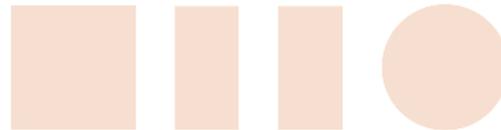


GRADO EN INGENIERÍA EN ELECTRÓNICA Y
AUTOMÁTICA INDUSTRIAL



Trabajo Fin de Grado

RECONOCIMIENTO DE FICHAS DE DOMINÓ MEDIANTE
TÉCNICAS DE VISIÓN ARTIFICIAL



ESCUELA POLITECNICA
SUPERIOR

Autor: David Herrero Gordo

Tutor: M^a Soledad Escudero Hernanz

UNIVERSIDAD DE ALCALÁ
Escuela Politécnica Superior

GRADO EN INGENIERÍA EN ELECTRÓNICA Y AUTOMÁTICA
INDUSTRIAL

Trabajo Fin de Grado

RECONOCIMIENTO DE FICHAS DE DOMINÓ MEDIANTE
TÉCNICAS DE VISIÓN ARTIFICIAL

Autor: David Herrero Gordo

Director: M^a Soledad Escudero Hernanz

TRIBUNAL:

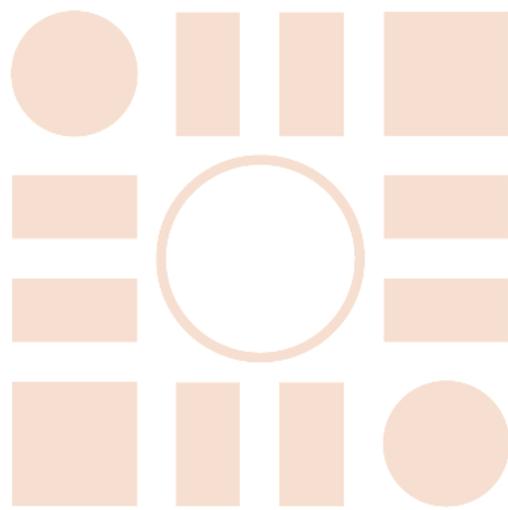
Presidente: D. LUCIANO BOQUETE VÁZQUEZ

Vocal 1º: D. MANUEL GÁLVEZ GÁLVEZ

Vocal 2º: D^a M^a SOLEDAD ESCUDERO HERNANZ

CALIFICACIÓN:

FECHA:



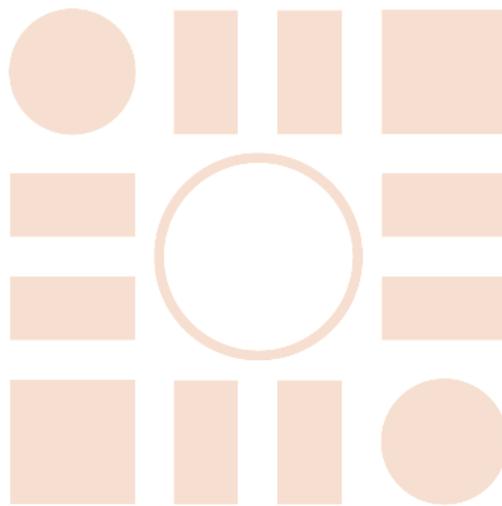
ESCUELA POLITECNICA
SUPERIOR

ÍNDICE

| | Página |
|-----------------------------------------|--------|
| Capítulo 1 | |
| 1.1- Resumen..... | 11 |
| 1.2- Abstract..... | 12 |
| 1.3- Palabras clave..... | 13 |
| 1.4- Resumen extendido..... | 14 |
| Capítulo 2 | |
| 2.1- Introducción..... | 19 |
| 2.2- Estado del arte..... | 22 |
| Capítulo 3 | |
| 3.1- Objetivos..... | 27 |
| 3.2- Diagrama de bloques..... | 28 |
| Capítulo 4 | |
| 4.1- Desarrollo del proyecto..... | 33 |
| 4.2- Punto de partida..... | 34 |
| 4.3- Adquisición y pre-procesado..... | 38 |
| 4.4- Localización de fichas..... | 42 |
| 4.5- Recuperación de información..... | 45 |
| 4.6- Identificación de fichas..... | 57 |
| Capítulo 5 | |
| 5.1- Condiciones de adquisición..... | 67 |
| 5.2- Resultados..... | 71 |
| 5.3- Conclusiones y líneas futuras..... | 79 |
| Capítulo 6 | |
| 6.1- Pliego de condiciones..... | 85 |
| 6.2- Presupuesto..... | 87 |
| 6.3- Manual de usuario..... | 90 |

Capítulo 7

| | |
|------------------------|----|
| 7.1- Referencias..... | 95 |
| 7.2- Bibliografía..... | 96 |

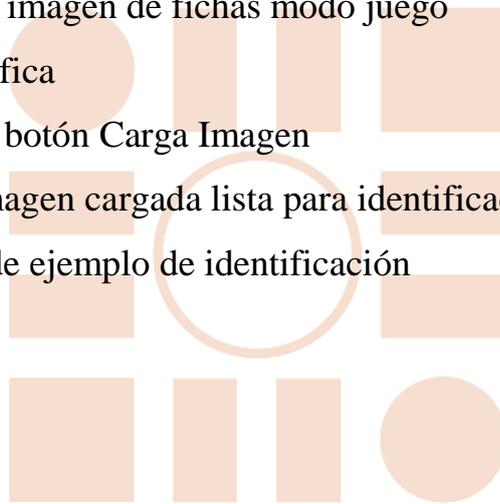


ESCUELA POLITECNICA
SUPERIOR

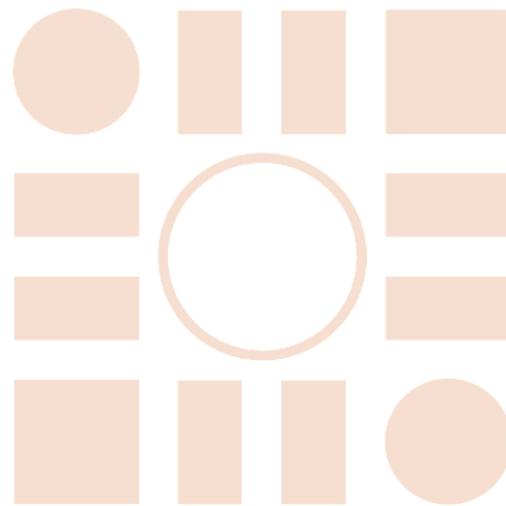
ÍNDICE DE FIGURAS

| | Página |
|------------------------------------------------------------------------------|--------|
| Figura 2.1: Ejemplo de ficha utilizada en proyecto | 20 |
| Figura 2.2: Ejemplo de cadena de juego | 21 |
| Figura 3.1: Diagrama de bloques | 28 |
| Figura 4.1: Esquema de distribución de fotodiodos para el filtro de Bayer | 34 |
| Figura 4.2: Absorción de Rayos en fotodiodos | 35 |
| Figura 4.3: Imagen RGB y distribución por canales | 36 |
| Figura 4.4: Ficha de dominó | 37 |
| Figura 4.5: Método de umbralización | 38 |
| Figura 4.6: Canales RGB | 39 |
| Figura 4.7: Proceso de umbralización | 40 |
| Figura 4.8: Imagen de salida bloque 1 | 41 |
| Figura 4.9: Perfil de ficha | 42 |
| Figura 4.10: Perfil de ficha | 43 |
| Figura 4.11: Detalle de separaciones | 43 |
| Figura 4.12: Separaciones partidas | 46 |
| Figura 4.13: Proceso de unión | 47 |
| Figura 4.14: Separaciones partidas | 49 |
| Figura 4.15: Puntos extremos | 50 |
| Figura 4.16: Imagen de separaciones a unir | 51 |
| Figura 4.17: Imagen de separación patrón (a) y unida (b). | 52 |
| Figura 4.18: Selección de objetos para superposición | 53 |
| Figura 4.19: Objetos de búsqueda | 54 |
| Figura 4.20: Imagen capturada y entregado por bloque 3 | 57 |
| Figura 4.21: Imagen de error de identificación de punto de referencia | 58 |

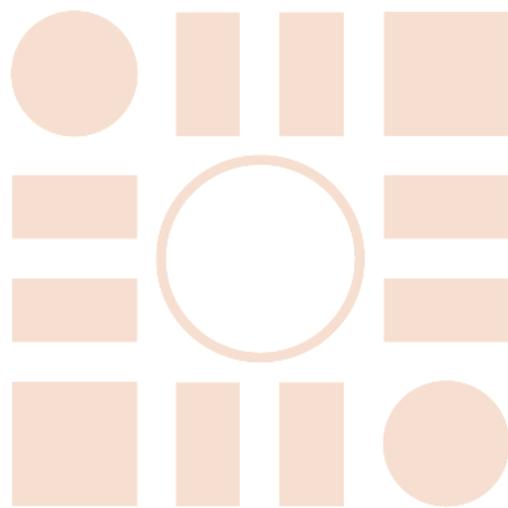
| | |
|--------------------------------------------------------------|----|
| Figura 4.22: Dimensiones de ficha | 60 |
| Figura 4.23: Imagen recortada | 62 |
| Figura 5.1: Imagen de fichas utilizadas | 68 |
| Figura 5.2: Ejemplo de imagen con reflejos | 68 |
| Figura 5.3: Ejemplo de imagen con reflejos procesada | 69 |
| Figura 5.4: Ejemplo de imagen a distintas alturas | 70 |
| Figura 5.5: Ejemplo de imagen de fichas separadas | 71 |
| Figura 5.6: Ejemplo de imagen de fichas juntas | 72 |
| Figura 5.7: Ejemplo de imagen de fichas modo juego | 73 |
| Figura 6.1: Interfaz gráfica | 90 |
| Figura 6.2: Ventana de botón Carga Imagen | 91 |
| Figura 6.3: GUI con imagen cargada lista para identificación | 91 |
| Figura 6.4: Resultado de ejemplo de identificación | 92 |



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA
SUPERIOR



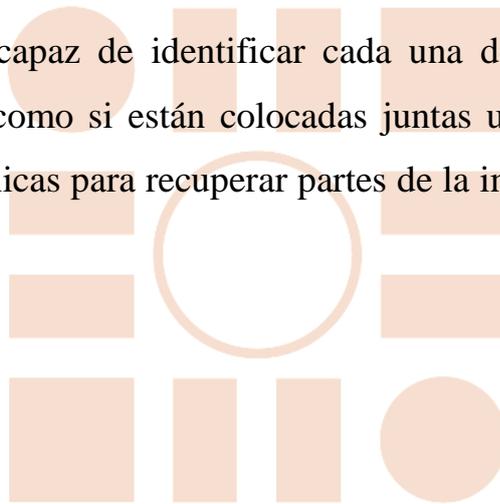
ESCUELA POLITECNICA SUPERIOR

1.1 RESUMEN

El presente Trabajo Fin de Grado consiste en la identificación de fichas de dominó por un sistema, de forma automática a partir de imágenes tomadas por una cámara digital.

Para ello se ha implementado un algoritmo en Matlab, el cual recibe un archivo de imagen (típicamente jpg, tiff...etc.) en el que aparecen fichas de dominó sobre un tapete color verde.

El sistema es capaz de identificar cada una de las fichas tanto si se encuentran separadas como si están colocadas juntas unas a otras. Además se han implementado técnicas para recuperar partes de la imagen perdidas debido a la iluminación.



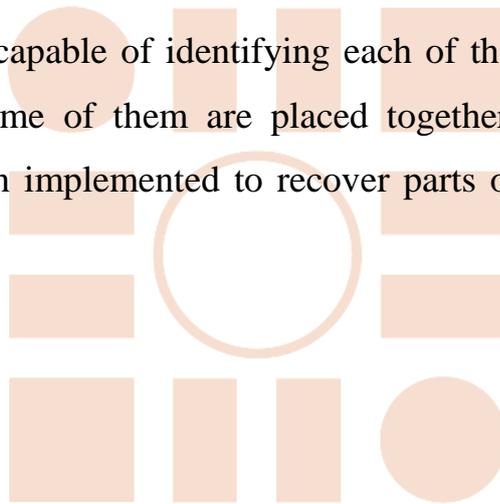
ESCUELA POLITECNICA
SUPERIOR

1.2 ABSTRACT

This document corresponding to a Final Degree Project is dedicated to the automatic identification of dominoes from images taken by a digital camera.

To reach this objective an algorithm has been implemented in Matlab receiving an image file (typically jpg, tiff...etc) where dominoes appear on a green color mat.

The system is capable of identifying each of the cards so much if they are separated as if some of them are placed together to others. In addition technologies have been implemented to recover parts of the image lost due to the lighting.



ESCUELA POLITECNICA
SUPERIOR

1.3 PALABRAS CLAVE

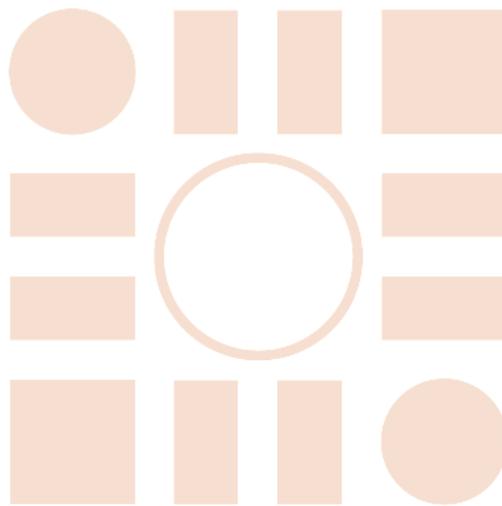
Fichas de dominó

Visión Artificial

Matlab

Identificación

Reconocimiento



ESCUELA POLITECNICA
SUPERIOR

1.4 RESUMEN EXTENDIDO

Se realiza este Trabajo Fin de Grado con el objetivo de diseñar un sistema que sea capaz de reconocer fichas de dominó. Para ello se pretende implementar un algoritmo en Matlab, el cual reciba un archivo de imagen en el que aparezcan las fichas de dominó y el programa de forma automática reconozca cada una de las fichas, independientemente de su posición, orientación o distribución en la imagen.

Se ha establecido un requisito a la captura de imágenes y que consiste en que las fichas se colocarán sobre un tapete de color verde. Este requisito se impone por lógica, ya que en principio el juego de dominó se desarrolla sobre una mesa de juego que habitualmente se encuentra cubierta por un tapete de color verde.

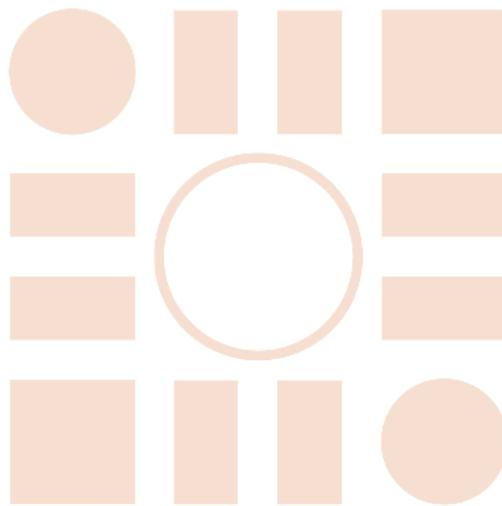
Para realizar la identificación, el algoritmo realiza en primer lugar una serie de filtrados de la imagen, para posteriormente segmentar la imagen y extraer los contornos de las fichas. A continuación se localiza dentro de ellos la línea intermedia de separación de las dos partes de que consta la ficha.

Una vez que se ha localizada la línea de separación, se procede a la identificación de la ficha mediante el recuento de puntos existentes a cada lado de dicha separación.

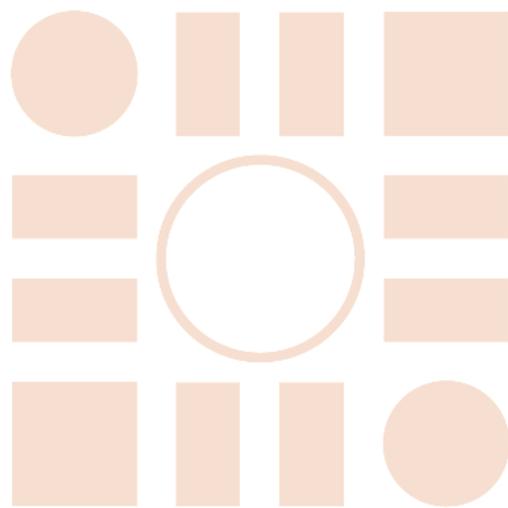
Inicialmente se procede a implementar el algoritmo básico de identificación. Con posterioridad se han realizado una serie de mejoras para obtener un sistema lo más robusto posible ante variaciones de luminosidad dentro de la imagen, brillo, orientación e identificación de las fichas estando colocadas de forma contiguas unas a otras a modo de juego, con el objetivo de que más adelante se pueda implementar un sistema capaz de analizar una partida de dominó y participar en ella.

En el capítulo 4 se explica de forma detallada el algoritmo final implementado, indicando las razones por las cuales se ha decidido optar por una solución, frente al resto.

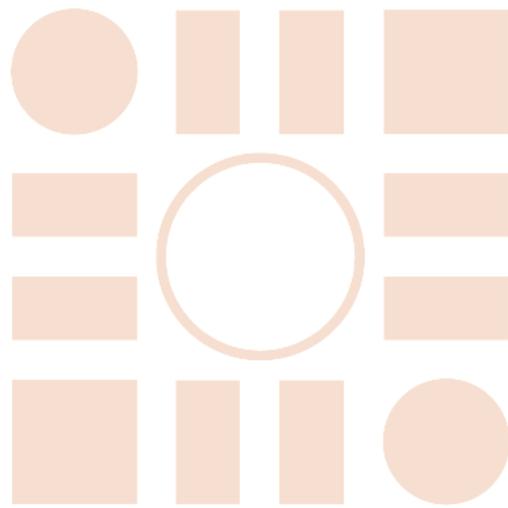
Por último, en el capítulo 5 se hace un resumen de los resultados obtenidos por el algoritmo programado tras el análisis masivo de imágenes y de las condiciones de adquisición que tienen que cumplir estas para obtener dichos resultados.



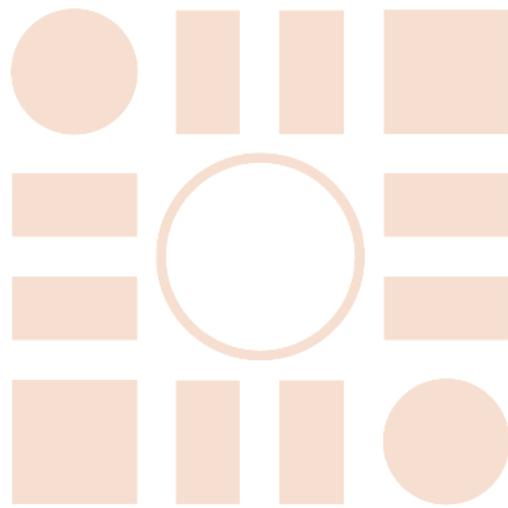
ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA SUPERIOR



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA
SUPERIOR

2.1 INTRODUCCIÓN

El procesamiento digital de imágenes se refiere al conjunto de técnicas utilizadas para tratar imágenes de forma que se pueda extraer de ellas la información deseada. Comprende una serie de procesos como el filtrado, la detección de patrones, la comparación entre imágenes...etc.

El tratamiento de imágenes se engloba dentro de la visión artificial que es uno de los campos más importantes y con mayor auge dentro de la inteligencia artificial, ya que es uno de los canales de comunicación entre los ordenadores y el medio que les rodea. Dentro del objetivo de crear sistemas robotizados que sean capaces de interactuar con el entorno, es fundamental que estos sistemas sean capaces de “ver” el espacio que les rodea y más importante aún, ser capaces de interpretarlo.

Inicialmente, la visión artificial era un paradigma reservado únicamente a la investigación, pero a medida que la electrónica se ha ido incorporando a la industria, la seguridad e incluso al hogar, se ha convertido en una de las líneas fundamentales de desarrollo tecnológico.

Una componente muy importante en el tratamiento de imágenes radica en la adquisición de estas y de parámetros como la iluminación, la óptica, el sensor o la conversión A/D. En un segundo bloque estaría el análisis de dichas imágenes y los distintos procesos que se realizan sobre ellas para conseguir extraer la información que deseamos. En este proyecto se va a hacer hincapié en este segundo bloque para intentar obtener un sistema lo más robusto posible, ante los diferentes factores que pueden aparecer en la adquisición de las imágenes.

Para implementar las diferentes técnicas de procesado digital, se va hacer uso del programa Matlab que es una de las herramientas de software matemático más populares y con mayor potencia de cálculo. Además cuenta entre sus

herramientas con una “Toolbox” específica para el tratamiento digital de imágenes con lo que se pueden implementar programas con una gran capacidad de procesado.

En este proyecto se van a utilizar las herramientas más comunes existentes en esta Toolbox y se van a implementar una serie de técnicas que permitan la identificación de forma correcta de las fichas de dominó incluso en condiciones en las que la iluminación sea deficiente o existan reflejos que distorsionen las imágenes obtenidas, de forma que el sistema sea capaz de recuperar la información de las fichas a pesar del “ruido” que pueda existir en la imagen a analizar. De esta forma se ha procurado diseñar un algoritmo lo más robusto posible, que obtenga resultados aceptables incluso en condiciones no óptimas de adquisición de las imágenes.

El dominó es un juego de mesa que consta típicamente de veintiocho fichas. Cada ficha está compuesta de dos partes separadas habitualmente por una línea negra intermedia y en ocasiones con una “chincheta” en el centro de la línea. En la figura 2.1 se muestra un ejemplo de una de las fichas utilizadas.



Figura 2.1: Ejemplo de ficha utilizada en proyecto

En el dominó de 28 fichas a cada lado de la separación aparecen una cantidad de puntos que van desde cero (blanca) hasta seis.

En el desarrollo del juego se van colocando fichas por uno u otro extremo de las que se encuentran colocadas en la mesa, de forma que van formando una cadena en la que los extremos iguales de las fichas se colocan juntos. En el caso de las fichas dobles (las que tienen igual número de puntos a cada lado de la separación), se colocan en dirección perpendicular a la cadena de fichas. En la figura 2.2 se puede observar un ejemplo de cadena de juego.

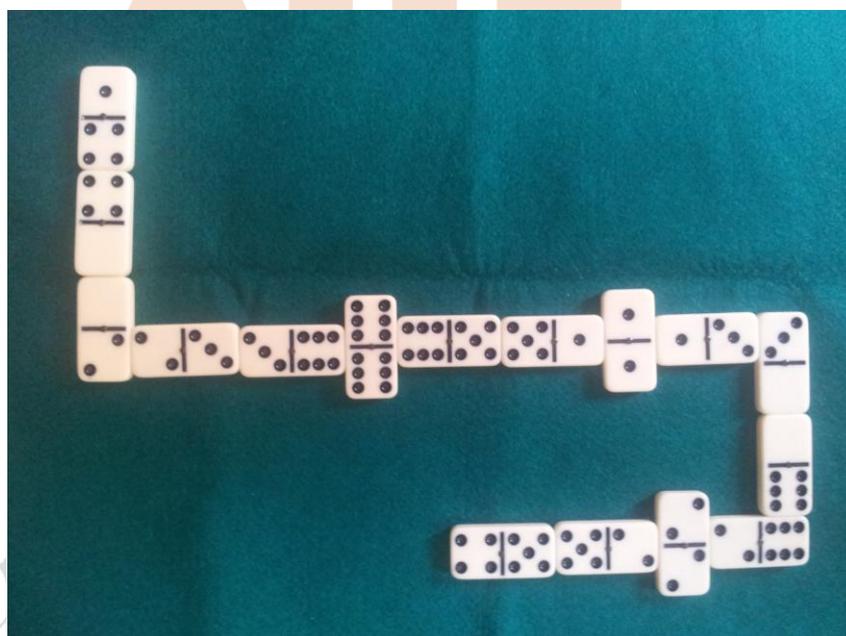


Figura 2.2: Ejemplo de cadena de juego

Es por tanto imprescindible, ser capaces de separar las fichas cuando se encuentran formando un solo objeto.

2.2 *ESTADO DEL ARTE*

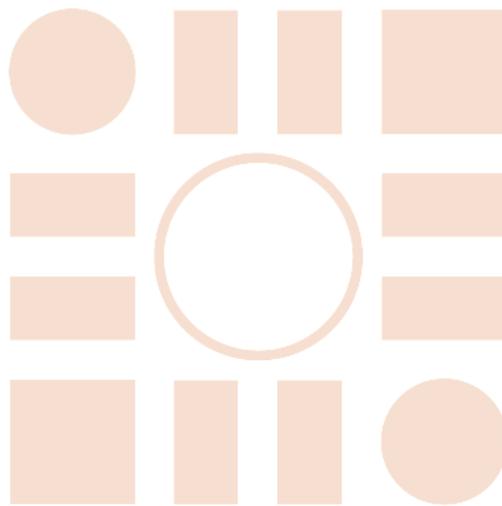
Actualmente existen diversos estudios en lo que se refiere a la identificación de patrones u objetos en imágenes aplicado a campos tan diferentes como pueden ser la conducción de vehículos inteligentes [1], la seguridad (cuento y reconocimiento de personas en grandes acumulaciones de gente) [2] o incluso la identificación de osos polares [3].

Con la introducción de la electrónica en la vida cotidiana de las personas a través de lo que denominan dispositivos inteligentes, existe un gran campo de trabajo relacionado con la forma en que estos dispositivos son capaces de interactuar con el entorno que les rodea, y una parte muy importante de esta interacción viene determinada por la forma en que estos dispositivos son capaces de interpretar las imágenes adquiridas mediante sistemas de visión artificial como pueden ser por ejemplo una simple cámara digital de un teléfono móvil, y analizarlas, la mayor parte de las veces, buscando patrones que les proporcionen información.

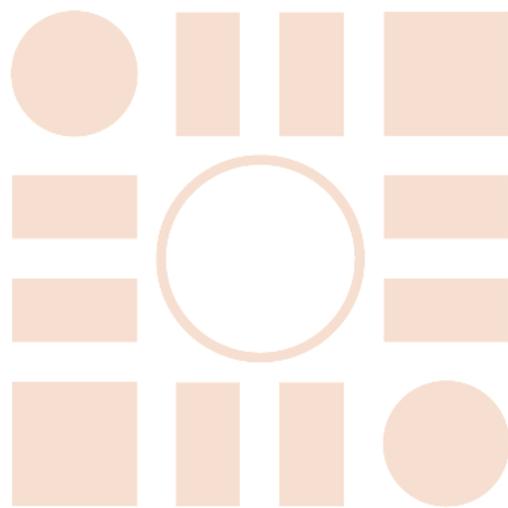
En esta ocasión, se desea implementar un sistema que sea capaz de reconocer fichas de dominó. En el artículo [4] se utiliza un sistema de reconocimiento de fichas de dominó implementado en un sistema robótico, pero que tiene una serie de limitaciones debido a la altura de la cámara de adquisición de las imágenes y que además únicamente identifica fichas de forma individual esparcidas sobre una superficie.

En la mayoría de los casos estudiados, el reconocimiento de estos objetos se realiza en unas condiciones óptimas de adquisición, con lo que cuando estos sistemas son aplicados en entornos más complejos y con unas condiciones de adquisición de imágenes en los que no es posible controlar todos los parámetros, a menudo se producen errores en la identificación de estos patrones. Es por ello por lo que este proyecto no se va a limitar a implementar

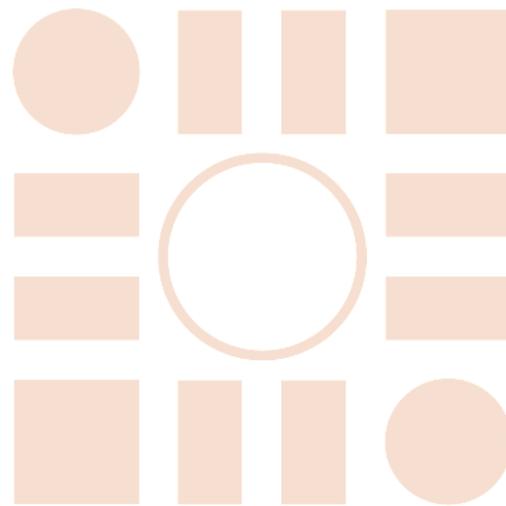
un simple reconocimiento de patrones sino que además se va a realizar un sistema lo suficientemente robusto como para que sea capaz de reconocer las fichas de dominó aun estando pegadas unas a otras, de forma que más adelante el sistema pueda interactuar en una partida de dominó.



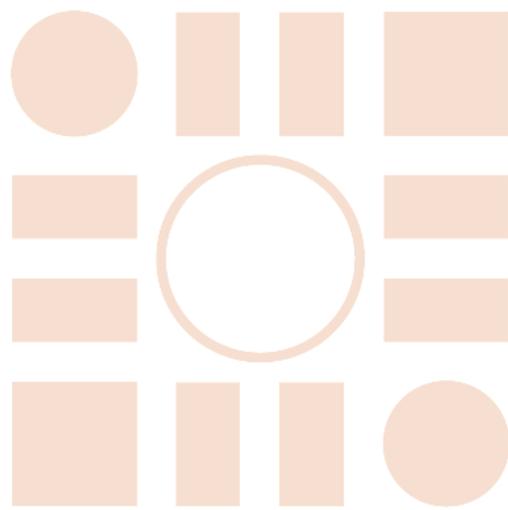
ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA SUPERIOR

3.1 OBJETIVOS

El objetivo principal del presente proyecto es crear un sistema que sea capaz de identificar las fichas de dominó existentes en una imagen.

Para considerar que el objetivo principal se ha alcanzado, se han establecido una serie de hitos que se deben conseguir a lo largo del trabajo, que se enumeran a continuación:

- Se debe implementar un algoritmo simple que identifique las fichas de dominó en condiciones de adquisición óptimas.
- Mejorar el algoritmo para que identifique las fichas, independientemente de la orientación y de la distribución que tengan (tanto si se encuentran aisladas, como si están pegadas unas a otras).
- Obtener un algoritmo robusto, mejorando el filtrado y pre-procesado de la imagen para una correcta identificación de las fichas en diferentes condiciones de iluminación.
- Implementar un sistema que sea capaz de recuperar parte de la información perdida en la captura de la imagen, debido a los reflejos que se producen en partes de las fichas debido a su geometría e iluminación.

3.2 DIAGRAMA DE BLOQUES

El algoritmo de reconocimiento de fichas de dominó lo podríamos dividir en diferentes bloques.

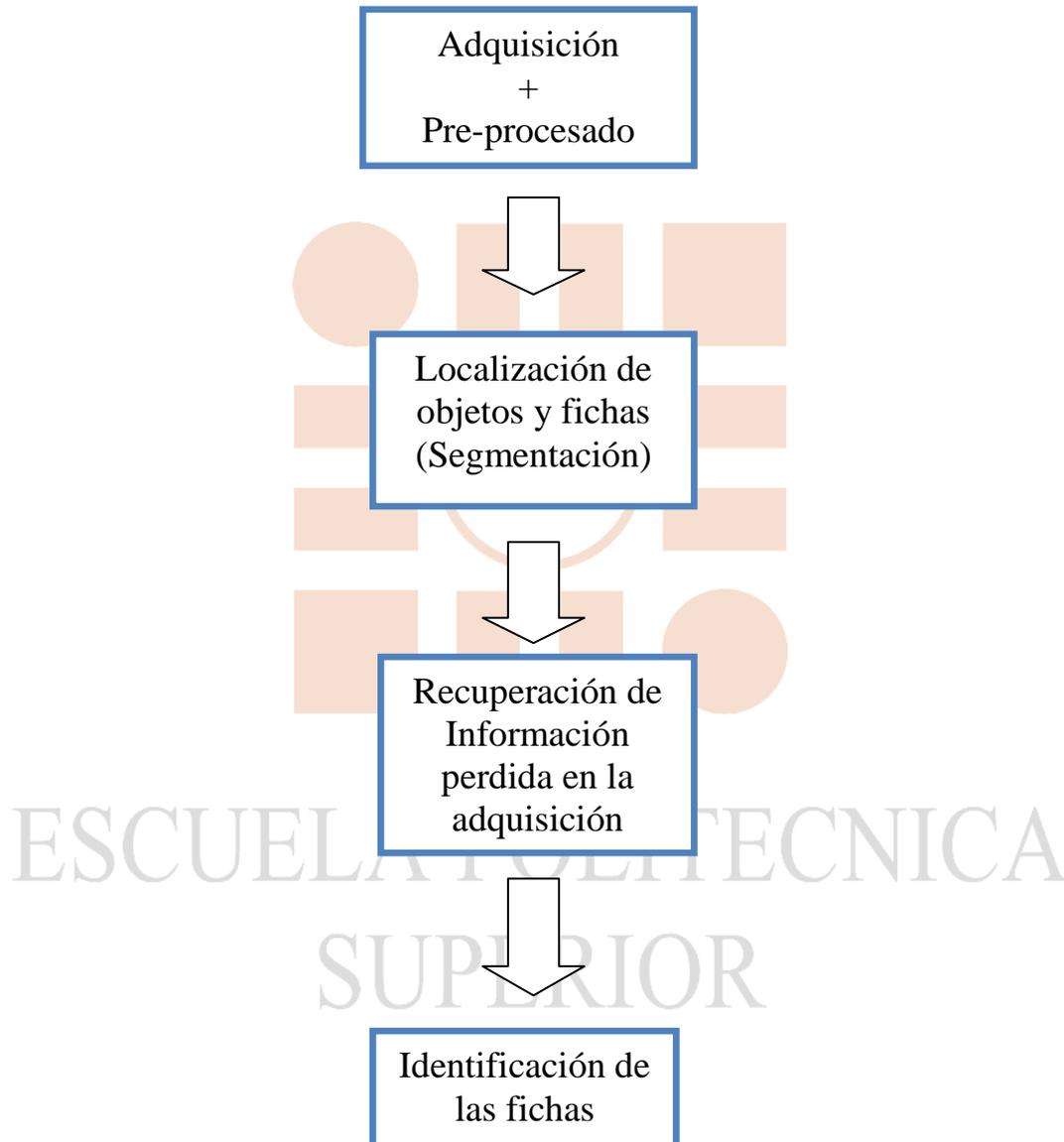


Figura 3.1: Diagrama de bloques

Estos son los cuatro bloques principales en los que se divide el algoritmo implementado en Matlab. Se explicarán más detalladamente en el siguiente capítulo, pero básicamente cada uno realiza las siguientes operaciones:

- Bloque 1: Adquisición y Pre-procesado.

Este bloque se encargará de adquirir la imagen, transformarla en B/N y prepararla para la segmentación.

- Bloque 2: Localización.

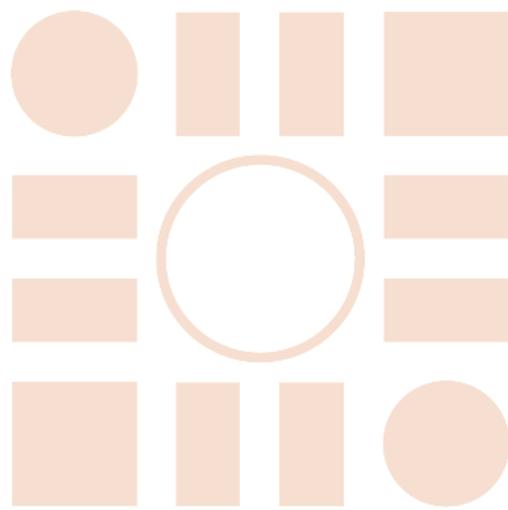
Una vez que la imagen está preparada para su procesado, se realiza la segmentación de la misma en dos fases. En una primera fase, se identifican los objetos que hay en la imagen. En la segunda fase se van a localizar las fichas que hay dentro de cada objeto, ya que dos fichas que se encuentran pegadas una a otra, serán un solo objeto pero dos fichas.

- Bloque 3: Recuperación de información.

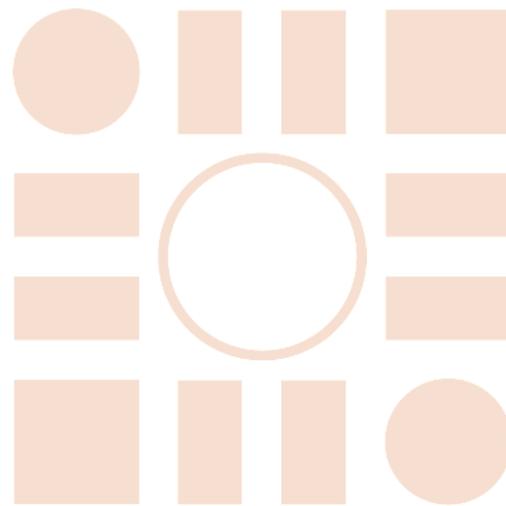
Tanto durante la adquisición de la imagen como durante el procesado, se puede llegar a perder parte de la información imprescindible para reconocer la ficha (principalmente la línea de separación intermedia que tienen las fichas) por lo que antes de proceder a su identificación, es necesario recuperar esta información.

- Bloque 4: Identificación.

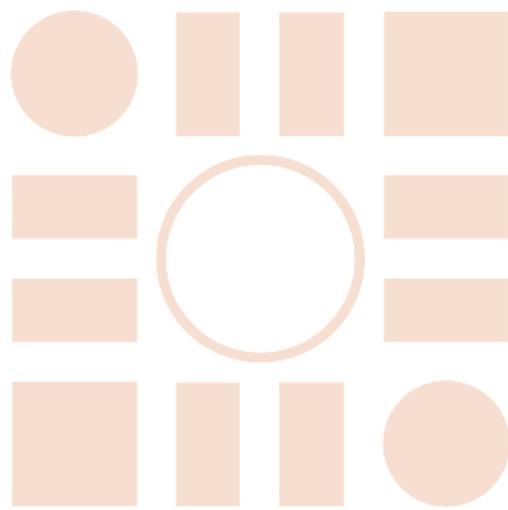
Por último, una vez que la imagen ha sido tratada, ya se puede proceder a su reconocimiento.



ESCUELA POLITECNICA SUPERIOR



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA SUPERIOR

4.1 *DESARROLLO DEL PROYECTO*

Para la realización de este proyecto se ha procurado seguir la planificación temporal de los hitos que se han marcado en el capítulo de objetivos. Se ha partido de una programación básica del algoritmo de identificación y a medida que se ha ido experimentando con él, se han ido mejorando cada uno de los objetivos parciales hasta llegar al algoritmo final implementado. Se podría explicar la evolución del algoritmo de una forma temporal tal y como ha ido evolucionando secuencialmente pero parece más didáctico explicarlo dividido en los bloques que lo componen y dentro de cada bloque dar cuenta de la sucesión temporal y el motivo de elección de la solución adoptada. Por tanto se va a dividir este capítulo de desarrollo en los cuatro bloques que se han indicado en el diagrama de bloques de la figura 3.1.

Antes de desarrollar el funcionamiento del algoritmo, se va a realizar una breve introducción de los aspectos más importantes que se han tenido en cuenta a la hora de su implementación, tanto técnicos como materiales.

ESCUELA POLITECNICA
SUPERIOR

4.2 PUNTO DE PARTIDA

Antes de comenzar la explicación del algoritmo es necesario realizar antes una serie de reflexiones sobre los materiales a utilizar.

En primer lugar es necesario hacer una breve introducción sobre la cámara que se utilizará para adquirir las imágenes y su funcionamiento. En este caso se trata de una cámara digital cuyo dispositivo de captación es un sensor CMOS de 8 Megapixels, capaz de capturar imágenes con una resolución de 3264x2448 pixels. Este dispositivo basa la adquisición de la imagen en lo que se denomina “filtro de Bayer” [5]. Este filtro, que fue inventado por *Bryce Bayer*, para obtener las imágenes a color utiliza cuatro fotodiodos para asignar un valor de color a cada píxel. De los cuatro fotodiodos, dos captan el verde, uno el rojo y otro el azul. El objeto de tener dos fotodiodos que capturen los colores verdes, es que el ojo humano es más sensible a este color. En la imagen de la figura 4.1 podemos ver como se distribuyen los fotodiodos correspondientes a cuatro píxeles:

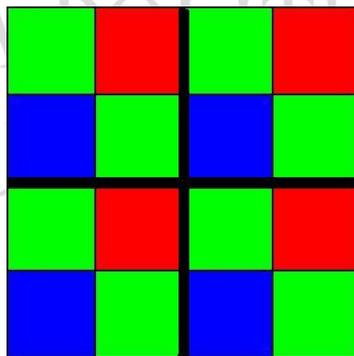


Figura 4.1: Esquema de distribución de fotodiodos para el filtro de Bayer

Se puede ver como a cada píxel le corresponde la información proporcionada por la interpolación de dos fotodiodos verdes llamados elementos sensores de luminosidad y un fotodiodo rojo y otro azul, que se denominan

elementos sensores de color. De esta forma el patrón para la formación de una imagen es el mostrado en la figura 4.2:

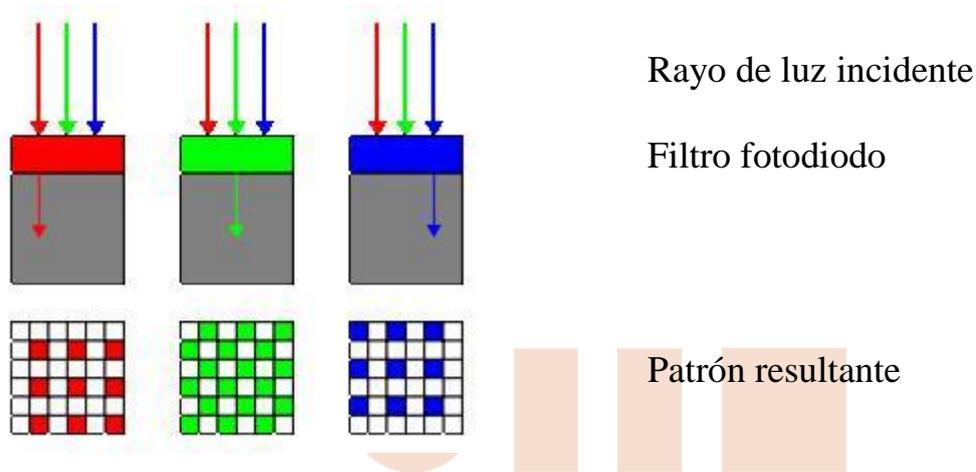


Figura 4.2: Absorción de Rayos en fotodiodos

A partir de estos patrones y en función ya del algoritmo que se utilice para procesar las imágenes, nos da lugar a los distintos formatos de imagen que conocemos: JPEG, TIFF...etc.

Esta imagen que nos proporciona la cámara en formato JPG, debemos transformarla al formato RGB para poder trabajar adecuadamente con ella.

Las imágenes en color RGB están compuestas por la superposición de tres imágenes, cada una de ellas en uno de los tres colores fundamentales (Red, Green, Blue). A su vez la intensidad de cada color se mide en valores que van, en este proyecto, desde el cero hasta el 255. Por tanto cada píxel se obtiene de la combinación de un valor de intensidad (de 0 a 255) del color rojo, más un valor del azul más otro del verde.

La manera que tiene Matlab de representar estas imágenes es mediante tres matrices (una por cada color) cuyos elementos oscilan entre 0 y 255. En la figura 4.3 vemos una imagen RGB utilizada en este proyecto y su descomposición en los tres canales de los que hablamos:

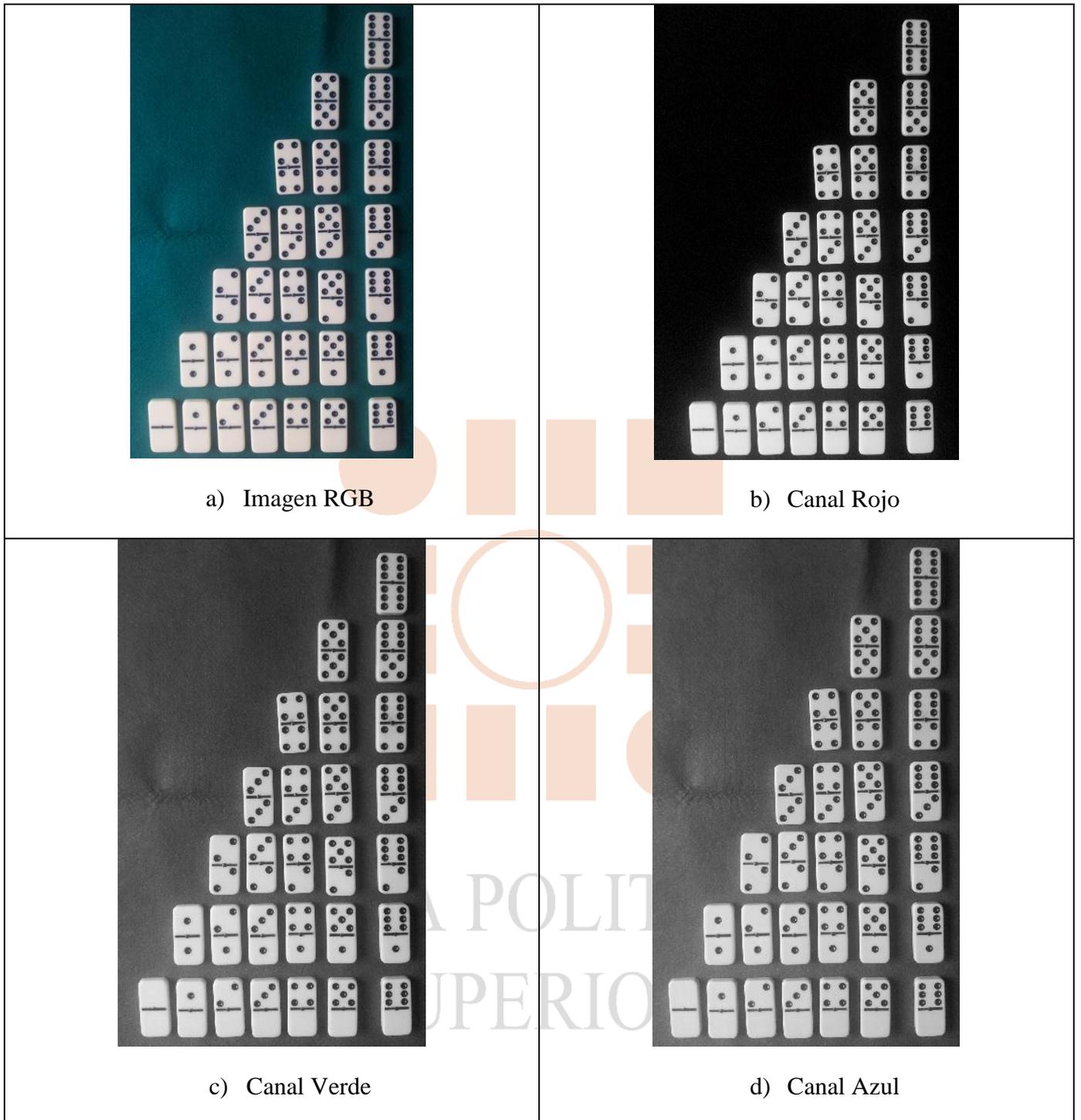


Figura 4.3: Imagen RGB y distribución por canales

Esta manera de componer las imágenes es importante, ya que se va a utilizar a la hora de realizar su umbralización.

El segundo aspecto a tener en cuenta será el método de identificación de la ficha de dominó. Las fichas utilizadas serán como la mostrada en la figura 4.4:



Figura 4.4: Ficha de Dominó

Inicialmente se pensó en realizar el reconocimiento de las fichas basándose en los descriptores sift. Este método basa su funcionamiento en que a partir de un grupo de imágenes patrón a las que asigna una serie de puntos clave, busca esos puntos clave en la imagen que se quiere identificar. Pero este método utilizado ya en otros proyectos parecidos [6], produce errores de identificación con imágenes que son muy similares entre ellas. Precisamente, este es el caso de las fichas de dominó.

Por tanto, ante la restricción de utilizar métodos de identificación de imágenes basadas en el “*matching*”, se opta por localizar las fichas dentro de la imagen buscando la línea de separación intermedia que tienen todas ellas, y contar el número de puntos existentes a cada lado de dicha línea. Previo a este conteo, es necesario preparar la imagen como se explicará más adelante, para conseguir una identificación correcta.

4.3 ADQUISICIÓN Y PRE-PROCESADO

Este bloque inicial, se va a encargar de leer la imagen y realizar los filtrados y transformaciones necesarias para que se pueda umbralizar y obtener una imagen en B/N con la información suficiente para ser analizada.

La adquisición de las imágenes, como ya se ha dicho, se realiza con una cámara digital y se guardan con el formato jpg. Por tanto se ha creado una función en Matlab que en su llamada, recibe como argumento el archivo de imagen que se quiere analizar. En este punto ya se tiene la imagen en RGB en el workspace de Matlab y el primer objetivo consiste en umbralizarla.

En una primera versión del programa, se realizaba una transformación de la imagen en escala de grises utilizando en comando *rgb2gray*, y posteriormente de manera arbitraria se establecía un valor de corte basado en el histograma, para realizar la umbralización. En la figura 4.5 se puede observar un ejemplo de este método de umbralización.

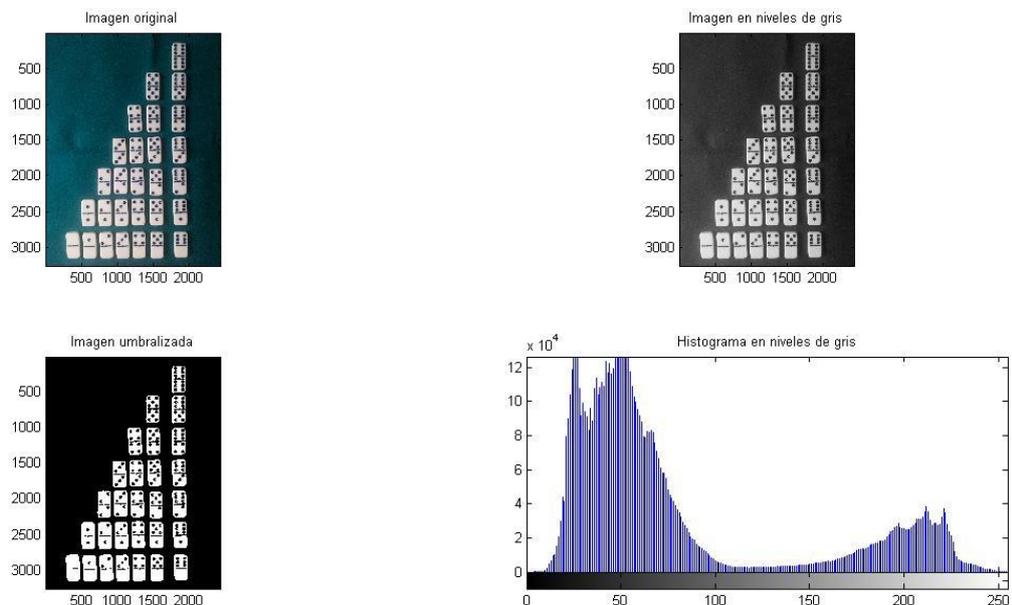


Figura 4.5: Método de umbralización

En la imagen del histograma, se puede observar como el valor de corte umbral, debe estar entre 100 y 150. Concretamente en este caso, está situado en 130. Obviamente este método no es nada flexible con respecto al punto de corte umbral, con lo que al variar la iluminación de las imágenes, la ventana de umbralización se desplaza y el algoritmo no es capaz de adaptarse a esos cambios.

Para conseguir un punto de corte flexible se recurre entonces a la herramienta *graythresh*, pero no resulta efectiva al enfrentarse con imágenes en las que la iluminación no es uniforme, ya que en las zonas menos iluminadas, se difuminan los bordes de las fichas. Por tanto es necesario encontrar otro método más efectivo de realizar la umbralización, si se desea obtener un algoritmo robusto.

Dado que el problema principal para la umbralización proviene de la iluminación, y esta se muestra principalmente en el canal G, no parece adecuado realizar un promediado de los tres canales juntos (R, G, B) para obtener la imagen en escala de grises. En la figura 4.6 se puede observar los distintos canales en escala de grises de una imagen a analizar.

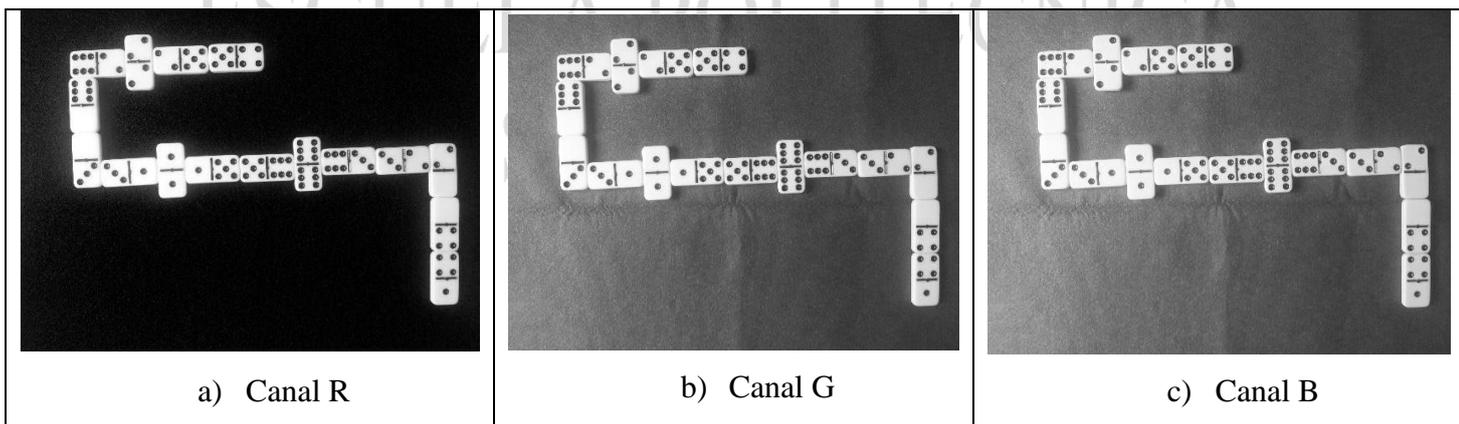


Figura 4.6: Canales RGB

Como se puede apreciar en la imagen 4.6, el canal más inmune a la variación de iluminación es el canal Rojo, con lo que se opta por realizar una umbralización de cada canal por separado y tomando como base el canal R.

Se umbralizan los tres canales independientemente utilizando *im2bw*, cada uno de ellos con el umbral proporcionado por la herramienta *graythresh*.

Partiendo de la imagen umbralizada del canal rojo, se realiza consecutivamente una umbralización de cada canal independiente con el nivel proporcionado por el comando *graythresh*, un relleno y filtrado de los objetos pequeños para eliminar el ruido. Con esta operación, se consigue obtener el perfil de los objetos existentes en la imagen sin ruido.

A continuación, sobre la imagen de objetos sin ruido se realiza una composición con los tres canales mediante una AND lógica, con lo que se obtiene ya toda la información de cada canal filtrado independientemente en una sola imagen en B/N.

En la figura 4.7 podemos ver en imágenes, los pasos explicados anteriormente.

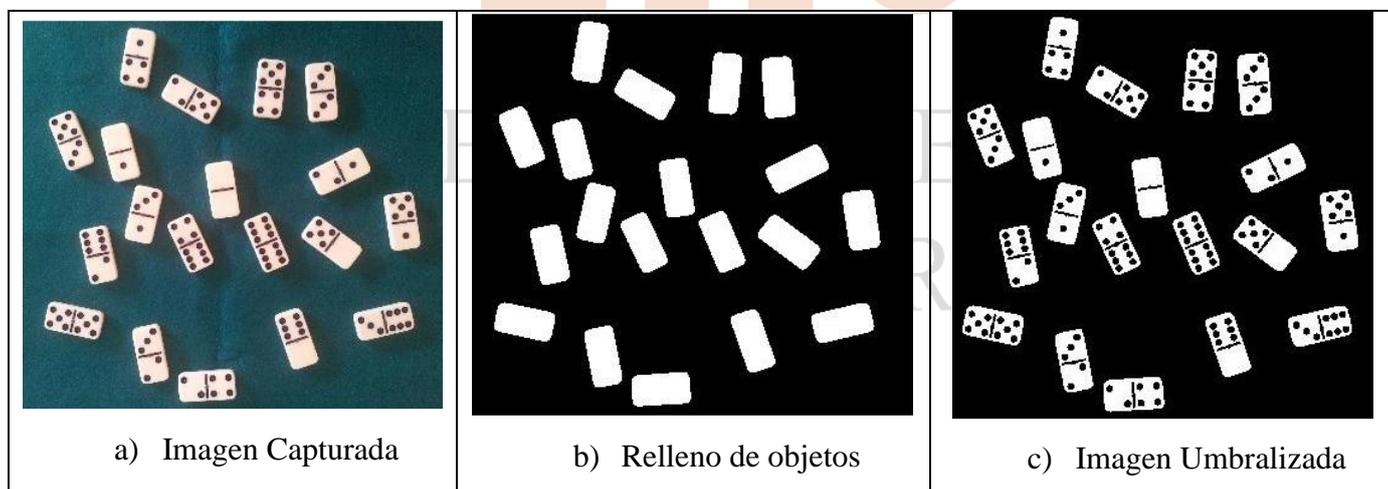


Figura 4.7: Proceso de Umbralización

Por último, antes de tener la imagen ya preparada para comenzar la segmentación se procede a invertir la imagen y eliminar el fondo, de manera que ya sólo quedan los puntos de las fichas y la línea de separación, que son los

objetos que interesan para identificar las fichas. En la figura 4.8 se puede ver el resultado obtenido por el bloque de adquisición y pre-procesado a partir de la imagen inicial.

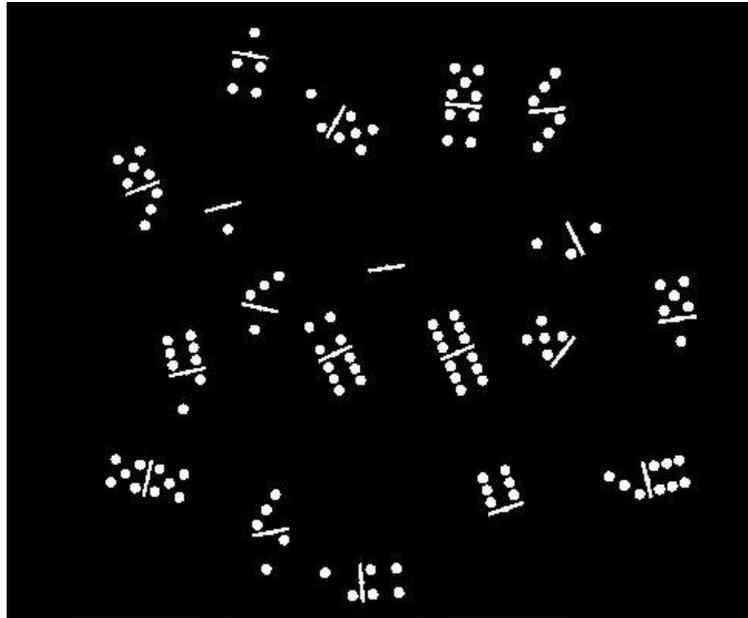


Figura 4.8: Imagen de salida bloque 1

A partir de la imagen anterior, ya se puede proceder a la localización de las fichas y su identificación, tal y como se explicará en el siguiente bloque.

ESCUELA POLITECNICA
SUPERIOR

4.4 LOCALIZACIÓN DE FICHAS

El segundo bloque del programa se va a encargar de realizar la localización de cada ficha dentro de la imagen. Para ello como ya se explicó anteriormente, se va a utilizar la línea de separación intermedia que existe en cada ficha de dominó y que es exactamente igual en todas ellas.

Para identificar y localizar las líneas de separación, se va a etiquetar y extraer las características de cada objeto de la imagen utilizando la función *regionprops*. Como la imagen a tratar, ya está compuesta exclusivamente de los puntos y las líneas, se aplicará el parámetro de excentricidad (*Eccentricity*) de *regionprops*, para identificar las líneas.

Por la relación de dimensiones que tiene la separación, los objetos con valores de excentricidad mayores a 0,95 se pueden identificar como las líneas de separación.

En principio, este bloque no realiza ninguna acción más y se podría empezar a realizar la identificación de las fichas del bloque 4. Pero tras analizar varias imágenes y debido al relieve de la “chincheta” que existe en el medio de la línea de separación (véase la figura 4.9), la iluminación de las fichas provoca reflejos sobre la superficie esférica de la “chincheta”.



Figura 4.9: Perfil de ficha

En ocasiones, dichos reflejos hacen que al umbralizar las imágenes, la separación de las fichas se “parta” en dos. En la figura 4.10 se puede ver un ejemplo.

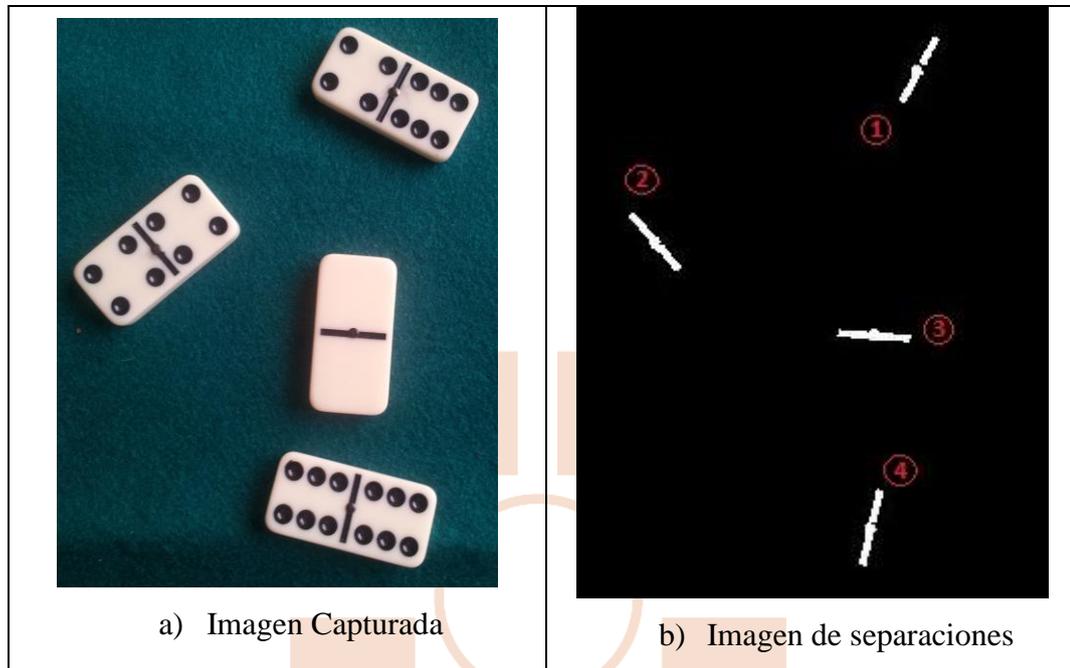


Figura 4.10: Perfil de ficha

A la izquierda aparece la imagen capturada y a la derecha está la imagen de las separaciones intermedias de las fichas que devuelve este bloque 2 de localización de fichas. En la imagen se han numerado las separaciones y tal y como se puede observar más claramente en la figura de detalle 4.11, debido a los reflejos, las separaciones 1 y 4 aparecen partidas.

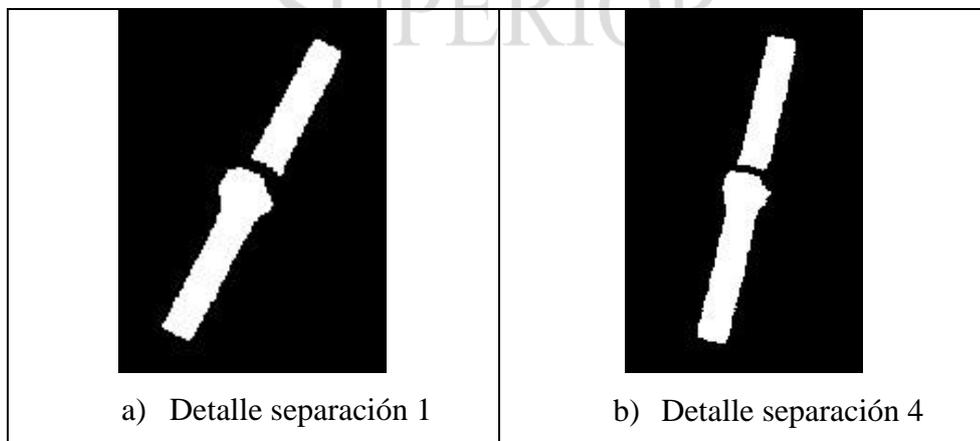
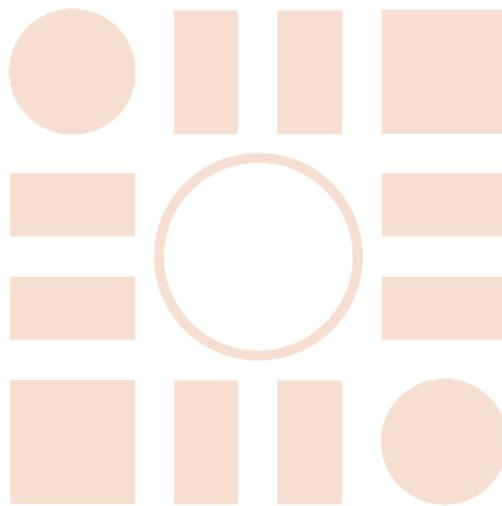


Figura 4.11: Detalle de separaciones

Este defecto, no permite una identificación correcta de la ficha como se explicará en el siguiente punto. Este es el motivo de la existencia del tercer bloque del programa que consistirá en realizar las operaciones necesarias para tratar de unir las dos partes de estas separaciones.



ESCUELA POLITECNICA
SUPERIOR

4.5 RECUPERACIÓN DE INFORMACIÓN

La existencia de este bloque del programa viene motivado por la intención de obtener un algoritmo lo más robusto posible, por lo que para solucionar los defectos puntuales producidos por el bloque 1 de adquisición y pre-procesado, se desarrolla el método expuesto a continuación para recuperar los píxeles perdidos que unen las separaciones de las fichas.

La importancia de que la separación de las fichas sea un único objeto y que tenga unas dimensiones adecuadas, se desarrollará más detenidamente en el último bloque. Ahora simplemente indicar que para lograr una correcta identificación de las fichas se deben cumplir esos requisitos.

Este bloque parte de la imagen entregada por el anterior, la cual contiene exclusivamente los objetos identificados como separaciones (figura 4.10 b) tanto las completas como las “partidas”.

Se etiquetan todos los objetos de la imagen y utilizando el parámetro “*MajorAxisLength*” y “*MinorAxisLength*” del comando *regionprops*, se obtienen las dimensiones de largo y ancho de la separación de mayor tamaño. Por tanto, es necesario que al menos exista en toda la imagen una separación que no se encuentre partida. Esta condición es imprescindible para que el sistema funcione correctamente, pero es una condición que habitualmente se cumple, ya que los casos en los que la separación aparece partida, son puntuales y es altamente improbable que todas las fichas que aparezcan en la imagen tengan ese defecto. Se establecen entonces estas dimensiones como las dimensiones patrón de la separación.

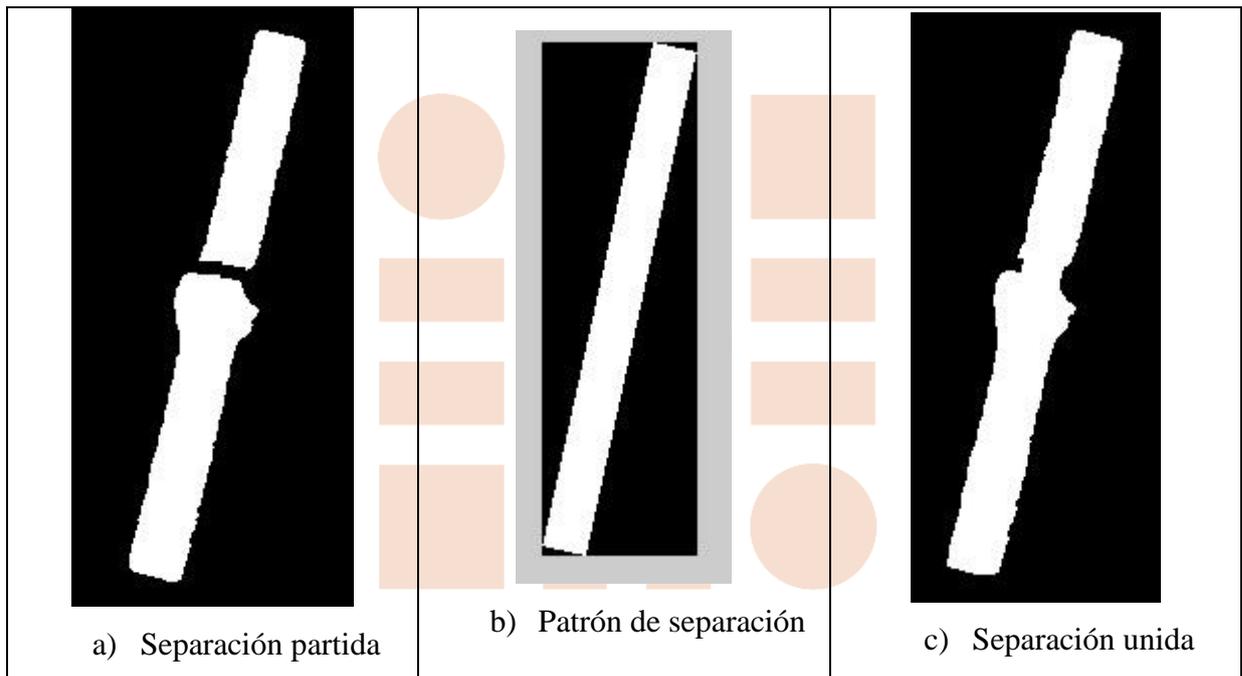
A continuación, teniendo en cuenta que la partición de la separación se produce en torno al punto medio, se realiza una selección de los objetos cuyo eje longitudinal mayor sea inferior al 70% del eje mayor del patrón y se toman en una imagen aparte. En la figura 4.12 se puede ver esta imagen correspondiente con el ejemplo mostrado en la figura 4.10.



Figura 4.12: Separaciones partidas

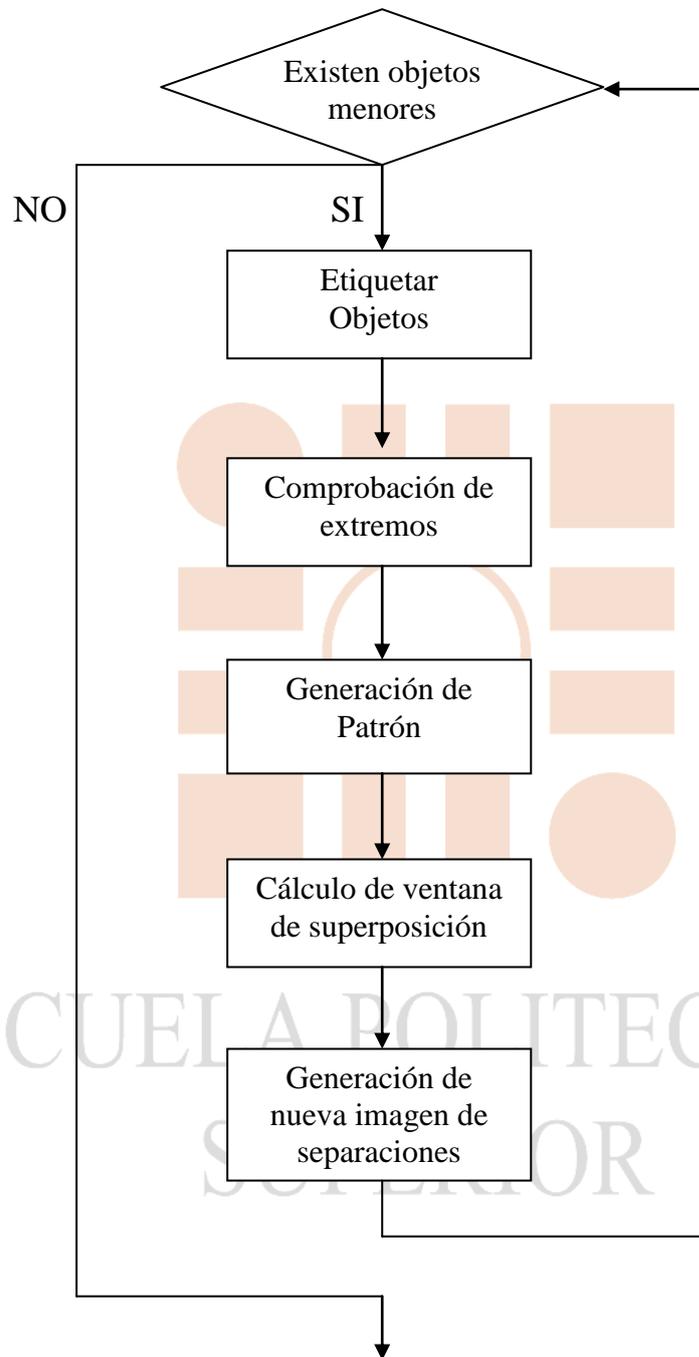
A partir de aquí, el programa entra en un bucle que se repite tantas veces como objetos hay en la imagen dividido entre dos, ya que cada dos objetos forma una sola separación.

La idea de este bloque consiste en que a partir de las dimensiones patrón de la separación, crear un objeto similar y superponerlo a la separación partida, de forma que queden unidas sin alterar sus características. Para ello, se deben conocer tanto la ubicación de las separaciones partidas como su orientación (que será prácticamente la misma en las dos partes). En la figura 4.13 se puede ver la secuencia de lo que se pretende hacer.



ESCUELA POLITÉCNICA
Figura 4.13: Proceso de unión
SUPERIOR

A continuación se puede observar el diagrama del bucle:



Continúa la ejecución del programa

A continuación se va a proceder a explicar cada una de las fases del bucle de unión de separaciones:

- Fase 1. Etiquetado de objetos: En esta fase inicial, simplemente se etiquetarán los objetos existentes en la imagen y se extraerán las características de extremos y ángulo de inclinación. Estos parámetros se obtienen con el comando *regionprops*, y proporcionan la información del siguiente modo (extraído de la ayuda de Matlab):
 - “*Orientation*”: Este parámetro es un número escalar con el valor del ángulo en grados decimales entre -90° y 90° . Para obtener la orientación, Matlab crea una elipse imaginaria que contiene al objeto (Figura 4.14).

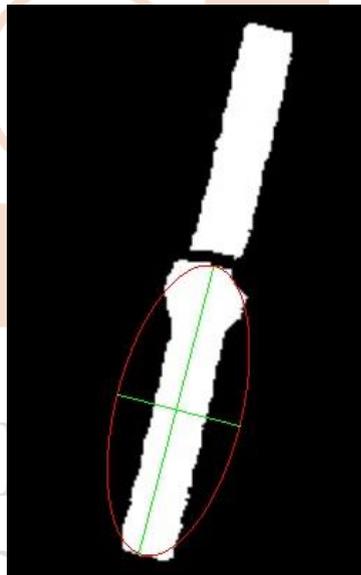


Figura 4.14: Separaciones partidas

En la figura anterior se puede observar en color rojo la elipse que contiene al objeto que forma parte de la separación, y en color verde los ejes de dicha elipse. Pues bien, el valor entregado por el parámetro *Orientation*, es el ángulo que forma el eje mayor de la elipse con la horizontal.

- “*Extrema*”: Se trata de una matriz de 8×2 elementos en la que se incluyen las coordenadas x e y de los ocho puntos que

definen los extremos del objeto, en el orden siguiente [top-left, top-right, right-top, right-bottom, bottom-right, bottom-left, left-bottom, left-top]. En la figura 4.15, se observa dónde corresponde cada punto.

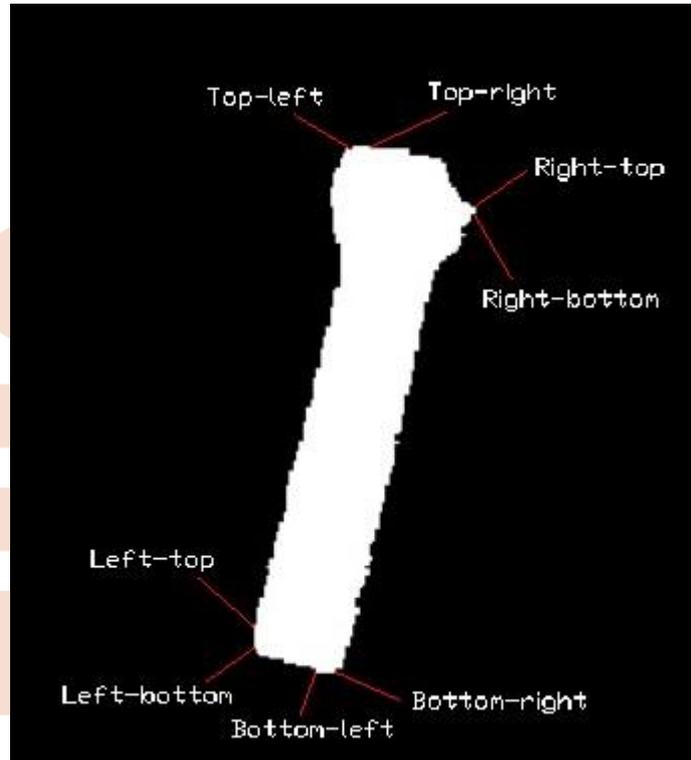


Figura 4.15: Puntos extremos

Para seleccionar el objeto con el que comenzar, simplemente se elige el primero de la imagen, ya que la forma que tiene Matlab de enumerar los objetos es comenzado desde la esquina superior izquierda ir recorriendo las filas de arriba a abajo y las columnas de izquierda a derecha. Por tanto, en principio el primer objeto que aparece etiquetado debería ser la esquina *left-top* de la separación que se pretende unir. Esto es así la mayor parte de las veces, pero se dan casos en los que no se cumple por los motivos que se exponen en la siguiente fase y los que motivan su existencia.

- Fase 3. Cálculo de ventana de superposición: Se va a adelantar la explicación de esta fase a la de la fase 2, ya que se trata del caso general y la fase 2 es una corrección que se realiza sobre algunos casos particulares detectados posteriormente.

Se va a analizar el caso de la figura 4.16, a modo de ejemplo de cómo realizar la superposición.

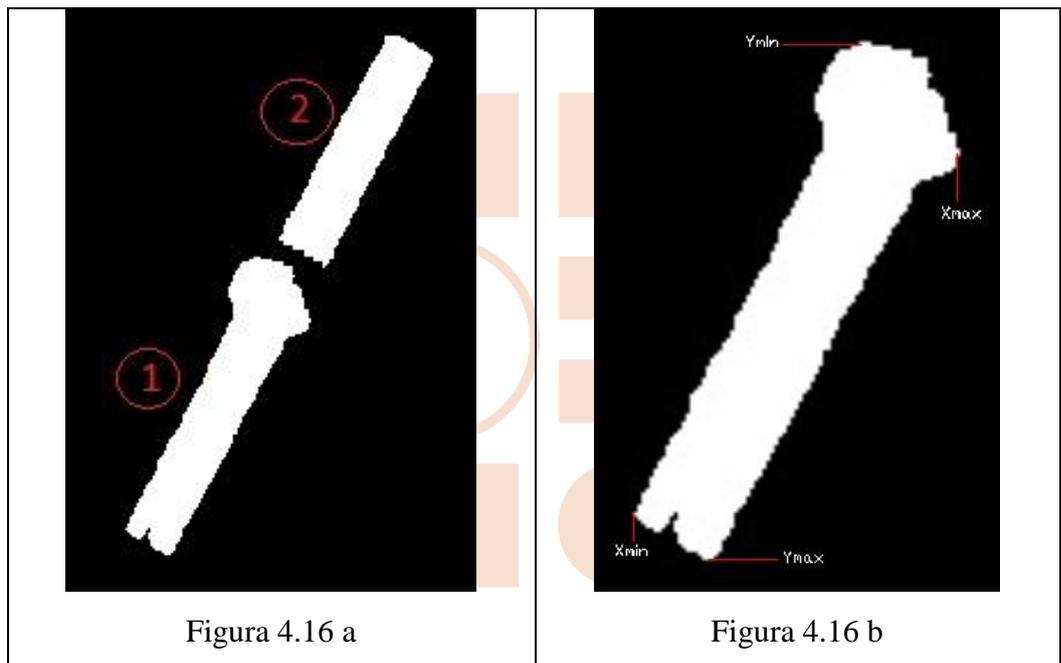


Figura 4.16: Imagen de separaciones a unir

Se selecciona el primer elemento de la imagen etiquetado, que se corresponde con el elemento 1, y se extraen los valores extremos y su orientación (fase 1).

En este caso, al tener un ángulo de orientación positivo, los puntos importantes para calcular la ventana de superposición serán Y_{max} y X_{min} . Por tanto la esquina inferior izquierda de la ventana será el punto (X_{min}, Y_{max}) .

A continuación se genera la separación a partir de las medidas patrón. Se crea una matriz de “unos” con las dimensiones

del patrón, y se gira utilizando el comando *imrotate*, tantos grados como tenga la orientación de la separación.

Queda calcular el punto superior derecho de la ventana de superposición, para lo cual se calcula el numero de filas y columnas que tiene la matriz de la separación patrón girada, y se suman/restan al punto de la esquina inferior izquierda de superposición. En la figura 4.17a se puede ver la imagen de la separación patrón, dispuesta para la superposición.



Figura 4.17: Imagen de separación patrón (a) y unida (b).

Con la medida de la ventana de superposición, ya tan solo queda realizar una OR entre la imagen de la separación patrón y la separación partida. El resultado se observa en la figura 4.17b.

En el caso de que la separación tenga un ángulo de orientación negativo, el procedimiento es similar con la excepción de que para obtener la ventana de superposición, los puntos a extraer de la imagen serán X_{min} e Y_{min} y corresponderán con la

esquina superior izquierda. A continuación con el tamaño de la imagen patrón, se obtendrá la esquina inferior derecha.

- Fase 2. Comprobación de extremos: El sistema ideado para unir las dos partes de la separación, consiste en la obtención de la ventana que contiene las dos partes y superponerla con la ventana que contiene al patrón, con la orientación adecuada. Como la orientación de la separación varía entre -90° y 90° , y el etiquetado de objetos en Matlab se realiza de izquierda a derecha, el primer objeto que se etiqueta contiene al extremo *left-top* de la separación. Con la excepción de los casos en los que el ángulo de orientación de la separación es muy grande, en los cuales debido al ensanchamiento en la parte central producido por la “chincheta” en función del punto donde se haya “partido” la separación, puede ser que se esté identificando la otra parte. En la figura 4.18 se ve claramente lo que se quiere explicar.

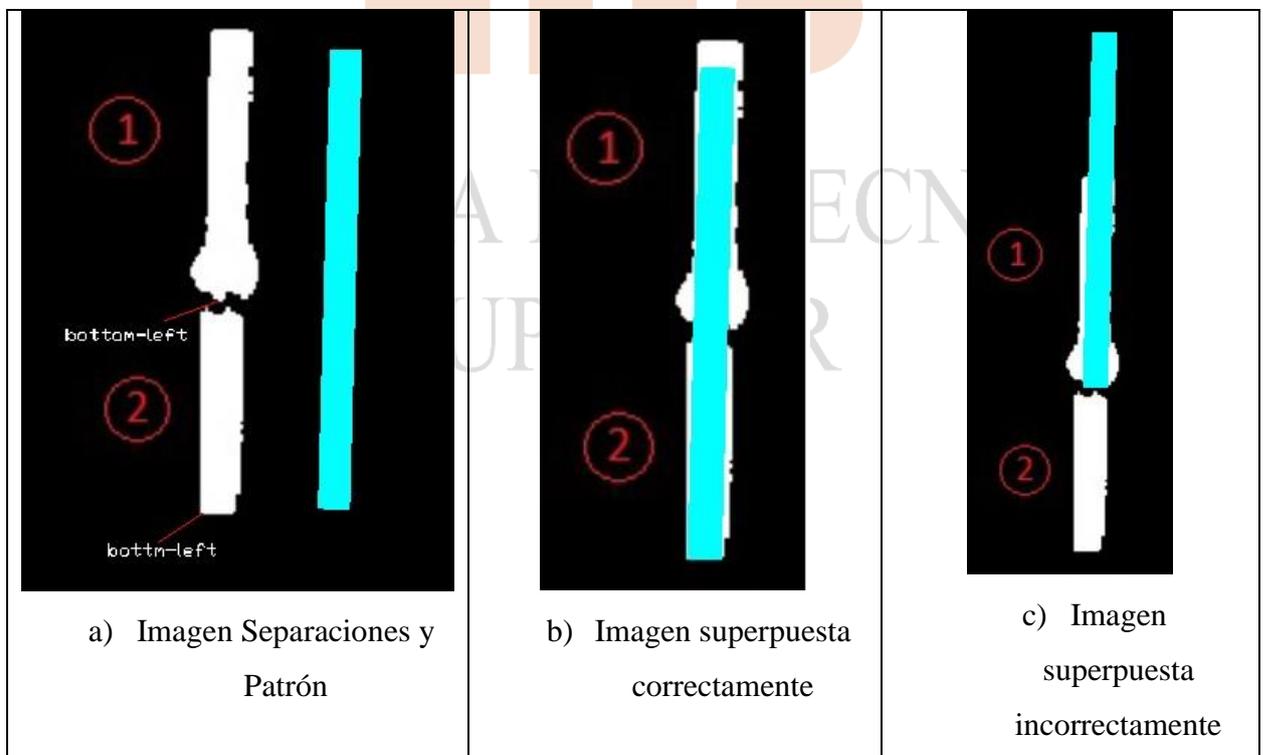


Figura 4.18: Selección de objetos para superposición

Si debido al ensanchamiento central se etiqueta primero el objeto 1, al tomar el punto “bottom-left” de ese objeto como esquina inferior izquierda de la ventana de superposición, el resultado se muestra en la figura 4.18c. Por tanto, si se quiere realizar la composición correctamente (Figura 4.18b), hay que tomar de referencia el objeto 2.

Por consiguiente, esta fase consistirá en analizar la imagen para identificar si el objeto seleccionado es el adecuado para realizar posteriormente la superposición.

Se establece un umbral de detección de este posible error para ángulos mayores a 85° en valor absoluto, ya que para ángulos menores no se puede producir este error y de esta forma se aligera la ejecución del algoritmo. Se va a explicar el proceso para ángulos negativos, teniendo en cuenta que para los positivos se seguirá un método análogo.

Los objetos a analizar, se muestran en la imagen 4.19a.

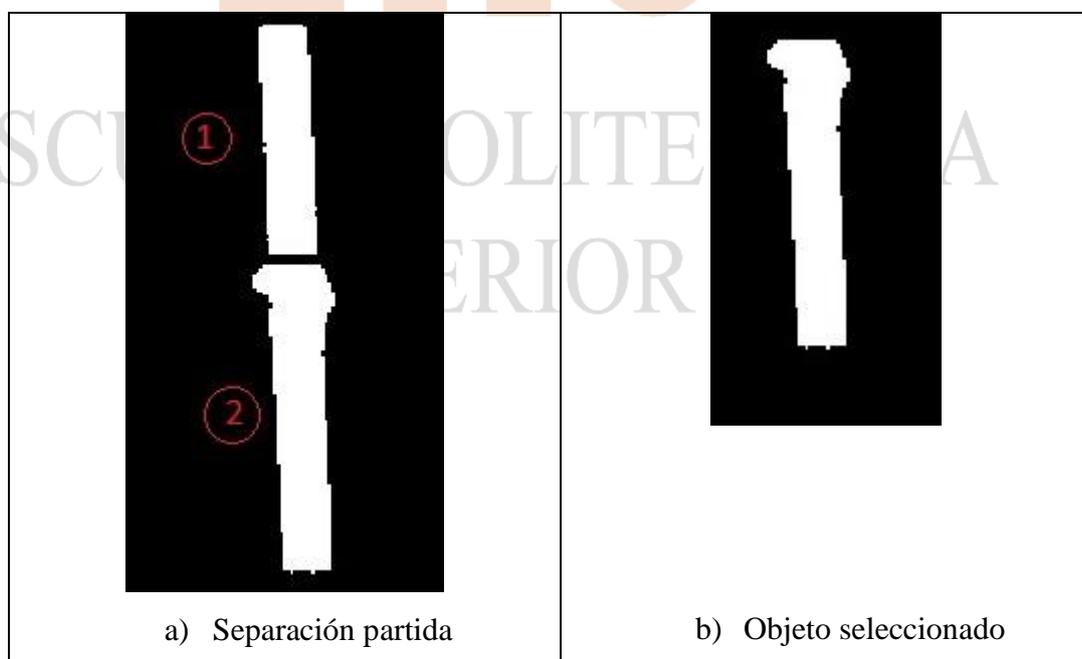


Figura 4.19: Objetos de búsqueda

En este caso, debida al ensanchamiento central, el primer objeto etiquetado es el 2 (figura 4.19b). Al tener un ángulo de orientación negativo (ángulo=-87,35°), inicialmente el programa, realizaría la superposición hacia la parte inferior del objeto, por lo que no se unirían los segmentos correctamente. Es por tanto necesario realizar la corrección de este bloque.

De la imagen 4.19b se obtienen sus extremos y se extraen los puntos X_{min} , X_{max} , Y_{min} , Y_{max} . La ventana delimitada por los puntos (X_{min}, Y_{min}) y (X_{max}, Y_{max}) , contendrá al objeto a analizar.

Se va a extraer una ventana de búsqueda de la imagen de separaciones para comprobar si el segmento que hay que unir se encuentra en el lado correcto. En este caso en el que la orientación tiene un valor negativo, la otra mitad del segmento a unir debería estar hacia abajo. En caso contrario, el segmento seleccionado no es el correcto.

Se establece el punto inferior derecho de la ventana de búsqueda en (X_{max}, Y_{max}) . Para el punto superior izquierdo de la ventana se utiliza el punto $(X_{min}, [Y_{min}-(Y_{max}-Y_{min})])$. En este caso en la imagen mostrada será similar a la 4.19a.

En el caso de que el segmento elegido hubiese sido el correcto, en la imagen de búsqueda tan sólo aparecería un objeto.

A continuación se realiza un etiquetado de objetos en la imagen obtenida y en el caso de que exista más de un objeto, se identifica el extremo *bottom-left* del otro segmento y se busca en la imagen de separaciones el segmento que está en esa posición y se establece como objeto a partir del cual obtener la ventana de superposición.

De esta forma se ha conseguido que el objeto a partir del cual realizar la superposición sea el objeto 1 de la imagen 4.19a, y no el objeto 2 como se había elegido inicialmente.

A partir de aquí, se continúa con la ejecución de la fase 3, tal y como ya se ha explicado.

- Fase 4. Generación de nueva imagen de separaciones: Para finalizar se realiza otra OR con la imagen de separaciones, se etiqueta de nuevo y se genera una nueva imagen con los objetos cuyo eje mayor, sea inferior al 70% del patrón.

De esta forma, una vez ejecutado el bucle anterior tantas veces como sean necesarias para unir todas las separaciones que se encuentren “partidas”, se realiza una última OR entre la imagen con las separaciones partidas y la imagen que se tenía de entrada al bloque 2 del programa, con lo que ya se obtiene una imagen sin “defecto” en las separaciones y lista para poder ser identificada en el bloque 4.

ESCUELA POLITECNICA
SUPERIOR

4.6 IDENTIFICACIÓN DE FICHAS

Tal y como ya se ha explicado, para realizar la identificación de las fichas se ha utilizado un método basado en la localización de las separaciones intermedias y a partir de estas, realizar una cuenta de los puntos existentes a cada lado de la separación.

Para ello, con la imagen entregada por el bloque 3 (véase un ejemplo en la figura 4.20b), se ejecuta el bucle que se explicará a continuación y con el cual se finalizará la ejecución del algoritmo con la identificación de la ficha.

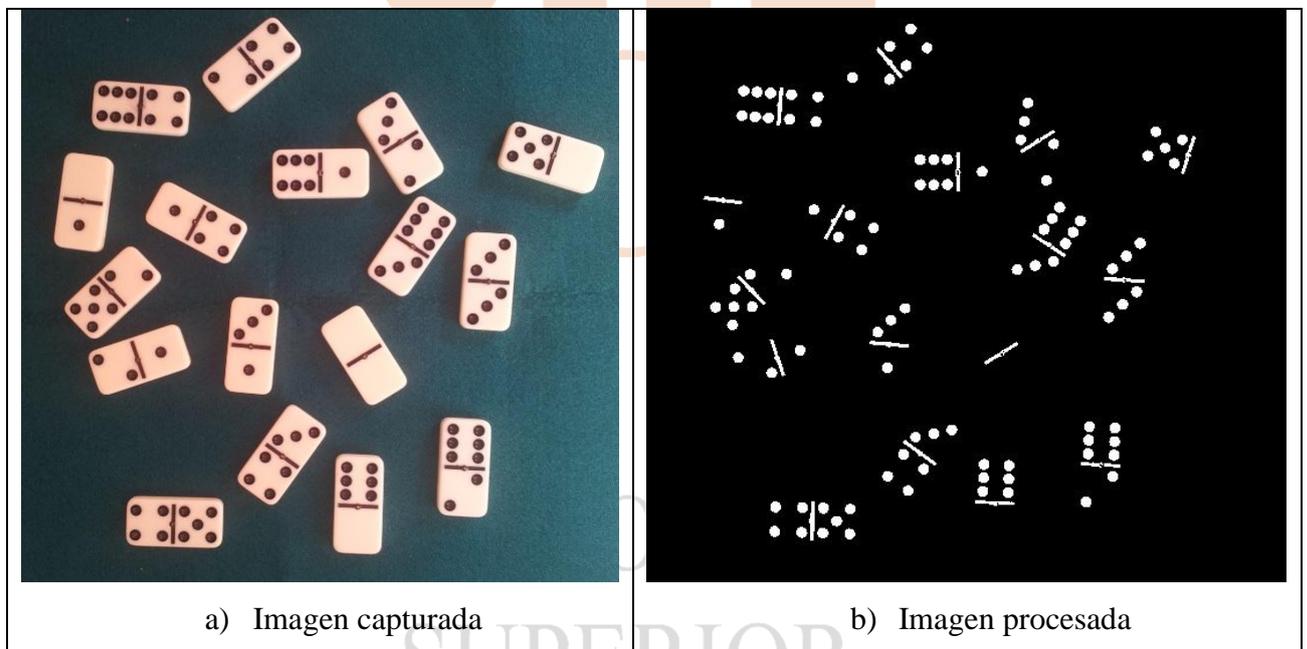


Figura 4.20: Imagen capturada (a) y entregado por bloque 3 (b)

Este bucle se estructura en tres partes fundamentales:

- Localización de la ficha
- Determinación del área de la ficha
- Identificación de la ficha

Además, en el apartado de localización de la ficha se ha añadido posteriormente una parte de verificación, debido a algunos errores que se

producían en la identificación. A continuación se va a proceder a explicar la implementación de cada uno de los apartados.

- Localización de la ficha: Para determinar el área de la imagen que ocupa la ficha, es necesario tener un punto de referencia de la separación y a partir de él calcular los puntos de cada esquina de la ficha. Para ello, a partir de los valores de extremos, orientación y longitud del eje mayor, entregados por el comando *regionprops* de la imagen de separaciones ya corregida se establece el ángulo α , el valor de L y el punto de referencia sobre el que aplicar los cálculos de los puntos geométricos, tal y como se explica más adelante.

Para determinar el punto de referencia, ya que la orientación de la separación varía entre -90° y 90° , se escogerá el punto *left-top*. Tan sólo será necesario realizar una comprobación para ángulos cercanos a ± 90 similar a la que se hacía en el bloque 3.

Para valores del ángulo por encima de 80° se puede dar la circunstancia de que el punto *left-top* se corresponda con la “chincheta” de la separación y no con el extremo (Figura 4.21).

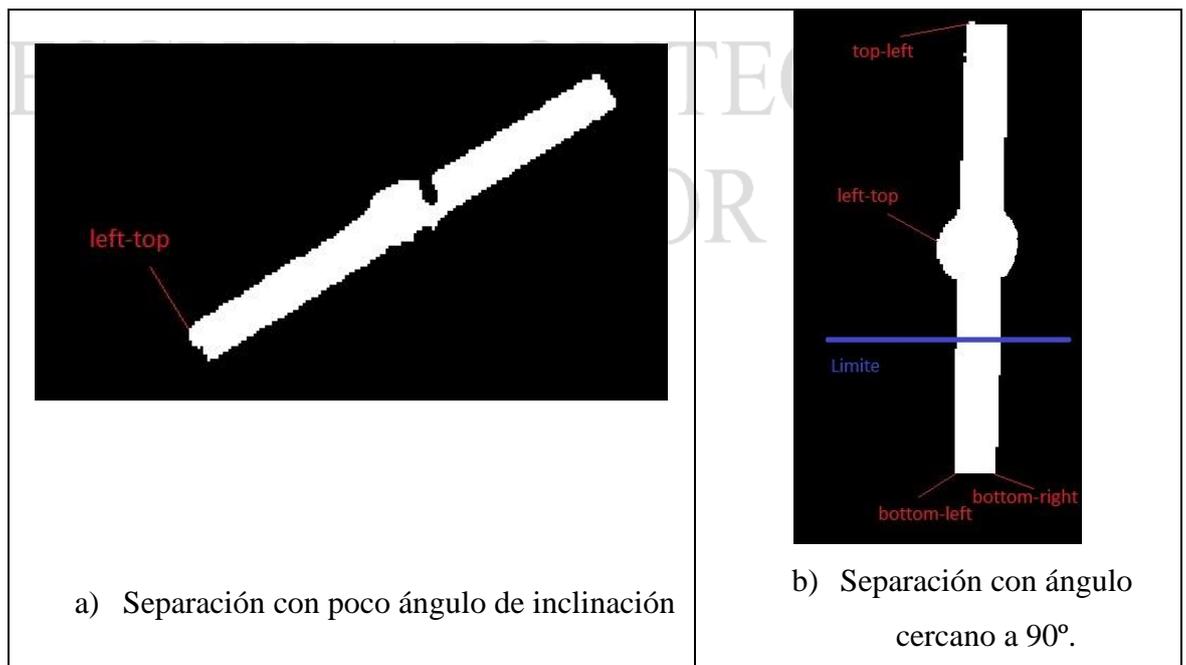


Figura 4.21: Imagen de error de identificación de punto de referencia

En la figura 4.21b se puede ver como la identificación del punto *left-top*, no se corresponde con el extremo de la separación. Por tanto para estos valores de ángulo, es necesario realizar una comprobación para determinar si el punto seleccionado es el adecuado.

Para comprobar que el punto es el correcto, en primer lugar hay que tener en cuenta que el eje y crece de arriba a abajo, es decir los puntos *top*, tienen un valor de *coordenada* y menor que los *bottom*. Por tanto se toman las *coordenadas* y de los puntos *top-left* y *bottom-right*, y a la cota de *top-left* se le suma un 70% de la diferencia entre ambas. Se establece esta cota como límite.

Si el valor de la coordenada y, del punto de referencia es menor que el límite; en el caso de ángulos positivos se escogerá como punto de referencia el *bottom-right*. Para ángulos negativos se elegirá el *top-left*.

Una vez comprobado que el punto de referencia es el correcto, ya se puede pasar al siguiente apartado.

- Determinación del área de la ficha: para determinar el área exacta ocupada por la ficha, en primer lugar se calculan los puntos geométricos A, B, C y D, tal y como ya se explica a continuación.

Para realizar el conteo de los puntos, es necesario en primer lugar delimitar el área que está ocupando la ficha en la imagen. Continuando con la idea de conseguir un sistema lo más robusto posible, se va a realizar el cálculo del área ocupada por la ficha a partir de las dimensiones de la separación intermedia, ya que dicha separación es idéntica en todas las fichas y de esta forma, se relativiza las dimensiones de la ficha obteniendo un resultado óptimo independientemente de la distancia o inclinación con que se tomen las

imágenes. Esta es la razón de tener unas dimensiones de las separaciones correctas, ya que son imprescindibles para la identificación. A continuación se expone el desarrollo geométrico realizado para calcular el área de la ficha en función de la separación.

En la figura 4.22 se muestran las dimensiones relativas de una ficha de dominó y los puntos geométricos que determinan su área.

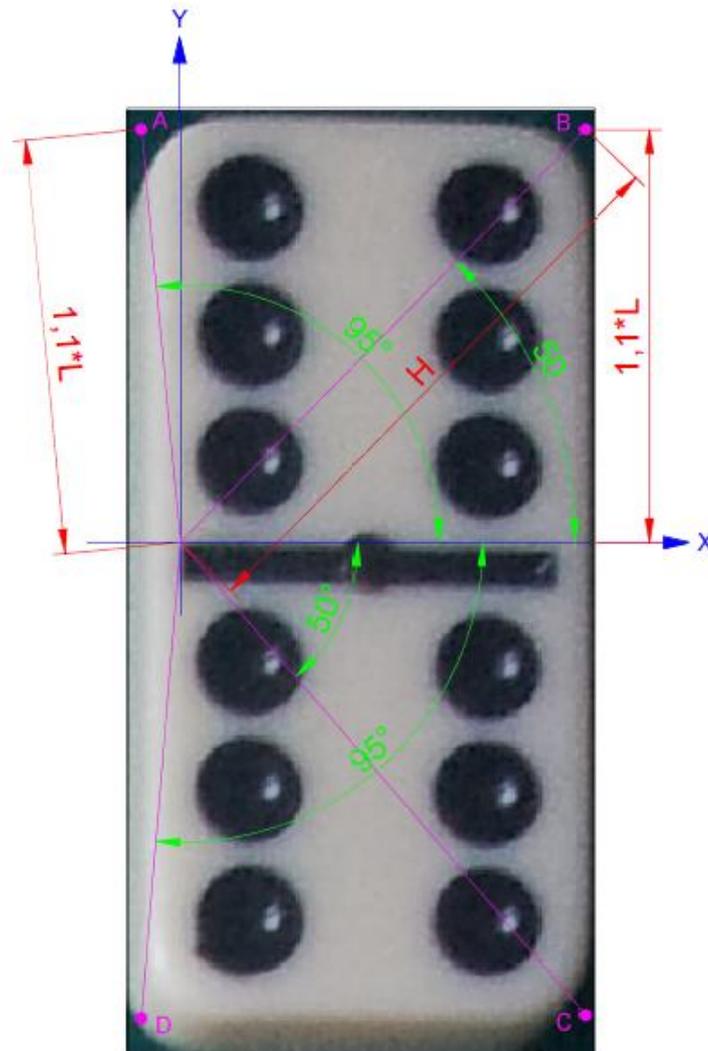


Figura 4.22: Dimensiones de ficha

Para el cálculo de las dimensiones se realiza una aproximación que es que la distancia desde el punto A hasta el centro de coordenadas es prácticamente igual a la coordenada y de dicho punto A, ya que el factor de corrección sería $\cos(5^\circ) = 0,996$,

por lo que se puede considerar despreciable. Teniendo en cuenta esta aproximación, los valores de las coordenadas de cada punto serán.

- Punto A: $A_x = 1,1 * L * \cos(95^\circ)$
 $A_y = 1,1 * L * \sen (95^\circ)$

- Punto B: $H = \sqrt{2 * (1,1 * L)^2}$
 $B_x = H * \cos(50^\circ)$

- Punto C: $B_y = H * \sen (50^\circ)$

- Punto C: $C_x = H * \cos(50^\circ)$
 $C_y = H * \sen (50^\circ)$

- Punto D: $D_x = 1,1 * L * \sen(95^\circ)$
 $D_y = 1,1 * L * \cos (95^\circ)$

Estos valores están calculados teniendo en cuenta que el ángulo de inclinación de la separación con respecto al eje x es de 0° . Como las fichas de dominó pueden estar orientadas en cualquier ángulo, es necesario incluir este ángulo de orientación (α) de la separación de la ficha con respecto al eje x en las ecuaciones.

Teniendo esto en cuenta, finalmente las ecuaciones implementadas para obtener los puntos geométricos de las fichas son:

- Punto A: $A_x = 1,1 * L * \cos(\alpha + 95^\circ)$
 $A_y = 1,1 * L * \sen (\alpha + 95^\circ)$

- Punto B: $H = \sqrt{2 * (1,1 * L)^2}$

$$B_x = H \cdot \cos(\alpha + 50^\circ)$$

$$B_y = H \cdot \sin(\alpha + 50^\circ)$$

- Punto C: $C_x = H \cdot \cos(\alpha - 50^\circ)$

$$C_y = H \cdot \sin(\alpha - 50^\circ)$$

- Punto D: $D_x = 1,1 \cdot L \cdot \sin(\alpha - 95^\circ)$

$$D_y = 1,1 \cdot L \cdot \cos(\alpha - 95^\circ)$$

A partir de las coordenadas x e y de los puntos geométricos se obtienen los límites de la ventana que contiene a la ficha en cuestión, eligiendo la coordenada máxima y mínima de cada eje (x e y).

Esta ventana es un rectángulo que se recorta de la imagen original, con lo que si la ficha está inclinada y se encuentran varias juntas, pueden aparecer en ella objetos que no pertenecen al área de la ficha. En la figura 4.23 se muestra un ejemplo.

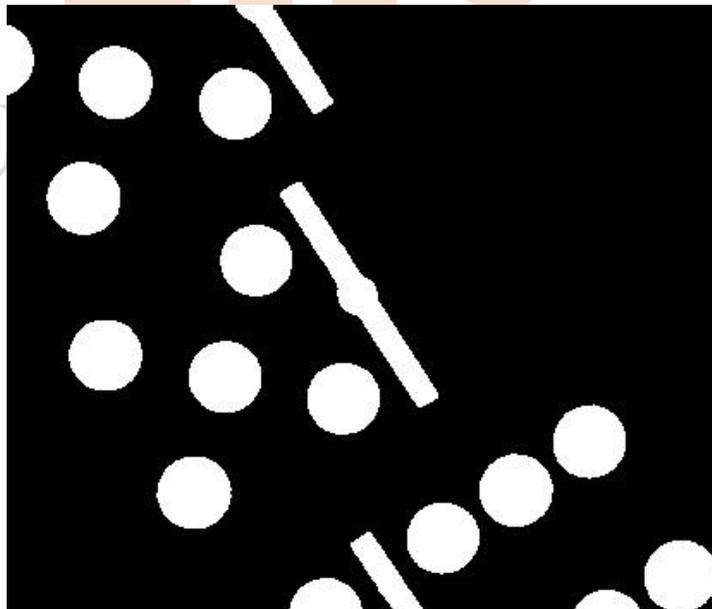


Figura 4.23: Imagen recortada

En la imagen recortada se realiza un nuevo etiquetado de objetos, con el objetivo de determinar cuál de todos los que hay en la imagen es la separación de la ficha que se quiere identificar.

Para ello, de entre todos los objetos se seleccionan los que tienen aspecto de línea y de estos se coge el que tenga un valor de eje mayor igual al de la ficha que estamos identificando.

Una vez identificada la separación de la ficha, se rota la imagen el ángulo indicado en el campo *Orientation* de la separación para tener la ficha orientada verticalmente. De esta forma ya se puede obtener una ventana que contenga única y exclusivamente el área de la ficha.

Sobre la imagen rotada se etiquetan de nuevo los objetos y se obtiene la *BoundingBox* que contiene a la separación. Con la referencia de la *BoundingBox* y las dimensiones de la separación, se obtiene el área exacta que contiene a la ficha. En este momento ya se está en disposición de identificar la ficha.

- Identificación de la ficha: Como último paso del bloque 4 de identificación de fichas, será el propio reconocimiento.

Ya se ha tratado la imagen para obtener exclusivamente el área que contiene a la ficha, debidamente filtrada para que aparezcan solamente los objetos necesarios para su correcta identificación.

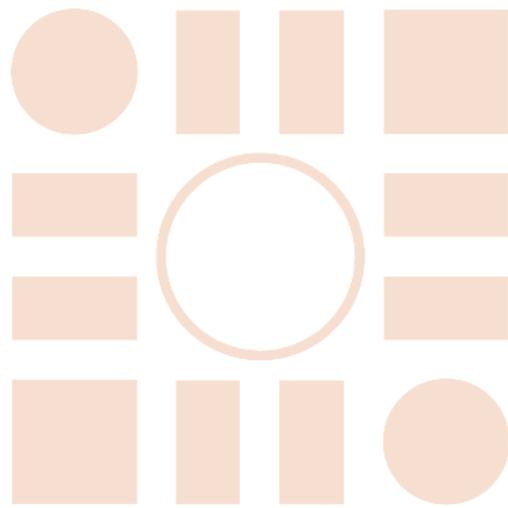
Se etiquetan los objetos de la imagen, y se localiza la separación intermedia. Al estar la ficha en posición vertical, se toma como referencia la coordenada 'y' de la *BoundingBox* que contiene a la separación y se inicializan dos contadores de 'puntos'. Uno para los que se encuentren en la parte superior de la separación y otro para la inferior.

Seguidamente se ejecuta un bucle que recorre todos los elementos de la imagen en el cual en primer lugar comprueba que se trata de un 'punto', y a continuación se comprueba la coordenada 'y' donde se encuentra. Para saber si el elemento que se estudia es un punto, previamente se ha añadido un campo a la estructura entregada por *regionprops* llamado relación de ejes, en el que se calcula el cociente entre *MinorAxisLength* y *MajorAxisLength*. Si dicha relación es mayor de 0,7 se considerará que es un punto. A continuación se comprueba la coordenada y proporcionada por la *BoundingBox* y en función de si se encuentra por encima o por debajo de la separación, se incrementará un contador u otro.

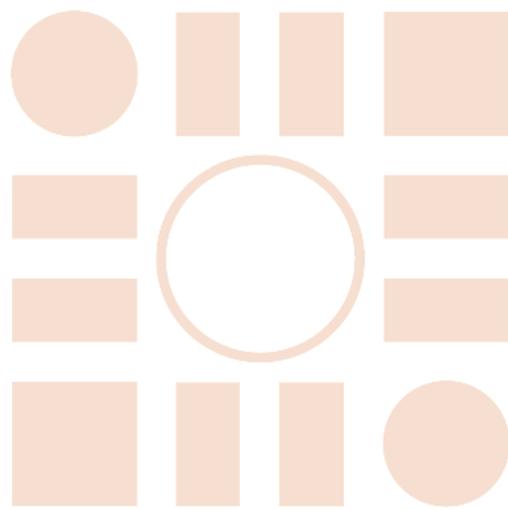
Por último, se identifica la ficha en función del número de puntos que se hayan contado a cada lado de la separación.

Con esto se finaliza la ejecución del cuarto bloque de la función, que se encuentra contenido dentro de un bucle que se ejecuta tantas veces como número de separaciones existen en la imagen entregada por el bloque tres.

En el siguiente capítulo se va a dar cuenta de los resultados obtenidos con el algoritmo que se ha implementado.



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA SUPERIOR

5.1 CONDICIONES DE ADQUISICIÓN

Tal y como sucede en la mayoría de los sistemas de visión artificial, este sistema necesita unas condiciones de adquisición adecuadas para conseguir un funcionamiento correcto. En este apartado se va a tratar de delimitar bajo qué condiciones el sistema es capaz de obtener un resultado óptimo.

El sistema está diseñado para identificar fichas de dominó en entornos de interior cuya iluminación sea más o menos uniforme y en imágenes en las que tan sólo aparezcan las fichas de dominó completas sobre un tapete de color verde.

Estas premisas, se han establecido por las razones que se exponen a continuación:

- La condición de que una ficha debe aparecer completa es consecuencia de que para que una ficha pueda ser identificada correctamente, debe de aparecer tanto la separación como todos los puntos de la ficha, ya que se utiliza la separación para delimitar el área que ocupa la ficha y se realiza una cuenta de los puntos para identificarla, con lo que si falta alguno la identificación será errónea.
- Otra limitación, está en que las imágenes deben tomarse en el interior y con una iluminación más o menos uniforme. Esto es debido a que si existe una zona iluminada por ejemplo con la luz del sol y en una zona de la imagen aparece una sombra, la diferencia de niveles tan grande hace que el comando *graythresh* con el que se establece el nivel de corte para la umbralización, se encuentre en un valor excesivamente alto y no se umbralice correctamente.
- Un aspecto importante es el tratamiento de los reflejos sobre las fichas producidos por fuentes de iluminación puntuales. Las fichas utilizadas tal y como se muestran en la figura 5.1 tienen relieve, de

forma que los puntos son semiesferas huecas y la separación son semicilindros huecos. Esta forma de fabricar las fichas, es bastante común con lo que el efecto de los reflejos se producirá en todos los juegos de dominó realizados así.

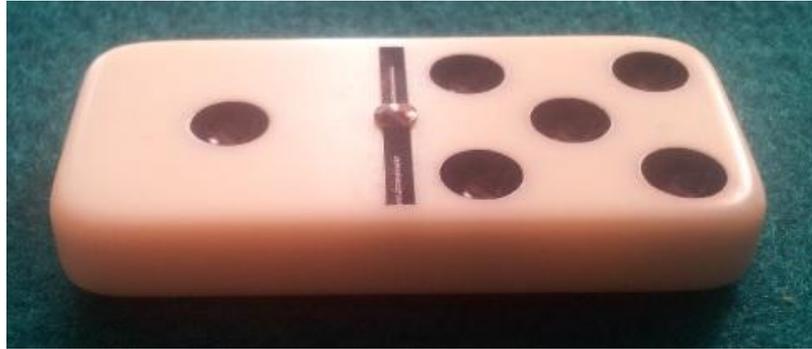


Figura 5.1: Imagen de fichas utilizadas

Las superficies esféricas hacen que se produzcan siempre reflejos en los objetos que conforman la ficha. En el caso de que estos reflejos sean muy intensos, debido al tipo de filtrado que se aplica en el bloque de pre-procesado, se pueden llegar a eliminar elementos de la ficha. En la figura 5.2 se puede ver un ejemplo:

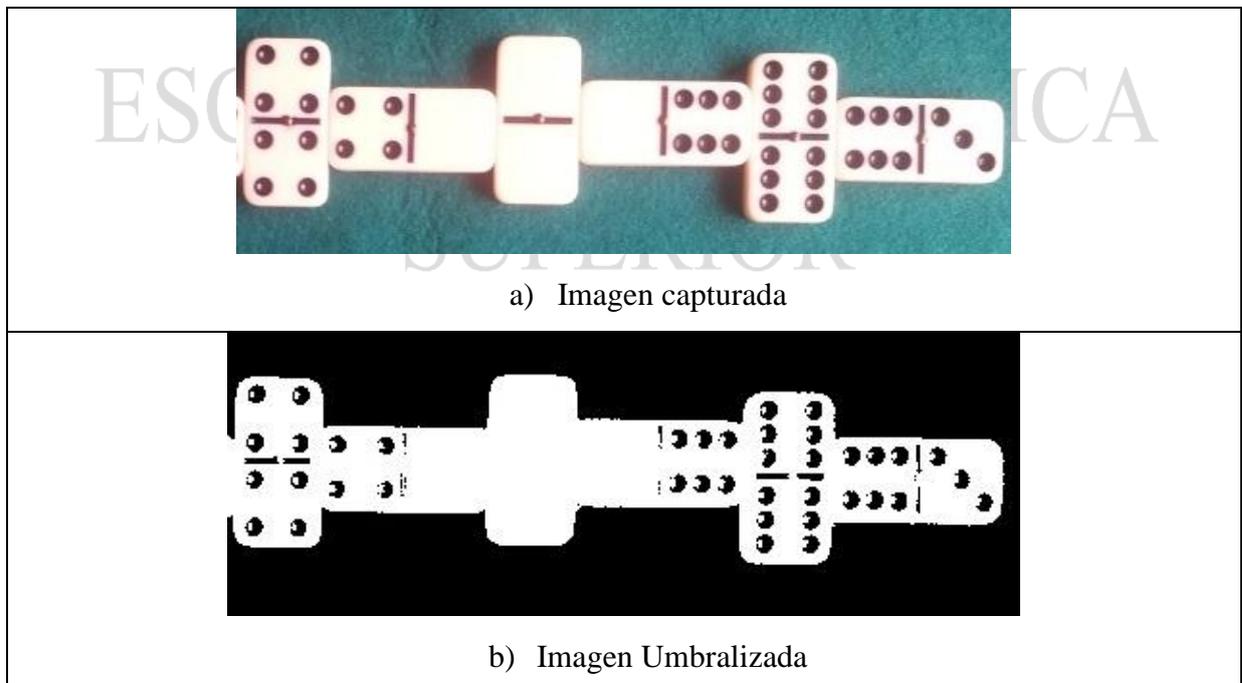


Figura 5.2: Ejemplo de imagen con reflejos

En este caso se puede ver que la imagen está iluminada por una fuente puntual en la parte derecha, lo que produce grandes reflejos en las superficies esféricas de las fichas. En la figura 5.3 se puede ver la imagen a tratar.

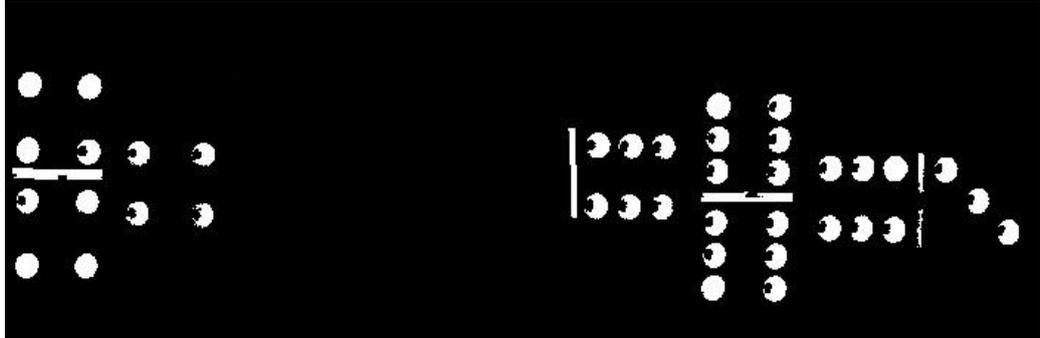


Figura 5.3: Ejemplo de imagen con reflejos procesada

Tal y como se puede apreciar, una vez realizado el procesado de la imagen, a pesar de existir una mejora de la imagen, se han perdido completamente algunas separaciones, con lo que es imposible identificar esas fichas.

Por tanto es necesario que no existan en las imágenes de adquisición unos reflejos excesivos que hagan desaparecer por completo algún objeto de la ficha. Esta limitación se podrá eliminar en posteriores proyectos que mejoren del proceso de umbralización de la imagen.

- En la imagen a tratar no pueden existir objetos ajenos a las fichas cuya excentricidad sea cercana a uno (es decir que tengan una relación de aspecto similar a la de la separación) ya que de esta forma, se identificaría esa zona de la imagen como una ficha de dominó.
- El enfoque en cambio no supone una limitación, ya que para imágenes desenfocadas, el sistema ha realizado una identificación óptima.

- Tampoco la distancia entre el objetivo de la cámara y las fichas influye en la identificación, ya que se realiza de manera óptima en imágenes tan dispares como las mostradas en la figura 5.4.

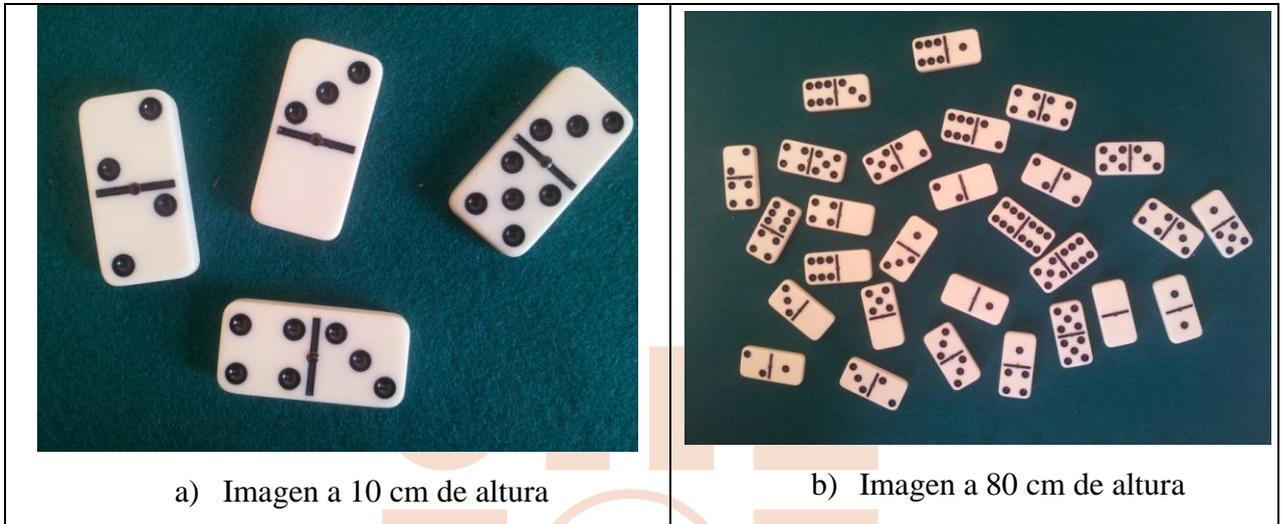


Figura 5.4: Ejemplo de imagen a distintas alturas

- El ángulo de inclinación del foco de la cámara con respecto a la vertical también es un factor importante, ya que a medida que se aumenta, los objetos empiezan a tomar perspectiva y su relación de aspecto cambia. Se han realizado pruebas y hasta inclinaciones aproximadas de $\pm 30^\circ$ con respecto a la vertical, el sistema funciona correctamente.

ESCUELA POLITÉCNICA
SUPERIOR

5.2 RESULTADOS

En este capítulo va a dar cuenta de los resultados obtenidos con el algoritmo desarrollado y siempre que se cumplan las condiciones de adquisición establecidas en el punto anterior.

Los ensayos se han dividido en tres grandes grupos, intentando dar cabida a todas las posibilidades que se pueden dar.

El primer grupo está compuesto por imágenes en las que las fichas aparecen todas separadas unas de otras, distribuidas y orientadas de forma aleatoria en la imagen. En la figura 5.5 se muestra un ejemplo de imagen de este grupo.



Figura 5.5: Ejemplo de imagen de fichas separadas

Se han tomado imágenes a diferentes distancias y con distintas inclinaciones para comprobar la robustez del algoritmo.

El segundo grupo de imágenes consiste en imágenes con las fichas distribuidas de forma aleatoria, pero en las que unas fichas se tocan con otras de forma que al umbralizar, forman un solo objeto. Este grupo es un paso intermedio entre las fichas totalmente separadas y el tercer grupo que será con todas las fichas unidas en un solo objeto. En la figura 5.6 se puede ver un ejemplo de imagen de este grupo.



Figura 5.6: Ejemplo de imagen de fichas juntas

Por último se ha experimentado con un tercer grupo que consiste en imágenes colocadas a modo de juego, de forma que todas las fichas se encuentran unidas. En la figura 5.7 se puede ver un ejemplo.

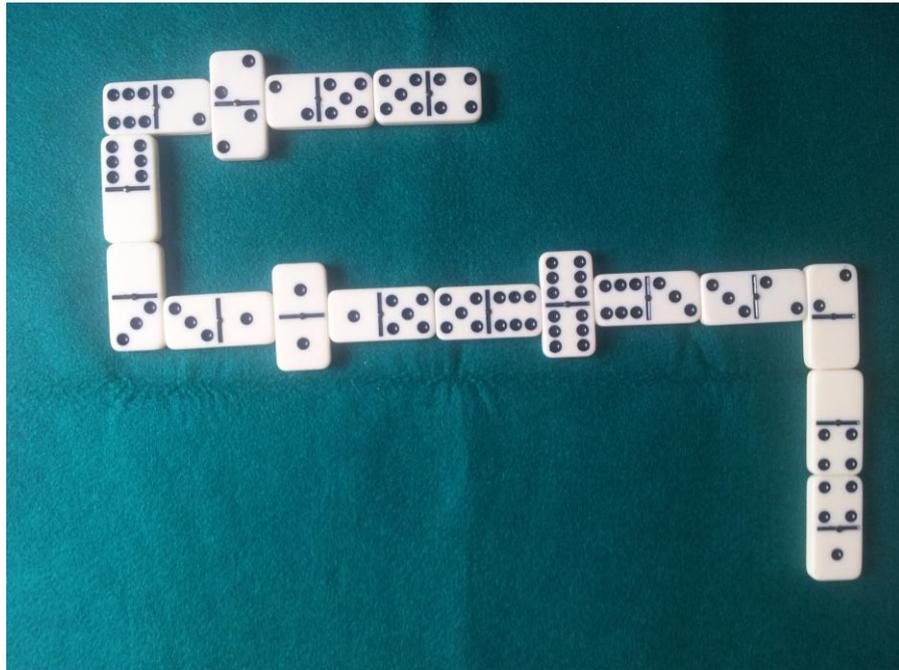


Figura 5.7: Ejemplo de imagen de fichas modo juego

A continuación se va a exponer de forma resumida los resultados de los ensayos obtenidos en cada uno de los grupos, bajo las condiciones de adquisición establecidas en el apartado 5.1:

- Imágenes con fichas separadas: Para este grupo, se han analizado cincuenta imágenes en las cuales aparecían un total de 801 fichas de dominó. En el siguiente cuadro se muestra una tabla con los resultados obtenidos.

| Nombre de Archivo | Numero de Fichas | Aciertos | Fallos |
|-------------------|------------------|----------|--------|
| Separadas 1 | 3 | 3 | 0 |
| Separadas 2 | 7 | 7 | 0 |
| Separadas 3 | 17 | 17 | 0 |
| Separadas 4 | 28 | 28 | 0 |
| Separadas 5 | 11 | 11 | 0 |
| Separadas 6 | 9 | 9 | 0 |
| Separadas 7 | 13 | 13 | 0 |
| Separadas 8 | 6 | 6 | 0 |
| Separadas 9 | 19 | 19 | 0 |
| Separadas 10 | 2 | 2 | 0 |

| | | | |
|--------------|-------|-----|---|
| Separadas 11 | 28 | 28 | 0 |
| Separadas 12 | 1 | 1 | 0 |
| Separadas 13 | 13 | 13 | 0 |
| Separadas 14 | 4 | 4 | 0 |
| Separadas 15 | 3 | 3 | 0 |
| Separadas 16 | 28 | 28 | 0 |
| Separadas 17 | 15 | 15 | 0 |
| Separadas 18 | 20 | 20 | 0 |
| Separadas 19 | 22 | 22 | 0 |
| Separadas 20 | 18 | 18 | 0 |
| Separadas 21 | 28 | 28 | 0 |
| Separadas 22 | 20 | 20 | 0 |
| Separadas 23 | 28 | 28 | 0 |
| Separadas 24 | 10 | 10 | 0 |
| Separadas 25 | 11 | 11 | 0 |
| Separadas 26 | 24 | 24 | 0 |
| Separadas 27 | 21 | 21 | 0 |
| Separadas 28 | 23 | 23 | 0 |
| Separadas 29 | 21 | 21 | 0 |
| Separadas 30 | 28 | 28 | 0 |
| Separadas 31 | 11 | 11 | 0 |
| Separadas 32 | 22 | 22 | 0 |
| Separadas 33 | 20 | 20 | 0 |
| Separadas 34 | 15 | 15 | 0 |
| Separadas 35 | 28 | 28 | 0 |
| Separadas 36 | 17 | 17 | 0 |
| Separadas 37 | 26 | 26 | 0 |
| Separadas 38 | 28 | 28 | 0 |
| Separadas 39 | 16 | 16 | 0 |
| Separadas 40 | 15 | 15 | 0 |
| Separadas 41 | 4 | 4 | 0 |
| Separadas 42 | 12 | 12 | 0 |
| Separadas 43 | 12 | 12 | 0 |
| Separadas 44 | 16 | 16 | 0 |
| Separadas 45 | 8 | 8 | 0 |
| Separadas 46 | 12 | 12 | 0 |
| Separadas 47 | 16 | 16 | 0 |
| Separadas 48 | 14 | 14 | 0 |
| Separadas 49 | 12 | 12 | 0 |
| Separadas 50 | 16 | 16 | 0 |
| TOTAL | 801 | 801 | 0 |
| % DE ACIERTO | 100 % | | |

Tal y como se puede ver, se han obtenido unos resultados inmejorables. En la adquisición de las imágenes se ha procurado alternar distintas opciones de iluminación y sin seguir un método específico, con la intención de que los resultados sean lo más fiables posible.

Se han analizado un total de 801 imágenes de fichas, teniendo en cuenta que tan solo hay 28 fichas diferentes, se obtiene que de media cada ficha se haya procesado en casi 29 configuraciones diferentes de posición, orientación e iluminación.

- Imágenes con fichas juntas: Para este grupo, se han analizado sesenta y una imágenes en las cuales aparecían un total de 1099 fichas de dominó. En el siguiente cuadro se muestra una tabla con los resultados obtenidos.

| Nombre de Archivo | Numero de Fichas | Aciertos | Fallos |
|-------------------|------------------|----------|--------|
| Juntas 1 | 17 | 17 | 0 |
| Juntas 2 | 24 | 24 | 0 |
| Juntas 3 | 18 | 18 | 0 |
| Juntas 4 | 8 | 8 | 0 |
| Juntas 5 | 25 | 25 | 0 |
| Juntas 6 | 20 | 20 | 0 |
| Juntas 7 | 14 | 14 | 0 |
| Juntas 8 | 21 | 21 | 0 |
| Juntas 9 | 23 | 23 | 0 |
| Juntas 10 | 18 | 18 | 0 |
| Juntas 11 | 23 | 23 | 0 |
| Juntas 12 | 19 | 19 | 0 |
| Juntas 13 | 22 | 22 | 0 |
| Juntas 14 | 28 | 28 | 0 |
| Juntas 15 | 24 | 24 | 0 |
| Juntas 16 | 15 | 15 | 0 |
| Juntas 17 | 12 | 12 | 0 |
| Juntas 18 | 25 | 25 | 0 |
| Juntas 19 | 14 | 14 | 0 |
| Juntas 20 | 23 | 23 | 0 |
| Juntas 21 | 23 | 23 | 0 |
| Juntas 22 | 19 | 19 | 0 |
| Juntas 23 | 28 | 28 | 0 |

| | | | |
|-----------|----|----|---|
| Juntas 24 | 25 | 25 | 0 |
| Juntas 25 | 16 | 16 | 0 |
| Juntas 26 | 16 | 16 | 0 |
| Juntas 27 | 15 | 15 | 0 |
| Juntas 28 | 14 | 14 | 0 |
| Juntas 29 | 14 | 14 | 0 |
| Juntas 30 | 23 | 23 | 0 |
| Juntas 31 | 19 | 19 | 0 |
| Juntas 32 | 28 | 28 | 0 |
| Juntas 33 | 20 | 20 | 0 |
| Juntas 34 | 24 | 24 | 0 |
| Juntas 35 | 8 | 8 | 0 |
| Juntas 36 | 14 | 14 | 0 |
| Juntas 37 | 14 | 14 | 0 |
| Juntas 38 | 18 | 18 | 0 |
| Juntas 39 | 21 | 21 | 0 |
| Juntas 40 | 22 | 22 | 0 |
| Juntas 41 | 20 | 20 | 0 |
| Juntas 42 | 20 | 20 | 0 |
| Juntas 43 | 17 | 17 | 0 |
| Juntas 44 | 26 | 26 | 0 |
| Juntas 45 | 14 | 14 | 0 |
| Juntas 46 | 22 | 22 | 0 |
| Juntas 47 | 14 | 14 | 0 |
| Juntas 48 | 10 | 10 | 0 |
| Juntas 49 | 14 | 14 | 0 |
| Juntas 50 | 8 | 8 | 0 |
| Juntas 51 | 18 | 18 | 0 |
| Juntas 52 | 8 | 8 | 0 |
| Juntas 53 | 10 | 10 | 0 |
| Juntas 54 | 12 | 12 | 0 |
| Juntas 55 | 18 | 18 | 0 |
| Juntas 56 | 16 | 16 | 0 |
| Juntas 57 | 15 | 15 | 0 |
| Juntas 58 | 13 | 13 | 0 |
| Juntas 59 | 3 | 3 | 0 |
| Juntas 60 | 28 | 28 | 0 |
| Juntas 61 | 19 | 19 | 0 |

TOTAL 1099 1099 0

PORCENTAJE DE ACIERTO 100 %

De nuevo, se han obtenido unos resultados inmejorables. Se han analizado un total de 1099 imágenes de fichas, obteniendo una media de procesamiento de cada ficha de más de 39, realizados en diferentes condiciones de adquisición.

- Imágenes con fichas tipo juego: Para este grupo, se han analizado treinta y siete imágenes en las cuales aparecían un total de 577 fichas de dominó. En el siguiente cuadro se muestra una tabla con los resultados obtenidos.

| Nombre de Archivo | Numero de Fichas | Aciertos | Fallos |
|-------------------|------------------|----------|--------|
| Juego 1 | 10 | 10 | 0 |
| Juego 2 | 10 | 10 | 0 |
| Juego 3 | 12 | 12 | 0 |
| Juego 4 | 12 | 12 | 0 |
| Juego 5 | 15 | 15 | 0 |
| Juego 6 | 15 | 15 | 0 |
| Juego 7 | 15 | 15 | 0 |
| Juego 8 | 18 | 18 | 0 |
| Juego 9 | 16 | 16 | 0 |
| Juego 10 | 10 | 10 | 0 |
| Juego 11 | 16 | 16 | 0 |
| Juego 12 | 16 | 16 | 0 |
| Juego 13 | 16 | 16 | 0 |
| Juego 14 | 19 | 19 | 0 |
| Juego 15 | 14 | 14 | 0 |
| Juego 16 | 16 | 16 | 0 |
| Juego 17 | 19 | 19 | 0 |
| Juego 18 | 20 | 20 | 0 |
| Juego 19 | 23 | 23 | 0 |
| Juego 20 | 28 | 28 | 0 |
| Juego 21 | 10 | 10 | 0 |
| Juego 22 | 27 | 27 | 0 |
| Juego 23 | 25 | 25 | 0 |
| Juego 24 | 27 | 27 | 0 |
| Juego 25 | 15 | 15 | 0 |
| Juego 26 | 15 | 15 | 0 |
| Juego 27 | 10 | 10 | 0 |
| Juego 28 | 11 | 11 | 0 |
| Juego 29 | 15 | 15 | 0 |

| | | | |
|-----------------------|-----|-------|---|
| Juego 30 | 8 | 8 | 0 |
| Juego 31 | 13 | 13 | 0 |
| Juego 32 | 6 | 6 | 0 |
| Juego 33 | 19 | 19 | 0 |
| Juego 34 | 12 | 12 | 0 |
| Juego 35 | 14 | 14 | 0 |
| Juego 36 | 16 | 16 | 0 |
| Juego 37 | 14 | 14 | 0 |
| TOTAL | 577 | 577 | 0 |
| PORCENTAJE DE ACIERTO | | 100 % | |

No es necesario analizar más imágenes, ya que son similares que las anteriores, con la excepción de que las fichas están unidas de forma alineada. Se han obtenido unos resultados similares a los precedentes.

En total se han analizado 577 imágenes de fichas, obteniendo una media de procesamiento de cada ficha de casi 21 configuraciones diferentes cada una.

En total entre los tres grupos, se han analizado 148 fotografías con un total de 2477 imágenes de fichas, lo que arroja un valor de más de 88 imágenes de media de cada ficha.

5.3 CONCLUSIONES Y LÍNEAS FUTURAS

Se han analizado una gran cantidad de imágenes para depurar el algoritmo al máximo. Además de los resultados con las imágenes que se han expuesto aquí, se han procesado muchas otras. De esta forma se ha conseguido un algoritmo que bajo los condicionamientos de adquisición es capaz de obtener unos resultados óptimos.

El bloque de identificación de fichas, funciona de manera perfecta, incluso con defectos de forma en los elementos de las fichas producidos por la pérdida de información en la imagen, debida tanto a reflejos como al desenfoque de las fotografías. Se ha llegado a dar el caso en el análisis de imágenes de fichas juntas, que con ciertos ángulos de orientación, al realizar la primera selección de la ventana de la ficha, aparecían en ésta dos separaciones de fichas, con lo que la identificación de la ficha no se realizaba correctamente. Ha sido necesario realizar modificaciones para identificar correctamente la separación de la ficha en cuestión, eliminando de esa ventana la que no correspondía.

El bloque de recuperación de información, también funciona perfectamente después de conseguir ajustar correctamente el punto de referencia a partir del cual superponer la separación patrón. Este punto ha necesitado varios reajustes, ya que se ha llegado a dar la circunstancia incluso de que estando la separación de la ficha partida con una orientación cercana a 90° , una mitad de la separación tenía una orientación de $+89,xx^\circ$ y la otra $-89,xx^\circ$, con lo que al seleccionar el ángulo de la otra mitad de la ficha para realizar la superposición, se identificaba incorrectamente. Pero todos estos errores que sólo son identificables tras el procesado de gran cantidad de imágenes han sido depurados hasta obtener un algoritmo robusto.

El bloque de identificación de localización de objetos y fichas, funciona bien pero depende en gran medida del bloque 1 de pre-procesado. Este bloque

localiza cada una de las fichas mediante la relación de excentricidad del objeto de separación de las fichas, luego bajo las condiciones de adquisición en las cuales no aparezcan objetos ajenos a las fichas de dominó, se produce una localización de fichas correcta.

El bloque de adquisición es quizá el más delicado, ya que las condiciones de adquisición y el posterior filtrado y umbralizado de la imagen hasta conseguir la imagen en blanco y negro son determinantes para el resto del algoritmo. Después de implementar diferentes métodos de filtrado, se ha optado por el expuesto en el desarrollo del programa por ser el que mejores resultados obtenía en los bloques posteriores, pero ha sido necesario establecer una serie de limitaciones a la hora de realizar la adquisición de las imágenes tales como las ya explicadas en el punto 5.1. Una vez cumplidas las premisas de adquisición establecidas, el funcionamiento del sistema es perfecto.

Como continuación a este proyecto, se puede probar a ejecutar el algoritmo con fondos más complicados y con más ruido que el fondo de tapete verde y a continuación mejorar el primer bloque, implementando mejores filtros y umbralización.

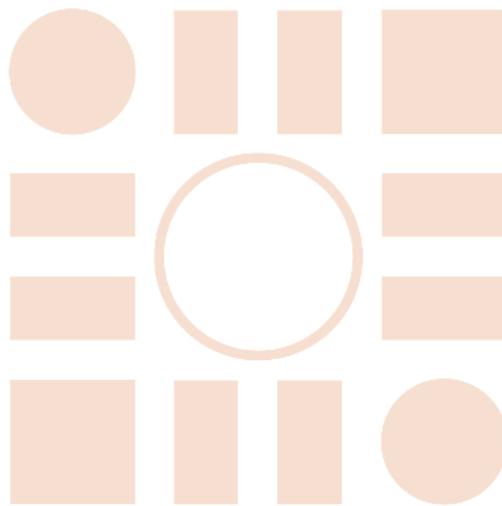
También se puede implementar un sistema que sea capaz de recuperar la separación intermedia de la ficha a partir del tamaño de los puntos, ya que en el caso de que sólo apareciese una ficha en la imagen y esta tuviese la separación partida, no se identificaría la ficha correctamente.

Además se puede probar el funcionamiento del algoritmo con distintos juegos de fichas de dominó y otros métodos de delimitación del área ocupada por la ficha.

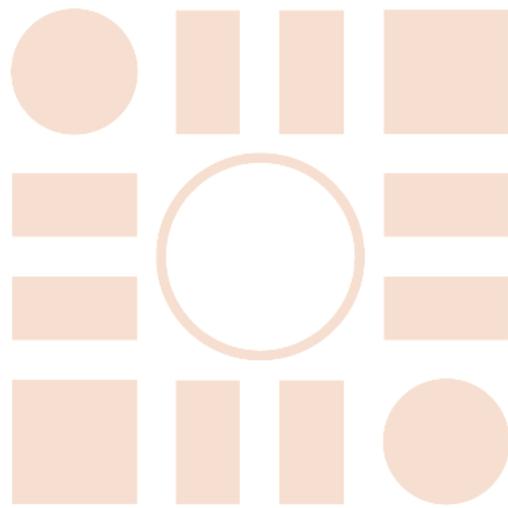
Otra mejora que se puede realizar del proyecto podría consistir en intentar identificar fichas que no se encuentren al completo, en función de la distribución de los puntos sobre la superficie de la ficha.

El presente proyecto trabaja únicamente con fotografías. Se puede realizar una modificación para que el sistema sea capaz de trabajar también con imágenes de video.

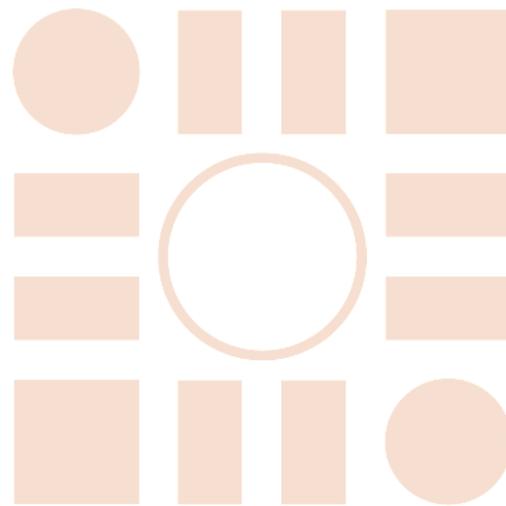
Por último se puede implementar un algoritmo con las reglas del juego de dominó, de manera que el sistema completo sea capaz de seguir una partida, proporcionándole instantáneas de la misma.



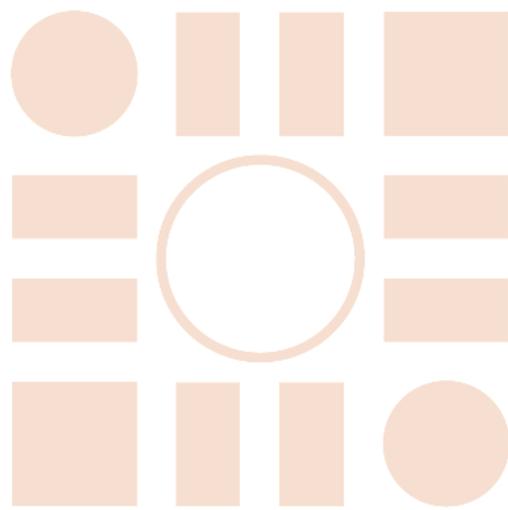
ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA SUPERIOR



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA
SUPERIOR

6.1 *PLIEGO DE CONDICIONES*

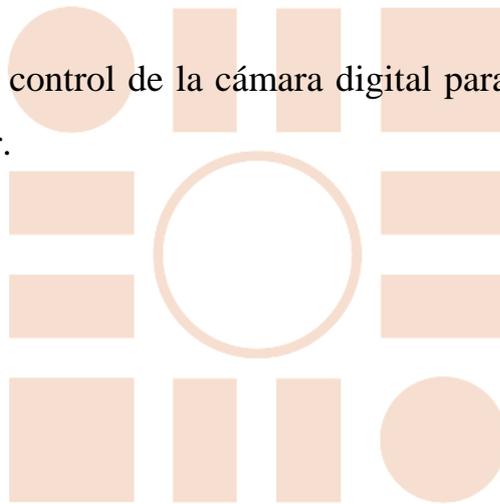
Para la realización de este proyecto es necesario contar tanto con una serie de elementos de hardware para realizar la parte experimental, como de software necesario para la parte relacionada con la implementación del programa realizado para el procesamiento de las imágenes adquiridas en el experimento.

Requisitos Hardware: En este apartado, se definen los elementos materiales necesarios para la reproducción del experimento:

- Juego completo de fichas de dominó: Se ha utilizado un juego de dominó de 28 fichas con el fondo en color marfil y los puntos y separaciones en color negro.
- Tapete: se trata de un tapete color verde típico de cualquier juego de mesa del estilo cartas.
- Cámara de fotos digital: en este caso se ha utilizado una cámara de fotos con sensor CMOS de 8 Megapixels, con una resolución de 3264x2448 y relación de aspecto 4:3.
- Ordenador con al menos los siguientes requerimientos:
 - 1 Gb de espacio libre en el disco duro
 - 1 Gb de memoria RAM
 - Puerto USB para conexión con cámara

Requisitos Software: En este apartado, se definen los programas software necesarios para la implementación y ejecución del algoritmo programado:

- Programa Matlab versión v7.11.0 (R2010b), con la Image Processing Toolbox.
- Sistema Operativo Windows 7.
- Software de control de la cámara digital para transferir las imágenes al ordenador.



ESCUELA POLITECNICA
SUPERIOR

6.2 PRESUPUESTO

Se van a desglosar los costes materiales asociados a la realización de este proyecto:

MATERIALES:

| ARTÍCULO | CANTIDAD | COSTE TOTAL |
|--------------------------|----------|-------------|
| Ordenador personal | 1 | 500 € |
| Impresora | 1 | 250 € |
| Windows | 1 | 160 € |
| Matlab 7 | 1 | 118 € |
| MS Office 2007 | 1 | 180 € |
| Cámara Digital | 1 | 100 € |
| Juego de dominó y tapete | 1 | 40 € |

TOTAL: 1348 €

MANO DE OBRA:

| ARTÍCULO | TIEMPO | EUROS / HORA | TOTAL |
|--------------|-----------|--------------|----------|
| Ingeniería | 360 Horas | 60 € | 21.600 € |
| Mecanografía | 80 Horas | 12 € | 960 € |

TOTAL:

22.560 €

GASTOS GENERALES:

| CONCEPTO | TOTAL |
|-----------------------|--------------|
| Encuadernación | 78 € |
| Consumibles impresora | 50 € |
| TOTAL: | 128 € |

Haciendo balance de los gastos generados en la realización del proyecto:

GASTOS DE EJECUCIÓN POR CONTRATA:

| | |
|-------------------------------------|-----------------|
| TOTAL EJECUCIÓN POR CONTRATA | 24.036 € |
|-------------------------------------|-----------------|

Según esta cantidad, y de acuerdo con las tarifas vigentes en España, el porcentaje a aplicar para el cálculo de honorarios es del 7%:

HONORARIOS:

| | |
|-------------------------|-------------------|
| TOTAL HONORARIOS | 1.682,52 € |
|-------------------------|-------------------|

ESCOLETA TECNICA
SUPERIOR

Por tanto, la cantidad resultante a pagar será la que resulte de sumar los gastos de ejecución por contrata, los honorarios y el IVA:

IMPORTE TOTAL:

| | |
|------------------------------|--------------------|
| Total Ejecución por contrata | 24.036 € |
| Total Honorarios | 1.682,52 € |
| TOTAL SIN IVA: | 25.718,52 € |

| | |
|--------------------|------------------|
| 21% DE IVA: | 5400,89 € |
|--------------------|------------------|

| | |
|------------------------|--------------------|
| TOTAL PROYECTO: | 31.119,41 € |
|------------------------|--------------------|

ESCUELA POLITECNICA
SUPERIOR

6.3 MANUAL DE USUARIO

Para poder ejecutar el algoritmo programado, se ha realizado una GUI (Graphical User Interface) muy simple, que se muestra en la figura 6.1.



Figura 6.1: Interfaz gráfica

La interfaz consta de una ventana donde se muestra la imagen a analizar y un panel con tres botones:

- **Carga Imagen:** Pulsando este botón se abre una ventana de selección, en la que se puede navegar por las carpetas del ordenador, para seleccionar la imagen que se quiere analizar. En la figura 6.2 se muestra esta ventana.

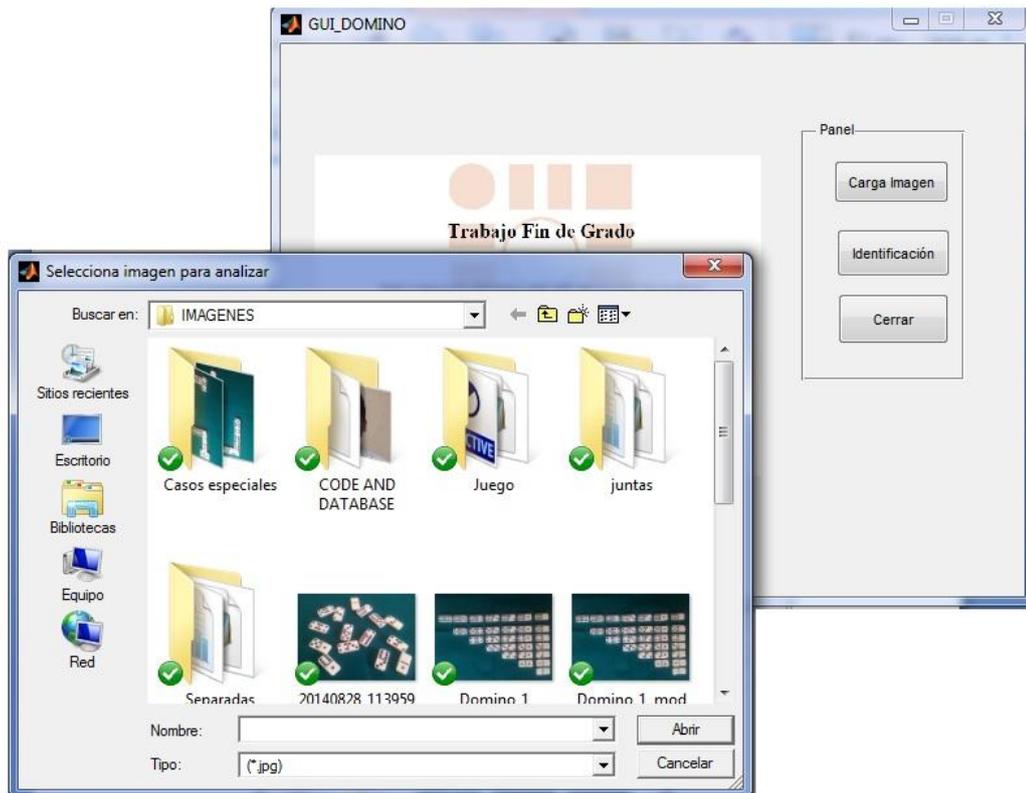


Figura 6.2: Ventana de botón Carga Imagen

- **Identificación:** Una vez cargada la imagen (Figura 6.3), se pulsará el botón *Identificación* para proceder a ejecutar el algoritmo de identificación.



Figura 6.3: GUI con imagen cargada lista para identificación

- Cerrar: Tras realizar la identificación, si se desean cerrar todas las figuras de Matlab, se deberá pulsar este botón.

Cuando se pulsa el botón de Identificación, se irán abriendo una serie de figuras que van mostrando la evolución del proceso de identificación y finalmente las imágenes de las fichas individuales identificadas.

En la figura 6.4 se puede ver el resultado de la identificación del ejemplo anterior.

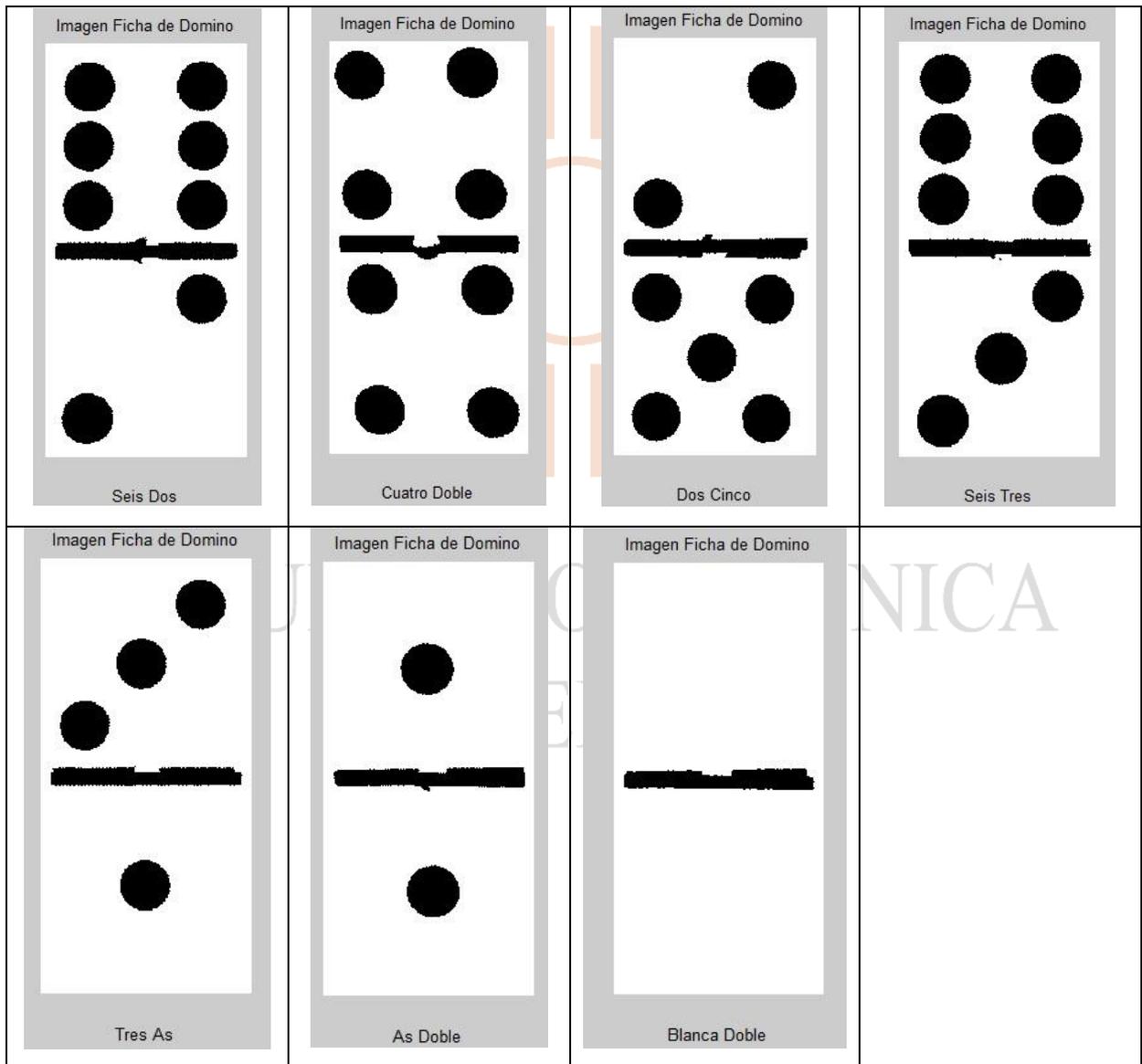
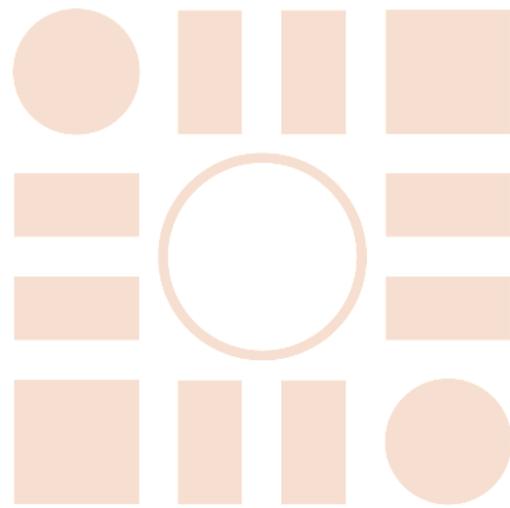
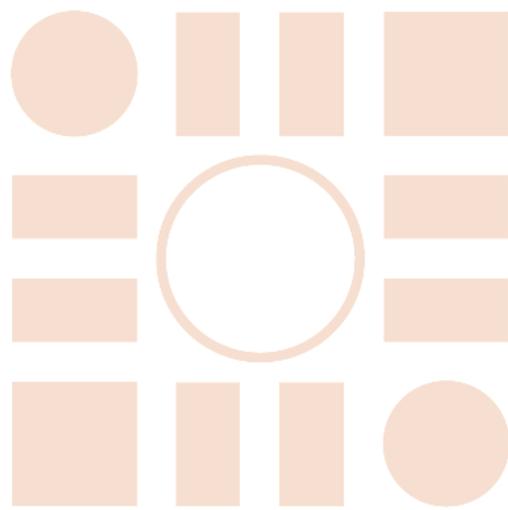


Figura 6.4: Resultado de ejemplo de identificación



ESCUELA POLITECNICA
SUPERIOR



ESCUELA POLITECNICA SUPERIOR

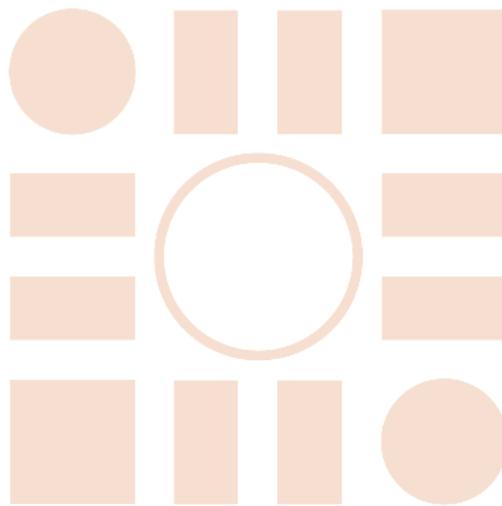
7.1 REFERENCIAS

- [1] *“Real-Time Object Detection for Smart vehicles” - D.M. Gavrila, V Philomin.*
- [2] *“Crowd Analysis Using Computer Vision Techniques” – Julio Cezar Silveira Jacques Junior, Soraia Rauup Musse, Cláudio Rosito Jung.*
- [3] *“Individual identification of polar bears by whisker spot patterns” – Carlos J. R. Anderson*
- [4] *“Playing Domino: A Case Study for an Active Vision System” – Maik Bollmann, Rainer Hoischen, Michael Jesikiewicz, Christoph Justkowski, Bärbel Mertsching*
- [6] *US patente 3971065, Bryce E. Bayer, "Color imaging array", emitted on 1976-07-20.*
- [6] *"Reconocimiento de naipes españoles mediante técnicas de visión artificial en el entorno de trabajo Matlab". Daniel Carrasco Muro. EPS (uah). Julio 2014.*

7.2 BIBLIOGRAFÍA

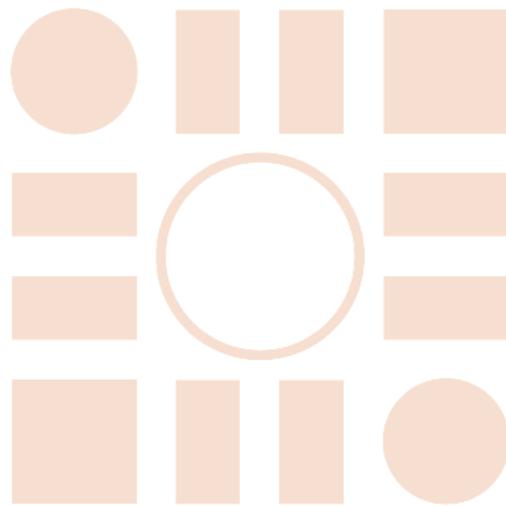
- Apuntes, transparencias y prácticas de la asignatura “Sistemas de Visión Artificial” del Departamento de Electrónica de la Universidad de Alcalá.
- Documentación de Ayuda del propio programa de Matlab, fundamentalmente la relacionada con la “Image Processing Toolbox”
- DIGITAL IMAGE PROCESSING, 2nd Edition. By Gonzalez, R.C.; Woods, R.E. McGraw-Hill 2002.
- TRATAMIENTO DIGITAL DE IMÁGENES. RAFAEL C. GONZÁLEZ. RICHARD E. WOODS. ADDISON WESLEY. 1996
- VISIÓN POR COMPUTADOR. IMÁGENES DIGITALES Y APLICACIONES. Pajares, G., de la Cruz, J. M. Ra-Ma. 2001.
- VISIÓN POR COMPUTADOR. González, J. Paraninfo. 2000.
- THE IMAGE PROCESSING HANDBOOK, 3RD ED. By Russ, J.C. CRC-IEEE Press 1999.
- IMAGE PROCESSING, ANALYSIS AND MACHINE VISION, 2ND ED. By Sonka, M.; Vaclav, H.; Boyle, R. PWS Publishing, 1999.

- RECONOCIMIENTO DE FORMAS Y VISIÓN ARTIFICIAL. D. Maravall Gómez-Allende. Ed. RAMA. 1993 o DIGITAL IMAGE PROCESSING. W. K. Pratt, J. Wiley&Sond. Inc. 1991



ESCUELA POLITECNICA
SUPERIOR

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR



Universidad
de Alcalá