, y por ende en el tiempo de ejecución, se encuentra íntegramente vinculada con los diversos métodos de ordenación de estructuras de datos. Inicialmente, se desarrolló una versión que empleaba una rutina de ordenación basada en el conocido método de la burbuja. Aunque funcionaba, resultaba poco eficiente ya que siempre iteraba n^2 veces siendo n la longitud del *array* que se tiene que ordenar, lo que suponía un coste asintótico $O(n^2)$ cuadrático. Posteriormente, al depurar el software para dar cabida a un nuevo proceso de refinamiento, se observó y se corrigió la deficiencia en la implementación con el control de los índices, lo que se tradujo en una reducción del número de iteraciones, pasando a un total de $(n * (n - 1) / 2)$, que ya es el que se corresponde realmente con el total de iteraciones que requiere un método puro de ordenación basado en el método de la burbuja.

Sin embargo, a pesar de que la reducción es considerable, el coste computacional o asintótico todavía sigue siendo prácticamente cuadrático, es decir, $O(n^2)$. Por tal motivo se sustituyó el algoritmo de ordenación por uno más eficiente. Como parte del estudio, se decidió que la actuación en este sentido, debía ir encaminada a sustituir la función de ordenación anterior por una implementación basada en el método de ordenación *quick sort*, que tiene un coste asintótico $O(n * \log n)$, lo que resulta ser casi lineal. Gracias a este cambio realizado en el método de ordenación, se consiguió hacer una reducción bastante significativa en el tiempo de ejecución, sobre todo cuando se genera un *array* con muchos círculos candidatos, tal y como suele suceder cuando se trabaja con resoluciones superiores a los 2 megapíxeles en lo que respecta al tamaño de imagen. El tamaño del vector de círculos candidatos depende esencialmente del entorno sobre el que se haya obtenido la imagen, y sobre todo de la calidad en el resultado del proceso de binarización.

Por consiguiente, las nuevas actuaciones realizadas han estado encaminadas a eliminar del *array* de círculos, todas aquellas regiones que no cumplen con las condiciones de filtro que se definen a partir del umbral y la estimación previa. De esta manera, se reducen notoriamente los tiempos de ordenación, que con el algoritmo empleado tiene un coste asintótico total de $O(n * \log n)$. Finalmente, y siguiendo esta línea de actuación, la función *IdentificarC1_C2* sigue una estructura híbrida en cuanto a la identificación se refiere, y aunque devuelve los mismos resultados, tiene un coste computacional diferente por la cantidad de operaciones distintas que utiliza en base a la naturaleza de las mismas.

Una vez aclarados estos aspectos, a continuación se adjuntan cuatro tablas en las que se muestran los datos de los resultados obtenidos en relación al estudio de eficiencia computacional de las funciones *IdentificarC1_C2* e *Identificar_C3_C4*. La tabla 4-1 compara las dos estrategias usando el método de ordenación basado en el algoritmo de la burbuja, mientras que la tabla 4-3 las compara empleando el

método de ordenación *quick sort*. La tabla 4-2, y la tabla 4-4, realizan una síntesis comparativa de aquellos indicadores y aspectos que se han considerado relevantes de cara al estudio propuesto. Además deben tenerse en consideración las siguientes indicaciones que se citan a continuación:

- Para la estimación del tiempo de ejecución, se realizaron 8 ejecuciones por cada imagen, y se obtuvo la media aritmética a partir de estos resultados parciales gracias al uso de una hoja de cálculo que sirvió de base para poder automatizar los cálculos.
- Los cálculos efectuados tienen en cuenta las operaciones que se realizan en el interior de la función *OrdenarArray* que es llamada por *IdentificarC1_C2* e *Identificar_C3_C4*.
- Los datos que aparecen en esta tabla son meramente estimativos, ya que el número de ciclos por operación depende de muchos factores: secuencialidad del código, uso de segmentación, arquitectura del procesador, etcétera. Para saber el número exacto debe usarse un software de carácter específico.
- El código se ha ejecutado en una versión MATLAB 7.12.0.365 R2011a que se encuentra instalada en una máquina virtual Microsoft Windows XP SP3 que se ejecuta con el hipervisor VMWare Player 6.0.1 build-1379776. En relación a este aspecto, hay que señalar, que la tecnología basada en virtualización ofrece un rendimiento computacional sensiblemente menor al convencional.
- El tipo de distancia que se ha usado como modelo es la distancia euclídea, que según la estimación realizada tiene un coste computacional de 21 ciclos de reloj. En la tabla 4-5 pueden observarse todos los supuestos que se han empleado en lo que al número de ciclos por operación se refiere.

ESTUDIO EFICIENCIA COMPUTACIONAL: FUNCIONES IDENTIFICARC1_C2 E IDENTIFICARC3_C4 (PlataformaUAV_v23.6)

Configuración		Totales		Operaciones					Ratio	Tiempo de Ejecución	Estrategia
Imagen	Escalado	Ciclos realizados	Círculos Candidatos	Condicionales	Distancia	Movimientos	Suma / Resta	Producto / División	Distancia / Condicional		
IMG_A063.JPG	2592 x 1944	1831	38	311	4	35	154	226	1.29	9.01	1
		9138	38	1464	38	929	1414	194	2.60	9.79	2
	1024 x 768	358	5	47	3	35	22	28	6.38	2.68	1
		426	5	45	5	38	28	29	11.11	2.70	2
IMG_A044.JPG	2592 x 1944	78685	246	8906	124	18448	7917	1632	1.39	17.01	1
		412094	246	60536	246	80315	60278	1232	0.41	25.14	2
	2000 x 1500	44030	129	5012	81	10114	4491	904	1.62	11.16	1
		114811	129	16667	129	21623	16523	655	0.77	12.87	2
IMG_A061.JPG	2592 x 1944	1117	20	172	6	43	87	118	3.49	9.15	1
		3252	20	420	20	416	388	104	4.76	9.12	2
	1024 x 768	510	7	65	5	42	32	40	7.69	2.70	1
		658	7	69	7	68	50	39	10.14	2.74	2
IMG_A045.JPG	2592 x 1944	31490	190	4541	83	4411	3776	1264	1.83	14.03	1
		226268	190	36128	190	37268	35922	962	0.53	18.76	2
	2534 x 1900	54332	185	6858	99	10988	6113	1270	1.44	13.89	1
		211880	185	34249	185	33932	34050	933	0.54	18.18	2
IMG_A064.JPG	2592 x 1944	459	6	60	4	43	31	34	6.67	7.27	1
		524	6	56	6	50	38	32	10.71	7.26	2
	1024 x 768	336	4	41	3	42	20	22	7.32	2.44	1
		344	4	36	4	38	20	22	11.11	2.42	2

IMG_A065.JPG	2592 x 1944	40771	923	7391	4	35	3694	5536	0.05	21.01	1
		4085412	923	851949	923	321281	851014	4619	0.11	150.57	2
	1024 x 768	2667	57	463	4	35	230	340	0.86	4.79	1
		17800	57	3269	57	1289	3200	289	1.74	5.23	2
IMG_A067.JPG	2592 x 1944	2725	48	429	8	128	232	294	1.86	10.62	1
		16636	48	2324	48	2768	2264	244	2.07	10.94	2
	1024 x 768	784	8	98	7	78	61	50	7.14	2.95	1
		816	8	84	8	98	64	44	9.52	2.94	2
IMG_A069.JPG	2592 x 1944	777	13	118	4	44	61	76	3.39	7.46	1
		1571	13	191	13	164	165	71	6.81	7.45	2
	1024 x 768	503	7	68	4	43	35	40	5.88	2.45	1
		653	7	71	7	59	51	41	9.86	2.48	2
IMG_A070.JPG	2592 x 1944	483	7	65	4	39	32	40	6.15	7.49	1
		670	7	69	7	74	50	39	10.14	7.55	2
	1024 x 768	401	5	49	4	42	24	28	8.16	2.56	1
		438	5	45	5	44	28	29	11.11	2.52	2

Tabla 4-1 – Estudio Computacional Identificación Círculos – Ordenación Burbuja

	Estrategia 1	Estrategia 2
Promedio Ratio Distancia / Condicional	4.03	5.78
Promedio Tiempo de Ejecución	8.26	16.59
Promedio Numero de Ciclos	14,569.94	283,521.72
Promedio Tiempo Numero de Ciclos > 5 círculos	9.40	19.40
Promedio Tiempo Numero de Ciclos ≤ 5 círculos	2.56	2.54
Promedio Tiempo de Ejecución Total	12.42	

Tabla 4-2 – Resumen Rendimiento Estrategias – v23.6 Burbuja

ESTUDIO EFICIENCIA COMPUTACIONAL: FUNCIONES IDENTIFICARC1_C2 E IDENTIFICARC3_C4 (PlataformaUAV_v24.5)

Configuración		Totales		Operaciones					Ratio	Tiempo de Ejecución	Estrategia
Imagen	Escalado	Ciclos realizados	Círculos Candidatos	Condicionales	Distancia	Movimientos	Suma / Resta	Producto / División	Distancia / Condicional		
IMG_A063.JPG	2592 x 1944	1905	38	355	4	47	156	226	1.13	9.11	1
		6477	38	917	38	862	754	194	4.14	8.75	2
	1024 x 768	420	5	58	4	47	24	28	6.90	2.68	1
		573	5	68	5	82	40	29	7.35	2.65	2
IMG_A044.JPG	2592 x 1944	27457	246	4603	124	2310	3034	1632	2.69	16.37	1
		54929	246	8535	246	6825	7550	1232	2.88	16.83	2
	2000 x 1500	17698	129	2900	82	1748	1988	904	2.83	10.69	1
		26537	129	4016	129	3370	3484	655	3.21	10.97	2
IMG_A061.JPG	2592 x 1944	1237	20	209	6	74	94	118	2.87	9.03	1
		2999	20	405	20	399	320	104	4.94	9.15	2
	1024 x 768	576	7	83	5	60	36	40	6.02	2.68	1
		906	7	114	7	129	77	39	6.14	2.66	2
IMG_A045.JPG	2592 x 1944	23746	190	3958	102	2206	2678	1300	2.58	13.33	1
		42615	190	6627	190	5285	5860	962	2.87	13.50	2
	2534 x 1900	24288	185	4087	99	2255	2844	1270	2.42	13.18	1
		41629	185	6460	185	5256	5680	933	2.86	13.36	2
IMG_A064.JPG	2592 x 1944	526	6	77	4	62	35	34	5.19	7.29	1
		726	6	89	6	109	55	32	6.74	8.33	2
	1024 x 768	420	4	56	4	60	24	22	7.14	2.50	1
		443	4	53	4	67	28	22	7.55	2.42	2

IMG_A065.JPG	2592 x 1944	41730	923	8320	4	47	3696	5536	0.05	20.81	1
		264800	923	43728	923	31578	40019	4619	2.11	24.51	2
	1024 x 768	2760	57	526	4	47	232	340	0.76	4.74	1
		10398	57	1510	57	1364	1269	289	3.77	4.88	2
IMG_A067.JPG	2592 x 1944	2898	48	516	8	150	246	294	1.55	10.42	1
		8403	48	1201	48	1115	996	244	4.00	10.44	2
	1024 x 768	1031	8	148	8	136	81	50	5.41	2.97	1
		1037	8	129	8	153	86	44	6.20	2.91	2
IMG_A069.JPG	2592 x 1944	946	13	153	5	87	70	76	3.27	7.44	1
		1819	13	236	13	252	174	71	5.51	7.46	2
	1024 x 768	610	7	92	4	74	42	40	4.35	2.45	1
		908	7	112	7	136	71	41	6.25	2.49	2
IMG_A070.JPG	2592 x 1944	555	7	83	4	60	36	40	4.82	7.51	1
		886	7	110	7	127	73	39	6.36	7.52	2
	1024 x 768	465	5	65	4	60	28	28	6.15	2.48	1
		622	5	76	5	95	45	29	6.58	2.47	2

Tabla 4-3 – Estudio Computacional Identificación Círculos – v24.5 Quick Sort

	Estrategia 1	Estrategia 2
Promedio Ratio Distancia / Condicional	3.67	4.97
Promedio Tiempo de Ejecución	8.09	8.41
Promedio Numero de Ciclos	8,292.67	25,928.17
Promedio Numero de Ciclos > 5 círculos	9.20	9.59
Promedio Numero de Ciclos ≤ 5 círculos	2.55	2.51
Promedio Tiempo de Ejecución Total	8.25	

Tabla 4-4 – Comparativa Rendimiento Estrategias – Ordenación Quick Sort

A partir de las tablas que recogen el estudio de la eficiencia computacional, se pueden postular las siguientes conclusiones que se detallan a continuación:

1. A la hora de realizar los cálculos en relación al número total de ciclos, se han realizado los siguientes supuestos en lo que concierne al número de ciclos que necesita cada tipo de operación. Dicha estimación queda sintetizada en la siguiente tabla que se muestra a continuación:

Número de ciclos estimado por operación condicional / comparativa	1
Número de ciclos estimado por operación distancia euclídea	21
Número de ciclos estimados por operación de movimiento de datos	2
Número de ciclos estimados por operación suma / resta	3
Número de ciclos estimados por operación multiplicación / división	4

Tabla 4-5 – Supuesto número de ciclos por tipo de operación

2. Tal y como ya se había previsto anteriormente, en líneas generales la opción más eficiente, tanto en relación al número de ciclos, como al tiempo de ejecución, es la que se corresponde con la estrategia 1 (8.09s frente a 8.41s en la v24.5, y 8.26s frente a 16.59s, en la v23.6). En este sentido, además también se ha demostrado formalmente, que la v24.5, que implementa la ordenación del array mediante una versión basada en el método *quick sort*, es más eficiente que la versión 23.6 que emplea un método de ordenación cimentado en el algoritmo de la burbuja (12.42s frente a 8.25s). Esta diferencia está vinculada con el tipo de estrategia y el número de elementos del *array* de círculos, por lo que se manifestará con más intensidad, cuanto más grande sea la estructura de datos, algo que ha quedado totalmente patente en la imagen IMG_A065.jpg si se usa la estrategia 2 con escalado de 2592 x 1944. Con la estrategia 1, las diferencias también existen, pero son notablemente inferiores.
3. El ratio *operación de distancia / operación condicional*, que se encarga de medir la relación existente entre los dos tipos de operaciones, resulta más elevado en términos generales con la estrategia 2 que con la 1, tal y como se había previsto. Además, también se ha observado que es más alto en la v24.5 que implementa la ordenación por *quick sort*, que en la v23.6 que se basa en el método de la burbuja. Este indicador es bastante relevante, ya que cuanto menor sea su valor, menos operaciones de distancia habrá que realizar, y por tanto más eficiente será el programa, sobre todo, si la operación de distancia es pesada desde un punto de vista aritmético y requiere de bastantes ciclos de reloj, tal y como sucede con la distancia euclídea. Por lo tanto, cuanto mayor sea el *array* de círculos candidatos, más pesada sea la operación de distancia, y más diferencia exista en el ratio *operación de distancia / operación condicional*, más beneficio computacional se obtendrá por haber elegido la estrategia 1 en vez de la 2.
4. A partir de la premisa anterior, se ha estimado que era conveniente llevar a cabo un análisis de los datos utilizando como indicador el tamaño del vector de círculos candidatos. Por consiguiente, a partir de la tabla resumen puede observarse, cómo para todas aquellas imágenes que después de haber sido segmentadas y generen un array de círculos candidatos con un valor inferior a 5 elementos (se define a través de una variable de configuración llamada *Configuracón.NumMaxCirculosEstrategia2*), proporciona mejor rendimiento el uso de la estrategia 2 que la 1, a pesar de que en términos generales (la

media aritmética de todos los registros) es más eficiente la 1. Aunque la diferencia no resulta ser palmaria (2.55s frente a 2.51s, lo que equivale a 4 centésimas de segundo), el ahorro se produce únicamente en tiempo de ejecución y no en ciclos (aunque la diferencia es muy pequeña, 0.39 ciclos), algo que probablemente tenga que ver con el uso que se hace internamente del bus E/S. Gracias a este estudio, se ha decidido que en la implementación final del algoritmo se utilice la estrategia 2 siempre que el tamaño del vector de círculos candidatos sea menor o igual a 5 elementos, ya que incluso existe ahorro computacional con la operación de distancia euclídea, la cual resulta ser bastante pesada desde un punto de vista aritmético.

5. Dada la aleatoriedad a la que está sujeto el posicionamiento dentro del *array* de círculos candidatos, no hay una relación fija y constante entre el número de operaciones condicionales y el número total de operaciones de distancia y viceversa. Si se toma como base el ratio *operación de distancia / operación condicional*, puede verse cómo hay imágenes que presentan una diferencia en dicho ratio de pocas décimas (por ejemplo las imágenes IMG_A044.jpg y la IMG_A061.jpg con escalado de 1024 x 768) mientras que otras pueden llegar hasta 3 puntos de diferencia, como resulta ser el caso de la imagen IMG_A063.jpg, donde hay una diferencia de 3,01 puntos entre la estrategia 1 y la 2 si se emplea un escalado de 2592 x 1944. La diferencia existente en relación a este ratio es importante, ya que es un indicador que sirve para decidir qué estrategia es la idónea si se toman como base los siguientes criterios: diferencia de ciclos existentes entre tipos de distancias, y longitud del *array* de círculos.
6. Dado que el ahorro en número de ciclos sigue una proporción constante para cada situación concreta, cuantos más elementos tenga el vector de círculos candidatos, más ciclos de reloj nos habremos ahorrado por haber elegido la estrategia de identificación adecuada desde el punto de vista de la eficiencia computacional. Por ejemplo, en este sentido el ahorro que se produce en la imagen IMG_A065.jpg con escalado 2592 x 1944 por haber empleado la estrategia 1, tanto en tiempo como en ciclos, es mucho más elevado que el que se obtiene en la imagen IMG_A064.jpg. Esto se debe a que el vector de círculos candidatos resulta mucho más elevado en la primera imagen que en la segunda, y por ende, el valor neto de operaciones totales sigue la misma tendencia. Para el primer caso la diferencia en segundos es 3.7s, mientras que para el segundo es 1.04s si se emplea con un escalado 2592 x 1944, y de 0.08s si se utiliza 1024 x 768.
7. En casi todas las imágenes, independientemente del tipo de escalado que se use, la estrategia 1 necesita un menor número de ciclos para su ejecución; de ahí que se haya marcado de color verde cada una de estas celdas donde el número total de ciclos es menor, y de color rojo (correspondientes a la estrategia 2) en donde el número total de ciclos es mayor. Sin embargo, por los motivos explicados anteriormente, cuando la imagen genera un vector de círculos candidatos de tamaño reducido, las diferencias son muy escasas. De hecho, en alguna situación arrojan valores prácticamente iguales, tal y como sucede en la imagen IMG_A065.jpg con escalado 1024 x 768 que se recoge en la tabla 4-1, donde la diferencia en ciclos únicamente es de 8. En casos como éste, cuando la diferencia es tan escasa, el tiempo de ejecución puede ser unas pocas decimas menor con la estrategia 2 que con la 1, cuando en teoría debiera ser al revés.
8. El objetivo fundamental de este estudio, es dar a conocer una estrategia o método formal que sirva de base para promover un estudio computacional

completo que sea trasladable a todas las funciones que forman parte del código fuente que se presenta a través del Anexo IV de este documento. Sin embargo, cabe reseñar, que el hecho de haber escogido estas dos funciones (*IdentificarC1_C2* e *Identificar_C3_C4*) se debe eminentemente a que, tras haber hecho un análisis de los tiempos de ejecución de cada módulo, se observó que éstas eran dos de las rutinas que más margen de mejora tenían en lo que a la eficiencia computacional se refiere.

Desde el punto de vista de la arquitectura de computadores, una modificación en el procesador que introdujera una instrucción específica para calcular la distancia euclídea podría reducir considerablemente el número de ciclos necesarios para cada instrucción. A pesar de que esto puede sonar utópico, algo similar se ha hecho con los procesadores digitales de señales (DSP), que tienen la arquitectura optimizada de tal forma que incluyen circuitería para reducir el número de ciclos necesarios en operaciones de multiplicar y acumular conocidas como MAC (*Multiply-Accumulate Operation*), ya que su uso es muy frecuente en las aplicaciones relacionadas con el procesado digital de señales. No obstante, en nuestro caso, resulta mucho más sencillo utilizar otro tipo de distancia que requiere menos esfuerzo computacional y que proporcione buenos resultados. El sistema está desarrollado de tal forma, que únicamente será necesario implementar la nueva distancia dentro de la función *CalcularDistancia*, utilizar un identificador o código de asociación, y emplearlo en la variable de configuración *Configuracion.TipoDistancia*, que por defecto tiene el valor 0, lo que significa distancia euclídea.

Finalmente, cabe reseñar, que otra de las actuaciones que se llevó a cabo en el ámbito de la eficiencia computacional, consistió en reducir el uso de las propiedades de las regiones a las que son estrictamente necesarias. Todas estas medidas, y las comentadas a través de este apartado 4.4 – *Problemas de Eficiencia Computacional*, permitieron reducir en más de un 60% el tiempo medio de ejecución. Este tiempo depende en gran parte del tipo de escalado, el método de umbralización, y el tipo de distancia que se use. No obstante, también depende en gran medida del número de círculos y elipses candidatos que se hayan generado tras el proceso de filtrado preliminar que toma como base al número de Euler, el área y la excentricidad.

Capítulo 5: Resultados y Pruebas

En este capítulo se describen los principales análisis, pruebas y resultados obtenidos una vez se han aplicado las diversas técnicas descritas en capítulo tres de esta memoria de TFM.

5.1 – Análisis Comparativos Efectuados

Antes de pasar a mostrar las pruebas y los resultados obtenidos, lo más coherente sería hacer una breve síntesis acerca de los principales análisis comparativos que se llevan a cabo sobre los elementos geométricos de la figura, para así poder discernir realmente a través de una base empírica, si la imagen que se está analizando pertenece realmente a la figura que hay sobre la plataforma de aproximación o no. Para ello, se ha decidido realizar una tabla en la que se recogen y sintetizan todos los análisis parciales comparativos (APC), que se efectúan sobre las imágenes de entrada con el fin de poder asignar una probabilidad final de reconocimiento.

Como todos los análisis tienen el mismo peso de cara a la evaluación final que dictamine el valor cuantitativo de la probabilidad, se ha decidido utilizar la media aritmética como expresión estadística fundamental para asignar probabilidades por cada acierto parcial. Una vez aclarado este aspecto, a continuación se muestra la tabla en la que se recoge la información más significativa asociada a cada análisis parcial comparativo (APC). No obstante, antes de mostrar la tabla, conviene aclarar lo siguiente:

- La columna descriptor hace referencia al descriptor geométrico empleado. La descripción de acrónimos es la siguiente: CNT: Centro ANG: Ángulo; DST: Distancia; ARE: Área; EXC: Excentricidad, POS: Posición. La columna valor hace mención a la probabilidad que otorga el análisis en caso de acierto.
- En todos los análisis se emplea un umbral de error que se obtiene tomando como base la estimación geométrica estipulada a través de la figura 6-1, y los valores de error asignados en la sección de configuración del sistema. Dicho valor es un porcentaje determinado respecto al valor del descriptor.

ID	Subfiguras	Descriptor	Valor	Descripción
APC1	E1, E2, E3	CNT	0.2	Análisis relativo a la alineación de las posiciones de los centros de la elipses
APC2	E1, E2, E3	ANG	0.2	Análisis relativo al ángulo que forman las rectas que unen centros de elipses
APC3	E1, E2, E3	DST	0.2	Análisis relativo a distancias euclídeas que hay entre los centros de elipses
APC4	E1, E2, E3	ARE	0.2	Análisis relativo a la similitud del área de las elipses en base a un umbral
APC5	E1, E2, E3	EXC	0.2	Análisis de similitud de excentricidad de las elipses en base a un umbral
APC6	C1, E1, E2, E3	ANG, DST, ARE	0.2	Análisis del ángulo, distancia y el área de C1 respecto a las elipses

APC7	C2, E1, E2, E3	ANG, DST, ARE	0.2	Análisis del ángulo, distancia y el área de C2 respecto a las elipses
APC8	C3, E1, E2, E3	ANG, DST, ARE	0.2	Análisis del ángulo, distancia y el área de C3 respecto a las elipses
APC9	C4, E1, E2, E3	ANG, DST, ARE	0.2	Análisis del ángulo, distancia y el área de C4 respecto a las elipses
APC10	C1, C2, C3, C4	DST, ANG	0.2	Análisis de los círculos en su conjunto para verificar la coherencia total

Tabla 5-1 – Listado de Análisis Parciales Comparativos

5.2 – Pruebas, Datos y Resultados

Los resultados que aquí se muestran, no sólo deben considerarse relevantes para llevar a cabo un estudio de la efectividad y fiabilidad del sistema, sino que son la base y la materia prima para determinar qué umbrales y estimaciones son las adecuadas para dar soporte al mayor número de perspectivas posible. Para ello, se han elaborado tres tablas distintas que se describen a continuación:

- **Tabla de Imágenes:** La tabla de imágenes (A.I), contiene toda la información relativa a cada imagen procesada, junto con las condiciones existentes durante su proceso de adquisición (luminosidad, distancia, inclinación, etc.). Esta tabla se ha elaborado con el fin de no redundar la información en las tablas de datos y resultados. Por lo tanto, esta tabla se encuentra relacionada con las otras dos tablas (resultados y datos) por medio del campo *Nombre*, el cual se encuentra presente en las tres tablas, ya que hace a su vez de clave primaria (no puede haber registros con el mismo nombre o nombre nulo) y clave foránea (sirve para relacionar las tablas), conceptos que surgen del álgebra relacional, que se trasladaron finalmente al modelo entidad-relación, que sustenta la teoría de las bases de datos relacionales convencionales.
- **Tabla de Datos:** La tabla de datos (Tabla A.II), contiene la información más relevante que se ha utilizado en la elaboración de las estimaciones que quedan reflejadas a través de la figura 6-1. Esta tabla arroja valores muy importantes, ya que permite observar de primera mano cuáles son los valores reales de los ratios que se emplean durante el proceso de reconocimiento de las diversas figuras geométricas que forman parte de la plataforma de aproximación. Se puede decir por tanto, que esta tabla es la fuente de datos principal que sirve de base para depurar y afinar el algoritmo, y por tanto decidir con fundamento los siguientes elementos. Por un lado, cuáles son los valores estimados, y por otro, cuál es el umbral que define el rango de máxima desviación posible respecto a tal estimación. Esta información resulta de gran relevancia para el proceso de identificación y reconocimiento de subfiguras que forman parte del dibujo que hay plasmado sobre la plataforma del UAV.
- **Tabla de Resultados:** La tabla de de resultados (A.III), contiene información relativa a los resultados que arrojan para cada imagen, los diversos análisis parciales comparativos que se describieron en la sección anterior. Además, incluye el tiempo de ejecución, y el resultado total de probabilidad de identificación, obtenido como resultado después de realizar el sumatorio de

todos los APCs que han resultado ciertos, para finalmente calcular su media aritmética. Por lo tanto, esta tabla muestra datos fundamentales para saber cuál es la tasa de acierto del algoritmo, y por ende, la eficiencia del mismo, puesto que indica el tiempo de ejecución necesario, expresado en segundos. Además, muestra un campo denominado *acierto*, que sirve para dictaminar tras el proceso de supervisión visual, si hay como mínimo tres regiones marcadas de color rojo sobre la imagen original, que coinciden con las figuras de la plataforma. Este campo resulta clave para poder evaluar el sistema de probabilidad final de la función gracias a la supervisión humana.

5.3 – Detalle de las imágenes procesadas

En esta sección se describen algunas de las imágenes más significativas que forman parte de la muestra del estudio que ha servido de base para evaluar la tasa de acierto y el rendimiento del sistema de identificación y reconocimiento. Dichas imágenes han sido escogidas escrupulosamente con el fin de señalar situaciones heterogéneas y divergentes, a las que el sistema es capaz de hacer frente. Por este motivo, se muestran imágenes variopintas desde el punto de vista del proceso de adquisición, de tal modo que se cubran las condiciones más extremas posibles en lo que a dicho proceso se refiere: imagen obtenida de cerca, de lejos, con poco ángulo de inclinación, con sombras, etcétera.

Para cada una de las imágenes escogidas, se muestra una serie de información que ha sido obtenida de forma exclusiva a partir de las funcionalidades de depuración y trazabilidad que implementa el algoritmo. Toda esta información, que se muestra acto seguido, queda sintetizada a través de una tabla que está asociada a cada una de las imágenes en la que se exhibe:

- Captura de pantalla de la imagen original de entrada.
- Captura de pantalla de regiones conexas de la imagen original sin orientar.
- Captura de pantalla de la imagen orientada con detección de elipses/círculos.
- Captura de pantalla de regiones conexas de la imagen orientada/reconocida.

Dado que todas las imágenes escogidas forman parte de la muestra empleada para testear el algoritmo, se ha estimado oportuno no hacer alusión a la probabilidad de acierto ni a las características del proceso de adquisición, puesto que se generaría información redundante ya que ésta está recogida en el anexo. No obstante, antes de mostrar las tablas con las imágenes, a continuación se muestra una tabla que sintetiza cuál ha sido la configuración empleada para obtener dichas imágenes:

		Escalado	
		1024 x 768	2592 x 1944
Umbralización	Directo	IMG_A066.jpg, IMG_A067.jpg IMG_A089.jpg,	IMG_A043.jpg, IMG_A044.jpg IMG_A045.jpg, IMG_A034.jpg IMG_A038.jpg
	Otsu		IMG_A065.jpg, IMG_A068.jpg

Tabla 5-2 – Configuración empleada en el detalle de imágenes procesadas



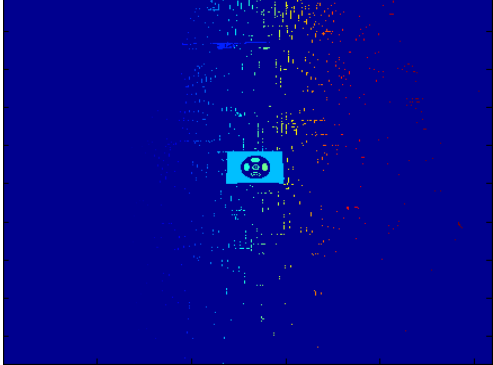
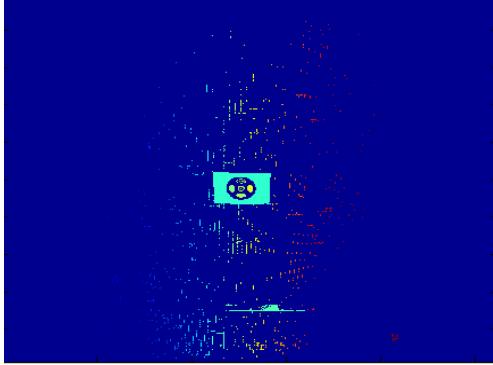
Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A043		
		

Tabla 5-3 – Detalle de Procesado IMG_A043.jpg



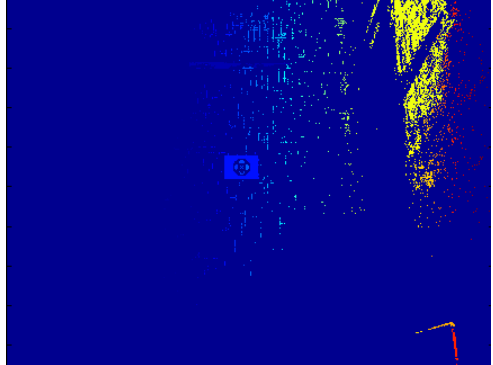
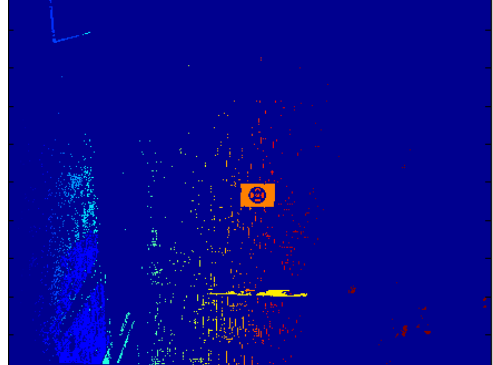
Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A044		
		

Tabla 5-4 – Detalle de Procesado IMG_A044.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A045		

Tabla 5-5 – Detalle de Procesado IMG_A045.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A065		

Tabla 5-6 – Detalle de Procesado IMG_A065.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A068		

Tabla 5-7 – Detalle de Procesado IMG_A068.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A066		

Tabla 5-8 – Detalle de Procesado IMG_A066.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A067		

Tabla 5-9 – Detalle de Procesado IMG_A067.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A069		

Tabla 5-10 – Detalle de Procesado IMG_A069.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A034		

Tabla 5-11 – Detalle de Procesado IMG_A034.jpg

Nombre	Imagen Original: RGB / Regiones Conexas	Imagen Orientada: Reconocida/ Regiones Conexas
IMG_A038		

Tabla 5-12 – Detalle de Procesado IMG_A038.jpg

5.4 – Comentarios de las imágenes identificadas

Todas las imágenes que se encuentran emparejadas en la misma página, tienen como objetivo revelar un determinado comportamiento del algoritmo. En relación a este aspecto se puede concluir lo siguiente:

1. Imágenes IMG A043.jpg e IMG A044.jpg: Estas imágenes han sido tomadas en un entorno de exteriores y a una altura considerable en relación al lugar donde se encuentra la plataforma de aterrizaje, tal y como denota el tamaño de la figura respecto al tamaño total de la imagen. El ángulo de inclinación respecto a la plataforma, sobre todo en la imagen IMG_A044 es considerable, aunque resulta un poco menor en la IMG_A043. Aunque la distancia en el eje OX es exactamente la misma en ambas tomas, en esta última la altura es sensiblemente mayor, lo que provoca que el ángulo disminuya. A través de estas dos imágenes, se trata de enfatizar la fiabilidad del algoritmo en relación a condiciones de altura estándar, puesto que el sistema es capaz de reconocer perfectamente la figura en dos imágenes que reproducen con exactitud las condiciones que pueden darse en una imagen que se haya obtenido durante el transcurso de un vuelo rutinario.
2. Imágenes IMG A045.jpg e IMG A065.jpg: Representan condiciones de adquisición contrapuestas en relación a la distancia. Ambas imágenes han sido obtenidas en un entorno de exterior con condiciones lumínicas similares, sin embargo, la distancia, que se denota en la perspectiva porque se obtiene un valor bajo en el ratio que representa el área que ocupa la figura respecto al área total, es manifiestamente divergente. La imagen IMG_A045, ha sido reconocida con éxito por el sistema a pesar de que dos de los círculos no han sido identificados adecuadamente, tal y como se puede apreciar si se hace zoom sobre la imagen. Esta imagen está bastante cerca de los límites máximos en los que el sistema puede trabajar si no se hace una ampliación de la misma. El objetivo de estas dos imágenes, es poner de manifiesto la versatilidad con la que trabaja el algoritmo desde el punto de vista de la distancia y la altura.
3. Imágenes IMG A068.jpg e IMG A066.jpg: Tienen como finalidad denotar de nuevo la versatilidad del algoritmo en cuanto a condiciones heterogéneas de trabajo se refiere, pero esta vez desde el punto de vista del ángulo de inclinación. En este sentido cabe subrayar, que el algoritmo ha sido capaz de trabajar con ángulos de inclinación muy elevados, siempre y cuando no se sobrepasen determinados límites en los que se provoca que la excentricidad de un círculo aumente de tal forma que se encuentre en valores propios de las elipses de la figura. La imagen IMG_A068 adolece de este problema, ya que ha sido obtenida con un grado de inclinación tan elevado, que genera una perspectiva en la que el sistema confunde las elipses con los círculos C3 y C4, y por consiguiente, no es capaz de reconocerla ni orientarla de manera adecuada. Sin embargo, la imagen IMG_A066, a pesar de tener un grado de inclinación palmario, el sistema identifica correctamente los círculos y las elipses tal y como se denota a través de las líneas rojas que genera el sistema. Cabe reseñar, que el método falla sobre esta imagen con escalado de 2592 x 1944.
4. Imágenes IMG A067.jpg e IMG A069.jpg: Estas dos imágenes tratan de representar la capacidad que tiene el sistema para trabajar ante diversos tipos de condiciones lumínicas con el fin de garantizar el mayor número de aciertos ante situaciones de luz heterogéneas que se puedan dar a lo largo

del día. La imagen IMG_A067 se obtuvo en un entorno de exteriores con una condición lumínica de sol, pero con una característica algo especial, debido a que existe una sombra que se cierne sobre la figura. Esta condición lumínica de sol y sombra, esta última provocada de manera intencionada, también se da sobre la imagen IMG_A066 que fue comentada con anterioridad. En lo que respecta a la imagen IMG_A069 hay que decir, que ha sido tomada en un entorno interior con el uso del flash, a fin de provocar reflejos sobre el papel que permita generar brillos aleatorios sobre el dibujo de la plataforma. Sin embargo, tanto las imágenes con condiciones lumínicas de sol / sombra (imágenes IMG_A066 y IMG_A067), como los reflejos provocados por el uso del flash en la imagen IMG_A069, no han sido impedimento alguno para que el sistema detectara y reconociera de una manera adecuada, la figura que hay dibujada sobre la superficie de la plataforma de aproximación.

5. Imágenes IMG_A034.jpg e IMG_A038.jpg: Finalmente, a través de estas dos imágenes, se trata de poner de manifiesto la versatilidad del sistema a la hora de reconocer la plataforma con independencia al grado de orientación que tenga ésta. El algoritmo, una vez haya reconocido las tres elipses (E1, E2, E3) es capaz de detectar y orientar correctamente la plataforma en gran variedad de situaciones para efectuar los análisis parciales comparativos (APCs) siguiendo los criterios de reconocimiento que aparecen en la figura 6-1 que forma parte de este documento. Tal y como se puede observar, las imágenes IMG_A034 e IMG_A038, se encuentran orientadas 135° y 225° respectivamente en el sentido de las agujas de reloj, si se toma como punto de partida la orientación de la figura 3-1 que tendría una orientación de 0° .

Ambas imágenes han sido identificadas de manera correcta, ya que con la configuración empleada, la cual ha sido indicada por medio de la tabla 5-2, se ha obtenido en el primer caso un 100%, y en el segundo, un 80% de probabilidad final de acierto. En este sentido hay que reseñar, que la orientación de la figura se realiza de manera correcta con independencia de la altura a la que se halle la plataforma, siempre y cuando se identifiquen de forma correcta las elipses E1, E2 y E3. Esto ha quedado de manifiesto a través de las imágenes IMG_A066 e IMG_A067, las cuales están a una distancia notablemente inferior (con relación a las imágenes IMG_A034 e IMG_A038), están orientadas 45° en el sentido de las agujas del reloj, y han sido identificadas de manera correcta.

En todas las imágenes, cuando se habla de distancia, se refiere a *la longitud del segmento imaginario que va desde el objetivo de la cámara hasta el centro de la figura que forma parte de la plataforma de aproximación*. Y cuando se habla de ángulo de inclinación se refiere: *al ángulo en grados que forma la línea recta imaginaria que une el objetivo de la cámara con el centro de la figura, con respecto al eje de ordenadas o eje OY, que es perpendicular al eje de abscisas o eje OX, que suele ser paralelo al lugar en donde está situada la plataforma de aterrizaje*.

Con el fin de explicar mejor el concepto que acaba de ser expuesto, a continuación se adjunta una imagen de elaboración propia que pone de manifiesto cuál ha sido la técnica empleada a la hora de hallar, tanto la distancia, como el ángulo de inclinación del objetivo de la cámara respecto al eje OY, para cada una de las imágenes que han sido utilizadas en la fase de pruebas y que quedan detalladas a través de la tabla A.I que se encuentra dentro del Anexo I. Tal y como se puede observar en la figura 5-1, a partir de la altura y la longitud sobre el eje OX (que es paralelo al suelo), por trigonometría resulta factible calcular de forma trivial tanto la distancia como el ángulo de inclinación que tiene la imagen respecto al eje OY que es vertical al suelo. Para corroborar que dicho ángulo era correcto, se instaló una

aplicación para Android denominada *Inclinometer Free v2.6*, dentro del *smartphone* Samsung Galaxy Ace. Esta aplicación, que permite averiguar el grado de inclinación del terminal, se utilizó conjuntamente con la cámara digital durante el proceso de adquisición de fotografías. Su uso dictaminó que los cálculos eran correctos.

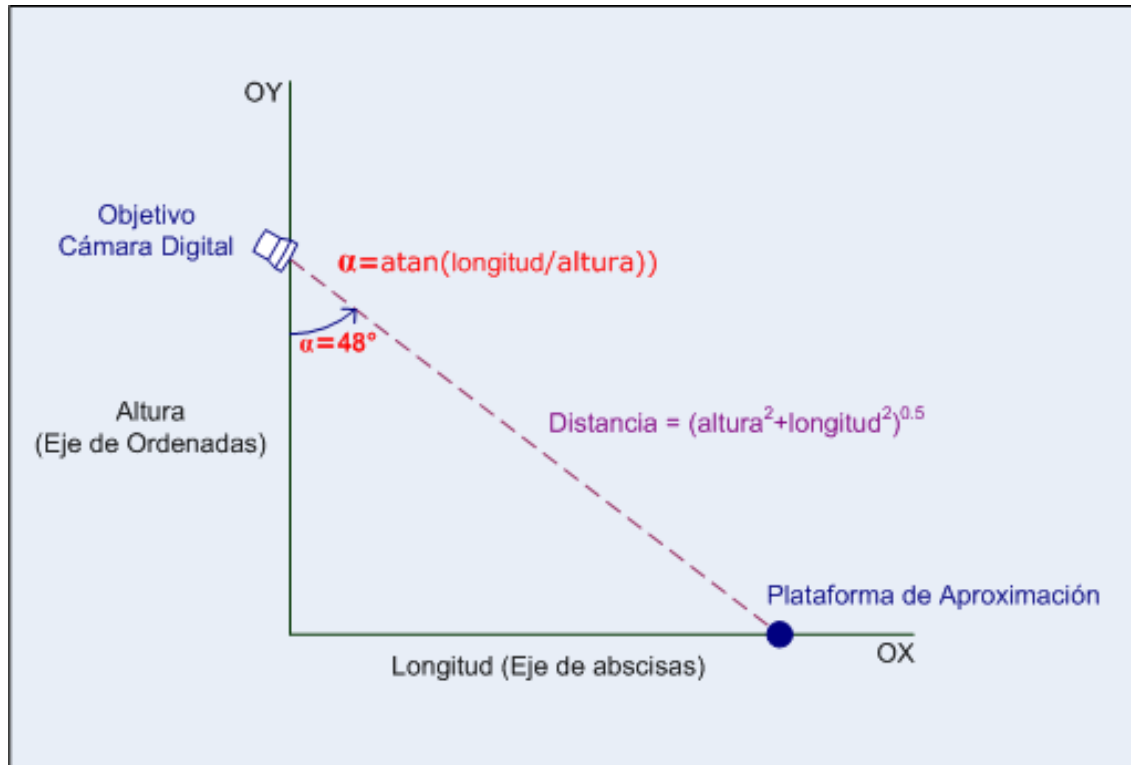


Figura 5-1 – Cálculo propiedades imágenes durante adquisición

5.5 – Síntesis y Resumen de Resultados

En la elaboración de las tablas que se muestran a continuación, se ha usado como fuente única de información los resultados generados como consecuencia de todas las pruebas realizadas durante la fase de *testing*, que han sido recogidos por medio de las tablas A.I A.II y A.III, que aparecen en los Anexos I, II y III. Con el fin de valorar con precisión los vínculos que pueden existir en relación a los diferentes escenarios que se pueden dar durante el proceso de adquisición, se ha decidido generar las tablas 5-13 y 5-14, que tienen en cuenta cierto tipo de condicionantes múltiples, y que han sido generadas a partir de la tabla 5-15 en la que se recoge de manera extendida los principales cruces de datos que se desprenden a partir de los datos de las tablas A.I y A.II que forman parte del anexo I y II respectivamente.

Cabe reseñar, que para poder elaborar esta síntesis de resultados en la que se ha realizado un análisis con alto grado de precisión y utilizando diferentes perspectivas desde el punto de vista de la información, se ha tenido que emplear una base de datos de tipo relacional para explotar los datos. Para ello, se ha optado por emplear Microsoft Access 2007, cuyas tablas se nutren a partir de vínculos con hojas de cálculo que recogen todos los datos en formato CSV que han sido generados por la función *PlataformaUAV.m* desarrollada en MATLAB. A partir de esta información, se han implementado en lenguaje SQL diversas consultas de referencias cruzadas (conocidas técnicamente como *crossstab queries*), que permiten construir matrices que relacionan filas con columnas, para tratar de observar posibles causalidades entre las condiciones del proceso de adquisición y los resultados obtenidos.

En primer lugar, se muestra la tabla 5-13 en la que se vincula información a partir de la distancia que hay entre el objetivo de la cámara y el centro de la figura que forma parte de la plataforma, y el ángulo de inclinación que forma la línea recta que une los dos puntos que se acaban de indicar, respecto al eje OY. Para calcular el valor que hay en cada una de sus celdas, es decir, la efectividad, se ha obtenido el porcentaje derivado del ratio *imagen acertada / imágenes totales* (para el campo cámara) y *prueba efectuada / pruebas totales* (para el resto de campos). En ambos casos, se considera que existe un *acierto*, cuando la supervisión visual del usuario dictamina que como mínimo hay 3 regiones marcadas de color rojo sobre la imagen original, y que éstas coinciden con algunas de las figuras de nuestra plataforma. Sin embargo, para el caso de las celdas asociadas al campo cámara, se considera un acierto cuando exista al menos un test (de las cuatro que se hacen por imagen) que sea considerado un acierto según los criterios que se acaban de definir.

		Distancia (metros)					Inclinación (grados)				
		0 - 2	2 - 4	4 - 6	6 - 8	> 8	0 - 10	10 - 20	20 - 30	30 - 40	40 >
Cámara	Powershot	100%	83.3%	66.7%	58.8%	85.7%	100%	100%	85.0%	60.0%	46.7%
	Galaxy Ace	91.7%	66.7%	41.7%	41.2%	66.7%	92.9%	81.3%	55.0%	46.7%	33.3%
Umbral	Otsu	81.3%	59.7%	25.0%	26.5%	36.9%	57.1%	64.1%	40.0%	38.3%	25.0%
	Directo	33.3%	19.4%	6.3%	2.9%	4.8%	14.3%	25.0%	7.5%	3.3%	11.7%
Escalado	1024x768	64.6%	40.3%	0.0%	1.5%	1.2%	19.6%	34.4%	11.3%	13.3%	20.0%
	2560x1920	50.0%	38.9%	31.3%	27.9%	40.5%	51.8%	54.7%	36.3%	28.3%	16.7%
TOTAL		70.1%	51.4%	28.5%	26.5%	39.3%	56.0%	59.9%	39.2%	31.7%	25.6%

Tabla 5-13 – Matriz de síntesis de resultados. Distancia e Inclinación

La tabla 5-14, es muy similar a la anterior, pero en vez de analizar los datos por el ángulo de inclinación y la distancia, lo hace por el tipo de condiciones lumínicas que existían en el momento en el que se captó la imagen, haciendo una clasificación también por el tipo de entorno, es decir, si es de exteriores o de interiores. El valor que hay dentro de cada una de sus celdas se obtiene igual que en la tabla 5-13:

		Exterior			Interior	Tiempo
		Soleado	Nublado	Sol y Sombra		
Cámara	Powershot	66.7%	84.4%	66.7%	100.0%	8.44 s
	Galaxy Ace	37.5%	65.6%	66.7%	91.7%	8.49 s
Umbral	Otsu	26.0%	46.9%	35.4%	85.4%	8.43 s
	Directo	13.5%	12.5%	16.7%	4.2%	8.23 s
Escalado	1024x768	8.3%	20.3%	8.3%	50.0%	8.19 s
	2560x1920	31.3%	39.1%	43.8%	39.6%	9.78 s
TOTAL		30.6%	44.8%	39.6%	61.8%	8.59 s

Tabla 5-14 – Matriz de síntesis de resultados. Luminosidad

A continuación se muestra la tabla 5-15, que contiene gran parte de los datos que han sido utilizados para elaborar las tablas 5-13 y 5-14. No obstante, debe tenerse en cuenta, que los datos relativos a la fila *cámara* de las dos tablas anteriores, no se pueden obtener del sumatorio por *cámara* de la tabla 5-15, ya que es muy probable que existan imágenes que tengan más de un acierto como consecuencia de que tengan más de un test correcto (existen cuatro por imagen). Sus celdas, al igual que en las tablas 5-13 y 5-14, registran la efectividad derivada del ratio *prueba acertada / pruebas totales*:

Cámara	Binarización	Escalado	Distancia (metros)					Inclinación (grados)					Time (seg)	Exterior			Interior
			0-2	2-4	4-6	6-8	> 8	0-10	10-20	20-30	30-40	40 >		Sol	Nublado	Sol y Sombra	Luz artificial con flash
Canon PowerShot A590 IS	Directo	1024x768	83.3%	61.1%	0.0%	5.9%	4.8%	35.7%	43.8%	20%	26.7%	20.0%	3.22	8.3%	25.0%	8.3%	100.0%
		2592x1944	66.7%	66.7%	66.7%	58.8%	85.7%	85.7%	100%	85%	53.3%	20.0%	13.40	66.7%	75.0%	66.7%	66.7%
	Otsu	1024x768	58.3%	27.8%	0.0%	0.0%	0.0%	7.1%	37.5%	10.0%	0.0%	20.0%	4.21	8.3%	21.9%	8.3%	16.7%
		2592x1944	8.3%	16.7%	8.3%	5.9%	0.0%	7.1%	18.8%	5.0%	6.7%	0.0%	12.93	12.5%	3.1%	16.7%	0.0%
Samsung Galaxy ACE	Directo	1024x768	83.3%	55.6%	0.0%	0.0%	0.0%	28.6%	37.5%	10.0%	26.7%	26.7%	3.42	8.3%	21.9%	8.3%	83.3%
		2592x1944	91.7%	55.6%	33.3%	41.2%	57.1%	78.6%	75.0%	45.0%	46.7%	33.3%	13.15	20.8%	65.6%	58.3%	91.7%
	Otsu	1024x768	33.3%	16.7%	0.0%	0.0%	0.0%	7.1%	18.8%	5.0%	0.0%	13.3%	3.41	8.3%	12.5%	8.3%	0.0%
		2592x1944	33.3%	16.7%	16.7%	5.9%	19.0%	35.7%	25.0%	10.0%	6.7%	13.3%	12.79	25.0%	12.5%	33.3%	0.0%

Tabla 5-15 – Matriz extendida de resultados finales

5.6 – Gráficas de Efectividad Por Imagen

En este subapartado, se muestran una serie de gráficas que permiten analizar de una forma visual la efectividad del sistema por cada imagen procesada, teniendo en cuenta que una imagen se considerará como reconocida, si el campo acierto de alguno de los cuatro tests que se llevan a cabo para cada imagen (1024x768 *directo*, 2592x1944 *directo*, 1024x768 *otsu* y 2592x1944 *otsu*) tiene como valor *true* (1). Los datos empleados para generar la gráfica, son los relativos a la fila *cámara* que aparecen en el interior de las tablas 5-14 y 5-15.

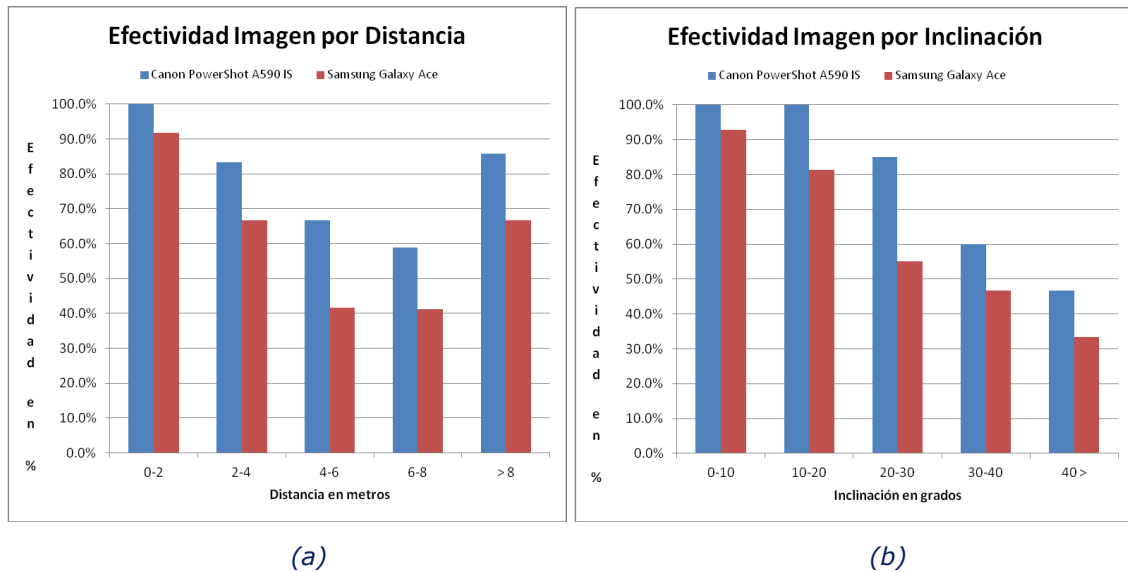


Figura 5-2 – Gráfica (a) Imagen Distancia; (b) Imagen Inclinación

Finalmente, se muestran dos gráficas a mayores, que sirven para analizar cuál es la efectividad por imagen según las condiciones lumínicas con las que se ha llevado a cabo la adquisición de la imagen, y determinar el tiempo medio de ejecución para su identificación y reconocimiento. En el cálculo del tiempo, que está expresado en segundos, sólo se han tenido en cuenta aquellas imágenes en las que como mínimo uno de sus cuatro tests sea considerado un acierto tras el proceso de supervisión.

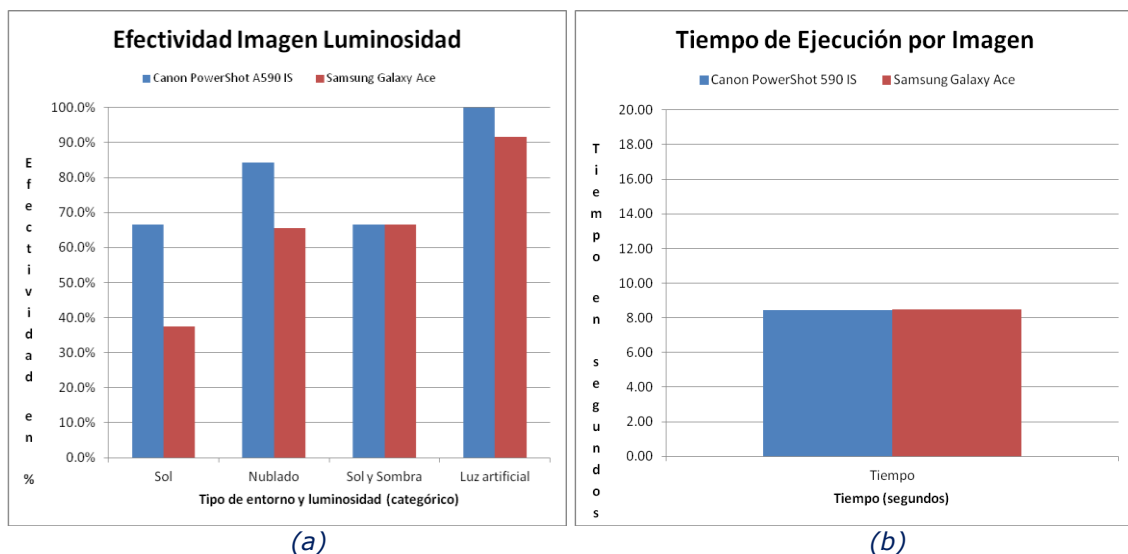


Figura 5-3 – Gráfica (a) Imagen Luminosidad; (b) Tiempo de Ejecución

5.7 – Gráficas de Efectividad Por Test

Seguidamente, para completar el análisis visual, se muestran las gráficas por test, que permiten analizar la efectividad del sistema para cada tipo de configuración empleada. Los datos utilizados para la implementación de dichas gráficas, han sido obtenidos de forma exclusiva a partir de la tabla 5-15. Al igual que en el apartado anterior, los gráficos empleados se basan en *columnas agrupadas* que sirven para comparar de forma rápida distintos elementos de una manera sencilla. Inicialmente, se muestran dos gráficas que analizan la efectividad por distancia e inclinación:

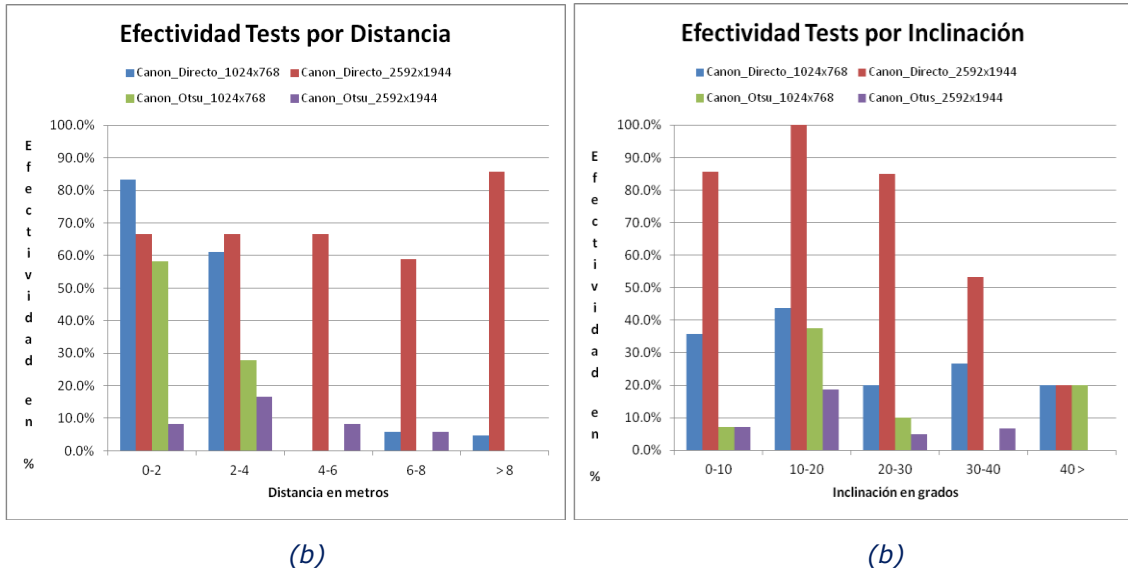


Figura 5-4 – Gráfica (a) Test Distancia; (b) Test Inclinación

A continuación, y siguiendo el hilo del apartado anterior, se muestran dos gráficas que permiten analizar la efectividad por test. Al igual que antes, el test se considera correcto, cuando el proceso de supervisión humana dicte que existe un acierto siguiendo los criterios previos. Para concluir, se muestra una gráfica con el tiempo de ejecución expresado en segundos para cada tipo de test realizado.

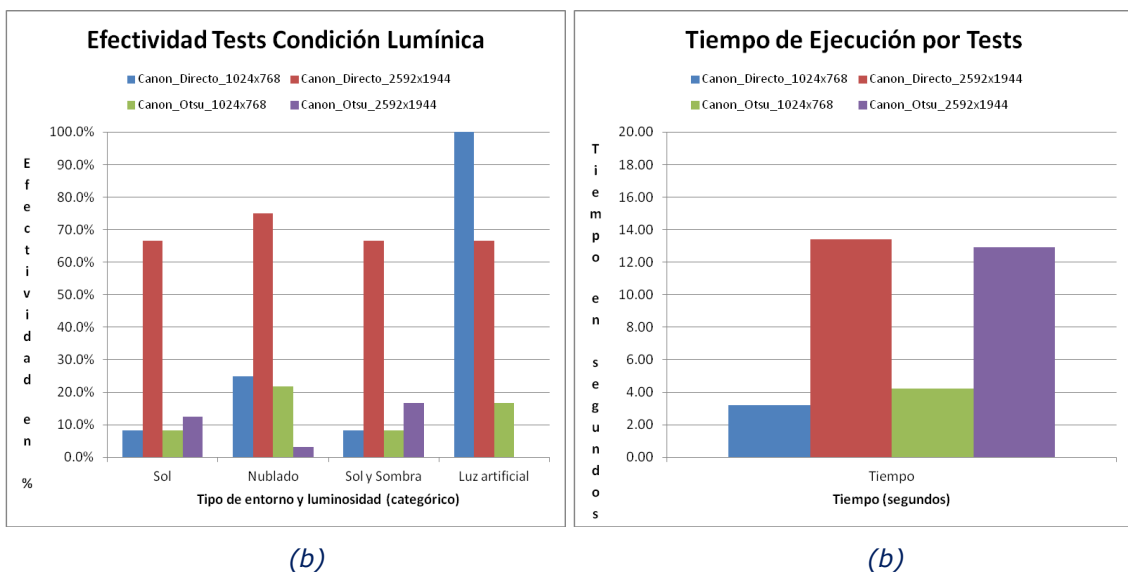


Figura 5-5 – Gráfica (a) Test Luminosidad; (b) Tiempo de Ejecución por Test

5.8 – Evaluación del Sistema de Probabilidad Final

Un aspecto muy importante que se debe tener en cuenta, es el relativo al estudio del resultado de la función, es decir, el valor numérico que retorna como salida la función *PlataformaUAV.m*, que nos indica la probabilidad final estimada de que la imagen contenga el dibujo de nuestra plataforma. Pues bien, a la hora de evaluar este sistema de probabilidad, aspecto fundamental para ver que fiabilidad tiene el sistema, y por lo tanto decidir qué nivel de credibilidad se le puede otorgar, ha sido necesario llevar a cabo un tipo de inspección visual (realizado por un humano), que se conoce formalmente como *supervisión* en el ámbito de la inteligencia artificial.

Este proceso de supervisión, consiste en hacer una inspección visual del resultado, que dictamine si el sistema de reconocimiento ha detectado como mínimo tres regiones marcadas de color rojo, y que éstas coincidan con alguna de las siete figuras {E1, E2, E3, C1, C2, C3, C4} que forman parte de la plataforma de aproximación. A pesar de que se puede emplear una gran variedad de criterios para decidir lo que es realmente un *acierto*, éste permite evaluar de forma objetiva, si el sistema de probabilidad tiene un comportamiento adecuado, es decir, comprobar que para probabilidades bajas no se detectan como mínimo tres regiones que se corresponden con alguna de las figuras {E1, E2, E3, C1, C2, C3, C4}, mientras que para probabilidades elevadas sí.

Los resultados de este proceso de supervisión se han plasmado a través del campo (columna) *acierto* que forma parte de la tabla A.II, que se encuentra recogida en el Anexo II. Este campo resulta clave para evaluar el sistema de probabilidad final de la función gracias a la supervisión humana. Dado el elevado número de tests que se han realizado, este trabajo ha consumido bastantes recursos temporales, pero aún así se consideró elemental hacer un estudio de este tipo. No obstante, con el fin de sintetizar esta información, se ha decidido mostrar inicialmente una tabla resumen en la que se recoge el porcentaje de *aciertos* por probabilidad final, es decir, el ratio *aciertos test por probabilidad final / test totales por probabilidad final*:

Adquisición			Aciertos en % (Probabilidad Final)										
Cámara	Umbral	Escalado	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Canon PowerShot A590 IS	Directo	1024x768	0.0%	0.0%	0.0%	0.0%	0.0%	100%	0.0%	100%	100%	100%	100%
		2592x1944	0.0%	20.0%	9.1%	0.0%	100%	57.1%	100%	100%	100%	100%	100%
	Otsu	1024x768	20.0%	9.1%	0.0%	0.0%	0.0%	50.0%	100%	100%	100%	100%	100%
		2592x1944	0.0%	0.0%	0.0%	0.0%	0.0%	12.5%	0.0%	100%	0.0%	0.0%	100%
Samsung Galaxy ACE	Directo	1024x768	0.0%	0.0%	14.3%	0.0%	0.0%	100%	0.0%	100%	100%	100%	100%
		2592x1944	50.0%	0.0%	14.3%	8.3%	33.3%	54.5%	85.7%	100%	100%	100%	100%
	Otsu	1024x768	0.0%	0.0%	0.0%	0.0%	0.0%	25.0%	100%	100%	0.0%	100%	100%
		2592x1944	0.0%	0.0%	3.8%	0.0%	16.7%	50.0%	0.0%	75.0%	66.7%	0.0%	100%
TOTAL			7.9%	2.9%	4.7%	1.4%	9.4%	46.8%	84%	96%	93.3%	100%	100%

Tabla 5-16 – Matriz de evaluación del resultado de probabilidad final

Para concluir este apartado, se muestra una gráfica basada en columnas verticales que representan de manera visual los valores recogidos en la tabla 5-16, es decir, el porcentaje de aciertos que hay por cada valor de probabilidad final obtenido:

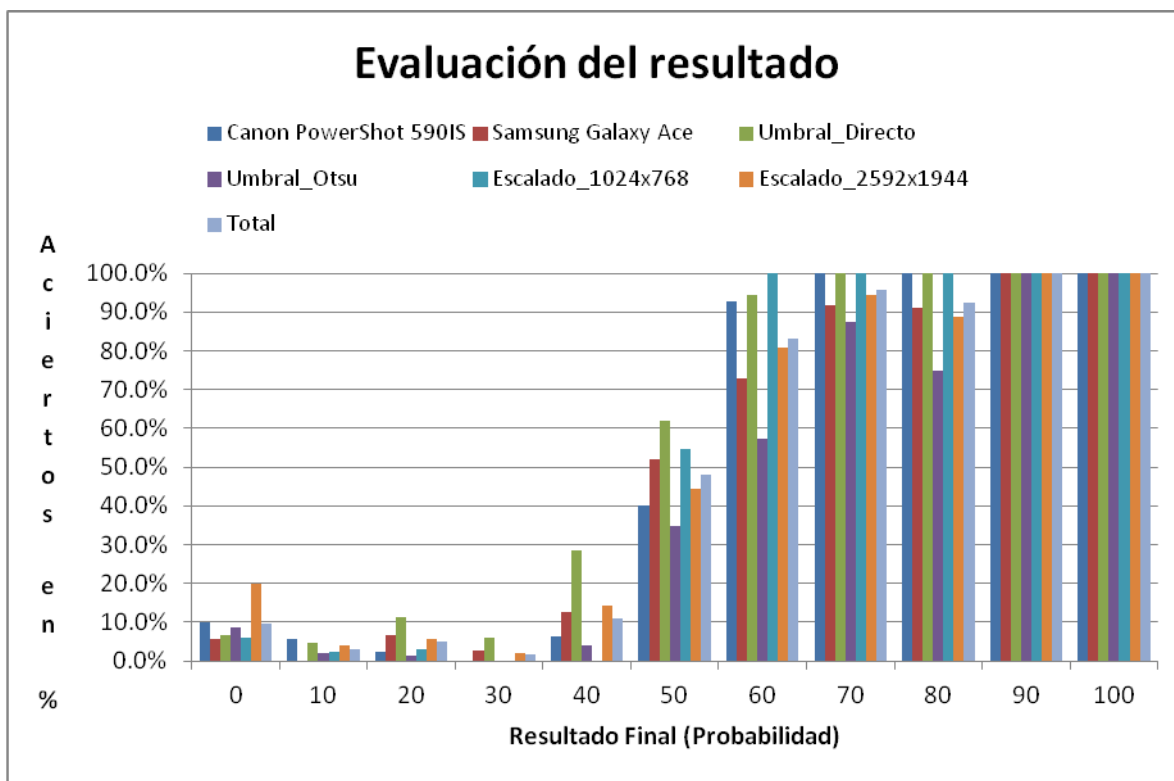


Figura 5-6 – Gráfica de evaluación del resultado (probabilidad final)

5.9 – Resumen y Conclusiones

A partir de las tablas 5-13, 5-14, 5-15, 5-16 y las gráficas 5-2, 5-3, 5-4, 5-5 y 5-6, se pueden postular las siguientes conclusiones:

- El tipo de configuración empleada con mayor nivel de acierto se basa en una resolución espacial (escalado) de 2592×1944 , con un tipo de umbralización *directo* (valor estático del umbral 0.8). Si nos centramos en este último aspecto, es decir, en el umbral, se puede concluir que la tasa de acierto más elevada ha correspondido sin lugar a dudas con el método *directo*, ya que ha tenido una efectividad media más alta que el basado en el método de Otsu.
- Para distancias comprendidas entre los 0 y los 4 metros, el sistema tiene un buen comportamiento cuando se emplean resoluciones espaciales bajas, es decir, de 1024×768 . No obstante, a medida que aumenta la distancia entre el objetivo de la cámara y la figura de la plataforma de aproximación, y por tanto el ratio *área figura / área total imagen* disminuye, se obtienen mejores resultados con resoluciones elevadas (2592×1944).
- En lo que al nivel de inclinación se refiere, el sistema se comporta de forma adecuada hasta los 40° . A partir de este valor, la tasa de acierto disminuye aunque no lo hace de una manera drástica. No se puede establecer una relación directa entre la inclinación y el umbral o la resolución, si bien es

cierto que las configuraciones *1024 x 768 directo*, y *2592 x 1944 directo*, son las que mejores resultados presentan en casi todos los intervalos.

- El tiempo de ejecución para una configuración que emplee una resolución espacial *2592 x 1944* ronda los 13 segundos, mientras que si se utiliza un tipo de resolución inferior (*1024 x 768*), sólo son necesarios 3 segundos. El umbral basado en el método de Otsu, es más pesado computacionalmente, lo que se traduce en un mayor tiempo de ejecución. El tiempo medio de ejecución por acierto de imagen está en torno a los 9 segundos, algo que se justifica porque es necesario emplear muchas más configuraciones basadas en resoluciones altas que bajas, para tener probabilidades superiores al 50%.
- El sistema de probabilidad final que retorna la función como salida, aporta unos resultados bastante coherentes una vez se ha contrastado con el valor devuelto por el análisis que se desprende de la depuración visual basada en el proceso de supervisión humana. Esto se traduce, en que la probabilidad final se encuentra alineada con los resultados reales y objetivos que se obtienen a partir de la evaluación del resultado, lo que otorga un buen nivel de fiabilidad al resultado final devuelto por la función.
- Tal y como se puede observar a partir de las gráficas, el tipo de cámara tiene un impacto bastante significativo sobre el porcentaje de acierto, y por ende, sobre el resultado final. En este sentido cabe reseñar, que con el fin de poder evaluar objetivamente este aspecto, las condiciones del proceso de adquisición han sido exactamente las mismas para las imágenes tomadas con ambas cámaras. En este sentido, la Canon Powershot A590 IS arroja unos resultados mejores en prácticamente todos los ámbitos, algo que se manifiesta con más notoriedad cuando las imágenes se toman en entornos de exteriores con una condición lumínica de sol.

Capítulo 6: Conclusiones y Trabajo Futuro

En este capítulo se describen las principales conclusiones obtenidas, así como las líneas de actuación futuras que se proponen para seguir perfeccionando y profundizando en el objetivo de este Trabajo Final de Máster.

6.1 – Conclusiones del Análisis

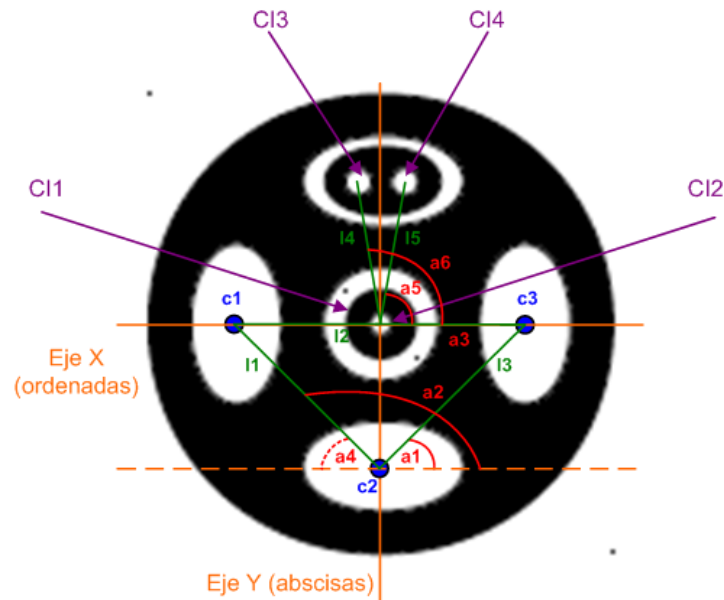
A partir de los resultados y las diversas pruebas que se han efectuado, se han ido registrando las principales conclusiones, las cuales se ponen de manifiesto en forma de lista enumerada a continuación:

1. La resolución espacial de la imagen no tiene una influencia notoria sobre el proceso de binarización en cuanto a la calidad del resultado se refiere. De ahí, que se haya propuesto hacer un escalado para reducir la resolución, ya que disminuye el coste computacional, y por ende el tiempo de ejecución, algo crítico en entornos de tiempo real. No obstante, sí que se debe tener en cuenta, que para aquellas imágenes que denoten una gran sensación de lejanía, es decir, el ratio *área figura / área total de la imagen* tenga un valor bajo como consecuencia de haberse obtenido a gran distancia y sin zoom, la resolución espacial juega un papel decisivo, tal y como se indicó a través del apartado 5.8 – *Resumen y Resultados*. Además debe tenerse en cuenta que su valor influye manifiestamente en muchos de los descriptores geométricos que se basan en el uso de propiedades métricas, como pueden ser el área y la excentricidad, aunque esta última tiene un comportamiento mucho más estable en lo que a operaciones de escalado se refiere.
2. En caso de utilizar el área como criterio de identificación, resulta necesario controlar el nivel de zoom, la distancia, (y la altura en el caso de un UAV) a la plataforma cuando se lleva a cabo el proceso de adquisición de la imagen para poder hacer un primer filtrado preliminar que permita mejorar la eficiencia y efectividad del algoritmo. Se debe tener en cuenta, que el área de una figura geométrica depende de la resolución, y del ratio *área figura / área total imagen*, es decir, del zoom y la distancia a la que se encuentra del objetivo de la cámara. Lo mismo sucede con la excentricidad y el ángulo de inclinación que se forma entre el eje OY (perpendicular al eje OX que suele estar en paralelo al sitio en donde se ubica la plataforma), y la línea recta imaginaria con origen el centro del objetivo de la cámara, y final el centro de la figura dibujada sobre la plataforma de aproximación.
3. A partir de las pruebas efectuadas podemos postular las siguientes premisas en lo que se refiere a la excentricidad de la tres elipses con fondo blanco (E1, E2, y E3) que forman parte de la figura geométrica:
 - a. La excentricidad de las tres elipses tienen valores cambiantes, según el nivel de inclinación y los valores de otros parámetros con los que se haya llevado a cabo el proceso de adquisición. No obstante, es posible afirmar, que un rango aceptable de filtrado de excentricidad podría estar en el intervalo cerrado [0.55, 0.95].
 - b. A partir de la premisa anterior, se deduce que uno de los mejores criterios para evaluar el área y la excentricidad, es comprobar la proporcionalidad entre ambas. En este sentido se dictamina, que el

umbral de diferenciación entre las tres elipses no puede ser superior al 10% en el caso del área, y al 30% en el caso de la excentricidad, si bien es cierto que los análisis parciales comparativos basados en estos dos descriptores (APC4 y APC5 respectivamente), no darán por bueno el resultado si los valores superan el 6.5%, para el caso del área, y el 9.4% en el caso de la excentricidad.

4. Dado que nuestro UAV puede aproximarse desde cualquier dirección a la plataforma de aterrizaje, resulta necesario que el sistema pueda identificar la figura desde cualquier perspectiva. Para ello, se diseñó un algoritmo que una vez han sido etiquetadas las regiones, es capaz de identificar la elipse E2 (elipse central, es decir, la segunda elipse vista en perspectiva de planta contando de izquierda a derecha) de las demás, y a partir de su orientación, determinar el número de grados que hay que rotar la imagen para que ésta siempre quede alineada según se muestra en la figura 3-1, que ha sido mostrada al inicio del capítulo 3.
5. El proceso de identificación de la figura geométrica siempre se lleva a cabo una vez la imagen original ha sido orientada correctamente, quedando con una perspectiva análoga a la que aparece reflejada en la figura 3-1. En este sentido, el proceso de análisis para su identificación analiza los siguientes elementos de la figura geométrica:
 - a. Análisis y estudio de las posiciones de los centros de las elipses sobre un plano cartesiano bidimensional (APC 1).
 - b. Análisis y estudio de los ángulos de los centros de las elipses una vez han sido etiquetadas e identificadas (APC 2).
 - c. Análisis y estudio de las proporciones relativas a las distancias euclídeas existentes entre los centros de las elipses (APC 3).
 - d. Determinación del nivel de similitud del área de las elipses en base al uso de un umbral de error basado en porcentaje (APC 4).
 - e. Determinación del nivel de similitud de la excentricidad de las elipses en base al uso de un umbral de error basado en porcentaje (APC 5).
 - f. Análisis particularizado cada uno de los cuatro círculos por separado, en los que se ha estudiado el ángulo, la distancia, y el ratio del área en tanto por ciento respecto a la media de las elipses (APC 6, 7, 8, 9).
 - g. Análisis de vínculos y relaciones entre los cuatro círculos para poder determinar su coherencia en base a las proporciones geométricas de la figura 6-1 (APC 10).
6. A partir del análisis geométrico realizado, se muestra la imagen de la figura 6-1, en la que se ponen de manifiesto las principales relaciones geométricas existentes entre los elementos más relevantes que componen la plataforma. Dichas relaciones son la base de los análisis parciales comparativos (APCs), que se emplean durante el proceso de identificación y reconocimiento del dibujo. Tal y como se puede observar, la figura 6-1 contiene una serie de números, de tal forma, que los que están entre paréntesis sirven para indicar que son valores usados en la configuración del algoritmo, mientras que los que están fuera (de los paréntesis), son los valores reales que se han obtenido a raíz del análisis geométrico de la perspectiva en planta (cero

grados de inclinación) que tiene la imagen de la figura 6-1. Otro aspecto a destacar, es el relativo a la simetría de la figura, ya que a pesar de que parece que la distancia que hay entre el centro de las elipses E1-E2, y E2-E3 es la misma, resulta levemente menor entre las elipses E2-E3; algo que a simple vista es difícil de apreciar.



Valores Empleados en la Identificación de la Figura	
<p>1) Posiciones Centros Elipses: ErrorAlineacionEjeY_C1_C3 = 12° ErrorAlineacionEjeX_C2 = 29.2% distancia euclidea (C1, C3)</p> <p>2) Ángulos Elipses con Eje X: a1 = 45° a2 = 135° a3 = 0° GradosUmbralErrorOrientacionAngulosElipses = 58.8%</p> <p>3) Ratios Distancias Elipses: l1/l2 ≈ 72% l3/l2 ≈ 69% l3/l1 ≈ 97% PorcentajeUmbralErrorDistanciaElipses = 22.4%</p> <p>4) Area y Excentricidad Elipses 1, 2 y 3 PorcentajeUmbralDiferenciaArea = 6.5% PorcentajeUmbralDiferenciaExcentricidad = 9.4%</p>	<p>5) Ratios Área Círculos CI1.Área / Área_Media_Elipses_1_2_3 ≈ 53% (50%) CI2.Área / Área_Media_Elipses_1_2_3 ≈ 2.4% (2.29%) CI3.Área / Área_Media_Elipses_1_2_3 ≈ 3.8% (3.34%) CI4.Área / Área_Media_Elipses_1_2_3 ≈ 3.8% (3.34%) CI2.Área / CI1.Área ≈ 4.6% (4.4%) CI3.Área / CI1.Área ≈ 7.35% (6.44%) PorcentajeUmbralDiferenciaAreaCirculos_C1 = 45% PorcentajeUmbralDiferenciaAreaCirculos_Elipses = 48%</p> <p>6) Ratios Distancias Círculos: l4/l2 ≈ 51% l5/l2 ≈ 51% PorcentajeUmbralErrorDistanciaCirculos = 28%</p> <p>7) Ángulos Círculos con Eje X: a5 = 80° a6 = 99° PorcentajeUmbralDiferenciaAnguloCirculos = 41.4%</p>

Figura 6-1 – Características geométricas y umbrales de detección

7. Esta imagen, junto con las pruebas efectuadas -que quedan reflejadas en el Anexo I, II y III-, han servido de base a la hora de calibrar el sistema, es decir, asignar los umbrales y las estimaciones geométricas en la sección de configuración del algoritmo. A partir de la figura 6-1, deben tenerse en cuenta las siguientes premisas que se indican a continuación:
 - a. A pesar de que la figura geométrica no es completamente simétrica (algo que a simple vista resulta difícil de dilucidar), esto no ha sido un impedimento para diseñar un algoritmo que, utilizando diversos patrones geométricos basados en relaciones geométricas, sea capaz de identificarla. Es más, esta asimetría ha sido de valiosa utilidad de

- cara a la identificación de cada elemento, ya que ha servido de base para calcular la orientación de la figura, gracias a que las distancias entre las elipses E1-E2, y E2-E3, son distintas.
- b. Al no ser simétrica, el triángulo que forman los centros de las elipses por medio de los segmentos $I1$, $I2$ y $I3$ no constituyen un triángulo equilátero. De hecho, el valor de los ángulos $a1$ y $a3$ no es el mismo, algo que resulta difícil de analizar a simple vista. También sucede con la distancia entre los centros $c1$ y $c3$.
 - c. En la leyenda correspondiente a la figura 6-1, se muestra gran parte de los valores empleados en la sección de configuración del programa, que sirve para parametrizar el análisis de la figura una vez se han segmentado y etiquetado las diversas regiones. En este sentido, cabe reseñar, que todos los valores indicados en la leyenda, coinciden con los valores asignados a los umbrales que se parametrizan en la sección de configuración del sistema. Tales valores son la base con la que trabajan los análisis parciales comparativos (APCs).
 - d. Los distintos colores, representan y clasifican los diversos tipos de criterios que han sido utilizados a la hora de analizar la figura, para poder deliberar geoméricamente si dicha figura se corresponde con el dibujo serigrafiado sobre la superficie de nuestra plataforma de aproximación.
8. Para la detección de elipses no basta sólo con utilizar umbrales de filtrado basados en el área y la excentricidad, sino que hay que utilizar también algún descriptor de regiones adicional basado en *propiedades topológicas* como puede ser el *número de Euler*. Este número que se define formalmente como: $E = C - H$, donde C es el número de componentes conexas, y H el número de huecos de la imagen. Permite discriminar de forma sencilla entre ciertas clases de objetos, y además resulta invariante frente a traslaciones, rotaciones y cambios de escala. Por ejemplo, en el caso de las elipses, se sabe de antemano que su valor será 1, lo que posibilita el uso de dicha propiedad como un criterio adicional de discriminación.
9. En lo que concierne al proceso de identificación de los cuatro círculos ($C1$, $C2$, $C3$ y $C4$), hay que subrayar, que a mayores del área y la excentricidad, también se ha utilizado el número de Euler como criterio de filtrado, de tal modo, que la región tendrá que tener un valor que coincida con alguno de los elementos del siguiente conjunto: $\{-2, -1, 0, 1\}$ para poder ser analizada geoméricamente mediante el uso de descriptores basados en propiedades métricas. La justificación de tales valores se fundamenta en el soporte que se desea otorgar a imágenes que hayan sido adquiridas con niveles de inclinación heterogéneos durante el proceso de adquisición. Este nivel de inclinación respecto al suelo, hace que los círculos puedan ser vistos como elipses, lo que repercute en los descriptores basados en las propiedades métricas, como puede ser la excentricidad o el área.

6.2 – Conclusiones de Resultados

A partir de los resultados y las diversas pruebas que se han efectuado en el capítulo 5 – *Resultados y Pruebas*, para satisfacer los objetivos perseguidos en este TFM, se han registrado las siguientes conclusiones que se exponen a continuación:

1. El sistema diseñado es capaz de detectar correctamente un gran número de imágenes heterogéneas que han sido obtenidas en condiciones variables de distancia, inclinación, resolución y luminosidad. En este sentido, a pesar de que el sistema diseñado puede trabajar con distintos niveles de resolución, se recomienda en primera instancia por temas de eficiencia y tiempo de ejecución, hacer un proceso de escalado y trabajar con resoluciones bajas, en torno a un mega píxel, siempre que la distancia entre el objetivo de la cámara y la figura no sea superior a los 4 metros, aunque dependerá del nivel de zoom que se utilice. No obstante, en este sentido, tras analizar los resultados, deben tenerse presentes estas dos consideraciones:
 - a. Si la figura presenta una sensación de lejanía en la imagen, esto es, el cociente entre el área de la figura y el área total de la imagen tiene un valor muy bajo, los resultados son mejores cuando se emplean niveles altos de resolución. En este sentido cabe reseñar, que si se trabaja con resoluciones inferiores a 2 megapíxeles, el sistema no va a ser capaz de detectar los elementos que componen la figura ya que el nivel de detalle no es lo suficientemente bueno como para poder hacer un análisis. Esto se observa en la imagen IMG_A044, ya que cuando trabaja con una resolución espacial de 1024 x 768 no puede detectar las elipses, mientras que si trabaja con resoluciones de 2592 x 1944, sí que se detectan, y por ende también se reconoce de forma adecuada la plataforma.
 - b. Por contra, si se utiliza un nivel de zoom apropiado, y/o la distancia desde el objetivo de la cámara a la plataforma es baja, es decir, la foto transmite sensación de cercanía porque el ratio *área figura / área imagen* es elevado, entonces se obtienen mejores resultados cuando se trabajan con resoluciones pequeñas. En este sentido, cuando se usan resoluciones superiores a 2 megapíxeles (1600 x 1200 píxeles, por ejemplo) existe una tasa de fallo más elevada como consecuencia de problemas relacionados con la identificación de elementos. Estos problemas surgen como consecuencia del aumento en el número de regiones que poseen atributos análogos durante el proceso de segmentación.
2. A pesar de que el sistema se comporta de forma apropiada cuando se obtienen imágenes a diferentes alturas, o en la misma altura con diferentes niveles de zoom, en un entorno real y productivo se recomienda llevar a cabo el proceso de adquisición con unos niveles de distancia, altura, zoom e inclinación constantes. Lo importante es que el ángulo de inclinación y el porcentaje que ocupa el área de la figura respecto al tamaño total de ésta sea constante, ya que facilita la asignación de umbrales para una resolución determinada que permita mantener el proceso de adquisición dentro de un entorno controlado. No obstante, los resultados han sido muy alentadores si por la casuística operativa, se tienen que obtener las imágenes a diferentes alturas y distancias, lo que implica una heterogeneidad tanto en el ángulo de inclinación como en el valor del ratio que relaciona el tamaño de la figura con el total de la imagen. A pesar de que los resultados son excelentes ante este tipo de situaciones dinámicas, el riesgo de que existan imágenes que no sean reconocidas aumenta en consideración, respecto al hipotético escenario que se daría si todos los parámetros estuvieran bajo valores controlados.
3. A la hora de evaluar el reconocimiento final de los cuatro círculos mediante el uso de los análisis parciales comparativos (APC6, APC7, APC8 y APC9), resulta conveniente usar la diferencia que hay entre los ángulos que forman

dichas figuras, ya que este tipo de relación geométrica ha demostrado ser más estable que el vínculo que se establece en función a las proporciones o ratios existentes entre sus áreas. Esto se debe primordialmente, a que el área de estos círculos es bastante reducida (sobre todo cuando se trabaja en imágenes aéreas y el zoom no resulta suficiente), y por tanto, cualquier detalle o imperfección que se produzca en la imagen (sobre todo si el valor del ratio área figura / área total imagen es bastante bajo), tiene un impacto palmario en el porcentaje final, y por ende sobre el proceso de identificación. A raíz de las pruebas efectuadas, y los resultados alcanzados a la hora de valorar este descriptor geométrico, se decidió establecer como umbral una media entre las imágenes que presentaban un valor más bajo (como puede ser la imagen IMG_A069), y las imágenes que arrojan un valor geométrico más preciso (como puede ser la IMG_A065), por haber sido adquiridas con valores prácticamente nulos en lo que al ángulo de inclinación se refiere.

4. Cabe reseñar que durante el desarrollo del software, sobre todo en lo que al proceso de detección de círculos se refiere, se implementaron y probaron algoritmos más complejos y elaborados que introducían una leve mejora en cuanto a la tasa de acierto, pero que requerían de un coste computacional tan elevado que no se justificaba en absoluto su uso. Por ejemplo, para la función *IdentificarC3_C4*, se elaboró una versión previa más compleja, que hacía cálculos con pares de puntos (el número de emparejamientos sale del total de combinaciones sin repetición), para hallar la diferencia en términos de valor absoluto en base a la distancia euclídea y al diferencial de ángulos. Sin embargo, el coste computacional requerido, algo que se agrava cuando el arreglo de círculos es elevado (situación que se da en imágenes obtenidas a gran altura donde el ratio figura / imagen es muy bajo, y por tal motivo se requieren resoluciones altas), comprometía en exceso el rendimiento y tuvo que ser descartado. En aplicaciones donde se toman decisiones en tiempo real como ésta, la relación entre la tasa de acierto y el tiempo de ejecución, juega un papel fundamental que ha de ser estudiado con detenimiento.
5. El entorno donde se ubique la plataforma tiene una influencia notoria: hojas que pueden dar lugar a elipses, árboles que generen zonas de sombra, etc. Se considera por tanto muy importante, que la plataforma de aproximación del UAV, se encuentre impresa en una superficie que sea lo más uniforme posible en lo que al color y a la textura se refiere. Todas las pruebas e imágenes obtenidas, han tomado como base un papel blanco de tamaño A3 sobre el que se ha impreso la figura de aproximación. Pues bien, en este sentido cabe reseñar, que el rectángulo blanco que forma el papel A3 en torno a la figura, ha sido de una utilidad muy valiosa durante el proceso de identificación, ya que al binarizar y segmentar la imagen se forma una región homogénea en los alrededores a la figura, que impide que surjan regiones espúreas que puedan "confundir" al sistema. Además, dado que todas las rutinas de identificación de círculos y elipses, utilizan la distancia euclídea como descriptor geométrico (propiedad métrica) para seleccionar los elementos finales de entre los posibles candidatos, se reduce la tasa de error y aumenta la efectividad total.

6.3 – Líneas de Actuación Futura

Finalmente, para concluir con esta memoria, se ha estimado que sería oportuno comentar una serie de propuestas de mejora, que sirviesen como punto de partida para tratar de evolucionar y perfeccionar el sistema descrito en este documento:

1. Implementar un algoritmo que establezca un vínculo entre los umbrales y el porcentaje que representa el área de la figura con relación al número total de píxeles que posee la imagen. Cuando la imagen da una sensación de cercanía, es decir, el cociente entre el área de la figura y el área total de la imagen se aproxima a 1, se obtienen mejores resultados cuando se trabaja con resoluciones elevadas (superiores a los 3 megapíxeles). Sin embargo, cuando la imagen da sensación de lejanía, es decir, el cociente entre el área de la figura y el área total de la imagen se aproxima a 0, se obtienen buenos resultados cuando se utilizan resoluciones más bajas con un valor cercano al megapíxel (por ejemplo 1024 x 768).
2. Siguiendo la línea anterior, elaborar un estudio que permita desarrollar un algoritmo que decida con qué resolución tiene que acometer el proceso de reconocimiento en base al ratio que presenta el área de la figura respecto al número total de píxeles de la imagen. Una misma imagen, con niveles de escalado distintos, puede arrojar resultados totalmente dispares, ya que se ha comprobado que cuanto mayor sea el ratio entre la figura y el total de la imagen (vinculado a la distancia física de la plataforma de aproximación respecto al objetivo de la cámara digital y al nivel de zoom utilizado), más nivel de resolución hay que emplear durante el proceso de reconocimiento de la figura. Por ejemplo, cuando la imagen IMG_A043 es procesada con una resolución total de 2592 x 1944, el sistema le otorga una probabilidad de reconocimiento del 100%, mientras que si se lleva a cabo una operación de escalado para trabajar con una resolución de 1600 x 1200 la probabilidad de reconocimiento desciende bruscamente hasta el 0%.
3. El algoritmo desarrollado para la identificación y reconocimiento de la figura, arroja unos buenos resultados cuando el ángulo de inclinación sobre la horizontal no es excesivamente acusado, puesto que dependiendo del grado de inclinación y la perspectiva, se pueden dar ciertas circunstancias en las que existan diferencias notorias en la excentricidad de las elipses (que sean superiores incluso al umbral del 25%). Por lo tanto, para aquellos niveles de inclinación muy elevados, donde no sea posible realizar una rotación para tratar de orientar la imagen haciendo que tenga la misma orientación que la de la figura 3-1, el sistema no se comportará de manera adecuada. De este modo, si no se introduce la línea de mejora anterior, pero se desea que el algoritmo sea operativo cuando la adquisición de las imágenes se realiza a muy baja altura pero a una distancia normal, resulta necesario realizar un estudio para poder hacer frente al problema derivado de la rotación en perspectivas de imágenes en dos dimensiones (2D) que posean un ángulo de inclinación ostensible.
4. Introducir un sistema basado en probabilidades más sofisticado, que tenga en cuenta los umbrales de error que se han utilizado a la hora de realizar el cálculo de probabilidades una vez se ha llevado a cabo el análisis geométrico de la imagen. Esto se debe, por ejemplo, a que no es lo mismo que el análisis parcial de diferencia de excentricidades de elipses se acierte cuando se ha establecido un umbral de diferenciación de excentricidades que tiene un valor del 5% (*Configuracion.PorcentajeUmbralDiferenciaExcentricidad*), que si se utiliza un valor del 10% para el mismo umbral. En este segundo caso, si la probabilidad máxima de identificación para este tipo de análisis es de 0.2, habría que idear e implementar un sistema que tenga en cuenta el valor del umbral a la hora de hacer el cálculo parcial de las probabilidades para cada Análisis Parcial Comparativo (APC).

5. Añadir nuevos elementos geométricos durante el proceso de identificación y análisis de la figura, de tal modo que se introduzcan nuevas figuras (más círculos o elipses que quedaron sin utilizar), a la hora de determinar si la imagen que se está procesando contiene la figura geométrica que está dibujada sobre la plataforma de aproximación. De esta manera, se puede llegar a o obtener un doble beneficio tal y como se indica a continuación:
 - a. Por un lado, se conseguiría que aumentara el nivel de convicción a la hora de realizar el proceso de identificación de la figura, dado que al introducir nuevos elementos que debieran ser característicos, propios, y unívocos de la figura, se provocaría un aumento en relación al nivel de certeza asociado a la probabilidad de detección.
 - b. Por otro lado, se podría aumentar el número de situaciones adversas en los que sería posible perpetrar el análisis de la figura, puesto que habría nuevos elementos a considerar. Tal y como está planteada la solución, si existe un problema de reflejos y no se pueden detectar las tres elipses, el sistema no podría llevar a cabo el reconocimiento final, dado que la identificación de los círculos está vinculada en parte a la identificación de las elipses. Sin embargo, de esta manera, si se introducen reflejos o partes ciegas en la imagen que provoquen que alguna de las elipses sea indetectable, todavía podría ser factible la identificación de la plataforma. Resumiendo, se minimizarían puntos troncales de fallo ante problemas que puedan surgir durante el proceso de adquisición, siempre que la detección de estos nuevos elementos no se encuentre vinculada a los que ya han sido utilizados.
6. Realizar la identificación y el reconocimiento de la figura sin tener que orientar siempre la imagen de una forma determinada, ya que de este modo se reducen las operaciones necesarias para efectuar la rotación, y por ende se disminuye el tiempo de ejecución.
7. Tomando como eje de desempeño las actuaciones que se han perpetrado de cara a mejorar la eficiencia del algoritmo, y empleando como base el estudio realizado de cara al perfeccionamiento de las funciones *IdentificarC1_C2* e *IdentificarC3_C4*, se recomienda llevar a cabo un estudio de la eficiencia computacional que determine el coste asintótico final con el fin de diseñar un algoritmo que sea más eficiente desde el punto de vista computacional, y por lo tanto, que necesite menos ciclos de reloj. Esto se traducirá, casi con toda seguridad, en un menor tiempo de ejecución a igualdad de condición de hardware. Esta mejora resulta especialmente relevante, cuando se trabaja con imágenes que producen una gran sensación de lejanía (el ratio entre la figura y el total de la imagen es muy alto), ya que suelen requerir de gran resolución para poder realizar la identificación de la imagen. Esto ocasiona un aumento en el número de operaciones como consecuencia de tener que trabajar con un mayor número de píxeles, lo que da lugar normalmente, a un aumento en el tamaño de los *arrays* de círculos candidatos a raíz de obtener un mayor número de regiones tras el proceso de segmentación.

Referencias y Herramientas Empleadas

A continuación se indican cuales han sido las principales referencias consultadas:

- [1] Pajares, G., Cruz, J.M. (2007). *Visión por computador: imágenes digitales y aplicaciones Fundamentos de Robótica*. Madrid: RA-MA., Segunda Edición.
- [2] Barrientos, A., Peñin, L.F., Balaguer, C., Aracil, R. (2007). *Fundamentos de Robótica*. Madrid: McGraw-Hill Interamericana de España S.L., Segunda Edición.
- [3] Cruz, J.M., Sánchez, B., Pajares, G. (2012). *Sistema de aproximación a una plataforma de un vehículo no tripulado mediante análisis visual*. Patente WO2012085305A1 [En línea]
<http://www.google.com/patents/WO2012085305A1?cl=es>
[Fecha de la consulta: Junio/2014]
- [4] Guili, X., Yong, Z., Shengyu, J., Yuehua, Ch., Yupeng, T. (2009). *Research on computer vision-based for UAV autonomous landing on a ship*. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu, China [En línea]
http://www.ideas.vbotix.com/?wpfb_dl=46
[Fecha de la consulta: Junio/2014]
- [5] Lange, S., Sünderhauf, N., Protze, P. (2008). *Autonomous Landing for a Multicopter UAV Using Vision*. Chemnitz University of Technology, 09126 Chemnitz, German [En línea]
<https://www.tu-chemnitz.de/etit/proaut/rsrsc/simpar08.pdf>
[Fecha de la consulta: Junio/2014]
- [6] Srikanth, S., Montgomery, J.F., Sukhatme, G. S. (2003). *Vision-based Autonomous Landing of an Unmanned Aerial Vehicle*. California Institute of Technology, Pasadena, Los Angeles [En línea]
http://www.researchgate.net/publication/2488105_Vision-based_Autonomous_Landing_of_an_Unmanned_Aerial_Vehicle/file/5046351c7ada50dfe8.pdf
[Fecha de la consulta: Junio/2014]
- [7] Sharp, C.S., Shakernia, O., Shankar, S.S. (2001). *A Vision System for Landing an Unmanned Aerial Vehicle*. University of California, Berkley. [En línea]
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.24.7707&rep=rep1&type=pdf>
[Fecha de la consulta: Junio/2014]
- [8] Cesetti, A., Frontoni, E., Mancini, A., Zingaretti, P., Longhi, S. (2009). *A Vision-Based Guidance System for UAV Navigation*. Marche Polytechnic University, Ancona, Italy [En línea]
<http://cmapspublic.ihmc.us/rid=1K7WQVGMR-Y19F08-1YKQ/A%20Vision-Based%20Guidance%20System%20for%20UAV%20Navigation.pdf>
[Fecha de la consulta: Junio/2014]

- [9] Sin-Der, L., Ching-Yaw, T., Young-Zehr, K., Chi-Chun, H., Chih-Kai, K. (2010). *Autopilot System Based on Color Recognition Algorithm and Internal Model Control Scheme for Controlling Approaching Maneuvers of a Small Boat*. IEEE Journal of Oceanic Engineering (Vol.35, No. 2, Pag(s) 376 – 387), China [En línea]
<http://www.deepdyve.com/lp/institute-of-electrical-and-electronics-engineers/autopilot-system-based-on-color-recognition-algorithm-and-internal-8ZHdV52WIF> [Fecha de la consulta: Junio/2014]
- [10] Zongji, Ch., Lei, C., Rui, Z., Weiqi, L. (2008). *Unmanned aircraft landing navigation system based on vision*. Patente CN 101109640 A [En línea]
<http://www.google.com/patents/CN101109640A?cl=en>
[Fecha de la consulta: Junio/2014]
- [11] Roy, P., Yu, J., Linden, D.S. (2008). *Methods, apparatus and systems for enhanced synthetic vision and multi-sensor data fusion to improve operational capabilities of unmanned aerial vehicles*. Patente US20080215204 [En línea]
<http://www.google.sc/patents/US20080215204>
[Fecha de la consulta: Junio/2014]
- [12] Blenkhorn, K.P., O'Hara, S.V. (2009). *Vision-based automated landing system for unmanned aerial vehicles*. Patente US20090306840 [En línea]
<http://www.google.com/patents/US20090306840>
[Fecha de la consulta: Junio/2014]
- [13] Grzywna, J., Da Frota, B., Meuse, S. (2007). *System and method for onboard vision processing*. Patent US20070093945 [En línea]
<http://www.google.nl/patents/US20070093945>
[Fecha de la consulta: Junio/2014]
- [14] Shin-Je, Ch., Jung-Ho, M., Seung-Kie, Ch., San, K, Sung-Sik, S. (2008). *Automatic recovery method of a UAV using image information to stably recover the UAV by obtaining real-time image information of a landing symbol on a recovery netting*. Patente 1020070058957 [En línea]
<http://patent.ipexl.com/KR/1020070058957.html>
[Fecha de la consulta: Junio/2014]
- [15] Avitzour, D. (1996). *Mobile robot location determination employing error-correcting distributed landmarks*. Patente US5525883 [En línea]
<http://www.google.com/patents/US5525883>
[Fecha de la consulta: Junio/2014]
- [16] Domenech, A. (2009). *Proyecto Fin de Carrera, "Reconocimiento Gestual a través de Webcam"*. Escuela Técnica Ingeniería Superior ICAI. Universidad Pontificia de Comillas [En línea]
<http://es.scribd.com/doc/97504792/9/BINARIZAR-UNA-IMAGEN>
[Fecha de la consulta: Junio/2014]
- [17] Carcedo, Y., Franco, A. (2004). *Programa de segmentación de regiones en imágenes médicas en MATLAB*. Escuela de Ingeniería. Departamento de Ingeniería Electrónica. Universidad de las Américas Puebla. Cholula, Puebla, México [En línea]
http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/carcedo_y_a/por_tada.html
[Fecha de la consulta: Junio/2014]

- [18] Elipse - Wikipedia, la enciclopedia libre [En línea]
<http://es.wikipedia.org/wiki/Elipse>
[Fecha de la consulta: Junio/2014]
- [19] Definición de Elipse MATEMÁTICAS [En línea]
<http://es.scribd.com/doc/72512972/Definicion-de-Elipse-MATEMATICAS>
[Fecha de la consulta: Junio/2014]
- [20] Definición: Elipse [En línea]
<http://www.mathematicsdictionary.com/spanish/vmd/full/e/ellipse.htm>
[Fecha de la consulta: Junio/2014]
- [21] Cónicas [En línea]
<http://www.ehu.es/~mtpalezp/conicas.pdf>
[Fecha de la consulta: Junio/2014]
- [22] Circunferencia - Wikipedia, la enciclopedia libre [En línea]
<http://es.wikipedia.org/wiki/Circunferencia>
[Fecha de la consulta: Junio/2014]
- [23] Trigonometry - How to calculate the angle between two points relative to the horizontal axis? - Stack Overflow [En línea]
<http://stackoverflow.com/questions/7586063/how-to-calculate-the-angle-between-two-points-relative-to-the-horizontal-axis>
[Fecha de la consulta: Junio/2014]
- [24] Distancia Euclídea - Wikipedia, la enciclopedia libre [En línea]
http://es.wikipedia.org/wiki/Distancia_euclidiana
[Fecha de la consulta: Junio/2014]
- [25] Measure properties of image regions - MATLAB regionprops – MathWorks [En línea]
<http://www.mathworks.es/es/help/images/ref/regionprops.html>
[Fecha de la consulta: Junio/2014]
- [26] Segmentación (electrónica) - Wikipedia, la enciclopedia libre [En línea]
http://es.wikipedia.org/wiki/Segmentaci%C3%B3n_%28electr%C3%B3nica%29
[Fecha de la consulta: Junio/2014]
- [27] Sorting Algorithms sample codes on JAVA, C++ and MATLAB [En línea]
http://thilinasameera.wordpress.com/2011/06/01/sorting-algorithms-sample-codes-on-java-c-and-matlab/#_Bubble_Sort
[Fecha de la consulta: Junio/2014]
- [28] Complejidad Computacional [En línea]
<http://www.slideshare.net/joemmanuel/complejidad-computacional>
[Fecha de la consulta: Junio/2014]
- [29] Segmentación por Umbralización – Método de Otsu [En línea]
<http://iaci.unq.edu.ar/materias/vision/archivos/apuntes/Segmentaci%C3%B3n%20por%20umbralizaci%C3%B3n%20-%20M%C3%A9todo%20de%20Otsu.pdf>
[Fecha de la consulta: Junio/2014]

- [30] Niblack, W. (1986). *An introduction to digital image processing*. Englewood Cliffs: Prentice-Hall International.
- [31] Rosenfeld, A., Pfaltz, D. (1966). *Sequential Operations in Digital Picture Processing*. Journal of the ACM. Vol. 13, no 4, pp. 471-494.
- [32] Duda, R., O., Hart, P.E., Stork, D.G. (2000). *Pattern Classification*. EEUU: Wiley-Interscience, Segunda Edición.
- [33] Hu, M.K. (1962). *Visual pattern recognition by moment invariants*. IRE Transaction on Information Theory. Vol. 8, pp. 179-187.
- [34] Flusser, J., Suk, T. (1993). *Pattern recognition by affine moments invariants*. Pattern Recognition. Vol. 24, no. 1, pp. 167-174.
- [35] Hough, P. V. C. (1962). *Method and means for recognizing complex patterns*. U.S. Patent 3 069 654, Dec. 18, 1962.
- [36] Duda, R. O., Hart, P. E. (1972). *Use of the Hough transformation to detect lines and curves in pictures*. Commun. ACM. Vol. 15, no. 1, pp. 11-15.
- [37] Yin, P. (1998). *A new circle/ellipse detector using genetic algorithms*, Proceedings of the IPPR on CVGIP Taipei, pp. 362-368.
- [38] Dave, R.N. (1992). *Generalized fuzzy c-shells clustering and detection of circular and elliptical boundaries*, Pattern Recognition, Vol. 25, No 7, pp. 713-721.
- [39] Fitzgibbon, A., Pilu, M., Fisher, R. (1999). *Direct Least Square Fitting of Ellipses*, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 21.5, pp. 476-480.
- [40] Zhang, S.C., Liu, Z.Q. (2004). *A Robust, real-time ellipse detector*. Pattern Recognition, Vol. 38, pp. 273-287.
- [41] Otsu, N. (1979). *A threshold selection method from gray-level histogram*. IEEE Transactions on Systems Man and Cybernetics, 9, pp. 62-66.
- [42] Sauvola, J., Pietikäinen, M. (2000). *Adaptive document image binarization*. Pattern Recognition. Vol 33(2), pp. 225-236.

Por otro lado, es necesario comentar, que las herramientas software empleadas para la elaboración de este documento han sido las siguientes:

- MATLAB 7.12.0.365 R2011a (18 de Marzo de 2011)
- VMWare Player 6.0.1 build-1379776
- Microsoft Word y Excel 2007
- Microsoft Visio 2003
- Microsoft Access 2007
- Inclinometer Free versión 2.6 para Android

ANEXO I – Tabla de Imágenes (A.I)

A continuación se muestra la tabla de imágenes en la que se describen las principales características de las fotografías empleadas en la fase de pruebas:

DETALLE DE LAS IMÁGENES DE ENTRADA											
Nombre	Cámara	Tipo	Flash	Inclinación (Angulo °)	Altura (OY)	Longitud (OX)	Distancia (Total)	Tipo de Luz	Orientación (Figura)	Zoom	Resolución
IMG_A001	Canon Powershot	Exterior	No	10.39	3.00	0.55	3.05	Nublado	180	No	2592 x 1944
IMG_A002	Canon Powershot	Exterior	No	5.42	5.8	0.55	5.83	Nublado	180	No	2592 x 1944
IMG_A003	Canon Powershot	Exterior	No	3.66	8.6	0.55	8.62	Nublado	180	No	2592 x 1944
IMG_A004	Canon Powershot	Exterior	No	20.97	3.00	1.15	3.21	Nublado	180	No	2592 x 1944
IMG_A005	Canon Powershot	Exterior	No	11.21	5.8	1.15	5.91	Nublado	180	No	2592 x 1944
IMG_A006	Canon Powershot	Exterior	No	7.62	8.6	1.15	8.68	Nublado	180	No	2592 x 1944
IMG_A007	Canon Powershot	Exterior	No	46.40	3.00	3.15	4.35	Nublado	180	No	2592 x 1944
IMG_A008	Canon Powershot	Exterior	No	28.51	5.8	3.15	6.60	Nublado	180	No	2592 x 1944
IMG_A009	Canon Powershot	Exterior	No	20.12	8.6	3.15	9.16	Nublado	180	No	2592 x 1944
IMG_A010	Canon Powershot	Exterior	No	59.78	3.00	5.15	5.96	Nublado	180	No	2592 x 1944
IMG_A011	Canon Powershot	Exterior	No	41.60	5.8	5.15	7.76	Nublado	180	No	2592 x 1944
IMG_A012	Canon Powershot	Exterior	No	30.91	8.6	5.15	10.02	Nublado	180	No	2592 x 1944
IMG_A013	Canon Powershot	Exterior	No	14.57	3.00	0.78	3.10	Sombra y Sol	180	No	2592 x 1944
IMG_A014	Canon Powershot	Exterior	No	7.66	5.8	0.78	5.85	Sombra y Sol	180	No	2592 x 1944
IMG_A015	Canon Powershot	Exterior	No	5.18	8.6	0.78	8.64	Sombra y Sol	180	No	2592 x 1944
IMG_A016	Canon Powershot	Exterior	No	48.99	3.00	3.45	4.57	Sombra y Sol	180	No	2592 x 1944

IMG_A017	Canon Powershot	Exterior	No	30.75	5.8	3.45	6.75	Sombra y Sol	180	No	2592 x 1944
IMG_A018	Canon Powershot	Exterior	No	21.86	8.6	3.45	9.27	Sombra y Sol	180	No	2592 x 1944
IMG_A019	Canon Powershot	Exterior	No	61.17	3.00	5.45	6.22	Sombra y Sol	180	No	2592 x 1944
IMG_A020	Canon Powershot	Exterior	No	43.22	5.8	5.45	7.96	Sombra y Sol	180	No	2592 x 1944
IMG_A021	Canon Powershot	Exterior	No	32.36	8.6	5.45	10.18	Sombra y Sol	180	No	2592 x 1944
IMG_A022	Canon Powershot	Exterior	No	10.39	3.00	0.55	3.05	Soleado	180	No	2592 x 1944
IMG_A023	Canon Powershot	Exterior	No	5.42	5.8	0.55	5.83	Soleado	180	No	2592 x 1944
IMG_A024	Canon Powershot	Exterior	No	3.66	8.6	0.55	8.62	Soleado	180	No	2592 x 1944
IMG_A025	Canon Powershot	Exterior	No	20.97	3.00	1.15	3.21	Soleado	180	No	2592 x 1944
IMG_A026	Canon Powershot	Exterior	No	11.21	5.8	1.15	5.91	Soleado	180	No	2592 x 1944
IMG_A027	Canon Powershot	Exterior	No	7.62	8.6	1.15	8.68	Soleado	180	No	2592 x 1944
IMG_A028	Canon Powershot	Exterior	No	46.40	3.00	3.15	4.35	Soleado	180	No	2592 x 1944
IMG_A029	Canon Powershot	Exterior	No	28.51	5.8	3.15	6.60	Soleado	180	No	2592 x 1944
IMG_A030	Canon Powershot	Exterior	No	20.12	8.6	3.15	9.16	Soleado	180	No	2592 x 1944
IMG_A031	Canon Powershot	Exterior	No	59.78	3.00	5.15	5.96	Soleado	180	No	2592 x 1944
IMG_A032	Canon Powershot	Exterior	No	41.60	5.8	5.15	7.76	Soleado	180	No	2592 x 1944
IMG_A033	Canon Powershot	Exterior	No	30.91	8.6	5.15	10.02	Soleado	180	No	2592 x 1944
IMG_A034	Canon Powershot	Exterior	No	46.40	3.00	3.15	4.35	Sombra y Sol	135	No	2592 x 1944
IMG_A035	Canon Powershot	Exterior	No	28.51	5.8	3.15	6.60	Sombra y Sol	135	No	2592 x 1944
IMG_A036	Canon Powershot	Exterior	No	20.12	8.6	3.15	9.16	Sombra y Sol	135	No	2592 x 1944
IMG_A037	Canon Powershot	Exterior	No	46.40	3.00	3.15	4.35	Soleado	225	No	2592 x 1944
IMG_A038	Canon Powershot	Exterior	No	28.51	5.8	3.15	6.60	Soleado	225	No	2592 x 1944
IMG_A039	Canon Powershot	Exterior	No	20.12	8.6	3.15	9.16	Soleado	225	No	2592 x 1944
IMG_A040	Canon Powershot	Exterior	No	39.24	3.00	2.45	3.87	Nublado	225	No	2592 x 1944
IMG_A041	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Nublado	225	No	2592 x 1944

IMG_A042	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Nublado	225	No	2592 x 1944
IMG_A043	Canon Powershot	Exterior	No	39.24	3.00	2.45	3.87	Nublado	180	No	2592 x 1944
IMG_A044	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Nublado	180	No	2592 x 1944
IMG_A045	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Nublado	180	No	2592 x 1944
IMG_A046	Canon Powershot	Exterior	No	39.24	3.00	2.45	3.87	Soleado	180	No	2592 x 1944
IMG_A047	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Soleado	180	No	2592 x 1944
IMG_A048	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Soleado	180	No	2592 x 1944
IMG_A049	Canon Powershot	Exterior	No	39.24	3.00	2.45	3.87	Nublado	45	No	2592 x 1944
IMG_A050	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Nublado	45	No	2592 x 1944
IMG_A051	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Nublado	45	No	2592 x 1944
IMG_A052	Canon Powershot	Exterior	No	39.24	3.00	2.45	3.87	Soleado	45	No	2592 x 1944
IMG_A053	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Soleado	45	No	2592 x 1944
IMG_A054	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Soleado	45	No	2592 x 1944
IMG_A055	Canon Powershot	Exterior	No	39.24	3.00	2.45	3.87	Nublado	45	No	2592 x 1944
IMG_A056	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Nublado	45	No	2592 x 1944
IMG_A057	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Nublado	45	No	2592 x 1944
IMG_A058	Canon Powershot	Exterior	No	39.24	3.00	2.45	3.87	Soleado	45	No	2592 x 1944
IMG_A059	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Soleado	45	No	2592 x 1944
IMG_A060	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Soleado	45	No	2592 x 1944
IMG_A061	Canon Powershot	Exterior	No	22.90	5.8	2.45	6.30	Nublado	180	Si	2592 x 1944
IMG_A062	Canon Powershot	Exterior	No	15.90	8.6	2.45	8.94	Nublado	180	Si	2592 x 1944
IMG_A063	Canon Powershot	Exterior	No	12.33908728	1.6	0.35	1.637833935	Nublado	0	No	2592 x 1944
IMG_A064	Canon Powershot	Exterior	No	16.38954033	1.7	0.50	1.772004515	Nublado	180	No	2592 x 1944
IMG_A065	Canon Powershot	Exterior	No	8.746162263	0.65	0.10	0.657647322	Nublado	180	No	2592 x 1944
IMG_A066	Canon Powershot	Exterior	No	50.71059314	0.45	0.55	0.71063352	Nublado	315	No	2592 x 1944

IMG_A067	Canon Powershot	Exterior	No	20.13630343	1.5	0.55	1.597654531	Nublado	315	No	2592 x 1944
IMG_A068	Canon Powershot	Exterior	No	56.30993247	0.4	0.60	0.721110255	Nublado	90	No	2592 x 1944
IMG_A069	Canon Powershot	Interior	Si	9.727578551	1.75	0.30	1.77552809	Luz artificial	0	No	2592 x 1944
IMG_A070	Canon Powershot	Interior	Si	12.87500156	1.75	0.40	1.795132307	Luz artificial	270	No	2592 x 1944
IMG_A071	Canon Powershot	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	45	No	2592 x 1944
IMG_A072	Canon Powershot	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	135	No	2592 x 1944
IMG_A073	Canon Powershot	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	225	No	2592 x 1944
IMG_A074	Canon Powershot	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	315	No	2592 x 1944
IMG_A075	Canon Powershot	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	45	No	2592 x 1944
IMG_A076	Canon Powershot	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	135	No	2592 x 1944
IMG_A077	Canon Powershot	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	225	No	2592 x 1944
IMG_A078	Canon Powershot	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	315	No	2592 x 1944
IMG_A079	Canon Powershot	Interior	Si	45.00	1.75	1.75	2.47	Luz artificial	180	No	2592 x 1944
IMG_A080	Canon Powershot	Interior	Si	45.00	1.75	1.75	2.47	Luz artificial	45	No	2592 x 1944
IMG_B001	Galaxy Ace	Exterior	No	10.39	3.00	0.55	3.05	Nublado	180	No	2560 x 1920
IMG_B002	Galaxy Ace	Exterior	No	5.42	5.8	0.55	5.83	Nublado	180	No	2560 x 1920
IMG_B003	Galaxy Ace	Exterior	No	3.66	8.6	0.55	8.62	Nublado	180	No	2560 x 1920
IMG_B004	Galaxy Ace	Exterior	No	20.97	3.00	1.15	3.21	Nublado	180	No	2560 x 1920
IMG_B005	Galaxy Ace	Exterior	No	11.21	5.8	1.15	5.91	Nublado	180	No	2560 x 1920
IMG_B006	Galaxy Ace	Exterior	No	7.62	8.6	1.15	8.68	Nublado	180	No	2560 x 1920
IMG_B007	Galaxy Ace	Exterior	No	46.40	3.00	3.15	4.35	Nublado	180	No	2560 x 1920
IMG_B008	Galaxy Ace	Exterior	No	28.51	5.8	3.15	6.60	Nublado	180	No	2560 x 1920
IMG_B009	Galaxy Ace	Exterior	No	20.12	8.6	3.15	9.16	Nublado	180	No	2560 x 1920
IMG_B010	Galaxy Ace	Exterior	No	59.78	3.00	5.15	5.96	Nublado	180	No	2560 x 1920
IMG_B011	Galaxy Ace	Exterior	No	41.60	5.8	5.15	7.76	Nublado	180	No	2560 x 1920

IMG_B012	Galaxy Ace	Exterior	No	30.91	8.6	5.15	10.02	Nublado	180	No	2560 x 1920
IMG_B013	Galaxy Ace	Exterior	No	14.57	3.00	0.78	3.10	Sombra y Sol	180	No	2560 x 1920
IMG_B014	Galaxy Ace	Exterior	No	7.66	5.8	0.78	5.85	Sombra y Sol	180	No	2560 x 1920
IMG_B015	Galaxy Ace	Exterior	No	5.18	8.6	0.78	8.64	Sombra y Sol	180	No	2560 x 1920
IMG_B016	Galaxy Ace	Exterior	No	48.99	3.00	3.45	4.57	Sombra y Sol	180	No	2560 x 1920
IMG_B017	Galaxy Ace	Exterior	No	30.75	5.8	3.45	6.75	Sombra y Sol	180	No	2560 x 1920
IMG_B018	Galaxy Ace	Exterior	No	21.86	8.6	3.45	9.27	Sombra y Sol	180	No	2560 x 1920
IMG_B019	Galaxy Ace	Exterior	No	61.17	3.00	5.45	6.22	Sombra y Sol	180	No	2560 x 1920
IMG_B020	Galaxy Ace	Exterior	No	43.22	5.8	5.45	7.96	Sombra y Sol	180	No	2560 x 1920
IMG_B021	Galaxy Ace	Exterior	No	32.36	8.6	5.45	10.18	Sombra y Sol	180	No	2560 x 1920
IMG_B022	Galaxy Ace	Exterior	No	10.39	3.00	0.55	3.05	Soleado	180	No	2560 x 1920
IMG_B023	Galaxy Ace	Exterior	No	5.42	5.8	0.55	5.83	Soleado	180	No	2560 x 1920
IMG_B024	Galaxy Ace	Exterior	No	3.66	8.6	0.55	8.62	Soleado	180	No	2560 x 1920
IMG_B025	Galaxy Ace	Exterior	No	20.97	3.00	1.15	3.21	Soleado	180	No	2560 x 1920
IMG_B026	Galaxy Ace	Exterior	No	11.21	5.8	1.15	5.91	Soleado	180	No	2560 x 1920
IMG_B027	Galaxy Ace	Exterior	No	7.62	8.6	1.15	8.68	Soleado	180	No	2560 x 1920
IMG_B028	Galaxy Ace	Exterior	No	46.40	3.00	3.15	4.35	Soleado	180	No	2560 x 1920
IMG_B029	Galaxy Ace	Exterior	No	28.51	5.8	3.15	6.60	Soleado	180	No	2560 x 1920
IMG_B030	Galaxy Ace	Exterior	No	20.12	8.6	3.15	9.16	Soleado	180	No	2560 x 1920
IMG_B031	Galaxy Ace	Exterior	No	59.78	3.00	5.15	5.96	Soleado	180	No	2560 x 1920
IMG_B032	Galaxy Ace	Exterior	No	41.60	5.8	5.15	7.76	Soleado	180	No	2560 x 1920
IMG_B033	Galaxy Ace	Exterior	No	30.91	8.6	5.15	10.02	Soleado	180	No	2560 x 1920
IMG_B034	Galaxy Ace	Exterior	No	46.40	3.00	3.15	4.35	Sombra y Sol	225	No	2560 x 1920
IMG_B035	Galaxy Ace	Exterior	No	28.51	5.8	3.15	6.60	Sombra y Sol	225	No	2560 x 1920
IMG_B036	Galaxy Ace	Exterior	No	20.12	8.6	3.15	9.16	Sombra y Sol	225	No	2560 x 1920

IMG_B037	Galaxy Ace	Exterior	No	46.40	3.00	3.15	4.35	Soleado	135	No	2560 x 1920
IMG_B038	Galaxy Ace	Exterior	No	28.51	5.8	3.15	6.60	Soleado	135	No	2560 x 1920
IMG_B039	Galaxy Ace	Exterior	No	20.12	8.6	3.15	9.16	Soleado	135	No	2560 x 1920
IMG_B040	Galaxy Ace	Exterior	No	39.24	3.00	2.45	3.87	Nublado	225	No	2560 x 1920
IMG_B041	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Nublado	225	No	2560 x 1920
IMG_B042	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Nublado	225	No	2560 x 1920
IMG_B043	Galaxy Ace	Exterior	No	39.24	3.00	2.45	3.87	Nublado	180	No	2560 x 1920
IMG_B044	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Nublado	180	No	2560 x 1920
IMG_B045	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Nublado	180	No	2560 x 1920
IMG_B046	Galaxy Ace	Exterior	No	39.24	3.00	2.45	3.87	Soleado	180	No	2560 x 1920
IMG_B047	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Soleado	180	No	2560 x 1920
IMG_B048	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Soleado	180	No	2560 x 1920
IMG_B049	Galaxy Ace	Exterior	No	39.24	3.00	2.45	3.87	Nublado	45	No	2560 x 1920
IMG_B050	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Nublado	45	No	2560 x 1920
IMG_B051	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Nublado	45	No	2560 x 1920
IMG_B052	Galaxy Ace	Exterior	No	39.24	3.00	2.45	3.87	Soleado	45	No	2560 x 1920
IMG_B053	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Soleado	45	No	2560 x 1920
IMG_B054	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Soleado	45	No	2560 x 1920
IMG_B055	Galaxy Ace	Exterior	No	39.24	3.00	2.45	3.87	Nublado	45	No	2560 x 1920
IMG_B056	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Nublado	45	No	2560 x 1920
IMG_B057	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Nublado	45	No	2560 x 1920
IMG_B058	Galaxy Ace	Exterior	No	39.24	3.00	2.45	3.87	Soleado	45	No	2560 x 1920
IMG_B059	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Soleado	45	No	2560 x 1920
IMG_B060	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Soleado	45	No	2560 x 1920
IMG_B061	Galaxy Ace	Exterior	No	22.90	5.8	2.45	6.30	Nublado	180	Si	2560 x 1920

IMG_B062	Galaxy Ace	Exterior	No	15.90	8.6	2.45	8.94	Nublado	180	Si	2560 x 1920
IMG_B063	Galaxy Ace	Exterior	No	12.34	1.6	0.35	1.64	Nublado	0	No	2560 x 1920
IMG_B064	Galaxy Ace	Exterior	No	16.39	1.7	0.5	1.77	Nublado	180	No	2560 x 1920
IMG_B065	Galaxy Ace	Exterior	No	8.75	0.65	0.1	0.66	Nublado	180	No	2560 x 1920
IMG_B066	Galaxy Ace	Exterior	No	50.71	0.45	0.55	0.71	Nublado	315	No	2560 x 1920
IMG_B067	Galaxy Ace	Exterior	No	20.14	1.5	0.55	1.60	Nublado	315	No	2560 x 1920
IMG_B068	Galaxy Ace	Exterior	No	56.31	0.4	0.6	0.72	Nublado	90	No	2560 x 1920
IMG_B069	Galaxy Ace	Interior	Si	9.73	1.75	0.3	1.78	Luz artificial	0	No	2560 x 1920
IMG_B070	Galaxy Ace	Interior	Si	12.88	1.75	0.4	1.80	Luz artificial	270	No	2560 x 1920
IMG_B071	Galaxy Ace	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	45	No	2560 x 1920
IMG_B072	Galaxy Ace	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	135	No	2560 x 1920
IMG_B073	Galaxy Ace	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	225	No	2560 x 1920
IMG_B074	Galaxy Ace	Interior	Si	3.27	1.75	0.1	1.75	Luz artificial	315	No	2560 x 1920
IMG_B075	Galaxy Ace	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	45	No	2560 x 1920
IMG_B076	Galaxy Ace	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	135	No	2560 x 1920
IMG_B077	Galaxy Ace	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	225	No	2560 x 1920
IMG_B078	Galaxy Ace	Interior	Si	34.44	1.75	1.2	2.12	Luz artificial	315	No	2560 x 1920
IMG_B079	Galaxy Ace	Interior	Si	45.00	1.75	1.75	2.47	Luz artificial	180	No	2560 x 1920
IMG_B080	Galaxy Ace	Interior	Si	45.00	1.75	1.75	2.47	Luz artificial	45	No	2560 x 1920

Tabla A.I – Tabla de Imágenes

ANEXO II – Tabla de Resultados (A.II)

A continuación se muestra la tabla de resultados que contiene la probabilidad relativa a cada Análisis Parcial Comparativo (APC):

CONFIGURACIÓN DEL PROCESADO			Datos Geométricos Reales Obtenidos Según Análisis Figura 6-1										Resultado		
Nombre	Binarización	Escalado	APC1	APC2	APC3	APC4	APC5	APC6	APC7	APC8	APC9	APC10	Tiempo	Total	Acierto
IMG_A001	1024x768	directo	1	1	1	1	1	1	1	1	0	0	3.00	80	1
IMG_A002	1024x768	directo	0	1	1	0	0	1	0	0	0	0	4.70	30	0
IMG_A003	1024x768	directo	0	1	0	0	0	0	1	0	0	0	3.48	20	0
IMG_A004	1024x768	directo	1	1	1	1	1	1	1	0	1	0	2.95	80	1
IMG_A005	1024x768	directo	0	0	0	0	0	0	1	0	0	0	4.80	10	0
IMG_A006	1024x768	directo	0	0	0	0	0	0	0	0	0	0	3.98	0	0
IMG_A009	1024x768	directo	0	1	0	0	0	0	0	0	0	0	4.27	10	0
IMG_A010	1024x768	directo	0	0	0	0	0	0	0	0	0	0	3.89	0	0
IMG_A011	1024x768	directo	0	1	0	0	0	0	0	0	0	0	4.23	10	0
IMG_A012	1024x768	directo	0	0	0	0	0	0	0	0	0	0	3.50	0	0
IMG_A013	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.97	100	1
IMG_A014	1024x768	directo	0	0	0	0	0	0	0	0	0	0	4.06	0	0
IMG_A015	1024x768	directo	0	0	1	0	0	0	1	0	0	0	3.13	20	0
IMG_A016	1024x768	directo	0	0	0	1	0	0	0	0	1	0	4.02	20	0
IMG_A017	1024x768	directo	1	1	0	0	1	0	0	0	0	0	3.94	30	0
IMG_A018	1024x768	directo	0	0	0	0	0	0	1	0	0	0	3.91	10	0
IMG_A019	1024x768	directo	1	1	0	1	0	0	0	0	1	0	4.31	40	0
IMG_A020	1024x768	directo	0	0	0	0	1	0	0	0	0	0	4.06	10	0

IMG_A022	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.84	100	1
IMG_A024	1024x768	directo	0	1	0	0	0	0	0	0	0	0	3.58	10	0
IMG_A025	1024x768	directo	1	1	1	1	1	1	0	0	1	0	2.89	70	1
IMG_A026	1024x768	directo	0	0	0	0	0	0	0	0	0	0	5.52	0	0
IMG_A027	1024x768	directo	0	0	0	0	0	0	1	0	0	0	4.48	10	0
IMG_A045	1024x768	directo	0	0	0	0	0	0	0	0	1	0	4.34	10	0
IMG_A050	1024x768	directo	0	0	0	0	0	0	0	0	0	0	3.83	0	0
IMG_A056	1024x768	directo	0	0	0	0	0	0	0	0	0	0	3.69	0	0
IMG_A061	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.80	100	1
IMG_A062	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.61	100	1
IMG_A063	1024x768	directo	1	1	1	1	1	1	1	1	1	1	3.11	100	1
IMG_A064	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.81	100	1
IMG_A066	1024x768	directo	1	1	0	0	0	1	1	1	0	0	3.17	50	1
IMG_A067	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.19	90	1
IMG_A068	1024x768	directo	0	0	0	0	0	0	0	0	0	0	3.75	0	0
IMG_A069	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.63	100	1
IMG_A070	1024x768	directo	1	1	1	0	0	1	0	1	1	1	2.64	70	1
IMG_A071	1024x768	directo	1	1	1	1	1	1	0	1	1	1	3.42	90	1
IMG_A072	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.67	90	1
IMG_A073	1024x768	directo	1	1	1	1	1	1	0	1	1	1	4.11	90	1
IMG_A074	1024x768	directo	1	1	1	0	1	1	1	1	1	1	4.05	90	1
IMG_A075	1024x768	directo	1	1	1	1	1	1	1	1	1	1	3.58	100	1
IMG_A076	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.48	90	1
IMG_A077	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.64	90	1
IMG_A078	1024x768	directo	1	1	1	0	1	1	1	1	1	1	4.02	90	1

IMG_A079	1024x768	directo	1	1	1	1	0	1	1	1	1	1	3.02	90	1
IMG_A080	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.55	90	1
IMG_A001	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	10.39	100	1
IMG_A002	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	14.77	100	1
IMG_A003	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	12.61	60	1
IMG_A004	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	12.30	100	1
IMG_A005	2592x1944	directo	1	1	1	1	1	1	0	0	1	0	14.64	70	1
IMG_A006	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	13.05	60	1
IMG_A007	2592x1944	directo	1	1	1	1	0	1	0	0	0	0	11.45	50	1
IMG_A008	2592x1944	directo	1	1	1	1	0	1	1	0	0	0	16.94	60	1
IMG_A009	2592x1944	directo	1	1	1	1	1	0	0	0	0	0	15.31	50	1
IMG_A010	2592x1944	directo	0	1	0	0	0	0	1	0	0	0	16.09	20	0
IMG_A011	2592x1944	directo	0	0	0	1	1	0	0	0	0	0	17.80	20	0
IMG_A012	2592x1944	directo	1	0	0	1	0	0	0	0	0	0	19.56	20	0
IMG_A013	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	10.73	100	1
IMG_A014	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	10.95	100	1
IMG_A015	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	10.09	100	1
IMG_A016	2592x1944	directo	0	0	0	1	0	1	0	0	0	0	12.30	20	0
IMG_A017	2592x1944	directo	1	1	1	1	0	1	0	0	1	0	10.38	60	1
IMG_A018	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	10.06	60	1
IMG_A019	2592x1944	directo	0	0	0	1	1	1	0	0	0	0	14.48	30	0
IMG_A020	2592x1944	directo	1	0	0	1	1	1	1	0	0	0	13.36	50	0
IMG_A021	2592x1944	directo	0	0	0	1	1	1	1	0	1	0	22.84	50	0
IMG_A022	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	10.31	100	1
IMG_A023	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	12.34	100	1

IMG_A024	2592x1944	directo	1	1	1	1	1	1	0	1	0	0	11.41	70	1
IMG_A025	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	10.03	100	1
IMG_A026	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	15.86	100	1
IMG_A027	2592x1944	directo	1	1	1	1	1	1	0	0	1	0	11.06	70	1
IMG_A028	2592x1944	directo	1	1	1	0	0	0	1	0	0	0	10.84	40	1
IMG_A029	2592x1944	directo	1	1	1	1	0	1	1	1	0	0	14.72	70	1
IMG_A030	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	16.88	60	1
IMG_A032	2592x1944	directo	0	0	0	0	1	0	0	0	0	0	13.33	10	0
IMG_A034	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	15.91	100	1
IMG_A035	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	16.42	60	1
IMG_A036	2592x1944	directo	1	1	1	1	1	0	0	0	0	0	19.03	50	1
IMG_A037	2592x1944	directo	0	0	0	0	0	0	1	0	0	0	14.70	10	0
IMG_A038	2592x1944	directo	1	1	1	1	1	1	1	0	1	0	20.44	80	1
IMG_A039	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	31.69	60	1
IMG_A040	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	18.31	100	1
IMG_A041	2592x1944	directo	0	1	1	1	0	1	1	0	0	0	15.28	50	0
IMG_A042	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	21.95	60	1
IMG_A043	2592x1944	directo	1	1	1	1	0	1	1	1	1	1	12.84	90	1
IMG_A044	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	17.00	100	1
IMG_A045	2592x1944	directo	1	1	1	1	1	1	1	0	0	0	14.56	70	1
IMG_A047	2592x1944	directo	0	0	0	0	1	0	0	0	0	0	9.56	10	1
IMG_A048	2592x1944	directo	1	1	1	1	1	1	1	0	0	0	9.61	70	1
IMG_A049	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	14.86	100	1
IMG_A050	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	15.67	100	1
IMG_A051	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	16.86	60	1

IMG_A053	2592x1944	directo	0	0	0	1	1	0	0	0	0	0	24.41	20	0
IMG_A054	2592x1944	directo	1	1	1	0	1	1	0	0	0	0	14.27	50	1
IMG_A055	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	14.50	100	1
IMG_A056	2592x1944	directo	0	0	0	1	0	0	1	0	0	0	14.92	20	0
IMG_A057	2592x1944	directo	1	1	1	1	1	1	0	0	0	0	16.50	60	1
IMG_A059	2592x1944	directo	1	1	1	1	0	1	1	1	1	1	12.66	90	1
IMG_A060	2592x1944	directo	1	1	1	1	1	1	1	0	0	0	13.83	70	1
IMG_A061	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	9.89	100	1
IMG_A062	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	8.41	100	1
IMG_A063	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	9.27	100	1
IMG_A064	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	7.92	100	1
IMG_A066	2592x1944	directo	0	0	0	1	0	0	0	0	0	0	11.88	10	0
IMG_A067	2592x1944	directo	1	1	1	0	1	1	1	1	1	1	11.02	90	1
IMG_A068	2592x1944	directo	0	0	0	0	0	1	0	0	0	0	10.13	10	0
IMG_A069	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	8.39	100	1
IMG_A070	2592x1944	directo	1	1	1	0	0	1	0	1	1	1	8.11	70	1
IMG_A071	2592x1944	directo	1	1	1	1	1	1	0	1	1	1	11.61	90	1
IMG_A072	2592x1944	directo	1	0	0	0	1	0	0	0	0	0	11.88	20	1
IMG_A073	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	11.98	100	1
IMG_A074	2592x1944	directo	0	0	0	1	0	0	0	0	1	0	12.45	20	0
IMG_A075	2592x1944	directo	1	1	1	1	1	1	1	1	1	1	12.63	100	1
IMG_A076	2592x1944	directo	1	1	1	0	1	1	1	1	1	1	12.22	90	1
IMG_A077	2592x1944	directo	1	1	1	0	1	1	1	1	1	1	13.52	90	1
IMG_A078	2592x1944	directo	1	1	0	0	0	0	0	0	0	0	13.80	20	0
IMG_A079	2592x1944	directo	0	0	0	1	1	0	0	0	0	0	9.78	20	0

IMG_A080	2592x1944	directo	0	0	0	1	1	0	0	0	0	0	11.28	20	0
IMG_A001	1024x768	otsu	1	1	1	1	1	0	0	0	0	0	6.61	50	1
IMG_A002	1024x768	otsu	0	1	0	0	0	0	0	0	0	0	4.78	10	0
IMG_A003	1024x768	otsu	0	0	0	0	0	0	0	1	0	0	5.30	10	0
IMG_A004	1024x768	otsu	0	0	0	1	0	0	0	0	0	0	4.72	10	0
IMG_A005	1024x768	otsu	0	0	0	0	0	1	0	0	0	0	4.34	10	0
IMG_A006	1024x768	otsu	0	1	0	0	0	0	0	0	0	0	4.72	10	0
IMG_A007	1024x768	otsu	0	1	0	1	0	1	0	0	0	0	4.69	30	0
IMG_A008	1024x768	otsu	0	1	0	1	0	1	0	0	1	0	5.31	40	0
IMG_A009	1024x768	otsu	0	0	0	1	0	1	0	0	0	0	4.31	20	0
IMG_A010	1024x768	otsu	0	1	0	1	0	1	0	0	1	0	5.13	40	0
IMG_A011	1024x768	otsu	0	0	0	1	0	1	1	0	0	0	4.75	30	0
IMG_A012	1024x768	otsu	0	0	0	1	0	1	1	0	0	0	5.84	30	0
IMG_A013	1024x768	otsu	1	1	1	1	1	1	0	0	1	0	3.83	70	1
IMG_A014	1024x768	otsu	0	0	0	1	1	1	1	0	0	0	5.05	40	0
IMG_A015	1024x768	otsu	0	1	0	1	1	1	1	0	0	0	5.45	50	0
IMG_A016	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	3.19	0	0
IMG_A017	1024x768	otsu	0	0	0	0	0	0	0	0	1	0	4.72	10	0
IMG_A018	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	3.97	0	0
IMG_A020	1024x768	otsu	0	1	0	0	1	0	1	0	0	0	3.55	30	0
IMG_A021	1024x768	otsu	0	0	1	0	0	0	1	0	0	0	3.59	20	0
IMG_A022	1024x768	otsu	1	1	1	1	1	1	0	0	0	0	3.41	60	1
IMG_A023	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	3.69	0	0
IMG_A025	1024x768	otsu	1	1	1	1	1	0	0	0	0	0	3.33	50	1
IMG_A028	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	3.41	0	0

IMG_A031	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	3.41	0	0
IMG_A034	1024x768	otsu	0	0	0	1	1	1	0	0	0	0	4.19	30	0
IMG_A035	1024x768	otsu	0	0	0	0	0	0	1	0	1	0	5.69	20	0
IMG_A036	1024x768	otsu	1	1	0	1	0	1	1	0	0	0	5.17	50	0
IMG_A037	1024x768	otsu	0	1	0	0	0	0	1	0	0	0	4.88	20	0
IMG_A038	1024x768	otsu	0	0	0	0	0	1	0	0	1	0	4.52	20	0
IMG_A039	1024x768	otsu	0	0	0	0	1	1	0	0	0	0	4.73	20	0
IMG_A040	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	4.30	10	0
IMG_A044	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.63	0	0
IMG_A045	1024x768	otsu	0	0	0	0	0	0	0	1	1	0	5.38	20	0
IMG_A055	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.16	0	0
IMG_A057	1024x768	otsu	1	1	0	0	1	0	0	0	0	0	4.14	30	0
IMG_A062	1024x768	otsu	0	0	0	0	1	0	1	1	0	0	3.94	30	0
IMG_A063	1024x768	otsu	1	1	1	1	1	1	1	1	1	1	4.39	100	1
IMG_A064	1024x768	otsu	1	1	1	1	1	1	1	1	1	1	2.89	100	1
IMG_A065	1024x768	otsu	1	1	1	0	1	1	1	1	1	1	5.08	90	1
IMG_A066	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	5.36	0	1
IMG_A067	1024x768	otsu	1	1	1	0	1	1	0	1	1	1	4.27	80	1
IMG_A068	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	3.80	10	1
IMG_A069	1024x768	otsu	1	0	0	0	0	0	1	0	0	0	4.03	20	0
IMG_A070	1024x768	otsu	1	1	1	0	0	1	0	0	1	1	3.59	60	1
IMG_A071	1024x768	otsu	0	0	0	0	0	0	1	1	1	0	3.53	30	0
IMG_A072	1024x768	otsu	0	1	0	1	0	0	1	0	0	0	4.75	30	0
IMG_A073	1024x768	otsu	0	0	0	0	0	1	0	0	0	0	3.86	10	0
IMG_A074	1024x768	otsu	0	0	0	0	0	1	0	0	0	0	3.98	10	0

IMG_A075	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	5.05	0	0
IMG_A076	1024x768	otsu	0	0	0	1	1	0	0	0	0	0	5.06	20	0
IMG_A077	1024x768	otsu	0	0	0	1	0	0	0	0	0	0	5.11	10	0
IMG_A078	1024x768	otsu	0	0	0	0	1	0	1	0	0	0	4.39	20	0
IMG_A079	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	3.94	0	1
IMG_A080	1024x768	otsu	0	0	0	1	1	1	0	0	0	0	4.67	30	0
IMG_A001	2592x1944	otsu	0	0	0	0	0	0	1	0	0	0	27.88	10	0
IMG_A002	2592x1944	otsu	0	1	0	1	0	1	0	0	0	0	17.41	30	0
IMG_A003	2592x1944	otsu	1	0	0	1	0	1	0	0	0	0	20.81	30	0
IMG_A004	2592x1944	otsu	1	1	1	0	0	0	1	0	0	0	14.91	40	0
IMG_A005	2592x1944	otsu	0	1	0	1	0	1	1	1	0	0	19.95	50	0
IMG_A006	2592x1944	otsu	0	0	0	1	1	0	1	0	0	0	19.16	30	0
IMG_A007	2592x1944	otsu	0	1	0	1	1	0	0	0	0	0	17.89	30	0
IMG_A008	2592x1944	otsu	0	0	0	1	0	1	0	0	0	0	19.42	20	0
IMG_A009	2592x1944	otsu	0	0	0	1	1	1	0	0	0	0	30.83	30	0
IMG_A010	2592x1944	otsu	0	0	0	1	0	0	1	0	0	0	17.91	20	0
IMG_A011	2592x1944	otsu	1	0	0	1	0	1	1	0	0	0	19.70	40	0
IMG_A012	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	20.86	10	0
IMG_A013	2592x1944	otsu	1	1	1	1	1	1	1	1	1	1	18.23	100	1
IMG_A014	2592x1944	otsu	0	0	0	1	0	1	1	0	0	0	23.61	30	0
IMG_A015	2592x1944	otsu	0	0	0	1	1	1	1	0	0	0	17.92	40	0
IMG_A016	2592x1944	otsu	1	0	0	0	0	1	0	0	0	0	15.33	20	0
IMG_A017	2592x1944	otsu	1	1	1	1	0	1	1	0	1	0	10.88	70	1
IMG_A018	2592x1944	otsu	0	0	0	1	1	1	1	0	0	0	18.83	40	0
IMG_A019	2592x1944	otsu	0	0	0	1	1	0	0	0	0	0	16.52	20	0

IMG_A020	2592x1944	otsu	0	1	1	1	0	1	1	0	0	0	16.83	50	0
IMG_A021	2592x1944	otsu	0	0	0	1	0	0	0	1	1	1	17.80	40	0
IMG_A022	2592x1944	otsu	1	1	1	1	1	1	1	1	1	1	12.42	100	1
IMG_A023	2592x1944	otsu	1	1	1	1	1	0	0	0	0	0	12.14	50	1
IMG_A024	2592x1944	otsu	0	0	0	1	0	0	1	0	0	0	15.31	20	0
IMG_A025	2592x1944	otsu	1	1	1	1	1	1	0	0	0	1	12.55	70	1
IMG_A026	2592x1944	otsu	0	0	0	0	0	0	0	1	1	0	14.95	20	0
IMG_A027	2592x1944	otsu	1	1	0	0	0	1	0	0	0	0	12.55	30	0
IMG_A028	2592x1944	otsu	0	0	0	0	0	0	0	0	0	0	13.72	0	0
IMG_A029	2592x1944	otsu	0	0	0	1	1	1	1	0	0	0	18.33	40	0
IMG_A030	2592x1944	otsu	0	0	0	1	0	1	1	1	1	0	17.39	50	0
IMG_A031	2592x1944	otsu	0	1	0	1	0	1	1	0	0	0	17.28	40	0
IMG_A032	2592x1944	otsu	0	1	0	1	0	1	0	0	0	0	17.25	30	0
IMG_A033	2592x1944	otsu	1	1	1	1	0	0	0	0	0	0	15.66	40	0
IMG_A034	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	16.13	10	0
IMG_A035	2592x1944	otsu	0	0	1	1	0	1	0	1	0	1	26.66	50	0
IMG_A036	2592x1944	otsu	0	0	0	1	0	1	1	0	0	0	34.22	30	0
IMG_A037	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	12.52	10	0
IMG_A038	2592x1944	otsu	0	0	0	0	0	0	0	0	1	0	12.06	10	0
IMG_A039	2592x1944	otsu	0	0	0	1	0	1	0	0	0	0	15.55	20	0
IMG_A040	2592x1944	otsu	0	0	0	1	0	0	1	0	1	0	16.83	30	0
IMG_A041	2592x1944	otsu	1	1	1	1	1	1	0	0	0	0	19.84	60	0
IMG_A042	2592x1944	otsu	0	1	1	1	0	1	0	0	1	0	20.34	50	0
IMG_A043	2592x1944	otsu	0	1	0	1	0	0	0	0	0	0	15.48	20	0
IMG_A044	2592x1944	otsu	0	1	0	0	0	0	1	0	0	0	20.72	20	0

IMG_A045	2592x1944	otsu	0	1	0	1	1	0	0	0	0	0	22.55	30	0
IMG_A046	2592x1944	otsu	1	1	0	0	0	0	0	0	0	0	19.80	20	0
IMG_A047	2592x1944	otsu	0	0	0	0	1	0	1	0	0	0	19.17	20	0
IMG_A048	2592x1944	otsu	0	0	0	1	0	0	0	0	1	0	15.58	20	0
IMG_A049	2592x1944	otsu	0	1	0	0	0	0	0	0	0	0	18.11	10	0
IMG_A050	2592x1944	otsu	0	0	0	1	1	0	1	0	0	0	18.31	30	0
IMG_A051	2592x1944	otsu	0	0	0	1	0	0	1	0	0	0	17.72	20	0
IMG_A052	2592x1944	otsu	0	0	1	1	0	0	0	0	0	0	17.55	20	0
IMG_A053	2592x1944	otsu	0	0	0	0	0	0	0	0	0	0	17.48	0	0
IMG_A054	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	14.95	10	0
IMG_A055	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	16.66	10	0
IMG_A056	2592x1944	otsu	0	0	0	1	1	0	0	0	0	0	13.52	20	0
IMG_A057	2592x1944	otsu	1	1	0	1	0	0	0	0	0	0	20.61	30	0
IMG_A058	2592x1944	otsu	0	0	0	1	1	0	0	1	0	1	18.23	40	0
IMG_A059	2592x1944	otsu	1	0	0	0	0	1	1	0	0	0	14.80	30	0
IMG_A060	2592x1944	otsu	0	0	0	1	0	0	0	0	1	0	16.97	20	0
IMG_A061	2592x1944	otsu	0	0	0	1	1	0	0	0	0	0	49.81	20	0
IMG_A062	2592x1944	otsu	0	0	0	1	1	1	0	0	0	0	18.81	30	0
IMG_A063	2592x1944	otsu	0	0	0	1	1	0	0	0	0	0	19.34	20	0
IMG_A064	2592x1944	otsu	1	1	1	1	1	1	1	1	1	1	11.38	100	1
IMG_A065	2592x1944	otsu	0	0	1	1	0	0	0	0	0	0	25.30	20	0
IMG_A066	2592x1944	otsu	1	1	0	1	0	1	1	0	0	0	39.63	50	0
IMG_A067	2592x1944	otsu	0	0	0	1	0	0	0	0	1	0	19.02	20	0
IMG_A068	2592x1944	otsu	0	0	0	0	1	0	0	1	1	1	28.88	40	0
IMG_A069	2592x1944	otsu	0	0	0	1	0	1	1	0	1	1	11.78	50	0

IMG_A070	2592x1944	otsu	0	0	0	1	0	1	1	0	0	0	16.81	30	0
IMG_A071	2592x1944	otsu	1	0	0	1	0	1	0	0	0	0	11.08	30	0
IMG_A072	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	14.67	10	0
IMG_A073	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	14.31	10	0
IMG_A074	2592x1944	otsu	0	0	0	1	1	0	0	1	0	0	15.64	30	0
IMG_A075	2592x1944	otsu	0	0	0	1	1	1	0	0	0	0	16.80	30	0
IMG_A076	2592x1944	otsu	0	1	0	1	0	1	1	0	0	0	16.34	40	0
IMG_A077	2592x1944	otsu	0	0	0	1	0	1	1	0	0	0	16.53	30	0
IMG_A078	2592x1944	otsu	0	0	0	1	1	0	0	0	0	0	15.20	20	0
IMG_A079	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	18.23	10	0
IMG_A080	2592x1944	otsu	0	0	0	1	0	0	0	0	0	0	21.31	10	0
IMG_B001	1024x768	directo	1	1	1	1	1	1	1	0	0	0	3.52	70	1
IMG_B003	1024x768	directo	0	0	0	0	0	0	0	0	0	0	5.14	0	0
IMG_B005	1024x768	directo	0	0	0	0	0	1	1	0	0	0	5.75	20	0
IMG_B006	1024x768	directo	0	0	0	0	0	1	1	0	0	0	4.16	20	0
IMG_B010	1024x768	directo	1	1	0	1	1	0	0	0	0	0	4.44	40	0
IMG_B011	1024x768	directo	0	0	0	0	0	0	0	0	0	0	4.11	0	0
IMG_B012	1024x768	directo	0	0	1	0	0	0	0	0	0	0	5.30	10	0
IMG_B013	1024x768	directo	1	1	1	1	1	1	1	0	1	0	2.92	80	1
IMG_B014	1024x768	directo	0	0	0	1	0	0	0	0	0	0	4.00	10	0
IMG_B015	1024x768	directo	0	1	0	0	0	1	1	0	0	0	3.92	30	0
IMG_B018	1024x768	directo	0	0	0	0	0	0	0	0	0	0	3.89	0	0
IMG_B019	1024x768	directo	0	0	0	1	0	0	0	0	0	0	4.75	10	0
IMG_B020	1024x768	directo	0	0	0	0	1	0	0	0	0	0	5.00	10	0
IMG_B022	1024x768	directo	1	1	1	1	1	1	1	1	1	1	4.97	100	1

IMG_B023	1024x768	directo	0	0	0	1	0	0	0	0	0	0	4.84	10	0
IMG_B025	1024x768	directo	1	1	1	1	1	1	1	0	0	0	5.05	70	1
IMG_B031	1024x768	directo	0	0	1	0	0	1	0	0	0	0	5.05	20	0
IMG_B032	1024x768	directo	0	0	0	0	0	0	0	0	0	0	4.25	0	0
IMG_B037	1024x768	directo	0	0	0	0	1	1	0	0	0	0	4.89	20	0
IMG_B039	1024x768	directo	0	0	0	0	0	1	1	0	0	0	6.92	20	0
IMG_B041	1024x768	directo	0	0	0	0	0	0	0	0	0	0	4.47	0	0
IMG_B044	1024x768	directo	0	0	0	0	0	0	0	1	1	0	4.69	20	0
IMG_B050	1024x768	directo	0	1	0	0	0	1	1	0	0	0	4.41	30	0
IMG_B056	1024x768	directo	0	0	0	1	1	0	0	0	1	1	4.48	40	0
IMG_B061	1024x768	directo	0	0	0	0	1	0	0	0	0	0	4.94	10	0
IMG_B062	1024x768	directo	0	0	0	0	1	0	0	0	0	0	4.59	10	0
IMG_B063	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.67	100	1
IMG_B064	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.73	100	1
IMG_B065	1024x768	directo	1	1	1	1	1	1	1	1	1	1	2.75	100	1
IMG_B066	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.69	90	1
IMG_B067	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.52	90	1
IMG_B068	1024x768	directo	1	1	1	0	0	1	1	1	1	1	2.97	80	1
IMG_B070	1024x768	directo	1	1	1	0	1	1	0	0	0	0	2.77	50	1
IMG_B071	1024x768	directo	1	1	1	1	1	1	0	1	0	0	3.30	70	1
IMG_B072	1024x768	directo	1	1	1	1	1	1	1	1	1	1	3.56	100	1
IMG_B073	1024x768	directo	1	1	1	1	1	1	1	1	1	1	3.86	100	1
IMG_B075	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.41	90	1
IMG_B076	1024x768	directo	1	1	1	0	1	1	0	0	0	0	3.42	50	1
IMG_B077	1024x768	directo	1	1	1	1	1	1	1	1	1	1	3.28	100	1

IMG_B078	1024x768	directo	1	1	1	0	1	1	1	1	1	1	3.55	90	1
IMG_B079	1024x768	directo	1	1	1	0	0	1	1	1	1	1	2.84	80	1
IMG_B080	1024x768	directo	0	1	0	1	0	0	0	0	0	0	3.66	20	1
IMG_B001	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	12.08	100	1
IMG_B002	2560x1920	directo	1	1	1	1	1	1	0	1	1	0	13.89	80	1
IMG_B003	2560x1920	directo	1	1	1	1	1	1	0	0	0	0	14.39	60	1
IMG_B004	2560x1920	directo	0	0	0	1	0	1	0	0	0	0	23.47	20	0
IMG_B005	2560x1920	directo	0	0	0	1	0	1	0	0	1	0	18.39	30	0
IMG_B006	2560x1920	directo	1	1	1	1	1	1	0	0	0	0	14.02	60	1
IMG_B007	2560x1920	directo	0	0	0	1	1	0	0	0	0	0	14.81	20	0
IMG_B008	2560x1920	directo	1	1	1	1	0	1	0	0	0	0	19.16	50	1
IMG_B009	2560x1920	directo	1	1	1	1	1	0	0	0	0	0	13.55	50	1
IMG_B010	2560x1920	directo	0	0	0	1	1	0	0	0	0	0	22.81	20	0
IMG_B011	2560x1920	directo	0	0	0	0	0	0	0	0	0	0	20.22	0	0
IMG_B012	2560x1920	directo	0	0	0	1	0	0	0	0	1	0	19.48	20	0
IMG_B013	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	9.66	100	1
IMG_B014	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	12.77	100	1
IMG_B015	2560x1920	directo	1	1	1	1	1	1	0	0	1	0	10.92	70	1
IMG_B016	2560x1920	directo	0	1	0	1	0	0	1	1	1	0	12.25	50	0
IMG_B017	2560x1920	directo	1	1	1	1	0	1	0	0	0	0	9.92	50	1
IMG_B018	2560x1920	directo	0	0	0	1	1	1	1	0	1	0	10.66	50	0
IMG_B019	2560x1920	directo	0	0	0	1	1	0	1	0	0	0	19.39	30	0
IMG_B020	2560x1920	directo	0	0	0	1	1	1	0	0	0	0	16.92	30	0
IMG_B021	2560x1920	directo	0	0	0	1	1	1	1	0	0	0	17.52	40	0
IMG_B022	2560x1920	directo	0	0	0	1	1	1	0	0	0	0	28.64	30	0

IMG_B023	2560x1920	directo	1	0	0	1	0	1	0	0	0	0	14.91	30	0
IMG_B024	2560x1920	directo	1	1	1	1	1	1	0	0	1	0	16.50	70	1
IMG_B025	2560x1920	directo	0	0	1	1	0	1	0	0	0	0	71.97	30	0
IMG_B026	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	13.00	100	1
IMG_B027	2560x1920	directo	0	0	0	1	1	0	0	0	0	0	20.17	20	0
IMG_B028	2560x1920	directo	0	0	0	1	0	0	1	0	0	0	65.17	20	0
IMG_B029	2560x1920	directo	0	0	0	1	1	0	0	1	1	1	19.03	50	0
IMG_B030	2560x1920	directo	0	1	0	1	0	0	1	0	0	0	38.20	30	0
IMG_B031	2560x1920	directo	0	1	1	1	0	1	0	0	1	0	24.22	50	0
IMG_B032	2560x1920	directo	0	0	1	1	1	0	1	0	0	0	27.09	40	0
IMG_B033	2560x1920	directo	0	1	0	1	0	0	0	0	0	0	41.63	20	0
IMG_B034	2560x1920	directo	0	0	0	1	0	1	1	0	0	0	17.05	30	1
IMG_B035	2560x1920	directo	1	1	1	1	1	1	0	0	0	0	26.47	60	1
IMG_B036	2560x1920	directo	1	1	1	1	1	0	0	0	0	0	16.31	50	1
IMG_B037	2560x1920	directo	0	0	1	1	0	1	0	1	1	1	55.39	60	0
IMG_B038	2560x1920	directo	0	0	0	1	0	1	0	0	0	0	18.48	20	0
IMG_B039	2560x1920	directo	0	0	0	1	0	0	0	1	0	0	25.36	20	0
IMG_B040	2560x1920	directo	0	0	0	1	0	0	0	0	1	0	14.80	20	0
IMG_B041	2560x1920	directo	0	0	0	1	1	0	0	0	0	0	17.20	20	0
IMG_B042	2560x1920	directo	1	1	1	1	1	1	0	0	0	0	22.36	60	1
IMG_B043	2560x1920	directo	0	0	0	0	0	1	1	0	0	0	18.13	20	0
IMG_B044	2560x1920	directo	1	1	1	1	0	1	1	1	1	1	14.64	90	1
IMG_B045	2560x1920	directo	1	1	1	1	1	1	0	0	0	0	13.75	60	1
IMG_B047	2560x1920	directo	0	0	0	1	1	1	0	0	0	0	24.30	30	0
IMG_B048	2560x1920	directo	1	1	1	1	1	0	0	0	0	0	14.77	50	1

IMG_B049	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	13.59	100	1
IMG_B050	2560x1920	directo	1	1	1	1	1	1	1	0	1	0	18.30	80	1
IMG_B051	2560x1920	directo	0	0	0	1	0	0	1	0	1	0	22.28	30	0
IMG_B053	2560x1920	directo	0	0	0	1	1	1	1	0	1	0	23.44	50	0
IMG_B054	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	12.97	100	1
IMG_B055	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	13.06	100	1
IMG_B056	2560x1920	directo	0	0	0	1	1	0	0	0	0	0	20.66	20	1
IMG_B057	2560x1920	directo	0	0	0	1	0	0	0	1	1	0	21.47	30	0
IMG_B059	2560x1920	directo	0	0	0	1	0	1	1	0	0	0	19.47	30	0
IMG_B060	2560x1920	directo	0	0	1	1	1	1	0	0	0	0	15.94	40	1
IMG_B061	2560x1920	directo	1	1	1	1	0	1	1	1	1	1	13.39	90	1
IMG_B062	2560x1920	directo	1	1	1	1	1	1	1	0	0	0	11.69	70	1
IMG_B063	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	7.59	100	1
IMG_B064	2560x1920	directo	1	1	1	0	1	1	1	1	1	1	7.86	90	1
IMG_B065	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	8.19	100	1
IMG_B066	2560x1920	directo	1	1	1	0	1	1	1	1	1	1	12.97	90	1
IMG_B067	2560x1920	directo	1	1	1	0	1	1	1	1	1	1	12.19	90	1
IMG_B068	2560x1920	directo	1	1	1	0	0	1	1	1	1	1	8.94	80	1
IMG_B069	2560x1920	directo	0	0	0	0	0	0	0	0	0	0	9.59	0	1
IMG_B070	2560x1920	directo	1	1	1	0	1	1	0	0	0	0	7.86	50	1
IMG_B071	2560x1920	directo	1	1	1	1	1	1	0	1	0	0	10.70	70	1
IMG_B072	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	11.84	100	1
IMG_B073	2560x1920	directo	1	1	1	1	1	1	1	1	1	1	12.55	100	1
IMG_B074	2560x1920	directo	0	1	0	0	0	0	0	0	0	0	9.11	10	0
IMG_B075	2560x1920	directo	1	1	1	0	1	1	0	1	1	1	11.63	80	1

IMG_B076	2560x1920	directo	1	1	1	0	1	1	0	1	0	0	11.02	60	1
IMG_B077	2560x1920	directo	1	1	1	0	1	1	1	1	1	1	11.00	90	1
IMG_B078	2560x1920	directo	1	1	1	0	1	1	1	1	1	1	10.86	90	1
IMG_B079	2560x1920	directo	1	1	1	0	0	1	1	1	1	1	8.17	80	1
IMG_B080	2560x1920	directo	0	1	0	1	0	0	0	0	0	0	10.91	20	1
IMG_B002	1024x768	otsu	0	0	0	0	0	1	0	0	0	0	4.64	10	0
IMG_B003	1024x768	otsu	0	0	0	1	0	0	1	0	0	0	5.39	20	0
IMG_B004	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.75	0	0
IMG_B005	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.73	0	0
IMG_B006	1024x768	otsu	0	0	0	1	0	0	0	0	0	0	4.55	10	0
IMG_B007	1024x768	otsu	0	1	0	0	0	1	0	0	0	0	4.30	20	0
IMG_B008	1024x768	otsu	0	1	0	0	0	0	0	0	0	0	5.17	10	0
IMG_B009	1024x768	otsu	0	1	0	1	0	1	0	0	0	0	5.50	30	0
IMG_B010	1024x768	otsu	0	0	0	1	0	0	0	0	0	0	4.78	10	0
IMG_B011	1024x768	otsu	0	1	0	1	1	1	0	0	0	0	4.36	40	0
IMG_B012	1024x768	otsu	0	1	0	0	1	0	0	0	0	0	5.06	20	0
IMG_B013	1024x768	otsu	1	1	1	1	1	1	0	0	0	0	4.72	60	1
IMG_B014	1024x768	otsu	0	0	0	1	1	1	0	0	0	0	5.13	30	0
IMG_B016	1024x768	otsu	0	1	0	0	0	0	0	0	0	0	3.80	10	0
IMG_B017	1024x768	otsu	0	0	0	1	1	0	0	0	0	0	4.78	20	0
IMG_B018	1024x768	otsu	0	1	0	0	0	0	0	0	0	0	5.17	10	0
IMG_B019	1024x768	otsu	0	0	0	0	0	1	1	0	0	0	3.63	20	0
IMG_B020	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	3.84	10	0
IMG_B021	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	4.77	10	0
IMG_B022	1024x768	otsu	1	1	1	1	1	1	0	0	0	0	3.31	60	1

IMG_B025	1024x768	otsu	1	1	1	1	1	0	0	0	0	0	3.16	50	1
IMG_B026	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.47	0	0
IMG_B028	1024x768	otsu	1	1	0	0	0	0	1	0	0	0	4.22	30	0
IMG_B029	1024x768	otsu	0	0	0	0	0	1	0	0	0	0	3.78	10	0
IMG_B030	1024x768	otsu	0	1	0	0	0	1	0	0	0	0	4.66	20	0
IMG_B033	1024x768	otsu	0	1	0	0	0	0	1	0	0	0	3.98	20	0
IMG_B034	1024x768	otsu	1	1	0	1	0	1	0	0	0	0	4.03	40	0
IMG_B035	1024x768	otsu	0	0	0	1	0	1	1	0	1	0	5.25	40	0
IMG_B036	1024x768	otsu	0	0	0	1	1	1	0	0	0	0	5.44	30	0
IMG_B037	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	3.69	0	0
IMG_B038	1024x768	otsu	0	1	0	0	0	1	1	1	0	1	3.64	50	0
IMG_B039	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	4.55	10	0
IMG_B040	1024x768	otsu	0	0	0	1	1	1	0	0	0	0	4.27	30	0
IMG_B041	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.66	0	0
IMG_B043	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	4.48	10	0
IMG_B044	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	4.48	10	0
IMG_B045	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.52	0	0
IMG_B046	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.70	0	0
IMG_B049	1024x768	otsu	0	0	0	0	0	1	0	0	0	0	4.34	10	0
IMG_B050	1024x768	otsu	0	0	0	0	0	0	0	0	1	0	4.84	10	0
IMG_B051	1024x768	otsu	0	0	0	0	1	0	1	0	0	0	4.05	20	0
IMG_B054	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.53	0	0
IMG_B055	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	4.97	0	0
IMG_B056	1024x768	otsu	0	0	0	0	1	1	0	0	0	0	3.88	20	0
IMG_B057	1024x768	otsu	0	0	0	0	0	0	0	0	0	0	5.16	0	0

IMG_B061	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	4.66	10	0
IMG_B063	1024x768	otsu	1	1	1	1	1	1	1	1	1	1	2.81	100	1
IMG_B065	1024x768	otsu	1	1	1	1	1	1	1	1	1	1	2.64	100	1
IMG_B066	1024x768	otsu	1	1	1	0	1	1	1	1	1	1	4.28	90	1
IMG_B068	1024x768	otsu	1	1	1	0	0	1	0	1	1	1	2.92	70	1
IMG_B069	1024x768	otsu	1	1	1	1	1	0	0	0	0	0	3.97	50	0
IMG_B070	1024x768	otsu	0	0	0	1	1	0	0	0	0	0	4.20	20	0
IMG_B071	1024x768	otsu	0	0	0	0	1	0	0	0	0	0	4.20	10	0
IMG_B072	1024x768	otsu	0	1	0	1	0	1	1	0	0	0	4.56	40	0
IMG_B073	1024x768	otsu	1	1	1	0	0	0	0	0	0	0	3.77	30	0
IMG_B074	1024x768	otsu	0	0	0	1	0	0	0	0	0	0	3.78	10	0
IMG_B075	1024x768	otsu	0	0	0	0	0	1	1	0	0	0	3.89	20	0
IMG_B076	1024x768	otsu	1	0	0	1	0	1	1	0	0	0	4.70	40	0
IMG_B077	1024x768	otsu	0	0	0	1	0	0	1	0	0	0	4.11	20	0
IMG_B078	1024x768	otsu	0	1	0	1	0	0	0	0	0	0	4.14	20	0
IMG_B079	1024x768	otsu	0	1	0	1	1	1	1	0	0	0	4.28	50	0
IMG_B080	1024x768	otsu	0	0	0	1	0	0	0	1	0	0	5.03	20	0
IMG_B001	2560x1920	otsu	0	0	0	0	1	0	1	0	0	0	22.63	20	0
IMG_B002	2560x1920	otsu	1	1	0	1	1	1	1	0	0	0	16.31	60	0
IMG_B003	2560x1920	otsu	1	1	0	1	0	1	1	0	0	0	21.11	50	0
IMG_B004	2560x1920	otsu	0	0	0	0	0	1	0	0	0	0	12.94	10	0
IMG_B005	2560x1920	otsu	0	0	0	1	0	0	0	0	0	0	21.16	10	0
IMG_B006	2560x1920	otsu	1	1	1	1	1	1	1	0	0	0	17.34	70	0
IMG_B007	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	14.81	20	0
IMG_B008	2560x1920	otsu	0	0	0	1	0	0	1	0	0	0	23.23	20	0

IMG_B009	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	32.03	20	0
IMG_B010	2560x1920	otsu	0	0	0	1	0	0	1	0	0	0	16.27	20	0
IMG_B011	2560x1920	otsu	0	0	0	1	0	1	1	0	0	0	22.75	30	0
IMG_B012	2560x1920	otsu	0	0	0	1	1	1	0	0	0	0	19.02	30	0
IMG_B013	2560x1920	otsu	1	1	1	1	1	1	0	0	0	1	15.44	70	1
IMG_B014	2560x1920	otsu	1	1	0	1	1	0	0	0	0	0	17.22	40	0
IMG_B015	2560x1920	otsu	1	1	1	1	1	0	0	0	0	0	20.59	50	1
IMG_B016	2560x1920	otsu	0	1	1	1	0	1	0	1	0	0	14.88	50	0
IMG_B017	2560x1920	otsu	1	1	1	1	0	1	0	0	0	0	16.08	50	1
IMG_B018	2560x1920	otsu	1	1	1	1	0	0	0	0	0	0	13.86	40	1
IMG_B019	2560x1920	otsu	1	1	0	1	0	0	0	0	0	0	14.05	30	0
IMG_B020	2560x1920	otsu	0	0	0	1	1	1	0	0	0	0	15.59	30	0
IMG_B021	2560x1920	otsu	1	1	0	1	0	1	1	1	1	1	17.14	80	0
IMG_B022	2560x1920	otsu	1	1	1	1	1	1	0	0	1	1	11.22	80	1
IMG_B023	2560x1920	otsu	1	1	1	1	1	0	0	0	0	0	11.50	50	1
IMG_B024	2560x1920	otsu	1	1	1	1	1	0	0	0	0	0	12.27	50	1
IMG_B025	2560x1920	otsu	1	1	1	1	1	1	0	0	0	1	11.14	70	1
IMG_B026	2560x1920	otsu	1	1	1	1	1	0	0	0	0	0	11.27	50	1
IMG_B027	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	11.92	20	1
IMG_B028	2560x1920	otsu	0	0	0	0	0	0	0	0	0	0	10.77	0	0
IMG_B029	2560x1920	otsu	0	0	0	1	0	1	1	0	0	0	16.27	30	0
IMG_B030	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	12.83	20	0
IMG_B031	2560x1920	otsu	0	0	0	1	0	1	1	0	0	0	14.66	30	0
IMG_B032	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	15.23	20	0
IMG_B033	2560x1920	otsu	0	1	0	1	0	0	0	0	0	0	12.91	20	0

IMG_B034	2560x1920	otsu	0	0	1	1	1	0	0	0	1	0	16.00	40	0
IMG_B035	2560x1920	otsu	0	0	0	1	0	0	1	0	0	0	23.33	20	0
IMG_B036	2560x1920	otsu	1	1	0	1	0	1	1	0	0	0	27.64	50	0
IMG_B037	2560x1920	otsu	0	0	1	1	0	0	0	0	0	0	17.72	20	0
IMG_B038	2560x1920	otsu	1	0	0	1	0	1	1	0	0	0	14.33	40	0
IMG_B039	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	16.42	20	0
IMG_B040	2560x1920	otsu	0	0	0	1	0	0	0	0	0	0	17.41	10	0
IMG_B041	2560x1920	otsu	0	0	0	1	0	0	1	0	0	0	17.61	20	0
IMG_B042	2560x1920	otsu	0	1	0	1	0	1	0	1	1	0	18.98	50	0
IMG_B043	2560x1920	otsu	0	1	0	1	0	1	0	0	0	0	19.52	30	0
IMG_B044	2560x1920	otsu	0	0	0	1	0	0	0	0	0	0	18.88	10	0
IMG_B045	2560x1920	otsu	0	0	0	1	0	0	1	0	0	0	17.50	20	0
IMG_B046	2560x1920	otsu	0	1	0	0	0	0	0	0	1	0	19.81	20	0
IMG_B047	2560x1920	otsu	0	0	0	1	0	0	0	0	0	0	15.22	10	0
IMG_B048	2560x1920	otsu	0	1	0	1	0	0	1	0	0	0	16.98	30	0
IMG_B049	2560x1920	otsu	0	1	0	0	0	0	0	0	0	0	20.08	10	0
IMG_B050	2560x1920	otsu	0	1	0	1	0	0	0	0	0	0	21.09	20	0
IMG_B051	2560x1920	otsu	0	1	0	1	1	0	1	0	0	0	15.02	40	0
IMG_B052	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	15.44	20	0
IMG_B054	2560x1920	otsu	0	1	0	1	0	0	0	0	0	0	20.39	20	0
IMG_B055	2560x1920	otsu	0	1	0	1	0	0	0	0	0	0	20.63	20	0
IMG_B056	2560x1920	otsu	0	1	1	1	0	0	0	0	0	0	19.58	30	0
IMG_B057	2560x1920	otsu	0	0	0	1	1	1	0	0	0	0	19.95	30	0
IMG_B058	2560x1920	otsu	0	1	0	1	0	0	0	0	0	0	16.17	20	0
IMG_B060	2560x1920	otsu	0	0	0	1	0	1	1	0	0	0	12.48	30	0

IMG_B061	2560x1920	otsu	0	0	0	1	0	0	0	0	0	0	16.28	10	0
IMG_B062	2560x1920	otsu	0	1	0	1	0	0	0	0	0	0	12.97	20	0
IMG_B063	2560x1920	otsu	1	1	1	1	1	1	1	1	1	1	7.97	100	1
IMG_B065	2560x1920	otsu	1	1	1	1	1	1	1	1	1	1	8.30	100	1
IMG_B066	2560x1920	otsu	1	1	1	0	1	1	0	1	1	1	16.70	80	1
IMG_B067	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	16.33	20	0
IMG_B068	2560x1920	otsu	1	1	1	0	0	1	0	1	1	1	10.83	70	1
IMG_B069	2560x1920	otsu	0	0	0	1	0	0	1	0	0	0	14.89	20	0
IMG_B070	2560x1920	otsu	0	1	0	1	0	0	1	0	0	0	12.28	30	0
IMG_B071	2560x1920	otsu	0	1	0	1	1	1	1	0	0	0	14.53	50	0
IMG_B072	2560x1920	otsu	0	0	1	1	0	0	1	0	0	0	18.67	30	0
IMG_B073	2560x1920	otsu	0	1	0	1	0	1	0	0	0	0	15.14	30	0
IMG_B074	2560x1920	otsu	0	0	0	1	1	0	0	0	0	0	14.22	20	0
IMG_B075	2560x1920	otsu	0	0	0	1	0	0	0	0	1	0	18.83	20	0
IMG_B076	2560x1920	otsu	0	0	0	1	1	1	1	0	0	0	15.36	40	0
IMG_B077	2560x1920	otsu	0	0	0	1	0	1	1	0	0	0	22.25	30	0
IMG_B078	2560x1920	otsu	0	1	0	1	0	1	1	1	1	0	22.64	60	0
IMG_B079	2560x1920	otsu	0	1	0	1	0	0	0	0	0	0	17.66	20	0
IMG_B080	2560x1920	otsu	0	0	0	1	1	0	0	0	1	0	15.72	30	0

Tabla A.II – Tabla de Resultados

ANEXO III – Tabla de Datos (A.III)

A continuación se muestra la tabla de datos que contiene la información más relevante que se ha utilizado en la elaboración de las estimaciones que quedan reflejadas a través de la figura 6-1:

CONFIGURACIÓN PROCESADO			DATOS GEOMÉTRICOS REALES OBTENIDOS SEGÚN ANÁLISIS FIGURA 6-1																		
Nombre	Escalado	Umbrali.	Posición Centros		Ángulos Elipses			Ratio Distancias Elipses			Ratio Área Círculos con Media Elipses				Ratio Área Círculos			Distancia Círculos		Ángulo Círculos	
			OY E1_E3	OX E1_E3	a1	a2	a3	I1/I2	I3/I2	I3/I1	C1 /Elps	C2 /Elps	C3 /Elps	C4 /Elps	C2 /C1	C3 /C1	C4 /C1	I4 /I2	I5 /I2	a5	a6
IMG_A001	1024x768	directo	0.63	0.81	45.90	136.71	0.73	70.92	69.50	98.00	56.08	3.32	4.22	2.11	5.91	7.53	3.76	102.19	1045.46	29.53	81.04
IMG_A002	1024x768	directo	233.61	126.14	22.56	111.53	22.23	70.47	72.24	102.52	28.22	1.68	1.68	1.68	5.95	5.95	5.95	9.02	6.80	226.38	297.17
IMG_A003	1024x768	directo	38.90	396.09	28.24	175.54	2.50	94.62	6.33	6.69	30.37	1.69	1.69	1.41	5.56	5.56	4.63	3.85	7.57	20.88	162.52
IMG_A004	1024x768	directo	0.09	0.38	43.60	137.26	0.11	69.24	67.86	98.00	63.16	3.38	3.01	3.76	5.36	4.76	5.95	925.97	105.71	79.38	306.74
IMG_A005	1024x768	directo	374.51	34.44	88.64	128.39	76.28	33.47	123.42	368.71	33.69	2.01	2.81	2.01	5.95	8.33	5.95	16.21	23.95	153.06	179.45
IMG_A006	1024x768	directo	449.87	60.20	324.71	93.54	70.01	73.11	36.35	49.72	65.22	2.04	5.71	3.67	3.13	8.75	5.63	10.86	15.57	123.93	178.78
IMG_A009	1024x768	directo	334.28	437.78	22.75	113.65	20.15	4.52	99.83	2206.51	63.28	3.39	3.13	3.13	5.35	4.94	4.94	0.34	0.19	47.56	358.59
IMG_A010	1024x768	directo	679.10	70.85	323.99	91.34	76.10	81.00	27.37	33.80	41.82	2.58	3.10	3.61	6.17	7.41	8.64	3.64	15.91	71.52	232.70
IMG_A011	1024x768	directo	151.99	97.93	56.60	130.84	13.97	98.00	59.88	61.11	35.42	0.00	3.22	2.68	0.00	9.09	7.58	11.06	7.31	295.47	318.15
IMG_A012	1024x768	directo	16.47	15.31	11.72	107.37	19.74	52.44	80.14	152.81	30.25	0.00	2.52	2.10	0.00	8.33	6.94	2046.15	1884.37	339.28	339.46
IMG_A013	1024x768	directo	0.08	0.44	44.39	136.54	0.10	70.13	68.70	97.96	52.51	2.34	3.34	3.68	4.46	6.37	7.01	103.03	102.97	80.63	99.31
IMG_A014	1024x768	directo	421.42	174.91	17.82	227.75	46.43	95.97	4.62	4.81	0.00	0.00	1.07	0.71	NaN	Inf	Inf	10.62	9.05	292.73	294.51
IMG_A015	1024x768	directo	182.72	65.85	4.59	114.08	30.45	60.90	61.56	101.08	70.05	1.88	3.76	4.77	2.68	5.36	6.80	14.31	13.98	241.43	280.49
IMG_A016	1024x768	directo	177.26	28.74	0.00	107.96	72.04	100.00	0.00	0.00	36.15	0.00	3.29	2.82	0.00	9.09	7.79	26.83	23.40	101.08	221.33
IMG_A017	1024x768	directo	53.30	69.03	51.64	146.24	11.21	89.26	38.47	43.10	52.27	2.12	2.12	4.25	4.07	4.07	8.13	19.80	17.21	195.59	328.83

IMG_A018	1024x768	directo	446.63	135.10	42.40	258.75	52.92	30.81	73.50	238.55	50.95	2.84	3.06	4.37	5.58	6.01	8.58	5.75	9.28	228.73	283.17
IMG_A019	1024x768	directo	49.95	65.53	61.60	138.66	10.65	97.72	52.37	53.59	40.86	1.99	2.66	2.99	4.88	6.50	7.32	19.31	18.89	36.90	170.67
IMG_A020	1024x768	directo	97.90	10.30	0.00	101.89	78.11	100.00	0.00	0.00	50.14	1.33	1.83	2.67	2.66	3.65	5.32	59.13	125.68	98.55	195.43
IMG_A022	1024x768	directo	0.73	0.21	43.39	135.18	0.88	69.85	69.41	99.38	46.50	1.88	3.00	3.38	4.03	6.45	7.26	107.31	107.29	79.31	98.33
IMG_A024	1024x768	directo	265.51	276.73	38.79	152.41	20.19	93.53	14.06	15.03	0.00	0.00	4.05	1.69	NaN	Inf	Inf	9.26	11.37	302.59	303.99
IMG_A025	1024x768	directo	0.43	0.09	43.55	137.27	0.55	68.34	68.70	100.53	47.64	3.00	2.15	3.00	6.31	4.50	6.31	1061.64	108.37	79.80	275.64
IMG_A026	1024x768	directo	14.92	188.18	2.27	0.00	2.27	0.00	100.00	Inf	35.36	0.00	2.68	2.68	0.00	7.58	7.58	13.41	13.45	48.78	141.59
IMG_A027	1024x768	directo	384.26	323.25	0.00	210.73	30.73	100.00	0.00	0.00	66.24	1.75	4.91	5.96	2.65	7.41	8.99	6.98	7.27	214.77	218.00
IMG_A045	1024x768	directo	775.52	156.99	0.00	247.96	67.96	100.00	0.00	0.00	37.14	2.06	2.06	2.06	5.56	5.56	5.56	10.49	1.79	118.05	257.36
IMG_A050	1024x768	directo	148.15	11.34	23.14	261.04	66.78	81.46	29.09	35.71	32.21	0.00	2.65	2.21	0.00	8.22	6.85	26.30	30.27	20.19	182.84
IMG_A056	1024x768	directo	226.43	75.76	326.44	131.15	47.12	88.96	11.40	12.82	64.85	1.57	4.09	5.98	2.43	6.31	9.22	8.05	18.43	275.09	294.87
IMG_A061	1024x768	directo	0.51	0.47	42.70	137.67	0.48	68.68	66.97	97.51	54.00	2.42	3.75	3.53	4.49	6.94	6.53	76.71	77.38	78.84	98.65
IMG_A062	1024x768	directo	0.13	0.43	43.98	136.70	0.13	69.68	68.50	98.31	54.04	2.14	4.50	3.86	3.97	8.33	7.14	80.54	80.61	79.66	98.75
IMG_A063	1024x768	directo	0.85	2.03	46.38	136.17	0.44	71.86	69.80	97.13	52.98	2.54	3.40	3.47	4.79	6.43	6.54	45.62	45.29	81.66	100.07
IMG_A064	1024x768	directo	1.34	1.98	45.86	136.71	0.65	70.98	69.39	97.77	47.01	1.61	2.88	2.71	3.43	6.13	5.76	42.45	42.60	81.44	99.98
IMG_A066	1024x768	directo	8.98	31.72	49.87	157.14	3.28	76.08	46.14	60.64	44.66	1.58	2.30	2.25	3.54	5.15	5.04	24.40	20.05	106.33	129.17
IMG_A067	1024x768	directo	1.95	5.68	47.24	140.67	1.15	72.17	65.04	90.12	45.90	1.54	2.71	2.53	3.35	5.91	5.51	48.25	46.93	85.32	104.98
IMG_A068	1024x768	directo	368.35	10.28	72.56	255.02	73.64	44.11	55.91	126.77	52.84	2.48	3.03	1.93	4.69	5.73	3.65	6.75	6.97	232.95	239.09
IMG_A069	1024x768	directo	0.62	0.86	45.23	135.36	0.33	71.40	69.85	97.83	42.97	1.36	2.07	2.14	3.16	4.82	4.98	46.74	46.45	80.68	99.08
IMG_A070	1024x768	directo	3.40	0.88	46.14	130.73	2.20	75.04	73.54	98.00	41.87	1.06	2.21	2.21	2.52	5.28	5.28	62.60	62.59	78.62	95.08
IMG_A071	1024x768	directo	0.05	0.33	45.17	135.27	0.04	70.97	70.33	99.10	38.90	0.97	2.08	1.94	2.49	5.34	4.98	63.01	63.70	79.42	97.63
IMG_A072	1024x768	directo	2.75	1.64	47.20	137.12	1.97	70.99	70.52	99.34	42.50	1.30	2.35	2.35	3.07	5.52	5.52	62.93	62.77	82.47	100.50
IMG_A073	1024x768	directo	1.66	0.25	44.70	134.51	1.12	71.72	69.92	97.49	40.18	1.07	2.49	1.90	2.65	6.19	4.72	60.07	60.80	79.86	97.95
IMG_A074	1024x768	directo	0.24	1.55	46.13	135.94	0.17	72.30	69.32	95.88	43.66	1.31	2.23	2.23	3.00	5.11	5.11	63.99	63.78	81.25	99.37
IMG_A075	1024x768	directo	2.36	4.46	39.91	130.06	2.09	66.92	74.13	110.78	46.78	1.70	3.03	2.84	3.64	6.48	6.07	75.58	78.53	72.14	90.14

IMG_A076	1024x768	directo	5.34	6.54	53.32	143.27	4.79	74.92	66.29	88.48	44.92	1.80	3.01	3.01	4.02	6.70	6.70	81.79	78.95	92.07	110.50
IMG_A077	1024x768	directo	3.97	4.38	40.22	128.58	3.52	69.16	74.24	107.33	46.77	1.33	2.85	2.85	2.85	6.10	6.10	80.32	82.63	71.94	89.59
IMG_A078	1024x768	directo	1.83	5.16	51.33	139.45	1.63	76.30	67.18	88.04	45.09	1.52	2.28	2.47	3.38	5.06	5.49	81.43	79.17	88.02	105.67
IMG_A079	1024x768	directo	0.92	0.18	37.72	140.87	0.88	64.07	63.58	99.24	43.40	2.18	2.73	2.73	5.03	6.29	6.29	66.19	64.85	76.29	99.88
IMG_A080	1024x768	directo	4.25	8.12	33.76	124.55	4.43	61.84	77.74	125.70	46.47	1.92	3.30	2.20	4.14	7.10	4.73	84.85	92.68	62.97	81.00
IMG_A001	2592x1944	directo	0.25	1.10	45.08	135.80	0.11	70.95	69.57	98.06	55.50	2.69	4.24	4.20	4.84	7.65	7.56	39.64	39.63	80.46	99.06
IMG_A002	2592x1944	directo	0.10	1.13	46.16	135.81	0.09	72.24	69.58	96.32	59.16	3.11	4.21	4.21	5.26	7.12	7.12	81.15	80.68	80.42	98.65
IMG_A003	2592x1944	directo	0.32	0.42	45.82	135.77	0.45	71.16	70.31	98.79	70.52	3.19	5.58	3.59	4.52	7.91	5.08	1301.60	1198.22	83.64	123.02
IMG_A004	2592x1944	directo	0.03	1.19	43.54	137.67	0.01	69.05	67.54	97.82	58.78	3.11	4.31	4.43	5.30	7.33	7.54	41.90	41.41	79.77	99.28
IMG_A005	2592x1944	directo	0.68	0.62	43.34	134.85	0.64	69.47	70.13	100.96	63.51	3.58	3.39	4.52	5.64	5.34	7.12	883.18	82.34	80.77	230.00
IMG_A006	2592x1944	directo	0.67	0.19	45.29	136.16	0.92	69.95	70.42	100.67	64.72	3.66	2.44	2.44	5.66	3.77	3.77	2181.48	2172.77	82.25	192.48
IMG_A007	2592x1944	directo	0.38	0.82	35.30	145.42	0.27	61.95	60.03	96.90	68.91	2.31	4.93	4.62	3.36	7.16	6.71	43.75	43.42	76.59	103.64
IMG_A008	2592x1944	directo	0.89	0.82	42.50	140.28	0.94	66.96	65.76	98.22	60.87	3.00	5.19	3.28	4.93	8.52	5.38	840.64	731.14	17.25	50.18
IMG_A009	2592x1944	directo	0.25	0.31	44.35	136.95	0.37	69.51	68.81	98.98	27.63	0.00	2.42	2.42	0.00	8.77	8.77	1141.44	698.10	240.11	240.90
IMG_A010	2592x1944	directo	37.34	87.62	69.97	150.79	9.19	99.49	34.67	34.85	52.57	1.42	2.49	2.02	2.71	4.74	3.84	30.27	27.04	174.69	179.00
IMG_A011	2592x1944	directo	19.89	53.07	0.00	169.38	10.62	100.00	0.00	0.00	39.73	2.05	3.42	1.71	5.17	8.62	4.31	34.85	176.48	158.00	169.81
IMG_A012	2592x1944	directo	32.05	37.54	5.84	163.33	10.55	73.70	27.85	37.79	32.48	1.14	1.82	1.36	3.50	5.59	4.20	41.79	67.42	139.82	170.01
IMG_A013	2592x1944	directo	0.30	0.61	44.14	136.29	0.14	69.87	68.98	98.73	53.46	2.50	3.95	3.69	4.67	7.39	6.91	40.57	40.50	79.57	98.26
IMG_A014	2592x1944	directo	0.06	0.39	44.73	135.92	0.05	70.46	69.51	98.66	54.94	2.25	4.13	3.94	4.10	7.51	7.17	78.96	79.47	79.17	97.52
IMG_A015	2592x1944	directo	0.47	0.07	44.64	134.93	0.63	71.04	70.01	98.56	52.06	2.13	3.84	3.41	4.10	7.38	6.56	118.74	119.29	78.91	97.35
IMG_A016	2592x1944	directo	271.28	15.50	303.81	163.43	39.97	43.81	62.27	142.13	37.80	0.00	3.00	3.00	0.00	7.94	7.94	12.82	11.44	317.84	330.30
IMG_A017	2592x1944	directo	0.13	0.25	40.20	140.49	0.14	65.41	64.86	99.16	47.09	2.35	2.02	3.03	5.00	4.29	6.43	1340.96	81.56	79.85	301.44
IMG_A018	2592x1944	directo	0.32	0.04	42.73	137.92	0.48	67.51	67.91	100.59	56.71	3.40	2.84	5.10	6.00	5.00	9.00	2927.50	1902.77	75.25	150.27
IMG_A019	2592x1944	directo	73.95	216.31	9.70	0.00	9.70	0.00	100.00	Inf	30.31	0.00	2.23	2.23	0.00	7.35	7.35	12.33	12.29	179.38	285.94
IMG_A020	2592x1944	directo	202.47	205.13	340.11	172.78	11.16	69.21	31.32	45.25	40.16	2.36	2.83	2.36	5.88	7.06	5.88	4.88	4.83	243.23	269.41

IMG_A021	2592x1944	directo	198.30	339.07	16.30	0.00	16.30	0.00	100.00	Inf	51.52	2.68	4.29	3.22	5.21	8.33	6.25	7.22	4.59	115.89	352.00
IMG_A022	2592x1944	directo	0.05	1.59	45.32	136.18	0.03	71.14	69.22	97.29	48.19	1.75	3.22	3.05	3.63	6.68	6.33	42.11	42.21	79.86	98.21
IMG_A023	2592x1944	directo	0.33	0.74	45.32	136.36	0.31	70.73	69.41	98.13	50.41	2.07	3.10	3.31	4.10	6.15	6.56	81.19	82.30	80.37	98.88
IMG_A024	2592x1944	directo	0.27	0.58	45.88	135.35	0.38	72.25	69.81	96.62	51.22	2.74	3.20	3.20	5.36	6.25	6.25	122.79	315.82	85.16	97.13
IMG_A025	2592x1944	directo	0.73	0.18	43.33	137.24	0.37	68.32	68.52	100.29	48.87	2.07	3.23	3.30	4.23	6.61	6.75	41.69	42.35	78.96	98.31
IMG_A026	2592x1944	directo	0.59	0.58	45.05	136.60	0.55	70.12	69.43	99.01	51.67	2.13	3.40	3.83	4.12	6.58	7.41	80.62	82.86	81.49	100.29
IMG_A027	2592x1944	directo	0.13	0.27	45.16	135.40	0.18	71.14	69.99	98.38	49.30	2.82	3.29	4.23	5.71	6.67	8.57	526.85	124.95	81.06	260.94
IMG_A028	2592x1944	directo	2.33	0.50	37.58	145.15	1.59	61.63	62.31	101.11	49.31	2.44	3.58	4.16	4.94	7.27	8.43	155.35	280.68	77.13	197.64
IMG_A029	2592x1944	directo	1.77	1.04	43.76	140.51	1.87	67.24	66.54	98.95	53.94	2.31	4.33	2.88	4.28	8.02	5.35	82.51	349.26	54.25	80.86
IMG_A030	2592x1944	directo	0.73	0.37	44.96	137.24	1.07	69.39	69.30	99.87	56.05	2.67	3.74	4.80	4.76	6.67	8.57	490.89	876.35	28.00	55.29
IMG_A032	2592x1944	directo	251.57	93.81	0.00	233.28	53.28	100.00	0.00	0.00	30.30	0.00	2.48	2.48	0.00	8.20	8.20	16.69	15.00	320.13	321.34
IMG_A034	2592x1944	directo	12.26	18.93	71.43	152.71	10.90	88.09	62.54	71.00	59.44	3.27	4.67	4.20	5.50	7.85	7.07	94.63	86.38	111.93	126.84
IMG_A035	2592x1944	directo	2.07	4.98	53.34	140.61	2.41	77.73	66.73	85.85	62.76	3.79	3.79	2.98	6.03	6.03	4.74	1091.24	1395.43	78.92	121.08
IMG_A036	2592x1944	directo	0.40	1.88	49.01	137.43	0.63	74.78	68.48	91.57	58.64	2.91	4.85	4.36	4.96	8.26	7.44	215.05	216.36	47.84	202.67
IMG_A037	2592x1944	directo	810.44	382.52	2.33	251.75	32.50	53.68	67.58	125.91	53.75	2.30	2.30	3.68	4.27	4.27	6.84	2.17	3.42	62.60	280.30
IMG_A038	2592x1944	directo	1.79	2.93	42.39	129.00	2.05	70.13	75.56	107.73	50.69	2.22	4.43	3.88	4.37	8.74	7.65	344.48	108.80	74.79	235.52
IMG_A039	2592x1944	directo	0.07	0.41	45.26	133.60	0.11	70.92	72.58	102.34	58.70	2.51	5.02	5.02	4.27	8.55	8.55	430.59	403.99	103.66	107.44
IMG_A040	2592x1944	directo	9.06	14.32	30.91	120.72	6.71	61.04	79.40	130.08	56.23	2.28	4.05	4.18	4.05	7.21	7.43	64.32	69.76	58.54	75.77
IMG_A041	2592x1944	directo	286.33	45.36	41.74	159.10	11.99	55.88	61.14	109.41	67.64	2.07	3.10	3.10	3.05	4.58	4.58	3.49	4.63	19.98	142.31
IMG_A042	2592x1944	directo	0.61	0.91	43.06	133.33	0.91	69.43	71.64	103.18	59.81	2.34	2.80	2.80	3.91	4.69	4.69	1705.60	1660.94	2.52	4.40
IMG_A043	2592x1944	directo	0.64	1.00	38.03	142.64	0.38	64.20	62.17	96.83	55.91	2.55	4.07	3.97	4.55	7.29	7.10	41.73	41.55	78.13	101.55
IMG_A044	2592x1944	directo	0.96	0.38	41.96	137.74	0.94	68.42	66.36	96.99	57.06	3.22	4.60	4.60	5.65	8.06	8.06	79.47	80.81	77.53	96.83
IMG_A045	2592x1944	directo	0.07	0.42	44.42	136.85	0.09	69.94	68.58	98.06	59.18	3.19	2.28	3.19	5.38	3.85	5.38	335.51	548.34	132.51	263.61
IMG_A047	2592x1944	directo	450.04	89.24	66.17	143.25	62.01	7.43	101.40	1364.96	45.67	1.31	3.67	3.94	2.87	8.05	8.62	23.99	0.19	7.17	216.94
IMG_A048	2592x1944	directo	0.90	0.58	45.08	137.92	1.29	69.29	68.75	99.22	62.25	2.94	2.94	2.94	4.72	4.72	4.72	1905.90	755.83	244.06	350.26

IMG_A049	2592x1944	directo	8.55	14.80	30.15	120.90	6.35	59.49	79.60	133.81	53.51	2.35	3.40	3.40	4.40	6.36	6.36	60.01	67.14	57.17	74.52
IMG_A050	2592x1944	directo	1.66	3.07	40.69	130.84	1.80	67.55	73.56	108.91	57.36	2.64	4.23	4.76	4.61	7.37	8.29	93.52	96.50	72.82	90.85
IMG_A051	2592x1944	directo	0.55	1.00	43.05	133.21	0.81	69.28	71.92	103.81	58.17	2.40	2.88	3.37	4.13	4.96	5.79	1663.22	1516.91	180.63	201.59
IMG_A053	2592x1944	directo	26.84	0.87	5.40	208.76	17.59	53.25	48.86	91.75	70.32	1.87	5.99	5.24	2.66	8.51	7.45	32.97	63.05	83.20	98.98
IMG_A054	2592x1944	directo	1.26	1.20	42.26	132.37	1.84	69.59	71.68	103.00	61.78	2.34	2.34	2.34	3.79	3.79	3.79	1327.31	1581.43	238.04	269.87
IMG_A055	2592x1944	directo	9.49	15.09	29.05	120.28	7.16	59.08	79.42	134.42	53.05	2.32	3.56	3.97	4.38	6.70	7.47	61.22	67.90	56.64	73.59
IMG_A056	2592x1944	directo	124.33	153.78	337.99	0.00	22.01	0.00	100.00	Inf	53.81	3.33	2.86	2.86	6.19	5.31	5.31	23.91	27.07	291.83	314.26
IMG_A057	2592x1944	directo	0.25	0.64	44.29	134.19	0.38	69.36	72.16	104.03	70.05	3.38	4.83	3.38	4.83	6.90	4.83	1556.75	1695.19	186.09	216.30
IMG_A059	2592x1944	directo	8.52	7.20	34.58	121.55	9.02	69.06	76.07	110.15	59.68	1.80	4.28	4.05	3.02	7.17	6.79	87.83	89.77	66.45	85.30
IMG_A060	2592x1944	directo	0.36	0.78	43.91	133.29	0.53	70.01	72.16	103.06	60.76	2.37	2.37	3.32	3.91	3.91	5.47	1684.21	1870.31	208.73	275.87
IMG_A061	2592x1944	directo	1.16	1.18	42.78	137.63	0.43	68.71	67.07	97.62	54.06	2.50	3.93	3.83	4.62	7.27	7.08	30.18	30.14	79.52	99.49
IMG_A062	2592x1944	directo	0.16	1.11	44.08	136.69	0.06	69.71	68.59	98.39	53.58	2.40	3.77	3.64	4.49	7.04	6.79	31.84	31.73	80.00	99.02
IMG_A063	2592x1944	directo	2.16	4.21	46.21	135.98	0.45	71.64	70.05	97.78	52.81	2.49	3.52	3.58	4.71	6.66	6.79	18.04	17.99	81.57	99.78
IMG_A064	2592x1944	directo	3.23	4.93	45.81	136.69	0.61	70.96	69.38	97.77	46.86	1.67	2.90	2.88	3.56	6.19	6.15	16.80	16.82	81.43	99.80
IMG_A066	2592x1944	directo	1337.56	634.42	8.36	140.30	32.49	87.93	16.87	19.19	28.59	0.00	2.20	2.64	0.00	7.69	9.23	2.02	2.06	70.27	186.96
IMG_A067	2592x1944	directo	4.16	14.19	47.05	140.58	0.97	72.17	64.91	89.94	47.03	1.66	2.83	2.79	3.53	6.03	5.94	19.12	18.60	85.32	104.77
IMG_A068	2592x1944	directo	79.70	32.16	358.11	147.08	24.34	74.08	28.94	39.06	29.23	0.00	2.56	2.56	0.00	8.77	8.77	24.30	22.85	163.07	169.42
IMG_A069	2592x1944	directo	1.99	1.84	45.15	135.20	0.42	71.42	69.93	97.92	42.93	1.42	2.17	2.10	3.30	5.06	4.90	18.52	18.41	80.53	98.83
IMG_A070	2592x1944	directo	9.16	2.51	46.08	130.53	2.34	75.15	73.63	97.98	41.76	1.18	2.25	2.18	2.82	5.40	5.22	24.95	24.90	78.57	95.08
IMG_A071	2592x1944	directo	1.09	1.09	45.42	135.52	0.31	70.84	70.45	99.45	40.68	1.07	2.18	2.18	2.62	5.35	5.35	25.21	25.49	79.99	98.17
IMG_A072	2592x1944	directo	301.48	157.36	343.01	180.70	7.17	56.13	45.06	80.29	0.00	0.00	0.03	0.03	NaN	Inf	Inf	2.08	2.12	317.12	318.54
IMG_A073	2592x1944	directo	4.36	0.66	44.73	134.46	1.17	71.81	69.93	97.38	40.99	1.26	2.54	1.88	3.08	6.20	4.59	23.69	23.92	79.36	97.61
IMG_A074	2592x1944	directo	584.07	374.05	35.79	234.43	36.46	3.70	96.49	2609.72	35.98	0.00	2.65	3.17	0.00	7.35	8.82	6.67	3.82	34.70	232.62
IMG_A075	2592x1944	directo	6.72	11.58	39.60	129.91	2.35	66.85	74.01	110.71	50.04	2.26	3.30	3.09	4.51	6.59	6.18	29.75	30.90	71.82	90.30
IMG_A076	2592x1944	directo	12.68	16.45	53.20	143.03	4.50	75.12	66.22	88.15	48.95	1.71	3.36	3.29	3.49	6.85	6.73	32.31	31.24	91.87	109.99

IMG_A077	2592x1944	directo	10.16	11.27	40.13	128.52	3.55	69.09	74.26	107.48	48.84	1.82	3.32	3.15	3.72	6.80	6.44	31.67	32.65	71.78	89.17
IMG_A078	2592x1944	directo	38.44	81.51	56.78	137.44	4.16	88.58	62.95	71.07	31.98	0.00	2.25	2.70	0.00	7.04	8.45	8.94	9.72	91.34	311.36
IMG_A079	2592x1944	directo	74.42	8.27	318.66	135.65	43.19	61.61	38.43	62.37	63.36	2.16	2.59	4.74	3.40	4.08	7.48	162.38	133.87	195.53	339.59
IMG_A080	2592x1944	directo	80.65	20.54	78.93	181.61	48.24	52.32	74.51	142.42	70.77	1.88	3.39	4.89	2.66	4.79	6.91	89.61	114.32	231.20	235.11
IMG_A001	1024x768	otsu	0.19	0.36	44.98	135.63	0.22	70.95	69.66	98.18	29.35	1.35	2.42	2.15	4.59	8.26	7.34	91.22	200.09	29.80	266.10
IMG_A002	1024x768	otsu	30.42	26.31	32.79	184.68	13.73	69.33	33.36	48.13	40.24	2.24	2.68	3.13	5.56	6.67	7.78	86.24	67.49	60.77	357.05
IMG_A003	1024x768	otsu	93.82	25.27	0.00	241.69	61.69	100.00	0.00	0.00	47.01	1.40	2.80	4.20	2.98	5.95	8.93	103.19	53.27	47.67	49.23
IMG_A004	1024x768	otsu	195.88	45.95	50.45	217.15	41.66	66.43	34.17	51.44	51.27	2.54	3.05	2.54	4.95	5.94	4.95	43.64	26.95	48.83	213.94
IMG_A005	1024x768	otsu	50.41	5.91	351.96	163.90	11.93	48.39	51.85	107.15	49.61	2.81	3.28	3.28	5.66	6.60	6.60	22.70	23.84	32.02	242.02
IMG_A006	1024x768	otsu	335.27	52.69	61.07	146.09	42.98	31.16	97.76	313.75	31.30	0.00	2.68	2.68	0.00	8.57	8.57	9.12	9.17	211.86	213.22
IMG_A007	1024x768	otsu	114.18	49.09	48.43	132.79	24.94	40.06	95.65	238.75	71.13	1.99	3.58	3.18	2.79	5.03	4.47	38.49	26.02	236.83	275.21
IMG_A008	1024x768	otsu	318.69	179.13	64.45	203.65	33.24	79.29	25.51	32.18	41.01	0.00	3.37	2.81	0.00	8.22	6.85	7.69	8.01	71.01	203.92
IMG_A009	1024x768	otsu	27.22	137.40	8.70	136.33	4.99	8.16	94.81	1161.58	28.46	0.00	2.85	2.85	0.00	10.00	10.00	16.36	16.93	155.13	333.35
IMG_A010	1024x768	otsu	44.03	51.49	25.80	146.67	14.60	22.62	86.49	382.32	44.90	2.55	2.55	2.55	5.68	5.68	5.68	43.33	30.26	42.46	56.42
IMG_A011	1024x768	otsu	64.11	11.18	18.47	144.96	12.63	64.25	47.42	73.81	42.32	1.71	1.71	1.71	4.03	4.03	4.03	17.82	16.05	210.17	239.00
IMG_A012	1024x768	otsu	145.73	77.74	0.00	223.14	43.14	100.00	0.00	0.00	62.12	1.97	4.33	3.93	3.16	6.96	6.33	24.07	21.88	160.02	204.84
IMG_A013	1024x768	otsu	0.65	0.24	43.57	135.07	0.77	69.91	69.70	99.69	60.12	3.70	2.16	4.62	6.15	3.59	7.69	968.88	102.98	79.87	258.10
IMG_A014	1024x768	otsu	203.44	118.25	0.00	220.70	40.70	100.00	0.00	0.00	51.08	3.03	3.46	2.60	5.93	6.78	5.08	14.39	13.56	237.83	239.83
IMG_A015	1024x768	otsu	280.00	41.56	36.25	206.35	32.49	38.17	62.19	162.94	70.23	3.26	4.65	3.26	4.64	6.62	4.64	9.94	9.34	8.99	221.81
IMG_A016	1024x768	otsu	629.88	36.67	15.02	101.41	60.91	97.19	30.44	31.31	47.29	1.34	2.01	2.01	2.83	4.25	4.25	12.83	11.24	280.52	283.54
IMG_A017	1024x768	otsu	352.73	54.07	0.00	252.95	72.95	100.00	0.00	0.00	61.00	2.07	2.44	2.20	3.40	4.00	3.60	34.41	36.69	46.30	357.51
IMG_A018	1024x768	otsu	700.11	27.54	304.51	100.70	71.16	66.90	35.08	52.43	0.00	0.00	1.37	2.34	NaN	Inf	Inf	16.59	16.80	278.28	280.12
IMG_A020	1024x768	otsu	310.36	31.56	64.46	126.61	29.87	64.21	112.32	174.91	27.54	1.71	1.22	1.46	6.19	4.42	5.31	4.01	3.43	230.93	245.74
IMG_A021	1024x768	otsu	333.85	172.52	83.27	165.94	36.74	73.17	78.14	106.79	65.46	1.71	3.41	4.05	2.61	5.21	6.19	13.79	12.94	196.87	199.60
IMG_A022	1024x768	otsu	1.25	0.39	42.98	134.50	1.53	70.12	69.46	99.05	61.26	1.88	4.08	3.46	3.08	6.67	5.64	0.53	501.31	145.40	298.81

IMG_A023	1024x768	otsu	20.76	103.83	0.00	174.29	5.71	100.00	0.00	0.00	29.20	0.00	2.65	2.65	0.00	9.09	9.09	135.84	15.02	72.55	89.80
IMG_A025	1024x768	otsu	0.19	0.04	42.75	136.88	0.24	68.36	68.22	99.78	38.75	2.08	2.42	2.42	5.36	6.25	6.25	1343.88	2082.33	55.48	320.85
IMG_A028	1024x768	otsu	386.65	43.27	56.56	258.28	66.61	47.16	54.66	115.91	60.31	1.95	3.11	4.28	3.23	5.16	7.10	6.98	29.22	87.66	173.27
IMG_A031	1024x768	otsu	484.48	186.23	312.15	152.83	46.32	7.57	92.88	1226.84	40.36	2.04	1.81	3.40	5.06	4.49	8.43	7.76	7.58	231.09	256.66
IMG_A034	1024x768	otsu	61.58	59.97	27.17	0.00	27.17	0.00	100.00	Inf	27.77	0.00	2.28	2.28	0.00	8.20	8.20	158.38	190.69	235.11	283.85
IMG_A035	1024x768	otsu	205.14	25.24	30.52	218.23	35.22	61.03	39.18	64.20	55.01	1.71	3.05	2.86	3.11	5.54	5.19	5.81	14.47	62.21	152.80
IMG_A036	1024x768	otsu	40.40	51.81	27.28	146.43	8.21	37.42	76.28	203.82	65.93	2.35	3.30	2.35	3.57	5.00	3.57	16.45	15.65	263.91	265.74
IMG_A037	1024x768	otsu	163.37	21.89	22.27	123.61	36.37	87.09	34.92	40.09	43.66	1.88	3.29	2.11	4.30	7.53	4.84	23.38	17.15	212.71	280.70
IMG_A038	1024x768	otsu	318.31	41.88	40.20	230.30	44.38	41.54	58.84	141.64	29.32	0.00	2.62	2.62	0.00	8.93	8.93	13.26	13.33	44.46	217.86
IMG_A039	1024x768	otsu	335.61	36.12	342.20	109.15	65.64	92.75	11.37	12.25	27.66	0.00	2.30	2.30	0.00	8.33	8.33	16.74	11.82	125.60	133.32
IMG_A040	1024x768	otsu	349.23	6.01	291.51	161.12	56.24	27.85	79.68	286.08	57.32	2.28	2.85	2.85	3.98	4.98	4.98	12.45	15.92	162.88	167.95
IMG_A044	1024x768	otsu	264.23	48.35	7.72	254.53	48.87	71.60	47.10	65.78	33.64	0.00	2.27	2.73	0.00	6.76	8.11	5.07	7.83	250.25	354.12
IMG_A045	1024x768	otsu	200.68	42.38	78.12	173.65	48.87	49.10	82.52	168.07	45.12	1.76	2.47	2.82	3.91	5.47	6.25	30.27	15.98	109.92	144.45
IMG_A055	1024x768	otsu	100.14	26.00	0.00	117.44	62.56	100.00	0.00	0.00	50.46	2.77	3.33	3.88	5.49	6.59	7.69	82.72	39.70	46.48	248.19
IMG_A057	1024x768	otsu	5.90	9.43	29.85	155.61	5.36	71.05	40.18	56.55	28.42	1.80	2.16	2.52	6.33	7.59	8.86	29.66	109.41	44.50	227.73
IMG_A062	1024x768	otsu	470.64	398.93	332.21	132.17	28.71	4.67	95.60	2045.89	68.75	2.21	2.94	2.57	3.21	4.28	3.74	7.40	5.53	81.47	96.59
IMG_A063	1024x768	otsu	0.03	1.63	46.05	135.67	0.02	71.97	69.90	97.13	56.01	3.04	3.99	4.05	5.42	7.12	7.23	45.56	45.41	81.65	99.84
IMG_A064	1024x768	otsu	1.29	2.01	45.87	136.69	0.62	71.03	69.38	97.68	52.08	2.40	3.60	3.77	4.61	6.92	7.23	42.35	42.41	81.49	99.85
IMG_A065	1024x768	otsu	6.14	5.88	46.46	137.25	1.27	70.96	69.50	97.94	53.09	2.44	3.90	3.94	4.60	7.34	7.42	18.42	18.43	82.49	100.76
IMG_A066	1024x768	otsu	30.94	32.85	334.78	0.00	25.22	0.00	100.00	Inf	51.43	2.66	3.38	3.38	5.17	6.56	6.56	114.15	94.00	31.46	54.06
IMG_A067	1024x768	otsu	1.53	5.54	46.95	140.53	0.90	72.13	64.90	89.98	59.33	3.52	4.67	4.67	5.94	7.87	7.87	48.41	47.11	85.36	104.89
IMG_A068	1024x768	otsu	293.89	231.12	32.45	0.00	32.45	0.00	100.00	Inf	58.59	1.82	3.64	5.51	3.10	6.21	9.40	3.31	0.44	98.09	178.03
IMG_A069	1024x768	otsu	45.52	54.62	2.48	166.13	4.32	42.05	58.95	140.20	30.81	1.62	1.62	2.27	5.26	5.26	7.37	4.93	8.43	113.61	179.04
IMG_A070	1024x768	otsu	4.07	1.31	45.68	130.26	2.62	75.00	73.60	98.13	60.28	3.68	5.02	4.78	6.11	8.32	7.93	62.92	62.66	78.31	94.87
IMG_A071	1024x768	otsu	76.39	164.95	276.29	179.84	12.43	95.31	21.39	22.44	68.18	2.92	2.92	2.44	4.29	4.29	3.57	4.95	9.03	65.53	100.92

IMG_A072	1024x768	otsu	96.96	23.62	22.40	123.58	45.77	94.63	18.84	19.91	67.60	2.96	3.45	2.96	4.38	5.11	4.38	6.68	9.97	58.33	139.94
IMG_A073	1024x768	otsu	322.57	380.96	88.40	185.78	22.58	91.99	29.15	31.69	35.56	0.00	2.87	2.87	0.00	8.06	8.06	6.01	6.67	194.42	196.29
IMG_A074	1024x768	otsu	210.71	185.44	343.42	152.47	18.89	21.22	79.09	372.76	66.44	3.58	3.58	2.56	5.38	5.38	3.85	10.05	9.83	169.57	185.45
IMG_A075	1024x768	otsu	572.30	25.57	59.08	235.91	57.78	40.81	59.22	145.11	53.13	2.45	2.04	2.45	4.62	3.85	4.62	6.42	6.29	204.83	205.19
IMG_A076	1024x768	otsu	506.03	211.01	50.17	0.00	50.17	0.00	100.00	Inf	61.86	2.32	2.32	5.67	3.75	3.75	9.17	10.95	7.39	169.59	202.54
IMG_A077	1024x768	otsu	453.39	8.66	40.03	267.34	82.89	92.54	10.56	11.41	30.15	1.46	1.30	1.13	4.84	4.30	3.76	8.12	13.16	41.49	247.25
IMG_A078	1024x768	otsu	319.10	166.81	1.56	241.88	25.80	47.25	67.79	143.48	55.97	2.42	2.01	2.01	4.32	3.60	3.60	4.68	4.83	22.19	22.76
IMG_A079	1024x768	otsu	28.20	13.01	312.70	0.00	47.30	0.00	100.00	Inf	55.39	3.09	3.80	3.09	5.58	6.87	5.58	162.26	594.16	212.30	350.95
IMG_A080	1024x768	otsu	23.50	17.67	0.00	146.37	33.63	100.00	0.00	0.00	58.63	3.42	3.88	4.11	5.84	6.61	7.00	483.85	613.97	258.52	285.00
IMG_A001	2592x1944	otsu	1255.36	34.67	68.19	226.58	59.68	40.19	61.53	153.09	55.06	1.47	2.58	2.03	2.68	4.68	3.68	0.64	0.78	156.96	178.77
IMG_A002	2592x1944	otsu	258.82	397.74	51.37	185.94	13.78	85.63	19.15	22.36	70.96	3.43	4.89	2.94	4.83	6.90	4.14	4.65	4.89	202.12	236.40
IMG_A003	2592x1944	otsu	93.31	182.42	6.07	192.46	7.61	24.23	75.89	313.22	32.96	0.91	1.36	1.21	2.75	4.13	3.67	7.84	6.17	248.47	305.28
IMG_A004	2592x1944	otsu	25.31	3.71	25.93	149.93	3.15	58.62	54.62	93.17	71.29	2.12	2.97	5.94	2.98	4.17	8.33	22.08	10.41	340.39	345.70
IMG_A005	2592x1944	otsu	242.54	16.47	42.74	163.97	16.39	51.90	62.70	120.81	53.18	3.01	3.51	3.51	5.66	6.60	6.60	6.13	5.78	10.91	118.09
IMG_A006	2592x1944	otsu	164.12	244.82	341.47	0.00	18.53	0.00	100.00	Inf	34.87	2.09	2.25	1.61	5.99	6.45	4.61	9.56	14.07	295.59	343.94
IMG_A007	2592x1944	otsu	733.37	764.96	22.56	179.74	20.09	11.14	89.64	804.86	29.10	1.40	1.68	1.40	4.81	5.77	4.81	3.89	3.85	195.12	203.89
IMG_A008	2592x1944	otsu	360.74	9.61	334.35	165.37	20.23	49.44	51.02	103.20	41.36	1.02	1.57	1.76	2.46	3.80	4.25	4.79	4.53	17.11	27.10
IMG_A009	2592x1944	otsu	79.34	204.91	10.96	0.00	10.96	0.00	100.00	Inf	34.48	0.94	2.07	1.32	2.73	6.01	3.83	8.12	20.78	76.62	286.66
IMG_A010	2592x1944	otsu	371.57	127.64	304.49	0.00	55.51	0.00	100.00	Inf	30.15	1.54	2.15	1.54	5.10	7.14	5.10	8.82	19.36	112.40	277.49
IMG_A011	2592x1944	otsu	71.81	77.40	355.89	163.23	8.51	34.99	65.56	187.36	56.36	1.94	2.43	2.91	3.45	4.31	5.17	12.14	7.76	125.38	305.51
IMG_A012	2592x1944	otsu	390.57	262.20	0.00	143.32	36.68	100.00	0.00	0.00	28.38	0.00	2.90	2.90	0.00	10.20	10.20	8.08	7.94	240.19	273.40
IMG_A013	2592x1944	otsu	0.88	0.28	43.95	135.89	0.41	69.96	69.13	98.81	58.36	3.18	4.55	4.70	5.45	7.80	8.05	40.50	40.36	79.49	98.08
IMG_A014	2592x1944	otsu	27.15	586.58	0.00	178.67	1.33	100.00	0.00	0.00	56.54	2.88	2.88	4.04	5.10	5.10	7.14	3.88	4.37	202.11	358.20
IMG_A015	2592x1944	otsu	370.00	891.65	0.00	168.28	11.72	100.00	0.00	0.00	69.11	2.56	3.07	3.07	3.70	4.44	4.44	2.51	3.07	134.13	332.78
IMG_A016	2592x1944	otsu	56.67	199.08	9.26	168.24	4.24	24.37	76.87	315.36	36.17	0.00	3.19	2.66	0.00	8.82	7.35	6.52	6.39	173.16	222.14

IMG_A017	2592x1944	otsu	0.33	0.30	40.22	140.82	0.35	65.21	64.76	99.30	55.99	2.69	2.10	4.19	4.81	3.74	7.49	664.58	81.78	80.95	246.33
IMG_A018	2592x1944	otsu	83.64	187.70	0.00	192.56	12.56	100.00	0.00	0.00	50.61	2.76	3.68	2.30	5.45	7.27	4.55	13.27	10.79	167.50	276.25
IMG_A019	2592x1944	otsu	76.70	241.27	9.03	0.00	9.03	0.00	100.00	Inf	65.04	2.93	5.27	2.93	4.50	8.11	4.50	10.59	19.84	26.82	40.17
IMG_A020	2592x1944	otsu	612.91	511.73	68.81	164.02	18.67	77.08	57.10	74.08	46.51	1.91	3.43	2.29	4.10	7.38	4.92	2.67	3.55	237.24	291.24
IMG_A021	2592x1944	otsu	639.49	169.41	297.92	0.00	62.08	0.00	100.00	Inf	47.35	2.09	2.09	2.93	4.42	4.42	6.19	6.42	5.68	102.57	115.40
IMG_A022	2592x1944	otsu	2.06	0.38	43.34	135.28	0.99	69.92	69.17	98.93	58.92	3.31	4.72	4.77	5.62	8.01	8.09	42.02	42.13	79.96	98.31
IMG_A023	2592x1944	otsu	0.77	0.33	43.96	134.70	0.72	70.31	70.21	99.85	56.24	2.92	3.40	2.92	5.19	6.05	5.19	907.40	703.70	48.50	52.83
IMG_A024	2592x1944	otsu	43.34	302.47	0.00	175.90	4.10	100.00	0.00	0.00	67.83	2.17	3.48	2.61	3.21	5.13	3.85	23.86	9.64	351.52	353.81
IMG_A025	2592x1944	otsu	0.42	0.06	42.78	136.94	0.21	68.36	68.19	99.75	60.91	3.44	5.16	5.22	5.65	8.48	8.57	41.39	41.98	79.00	98.24
IMG_A026	2592x1944	otsu	875.34	175.25	87.37	240.76	66.99	77.73	24.25	31.19	70.45	1.97	2.88	3.64	2.80	4.09	5.17	2.44	2.86	91.22	123.27
IMG_A027	2592x1944	otsu	14.09	35.77	28.73	167.00	3.05	65.11	41.53	63.78	32.90	1.39	2.78	2.78	4.23	8.45	8.45	31.59	3.23	247.02	357.31
IMG_A028	2592x1944	otsu	341.95	64.26	276.22	156.84	61.16	44.14	70.65	160.07	57.82	3.64	3.64	2.43	6.29	6.29	4.20	30.02	10.21	44.64	296.59
IMG_A029	2592x1944	otsu	8.00	466.64	0.49	0.00	0.49	0.00	100.00	Inf	42.23	2.33	3.20	2.04	5.52	7.59	4.83	5.19	9.33	217.65	313.46
IMG_A030	2592x1944	otsu	900.52	31.15	8.70	246.72	49.04	76.32	35.80	46.91	33.25	1.96	2.74	1.96	5.88	8.24	5.88	4.23	4.89	74.12	123.03
IMG_A031	2592x1944	otsu	1032.67	296.72	18.59	126.12	45.41	94.26	15.44	16.38	33.96	2.12	2.12	2.12	6.25	6.25	6.25	3.70	3.68	307.67	311.52
IMG_A032	2592x1944	otsu	672.09	224.50	45.25	173.07	35.48	21.48	85.38	397.46	28.39	0.00	2.78	2.78	0.00	9.80	9.80	4.18	4.06	2.98	169.67
IMG_A033	2592x1944	otsu	168.67	99.71	36.09	124.12	9.21	71.12	72.78	102.34	31.95	0.00	2.07	2.90	0.00	6.49	9.09	4.95	4.89	116.48	305.97
IMG_A034	2592x1944	otsu	318.41	153.62	33.81	242.37	38.90	18.57	83.30	448.56	42.86	2.16	2.16	3.03	5.05	5.05	7.07	18.58	10.31	122.17	250.04
IMG_A035	2592x1944	otsu	554.28	189.90	2.58	269.89	55.43	79.79	56.64	70.98	45.57	1.40	2.24	1.68	3.07	4.91	3.68	5.85	7.02	71.31	131.38
IMG_A036	2592x1944	otsu	401.76	346.93	30.07	0.00	30.07	0.00	100.00	Inf	66.41	3.25	3.25	3.72	4.90	4.90	5.59	5.54	6.75	42.29	256.17
IMG_A037	2592x1944	otsu	721.33	66.67	270.34	121.01	79.08	36.01	67.32	186.95	37.24	2.09	2.93	2.93	5.62	7.87	7.87	6.94	6.76	178.66	189.23
IMG_A038	2592x1944	otsu	435.53	1053.03	348.32	0.00	11.68	0.00	100.00	Inf	43.06	2.49	3.20	2.49	5.79	7.44	5.79	2.37	1.15	66.40	78.24
IMG_A039	2592x1944	otsu	411.55	51.32	294.37	115.94	64.45	75.01	25.00	33.32	51.46	2.57	2.57	4.12	5.00	5.00	8.00	5.64	10.15	114.36	141.94
IMG_A040	2592x1944	otsu	537.83	212.67	303.32	130.79	49.53	95.73	4.31	4.50	42.50	2.53	3.04	2.53	5.95	7.14	5.95	9.05	7.83	98.90	333.99
IMG_A041	2592x1944	otsu	5.99	48.11	41.41	149.15	0.63	68.58	54.83	79.96	42.43	2.25	3.49	3.74	5.29	8.24	8.82	9.28	9.98	117.99	246.02

IMG_A042	2592x1944	otsu	90.21	35.64	66.10	168.79	27.59	63.83	64.23	100.63	71.25	4.12	5.30	2.94	5.79	7.44	4.13	31.43	24.59	84.95	344.62
IMG_A043	2592x1944	otsu	311.35	79.20	18.99	134.99	26.50	79.34	35.32	44.52	35.85	1.24	1.49	1.49	3.47	4.17	4.17	1.87	1.77	315.80	316.28
IMG_A044	2592x1944	otsu	747.82	83.71	24.29	118.37	43.33	92.70	31.47	33.95	48.65	2.53	3.55	2.53	5.21	7.29	5.21	4.73	7.79	156.72	201.75
IMG_A045	2592x1944	otsu	308.49	104.26	34.83	127.11	44.54	98.36	14.54	14.78	35.48	2.25	3.14	3.14	6.33	8.86	8.86	11.73	11.45	281.52	313.37
IMG_A046	2592x1944	otsu	229.77	443.92	32.04	129.09	6.67	43.18	85.06	196.97	32.46	1.87	2.61	1.87	5.75	8.05	5.75	2.54	1.78	141.02	141.94
IMG_A047	2592x1944	otsu	369.34	672.14	334.76	237.08	11.37	24.18	93.85	388.12	38.14	1.71	2.20	2.20	4.49	5.77	5.77	5.11	5.25	169.30	171.12
IMG_A048	2592x1944	otsu	436.21	14.72	280.50	148.70	74.39	11.94	91.64	767.20	68.12	2.50	5.56	4.45	3.67	8.16	6.53	11.35	11.21	69.24	304.75
IMG_A049	2592x1944	otsu	387.26	59.02	64.59	206.22	46.20	50.84	55.03	108.23	34.79	1.20	1.52	1.68	3.46	4.38	4.84	8.04	6.84	170.84	184.62
IMG_A050	2592x1944	otsu	295.00	175.65	0.00	139.98	40.02	100.00	0.00	0.00	31.52	1.82	2.73	2.12	5.77	8.65	6.73	11.22	11.29	239.54	261.03
IMG_A051	2592x1944	otsu	58.05	183.56	338.86	177.93	5.73	81.36	19.51	23.97	28.85	1.74	2.09	2.09	6.02	7.23	7.23	8.70	8.63	151.79	193.25
IMG_A052	2592x1944	otsu	549.77	626.07	72.41	161.58	15.84	83.47	56.31	67.46	44.60	2.82	2.35	2.35	6.32	5.26	5.26	4.42	4.44	178.65	180.69
IMG_A053	2592x1944	otsu	1087.37	75.70	82.07	0.00	82.07	0.00	100.00	Inf	34.62	1.60	1.92	2.56	4.63	5.56	7.41	0.83	4.46	105.59	224.98
IMG_A054	2592x1944	otsu	854.97	76.41	292.29	109.47	68.55	29.90	70.13	234.54	45.71	2.86	2.86	2.86	6.25	6.25	6.25	5.23	5.60	118.36	129.56
IMG_A055	2592x1944	otsu	474.98	109.94	0.00	114.84	65.16	100.00	0.00	0.00	48.15	1.24	2.39	4.35	2.58	4.97	9.02	4.51	9.75	262.85	301.11
IMG_A056	2592x1944	otsu	80.77	380.68	0.00	173.94	6.06	100.00	0.00	0.00	71.89	2.76	4.61	5.07	3.85	6.41	7.05	6.75	6.67	336.24	345.08
IMG_A057	2592x1944	otsu	26.89	16.57	22.60	153.10	6.02	63.00	46.86	74.39	32.01	0.00	2.58	2.58	0.00	8.06	8.06	23.46	22.54	316.91	342.39
IMG_A058	2592x1944	otsu	912.94	98.55	89.58	256.61	77.75	91.37	8.83	9.67	53.07	2.25	2.25	3.60	4.24	4.24	6.78	4.99	5.36	91.93	97.00
IMG_A059	2592x1944	otsu	45.38	69.56	10.22	164.49	1.64	47.32	55.23	116.71	49.76	1.69	3.62	2.17	3.40	7.28	4.37	3.31	3.54	19.62	163.33
IMG_A060	2592x1944	otsu	675.28	175.72	62.51	0.00	62.51	0.00	100.00	Inf	52.08	1.41	2.12	1.89	2.71	4.07	3.62	8.78	4.46	84.17	253.60
IMG_A061	2592x1944	otsu	59.33	306.44	0.00	185.53	5.53	100.00	0.00	0.00	0.00	0.00	5.86	5.86	NaN	Inf	Inf	58.45	55.65	273.65	277.87
IMG_A062	2592x1944	otsu	1021.58	308.60	0.00	121.14	58.86	100.00	0.00	0.00	34.29	0.00	2.38	2.38	0.00	6.94	6.94	1.72	1.81	76.00	151.92
IMG_A063	2592x1944	otsu	2476.90	176.85	85.33	228.59	74.99	30.01	74.32	247.66	67.39	2.31	5.78	2.70	3.43	8.57	4.00	1.99	1.06	103.27	257.64
IMG_A064	2592x1944	otsu	3.21	5.04	45.87	136.67	0.61	71.03	69.40	97.70	52.16	2.34	3.74	3.80	4.49	7.18	7.28	16.78	16.76	81.50	99.88
IMG_A065	2592x1944	otsu	1365.81	683.73	17.73	95.90	37.92	84.36	73.71	87.38	49.35	2.77	3.33	2.77	5.62	6.74	5.62	2.29	2.71	229.64	245.83
IMG_A066	2592x1944	otsu	123.46	180.59	39.75	122.41	8.39	52.48	92.09	175.48	54.09	3.35	4.46	2.79	6.19	8.25	5.15	6.03	6.33	25.71	185.54

IMG_A067	2592x1944	otsu	539.80	59.42	0.00	257.58	77.58	100.00	0.00	0.00	46.58	2.77	2.77	2.77	5.95	5.95	5.95	12.30	7.45	104.21	252.29
IMG_A068	2592x1944	otsu	179.36	234.17	77.15	164.82	16.20	87.49	52.11	59.56	67.92	3.40	2.83	2.83	5.00	4.17	4.17	6.28	8.50	63.11	110.83
IMG_A069	2592x1944	otsu	1603.48	37.70	315.85	130.61	47.03	55.00	45.10	82.01	29.42	1.55	1.55	1.74	5.26	5.26	5.92	2.28	2.21	85.02	137.94
IMG_A070	2592x1944	otsu	644.02	684.46	0.00	154.81	25.19	100.00	0.00	0.00	64.47	2.11	4.63	2.53	3.27	7.19	3.92	3.35	3.36	121.22	267.26
IMG_A071	2592x1944	otsu	39.18	340.03	1.96	177.27	1.35	70.48	29.59	41.99	29.88	0.00	2.99	2.99	0.00	10.00	10.00	3.27	3.52	200.40	266.06
IMG_A072	2592x1944	otsu	2044.17	303.36	48.24	255.47	60.54	46.53	56.33	121.05	38.27	0.00	3.57	3.57	0.00	9.33	9.33	2.18	2.13	121.44	242.52
IMG_A073	2592x1944	otsu	86.10	2.16	82.03	262.52	82.37	70.11	29.89	42.62	48.84	2.79	2.33	2.79	5.71	4.76	5.71	370.20	309.34	182.84	188.65
IMG_A074	2592x1944	otsu	1246.51	41.29	77.21	248.58	74.89	26.93	73.30	272.20	34.91	0.95	1.99	1.73	2.72	5.69	4.95	6.31	6.37	83.90	84.44
IMG_A075	2592x1944	otsu	765.01	390.49	0.00	135.59	44.41	100.00	0.00	0.00	33.84	0.92	1.66	1.66	2.72	4.89	4.89	5.47	3.43	239.82	340.39
IMG_A076	2592x1944	otsu	290.19	109.58	46.74	130.30	20.80	93.00	48.64	52.31	58.05	2.93	2.44	2.44	5.04	4.20	4.20	4.78	4.70	325.54	327.38
IMG_A077	2592x1944	otsu	365.32	173.55	342.22	162.40	17.72	34.80	65.20	187.34	49.04	3.07	3.83	3.07	6.25	7.81	6.25	4.27	4.28	278.52	347.27
IMG_A078	2592x1944	otsu	71.85	127.35	344.25	0.00	15.75	0.00	100.00	Inf	37.84	0.00	3.29	3.29	0.00	8.70	8.70	16.51	19.85	169.76	175.77
IMG_A079	2592x1944	otsu	662.34	376.68	289.85	166.47	31.69	74.49	37.32	50.10	37.72	1.47	1.87	3.47	3.89	4.95	9.19	6.28	4.10	214.43	279.61
IMG_A080	2592x1944	otsu	904.29	29.87	0.00	266.22	86.22	100.00	0.00	0.00	48.68	2.65	3.70	2.65	5.43	7.61	5.43	5.54	4.50	149.16	285.77
IMG_B001	1024x768	directo	0.07	0.36	45.06	135.75	0.09	70.89	69.68	98.29	58.30	3.07	2.73	3.41	5.26	4.68	5.85	1138.52	1406.87	130.77	131.12
IMG_B003	1024x768	directo	658.04	68.31	84.27	191.46	68.40	28.63	87.73	306.41	28.57	0.00	1.98	2.38	0.00	6.94	8.33	6.68	8.08	70.59	113.73
IMG_B005	1024x768	directo	242.63	192.48	0.00	212.22	32.22	100.00	0.00	0.00	35.52	2.11	3.17	1.76	5.94	8.91	4.95	10.97	14.96	236.55	258.45
IMG_B006	1024x768	directo	16.87	39.97	0.00	168.09	11.91	100.00	0.00	0.00	61.06	2.47	3.88	4.94	4.05	6.36	8.09	110.15	32.13	341.34	354.17
IMG_B010	1024x768	directo	134.35	197.14	42.11	173.75	9.07	72.95	35.37	48.48	40.56	0.00	2.98	2.98	0.00	7.35	7.35	6.24	5.75	295.23	300.21
IMG_B011	1024x768	directo	455.16	161.78	2.33	130.62	44.56	93.02	10.69	11.49	30.58	0.00	2.88	2.88	0.00	9.43	9.43	12.79	12.55	307.22	324.76
IMG_B012	1024x768	directo	27.74	10.60	78.70	184.15	41.35	62.94	62.73	99.66	41.84	1.79	2.86	2.86	4.27	6.84	6.84	2505.80	3054.78	289.42	293.56
IMG_B013	1024x768	directo	0.79	0.33	43.12	135.03	1.00	69.66	69.48	99.74	57.39	1.86	5.22	3.73	3.25	9.09	6.49	1232.25	107.02	97.03	346.91
IMG_B014	1024x768	directo	445.08	215.75	45.89	0.00	45.89	0.00	100.00	Inf	44.19	0.00	3.49	2.91	0.00	7.89	6.58	8.61	8.97	127.74	250.35
IMG_B015	1024x768	directo	214.53	360.40	61.55	184.40	13.61	88.38	19.04	21.54	34.24	2.14	2.14	2.85	6.25	6.25	8.33	6.80	5.69	234.34	354.50
IMG_B018	1024x768	directo	438.71	29.73	56.70	235.20	55.82	59.05	40.96	69.36	28.67	1.28	1.54	1.28	4.46	5.36	4.46	12.92	2.39	17.83	335.57

IMG_B019	1024x768	directo	54.55	7.27	1.21	207.40	16.11	58.27	44.35	76.10	66.27	3.54	6.07	4.55	5.34	9.16	6.87	24.43	25.00	197.78	264.25
IMG_B020	1024x768	directo	170.62	112.92	322.93	0.00	37.07	0.00	100.00	Inf	60.50	2.10	4.20	5.46	3.47	6.94	9.03	18.36	18.41	86.37	220.90
IMG_B022	1024x768	directo	0.35	0.19	44.34	135.68	0.47	70.50	69.30	98.29	48.46	2.11	3.37	3.37	4.35	6.96	6.96	117.35	118.53	79.03	97.44
IMG_B023	1024x768	directo	315.52	141.17	0.00	131.82	48.18	100.00	0.00	0.00	43.15	1.70	2.38	2.04	3.94	5.51	4.72	15.33	19.99	39.18	41.28
IMG_B025	1024x768	directo	0.01	0.07	42.99	136.82	0.02	68.32	68.60	100.40	48.28	2.81	4.22	3.75	5.83	8.74	7.77	447.81	421.07	266.16	355.49
IMG_B031	1024x768	directo	76.12	52.13	81.59	166.13	29.32	79.45	68.76	86.55	30.90	1.88	2.26	2.64	6.10	7.32	8.54	40.11	29.27	73.12	266.85
IMG_B032	1024x768	directo	208.33	4.41	280.19	141.76	74.19	14.75	88.48	599.78	43.62	2.50	1.79	2.15	5.74	4.10	4.92	21.94	16.05	91.23	106.04
IMG_B037	1024x768	directo	413.91	93.56	0.00	245.67	65.67	100.00	0.00	0.00	33.98	1.95	3.13	2.73	5.75	9.20	8.05	9.99	10.43	111.89	149.66
IMG_B039	1024x768	directo	51.87	123.17	357.01	225.01	7.72	24.99	81.54	326.26	27.80	1.31	1.84	1.84	4.72	6.60	6.60	13.31	13.06	172.59	238.95
IMG_B041	1024x768	directo	587.87	287.38	316.01	176.14	42.63	3.69	97.15	2633.03	28.20	0.00	2.82	2.82	0.00	10.00	10.00	5.82	5.77	175.77	263.49
IMG_B044	1024x768	directo	150.75	15.64	85.34	205.38	70.76	29.08	82.22	282.76	71.56	2.22	4.44	3.11	3.11	6.21	4.35	31.71	38.59	60.59	107.94
IMG_B050	1024x768	directo	90.45	20.16	49.21	168.79	34.55	29.09	82.38	283.17	64.02	2.09	2.93	2.93	3.27	4.58	4.58	48.05	40.39	23.31	48.99
IMG_B056	1024x768	directo	87.78	203.47	347.83	0.00	12.17	0.00	100.00	Inf	45.29	2.24	3.59	2.24	4.95	7.92	4.95	12.50	13.04	59.81	94.41
IMG_B061	1024x768	directo	131.27	0.23	26.95	254.86	64.69	82.48	23.80	28.86	28.72	0.00	2.56	2.56	0.00	8.93	8.93	95.48	70.63	227.45	348.80
IMG_B062	1024x768	directo	503.88	49.79	48.21	258.02	65.07	58.35	45.07	77.23	40.38	2.40	3.37	3.37	5.95	8.33	8.33	10.53	10.59	170.11	171.40
IMG_B063	1024x768	directo	0.09	1.32	45.13	136.54	0.06	70.83	68.88	97.25	49.38	1.91	3.00	3.27	3.87	6.08	6.63	51.84	51.62	81.06	99.95
IMG_B064	1024x768	directo	2.09	2.57	45.81	139.42	1.49	70.01	67.13	95.89	51.79	2.28	3.80	3.55	4.40	7.33	6.85	60.65	60.82	83.16	102.36
IMG_B065	1024x768	directo	1.04	2.16	45.78	136.04	0.35	71.24	69.86	98.06	47.67	1.73	2.97	2.88	3.64	6.22	6.04	30.23	30.46	80.96	99.15
IMG_B066	1024x768	directo	2.01	15.58	52.50	138.60	0.67	78.80	67.17	85.24	43.73	1.50	2.46	2.12	3.43	5.62	4.85	30.78	29.59	87.91	105.34
IMG_B067	1024x768	directo	0.46	2.41	47.09	135.41	0.26	73.58	69.91	95.01	49.73	2.13	3.31	3.09	4.29	6.66	6.21	51.32	50.75	81.76	99.65
IMG_B068	1024x768	directo	12.43	7.00	55.87	135.21	5.77	78.07	78.58	100.66	48.36	1.80	3.36	3.19	3.73	6.94	6.59	49.06	49.58	88.34	103.40
IMG_B070	1024x768	directo	1.86	0.02	44.45	134.09	1.44	71.80	70.06	97.58	34.67	1.42	431.87	409.23	4.11	1245.66	1180.37	550.39	501.49	172.40	189.93
IMG_B071	1024x768	directo	0.61	1.00	46.14	135.84	0.46	71.54	70.25	98.20	44.60	2.41	2.27	1.70	5.40	5.08	3.81	66.28	291.76	29.46	99.54
IMG_B072	1024x768	directo	1.23	1.40	46.59	136.52	0.95	71.50	70.00	97.90	50.63	1.83	3.10	3.39	3.62	6.13	6.69	68.81	68.95	81.70	99.90

IMG_B073	1024x768	directo	0.42	0.05	44.59	135.17	0.32	70.60	70.10	99.29	49.79	1.42	2.98	2.98	2.85	5.98	5.98	68.07	68.47	79.73	98.20
IMG_B075	1024x768	directo	1.51	1.50	41.98	133.79	1.40	68.72	70.51	102.61	57.07	3.26	3.62	3.80	5.71	6.35	6.67	78.31	81.66	72.80	91.73
IMG_B076	1024x768	directo	2.87	4.25	49.72	140.14	2.48	73.42	67.36	91.75	38.06	1.76	97.01	3904.88	4.62	254.87	10259.49	74.92	19.38	11.42	182.49
IMG_B077	1024x768	directo	2.58	2.67	41.05	132.16	2.19	68.52	71.52	104.38	49.59	1.24	3.20	3.02	2.51	6.45	6.09	74.97	76.15	74.54	93.22
IMG_B078	1024x768	directo	1.07	4.09	49.23	138.74	0.92	74.68	67.14	89.90	49.47	1.58	2.82	2.64	3.20	5.69	5.34	75.99	74.11	85.81	104.13
IMG_B079	1024x768	directo	0.05	0.46	39.31	141.54	0.05	64.89	63.58	97.97	53.14	2.29	3.14	2.86	4.30	5.91	5.38	70.69	71.00	77.20	100.18
IMG_B080	1024x768	directo	232.51	155.36	34.32	143.43	31.68	96.68	9.03	9.34	52.42	2.85	1.99	2.28	5.43	3.80	4.35	0.01	7.33	256.05	320.05
IMG_B001	2560x1920	directo	0.08	0.69	44.81	135.84	0.04	70.54	69.62	98.70	58.06	3.05	4.47	4.52	5.26	7.70	7.79	44.16	43.84	80.53	99.09
IMG_B002	2560x1920	directo	1.12	0.71	45.64	136.91	1.11	70.14	69.73	99.42	60.63	1.47	4.63	4.84	2.43	7.64	7.99	88.18	87.13	81.88	100.18
IMG_B003	2560x1920	directo	1.00	0.47	47.05	135.78	1.48	71.44	71.58	100.20	66.35	3.85	3.85	5.29	5.80	5.80	7.97	1456.87	1289.03	78.25	125.72
IMG_B004	2560x1920	directo	189.41	122.08	12.68	116.73	19.95	55.59	70.72	127.22	71.93	3.95	2.63	5.26	5.49	3.66	7.32	4.75	9.27	65.71	194.99
IMG_B005	2560x1920	directo	843.26	297.88	298.22	174.56	31.08	61.33	52.00	84.77	31.46	0.00	2.82	2.35	0.00	8.96	7.46	3.15	3.10	122.18	302.40
IMG_B006	2560x1920	directo	0.52	0.08	45.32	135.68	0.77	70.15	70.82	100.95	68.50	3.31	3.31	2.83	4.83	4.83	4.14	2477.70	2234.73	195.16	207.66
IMG_B007	2560x1920	directo	1151.38	134.35	281.32	113.64	70.12	69.76	30.73	44.05	67.68	2.06	4.95	5.78	3.05	7.32	8.54	4.16	4.17	105.66	148.67
IMG_B008	2560x1920	directo	0.57	0.62	42.01	138.66	0.65	68.22	65.64	96.22	61.89	3.79	4.74	2.53	6.12	7.65	4.08	572.31	257.86	57.70	98.78
IMG_B009	2560x1920	directo	0.26	0.34	44.01	137.48	0.40	69.10	68.23	98.75	46.46	2.80	3.36	3.36	6.02	7.23	7.23	160.66	484.39	164.39	315.58
IMG_B010	2560x1920	directo	98.08	486.71	5.75	0.00	5.75	0.00	100.00	Inf	29.95	0.00	2.50	2.50	0.00	8.33	8.33	5.16	5.09	246.79	253.87
IMG_B011	2560x1920	directo	1478.89	30.76	41.94	263.91	80.89	94.00	7.88	8.38	54.63	2.44	3.97	4.27	4.47	7.26	7.82	3.39	3.38	104.26	121.72
IMG_B012	2560x1920	directo	1056.80	294.87	66.35	235.13	56.37	89.06	11.12	12.49	48.12	2.93	2.51	2.09	6.09	5.22	4.35	5.24	1.51	99.10	234.30
IMG_B013	2560x1920	directo	1.58	0.96	43.12	135.08	0.80	69.40	69.65	100.36	55.36	2.62	3.92	4.04	4.73	7.09	7.30	43.41	43.75	78.55	97.44
IMG_B014	2560x1920	directo	0.50	1.17	45.86	135.91	0.48	72.35	68.97	95.33	57.66	2.87	4.31	4.31	4.98	7.47	7.47	85.14	85.43	79.91	98.95
IMG_B015	2560x1920	directo	0.06	0.03	44.90	135.28	0.09	70.48	70.48	100.01	59.71	2.94	5.38	3.92	4.92	9.02	6.56	1532.35	134.43	79.85	331.39
IMG_B016	2560x1920	directo	100.86	193.12	63.31	178.40	11.40	86.90	24.84	28.59	53.01	2.26	2.63	3.01	4.26	4.96	5.67	6.70	9.75	62.06	94.46
IMG_B017	2560x1920	directo	0.08	0.49	40.44	140.78	0.09	65.81	64.40	97.86	45.30	2.66	1.90	2.66	5.88	4.20	5.88	2099.77	2200.05	10.48	16.97
IMG_B018	2560x1920	directo	334.45	237.06	0.00	144.80	35.20	100.00	0.00	0.00	42.76	1.75	2.80	2.80	4.10	6.56	6.56	8.84	9.01	106.50	166.83

IMG_B019	2560x1920	directo	384.50	219.50	0.00	221.21	41.21	100.00	0.00	0.00	55.36	2.68	2.68	4.91	4.84	4.84	8.87	8.83	8.66	78.93	125.44
IMG_B020	2560x1920	directo	340.46	273.97	31.85	0.00	31.85	0.00	100.00	Inf	29.64	0.00	2.56	2.56	0.00	8.62	8.62	7.84	7.53	185.83	205.62
IMG_B021	2560x1920	directo	76.76	318.97	353.14	0.00	6.86	0.00	100.00	Inf	49.81	2.90	4.05	2.90	5.81	8.14	5.81	7.91	6.81	241.77	291.08
IMG_B022	2560x1920	directo	252.92	142.76	73.22	147.35	22.48	80.50	85.29	105.95	33.89	0.00	2.78	2.78	0.00	8.20	8.20	7.72	7.71	165.67	232.51
IMG_B023	2560x1920	directo	58.80	22.18	356.73	174.68	4.24	47.28	52.74	111.55	66.40	3.56	4.15	2.96	5.36	6.25	4.46	6.52	6.73	210.96	254.30
IMG_B024	2560x1920	directo	0.35	0.20	44.33	135.72	0.52	70.55	69.18	98.05	55.17	3.45	3.94	2.46	6.25	7.14	4.46	340.24	131.80	79.48	96.71
IMG_B025	2560x1920	directo	294.68	183.21	78.79	168.07	29.52	75.78	66.21	87.37	32.35	0.00	2.94	2.94	0.00	9.09	9.09	8.55	8.55	176.56	272.19
IMG_B026	2560x1920	directo	0.58	0.46	45.62	135.86	0.58	70.77	70.36	99.42	56.39	2.01	4.23	4.46	3.56	7.51	7.91	87.34	86.45	81.81	100.39
IMG_B027	2560x1920	directo	652.95	58.24	79.89	0.00	79.89	0.00	100.00	Inf	54.92	2.95	4.13	3.54	5.38	7.53	6.45	7.56	7.58	98.50	244.91
IMG_B028	2560x1920	directo	322.40	132.45	84.32	159.03	42.44	69.21	92.70	133.95	71.61	2.90	4.84	6.29	4.05	6.76	8.78	10.73	10.68	302.00	325.86
IMG_B029	2560x1920	directo	431.03	39.68	61.75	253.83	66.65	40.81	59.73	146.38	66.05	3.84	2.98	4.69	5.81	4.52	7.10	13.67	10.93	110.39	122.35
IMG_B030	2560x1920	directo	365.90	495.94	19.45	90.69	19.93	67.00	98.84	147.53	46.32	2.76	2.76	3.31	5.95	5.95	7.14	6.94	4.75	9.71	161.38
IMG_B031	2560x1920	directo	184.26	128.55	68.88	145.23	16.06	81.99	79.79	97.31	31.23	0.00	2.79	2.79	0.00	8.93	8.93	7.94	7.48	66.43	255.44
IMG_B032	2560x1920	directo	390.38	120.99	72.14	175.51	37.29	58.74	68.48	116.59	36.68	2.08	3.11	1.73	5.66	8.49	4.72	7.92	9.24	241.29	273.39
IMG_B033	2560x1920	directo	350.58	155.79	40.06	146.37	14.86	44.36	78.03	175.90	39.11	0.00	3.09	3.60	0.00	7.89	9.21	3.73	3.73	41.10	137.35
IMG_B034	2560x1920	directo	12.42	18.19	72.86	152.95	11.88	88.78	63.78	71.85	59.48	3.28	4.14	4.83	5.51	6.96	8.12	324.20	93.18	112.39	282.88
IMG_B035	2560x1920	directo	3.99	5.33	55.48	142.59	4.97	77.27	67.49	87.35	69.28	2.47	2.47	2.47	3.57	3.57	3.57	672.63	470.48	32.99	52.57
IMG_B036	2560x1920	directo	1.36	2.34	51.10	138.85	2.26	75.34	68.77	91.28	29.94	0.00	2.77	2.77	0.00	9.26	9.26	1412.31	828.51	188.09	190.50
IMG_B037	2560x1920	directo	1070.75	357.88	16.90	264.80	51.00	60.52	60.03	99.19	57.40	3.55	2.96	2.96	6.19	5.15	5.15	4.03	4.05	99.23	113.59
IMG_B038	2560x1920	directo	238.70	86.72	306.00	0.00	54.00	0.00	100.00	Inf	29.70	0.00	2.97	2.97	0.00	10.00	10.00	17.60	17.44	244.47	263.39
IMG_B039	2560x1920	directo	42.13	606.96	358.01	0.00	1.99	0.00	100.00	Inf	43.11	0.00	2.99	2.99	0.00	6.94	6.94	4.17	4.20	34.49	86.00
IMG_B040	2560x1920	directo	840.70	178.09	346.41	114.46	40.78	58.03	53.19	91.66	32.39	0.00	2.49	2.99	0.00	7.69	9.23	3.86	3.92	75.17	330.75
IMG_B041	2560x1920	directo	933.08	681.49	0.00	145.60	34.40	100.00	0.00	0.00	32.43	0.00	2.56	2.99	0.00	7.89	9.21	3.05	3.09	208.20	228.76
IMG_B042	2560x1920	directo	0.40	0.80	43.65	133.19	0.63	69.81	72.15	103.34	68.26	3.30	3.85	2.75	4.84	5.65	4.03	1897.35	1873.38	53.26	359.03
IMG_B043	2560x1920	directo	137.97	50.68	53.69	0.00	53.69	0.00	100.00	Inf	63.73	2.45	5.39	2.45	3.85	8.46	3.85	30.01	30.32	140.58	144.48

IMG_B044	2560x1920	directo	1.78	1.20	45.12	139.58	1.95	68.63	67.59	98.49	56.58	2.90	4.06	4.64	5.13	7.18	8.21	90.27	89.55	82.41	101.55
IMG_B045	2560x1920	directo	0.74	0.99	45.66	138.57	1.14	70.21	67.74	96.48	58.38	3.24	3.78	2.70	5.56	6.48	4.63	699.06	1753.49	15.46	299.99
IMG_B047	2560x1920	directo	444.25	239.77	0.00	222.81	42.81	100.00	0.00	0.00	30.10	0.00	2.39	2.39	0.00	7.94	7.94	7.83	7.52	26.17	32.66
IMG_B048	2560x1920	directo	0.47	0.66	44.84	137.93	0.72	69.72	68.03	97.57	66.78	3.18	5.30	4.77	4.76	7.94	7.14	304.77	337.06	182.89	231.09
IMG_B049	2560x1920	directo	8.34	13.14	30.59	121.39	6.62	60.48	78.80	130.29	52.23	2.10	3.00	3.30	4.02	5.75	6.32	64.54	71.94	58.10	75.44
IMG_B050	2560x1920	directo	1.76	3.00	40.03	130.95	2.03	67.00	73.17	109.20	55.54	2.08	2.08	3.56	3.74	3.74	6.42	344.64	102.61	72.90	184.49
IMG_B051	2560x1920	directo	131.00	412.54	0.00	170.98	9.02	100.00	0.00	0.00	60.00	2.75	3.30	4.40	4.59	5.50	7.34	10.64	6.10	108.92	200.19
IMG_B053	2560x1920	directo	83.19	76.67	0.00	208.48	28.48	100.00	0.00	0.00	65.24	2.91	3.50	2.91	4.46	5.36	4.46	30.31	32.61	61.02	350.16
IMG_B054	2560x1920	directo	8.39	13.52	29.74	120.96	6.78	59.52	79.10	132.90	56.13	2.77	4.15	3.84	4.93	7.40	6.85	65.06	71.64	57.08	74.42
IMG_B055	2560x1920	directo	8.39	13.52	29.74	120.96	6.78	59.52	79.10	132.90	56.13	2.77	4.15	3.84	4.93	7.40	6.85	65.06	71.64	57.08	74.42
IMG_B056	2560x1920	directo	8.06	16.13	14.02	0.00	14.02	0.00	100.00	Inf	57.87	2.66	4.72	2.66	4.59	8.16	4.59	754.36	1886.64	42.96	75.44
IMG_B057	2560x1920	directo	1356.32	183.42	19.81	251.82	51.33	66.32	44.43	67.00	32.42	0.00	2.75	2.75	0.00	8.47	8.47	2.90	2.97	40.06	41.91
IMG_B059	2560x1920	directo	134.94	262.72	14.40	0.00	14.40	0.00	100.00	Inf	38.97	2.27	2.27	2.27	5.81	5.81	5.81	7.58	10.60	223.58	357.14
IMG_B060	2560x1920	directo	25.05	13.16	1.09	90.08	43.51	70.23	72.43	103.13	62.37	2.58	4.64	2.58	4.13	7.44	4.13	538.11	502.52	210.67	264.60
IMG_B061	2560x1920	directo	1.08	0.95	43.49	139.56	1.07	67.83	66.65	98.25	59.04	2.63	4.06	4.06	4.45	6.88	6.88	79.71	81.85	81.42	101.42
IMG_B062	2560x1920	directo	0.78	0.56	43.93	136.50	1.09	70.81	67.50	95.33	59.64	2.71	5.42	3.16	4.55	9.09	5.30	636.94	395.55	242.96	298.82
IMG_B063	2560x1920	directo	0.30	3.03	44.94	136.48	0.07	70.75	68.80	97.24	50.20	2.17	3.37	3.33	4.32	6.71	6.63	20.61	20.52	80.87	99.81
IMG_B064	2560x1920	directo	4.58	5.98	45.58	139.20	1.30	69.95	67.18	96.03	52.85	2.29	3.95	3.91	4.32	7.47	7.40	24.38	24.39	82.71	101.75
IMG_B065	2560x1920	directo	2.62	5.18	45.75	136.01	0.36	71.20	69.90	98.17	47.15	1.79	2.96	2.83	3.80	6.29	6.01	12.09	12.17	80.90	99.25
IMG_B066	2560x1920	directo	4.76	38.88	52.45	138.60	0.64	78.78	67.12	85.20	43.59	1.47	2.47	2.26	3.37	5.66	5.18	12.34	11.84	87.94	105.36
IMG_B067	2560x1920	directo	0.95	6.17	47.13	135.47	0.22	73.58	69.88	94.98	51.10	2.21	3.46	3.46	4.33	6.77	6.77	20.55	20.34	82.00	99.78
IMG_B068	2560x1920	directo	30.34	17.01	55.73	135.07	5.64	78.07	78.60	100.68	48.67	1.88	3.34	3.29	3.87	6.85	6.77	19.61	19.87	88.14	103.21
IMG_B069	2560x1920	directo	323.50	276.16	14.55	242.85	24.33	22.76	83.40	366.42	15.47	1.10	1.56	0.12	7.12	10.07	0.78	13.45	14.35	207.51	210.70
IMG_B070	2560x1920	directo	3.61	0.39	44.80	134.33	1.11	71.83	70.16	97.67	33.83	1.49	1.59	0.86	4.40	4.70	2.54	233.77	27.69	97.19	174.35
IMG_B071	2560x1920	directo	2.52	2.72	46.36	136.05	0.77	71.44	70.35	98.48	45.65	2.67	2.54	1.80	5.86	5.56	3.94	26.69	116.82	13.42	81.39

IMG_B072	2560x1920	directo	2.11	2.95	46.26	136.22	0.65	71.45	70.00	97.97	50.87	1.84	3.29	3.13	3.62	6.46	6.16	27.44	27.54	81.41	99.67
IMG_B073	2560x1920	directo	1.04	0.26	44.55	135.28	0.32	70.56	69.97	99.16	50.60	1.88	3.21	3.10	3.71	6.34	6.12	27.11	27.25	79.58	98.05
IMG_B074	2560x1920	directo	214.31	209.61	19.80	212.28	21.39	12.86	87.40	679.51	60.53	3.07	3.07	3.07	5.07	5.07	5.07	3.60	11.89	99.97	117.44
IMG_B075	2560x1920	directo	2.41	7.47	41.42	132.10	0.90	67.32	73.15	108.65	58.49	3.44	3.73	3.79	5.88	6.38	6.48	31.52	32.67	73.91	92.39
IMG_B076	2560x1920	directo	6.34	10.26	49.49	139.81	2.19	73.50	67.40	91.71	39.28	1.49	2.38	0.80	3.78	6.07	2.05	31.30	309.58	46.97	105.00
IMG_B077	2560x1920	directo	6.07	6.37	41.12	132.43	2.07	68.45	71.34	104.23	50.70	1.55	3.35	3.09	3.05	6.60	6.10	29.82	30.30	75.07	93.52
IMG_B078	2560x1920	directo	3.69	10.60	49.56	139.03	1.27	74.66	67.23	90.04	50.75	1.73	2.74	2.85	3.42	5.40	5.62	30.53	29.63	86.45	104.56
IMG_B079	2560x1920	directo	1.42	0.66	38.88	140.95	0.57	64.98	63.62	97.92	52.54	2.32	3.09	3.04	4.41	5.88	5.80	28.43	28.29	77.25	99.81
IMG_B080	2560x1920	directo	586.43	387.13	33.91	143.16	31.99	96.69	8.96	9.27	51.40	3.09	4.12	2.06	6.01	8.01	4.00	5.22	2.90	232.34	278.45
IMG_B002	1024x768	otsu	77.34	84.33	13.40	228.71	19.16	17.35	85.34	491.80	31.85	0.00	2.57	2.57	0.00	8.06	8.06	44.70	7.34	221.46	254.79
IMG_B003	1024x768	otsu	87.93	143.64	12.86	203.30	13.91	10.10	90.05	891.62	54.69	2.60	3.65	4.17	4.76	6.67	7.62	13.17	12.20	172.26	196.46
IMG_B004	1024x768	otsu	166.09	42.68	52.74	215.26	50.15	15.08	85.52	567.19	56.62	3.13	5.64	4.60	5.54	9.96	8.12	19.57	11.82	144.64	260.97
IMG_B005	1024x768	otsu	177.72	19.29	282.25	0.00	77.75	0.00	100.00	Inf	48.08	3.01	2.50	2.50	6.25	5.21	5.21	46.94	53.77	45.79	352.77
IMG_B006	1024x768	otsu	121.24	32.03	0.00	117.85	62.15	100.00	0.00	0.00	37.75	0.00	2.52	2.52	0.00	6.67	6.67	41.39	38.35	46.63	82.83
IMG_B007	1024x768	otsu	47.52	49.86	52.54	172.14	13.06	73.11	41.08	56.19	34.84	2.12	2.55	2.97	6.10	7.32	8.54	17.67	15.78	29.23	254.56
IMG_B008	1024x768	otsu	119.59	32.15	52.87	204.83	49.22	13.56	87.83	647.89	37.62	0.00	2.41	3.38	0.00	6.41	8.97	38.26	24.85	132.31	298.37
IMG_B009	1024x768	otsu	183.43	209.31	20.62	204.98	20.91	6.61	93.41	1413.93	31.34	0.00	2.80	2.80	0.00	8.93	8.93	13.15	6.74	170.63	190.45
IMG_B010	1024x768	otsu	338.22	52.72	287.31	0.00	72.69	0.00	100.00	Inf	53.00	3.18	3.71	4.24	6.00	7.00	8.00	14.54	14.00	133.53	154.59
IMG_B011	1024x768	otsu	174.32	98.48	29.48	98.53	30.96	93.15	82.64	88.72	44.49	0.00	2.85	3.42	0.00	6.41	7.69	15.45	15.74	330.76	331.41
IMG_B012	1024x768	otsu	61.52	18.63	38.66	110.27	24.83	94.30	74.40	78.89	36.49	2.12	3.39	2.12	5.81	9.30	5.81	33.26	53.13	141.19	279.96
IMG_B013	1024x768	otsu	0.12	0.02	43.52	136.38	0.15	69.14	68.88	99.63	66.44	1.69	3.73	2.37	2.55	5.61	3.57	1060.43	1028.80	257.68	306.42
IMG_B014	1024x768	otsu	70.90	16.60	0.00	244.91	64.91	100.00	0.00	0.00	30.22	0.00	2.70	2.70	0.00	8.93	8.93	140.73	59.50	95.84	116.67
IMG_B016	1024x768	otsu	409.76	427.05	25.04	207.67	25.07	1.27	98.74	7800.73	0.00	0.00	1.91	0.96	NaN	Inf	Inf	8.02	7.73	279.39	280.91
IMG_B017	1024x768	otsu	472.03	34.83	70.77	268.15	79.71	51.99	49.17	94.56	30.47	0.00	2.12	2.12	0.00	6.94	6.94	14.87	14.50	125.54	209.77
IMG_B018	1024x768	otsu	266.49	235.68	28.46	165.92	25.98	6.40	95.19	1487.91	48.06	2.91	2.43	2.91	6.06	5.05	6.06	6.99	7.87	289.68	299.31

IMG_B019	1024x768	otsu	98.76	51.49	6.33	196.07	12.39	62.47	37.88	60.63	38.97	2.15	3.52	2.35	5.53	9.05	6.03	10.83	8.74	24.50	310.63
IMG_B020	1024x768	otsu	325.69	262.59	82.74	196.65	29.89	87.19	25.05	28.73	48.81	1.32	2.48	2.15	2.71	5.08	4.41	11.12	12.91	165.47	172.31
IMG_B021	1024x768	otsu	63.27	258.96	0.00	186.96	6.96	100.00	0.00	0.00	38.88	2.39	1.50	2.09	6.15	3.85	5.38	15.87	10.33	237.34	300.83
IMG_B022	1024x768	otsu	0.12	0.32	44.71	136.29	0.15	70.19	69.31	98.75	67.49	2.22	2.97	2.60	3.30	4.40	3.85	1130.61	825.43	201.94	226.99
IMG_B025	1024x768	otsu	0.56	0.26	43.39	138.13	0.76	67.96	67.96	100.00	65.82	1.99	5.19	5.98	3.03	7.88	9.09	75.34	817.20	42.28	214.73
IMG_B026	1024x768	otsu	289.99	15.20	75.46	259.67	78.92	82.26	17.78	21.61	44.00	2.50	3.50	4.00	5.68	7.95	9.09	12.29	21.71	25.05	37.03
IMG_B028	1024x768	otsu	10.69	59.31	60.32	158.11	2.99	84.97	42.45	49.96	56.35	1.97	4.56	3.21	3.50	8.10	5.69	36.34	17.73	103.80	178.46
IMG_B029	1024x768	otsu	34.44	66.13	0.00	165.41	14.59	100.00	0.00	0.00	31.00	0.00	2.62	2.62	0.00	8.45	8.45	36.50	48.18	201.90	315.76
IMG_B030	1024x768	otsu	46.87	147.82	53.82	182.52	7.77	92.25	11.72	12.71	36.60	1.89	2.64	3.02	5.15	7.22	8.25	14.32	13.47	178.22	207.43
IMG_B033	1024x768	otsu	268.24	57.94	26.98	121.36	46.57	96.19	20.97	21.80	33.50	2.06	2.64	1.47	6.14	7.89	4.39	13.81	2.71	302.44	351.27
IMG_B034	1024x768	otsu	32.72	28.37	43.48	128.76	9.35	56.30	87.41	155.27	52.75	2.42	2.42	3.62	4.58	4.58	6.87	36.94	43.99	218.59	270.63
IMG_B035	1024x768	otsu	129.70	15.46	9.22	225.40	28.10	54.82	50.37	91.89	38.52	2.29	3.59	3.26	5.93	9.32	8.47	20.03	18.04	82.63	118.04
IMG_B036	1024x768	otsu	153.43	95.98	2.25	204.79	18.81	74.35	27.18	36.55	62.80	3.93	2.80	2.80	6.25	4.46	4.46	10.94	11.30	165.77	187.82
IMG_B037	1024x768	otsu	210.01	64.30	0.00	121.48	58.52	100.00	0.00	0.00	34.38	1.95	2.73	2.34	5.68	7.95	6.82	22.92	22.37	193.79	271.41
IMG_B038	1024x768	otsu	359.07	110.16	58.97	117.50	52.52	13.17	106.24	806.52	40.81	2.49	2.49	1.56	6.11	6.11	3.82	11.11	10.68	90.24	97.79
IMG_B039	1024x768	otsu	345.56	88.49	0.00	242.88	62.88	100.00	0.00	0.00	42.51	2.44	3.42	2.93	5.75	8.05	6.90	12.02	13.99	312.87	329.59
IMG_B040	1024x768	otsu	231.78	175.33	0.00	146.54	33.46	100.00	0.00	0.00	28.75	0.00	2.32	2.32	0.00	8.06	8.06	2.77	11.00	140.58	310.08
IMG_B041	1024x768	otsu	109.69	17.68	72.13	0.00	72.13	0.00	100.00	Inf	43.81	2.65	3.54	2.65	6.06	8.08	6.06	52.68	73.02	243.33	272.39
IMG_B043	1024x768	otsu	301.47	68.99	0.00	114.59	65.41	100.00	0.00	0.00	47.53	2.16	2.47	3.09	4.55	5.19	6.49	16.08	9.44	204.86	242.80
IMG_B044	1024x768	otsu	176.71	106.08	39.79	0.00	39.79	0.00	100.00	Inf	32.35	1.89	2.16	2.16	5.83	6.67	6.67	18.84	20.14	144.90	146.37
IMG_B045	1024x768	otsu	631.71	246.31	16.24	130.32	46.08	97.00	6.87	7.08	58.13	2.46	2.96	2.96	4.24	5.08	5.08	0.57	2.86	201.74	282.53
IMG_B046	1024x768	otsu	601.96	308.31	10.13	222.46	41.28	96.73	3.85	3.98	0.00	0.00	3.04	2.17	NaN	Inf	Inf	11.60	10.68	280.82	283.87
IMG_B049	1024x768	otsu	523.90	107.70	347.85	114.78	63.29	97.42	4.20	4.32	46.08	2.71	3.61	2.71	5.88	7.84	5.88	8.95	7.11	120.74	297.32
IMG_B050	1024x768	otsu	366.12	249.95	0.00	216.22	36.22	100.00	0.00	0.00	31.55	1.31	1.58	1.84	4.17	5.00	5.83	3.70	3.08	349.80	352.35
IMG_B051	1024x768	otsu	176.63	134.12	326.64	0.00	33.36	0.00	100.00	Inf	32.36	1.90	2.66	2.66	5.88	8.24	8.24	16.52	14.01	203.03	211.42

IMG_B054	1024x768	otsu	323.33	162.55	315.16	0.00	44.84	0.00	100.00	Inf	56.44	2.17	3.47	3.91	3.85	6.15	6.92	17.35	11.10	204.12	298.92
IMG_B055	1024x768	otsu	323.33	162.55	315.16	0.00	44.84	0.00	100.00	Inf	56.44	2.17	3.47	3.91	3.85	6.15	6.92	17.35	11.10	204.12	298.92
IMG_B056	1024x768	otsu	187.99	67.18	305.55	0.00	54.45	0.00	100.00	Inf	41.31	2.58	3.10	3.61	6.25	7.50	8.75	18.14	22.82	120.01	123.40
IMG_B057	1024x768	otsu	257.75	102.36	51.54	0.00	51.54	0.00	100.00	Inf	40.21	1.75	3.15	3.15	4.35	7.83	7.83	14.63	15.29	121.59	127.53
IMG_B061	1024x768	otsu	114.67	3.55	53.75	237.18	55.70	56.71	43.34	76.42	49.04	2.88	3.46	3.46	5.88	7.06	7.06	44.92	26.19	131.12	205.65
IMG_B063	1024x768	otsu	0.21	1.41	45.19	136.67	0.13	70.82	68.81	97.16	50.55	2.04	3.46	3.46	4.04	6.84	6.84	51.84	51.51	81.12	99.68
IMG_B065	1024x768	otsu	0.31	1.69	45.51	135.75	0.11	71.20	69.92	98.19	52.17	2.39	3.78	3.75	4.57	7.25	7.19	30.32	30.25	80.83	99.03
IMG_B066	1024x768	otsu	1.76	15.74	52.30	138.80	0.59	78.64	66.76	84.89	55.21	3.37	4.30	4.20	6.11	7.80	7.60	30.67	29.35	88.13	105.46
IMG_B068	1024x768	otsu	12.68	7.05	55.83	135.38	5.86	77.86	78.45	100.76	57.27	3.40	4.92	4.92	5.93	8.59	8.59	48.65	49.09	88.48	103.62
IMG_B069	1024x768	otsu	14.58	11.65	35.55	144.08	1.88	58.47	64.63	110.53	30.22	0.00	2.16	2.16	0.00	7.14	7.14	11.15	11.93	170.16	207.07
IMG_B070	1024x768	otsu	424.64	40.20	296.87	96.68	72.85	48.93	52.64	107.57	31.00	0.00	2.77	2.77	0.00	8.93	8.93	11.16	8.73	134.89	228.91
IMG_B071	1024x768	otsu	257.97	25.71	53.26	224.28	48.16	56.96	43.34	76.10	35.34	1.26	1.80	1.44	3.57	5.10	4.08	11.36	36.31	7.72	23.56
IMG_B072	1024x768	otsu	457.39	0.88	35.14	200.65	28.40	46.90	53.90	114.90	64.95	3.09	2.58	5.67	4.76	3.97	8.73	5.00	5.14	53.12	217.93
IMG_B073	1024x768	otsu	10.18	6.83	56.58	141.36	7.98	75.32	72.99	96.90	27.53	1.40	1.97	1.97	5.10	7.14	7.14	252.09	309.20	10.87	332.29
IMG_B074	1024x768	otsu	221.48	107.76	0.00	134.22	45.78	100.00	0.00	0.00	33.22	2.10	2.45	2.45	6.32	7.37	7.37	17.48	15.03	87.11	165.50
IMG_B075	1024x768	otsu	304.20	212.88	334.62	110.03	31.87	16.10	87.90	546.05	65.94	2.54	3.26	2.54	3.85	4.95	3.85	10.83	4.99	208.08	331.45
IMG_B076	1024x768	otsu	82.92	39.84	10.38	182.56	6.10	54.88	45.35	82.64	52.13	1.87	3.75	2.19	3.59	7.19	4.19	6.30	7.09	331.03	341.08
IMG_B077	1024x768	otsu	74.83	16.30	299.49	167.00	36.83	54.47	54.78	100.57	53.81	2.32	4.26	4.65	4.32	7.91	8.63	50.11	51.21	144.79	224.17
IMG_B078	1024x768	otsu	21.22	192.28	19.12	114.33	2.41	28.87	93.15	322.69	56.92	3.59	3.08	2.56	6.31	5.41	4.50	16.93	16.51	169.48	170.41
IMG_B079	1024x768	otsu	214.21	150.20	48.22	141.31	24.67	95.71	24.27	25.36	63.04	1.90	2.66	5.32	3.01	4.22	8.43	9.35	10.01	96.77	330.29
IMG_B080	1024x768	otsu	297.22	82.30	61.02	0.00	61.02	0.00	100.00	Inf	39.45	2.34	1.95	3.52	5.94	4.95	8.91	9.40	16.85	155.48	177.96
IMG_B001	2560x1920	otsu	98.55	46.69	9.11	213.64	18.42	38.99	63.21	162.10	64.23	2.39	2.79	3.59	3.73	4.35	5.59	28.24	5.32	144.28	198.02
IMG_B002	2560x1920	otsu	202.52	285.80	21.83	152.54	11.26	24.20	82.52	341.07	50.08	3.19	2.28	4.10	6.36	4.55	8.18	3.86	4.84	170.20	252.12
IMG_B003	2560x1920	otsu	73.49	137.38	34.18	152.90	3.91	70.34	44.91	63.84	28.54	1.71	1.71	2.20	5.98	5.98	7.69	4.42	4.74	175.57	213.86
IMG_B004	2560x1920	otsu	229.36	515.33	297.01	174.45	11.22	93.20	11.73	12.59	46.95	1.23	1.95	1.85	2.63	4.16	3.94	4.57	4.08	183.18	341.22

IMG_B005	2560x1920	otsu	785.85	131.96	72.95	181.38	46.77	46.49	75.05	161.42	43.77	0.00	4.09	2.92	0.00	9.33	6.67	4.78	4.62	266.43	280.58
IMG_B006	2560x1920	otsu	15.79	79.54	34.58	137.16	0.95	56.74	70.91	124.98	46.22	2.31	3.24	3.70	5.00	7.00	8.00	5.39	5.34	204.30	331.42
IMG_B007	2560x1920	otsu	188.42	25.07	284.90	0.00	75.10	0.00	100.00	Inf	64.11	1.94	4.35	2.90	3.02	6.79	4.53	22.77	26.25	186.41	202.11
IMG_B008	2560x1920	otsu	157.44	97.33	0.00	218.97	38.97	100.00	0.00	0.00	57.03	2.28	3.04	2.66	4.00	5.33	4.67	16.35	30.11	180.66	235.97
IMG_B009	2560x1920	otsu	32.95	182.35	0.00	185.16	5.16	100.00	0.00	0.00	54.81	2.45	2.45	2.94	4.46	4.46	5.36	10.18	5.26	202.03	309.73
IMG_B010	2560x1920	otsu	19.08	225.85	357.58	0.00	2.42	0.00	100.00	Inf	30.24	1.71	1.71	1.22	5.65	5.65	4.03	21.97	12.29	124.11	147.43
IMG_B011	2560x1920	otsu	430.90	380.48	29.52	0.00	29.52	0.00	100.00	Inf	41.67	2.38	3.57	3.17	5.71	8.57	7.62	5.83	5.88	190.73	202.71
IMG_B012	2560x1920	otsu	308.97	128.74	87.91	161.55	47.14	68.05	94.90	139.46	30.95	0.00	2.86	2.38	0.00	9.23	7.69	12.61	12.56	229.83	285.05
IMG_B013	2560x1920	otsu	0.59	0.44	43.94	136.82	0.30	69.10	68.89	99.70	60.56	3.62	5.01	4.96	5.98	8.28	8.19	43.52	43.58	79.34	98.09
IMG_B014	2560x1920	otsu	44.01	99.70	63.32	132.38	4.27	98.99	73.51	74.26	34.58	0.00	2.44	2.44	0.00	7.04	7.04	8.90	9.04	153.64	166.98
IMG_B015	2560x1920	otsu	0.80	0.43	45.52	134.58	1.17	72.77	69.79	95.90	28.65	0.00	2.05	2.05	0.00	7.14	7.14	105.00	108.43	43.11	250.56
IMG_B016	2560x1920	otsu	852.28	114.81	58.68	148.42	21.53	60.40	79.98	132.41	29.66	0.00	2.80	2.80	0.00	9.43	9.43	2.18	2.19	107.17	107.38
IMG_B017	2560x1920	otsu	0.75	0.36	41.10	140.72	0.86	65.53	65.38	99.77	71.82	2.22	6.67	6.36	3.10	9.29	8.85	627.59	642.50	173.72	264.47
IMG_B018	2560x1920	otsu	0.87	0.22	42.01	137.18	1.39	68.99	66.44	96.31	34.48	0.00	2.87	2.87	0.00	8.33	8.33	2301.21	1454.41	179.92	235.01
IMG_B019	2560x1920	otsu	99.48	82.60	68.99	135.85	11.63	91.57	89.91	98.19	39.51	0.00	3.64	2.60	0.00	9.21	6.58	10.44	10.15	154.55	359.78
IMG_B020	2560x1920	otsu	97.53	257.85	0.00	169.29	10.71	100.00	0.00	0.00	32.89	0.00	2.84	2.84	0.00	8.62	8.62	9.50	10.02	100.74	163.58
IMG_B021	2560x1920	otsu	79.80	193.63	29.05	157.88	3.20	68.50	41.62	60.77	66.25	3.18	2.65	2.65	4.80	4.00	4.00	4.36	4.48	86.08	94.49
IMG_B022	2560x1920	otsu	0.41	0.67	44.73	135.76	0.22	70.65	69.50	98.37	59.56	3.45	5.03	4.85	5.80	8.44	8.14	46.35	47.13	79.60	98.00
IMG_B023	2560x1920	otsu	0.59	0.52	44.75	135.70	0.59	71.13	69.12	97.18	31.37	1.71	2.85	2.09	5.45	9.09	6.67	160.06	87.69	93.88	248.20
IMG_B024	2560x1920	otsu	0.59	0.42	44.33	136.02	0.87	70.99	68.37	96.31	50.40	2.38	3.97	3.57	4.72	7.87	7.09	448.82	737.93	16.47	318.94
IMG_B025	2560x1920	otsu	1.16	0.61	43.42	137.93	0.64	68.13	68.03	99.85	61.08	3.85	5.24	5.11	6.30	8.59	8.37	45.79	45.72	79.88	99.36
IMG_B026	2560x1920	otsu	0.73	0.50	45.32	136.41	0.73	70.22	69.88	99.52	54.97	3.21	3.78	4.35	5.84	6.87	7.90	558.85	137.20	26.96	256.75
IMG_B027	2560x1920	otsu	19.18	9.53	0.00	134.82	45.18	100.00	0.00	0.00	54.14	2.85	2.44	2.44	5.26	4.51	4.51	1015.28	1099.92	333.01	333.32
IMG_B028	2560x1920	otsu	34.45	165.45	5.00	178.55	3.87	17.47	82.62	473.05	30.67	0.75	1.42	1.17	2.45	4.63	3.81	15.46	4.69	200.36	244.48
IMG_B029	2560x1920	otsu	1268.99	38.19	40.67	243.58	56.09	68.30	33.48	49.02	38.25	1.58	1.90	1.90	4.13	4.96	4.96	3.43	4.10	12.59	249.30

IMG_B030	2560x1920	otsu	331.03	48.85	286.44	0.00	73.56	0.00	100.00	Inf	41.47	2.24	2.99	3.36	5.41	7.21	8.11	14.44	14.25	159.03	261.13
IMG_B031	2560x1920	otsu	1128.22	383.08	285.83	150.26	45.64	68.22	39.14	57.37	28.43	1.65	1.85	1.44	5.80	6.52	5.07	3.28	2.95	290.89	292.88
IMG_B032	2560x1920	otsu	607.75	163.05	0.00	118.22	61.78	100.00	0.00	0.00	62.44	2.85	5.44	5.18	4.56	8.71	8.30	7.47	7.48	140.92	220.46
IMG_B033	2560x1920	otsu	775.88	356.03	45.73	164.63	42.97	5.51	97.22	1765.45	54.50	2.50	5.00	3.00	4.59	9.17	5.50	4.45	7.01	225.57	245.85
IMG_B034	2560x1920	otsu	118.00	103.44	84.41	162.42	25.81	87.26	70.23	80.49	52.09	3.04	3.76	4.65	5.84	7.22	8.93	19.81	15.48	149.27	302.48
IMG_B035	2560x1920	otsu	748.84	333.73	89.43	168.65	47.55	67.96	87.16	128.25	37.35	1.76	2.64	2.09	4.72	7.08	5.60	5.00	5.79	272.40	276.66
IMG_B036	2560x1920	otsu	261.11	38.68	33.14	165.61	9.91	53.48	55.78	104.30	43.17	2.59	3.88	2.16	6.00	9.00	5.00	3.36	3.18	220.30	229.17
IMG_B037	2560x1920	otsu	228.18	87.54	12.40	102.68	37.14	76.08	64.53	84.81	32.56	0.00	2.91	2.91	0.00	8.93	8.93	13.82	13.27	74.43	347.38
IMG_B038	2560x1920	otsu	171.43	180.79	18.14	158.73	7.22	67.46	38.25	56.70	41.82	2.58	3.61	2.58	6.17	8.64	6.17	3.72	3.57	10.36	49.29
IMG_B039	2560x1920	otsu	400.04	6.81	78.02	260.99	79.44	47.82	52.21	109.17	63.16	2.99	2.99	3.84	4.73	4.73	6.08	15.40	12.49	242.59	261.60
IMG_B040	2560x1920	otsu	1683.50	798.15	35.11	255.96	42.53	19.75	84.22	426.47	37.55	2.00	2.00	2.00	5.32	5.32	5.32	3.61	3.27	176.68	211.41
IMG_B041	2560x1920	otsu	107.72	107.04	333.29	0.00	26.71	0.00	100.00	Inf	64.04	1.89	4.73	5.68	2.96	7.39	8.87	25.98	26.38	21.69	324.71
IMG_B042	2560x1920	otsu	289.14	140.30	66.82	187.71	30.86	68.43	45.81	66.95	46.33	2.27	2.92	1.94	4.90	6.29	4.20	10.40	9.71	105.67	114.03
IMG_B043	2560x1920	otsu	320.48	216.43	18.67	204.20	20.10	26.03	74.06	284.58	56.44	3.56	4.16	2.97	6.32	7.37	5.26	3.78	4.89	35.83	344.55
IMG_B044	2560x1920	otsu	150.45	1.68	1.07	113.78	49.12	83.27	31.89	38.29	48.18	1.88	4.14	3.76	3.91	8.59	7.81	30.34	25.26	217.83	350.54
IMG_B045	2560x1920	otsu	116.55	0.63	310.85	141.76	44.25	45.12	55.33	122.65	44.16	2.54	3.05	3.55	5.75	6.90	8.05	35.90	27.59	84.73	206.68
IMG_B046	2560x1920	otsu	1151.48	368.46	66.91	96.19	51.85	53.13	142.91	268.97	46.52	2.61	3.14	2.61	5.62	6.74	5.62	3.42	1.35	70.70	338.30
IMG_B047	2560x1920	otsu	1289.18	327.21	312.57	136.01	44.88	74.14	25.89	34.93	27.97	0.00	2.74	2.74	0.00	9.80	9.80	3.27	3.14	164.68	165.04
IMG_B048	2560x1920	otsu	912.63	372.76	48.22	163.20	44.24	7.65	96.53	1261.38	54.50	3.32	4.74	2.84	6.09	8.70	5.22	3.87	4.60	222.82	234.96
IMG_B049	2560x1920	otsu	1105.33	543.00	43.98	139.86	28.61	26.65	93.69	351.55	30.84	1.32	1.54	1.10	4.29	5.00	3.57	3.53	3.78	211.93	227.82
IMG_B050	2560x1920	otsu	250.79	51.09	25.86	121.77	44.63	94.76	23.63	24.94	36.14	0.00	3.10	3.10	0.00	8.57	8.57	14.71	14.12	240.04	241.13
IMG_B051	2560x1920	otsu	411.49	30.55	59.95	149.33	34.45	43.05	90.73	210.74	42.19	2.13	2.56	2.13	5.05	6.06	5.05	3.59	10.96	184.21	309.81
IMG_B052	2560x1920	otsu	320.66	47.49	0.00	106.50	73.50	100.00	0.00	0.00	29.78	0.00	2.76	2.76	0.00	9.26	9.26	19.67	20.96	284.62	298.79
IMG_B054	2560x1920	otsu	1611.56	450.56	51.40	175.61	43.10	17.46	89.14	510.50	33.15	2.07	2.07	2.07	6.25	6.25	6.25	3.70	3.62	216.92	234.54
IMG_B055	2560x1920	otsu	1611.56	450.56	51.40	175.61	43.10	17.46	89.14	510.50	33.15	2.07	2.07	2.07	6.25	6.25	6.25	3.70	3.62	216.92	234.54

IMG_B056	2560x1920	otsu	119.39	82.84	52.30	161.23	12.60	67.53	55.04	81.51	40.84	0.00	2.62	3.66	0.00	6.41	8.97	10.29	9.04	307.02	356.53
IMG_B057	2560x1920	otsu	2.64	156.55	0.48	0.00	0.48	0.00	100.00	Inf	31.18	0.00	2.69	2.69	0.00	8.62	8.62	19.52	18.69	49.26	206.02
IMG_B058	2560x1920	otsu	1360.62	215.65	58.10	150.60	38.47	33.61	92.72	275.86	35.26	0.00	2.80	2.80	0.00	7.94	7.94	3.51	3.52	239.93	244.79
IMG_B060	2560x1920	otsu	268.49	135.01	350.18	144.57	19.19	37.67	64.70	171.74	64.43	2.52	2.52	3.02	3.91	3.91	4.69	7.04	5.61	155.73	350.80
IMG_B061	2560x1920	otsu	1561.94	0.75	274.04	92.87	86.63	57.68	42.33	73.39	27.72	1.47	1.72	1.72	5.31	6.19	6.19	1.24	0.81	224.14	241.21
IMG_B062	2560x1920	otsu	28.18	90.09	29.67	165.27	5.27	81.86	23.49	28.69	35.42	0.00	2.68	2.68	0.00	7.58	7.58	19.75	10.63	57.01	260.64
IMG_B063	2560x1920	otsu	0.20	3.12	44.94	136.54	0.05	70.73	68.75	97.20	51.40	2.36	3.52	3.64	4.60	6.84	7.09	20.59	20.57	80.91	99.85
IMG_B065	2560x1920	otsu	2.82	5.34	45.80	136.04	0.38	71.22	69.90	98.14	52.27	2.40	3.77	3.76	4.59	7.22	7.20	12.07	12.12	81.09	99.30
IMG_B066	2560x1920	otsu	5.25	39.61	52.41	138.88	0.70	78.63	66.81	84.97	55.37	3.43	4.29	4.26	6.20	7.75	7.69	12.26	11.75	88.30	105.50
IMG_B067	2560x1920	otsu	27.14	91.51	0.00	188.43	8.43	100.00	0.00	0.00	0.00	0.00	0.14	0.23	NaN	Inf	Inf	609.74	659.94	311.56	315.82
IMG_B068	2560x1920	otsu	32.13	18.01	55.92	135.48	5.93	77.88	78.41	100.68	56.33	3.45	4.75	4.83	6.12	8.44	8.58	19.44	19.60	88.65	103.67
IMG_B069	2560x1920	otsu	783.50	77.53	344.68	112.09	60.22	88.87	16.84	18.95	27.78	1.39	1.67	1.39	5.00	6.00	5.00	8.36	10.07	171.71	196.98
IMG_B070	2560x1920	otsu	810.70	104.61	44.77	207.07	35.28	54.21	46.99	86.69	31.79	1.63	1.90	2.45	5.13	5.98	7.69	2.54	4.06	18.66	50.72
IMG_B071	2560x1920	otsu	830.38	262.92	47.11	199.15	30.04	62.60	40.31	64.39	40.91	1.75	1.75	2.10	4.27	4.27	5.13	3.29	2.44	39.69	243.03
IMG_B072	2560x1920	otsu	683.19	508.09	86.06	159.28	30.34	86.31	81.25	94.14	34.36	1.92	1.74	1.22	5.58	5.08	3.55	7.30	7.16	197.94	204.19
IMG_B073	2560x1920	otsu	1373.39	774.38	46.56	138.34	40.58	99.92	1.89	1.89	36.52	0.98	2.54	2.34	2.67	6.95	6.42	2.45	2.43	173.15	337.11
IMG_B074	2560x1920	otsu	41.88	182.32	0.00	173.45	6.55	100.00	0.00	0.00	48.19	2.71	4.33	3.79	5.62	8.99	7.87	20.11	11.09	99.56	232.92
IMG_B075	2560x1920	otsu	1207.14	839.28	5.95	259.47	30.43	43.21	78.75	182.26	56.32	2.22	4.88	2.00	3.94	8.66	3.54	2.13	1.57	77.54	180.35
IMG_B076	2560x1920	otsu	439.38	404.59	87.66	160.55	26.82	91.38	75.61	82.74	48.93	2.47	3.95	2.97	5.05	8.08	6.06	5.13	4.42	194.75	252.74
IMG_B077	2560x1920	otsu	464.24	821.76	15.77	0.00	15.77	0.00	100.00	Inf	43.00	2.02	1.68	2.69	4.69	3.91	6.25	3.03	3.11	161.22	257.24
IMG_B078	2560x1920	otsu	594.69	48.03	51.91	171.33	32.41	38.32	75.44	196.88	63.10	1.58	2.89	2.63	2.50	4.58	4.17	6.44	6.26	100.31	129.67
IMG_B079	2560x1920	otsu	672.74	325.13	50.18	193.89	26.35	68.25	36.46	53.43	29.33	0.00	1.98	2.38	0.00	6.76	8.11	4.40	4.05	201.54	210.90
IMG_B080	2560x1920	otsu	17.28	229.45	2.16	0.00	2.16	0.00	100.00	Inf	31.82	0.00	2.73	2.73	0.00	8.57	8.57	13.75	9.80	92.59	167.75

Tabla A.III – Tabla de Datos

ANEXO IV – Código fuente

Todo el código fuente desarrollado se encuentra implementado en el interior de una función MATLAB denominada *PlataformaUAV.m* que tiene la siguiente sintaxis:

PlataformaUAV [*<ruta_imagen>*], [*escalado*], [*tipo_umbral*], [*auto_umbral*]: *probabilidad*

[*ruta_imagen*]: *ruta absoluta o relativa al lugar donde se encuentra almacenada la imagen que se desea procesar. En caso de omisión, se utiliza el valor de la variable Configuracion.ImagenEntrada que por defecto es 'Imagenes\IMG_A001.jpg';*

[*escalado*]: *permite variar la resolución de entrada; el número de filas, es decir, alto de la nueva resolución. En caso de omisión del parámetro entonces se utiliza el valor almacenado en la variable Configuracion.FilaResolucionDeseada. Por defecto es 1944.*

[*tipo_umbral*]: *método de umbralización que se desea utilizar. Los valores posibles son <directo|otsu|automatico>. En caso de omisión del parámetro se utiliza el valor almacenado en la variable Configuracion.TipoUmbral. Por defecto es directo.*

[*auto_umbral*]: *booleano que indica si se utiliza el sistema autónomo de umbral (hace un test de forma automática con otro umbral si $0.6 < \text{probabilidad}$. Si se omite utiliza el valor de la variable Configuracion.Autoumbralizacion. Por defecto es false.*

Una vez explicada la sintaxis, a continuación se muestra un ejemplo de invocación en el que se especifican los tres primeros parámetros y se obvia el cuarto.

PlataformaUAV('imagenes\IMG_A001.jpg', 1944, 'directo');

Finalmente, se muestra el código fuente desarrollado en MATLAB que se alberga en el interior del fichero *PlataformaUAV.m*:

```
%Curso: 2013/2014
%Autor: José Antonio García Pulido
%Titulación: Master Interuniversitario en Ingeniería de Sistemas y Control
%Universidad: UNED - UCM
%Nombre del Programa: PlataformaUAV.m

function Probabilidad = PlataformaUAV (RutaImagen, Escalado, TipoUmbral, AutoUmbral)

%Borramos la pantalla, limpiamos variables y cerramos todas las ventanas
clc;
close all;

%-----
% Estructuras de datos del programa principal
%-----
%Estructura de datos que almacena los datos relativos a las elipses. Tanto
%los relativos a la orientación como sus propiedades características
Elipse = zeros(1, 1000);
Elipse.CentroideEjeX = 0;
Elipse.CentroideEjeY = 0;
Elipse.Area = 0;
Elipse.Excentricidad = 0;
Elipse.NumeroRegion = -1;
Elipse.LongitudEjeGrande = 0;
Elipse.LongitudEjeChico = 0;
Elipse.Orientacion = 0;
Elipse.ElipseCentral = false;

%Datos de la elipse que está más a la izquierda
E1.CentroideEjeX = 999999;
E1.CentroideEjeY = 0;
E1.NumeroRegion = -1;
E1.Excentricidad = 0;
E1.Orientacion = 0;
```

```
E1.Area= 0;
```

```
%Datos de la elipse que está más a la derecha
```

```
E3.CentroideEjeX = 0;  
E3.CentroideEjeY = 0;  
E3.NumeroRegion = -1;  
E3.Excentricidad = 0;  
E3.Orientacion = 0;  
E3.Area = 0;
```

```
%Datos de la elipse que está en el medio de las otras dos elipses  
%vista la figura geométrica de manera frontal (con los dos puntos arriba)
```

```
E2.CentroideEjeX = 0;  
E2.CentroideEjeY = 0;  
E2.NumeroRegion = -1;  
E2.Excentricidad = 0;  
E2.Orientacion = 0;  
E2.Area = 0;
```

```
%Datos relativos a los círculos que forman parte de la figura geométrica
```

```
Circulo.CentroideEjeX = 0;  
Circulo.CentroideEjeY = 0;  
Circulo.Area = 0;  
Circulo.NumeroRegion = -1;  
Circulo.Diametro = 0;  
Circulo.Distancia = [];  
Circulo.RatioArea = 0;  
Circulo.DiferenciaRatioDistancia = 0;  
Circulo.Procesado = false;
```

```
%Datos del círculo 1, el más grande que tiene por centro el mismo que el  
círculo 2, que está justo en el centro geométrico de la figura
```

```
C1.CentroideEjeX = 0;  
C1.CentroideEjeY = 0;  
C1.Area = 0;  
C1.NumeroRegion = -1;
```

```
%Datos relativos al círculo 2
```

```
C2.CentroideEjeX = 0;  
C2.CentroideEjeY = 0;  
C2.Area = 0;  
C2.NumeroRegion = -1;
```

```
%Datos relativos al círculo 3
```

```
C3.CentroideEjeX = 0;  
C3.CentroideEjeY = 0;  
C3.Area = 0;  
C3.NumeroRegion = -1;
```

```
%Datos relativos al círculo 4
```

```
C4.CentroideEjeX = 0;  
C4.CentroideEjeY = 0;  
C4.Area = 0;  
C4.NumeroRegion = -1;
```

```
%Variable que almacena el número de región de la elipse central
```

```
Datos.NumRegionElipseCentral = -1;
```

```
%Variable que rota la imagen original para simular una imagen rotada
```

```
Datos.SimulacionRotacionImagenOriginal = 0;
```

```
%Variable que almacena el ángulo que hay entre las elipses laterales
```

```
Datos.Angulo_E1_E3 = -500;
```

```
%Variable que contiene el ángulo que hay entre la elipse izquierda y central
```

```
Datos.Angulo_E2_E3 = -500;
```

```
%Variable que contiene el ángulo que hay entre la elipse izquierda y central
```

```
Datos.Angulo_E2_E1 = -500;
```

```
%Variable que contiene la media aritmética del área de las tres elipses
```

```
Datos.MediaAreaElipses = 0;
```

```
%Variable que contiene el punto central de la figura según la orientación
```

```
%de las elipses 1 y 3, es decir, la elipse izquierda y derecha
```

```
Datos.PuntoCentralEjeX = 0;
```

```
Datos.PuntoCentralEjeY = 0;
```

```
%Array en el que se almacenan las distancias entre elipses cuando hay más  
de tres elipses y por tanto tenemos que hacer descartes.
```

```

Distancia.Elipse_A = 0;
Distancia.Elipse_B = 0;
Distancia.Elipse_C = 0;
Distancia.Elipse_A_B = 0;
Distancia.Elipse_A_C = 0;
Distancia.Elipse_B_C = 0;

%Booleanaos que almacenan los resultados parciales y totales del análisis
Resultado.PosicionesCentrosElipses = false;
Resultado.AngulosElipses = false;
Resultado.ProporcionalidadDistancias = false;
Resultado.SimilitudAreaElipses = false;
Resultado.SimilitudExcentricidadElipses = false;
Resultado.IdentificacionC1 = false;
Resultado.IdentificacionC2 = false;
Resultado.IdentificacionC3 = false;
Resultado.IdentificacionC4 = false;
Resultado.IdentificacionCirculos = false;
Resultado.ProbabilidadFinalIdentificacionFigura = 0;
Resultado.TotalAciertos = 0;
Resultado.TotalCriteriosAnalisis = 10;

%-----
% 0) Variables de configuración del programa
%-----
%---Opciones de configuración general.
%Los más relevantes son pasados como parámetros de la función principal
%Variable que controla los números de las ventanas de resultados
Configuracion.NumeroFigura = 1;
%Gestión de los parámetros de entrada de la función principal
%Ruta donde se encuentra la imagen de entrada a procesar
if ~exist('RutaImagen', 'var') || isempty(RutaImagen)
    Configuracion.ImagenEntrada = 'Imágenes\IMG_A001.jpg';
else
    Configuracion.ImagenEntrada = RutaImagen;
end

%Tipo de umbralización para binarización: Valores: [otsu, directo, automatico]
if ~exist('TipoUmbral', 'var') || isempty(TipoUmbral)
    Configuracion.TipoUmbral = 'directo';
else
    Configuracion.TipoUmbral = TipoUmbral;
    %Si se utiliza el tipo automático habilitamos la autoumbralizacion
    if (strcmp(TipoUmbral,'automatico'))
        Configuracion.Autoumbralizacion = true;
    end
end

%Resolución a la que se escala la imagen original: 1944 = 4Mpx 2,53125
if ~exist('Escalado', 'var') || isempty(Escalado)
    Configuracion.FilaResolucionDeseada = 1944;
else
    Configuracion.FilaResolucionDeseada = Escalado;
end

%Gestión del autoumbral para no entrar en bucle infinito en la recursividad
if ~exist('AutoUmbral', 'var') || isempty(AutoUmbral)
    %En caso de emplear el umbral automático se conmutan los tipos de
    %umbralización siempre y cuando la probabilidad final sea inferior al
    %valor establecido en la variable Configuracion.ProbabilidadMinimaAutoUmbral
    Configuracion.Autoumbralizacion = false;
else
    Configuracion.Autoumbralizacion = AutoUmbral;
end

%Valor de umbral cuando se utiliza el método directo: 0.80
Configuracion.Umbral = 0.80;
%En caso que se active la ExportarDatosCSV, indicar la ruta del fichero
Configuracion.RutaDatosCSV = 'DatosCSV.csv';
%En caso que se active la ExportarResultadosCSV, sirve para indicar la ruta
Configuracion.RutaResultadosCSV = 'ResultadosCSV.csv';
%Si está habilitada muestra el tiempo total de CPU o de ejecución.
Configuracion.MostrarTiempoCPU = true;
%Tipo de distancia que se debe usar. Por defecto 0, distancia euclídea

```

```

Configuracion.TipoDistancia = 0;
%Valor mínima que se debe alcanzar para no conmutar el tipo de umbralización
Configuracion.ProbabilidadMinimaAutoumbralizacion = 50;
%Numero de Elipses a detectar en la figura
Configuracion.NumeroElipses = 3;

%----Opciones de visualización para la depuración del proceso
Configuracion.ExportarDatosCSV = true;
Configuracion.ExportarResultadosCSV = true;
Configuracion.VerboseConsola = true;
Configuracion.MostrarImagenOriginal = false;
Configuracion.MostrarImagenIntensidad = false;
Configuracion.MostrarImagenBinaria = false;
Configuracion.MostrarImagenBinariaLimpia = false;
Configuracion.MostrarRegionesConexas = true;
Configuracion.MostrarImagenOriginalOrientada = false;
Configuracion.MostrarImagenBinariaOrientada = false;
Configuracion.MostrarImagenBinariaLimpiaOrientada = false;
Configuracion.MostrarRegionesConexasOrientada = true;

%----Opciones de detección de elipses
%Las elipses se dibujarán sobre la imagen binaria una vez se haya procesado
%con el operador morfológico close, es decir, sobre la imagen binaria limpia
Configuracion.DibujarElipses = true;
Configuracion.DibujarElipsesOrientadas = true;
Configuracion.DibujarCentroidesElipses = true;
Configuracion.ColorElipses = 'r';
Configuracion.SimboloCentroidesElipses = '*';
Configuracion.FiltradoAreaMinimaElipse = 165;
Configuracion.FiltradoAreaMaximaElipse = 80000;
Configuracion.FiltroExcentricidadMinElipse = 0.55;
Configuracion.FiltroExcentricidadMaxElipse = 0.95;
%Las siguientes opciones aplican cuando surgen más de 3 regiones en el
%proceso de obtención de elipses, y se aplican de tres en tres elipses, de
%tal modo que el total es igual al número de combinaciones sin repetición.
%Dejamos un mayor porcentaje en la excentricidad para las fotos inclinadas
Configuracion.DeteccionUmbralAreaElipse = 6.5;
Configuracion.DeteccionUmbralExcentricidadElipse = 30;
Configuracion.MaximoValorAutoUmbralElipses = 70;

%----Opciones de detección de círculos
%Los círculos se dibujarán sobre la imagen binaria, de tal forma, que se
%dibujaran en la misma figura que la que se utiliza para representar las
%elipses ya orientadas. Si DibujarElipsesOrientadas = false se dibuja en
%una nueva
Configuracion.DibujarCirculos = true;
Configuracion.DibujarCentroidesCirculos = false;
Configuracion.ColorCirculos = 'r';
Configuracion.UmbralDibujoCirculo = 0.4;
Configuracion.FactorRadio = 0.55;
Configuracion.SimboloCentroidesCirculos = '*';
Configuracion.FiltradoAreaMinimaCirculo = 5;
Configuracion.FiltradoAreaMaximaCirculo = 45000;
Configuracion.FiltroExcentricidadMinCirculo = 0;
Configuracion.FiltroExcentricidadMaxCirculo = 0.87;
Configuracion.MaximoValorAutoUmbralCirculos = 70;

%-----OPCIONES DE IDENTIFICACIÓN DE LA FIGURA-----
%Define hasta que longitud del array de círculos candidatos se va a emplear
%la estrategia 2. Según el estudio computacional, con arrays de hasta 10
%elementos interesa emplear la estrategia de identificación 2. Con mas la 1
Configuracion.NumMaxCirculosEstrategia2 = 5;
Configuracion.ProbabilidadMinimaDeteccionFigura = 0.6;

%-----ELIPSES-----
%1) Análisis de las posiciones de los centros de las elipses E1, E2 y E3
%Angulo de error alineación elipses laterales (E1 y E3) sobre Eje OY
Configuracion.ErrorAlineacionEjeY_C1_C3 = (12*pi)/180;
%Porcentaje error máximo eje X del centro de la elipse E2
Configuracion.ErrorAlineacionEjeX_C2 = 29.2;

%2) Ángulos que forman las rectas que unen los centros de las elipses
%Angulo entre las elipses 1 y 3, es decir, izquierda y derecha
Configuracion.ValorAnguloE1_3 = 0;

```

```

%Angulo entre las elipses 2 y 3, es decir, central y derecha
Configuracion.ValorAnguloE2_3 = 45;
%Angulo entre las elipses 1 y 2, es decir, central e izquierda
Configuracion.ValorAnguloE1_2 = 135;
%Máxima desviación en grados para la identificación de los ángulos
Configuracion.GradosUmbralErrorOrientacionAngulosElipses = 58.8;

%3) Ratios entre las distancias que hay en los centros de las elipses
%Constante de proporcionalidad entre las distancias 1_2 y 1_3
Configuracion.ProporcionalidadDistanciaE1_2_E1_3 = 72;
%Constante de proporcionalidad entre las distancias 2_3 y 1_3
Configuracion.ProporcionalidadDistanciaE2_3_E1_3 = 69;
%Constante de proporcionalidad entre las distancias 2_3 y 1_2
Configuracion.ProporcionalidadDistanciaE2_3_E1_2 = 97;
%Porcentaje relativo al umbral de error para el análisis de distancias
Configuracion.PorcentajeUmbralErrorDistancia_Elipses = 22.4;

%4) Área y excentricidad de las elipses de la figura
Configuracion.PorcentajeUmbralDiferenciaAreaElipse = 6.5;
Configuracion.PorcentajeUmbralDiferenciaExcentricidadElipse = 9.4;

%-----CIRCULOS-----
Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos = 41.4;
Configuracion.PorcentajeUmbralErrorDistanciaCirculos = 28;
Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_Elipses = 48;
Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_C1 = 45;

%5) Análisis del círculo1 de la figura
Configuracion.ValorAnguloC1_E3 = 180;
Configuracion.RatioAreaC1MediaElipses = 50;

%6) Análisis del círculo2 de la figura
Configuracion.ValorAnguloC2_E3 = 180;
Configuracion.RatioAreaC2MediaElipses = 2.29;
Configuracion.RatioC12Area_C11Area = 4.4;

%7) Análisis del círculo3 y 4 de la figura
Configuracion.RatioAreaC3_4_MediaElipses = 3.34;
Configuracion.RatioC13_4_Area_C11Area = 6.44;
Configuracion.ValorAnguloC3_E3_C1_C2 = 99;
Configuracion.ValorAnguloC4_E3_C1_C2 = 80;
%Proporcionalidad entre la distancia de la elipse 1_3 con el círculo 1_3
Configuracion.RatioDistanciaE1_3_C1_3 = 51;
%Proporcionalidad entre la distancia de la elipse 1_3 con el círculo 1_4
Configuracion.RatioDistanciaE1_3_C1_4 = 51;

%-----
% 1) Lectura de la imagen original
%-----
%Comprobamos si está activada la opción para mostrar el tiempo de ejecución
if (Configuracion.MostrarTiempoCPU)
    TiempoCPU = cputime;
end
%Llevamos a cabo la lectura de la imagen de entrada
Ioriginal = imread(Configuracion.ImagenEntrada);
Ioriginal = imrotate(Ioriginal, Datos.SimulacionRotacionImagenOriginal);
InfoImagen = imfinfo(Configuracion.ImagenEntrada);

%Mostramos información por la consola si está activada la opción
if (Configuracion.VerboseConsola)
    disp('Proceso de lectura de la imagen original efectuada');
    cadena = sprintf(' -> La imagen %s tiene %d píxeles ancho (columnas) y %d píxeles de largo (filas)' ...
        , Configuracion.ImagenEntrada, InfoImagen.Width, InfoImagen.Height);
    disp(cadena);
end

%Comprobamos si la resolución se ajusta a la determinada en configuración
Ioriginal = EscalarImagen(Ioriginal, Configuracion.ImagenEntrada, ...
    Configuracion.FilaResolucionDeseada, InfoImagen.Height);

%Mostramos la imagen que acabamos de leer si se activa la opción
Configuracion.NumeroFigura = VisualizarImagen(Ioriginal, ...
    Configuracion.MostrarImagenOriginal, Configuracion.NumeroFigura);

```

```

%-----
% 2) Transformación del espacio RGB al HSV para obtener la intensidad
%-----
I = rgb2hsv(Ioriginal);
Intensidad = I(:,:,3);
%Limpiamos variables para evitar problemas de falta de memoria en el HEAP
clear I; clear Ioriginal;

%Mostramos información por la consola si está activada la opción
EscribirConsola(Configuracion.VerboseConsola, ...
    'Conversión RGB a HSV para obtener la intensidad efectuado');

%Mostramos la imagen que acabamos de generar si se activa la opción
Configuracion.NumeroFigura = VisualizarImagen(Intensidad, ...
    Configuracion.MostrarImagenIntensidad, Configuracion.NumeroFigura);

%-----
% 3) Binarización de la imagen de intensidad
%-----
%Binarizamos la imagen según la configuración empleada
[IBinaria, Configuracion.TipoUmbral]= ...
BinarizarImagen(Configuracion.TipoUmbral, Intensidad, Configuracion.Umbral);
clear Intensidad;

%Obtenemos el número de Filas y Columnas de la imagen binaria;
[Fila, Columna] = size(IBinaria);

%Mostramos información por la consola si está activada la opción
EscribirConsola(Configuracion.VerboseConsola, ...
    'Proceso de binarización de la imagen efectuado');

%Mostramos la imagen que acabamos de generar si se activa la opción
Configuracion.NumeroFigura = VisualizarImagen(IBinaria, ...
    Configuracion.MostrarImagenBinaria, Configuracion.NumeroFigura);

%-----
% 4) Limpiar la imagen binarizada mediante operaciones morfológicas
%-----
IBinariaLimpia = bwmorph(IBinaria,'close',1);

%Mostramos información por la consola si está activada la opción
EscribirConsola(Configuracion.VerboseConsola, ...
    'Proceso de limpieza de la imagen binaria efectuado');

%Mostramos la imagen que acabamos de generar si se activa la opción
Configuracion.NumeroFigura = VisualizarImagen(IBinariaLimpia, ...
    Configuracion.MostrarImagenBinariaLimpia, Configuracion.NumeroFigura);

%-----
% 5) Etiquetado de regiones conexas de la imagen
%-----
% L es la imagen de regiones
% NUM es el número de regiones encontradas
[L,NUM] = bwlabel(IBinariaLimpia);

% Extracción de las propiedades de las regiones identificadas
propiedades = regionprops(L, 'Centroid','Area','EquivDiameter',...
    'MajorAxisLength','MinorAxisLength','Orientation','Eccentricity',...
    'EulerNumber');

%Mostramos información por la consola si está activada la opción
EscribirConsola(Configuracion.VerboseConsola, ...
    'Proceso de etiquetado de regiones conexas efectuado');

%Mostramos la imagen que acabamos de leer si se activa la opción
if (Configuracion.MostrarRegionesConexas)
    figure(Configuracion.NumeroFigura);
    imagesc(L); impixelinfo;
    Configuracion.NumeroFigura = Configuracion.NumeroFigura + 1;
end

%-----
% 6) Extraemos datos de elipses a partir de propiedades de las regiones
%-----

```

```

Elipse = ObtenerElipses(propiedades, Elipse, ...
Configuracion.FiltradoAreaMinimaElipse, ...
Configuracion.FiltradoAreaMaximaElipse, ...
Configuracion.FiltroExcentricidadMinElipse, ...
Configuracion.FiltroExcentricidadMaxElipse, NUM, Distancia, ...
Configuracion.DeteccionUmbralAreaElipse, ...
Configuracion.DeteccionUmbralExcentricidadElipse, ...
Configuracion.ProporcionalidadDistanciaE1_2_E1_3, ...
Configuracion.ProporcionalidadDistanciaE2_3_E1_3, ...
Configuracion.ProporcionalidadDistanciaE2_3_E1_2, ...
Configuracion.MaximoValorAutoUmbralElipses, Configuracion.TipoDistancia);

if (length(Elipse) < Configuracion.NumeroElipses)
    EscribirConsola(Configuracion.VerboseConsola, ...
        'Elipses no detectadas. Identificación incorrecta');
    %Llamamos a la función GestionAutoUmbral para que compruebe con las dos
    %estrategias de umbralización en el caso que sea necesario
    GestionAutoUmbral (0, Configuracion.ImagenEntrada, ...
        Configuracion.FilaResolucionDeseada, Configuracion.TipoUmbral, ...
        Configuracion.ProbabilidadMinimaAutoumbralizacion, Configuracion.Autoumbralizacion);
    return
end

%-----
% 7) Dibujado del contorno de las elipse y sus centros correspondiente
%-----
%Si está habilitada la opción de dibujar elipses marcamos su contorno
if (Configuracion.DibujarElipses)
    Configuracion.NumeroFigura = DibujarElipse(IBinariaLimpia, ...
        Configuracion.NumeroFigura, Elipse, Configuracion.ColorElipses, ...
        Configuracion.SimboloCentroidesElipses, ...
        Configuracion.DibujarCentroidesElipses, false);
end

%-----
% 8) Análisis orientación a partir de la distancia euclídea de sus centros
%-----
%Calculamos las distancias para identificar la elipse central, ya que se
%sabe de antemano por el análisis preliminar de la figura que la distancia
%que hay entre las elipses laterales es mayor que la distancia de la elipse
%central a cualquiera de las otras dos elipses.
%Distancia entre la elipse 1 y 2
[DistanciaEuclídeaE1_E2, DistanciaEuclídeaE1_E3, DistanciaEuclídeaE2_E3]...
= DistanciasElipses(Elipse, Configuracion.TipoDistancia);

%Identificamos cada una de las elipses según la distancia entre los centros
%ya que se sabe que las elipses laterales en vista frontal tienen una mayor
%distancia entre ellas que la central con cualquiera de las otras dos
if (DistanciaEuclídeaE1_E2 > DistanciaEuclídeaE1_E3 ...
    && DistanciaEuclídeaE1_E2 > DistanciaEuclídeaE2_E3)
    %La elipse 3 es la central según la perspectiva frontal
    Elipse(3).ElipseCentral = true;
    if (Elipse(3).Orientacion >= 0)
        AnguloARotar = -abs(Elipse(3).Orientacion);
    else
        AnguloARotar = abs(Elipse(3).Orientacion);
    end
    if (Elipse(3).CentroideEjeY < Elipse(1).CentroideEjeY)
        %Comprobamos si la distancia es negativa o no al darle medio giro
        if (AnguloARotar < 0)
            AnguloARotar = AnguloARotar - 180;
        else
            AnguloARotar = AnguloARotar + 180;
        end
    end
elseif (DistanciaEuclídeaE2_E3 > DistanciaEuclídeaE1_E3 ...
    && DistanciaEuclídeaE2_E3 > DistanciaEuclídeaE1_E2)
    %La elipse 1 es la central según la perspectiva frontal
    Elipse(1).ElipseCentral = true;
    if (Elipse(1).Orientacion >= 0)
        AnguloARotar = -abs(Elipse(1).Orientacion);
    else
        AnguloARotar = abs(Elipse(1).Orientacion);
    end
end

```

```

if (Elipse(1).CentroideEjeY < Elipse(2).CentroideEjeY)
    %Comprobamos si la distancia es negativa o no al darle medio giro
    if (AnguloARotar < 0)
        AnguloARotar = AnguloARotar - 180;
    else
        AnguloARotar = AnguloARotar + 180;
    end
end
else
    %La elipse 2 es la central según la perspectiva frontal
    Elipse(2).ElipseCentral = true;
    if (Elipse(2).Orientacion >= 0)
        AnguloARotar = -abs(Elipse(2).Orientacion);
    else
        AnguloARotar = abs(Elipse(2).Orientacion);
    end
    if (Elipse(2).CentroideEjeY < Elipse(1).CentroideEjeY)
        %Comprobamos si la distancia es negativa o no al darle medio giro
        if (AnguloARotar < 0)
            AnguloARotar = AnguloARotar - 180;
        else
            AnguloARotar = AnguloARotar + 180;
        end
    end
end
end

%-----
% 9) Orientación de la imagen a partir de las posiciones relativas
%-----
%Volvemos a leer la imagen original y la rotamos para orientarla (centrar)
loriginal = imread(Configuracion.ImagenEntrada);
loriginal = imrotate(loriginal, AnguloARotar + ...
    Datos.SimulacionRotacionImagenOriginal);

%Mostramos la imagen que acabamos de generar si se activa la opción
Configuracion.NumeroFigura = VisualizarImagen(loriginal, ...
    Configuracion.MostrarImagenOriginalOrientada, Configuracion.NumeroFigura);

%Comprobamos si la resolución se ajusta a la determinada en configuración
loriginal = EscalarImagen(loriginal, Configuracion.ImagenEntrada, ...
    Configuracion.FilaResolucionDeseada, InfoImagen.Height);

%Binarizamos la imagen orientada y limpiamos variables para liberar memoria
I = rgb2hsv(loriginal); clear loriginal;
Intensidad = I(:, :, 3); clear I;

% Binarizamos la imagen según la configuración empleada
[IBinariaRotada, Configuracion.TipoUmbral] = ...
    BinarizarImagen(Configuracion.TipoUmbral, Intensidad, Configuracion.Umbral);

%Limpiamos la imagen con el operador morfológico close
IBinariaLimpiaRotada = bwmorph(IBinariaRotada, 'close', 1);

%Etiquetamos las regiones conexas de la imagen rotada
[L, NUM] = bwlabel(IBinariaLimpiaRotada);
propiedades = regionprops(L, 'Centroid', 'Area', 'EquivDiameter', ...
    'MajorAxisLength', 'MinorAxisLength', 'Orientation', 'Eccentricity', ...
    'EulerNumber');

%Obtenemos las propiedades de las elipses de las regiones conexas
Elipse = ObtenerElipses(propiedades, Elipse, ...
    Configuracion.FiltradoAreaMinimaElipse, ...
    Configuracion.FiltradoAreaMaximaElipse, ...
    Configuracion.FiltroExcentricidadMinElipse, ...
    Configuracion.FiltroExcentricidadMaxElipse, NUM, Distancia, ...
    Configuracion.DeteccionUmbralAreaElipse, ...
    Configuracion.DeteccionUmbralExcentricidadElipse, ...
    Configuracion.ProporcionalidadDistanciaE1_2_E1_3, ...
    Configuracion.ProporcionalidadDistanciaE2_3_E1_3, ...
    Configuracion.ProporcionalidadDistanciaE2_3_E1_2, ...
    Configuracion.MaximoValorAutoUmbralElipses, Configuracion.TipoDistancia);

%Comprobamos si se han detectado la elipses. En caso contrario abortamos
if (length(Elipse) < Configuracion.NumeroElipses)

```

```

EscribirConsola(Configuracion.VerboseConsola, ...
    'Elipses no detectadas. Identificación incorrecta');
%Llamamos a la función GestionAutoUmbral para que compruebe con las dos
%estrategias de umbralización en el caso que sea necesario
GestionAutoUmbral (0, Configuracion.ImagenEntrada, ...
Configuracion.FilaResolucionDeseada, Configuracion.TipoUmbral, ...
Configuracion.ProbabilidadMinimaAutoumbralizacion, Configuracion.Autoumbralizacion);
return
end

AnguloARotar = 0;

%Calculamos las distancias entre los centros de las elipses
[DistanciaEuclideaE1_E2, DistanciaEuclideaE1_E3, DistanciaEuclideaE2_E3]...
= DistanciasElipses(Elipse, Configuracion.TipoDistancia);

if (DistanciaEuclideaE1_E2 > DistanciaEuclideaE1_E3 ...
&& DistanciaEuclideaE1_E2 > DistanciaEuclideaE2_E3)
    %La elipse 3 es la central según la perspectiva frontal
    Elipse(3).NumeroDeElipse = 2;
    if (Elipse(3).CentroideEjeY < Elipse(1).CentroideEjeY)
        AnguloARotar = 180;
    end
elseif (DistanciaEuclideaE2_E3 > DistanciaEuclideaE1_E3 ...
&& DistanciaEuclideaE2_E3 > DistanciaEuclideaE1_E2)
    %La elipse 1 es la central según la perspectiva frontal
    Elipse(1).NumeroDeElipse = 2;
    if (Elipse(1).CentroideEjeY < Elipse(2).CentroideEjeY)
        AnguloARotar = 180;
    end
else
    %La elipse 2 es la central según la perspectiva frontal
    Elipse(2).NumeroDeElipse = 2;
    if (Elipse(2).CentroideEjeY < Elipse(1).CentroideEjeY)
        AnguloARotar = 180;
    end
end

%Comprobamos si se de el caso en que sea necesaria una rotación adicional
%como consecuencia de que la orientación en el regionprops viene dada en
%el ángulo con intervalo (-90 90)
if (AnguloARotar ~= 0)
    %Volvemos a rotar la imagen para dejarla completamente orientada
    IBinariaRotada = imrotate (IBinariaRotada, AnguloARotar);
    IBinariaLimpiaRotada = imrotate (IBinariaLimpiaRotada, AnguloARotar);

    %Etiquetamos las regiones de la imagen que ya está totalmente orientada
    [L_NUM] = bwlabel (IBinariaLimpiaRotada);
    propiedades = regionprops (L, 'Centroid', 'Area', 'EquivDiameter', ...
'MajorAxisLength', 'MinorAxisLength', 'Orientation', 'Eccentricity', ...
'EulerNumber');

    %Volvemos a extraer propiedades una vez la imagen está totalmente orientada
    Elipse = ObtenerElipses(propiedades, Elipse, ...
Configuracion.FiltradoAreaMinimaElipse, ...
Configuracion.FiltradoAreaMaximaElipse, ...
Configuracion.FiltroExcentricidadMinElipse, ...
Configuracion.FiltroExcentricidadMaxElipse, NUM, Distancia, ...
Configuracion.DeteccionUmbralAreaElipse, ...
Configuracion.DeteccionUmbralExcentricidadElipse, ...
Configuracion.ProporcionalidadDistanciaE1_2_E1_3, ...
Configuracion.ProporcionalidadDistanciaE2_3_E1_3, ...
Configuracion.ProporcionalidadDistanciaE2_3_E1_2, ...
Configuracion.MaximoValorAutoUmbralElipses, Configuracion.TipoDistancia);

    %Comprobamos si se han detectado la elipses. En caso contrario abortamos
    if (length(Elipse) < Configuracion.NumeroElipses)
        EscribirConsola(Configuracion.VerboseConsola, ...
            'Elipses no detectadas. Identificación incorrecta');
        %Llamamos a la función GestionAutoUmbral para que compruebe con las
        %dos estrategias de umbralización en el caso que sea necesario
        GestionAutoUmbral (0, Configuracion.ImagenEntrada, ...
Configuracion.FilaResolucionDeseada, Configuracion.TipoUmbral, ...
Configuracion.ProbabilidadMinimaAutoumbralizacion, Configuracion.Autoumbralizacion);
    end
end

```

```

return
end

%Calculamos las distancias entre los centros de las elipses
[DistanciaEuclideaE1_E2, DistanciaEuclideaE1_E3, DistanciaEuclideaE2_E3]...
= DistanciasElipses(Elipse, Configuracion.TipoDistancia);

%Localización de la elipse central a partir de las distancias euclideas
if (DistanciaEuclideaE1_E2 > DistanciaEuclideaE1_E3 ...
&& DistanciaEuclideaE1_E2 > DistanciaEuclideaE2_E3)
    %La elipse 3 es la central según la perspectiva frontal
    Elipse(3).NumeroDeElipse = 2;
    Datos.NumRegionElipseCentral = Elipse(3).NumeroRegion;
elseif (DistanciaEuclideaE2_E3 > DistanciaEuclideaE1_E3 ...
&& DistanciaEuclideaE2_E3 > DistanciaEuclideaE1_E2)
    %La elipse 1 es la central según la perspectiva frontal
    Elipse(1).NumeroDeElipse = 2;
    Datos.NumRegionElipseCentral = Elipse(1).NumeroRegion;
else
    %La elipse 2 es la central según la perspectiva frontal
    Elipse(2).NumeroDeElipse = 2;
    Datos.NumRegionElipseCentral = Elipse(2).NumeroRegion;
end
end

%Mostramos las imágenes que acabamos de generar si se activa la opción
Configuracion.NumeroFigura = VisualizarImagen(IBinariaRotada, ...
Configuracion.MostrarImagenBinariaOrientada, Configuracion.NumeroFigura);

Configuracion.NumeroFigura = VisualizarImagen(IBinariaLimpiaRotada, ...
Configuracion.MostrarImagenBinariaLimpiaOrientada, Configuracion.NumeroFigura);

%Mostramos las regiones conexas si se activa la opción adecuada
if (Configuracion.MostrarRegionesConexasOrientada)
    figure(Configuracion.NumeroFigura);
    imagesc(L); impixelinfo;
    Configuracion.NumeroFigura = Configuracion.NumeroFigura + 1;
end

EscribirConsola(Configuracion.VerboseConsola, ...
'Proceso de orientacion de la imagen efectuado');

%-----
% 10) Identificación elipses izq. central y dch. según perspectiva frontal
%-----
%Recorremos todas las regiones conexas etiquetadas anteriormente y se
%guardan los datos de aquellas que nos interesan en base fundamentalmente
%a su excentricidad. También se puede utilizar el área y la orientación.
for m = 1:length(Elipse);
    %Si el área de la región, y la excentricidad están dentro del umbral
    %definido entonces comenzamos a determinar cuales será las elipses 1,
    %2 y 3, es decir, elipse izquierda, central y derecha.
    if(Elipse(m).Area >= Configuracion.FiltradoAreaMinimaElipse ...
&& Elipse(m).Area <= Configuracion.FiltradoAreaMaximaElipse ...
&& Elipse(m).Excentricidad >= Configuracion.FiltroExcentricidadMinElipse ...
&& Elipse(m).Excentricidad <= Configuracion.FiltroExcentricidadMaxElipse)

        %Obtenemos las coordenadas X e Y del punto central de cada elipse
        centroideEjeX = Elipse(m).CentroideEjeX;
        centroideEjeY = Elipse(m).CentroideEjeY;

        %Obtenemos los datos de la elipse que está en uno de los lados
        if (E1.CentroideEjeX > centroideEjeX)
            E1 = AsignarFigura(centroideEjeX, centroideEjeY,...
            Elipse(m).NumeroRegion, Elipse(m).Orientacion, ...
            Elipse(m).Excentricidad, Elipse(m).Area);
        end

        %Obtenemos los datos de la elipse que está en el otro lado
        if (E3.CentroideEjeX < centroideEjeX)
            E3 = AsignarFigura(centroideEjeX, centroideEjeY,...
            Elipse(m).NumeroRegion, Elipse(m).Orientacion, ...
            Elipse(m).Excentricidad, Elipse(m).Area);
        end
    end
end
end

```

```

%Obtenemos los datos de la elipse 2, es decir, aquella elipse que se
%marcara en el análisis de orientación como central
if (Elipse(m).NumeroDeElipse == 2)
    %Almacenamos los datos de la elipse central. Posteriormente se
    %harán más comprobaciones para ver si está bien alineada
    E2 = AsignarFigura(Elipse(m).CentroideEjeX, ...
    Elipse(m).CentroideEjeY, Elipse(m).NumeroRegion, ...
    Elipse(m).Orientacion, Elipse(m).Excentricidad, Elipse(m).Area);
end
end
%Obtenemos la media aritmética del área de las tres elipses
Datos.MedidaAreaElipses = (E2.Area + E1.Area + E3.Area) / 3;

%Calculamos las distancias una vez se han etiquetado las elipses
%Distancia entre la elipse 1 y 2
DistanciaEuclideaE1_E2 = CalcularDistancia(E1.CentroideEjeX, ...
E2.CentroideEjeX, E1.CentroideEjeY, E2.CentroideEjeY, ...
Configuracion.TipoDistancia);

%Distancia entre la elipse 1 y 3
DistanciaEuclideaE1_E3 = CalcularDistancia(E1.CentroideEjeX, ...
E3.CentroideEjeX, E1.CentroideEjeY, E3.CentroideEjeY, ...
Configuracion.TipoDistancia);

%Distancia entre la elipse 2 y 3
DistanciaEuclideaE2_E3 = CalcularDistancia(E2.CentroideEjeX, ...
E3.CentroideEjeX, E2.CentroideEjeY, E3.CentroideEjeY, ...
Configuracion.TipoDistancia);

%Obtenemos las coordenadas cartesianas del punto central de la figura
%según la distancia y la posición de las elipses 1 y 3.
Datos.PuntoCentralEjeY = (E1.CentroideEjeY + E3.CentroideEjeY) * 0.5;
Datos.PuntoCentralEjeX = E1.CentroideEjeX + ...
(DistanciaEuclideaE1_E3 * 0.5);
end

%-----
% 11) Dibujado del contorno de las elipse y sus centros correspondientes
%-----
%Si está habilitada la opción de dibujar círculos entonces habilitamos el
%modo hold para dibujar elipses y círculos en la misma figura
if (Configuracion.DibujarCirculos)
    Espera = true;
else
    Espera = false;
end

%Si está habilitada la opción de dibujar elipses marcamos su contorno
if (Configuracion.DibujarElipsesOrientadas)
    Configuracion.NumeroFigura = DibujarElipse(1BinariaRotada, ...
    Configuracion.NumeroFigura, Elipse, Configuracion.ColorElipses, ...
    Configuracion.SimboloCentroidesElipses, ...
    Configuracion.DibujarCentroidesElipses, ...
    Espera);
end

%-----
% 12) Búsqueda y reconocimiento de los 4 círculos interiores de la figura
%-----
Circulo = ObtenerCirculos(propiedades, Circulo, ...
Configuracion.FiltradoAreaMinimaCirculo, ...
Configuracion.FiltradoAreaMaximaCirculo, ...
Configuracion.FiltroExcentricidadMinCirculo, ...
Configuracion.FiltroExcentricidadMaxCirculo, NUM);

%Identificamos el círculo 1 y 2
[C1 C2 Circulo] = IdentificarC1_C2(Datos, Circulo, ...
Configuracion.RatioAreaC1MediaElipses, ...
Configuracion.RatioC12Area_C11Area, ...
Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_C1, ...
Configuracion.MaximoValorAutoUmbralCirculos, Configuracion.TipoDistancia,...
Configuracion.NumMaxCirculosEstrategia2);

```

```

EscribirConsola(Configuracion.VerboseConsola, ...
    'Proceso de identificación de círculos C1 y C2 realizado');

%Eliminamos el círculo1 y el círculo2 del array
Circulo = EliminarElementoArray(Circulo, C1, 'NumeroRegion');
Circulo = EliminarElementoArray(Circulo, C2, 'NumeroRegion');

%Identificamos el círculo 3 y 4
[C3 C4] = IdentificarC3_C4(Circulo, ...
    Configuracion.RatioCl3_4_Area_C11Area, DistanciaEuclideaE1_E3, ...
    Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_C1, C1, ...
    Configuracion.RatioDistanciaE1_3_C1_3, Configuracion.VerboseConsola, ...
    Configuracion.ValorAnguloC3_E3_C1_C2, Configuracion.ValorAnguloC4_E3_C1_C2, ...
    Configuracion.MaximoValorAutoUmbralCirculos, Configuracion.TipoDistancia, ...
    Configuracion.NumMaxCirculosEstrategia2);

EscribirConsola(Configuracion.VerboseConsola, ...
    'Proceso de identificación de círculos C3 y C4 realizado');

%-----
% 13) Dibujado del contorno de los 4 círculos interiores de la figura
%-----
%Dibujamos solo los 4 círculos identificados en el proceso anterior;
for m=length(Circulo):-1:1
    Circulo(m) = [];
end
Circulo(1) = C1; Circulo(2) = C2;
Circulo(3) = C3; Circulo(4) = C4;

%Si está habilitada la opción de dibujar elipses las dibujamos en la misma
%ventana. En caso contrario la dibujamos en una ventana nueva
if (Configuracion.DibujarElipsesOrientadas && Configuracion.DibujarCirculos)
    Configuracion.NumeroFigura = Configuracion.NumeroFigura - 1;
end

%Si está habilitada la opción de dibujar círculos mostramos su contorno
if (Configuracion.DibujarCirculos)
    Configuracion.NumeroFigura = DibujarCirculo(IBinariaRotada, ...
        Configuracion.NumeroFigura, Circulo, Configuracion.ColorCirculos, ...
        Configuracion.SimboloCentroidesCirculos, ...
        Configuracion.DibujarCentroidesCirculos, ...
        Configuracion.DibujarElipsesOrientadas, ...
        Configuracion.UmbralDibujoCirculo, Configuracion.FactorRadio);
end

%-----
% 14) APC1: Análisis de las posiciones de los centros de las elipses
%-----
%Dado que sabemos como tienen que estar alineadas las elipses, utilizamos
%esta razón geométrica para llevar a cabo el análisis parcial comparativo.
%Para ello, obtenemos el error máximo en píxeles que se permite, para poder
%determinar el punto del eje Y correspondiente al centroide de la elipse
%que está en el otro lado
UmbralErrorEjeY = abs(E1.CentroideEjeX - E3.CentroideEjeX) ...
    * sin(Configuracion.ErrorAlineacionEjeY_C1_C3);

%También determinados el máximo error con relación al eje X en donde se
%encuentra el centroide de la elipse central, que tiene que en teoría
%que ser la mitad de la distancia euclídea que hay entre las dos elipses
%que están a ambos lados de la figura (vista esta de forma frontal)
UmbralErrorEjeX = Configuracion.ErrorAlineacionEjeX_C2 * ...
    (CalcularDistancia(E1.CentroideEjeX, E3.CentroideEjeX, E1.CentroideEjeY, ...
    E3.CentroideEjeY, Configuracion.TipoDistancia)) / 100;

%Ahora trataremos de determinar el punto X en el que debe estar la elipse
%cuyo centro está en el punto medio que hay entre las 2 elipses laterales.
PuntoX_E2 = E1.CentroideEjeX + (E3.CentroideEjeX - E1.CentroideEjeX) / 2;

%Si se cumplen las condiciones damos por alineados los centros
if (E1.CentroideEjeY >= E3.CentroideEjeY - UmbralErrorEjeY ...
    && E1.CentroideEjeY <= E3.CentroideEjeY + UmbralErrorEjeY ...
    && E2.CentroideEjeX >= PuntoX_E2 - UmbralErrorEjeX ...
    && E2.CentroideEjeX <= PuntoX_E2 + UmbralErrorEjeX)
    %Proceso de identificación correcto

```

```

Resultado.PosicionesCentrosElipses = true;
Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
EscribirConsola(Configuracion.VerboseConsola, ...
'Posiciones de centros de elipses correctas');

%En caso contrario damos por no localizada la plataforma de aproximación
else
  %Proceso de identificación incorrecto
  Resultado.PosicionesCentrosElipses = false;
  EscribirConsola(Configuracion.VerboseConsola, ...
'Posiciones de centros de elipses incorrectas');
end

%-----
% 15) APC2: Análisis de los ángulos de los centros de las elipses
%-----
%Como se sabe de antemano que el ángulo entre las elipses laterales tiene
%ronda el valor 0, y que el ángulo entre la elipse central y las otras dos
%tiene un valor cercano a 45% podemos utilizarlo como un criterio más para
%identificar la figura.

%Angulo entre elipses laterales
%Obtenemos el valor absoluto porque el ángulo será negativo dado que Matlab
%representa el Eje Y con los valores altos abajo y los menores arriba.
%Utilizamos la función CalcularAngulo en modo 180 porque si se calcula el
%ángulo en el intervalo [0,360] y estimamos sobre 0, para valores cercanos
%a 360 que hayan dado casi una vuelta completa (359, 358, etcétera) la
%comparativa fallaría.
Datos.Angulo_E1_E3= abs(CalcularAngulo(E3.CentroideEjeX, ...
-1*(E3.CentroideEjeY), E1.CentroideEjeX, -1*(E1.CentroideEjeY), false));

%Angulo entre la elipse central y la izquierda.
%Multiplicamos por -1 el valor del Eje Y porque en MATLAB dicho eje se
%representa con los valores altos abajo y los menores arriba
Datos.Angulo_E2_E1 = CalcularAngulo(E1.CentroideEjeX, ...
-1*(E1.CentroideEjeY), E2.CentroideEjeX, -1*(E2.CentroideEjeY), true);

%Angulo entre la elipse central y la derecha
%Multiplicamos por -1 el valor del Eje Y porque en MATLAB dicho eje se
%representa con los valores altos abajo y los menores arriba
Datos.Angulo_E2_E3 = CalcularAngulo(E3.CentroideEjeX, ...
-1*(E3.CentroideEjeY), E2.CentroideEjeX, -1*(E2.CentroideEjeY), true);

%Si se cumplen las condiciones damos por correctos los ángulos
if (Datos.Angulo_E1_E3 >= ...
  Configuracion.ValorAnguloE1_3 - ...
  Configuracion.GradosUmbralErrorOrientacionAngulosElipses ...
&& Datos.Angulo_E1_E3 <= ...
  Configuracion.ValorAnguloE1_3 + ...
  Configuracion.GradosUmbralErrorOrientacionAngulosElipses ...
&& (Datos.Angulo_E2_E3 >= ...
  Configuracion.ValorAnguloE2_3 * ...
(1 - (Configuracion.GradosUmbralErrorOrientacionAngulosElipses) / 100))...
&& (Datos.Angulo_E2_E3 <= ...
  Configuracion.ValorAnguloE2_3 * ...
(1 + (Configuracion.GradosUmbralErrorOrientacionAngulosElipses) / 100))...
&& (Datos.Angulo_E2_E1 >= ...
  Configuracion.ValorAnguloE1_2 * ...
(1 - (Configuracion.GradosUmbralErrorOrientacionAngulosElipses) / 100))...
&& (Datos.Angulo_E2_E1 <= ...
  Configuracion.ValorAnguloE1_2 * ...
(1 + (Configuracion.GradosUmbralErrorOrientacionAngulosElipses) / 100)))
  % Identificación de ángulos correcta
  Resultado.AngulosElipses = true;
  Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
  EscribirConsola(Configuracion.VerboseConsola, ...
'Ángulos de elipses correctos');
else
  Resultado.AngulosElipses = false;
  EscribirConsola(Configuracion.VerboseConsola, ...
'Ángulos de elipses incorrectos');
end

%-----

```

```

% 16) APC3: Análisis de distancias relativas a los centros de las elipses
%-----
%A partir de las distancias de los centros de las elipses se calcula la
%proporcionalidad de las distancias de los centros de elipses. Dichas
%distancias tienen que analizarse en forma de ratios, dada la imposibilidad
%de dar valores constantes porque la distancia depende esencialmente del
%zoom y la distancia a la que haya sido tomada la imagen desde el UAV.
if (((DistanciaEuclideaE2_E3 / DistanciaEuclideaE1_E3) * 100) ...
    >= Configuracion.PorporcionalidadDistanciaE2_3_E1_3 * ...
    (1 - (Configuracion.PorcentajeUmbralErrorDistancia_Elipces / 100))) ...
    && (((DistanciaEuclideaE2_E3 / DistanciaEuclideaE1_E3) * 100) ...
    <= Configuracion.PorporcionalidadDistanciaE2_3_E1_3 * ...
    (1 + (Configuracion.PorcentajeUmbralErrorDistancia_Elipces / 100))) ...
    && (((DistanciaEuclideaE1_E2 / DistanciaEuclideaE1_E3) * 100) ...
    >= Configuracion.PorporcionalidadDistanciaE1_2_E1_3 * ...
    (1 - (Configuracion.PorcentajeUmbralErrorDistancia_Elipces / 100))) ...
    && (((DistanciaEuclideaE1_E2 / DistanciaEuclideaE1_E3) * 100) ...
    <= Configuracion.PorporcionalidadDistanciaE1_2_E1_3 * ...
    (1 + (Configuracion.PorcentajeUmbralErrorDistancia_Elipces / 100)))
    %Si se cumplen ambas condiciones entonces asignamos true
    Resultado.PorporcionalidadDistancias = true;
    Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
    'Proporcionalidad de distancias entre elipses correcta');
else
    %Si no se cumplen ambas condiciones entonces asignamos false
    Resultado.PorporcionalidadDistancias = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
    'Proporcionalidad de distancias entre elipses incorrecta');
end

%-----
% 17) APC4: Análisis de la similitud del área de las elipses 1, 2 y 3
%-----
%Comprobamos si el área de las tres elipses está dentro del umbral de
%detección, lo que quiere decir que su nivel de analogía será suficiente.
if (SimilitudElipses(E1.Area, E2.Area, ...
    E3.Area, Configuracion.PorcentajeUmbralDiferenciaAreaElipse))
    %Si se cumplan todas las condiciones asignamos true
    Resultado.SimilitudAreaElipses = true;
    Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
    'Similitud del área de Elipses correcta');
else
    %En caso contrario asignamos false
    Resultado.SimilitudAreaElipses = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
    'Similitud del área de las elipses incorrecta');
end

%-----
% 18) APC5: Análisis de la similitud de la excentricidad de las elipses
%-----
%Comprobamos si la excentricidad de las 3 elipses está dentro del umbral de
%detección, lo que quiere decir que su nivel de analogía será suficiente.
if (SimilitudElipses(E1.Excentricidad, ...
    E2.Excentricidad, E3.Excentricidad, ...
    Configuracion.PorcentajeUmbralDiferenciaExcentricidadElipse))
    %Si se cumplan todas las condiciones asignamos true
    Resultado.SimilitudExcentricidadElipses = true;
    Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
    'Similitud de la excentricidad de elipses correcta');
else
    %En caso contrario asignamos false
    Resultado.SimilitudExcentricidadElipses = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
    'Similitud de la excentricidad de elipses incorrecta');
end

%-----
% 19) APC6: Análisis del Círculo1 por separado respecto a las elipses
%-----
%Analizamos el ángulo, la distancia y la proporción del área respecto a

```

```

%las elipses identificadas anteriormente para reconocer un círculo. Este
% mismo procedimiento será aplicado a los cuatro círculos por separado.

%Calculamos el ángulo definido entre el círculo1 y la elipse3. Utilizamos
%este para que su valor estimado sea 180 y no 0, así se pueda aplicar
%mejor el umbral basado en porcentaje.
AnguloC1 = CalcularAngulo(C1.CentroideEjeX, -1*(C1.CentroideEjeY), ...
    E3.CentroideEjeX, -1*(E3.CentroideEjeY), true);

%Calculamos la distancia euclídea que hay desde el círculo1 hasta la elipse
%derecha que tiene que ser la misma que la distancia euclídea entre la
%elipse 1 y 3 dividida entre dos
DistanciaC1 = CalcularDistancia(C1.CentroideEjeX, E3.CentroideEjeX, ...
    C1.CentroideEjeY, E3.CentroideEjeY, Configuracion.TipoDistancia);

%Calculamos el ratio en tanto por ciento respecto a la media de las elipses
RatioAreaC1 = (C1.Area / Datos.MediaAreaElipses) * 100;

if(AnalizarCirculo(AnguloC1, Configuracion.ValorAnguloC1_E3, ...
    Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos, ...
    DistanciaC1, (DistanciaEuclideaE1_E3 * 0.5), ...
    Configuracion.PorcentajeUmbralErrorDistanciaCirculos, RatioAreaC1, ...
    Configuracion.RatioAreaC1MediaElipses, ...
    Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_Elipses))
    %Si se cumplan todas las condiciones asignamos true
    Resultado.IdentificacionC1 = true;
    Resultado.TotalAcieros = Resultado.TotalAcieros + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo1 ha sido identificado correctamente');
else
    %En caso contrario asignamos false
    Resultado.IdentificacionC1 = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo1 no ha sido identificado correctamente');
end

%-----
% 20) APC7: Análisis del Círculo2 por separado respecto a las elipses
%-----
%Calculamos el ángulo definido entre el círculo2 y la elipse3. Utilizamos
%este para que su valor estimado sea 180 y no 0, así se pueda aplicar mejor
%el umbral basado en porcentaje.
AnguloC2 = CalcularAngulo(C2.CentroideEjeX, -1*(C2.CentroideEjeY), ...
    E3.CentroideEjeX, -1*(E3.CentroideEjeY), true);

%Calculamos la distancia euclídea que hay desde el círculo2 hasta la elipse
%derecha que tiene que ser la misma que la distancia euclídea entre la
%elipse 1 y 3 dividida entre dos
DistanciaC2 = CalcularDistancia(C2.CentroideEjeX, E3.CentroideEjeX, ...
    C2.CentroideEjeY, E3.CentroideEjeY, Configuracion.TipoDistancia);

%Calculamos el ratio en tanto por ciento respecto a la media de las elipses
RatioAreaC2 = (C2.Area / Datos.MediaAreaElipses) * 100;

if(AnalizarCirculo(AnguloC2, Configuracion.ValorAnguloC2_E3, ...
    Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos, ...
    DistanciaC2, (DistanciaEuclideaE1_E3 * 0.5),...
    Configuracion.PorcentajeUmbralErrorDistanciaCirculos, RatioAreaC2, ...
    Configuracion.RatioAreaC2MediaElipses, ...
    Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_Elipses))
    %Si se cumplan todas las condiciones asignamos true
    Resultado.IdentificacionC2 = true;
    Resultado.TotalAcieros = Resultado.TotalAcieros + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo2 ha sido identificado correctamente');
else
    %En caso contrario asignamos false
    Resultado.IdentificacionC2 = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo2 no ha sido identificado correctamente');
end

%-----
% 21) APC8: Análisis del Círculo3 por separado respecto a las elipses

```

```

%-----
%Calculamos el ángulo definido entre el círculo3 y el punto medio que ha
%quedado definido por la localización de las elipses 1 y 3.
AnguloC3 = CalcularAngulo(C3.CentroideEjeX, -1*(C3.CentroideEjeY), ...
    Datos.PuntoCentralEjeX, -1*(Datos.PuntoCentralEjeY), true);

%Calculamos la distancia euclídea que hay desde el círculo3 hasta el punto
%medio para poder finalmente determinar el ratio con l2, es decir, con la
%distancia euclídea entre la elipse 1 y 2.
DistanciaC3 = ((CalcularDistancia(C3.CentroideEjeX, Datos.PuntoCentralEjeX, ...
C3.CentroideEjeY, Datos.PuntoCentralEjeY, Configuracion.TipoDistancia)) / ...
DistanciaEuclídeaE1_E3) * 100;

%Calculamos el ratio en tanto por ciento respecto a la media de las elipses
RatioAreaC3 = (C3.Area / Datos.MediaAreaElipses) * 100;

if(AnalizarCirculo(AnguloC3, Configuracion.ValorAnguloC3_E3_C1_C2, ...
Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos, DistanciaC3, ...
Configuracion.RatioDistanciaE1_3_C1_3, ...
Configuracion.PorcentajeUmbralErrorDistanciaCirculos, RatioAreaC3, ...
Configuracion.RatioAreaC3_4_MediaElipses, ...
Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_Elipses))
    %Si se cumplan todas las condiciones asignamos true
    Resultado.IdentificacionC3 = true;
    Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo3 ha sido identificado correctamente');
else
    %En caso contrario asignamos false
    Resultado.IdentificacionC3 = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo3 no ha sido identificado correctamente');
end

%-----
% 22) APC9: Análisis del Círculo4 por separado respecto a las elipses
%-----
%Calculamos el ángulo definido entre el círculo4 y el punto medio que ha
%quedado definido por la localización de las elipses 1 y 3.
AnguloC4 = CalcularAngulo(C4.CentroideEjeX, -1*(C4.CentroideEjeY), ...
    Datos.PuntoCentralEjeX, -1*(Datos.PuntoCentralEjeY), true);

%Calculamos la distancia euclídea que hay desde el círculo4 hasta el punto
%medio para poder finalmente determinar el ratio con l2, es decir, con la
%distancia euclídea entre la elipse 1 y 2.
DistanciaC4 = ((CalcularDistancia(C4.CentroideEjeX, Datos.PuntoCentralEjeX, ...
C4.CentroideEjeY, Datos.PuntoCentralEjeY, Configuracion.TipoDistancia)) / ...
DistanciaEuclídeaE1_E3) * 100;

%Calculamos el ratio en tanto por ciento respecto a la media de las elipses
RatioAreaC4 = (C4.Area / Datos.MediaAreaElipses) * 100;

%Determinamos que los valores estén dentro de los rangos de estimación que
%se ha definido a través de una análisis geométrico y los umbrales de error
if(AnalizarCirculo(AnguloC4, Configuracion.ValorAnguloC4_E3_C1_C2, ...
Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos, DistanciaC4, ...
Configuracion.RatioDistanciaE1_3_C1_4, ...
Configuracion.PorcentajeUmbralErrorDistanciaCirculos, RatioAreaC4, ...
Configuracion.RatioAreaC3_4_MediaElipses, ...
Configuracion.PorcentajeUmbralDiferenciaAreaCirculos_Elipses))
    %Si se cumplan todas las condiciones asignamos true
    Resultado.IdentificacionC4 = true;
    Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo4 ha sido identificado correctamente');
else
    %En caso contrario asignamos false
    Resultado.IdentificacionC4 = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'El Círculo4 no ha sido identificado correctamente');
end

%-----

```

```

% 23) APC10: Análisis de los círculos en conjunto para verificar coherencia
%-----
%Primero calculamos los ángulos forman los círculos entre si
AnguloC3_C1 = CalcularAngulo(C3.CentroideEjeX, -1*(C3.CentroideEjeY), ...
    C1.CentroideEjeX, -1*(C1.CentroideEjeY), true);
AnguloC3_C2 = CalcularAngulo(C3.CentroideEjeX, -1*(C3.CentroideEjeY), ...
    C2.CentroideEjeX, -1*(C2.CentroideEjeY), true);
AnguloC4_C1 = CalcularAngulo(C4.CentroideEjeX, -1*(C4.CentroideEjeY), ...
    C1.CentroideEjeX, -1*(C1.CentroideEjeY), true);
AnguloC4_C2 = CalcularAngulo(C4.CentroideEjeX, -1*(C4.CentroideEjeY), ...
    C2.CentroideEjeX, -1*(C2.CentroideEjeY), true);

%Cálculos de los ratios relativos a las distancias euclídeas entre círculos
DistanciaC3_C1 = (((CalcularDistancia(C3.CentroideEjeX, C1.CentroideEjeX, ...
    C3.CentroideEjeY, C1.CentroideEjeY, Configuracion.TipoDistancia)) / ...
    DistanciaEuclideaE1_E3)) * 100;

DistanciaC3_C2 = (((CalcularDistancia(C3.CentroideEjeX, C2.CentroideEjeX, ...
    C3.CentroideEjeY, C2.CentroideEjeY, Configuracion.TipoDistancia)) / ...
    DistanciaEuclideaE1_E3)) * 100;

DistanciaC4_C1 = (((CalcularDistancia(C4.CentroideEjeX, C1.CentroideEjeX, ...
    C4.CentroideEjeY, C1.CentroideEjeY, Configuracion.TipoDistancia)) / ...
    DistanciaEuclideaE1_E3)) * 100;

DistanciaC4_C2 = (((CalcularDistancia(C4.CentroideEjeX, C2.CentroideEjeX, ...
    C4.CentroideEjeY, C2.CentroideEjeY, Configuracion.TipoDistancia)) / ...
    DistanciaEuclideaE1_E3)) * 100;

%Determinamos si los ángulos y los ratios relativos a la distancias
%están dentro de los rangos definidos por la estimación derivada del
%análisis geométrico y los umbrales de error.
if(AnalizarValor(AnguloC3_C1, Configuracion.ValorAnguloC3_E3_C1_C2, ...
    Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos) && ...
    AnalizarValor(AnguloC3_C2, Configuracion.ValorAnguloC3_E3_C1_C2, ...
    Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos) && ...
    AnalizarValor(AnguloC4_C1, Configuracion.ValorAnguloC4_E3_C1_C2, ...
    Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos) && ...
    AnalizarValor(AnguloC4_C2, Configuracion.ValorAnguloC4_E3_C1_C2, ...
    Configuracion.PorcentajeUmbralDiferenciaAnguloCirculos) && ...
    AnalizarValor(DistanciaC3_C1, Configuracion.RatioDistanciaE1_3_C1_3, ...
    Configuracion.PorcentajeUmrErrorDistanciaCirculos) && ...
    AnalizarValor(DistanciaC3_C2, Configuracion.RatioDistanciaE1_3_C1_3, ...
    Configuracion.PorcentajeUmrErrorDistanciaCirculos) && ...
    AnalizarValor(DistanciaC4_C1, Configuracion.RatioDistanciaE1_3_C1_4, ...
    Configuracion.PorcentajeUmrErrorDistanciaCirculos) && ...
    AnalizarValor(DistanciaC4_C2, Configuracion.RatioDistanciaE1_3_C1_4, ...
    Configuracion.PorcentajeUmrErrorDistanciaCirculos))
    %Si se cumplan todas las condiciones asignamos true
    Resultado.IdentificacionCirculos = true;
    Resultado.TotalAciertos = Resultado.TotalAciertos + 1;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'Análisis de vinculos entre círculos correctos');
else
    %En caso contrario asignamos false
    Resultado.IdentificacionCirculos = false;
    EscribirConsola(Configuracion.VerboseConsola, ...
        'Análisis de vinculos entre círculos incorrectos');
end

%-----
% 24) Evaluación final para determinar si la figura ha sido reconocida
%-----
%Resultado final de la identificación de la elipse. Se da por identificada
%si el nivel de probabilidad es mayor o igual a 0.6. Para determinar el
%nivel de probabilidad hacemos una media aritmética de los resultados
%parciales. Por lo que su ponderación será equitativa en el análisis final
Resultado.ProbabilidadFinalIdentificacionFigura = ...
    Resultado.TotalAciertos / Resultado.TotalCriteriosAnalisis;
if (Resultado.ProbabilidadFinalIdentificacionFigura >= ...
    Configuracion.ProbabilidadMinimaDeteccionFigura)
    if (Configuracion.VerboseConsola)
        disp('Plataforma de aproximación identificada correctamente');
        cadena = sprintf('-> La probabilidad de identificación segun aciertos es %f ...

```

```

, Resultado.ProbabilidadFinalIdentificacionFigura * 100);
disp(cadena);
end
%En caso contrario damos por no localizada la plataforma de aproximación
else
%Proceso de identificación incorrecto
if (Configuracion.VerboseConsole)
disp('Plataforma de aproximación identificada incorrectamente');
cadena = sprintf('-> La probabilidad de identificación según aciertos es %f ...
, Resultado.ProbabilidadFinalIdentificacionFigura * 100);
disp(cadena);
end
end
%Llegados a este punto se finaliza el programa para que no se ejecuten las
%funciones que hay declaradas más abajo.

%-----
% 25) Exportamos los datos a un fichero CSV si está habilitada la opción
%-----
%*****
%Obtenemos los datos que aparecen en la figura 6-1 de la memoria*
%*****
%Exportamos los datos a un fichero de texto en formato CSV
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 0, ...,
Configuracion.RutaDatosCSV, Configuracion.ImagenEntrada, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 0, ...,
Configuracion.RutaDatosCSV, [int2str(Columna) 'X' int2str(Fila)], false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 0, ...,
Configuracion.RutaDatosCSV, Configuracion.TipoUmbral, false);

%1) Posiciones Centros Elipses
AlineacionCentrosE1_E3_EjeY = abs(E1.CentroideEjeY - E3.CentroideEjeY);
AlineacionCentrosE1_E3_EjeX = abs(PuntoX_E2 - E2.CentroideEjeX);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, AlineacionCentrosE1_E3_EjeY, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, AlineacionCentrosE1_E3_EjeX, false);

%2) Ángulos Elipses con Eje OX
a1 = Datos.Angulo_E2_E3;
a2 = Datos.Angulo_E2_E1;
a3 = Datos.Angulo_E1_E3;
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, a1, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, a2, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, a3, false);

%3) Ratios Distancias Elipses
L1_L2 = (DistanciaEuclideaE1_E2 / DistanciaEuclideaE1_E3) * 100;
L3_L2 = (DistanciaEuclideaE2_E3 / DistanciaEuclideaE1_E3) * 100;
L3_L1 = (DistanciaEuclideaE2_E3 / DistanciaEuclideaE1_E2) * 100;
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, L1_L2, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, L3_L2, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, L3_L1, false);

%5) Ratio Área Circulos
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, (C1.Area/Datos.MediaAreaElipses) * 100, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, (C2.Area/Datos.MediaAreaElipses) * 100, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, (C3.Area/Datos.MediaAreaElipses) * 100, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, (C4.Area/Datos.MediaAreaElipses) * 100, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, (C2.Area/C1.Area) * 100, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...,
Configuracion.RutaDatosCSV, (C3.Area/C1.Area) * 100, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...

```

```

Configuracion.RutaDatosCSV, (C4.Area/C1.Area) * 100, false);

%6) Ratio Distancias Círculos
L4_L2 = DistanciaC3_C1/DistanciaEuclideaE1_E3;
L5_L2 = DistanciaC4_C1/DistanciaEuclideaE1_E3;
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...
Configuracion.RutaDatosCSV, (L4_L2) * 100, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...
Configuracion.RutaDatosCSV, (L5_L2) * 100, false);

%7) Ángulos Círculos con Eje X
a5 = AnguloC4_C1;
a6 = AnguloC3_C1;
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...
Configuracion.RutaDatosCSV, a5, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 2, ...
Configuracion.RutaDatosCSV, a6, true);

%-----
% 26) Exportamos los resultado al fichero CSV si está habilitada la opción
%-----
%Exportamos los datos de configuración de la imagen relativos a los APCs
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 0, ...
Configuracion.RutaResultadosCSV, Configuracion.ImagenEntrada, false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 0, ...
Configuracion.RutaResultadosCSV, [int2str(Columna) 'x' int2str(Fila)], false);
ExportarResultadoCSV(Configuracion.ExportarDatosCSV, 0, ...
Configuracion.RutaResultadosCSV, Configuracion.TipoUmbral, false);

%Exportamos los resultados de los análisis parciales comparativos (APC)
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.PosicionesCentrosElipses, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.AngulosElipses, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.ProporcionalidadDistancias, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.SimilitudAreaElipses, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.SimilitudExcentricidadElipses, 0);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.IdentificacionC1, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.IdentificacionC2, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.IdentificacionC3, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.IdentificacionC4, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, Resultado.IdentificacionCirculos, false);
ExportarResultadoCSV(Configuracion.ExportarResultadosCSV, 1, ...
Configuracion.RutaResultadosCSV, (cputime - TiempoCPU), true);

%Devolvemos el valor de la función
Probabilidad = Resultado.ProbabilidadFinalIdentificacionFigura * 100;
%Mostramos el tiempo total de CPU si está habilitada la opción
if (Configuracion.MostrarTiempoCPU)
    fprintf('El tiempo total de CPU es: %f \n', (cputime - TiempoCPU));
end

%Llamamos a la función GestionAutoUmbral para que compruebe con las dos
%estrategias de umbralización en el caso que sea necesario
GestionAutoUmbral (Probabilidad, Configuracion.ImagenEntrada, ...
Configuracion.FilaResolucionDeseada, Configuracion.TipoUmbral, ...
Configuracion.ProbabilidadMinimaAutoumbralizacion, Configuracion.Autoumbralizacion);

%Limpiamos la memoria borrando las variables y salimos de la función
clear all

return

%=====
% Función EscalarImagen --> Escala la imagen según la resolución deseada =
%=====

```

```

function Resultado = EscalarImagen(Imagen, Nombre, FilaDeseada, FilaActual)
%Comprobamos si la resolución deseada es menor que la actual
if ((FilaDeseada / FilaActual) < 1)
Resultado = imresize(Imagen, FilaDeseada / FilaActual);
[fila columna] = size(Resultado);
disp('Proceso de reducción de resolución activado. ');
texto = sprintf(' -> La imagen %s tiene ahora %d pixeles de ancho (columnas) y %d pixeles de largo (filas)'...
, Nombre, columna / 3, fila);
disp(texto);
%Si no lo es entonces no le hacemos modificación alguna
else
Resultado = Imagen;
end
end

%=====
% Función VisualizarImagen --> Gestiona la visualización de las imágenes =
%=====
function NumeroFigura = VisualizarImagen(Nombre, OpcionVisual, Numero)
if (OpcionVisual)
%Controlamos el número de la ventana según las que hayan aparecido.
figure(Numero);
imshow(Nombre);
impixelinfo;
NumeroFigura = Numero + 1;
else
NumeroFigura = Numero;
end
end

%=====
% Función DibujarElipses --> Dibuja el contorno de las elipse y el centro =
%=====
function NumeroFigura = DibujarElipse(Imagen, NumeroFigura, ...
ArrayElipses, ColorElipse, SimboloCentros, DibujarCentro, Espera)

%Controlamos el número de la ventana según las que hayan aparecido.
figure(NumeroFigura);
NumeroFigura = NumeroFigura + 1;

%Las elipses se dibujarán sobre la imagen binaria limpia
imshow(Imagen);
hold on

phi = linspace(0,2*pi,50);
cosphi = cos(phi);
sinphi = sin(phi);

%Recorremos todas las regiones conexas etiquetadas anteriormente
for l = 1:length(ArrayElipses)

%Para cada elipse, vamos a utilizar una forma paramétrica de la
%ecuación de la elipse para trazar su contorno sobre la imagen.
a = ArrayElipses(l).LongitudEjeGrande / 2;
b = ArrayElipses(l).LongitudEjeChico / 2;

theta = pi * ArrayElipses(l).Orientacion / 180;
R = [ cos(theta) sin(theta)
-sin(theta) cos(theta)];

xy = [a * cosphi; b * sinphi];
xy = R * xy;

x = xy(1,:) + ArrayElipses(l).CentroideEjeX;
y = xy(2,:) + ArrayElipses(l).CentroideEjeY;

%Dibujamos el contorno de la elipse a partir de su ecuación
plot(x, y, ColorElipse, 'LineWidth', 1);
impixelinfo;

%Mostramos los centroides si está habilitada la opción
if (DibujarCentro)
plot(ArrayElipses(l).CentroideEjeX, ...
ArrayElipses(l).CentroideEjeY, SimboloCentros);

```

```

end
end
if (~Espera)
    hold off
end
end
end

%=====
% Función DibujarCírculo --> Dibuja el contorno de las elipse y el centro =
%=====
function NumeroFigura = DibujarCirculo(Imagen, NumeroFigura, Array, Color, ...
    SimboloCentros, DibujarCentro, DibujarElipses, Umbral, FactorRadio)

%Controlamos el número de la ventana según las que hayan aparecido.
figure(NumeroFigura);
NumeroFigura = NumeroFigura + 1;

%Los círculos se dibujarán sobre la imagen binaria limpia, teniendo en
%cuenta que si está activada la opción de dibujar elipses se dibujarán
%en la misma venta
if (~DibujarElipses)
    imshow(Imagen);
    hold on
end

%Definimos pi y hacemos el cálculo del seno y coseno para no tener que
%hacer el cálculo cada vez que se procesa una región nueva
phi = 0:pi/50:2*pi;
cosphi = cos(phi);
sinphi = sin(phi);

%Recorremos todas las regiones conexas etiquetadas anteriormente
for l = 1:length(Array)
    %Para cada región comprobamos si la excentricidad es propia de un
    %círculo o de una elipse para utilizar la expresión más conveniente
    if (Array(l).Excentricidad > Umbral)
        %Para cada círculo, se va utilizar una forma paramétrica de la
        %ecuación de la elipse para trazar su contorno sobre la imagen.
        %Se utiliza el mismo método que el de la elipse, ya que si nos
        %basamos solo en el seno, coseno y radio, no podremos dibujar
        %los círculos cuando la inclinación sea elevada, ya que con ese
        %tipo de perspectiva, los círculos de la figura son elipses
        a = Array(l).LongitudEjeGrande / 2;
        b = Array(l).LongitudEjeChico / 2;

        theta = pi * Array(l).Orientacion / 180;
        R = [ cos(theta) sin(theta)
            -sin(theta) cos(theta)];

        xy = [a * cosphi; b * sinphi];
        xy = R * xy;

        x = xy(1,:) + Array(l).CentroideEjeX;
        y = xy(2,:) + Array(l).CentroideEjeY;

        %En caso contrario utilizamos la expresión del círculo
    else
        %Para cada círculo, utilizamos el radio y el centro
        r = Array(l).Radio * FactorRadio;
        x = (r * cosphi) + Array(l).CentroideEjeX;
        y = (r * sinphi) + Array(l).CentroideEjeY;
    end

    %Dibujamos el contorno de la elipse a partir de su ecuación
    plot(x, y, Color, 'LineWidth', 1);
    impixelinfo;

    %Mostramos los centroides si está habilitada la opción
    if (DibujarCentro)
        plot(Array(l).CentroideEjeX, ...
            Array(l).CentroideEjeY, SimboloCentros);
    end
end
end
hold off

```

```

end

%=====
% Función BinarizarImagen --> Binariza la imagen según el tipo indicado =
%=====
function [IBinaria, Tipo] = BinarizarImagen(Tipo, ImagenIntensidad, Umbral)
    %Elección del tipo de umbralización según la configuración
    switch Tipo
    % Determinamos el tipo de umbral a implar en la binarización
    case 'directo'
        % Binarización de la imagen original utilizando procedimiento directo
        IBinaria = ImagenIntensidad > Umbral;
        % Binarización por el método de otsu
    case 'otsu'
        Umbral = graythresh(ImagenIntensidad);
        IBinaria = ImagenIntensidad > Umbral;
        %Si se utiliza el tipo automático comenzamos con el directo porque es el
        %que mejores resultados ha dado en las pruebas funcionales realizadas
    case 'automatico'
        IBinaria = ImagenIntensidad > Umbral;
        Tipo = 'directo';
    otherwise
        disp('Tipo de valor incorrecto: Valores [fijo, otsu, automatico]');
    end
end
end

%=====
% Función GestionAutoUmbral --> Hace de forma autónoma la umbralización =
%=====
function GestionAutoUmbral (probabilidad, rutalmagen, escalado, tipoUmbral, ...
    probabilidadMinima, autoumbral)
    %Gestionamos el umbral automático, de tal forma que se prueban las dos
    %implementaciones posibles en el caso que no se consiga un mínimo del 50%
    if (probabilidad < probabilidadMinima && autoumbral)
        if (strcmp(tipoUmbral,'directo'))
            PlataformaUAV(rutalmagen, escalado, 'otsu', false);
        elseif(strcmp(tipoUmbral,'otsu'))
            PlataformaUAV(rutalmagen, escalado, 'directo', false);
        end
    end
end
end

%=====
% Función ObtenerElipses --> Obtiene datos de elipses a partir de regiones=
%=====
function ArrayElipses = ObtenerElipses(Regiones, Elipses, AreaMinima, ...
    AreaMaxima,ExcentricidadMin, ExcentricidadMax, TotalRegiones, Distancia, ...
    UmbralArea, UmbralExcentricidad, RatioDistanciaE1_2_E1_3, ...
    RatioDistanciaE2_3_E1_2, RatioDistanciaE2_3_E1_3, MaxUmbral, TipoDistancia)

    %Recorremos todas las regiones conexas etiquetadas anteriormente y se
    %guardan los datos de aquellas que nos interesan en base fundamentalmente
    %a su excentricidad. También se puede utilizar el área y la orientación.
    indice = 1;
    for k = 1:TotalRegiones

        %Si el área de la región, y la excentricidad están dentro del umbral
        %definido entonces las mostramos en la imagen final de salida
        if(Regiones(k).Area >= AreaMinima ...
            && Regiones(k).Area <= AreaMaxima ...
            && Regiones(k).Eccentricity >= ExcentricidadMin ...
            && Regiones(k).Eccentricity <= ExcentricidadMax ...
            && Regiones(k).EulerNumber == 1)
            %Si se cumplen las condiciones hacemos las asignaciones
            Elipses(indice).CentroideEjeX = Regiones(k).Centroid(1);
            Elipses(indice).CentroideEjeY = Regiones(k).Centroid(2);
            Elipses(indice).NumeroRegion = k;
            Elipses(indice).Orientacion = Regiones(k).Orientation;
            Elipses(indice).Area = Regiones(k).Area;
            Elipses(indice).Excentricidad = Regiones(k).Eccentricity;
            Elipses(indice).LongitudEjeGrande = Regiones(k).MajorAxisLength;
            Elipses(indice).LongitudEjeChico = Regiones(k).MinorAxisLength;
            %Incrementamos el índice de las elipses
            indice = indice + 1;
        end
    end
end

```

```

end
end

%Comprobamos si salen más de tres regiones en el proceso de segmentado
indice = length(Elipsoes);
%Si salen más de tres regiones el criterio será buscar distancias entre
%tres puntos que tengan la proporcionalidad que se hace en el análisis
%geométrico de la figura
if (indice > 3)
    %Hacemos las combinaciones sin repetición por medio de la
    %función nchoosek. El total de combinaciones vendrá dado por
    %la siguiente expresión  $n! / r! * (n - r)!$ 
    for n=1:length(Elipsoes)
        vector(n) = n;
    end
    vector = nchoosek(vector, 3);

    %Inicializamos la variables que controla el bucle con centinela
    n = 1; k = 1; encontrado = false;
    cuentaCero = 0; autoumbrales = 0;
    %Recorremos todas las combinaciones sin repetición posibles hasta
    %que haya elipses que cumplan con la condición de similitud de
    %elipses. Si no coinciden en la primera iteración aumentamos el
    %los filtros de área y excentricidad un 1% hasta un umbral máximo
    %para que no se entre en un bucle infinito en caso que no existan
    while (~encontrado && autoumbrales <= MaxUmbral)

        %Asignamos las elipses para saber con cuales trabajamos
        A = vector(k,1); B = vector(k,2); C = vector(k,3);

        %Comprobamos en primer lugar que el área de las tres elipses
        %tenga el nivel de analogía determinado en base al umbral
        if(SimilitudElipses(Elipsoes(A).Area, Elipsoes(B).Area, ...
            Elipsoes(C).Area, (UmbralArea + autoumbrales)) ...
            && (SimilitudElipses(Elipsoes(A).Excentricidad, ...
            Elipsoes(B).Excentricidad, Elipsoes(C).Excentricidad, ...
            (UmbralExcentricidad + autoumbrales))))

            %Almacenamos números de región
            Distancia(n).Elipse_A = A;
            Distancia(n).Elipse_B = B;
            Distancia(n).Elipse_C = C;

            %Obtenemos la distancia euclídea entre la elipse A y B
            Distancia(n).Elipse_A_B = CalcularDistancia(...
            Elipsoes(A).CentroideEjeX, Elipsoes(B).CentroideEjeX, ...
            Elipsoes(A).CentroideEjeY, Elipsoes(B).CentroideEjeY, ...
            TipoDistancia);

            %Obtenemos la distancia euclídea entre la elipse A y C
            Distancia(n).Elipse_A_C = CalcularDistancia(...
            Elipsoes(A).CentroideEjeX, Elipsoes(C).CentroideEjeX, ...
            Elipsoes(A).CentroideEjeY, Elipsoes(C).CentroideEjeY, ...
            TipoDistancia);

            %Obtenemos la distancia euclídea entre la elipse B y C
            Distancia(n).Elipse_B_C = CalcularDistancia(...
            Elipsoes(B).CentroideEjeX, Elipsoes(C).CentroideEjeX, ...
            Elipsoes(B).CentroideEjeY, Elipsoes(C).CentroideEjeY, ...
            TipoDistancia);

            %Calculamos la proporción entre distancias teniendo en cuenta
            %siempre cual de las dos distancias es la mayor para que vaya
            %emplazada en el denominador. Para ello, en primer lugar se
            %genera un vector con las distancias y se ordena para averiguar
            %cual es el valor más pequeño, el medio y el grande
            auxv(1) = Distancia(n).Elipse_A_B;
            d1 = Distancia(n).Elipse_A_B;
            auxv(2) = Distancia(n).Elipse_A_C;
            d2 = Distancia(n).Elipse_A_C;
            auxv(3) = Distancia(n).Elipse_B_C;
            d3 = Distancia(n).Elipse_B_C;

            %Ordenamos el vector de manera ascendente.

```

```

auxv = OrdenarArray(auxv, 'null', 'ASC');

%Obtenemos el ratio de distancias y le restamos el valor que
%debe tener según la estimación hecha a través del análisis
%geométrico de la figura que se presenta en la figura 6-1
%que forma parte de la memoria.
aux1 = (auxv(1)/auxv(3))-(RatioDistanciaE2_3_E1_3 / 100);
aux2 = (auxv(2)/auxv(3))-(RatioDistanciaE1_2_E1_3 / 100);
aux3 = (auxv(1)/auxv(2))-(RatioDistanciaE2_3_E1_2 / 100);

%Inicializamos dos variables locales para almacenar las
%orientaciones de las supuestas E1 y E3.
orientacion1 = 0;
orientacion3 = 0;

%Determinamos cual de las dos elipses es E1 y E3 utilizando
%las distancias que hay entre los centros de las elipses.
if (auxv(3) == d1)
    orientacion1 = Elipses(A).Orientacion;
    orientacion3 = Elipses(B).Orientacion;
elseif (auxv(3) == d2)
    orientacion1 = Elipses(A).Orientacion;
    orientacion3 = Elipses(C).Orientacion;
elseif (auxv(3) == d3)
    orientacion1 = Elipses(B).Orientacion;
    orientacion3 = Elipses(C).Orientacion;
end

%Calculamos la diferencia de los ángulos en valor absoluto
diferencial_orientacion = DiferenciaAngulo(abs(orientacion1)...
,abs(orientacion3), 1);

%Sumatorio de la diferencia en valor absoluto de todas los
%ratios de distancias euclídeas. Calculamos también la
%media ponderada donde la desviación respecto al ratio de
%distancias tiene un peso del 95% y la del área del 5
Distancia(n).Ratio = (abs(aux1 + aux2 + aux3)) + ...
diferencial_orientacion;

%Incrementamos el contador del índice del array
n = n + 1;
end

%Una vez procesadas todas las combinaciones del vector
if (k == length(vector))
    %Se comprueba cuantas elipses no tienen numero de índice 0
    if (Distancia(1).Elipse_A == 0)
        cuentaCero = cuentaCero + 1;
    end
    %Comprobación para la elipses B
    if ((Distancia(1).Elipse_B == 0))
        cuentaCero = cuentaCero + 1;
    end
    %Comprobación para la elipses C
    if ((Distancia(1).Elipse_C == 0))
        cuentaCero = cuentaCero + 1;
    end
    %Comprobamos si alguna de la combinación de elipses dada
    %por el vector ha cumplido la condición de similitud en
    %base al área y la excentricidad.
    if (cuentaCero <= 1 && ~encontrado)
        encontrado = true;
    %Nos preparamos para una nueva iteración en el bucle
    else
        %Inicializamos las variables para la nueva iteración
        cuentaCero = 0; n = 1; k = 0;
        encontrado = false;
        autoumbral = autoumbral + 1;
    end
end
%Incrementamos el contador del bucle
k = k + 1;
end

```

```

%Ordenamos el array utilizando el método de la burbuja
Distancia = OrdenarArray(Distancia, 'Ratio', 'ASC');
aux1 = Distancia(1).Elipse_A;
aux2 = Distancia(1).Elipse_B;
aux3 = Distancia(1).Elipse_C;

%Si los índices son 0, no se puede acceder al vector y por tanto,
%debemos devolver las elipses encontradas hasta este instante.
if (aux1 == 0 || aux2 == 0 || aux3 == 0)
    ArrayElipses = Elipses;
    return
end

%Limpiamos el vector auxiliar, y asignamos los datos de las elipses
clear auxvector;
auxvector(1) = Elipses(aux1);
auxvector(2) = Elipses(aux2);
auxvector(3) = Elipses(aux3);

%Eliminamos todas las posiciones del array
Elipses = VaciarArray(Elipses);

%Volvemos a introducir los datos en el vector de Elipses
Elipses = auxvector;
end
%Hacemos la asignación final de la función para devolver el resultado
ArrayElipses = Elipses;
end

%=====
% Función ObtenerCírculos --> Obtiene datos de círculos a partir de regiones=
%=====
function ArrayCirculos = ObtenerCirculos(Regiones, Círculos, AreaMinima, ...
AreaMaxima,ExcentricidadMin, ExcentricidadMax, TotalRegiones)

%Recorremos todas las regiones conexas etiquetadas anteriormente y se
%guardan los datos de aquellas que nos interesan en base fundamentalmente
%a su excentricidad. También se puede utilizar el área y la orientación.
indice = 1;
for k = 1:TotalRegiones
    %Si el área de la región, y la excentricidad están dentro del umbral
    %definido entonces las mostramos en la imagen final de salida
    if(Regiones(k).Area >= AreaMinima ...
    && Regiones(k).Area <= AreaMaxima ...
    && Regiones(k).Eccentricity >= ExcentricidadMin ...
    && Regiones(k).Eccentricity <= ExcentricidadMax ...
    && (Regiones(k).EulerNumber == 1 || Regiones(k).EulerNumber == 0 ...
    || Regiones(k).EulerNumber == -2 || Regiones(k).EulerNumber == -1))
        %Si se cumplen las condiciones hacemos las asignaciones
        Círculos(indice).CentroideEjeX = Regiones(k).Centroid(1);
        Círculos(indice).CentroideEjeY = Regiones(k).Centroid(2);
        Círculos(indice).NumeroRegion = k;
        Círculos(indice).Area = Regiones(k).Area;
        Círculos(indice).Radio = Regiones(k).EquivDiameter;
        Círculos(indice).LongitudEjeGrande = Regiones(k).MajorAxisLength;
        Círculos(indice).LongitudEjeChico = Regiones(k).MinorAxisLength;
        Círculos(indice).Orientacion = Regiones(k).Orientation;
        Círculos(indice).Excentricidad = Regiones(k).Eccentricity;
        %Incrementamos el índice de los círculos
        indice = indice + 1;
    end
end
ArrayCirculos = Círculos;
end

%=====
% Función IdentificarC1_C2 --> Identifica los círculos 1 y 2 dado un array=
%=====
function [C_1, C_2, Círculos] = IdentificarC1_C2(Dato, Círculos, RatioC1,...
RatioC2, Umbral, MaxUmbral, TipoDistancia, NumMaxCírculosEstrategia2)

%Inicializamos las variables centinelas de los bucles que controlan el
%momento en el que se han encontrado los círculos demandadas C1 y C2.
encontrado_C1 = false;

```

```

encontrado_C2 = false;

%Inicializamos las variables que controlan la iteración para el umbral
autoumbral_C1 = 0;
autoumbral_C2 = 0;

%Inicializamos las variables que gestionan los centinelas de los bucles
encontrados = false;

%Calculamos el tamaño del array que alberga los círculos candidatos
totalCirculos = length(Circulos);

%El estudio de eficiencia computacional llevado a cabo con una muestra
%heterogénea de imágenes revela que para arrays inferiores a los 5
%elementos proporciona mejor rendimiento la estrategia 2 que la 1 a pesar
%de que la mejora es muy leve. Según la simulación esta mejora se produce
%para casi cualquier tipo de distancia que se utilice.
if(totalCirculos > NumMaxCirculosEstrategia2)
    %ESTRATEGIA DE IDENTIFICACIÓN 1
    %Calculamos la proporción del área de cada círculo en relación a la
    %media aritmética de las elipses. Este dato se utiliza como filtro
    for k = 1:totalCirculos
        CIRCULOS(k).RatioArea = ...
            (CIRCULOS(k).Area / Dato.MedianaAreaElipses) * 100;
    end

    %Nos quedamos únicamente con la estructura del array de círculos
    CIRCULOS_VACIOS = CrearEstructuraCirculos();

    %Se inicializa la estructura de datos en la que se almacenan todos
    %los círculos que se consideran válidos por superar el filtrado
    CIRCULOS_VALIDOS = CIRCULOS_VACIOS;

    %Utilizamos un bucle principal para asegurar que detectamos 2 círculos
    while (~encontrados && autoumbral_C1 < MaxUmbral && autoumbral_C2 < MaxUmbral)

        %Se inicializan índices cada vez que comienza una nueva iteración
        k = 1; j = 1;

        %Búsqueda del Círculo1 con centinela para no tener que llegar al final
        while (~encontrado_C1 && k <= length(Circulos));
            %Identificación del Círculo1, el más grande situado en el centro
            if(AnalizarValor(Circulos(k).RatioArea, RatioC1, ...
                (Umbral + autoumbral_C1)))
                %Si no se ha calculado la distancia con anterioridad la
                %volvemos a calcular. Así se ahorran cálculos cuando es
                %necesario utilizar el autoumbral.
                if isempty(Circulos(k).Distancia)
                    %Calculamos distancia con el punto central de las elipses,
                    %ya que será un elemento claro para identificar C1 y C2.
                    CIRCULOS(k).Distancia = CalcularDistancia(...
                        CIRCULOS(k).CentroideEjeX, Dato.PuntoCentralEjeX, ...
                        CIRCULOS(k).CentroideEjeY, Dato.PuntoCentralEjeY, ...
                        TipoDistancia);
                end
                %Asignamos al array de círculos válidos al elemento
                CIRCULOS_VALIDOS(j) = CIRCULOS(k);
                %Incrementamos el índice del array de círculos válidos
                j = j + 1;
            end
            %Incrementamos el contador del bucle while anidado
            k = k + 1;
        end
        %Una vez acabada la iteración para cada umbral se comprueba que
        %no exista sólo el elemento de inicialización
        if (k <= 1 && CIRCULOS_VALIDOS(1).NumeroRegion == -1)
            autoumbral_C1 = autoumbral_C1 + 1;
        %Si se ha encontrado algún elemento entonces acabamos
        else
            %Ordenamos descendientemente el array de círculos por distancia
            CIRCULOS_VALIDOS = OrdenarArray(CIRCULOS_VALIDOS, 'Distancia', 'ASC');
            C_1 = CIRCULOS_VALIDOS(1);
            encontrado_C1 = true;
            %Inicializamos para comenzar el reconocimiento de C2

```

```

    circulosValidos = circulosVacios;
    k = 1; j = 1;
end

%Búsqueda del Círculo2, el de menor distancia con el punto central y
%que su área dividida entre la de C1 entre dentro del umbral indicado
%Centinela para no llegar al final si no es estrictamente necesario
while (~encontrado_C2 && encontrado_C1 && k < length(Circulos));
    %Identificación del Círculo2, el más pequeño situado en el centro
    if(AnalizarValor((Circulos(k).Area / C_1.Area)* 100, ...
    RatioC2, (Umbral + autoumbral_C2)))
        %Si no se ha calculado la distancia con anterioridad la
        %volvemos a calcular. Así se ahorran cálculos que se hayan
        %efectuado en C1 o si se incrementa el autoumbral
        if isempty(Circulos(k).Distancia)
            %Calculamos distancia con el punto central de las elipses,
            %ya que será un elemento claro para identificar C1 y C2.
            Circulos(k).Distancia = CalcularDistancia(...
            Circulos(k).CentroideEjeX, Dato.PuntoCentralEjeX, ...
            Circulos(k).CentroideEjeY, Dato.PuntoCentralEjeY, ...
            TipoDistancia);
        end
        %Asignamos al array de círculos válidos el elemento
        circulosValidos(j) = Circulos(k);
        %Incrementamos el índice del array de círculos válidos
        j = j + 1;
    end
    %Incrementamos el contador del bucle while anidado
    k = k + 1;
end
%Una vez acabada la iteración para cada umbral se comprueba que
%no exista sólo el elemento de inicialización
if ((k <= 1 && circulosValidos(1).NumeroRegion == -1))
    autoumbral_C2 = autoumbral_C2 + 1;
    circulosValidos = circulosVacios;
%En el caso de que no se hayan eliminado todos los elementos
%del array quiere decir que ya hemos encontrado C_2
else
    %Ordenamos ascendentemente el array de círculos por distancia
    circulosValidos = OrdenarArray(circulosValidos, 'Distancia', 'ASC');
    C_2 = circulosValidos(1);
    encontrado_C2 = true;
end
%Si se detectan ambos ponemos el centinela a true para detener la
%búsqueda que se gestiona a través del bucle externo.
if (encontrado_C1 && encontrado_C2)
    encontrados = true;
end
end
%Si el número de posibles círculos no está dentro del intervalo entonces
%empleamos una estructura que emplea muchas más instrucciones comparativas
%que aritméticas. SE utiliza por tanto la estrategia de identificación 2
else
    %ESTRATEGIA DE IDENTIFICACIÓN 2
    %Calculamos distancia con el punto central de las elipses, ya que será
    %un elemento claro para identificar los círculos 1 y 2
    for k = 1:totalCirculos
        %Cálculo de la distancia entre los puntos
        Circulos(k).Distancia = CalcularDistancia(Circulos(k).CentroideEjeX,...
        Dato.PuntoCentralEjeX, Circulos(k).CentroideEjeY, ...
        Dato.PuntoCentralEjeY, TipoDistancia);
        %Cálculo de la proporción de área
        Circulos(k).RatioArea = (Circulos(k).Area / Dato.MediaAreaElipses) * 100;
    end

    %Ordenamos de forma ascendente el array de círculos por distancia
    Circulos = OrdenarArray(Circulos, 'Distancia', 'ASC');

    %Utilizamos un bucle principal para asegurar que detectamos 2 círculos
    while (~encontrados && autoumbral_C1 < MaxUmbral && autoumbral_C2 < MaxUmbral)

        %Buscamos el círculo con menor distancia y con área en el intervalo,
        %es decir, el Círculo1. Se utiliza centinela para control del bucle
        k = 1;
    end
end

```

```

%Centinela para no llegar al final si no es estrictamente necesario
while (~encontrado_C1 && k <= length(Circulos));
    %Identificación del Círculo1, el más grande situado en el centro
    if(AnalizarValor(Circulos(k).RatioArea, RatioC1, Umbral + ...
        autoumbra1_C1))
        %En caso que esté dentro del umbral, hacemos las asignaciones.
        C_1 = Circulos(k);
        encontrado_C1 = true;
    end
    k = k + 1;
end

%Búsqueda el Círculo2, el de menor distancia con el punto central y
%que su área dividida entre la de C1 entre dentro del umbral indicado
k = 1;

%Centinela para no llegar al final si no es estrictamente necesario
while (~encontrado_C2 && encontrado_C1 && k <= length(Circulos));
    %Identificación del Círculo2, el más pequeño situado en el centro
    if(AnalizarValor((Circulos(k).Area / C_1.Area)* 100, ...
        RatioC2, (Umbral + autoumbra1_C2)))
        C_2 = Circulos(k);
        encontrado_C2 = true;
    end
    k = k + 1;
end
%Comprobamos que no quede algún círculo por detectar.
if (~encontrado_C1)
    autoumbra1_C1 = autoumbra1_C1 + 1;

%Para incrementar el contador de C_2 tenemos que detectar antes C_1
elseif (~encontrado_C2)
    autoumbra1_C2 = autoumbra1_C2 + 1;
end
%Si se detectan ambos ponemos el centinela a true para
if (encontrado_C1 && encontrado_C2)
    encontrados = true;
end
end
end
%Para controlar errores en primer lugar tenemos que comprobar si hemos
%encontrado C1 y C2. Si no los hemos encontrado entonces se utilizan las
%primeras posiciones de los vectores para devolver valores y así ya se
%asegura el control del error. Posteriormente comprobamos si el área de C2
%es mayor que el de C1, algo que tiene que suceder a la inversa.
if (~encontrado_C1)
    C_1 = Circulos(1);
elseif (encontrado_C1 && ~encontrado_C2)
    C_2 = Circulos(2);
elseif (~encontrado_C1 && ~encontrado_C2)
    C_2 = Circulos(1);
elseif (C_2.Area > C_1.Area)
    Aux = C_1;
    C_1 = C_2;
    C_2 = Aux;
end
end

%=====
% Función IdentificarC3_C4 --> Identifica los círculos 3 y 4 dado un array=
%=====
function [C_3, C_4] = IdentificarC3_C4(Circulos, Ratio, DistanciaElipse,...
    Umbral, C1, RatioDistancia, Consola, AnguloC3, AnguloC4, ...
    MaxUmbral, TipoDistancia, NumMaxCirculosEstrategia2)

%Calculamos la distancia con el círculo 1 para determinar el ratio que
%hay con la distancia entre las elipses 1 y 3, para finalmente calcular
%la diferencia en términos de valor absoluto, que será comparada con la
%estimada a través de la configuración del programa.
k = 1;
encontrado = false;
autoumbra1 = 0;
totalCirculos = length(Circulos);

```

```

%Nos quedamos únicamente con la estructura del array de círculos
circulosVacios = CrearEstructuraCirculos();

%Se inicializa la estructura de datos en la que se almacenan todos
%los círculos que se consideran validos por superar el filtrado
circulosValidos = circulosVacios;

%El estudio de eficiencia computacional llevado a cabo con una muestra
%heterogénea de imágenes revela que para arrays inferiores a los 5
%elementos proporciona mejor rendimiento la estrategia 2 que la 1 a pesar
%de que la mejora es muy leve. Según la simulación esta mejora se produce
%para casi cualquier tipo de distancia que se utilice
if(totalCirculos > NumMaxCirculosEstrategia2)
    %ESTRATEGIA DE IDENTIFICACIÓN 1
    %Inicializamos el índice de la estructura de círculos válidos
    j = 1;

    %Comenzamos la búsqueda hasta que se encuentren los círculos pero
    %con un control que no permita entrar en un bucle infinito
    while (~encontrado && autoumbrales <= MaxUmbrales && k <= totalCirculos)
        %Comprobamos que el área de los círculos este dentro del limite
        %definido por la estimación y el umbral.
        %En cada iteración, el umbral se incrementará automáticamente
        %en un 1% con el fin de encontrar círculos, y que como mínimo
        %haya dos posiciones en el array.
        if(AnalizarValor((Circulos(k).Area / C1.Area)* 100, Ratio, ...
            (Umbrales + autoumbrales)))
            %Para aumentar la eficiencia, reaprovechamos los cálculos
            %realizados en la función IdentificarC1_C2, lo que quiere
            %decir que solo se calcula la distancia si es null.
            if isempty(Circulos(k).Distancia)
                %Calculamos nuevamente la distancia por estar a null.
                Circulos(k).Distancia = CalcularDistancia(...
                    Circulos(k).CentroideEjeX, C1.CentroideEjeX, ...
                    Circulos(k).CentroideEjeY, C1.CentroideEjeY, ...
                    TipoDistancia);
            end
            %Cálculo de la diferencia de la distancia según limite y umbral
            Circulos(k).DiferenciaRatioDistancia = abs(...
                ((Circulos(k).Distancia / DistanciaElipse) * 100 - RatioDistancia));
            %Asignamos al array de círculos válidos el elemento
            circulosValidos(j) = Circulos(k);
            %Incrementamos el índice del array de círculos válidos
            j = j + 1;
        end

        %Una vez se llegue al final de cada iteración
        if (k == totalCirculos)
            %Se comprueba si hay dos círculos en el array. En caso contrario
            %se vuelve a repetir el bucle
            if (length(circulosValidos) < 2)
                encontrado = false;
                autoumbrales = autoumbrales + 1;
                circulosValidos = circulosVacios;
                k = 0;
                j = 1;
            %Si tiene dos o más entonces ponemos a true el centinela para
            %detener el bucle cuando se vuelva a evaluar la condición
            else
                encontrado = true;
                %Ordenamos ascendentemente el array de círculos validos
                %por la diferencia de distancia según la estimación
                %realizada en la configuración inicial
                Circulos = OrdenarArray(circulosValidos, ...
                    'DiferenciaRatioDistancia', 'ASC');
            end
        end
        %Incrementamos el contador del bucle principal
        k = k + 1;
    end

    %Estrategia 2, que se basa en calcular la distancia para todos los
    %elementos del vector, ordenarlo y escoger el primero que cumpla con
    %las condiciones de filtrado.

```

```

else
%ESTRATEGIA DE IDENTIFICACIÓN 2
%Calculamos la diferencia de distancia para todos los elementos
for k = 1:totalCirculos
%Al igual que antes, para aumentar la eficiencia, reaprovechamos
%los cálculos realizados en la función IdentificarC1_C2, lo que
%quiere decir que solo se calcula la distancia si es null.
if isempty(Circulos(k).Distancia)
%Se calcula de nuevo la distancia
Circulos(k).Distancia = CalcularDistancia(...
Circulos(k).CentroideEjeX, C1.CentroideEjeX, ...
Circulos(k).CentroideEjeY, C1.CentroideEjeY, ...
TipoDistancia);
end
%Cálculo de la diferencia de la distancia según limite y umbral
Circulos(k).DiferenciaRatioDistancia = abs(...
((Circulos(k).Distancia/DistanciaElipse) * 100 - RatioDistancia));
end

%Ordenamos ascendentemente el array de círculos por la
%diferencia de distancia según la estimación realizada en la
%configuración inicial del programa
Circulos = OrdenarArray(Circulos, 'DiferenciaRatioDistancia', 'ASC');

%Comenzamos la búsqueda hasta que se encuentren los círculos pero
%con un control que no permita entrar en un bucle infinito
while (~encontrado && autoumbral <= MaxUmbral)
%Inicializamos el índice justo antes de entrar en el bucle
k = 1; j = 1;

%Centinela para no llegar al final si no es estrictamente necesario
while ((length(circulosValidos) < 2) && (~encontrado) && ...
(k <= totalCirculos))
%Comprobamos que el área de los círculos este dentro del limite
%definido por la estimación y el umbral.
%En cada iteración, el umbral se incrementará automáticamente
%en un 1% con el fin de encontrar círculos, y que como mínimo
%haya dos posiciones en el array.
if (AnalizarValor((Circulos(k).Area / C1.Area)* 100, Ratio, ...
(Umbral + autoumbral)))
%Introducir elementos en el array de círculos válidos
circulosValidos(j) = Circulos(k);
%Incrementamos el índice de círculos validos
j = j + 1;
end
%Incrementamos el índice de círculos candidatos
k = k + 1;
end
%Se comprueba si hay dos círculos en el array de círculos
%válidos. En caso contrario se vuelve a repetir el bucle
if (length(circulosValidos) < 2)
encontrado = false;
autoumbral = autoumbral + 1;
circulosValidos = circulosVacios;
%Si tiene dos o más entonces ponemos a true el centinela para
%detener el bucle cuando se vuelva a evaluar la condición
else
encontrado = true;
Circulos = circulosValidos;
end
end
end
%Comprobamos el total de círculos detectado
switch totalCirculos
%Si no detectamos círculos
case 0
EscribirConsola(Consola, ...
'Círculos C3 y C4 no detectados. ');
return
%Si detectamos únicamente uno de ellos
case 1
%Calculamos el ángulo con respecto al círculo 1
Angulo = CalcularAngulo(Circulos(1).CentroideEjeX, ...
-1*(Circulos(1).CentroideEjeY), C1.CentroideEjeX, ...

```

```

-1*(C1.CentroideEjeY), true);
%Comprobamos que ángulo se aproxima más a la estimación
if (DiferenciaAngulo(AnguloC3, Angulo, 0) >= ...
    (DiferenciaAngulo(AnguloC4, Angulo, 0)))
    %Obtenemos C4 que está en la primera posición del array
    C_4 = Circulos(1);
else
    %Se obtiene C3 que está en la primera posición del array
    C_3 = Circulos(1);
end
otherwise
    %Ángulo entre el círculo1 y el primer círculo del array
    Angulo1 = CalcularAngulo(Circulos(1).CentroideEjeX, ...
        -1*(Circulos(1).CentroideEjeY), C1.CentroideEjeX, ...
        -1*(C1.CentroideEjeY), true);

    %Ángulo entre el círculo1 y el segundo círculo del array
    Angulo2 = CalcularAngulo(Circulos(2).CentroideEjeX, ...
        -1*(Circulos(2).CentroideEjeY), C1.CentroideEjeX, ...
        -1*(C1.CentroideEjeY), true);

    %El círculo3 será aquel que tenga un mayor ángulo respecto al eje X
    if (Angulo1 >= Angulo2)
        C_3 = Circulos(1);
        C_4 = Circulos(2);
    else
        C_3 = Circulos(2);
        C_4 = Circulos(1);
    end
end
end

%=====
% Función DistanciaElipses --> Obtiene distancia euclídea de las 3 elipses=
%=====
function [DistanciaEuclídeaE1_E2, DistanciaEuclídeaE1_E3, ...
    DistanciaEuclídeaE2_E3] = DistanciasElipses(Elipse, TipoDistancia)

    %Distancia entre la elipse 1 y 2
    DistanciaEuclídeaE1_E2 = CalcularDistancia(Elipse(1).CentroideEjeX, ...
        Elipse(2).CentroideEjeX, Elipse(1).CentroideEjeY, ...
        Elipse(2).CentroideEjeY, TipoDistancia);

    %Distancia entre la elipse 1 y 3
    DistanciaEuclídeaE1_E3 = CalcularDistancia(Elipse(1).CentroideEjeX, ...
        Elipse(3).CentroideEjeX, Elipse(1).CentroideEjeY, ...
        Elipse(3).CentroideEjeY, TipoDistancia);

    %Distancia entre la elipse 2 y 3
    DistanciaEuclídeaE2_E3 = CalcularDistancia(Elipse(2).CentroideEjeX, ...
        Elipse(3).CentroideEjeX, Elipse(2).CentroideEjeY, ...
        Elipse(3).CentroideEjeY, TipoDistancia);
end

%=====
% Función EscribirConsola --> Escribe en la consola si se activa verbose =
%=====
function EscribirConsola(OpcionVerbose, texto)

    %Escribimos por la consola si está activada la opción
    if (OpcionVerbose)
        disp(texto);
    end
end

%=====
% Función ExportarResultadoCSV --> Exporta el resultado en CSV a un fichero
%=====
function ExportarResultadoCSV(Opcion, Tipo, Fichero, Valor, Fin)

    %Exportamos los datos a CSV si está activada la opción
    if (Opcion)
        %Abrimos el fichero en modo escritura para guardar la información
        FicheroID = fopen(Fichero, 'a+');
    end
end

```

```

%Comprobamos el tipo de dato para utilizar en el formato
switch Tipo
%En el caso en que sea 0 se tratará de un string
case 0
%Guardamos la información y escribimos punto y coma al final
fprintf(FicheroID, '%s;', Valor);
%En el caso en que sea 0 se tratará de un entero
case 1
%Guardamos la información y escribimos punto y coma al final
fprintf(FicheroID, '%i;', Valor);
%En el caso en que sea 0 se tratará de un float
case 2
%Guardamos la información y escribimos punto y coma al final
fprintf(FicheroID, '%f;', Valor);
otherwise
disp('Tipo de valor asignado incorrecto: Valores [0, 1, 2]');
return
end
%Se comprueba si se termina el proceso para generar un salto
if (Fin)
fprintf(FicheroID, '\n');
end
%Cerramos el fichero donde se almacenan los datos en CSV
fclose(FicheroID);
end
end

%=====
% Función OrdenarArray --> Ordena el array según el parámetro: ASC ó DSC =
%=====
function Array = OrdenarArray(Array, Campo, Orden)
%Implementación del algoritmo quicksort que tiene coste computacional
%asintótico logarítmico O(n *log(n)). El algoritmo funciona de forma
%recursiva, seleccionando un elemento que está en el centro del arreglo
%llamado pivote, de tal manera que sitúa los demás elementos del array
%a cada lado del pivote. Por consiguiente, el objetivo es que queden a
%su izquierda los elementos menores que él y a su derecha los mayores,
%o viceversa. Todo depende del tipo de orden ASC o DSC indicado a través
%del parámetro llamado Orden.

%Calculamos la longitud del array que tenemos que ordenar
longitud_array = size(Array, 2);

%Comprobamos si la longitud del array es superior a un elemento, en caso
%contrario salimos de la función puesto no tiene sentido su ordenación
if(longitud_array < 2)
return;
end

%Longitud del array dividida entre dos para casos pares e impares
indice_mitad = cast(floor(longitud_array / 2), 'uint8');
%Iniciamos las variables locales, esto es, índices y lista que
%contienen la ordenación parcial a la izquierda y derecha del pivote.
izquierda = VaciarArray(Array);
derecha = izquierda;
j = 1;
k = 1;

%Obtenemos el pivote que estará situado en la mitad del array
pivote = Array(indice_mitad);

%Elección del tipo de orden según el parámetro. ASC o DSC
switch Orden
%Ordenación ascendente por el método quick sort
case 'ASC'
%Orden si no se especifica un campo de ordenación
if (strcmp(Campo, 'null'))
%Recorremos el array con un único bucle
for i=1:longitud_array
%Comprobamos que el índice no coincida con el del pivote
if(i ~= indice_mitad)
%Situamos a la izquierda del pivote los elementos menores,
%lo que quiere decir que implementamos un orden ascendente
if(Array(i) < pivote)

```

```

        izquierda(j) = Array(i);
        j = j + 1;
        %En caso contrario situamos a la derecha del pivote
    else
        derecha(k) = Array(i);
        k = k + 1;
    end
end
end
%Orden si se especifica un campo de ordenación
else
    %Recorremos el array con un único bucle
    for i=1:longitud_array
        %Comprobamos que el índice no coincida con el del pivote
        if(i ~= indice_mitad)
            %Situamos a la izquierda del pivote los elementos menores,
            %lo que quiere decir que implementamos un orden ascendente
            if (Array(i).(genvarname(Campo)) < ...
                pivote.(genvarname(Campo)))
                izquierda(j) = Array(i);
                j = j + 1;
            %En caso contrario situamos a la derecha del pivote
            else
                derecha(k) = Array(i);
                k = k + 1;
            end
        end
    end
end
end
%Ordenación descendente por el método quick sort
case 'DSC'
    %Orden si no se especifica un campo de ordenación
    if (strcmp(Campo, 'null'))
        %Recorremos el array con un único bucle
        for i=1:longitud_array
            %Comprobamos que el índice no coincida con el del pivote
            if(i ~= indice_mitad)
                %Situamos a la izquierda del pivote los elementos menores,
                %lo que quiere decir que implementamos un orden ascendente
                if(Array(i) > pivote)
                    izquierda(j) = Array(i);
                    j = j + 1;
                %En caso contrario situamos a la derecha del pivote
                else
                    derecha(k) = Array(i);
                    k = k + 1;
                end
            end
        end
    end
    %Orden si se especifica un campo de ordenación
else
    %Recorremos el array con un único bucle
    for i=1:longitud_array
        %Comprobamos que el índice no coincida con el del pivote
        if(i ~= indice_mitad)
            %Situamos a la izquierda del pivote los elementos menores,
            %lo que quiere decir que implementamos un orden ascendente
            if (Array(i).(genvarname(Campo)) > ...
                pivote.(genvarname(Campo)))
                izquierda(j) = Array(i);
                j = j + 1;
            %En caso contrario situamos a la derecha del pivote
            else
                derecha(k) = Array(i);
                k = k + 1;
            end
        end
    end
end
end
otherwise
    disp('Parámetro de orden incorrecto. Los parámetros son ASC ó DSC');
end

%Hacemos llamadas recursivas hasta que el array quede totalmente ordenado

```

```

izquierda = OrdenarArray(izquierda, Campo, Orden);
derecha = OrdenarArray(derecha, Campo, Orden);

%El array ordenado será la unión de la lista que contiene los elementos
%de la izquierda, el pivote, más el array con los elementos de la derecha
Array = [izquierda pivote derecha];
end

%=====
% Función EliminarElementoArray --> Elimina celdas del array de figuras =
%=====
function Array = EliminarElementoArray(Array, Elemento, Campo)
    aux = length(Array);
    j = 1;
    while (j < aux)
        if (Array(j).(genvarname(Campo)) == Elemento.(genvarname(Campo)))
            Array(j) = [];
            aux = aux - 1;
        end
        j = j + 1;
    end
end

%=====
% Función VaciarArray --> Elimina todos los elementos del array =
%=====
function Array = VaciarArray(Array)
    %Eliminamos todas las posiciones del array
    for j=length(Array):-1:1
        Array(j) = [];
    end
end

%=====
% Función CrearEstructuraCirculos --> Estructura datos vacia para circulos=
%=====
function Circulos = CrearEstructuraCirculos()

    %Definimos la estructura de datos que servirá para almacenar circulos
    Circulos.CentroideEjeX = 0;
    Circulos.CentroideEjeY = 0;
    Circulos.Area = 0;
    Circulos.NumeroRegion = -1;
    Circulos.Diametro = 0;
    Circulos.Distancia = [];
    Circulos.RatioArea = 0;
    Circulos.DiferenciaRatioDistancia = 0;
    Circulos.Procesado = false;
    Circulos.Radio = 0;
    Circulos.LongitudEjeGrande = -1;
    Circulos.LongitudEjeChico = -1;
    Circulos.Orientacion = -1;
    Circulos.Excentricidad = -1;
end

%=====
% Función DiferenciaAngulo --> Calcula la diferencia entre 2 ángulos dados=
%=====
function Diferencia = DiferenciaAngulo (Angulo_1, Angulo_2, Tipo)

switch Tipo
    %En el caso en que sea 0 se calculará en el rango (0, 360]
    case 0
        %Comprobamos si la diferencia entre los ángulos es mayor que 180
        if (abs(Angulo_1 - Angulo_2)) > 180
            %Si lo es, entonces obtenemos la diferencia en relación a 360
            if (Angulo_1 > Angulo_2)
                Diferencia = 360 - Angulo_1 + Angulo_2;
            else
                Diferencia = 360 - Angulo_2 + Angulo_1;
            end
        %Si la diferencia es inferior se devuelve el valor absoluto
        else
            Diferencia = abs(Angulo_1 - Angulo_2);
        end
    end
end

```

```

end
%En el caso en que sea 0 se calculará en el rango [90,90]
case 1
%Comprobamos si la diferencia entre los ángulos es mayor que 45
if (abs(Angulo_1 - Angulo_2) > 45)
%Si lo es, entonces obtenemos la diferencia en relación a 90
if(Angulo_1 > Angulo_2)
Diferencia = 90 - Angulo_1 + Angulo_2;
else
Diferencia = 90 - Angulo_2 + Angulo_1;
end
%Si la diferencia es inferior se devuelve el valor absoluto
else
Diferencia = abs(Angulo_1 - Angulo_2);
end
%En el resto de los casos indicamos las opciones válidas
otherwise
disp('Tipo de intervalo de ángulo incorrecto: Valores [0, 1]');
return
end
end

%=====
% Función CalcularDistancia --> Distancia entre dos puntos según parámetro=
%=====
function Distancia = CalcularDistancia (PD_X, PI_X, PD_Y, PI_Y, Tipo)

%La distancia a usar se determina según el tipo enviado como parámetro
switch Tipo
%En el caso en que sea 0 se calculará la distancia euclídea
case 0
%Obtenemos la distancia euclídea
Distancia = sqrt((PD_X - PI_X)^2 + (PD_Y - PI_Y)^2);
%En el resto de los casos indicamos las opciones válidas
otherwise
disp('Tipo de distancia incorrecta: Valores [0, 1, 2]');
return
end
end

%=====
% Función CalcularAngulo --> Halla el ángulo que forman 2 puntos con EjeX =
%=====
function Angulo = CalcularAngulo (PD_X, PD_Y, PI_X, PI_Y, Rango360)

%Ángulo entre el punto destino (PD) y el punto de origen P(PI)
%En primer lugar calculamos la pendiente de la recta que une los puntos
EjeX = PD_X - PI_X;
EjeY = PD_Y - PI_Y;

%Calculamos el ángulo en el intervalo [-180, 180] con la función atan2
Angulo = (atan2(EjeY, EjeX) * 180) / pi;

%Comprobamos si está activada la opción de 360 para devolver el ángulo
%en el intervalo [0, 360]
if (Rango360)
%Como deseamos tener un ángulo en valor positivo en el intervalo
%[0,360) entonces obtenemos el resto de la división entre 360
Angulo = mod(Angulo, 360);
end
end

%=====
% Función SimilitudElipses --> Determina si hay analogía según el umbral =
%=====
function booleano = SimilitudElipses (Valor_X, Valor_Y, Valor_Z, Umbral)

%Llevamos a cabo todas las comparaciones entre los valores proporcionados
%en base al umbral pasado como parámetro. Servirá entre otros temas para
%determinar la analogía de las elipses en base al área y excentricidad
if(Valor_X >= Valor_Z * (1 - Umbral / 100) ...
&& Valor_X >= Valor_Y * (1 - Umbral / 100) ...
&& Valor_Y >= Valor_Z * (1 - Umbral / 100) ...
&& Valor_Y >= Valor_X * (1 - Umbral / 100) ...

```

```

&& Valor_Z >= Valor_Y * (1 - Umbral / 100) ...
&& Valor_Z >= Valor_X * (1 - Umbral / 100) ...
&& Valor_X <= Valor_Z * (1 + Umbral / 100) ...
&& Valor_X <= Valor_Y * (1 + Umbral / 100) ...
&& Valor_Y <= Valor_Z * (1 + Umbral / 100) ...
&& Valor_Y <= Valor_X * (1 + Umbral / 100) ...
&& Valor_Z <= Valor_Y * (1 + Umbral / 100) ...
&& Valor_Z <= Valor_X * (1 + Umbral / 100))
    %Si se cumplen todas las condiciones devolvemos true
    booleano = true;
else
    booleano = false;
end
end

%=====
% Función AnalizarCirculo --> Determina un círculo a partir de sus umbrales
%=====
function booleano = AnalizarCirculo (Angulo, AnguloEstimado, UmbralAngulo, ...
    Distancia, DistanciaEstimada, UmbralDistancia, RatioArea, ...
    RatioAreaEstimada, UmbralArea)

    %Si se cumplen todas las condiciones damos por correcto el círculo
    if (AnalizarValor(Angulo, AnguloEstimado, UmbralAngulo) ...
        && (AnalizarValor(Distancia, DistanciaEstimada, UmbralDistancia)) ...
        && (AnalizarValor(RatioArea, RatioAreaEstimada, UmbralArea)))
        booleano = true;
    else
        booleano = false;
    end
end

%=====
% Función AnalizarValor --> Determina si un valor está en el rango definido
%=====
function booleano = AnalizarValor (Valor, Estimacion, Umbral)

    if ((Valor >= Estimacion * (1 - Umbral / 100)) ...
        && (Valor <= Estimacion * (1 + Umbral / 100)))
        booleano = true;
    else
        booleano = false;
    end
end

%=====
% Función AsignarFigura --> Asigna los datos de una figura a su variable =
%=====
function Destino = AsignarFigura (EjeX, EjeY, NumeroRegion, Orientacion, ...
    Excentricidad, Area)
    %Hacemos las asignaciones en base a los parámetros recibidos
    Destino.CentroideEjeX = EjeX;
    Destino.CentroideEjeY = EjeY;
    Destino.NumeroRegion = NumeroRegion;
    Destino.Orientacion = Orientacion;
    Destino.Excentricidad = Excentricidad;
    Destino.Area = Area;
end

%Fin de la función principal, es decir, fin de la función PlataformaUAV.m
end

```