

UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

**CLASIFICACIÓN DE GARRAS DE POLLO MEDIANTE IMÁGENES DIGITALES**

**MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO**

**DIEGO SÁENZ VALENZUELA**

PROFESOR GUÍA:  
JAVIER RUIZ DEL SOLAR

MIEMBROS DE LA COMISIÓN:  
JORGE LOPEZ HIDALGO  
FERNANDO BERNUY BAHAMONDEZ

SANTIAGO DE CHILE  
NOVIEMBRE 2016

**RESUMEN DE LA MEMORIA PARA OPTAR  
AL TÍTULO DE:** Ingeniero Civil Eléctrico.  
**POR:** Diego Sáenz Valenzuela  
**FECHA:** 16/11/2016  
**PROFESOR GUÍA:** Javier Ruiz del Solar

## **CLASIFICACIÓN DE GARRAS DE POLLO MEDIANTE IMÁGENES DIGITALES**

En las plantas faenadoras de pollos se busca sacar el máximo provecho a cada animal, por lo que no solo se comercializan sus partes altas en carne como lo son la pechuga y el trutro, sino que también se venden partes como alas, interiores y patas. El presente trabajo trata de la categorización de patas, ya que no todas tienen el mismo precio, sino que se dividen en categorías dependiendo de su peso y cantidad de defectos, siendo las más grandes y sanas las mejor valoradas. Actualmente, la categorización de patas de pollo se hace de forma manual, lo cual puede estar sujeto a la subjetividad de cada operador, toma bastante tiempo y, a su vez, se refleja en el balance de las empresas como un alto costo fijo por mano de obra. Dado lo anterior, se vuelve muy interesante la automatización de procesos que permitan aumentar su eficiencia en la separación por categorías.

En la industria del pollo, las patas del animal son conocidas como garras y sus categorías son llamadas calibres, los que dependen del peso y cantidad de defectos de cada garra. En este trabajo de título se proponen y comparan distintas metodologías para definir el calibre de una garra a partir de dos imágenes, mostrando su parte superior e inferior. Debido a la complejidad del problema, solo se consideran garras sin fracturas o raspaduras, limitando la clasificación a garras con defectos reconocibles por su color como lo son las cutículas, callos y hematomas.

Tras probar distintos métodos para clasificar las garras según sus calibres, se llega a uno que consiste en una cuantificación de defectos utilizada como entrada para un clasificador en base a rangos, el cual, con un 97.4% de exactitud, demostró ser el mejor método para definir el calibre. La cuantificación mencionada consiste en utilizar una cascada de clasificadores binarios, con las intensidades RGB de los píxeles como entrada, identificando aquellos que muestran cutícula, callo, hematoma o garra sana, para finalmente obtener la cuantificación con la suma de píxeles de cada defecto.

Como conclusión se puede decir que, a pesar de las limitaciones impuestas para el desarrollo de la metodología propuesta, se han logrado los objetivos planteados en forma satisfactoria, logrando obtener un método de clasificación de garras, escalable a la industria faenadora de pollos.

## **Agradecimientos**

En primer lugar, a mi familia, sin su apoyo y cariño no estaría donde estoy ni sería la persona que soy.

A mis amigos, ya sean Ratas, Corillos u OPAMPS, todos son grandes amigos que me han ayudado más de lo que creen. En especial a Juan Neme, Ignacia Riquelme y José Rosales, quienes me empujaron cuando fue difícil, dejando su huella en este trabajo.

A los trabajadores de la planta faenadora de pollos de Agrosuper en San Vicente, quienes me ayudaron con una parte fundamental de este trabajo.

A mis profesores, del colegio y la universidad, todos han aportado a mi formación de una u otra manera.

Sin el aporte de cada uno de ustedes, hubiese sido imposible realizar este trabajo, ¡Muchísimas gracias!

# Índice de contenidos

<b>1</b>	<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>2</b>	<b>REVISIÓN BIBLIOGRÁFICA</b> .....	<b>3</b>
2.1	GARRAS DE POLLO.....	3
2.1.1	<i>Defectos</i> .....	3
2.1.2	<i>Calibres</i> .....	5
2.2	IMÁGENES DIGITALES .....	6
2.2.1	<i>Captura de imágenes</i> .....	6
2.2.2	<i>Cámaras fotográficas</i> .....	7
2.2.3	<i>Relación entre píxeles</i> .....	9
2.2.4	<i>Operaciones con imágenes en el dominio espacial</i> .....	10
2.2.5	<i>Imágenes a color</i> .....	13
2.2.6	<i>Resolución espacial y de intensidad</i> .....	15
2.2.7	<i>Procesamiento de imágenes</i> .....	17
2.3	EXTRACCIÓN DE CARACTERÍSTICAS .....	19
2.3.1	<i>Histogramas de escala de grises</i> .....	19
2.3.2	<i>Histogramas de color</i> .....	20
2.3.3	<i>Entropía de Shannon</i> .....	20
2.4	INTELIGENCIA COMPUTACIONAL .....	21
2.4.1	<i>Redes neuronales artificiales</i> .....	21
2.4.2	<i>Arboles de decisión</i> .....	24
2.4.3	<i>K-vecinos más cercanos</i> .....	25
2.5	HERRAMIENTAS DE REDUCCIÓN DE DIMENSIONALIDAD .....	26
2.5.1	<i>Cuantización de histogramas</i> .....	26
2.5.2	<i>Análisis de componentes principales</i> .....	26
2.6	HERRAMIENTAS DE INTERPRETACIÓN DE RESULTADOS .....	27
2.6.1	<i>Matriz de confusión</i> .....	27
2.6.2	<i>Curva ROC</i> .....	29
2.7	ESTADO DEL ARTE .....	30
2.8	MOTIVACIÓN Y APORTE DEL TRABAJO DE MEMORIA DE TÍTULO.....	30
<b>3</b>	<b>CREACIÓN DE BASE DE DATOS</b> .....	<b>32</b>
3.1	ESTRUCTURA PARA CAPTURA DE IMÁGENES.....	32
3.2	CÁMARA FOTOGRÁFICA Y SU CONFIGURACIÓN .....	33
3.3	TOMA DE MUESTRAS Y ETIQUETAS .....	34
<b>4</b>	<b>MÁSCARA DE SEGMENTACIÓN</b> .....	<b>35</b>
4.1	SEGMENTACIÓN CON MODELO DE FONDO .....	35
4.2	SEGMENTACIÓN CON UMBRALES RGB .....	37
4.3	SEGMENTACIÓN CON ESPACIO DE COLOR YCbCr Y UMBRAL .....	38
4.4	SEGMENTACIÓN CON ESPACIO DE COLOR HSV Y UMBRAL .....	39
<b>5</b>	<b>DESARROLLO DE MÉTODOS DE CLASIFICACIÓN</b> .....	<b>41</b>

5.1	CLASIFICACIÓN POR PESO .....	41
5.2	CLASIFICACIÓN POR DEFECTOS. HISTOGRAMAS DE COLOR.....	42
5.3	CLASIFICACIÓN POR DEFECTOS. CUANTIFICACIÓN Y CLASIFICACIÓN.....	43
5.3.1	<i>Cuantificación de defectos .....</i>	<i>43</i>
5.3.2	<i>Determinación de calibre con clasificador multiclase .....</i>	<i>55</i>
5.3.3	<i>Determinación de calibres con umbrales.....</i>	<i>56</i>
<b>6</b>	<b>ANÁLISIS DE RESULTADOS.....</b>	<b>61</b>
6.1	ANÁLISIS MÁSCARA DE SEGMENTACIÓN .....	61
6.1.1	<i>Método con modelo de fondo .....</i>	<i>61</i>
6.1.2	<i>Método con componentes RGB.....</i>	<i>62</i>
6.1.3	<i>Método con espacio de color YCbCr.....</i>	<i>62</i>
6.1.4	<i>Método con espacio de color HSV .....</i>	<i>63</i>
6.2	CLASIFICACIÓN POR PESO .....	64
6.3	CLASIFICACIÓN POR DEFECTOS. HISTOGRAMAS DE COLOR.....	64
6.4	CLASIFICACIÓN POR DEFECTOS. CUANTIFICACIÓN Y CLASIFICACIÓN.....	67
6.4.1	<i>Cuantificación .....</i>	<i>67</i>
6.4.2	<i>Clasificación.....</i>	<i>74</i>
<b>7</b>	<b>CONCLUSIONES .....</b>	<b>77</b>
<b>8</b>	<b>BIBLIOGRAFÍA .....</b>	<b>79</b>
<b>ANEXOS.....</b>		<b>81</b>
ANEXO A.....		81
ANEXO B.....		83
ANEXO C.....		87
ANEXO D.....		88
ANEXO E .....		92

# 1 Introducción

En las plantas faenadoras de pollos se busca sacar el máximo provecho a cada animal, por lo que no solo se comercializan sus partes altas en carne como lo son la pechuga y el trutro, sino que también se venden partes como alas, interiores y patas. Las partes de poca carne son poco tomadas en cuenta en el mercado chileno, no así en países de Asia del Este, donde partes como las alas y patas son valoradas por sobre la pechuga y el trutro.

La exportación de patas de pollo es un negocio en expansión y una oportunidad para países de Latinoamérica, debido a que las patas de origen estadounidense están gravadas con un impuesto del 105% para entrar a Asia, el cual rige desde el año 2010 [1].

Pero no todas las patas tienen el mismo precio, sino que se dividen en categorías dependiendo de su peso y cantidad de defectos, siendo las más grandes y sanas las mejores valoradas. Actualmente, la categorización de patas de pollo se hace de forma manual, lo cual puede estar sujeto a la subjetividad de cada operador, toma bastante tiempo y, a su vez, se refleja en el balance de las empresas como un alto costo fijo por mano de obra. Dado lo anterior, se vuelve muy llamativa la automatización de procesos que permitan aumentar su eficiencia en la separación por categorías.

En la industria del pollo, las patas del animal son conocidas como garras y sus categorías son llamadas calibres, por lo que durante el presente trabajo se usarán dichos términos.

El objetivo de este proyecto es desarrollar un método, escalable a la industria, para asignar calibres a garras de pollo mediante dos imágenes en las que se capture ambos lados del objeto. Debido a la complejidad del problema, se limitará la clasificación a garras que no tengan fracturas o raspaduras, acotando el trabajo al análisis de defectos percibidos en color. También, se asumirá una perfecta coordinación entre la línea de proceso y la captura de imágenes, por lo que las muestras a estudiar solo ubican a la garra al centro de la imagen y se supone como trabajo previo el reconocimiento de casos en que la imagen no incluye una garra o la toma solo muestra una parte del objeto.

En el capítulo 2, se presenta una revisión de conceptos y herramientas computacionales que serán utilizadas a lo largo de éste trabajo, esto tiene el propósito de ubicar al lector en el contexto en que lleva a cabo el proyecto.

La metodología parte en el capítulo 3, por la creación de base de datos, en la que se define la configuración de la cámara fotográfica a utilizar, el

fondo sobre el que se apoyan las garras y la iluminación que se les aplica. Luego, en el capítulo 4, se entra a una etapa de pre procesamiento, donde se comparan distintos métodos para separar los pixeles de garra y los de fondo.

Una vez que se cuenta con una forma de hacer la separación del fondo, se procesa la muestra para su clasificación y, con éste propósito, se proponen dos metodologías presentadas en el capítulo 5. La primera consiste en extraer información en forma de histogramas y estadísticas de los mismos, para luego, con un algoritmo basado en redes neuronales, clasificar las muestras. La segunda se basa en una cuantificación de distintos defectos por separado, para luego clasificar las muestras en base a la cuantificación realizada.

En el capítulo 6, se utilizan muestras de prueba para comparar las distintas metodologías propuestas en los capítulos 4 y 5, en base al desempeño de las mismas. El trabajo concluye en el capítulo 7, donde se entregan y comentan los resultados finales.

## 2 Revisión Bibliográfica

El presente capítulo tiene por objetivo ubicar al lector en el entorno en el cual se desarrolla este trabajo de título, entregando los antecedentes previos y necesarios para su contextualización.

En primer lugar, en la sección 2.1, se describen las garras de pollo, sus defectos y calibres. En 2.2 se define lo que es una imagen digital, como se capturan y propiedades de las mismas. En 2.3 se explican distintos métodos para extraer información de las imágenes y usarla como entrada en sistemas de inteligencia computacional que permitirán la clasificación (2.4), para terminar con 2.6, un aparatado de herramientas de interpretación de resultados.

### 2.1 Garras de pollo

Las garras de pollo son un producto que ha perdido espacio en el mercado local, pero se ha expandido en países de Asia del Este, tales como China, Vietnam, Corea y Tailandia. Es por esto que, se hace importante clasificar las garras en distintos calibres y, así, obtener una mayor ganancia que justifique la exportación del producto. Para su clasificación se deben tener en cuenta 2 características de la garra que son: su peso y sus defectos. A continuación, se describen los posibles defectos de las garras y, luego, los calibres según defectos y peso [2].

#### 2.1.1 Defectos

##### 2.1.1.1 Hematomas

Por distintos problemas en el desarrollo del pollo, sus garras pueden presentar hematomas, los cuales toman colores rojizos. La gravedad de un hematoma a nivel de defecto depende del tamaño de este. En la Ilustración 1 se pueden observar 2 garras con hematomas.



Ilustración 1: Garras con hematomas.

#### 2.1.1.2 Callos

Por problemas en la pisada del pollo, se pueden desarrollar callosidades en la parte inferior de la garra. El tamaño del callo define la gravedad del defecto.



Ilustración 2: Garra con callo.

#### 2.1.1.3 Restos de cutícula

Las garras que son procesadas han pasado por una etapa de pelado donde se le saca la cutícula (parte externa del cuero). Esta parte es amarilla y es la que estuvo en contacto con el medio durante la vida del pollo y, por lo tanto, en ella se encuentran las posibles infecciones. Las garras no pueden tener restos de cutícula para su exportación, por lo que las que sí lo tienen deben ser detectadas y enviadas de nuevo al proceso de pelado. En la Ilustración 3 se puede ver una garra con restos de cutícula.



Ilustración 3: Garra con resto de cutícula.

#### 2.1.1.4 Uñas rotas, raspaduras y fracturas

Tanto por heridas que el pollo trae del criadero como por las provocadas por la maquinaria del proceso, estos defectos deben ser detectados y cuantificados para que la garra entre en un calibre específico. En la Ilustración 4 se pueden observar 2 ejemplos de garras con heridas.



Ilustración 4: Garras con raspaduras y uñas rotas.

## 2.1.2 Calibres

A continuación, se describen los calibres de las garras que tienen precio comercial, se destaca que las garras que tienen cutícula deben ser detectadas por el mismo sistema, con el fin de ser reenviadas al proceso de pelado. Estas garras serán etiquetadas como reproceso.

### 2.1.2.1 Garra A

En este calibre, las garras pueden tener hematomas leves y no más de uno. Se acepta una uña rota, raspaduras leves y con fractura no expuesta. No se aceptan callos.

Dependiendo del peso las garras A tienen distintos calibres, las que pesan menos de 26 gramos son de calibre Medium A y las que pesan entre 26 y 33 gramos son de calibre Large A.

### 2.1.2.2 Garra B

En este calibre, las garras pueden tener un callo de hasta 1.5 cm. Se acepta más de 1 hematoma leve, muñón y fractura expuesta sólo en caso de presentar solo 1 defecto.

Dependiendo del peso las garras B tienen distintos calibres, las que pesan menos de 26 gramos son de calibre Medium B y las que pesan sobre 26 gramos son de calibre Large B.

### 2.1.2.3 Garra C

En este calibre, las garras pueden tener un callo de hasta 3 cm. Se aceptan hematomas de hasta 2 cm, dedos quebrados, raspaduras múltiples o roturas muy profundas y extensas. Se aceptan restos pequeños de cutícula.

Por su característica de defectos muy marcados, las garras C no hacen separación por peso, simplemente son todas clasificadas como Garra C.

#### 2.1.2.4 Jumbo

Es un caso especial de una garra A. Todas las garras A que pesen más de 34 gramos son del mejor calibre llamado Jumbo.

#### 2.1.2.5 Garra CHOICE

Son todas las garras que tienen demasiados defectos para entrar en uno de los calibres antes descritos.

## 2.2 Imágenes digitales

Se puede definir una imagen como una función bidimensional  $f(x,y)$ , donde  $x$  e  $y$  son coordenadas espaciales y la amplitud de  $f$  en cualquier punto  $(x,y)$  es la intensidad. Cuando los valores de  $x$ ,  $y$  y  $f$  son finitos y discretos, se habla de imagen digital. En las imágenes digitales, cada punto  $(x,y)$  es llamado pixel y la amplitud de  $f$  toma valores de escala de gris o colores según corresponda.

### 2.2.1 Captura de imágenes

Para transformar ondas de luz en información digital, es necesario un elemento que cumpla 2 funciones, representar la longitud de onda de la luz como un voltaje y transformar dicho voltaje en una cuantificación digital. Estos elementos se agrupan en arreglos bidimensionales dentro de un circuito integrado llamado dispositivo de carga acoplada (en inglés *charge-coupled device*, conocido también como CCD como se observa en la Ilustración 5) [3].

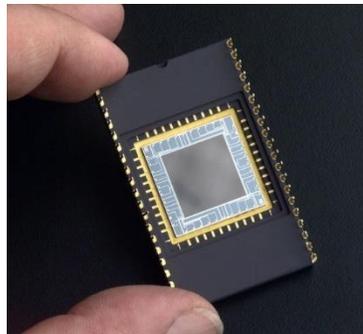


Ilustración 5: Dispositivo de carga acoplada.

El ojo humano solo puede ver luz cuya longitud de onda cae dentro del espectro visible. Los sensores CCD pueden estar configurados para captar longitudes de onda que escapan del espectro antes mencionado, por lo que pueden capturar imágenes infrarrojas, ultravioletas, rayos X, ondas radio, entre otras [4]. En este proyecto solo se utilizarán sensores que captan luz en longitudes de onda dentro del rango visible.

## 2.2.2 Cámaras fotográficas

A continuación, se dan nociones básicas de fotografía, los componentes de una cámara y los parámetros importantes de manejar al momento de hacer una buena fotografía. Este no es el foco del trabajo realizado, pero si se tuvo en cuenta para crear la base de datos.

Una cámara fotográfica hereda su nombre de la antigua cámara oscura, instrumento utilizado para proyectar imágenes reales sobre superficies planas para luego ser dibujadas o pintadas. Con la evolución de la óptica, se comenzó a agregar lentes en la entrada de la cámara para acercar o alejar la proyección del objeto [5], lo que hoy se llama objetivo y se muestra en la Ilustración 6.

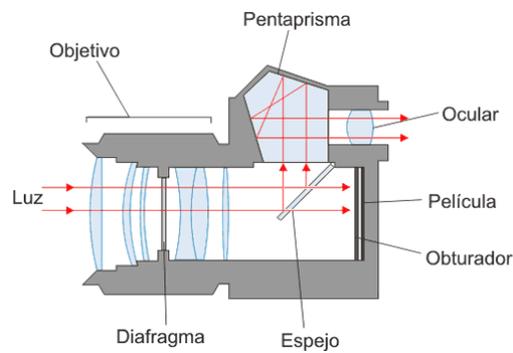


Ilustración 6: Esquema de una cámara fotográfica análoga.

La fotografía llega a revolucionar la forma de usar una cámara oscura, ya que en vez de dibujar o pintar la proyección, esta podía ser grabada en una película o film que se ubica al fondo de la cámara. Con el desarrollo y progreso de la fotografía aparecen los parámetros que hoy en día son importantes para utilizar correctamente una cámara fotográfica, estos parámetros son la sensibilidad ISO, la apertura del diafragma y la velocidad de obturación o tiempo de exposición [6], explicados más adelante.

En la fotografía digital se reemplaza la película o film por un sensor CCD, componentes electrónicos adicionales se encargan de procesar y almacenar las imágenes.

### a. Sensibilidad ISO

La Sensibilidad ISO es una medida inversa de cuanta luz es necesaria para que la imagen quede grabada en la película, de tal manera que, para un ambiente de mucha luz, se debe usar un film de baja sensibilidad. Cuando no se utiliza la sensibilidad adecuada, se puede compensar la luz que entra en la cámara con la apertura del diafragma y tiempo de exposición, pero siempre se debe tener en cuenta que mayor ISO significa mayor ruido en la imagen [7], como se puede ver en el ejemplo de la Ilustración 7.

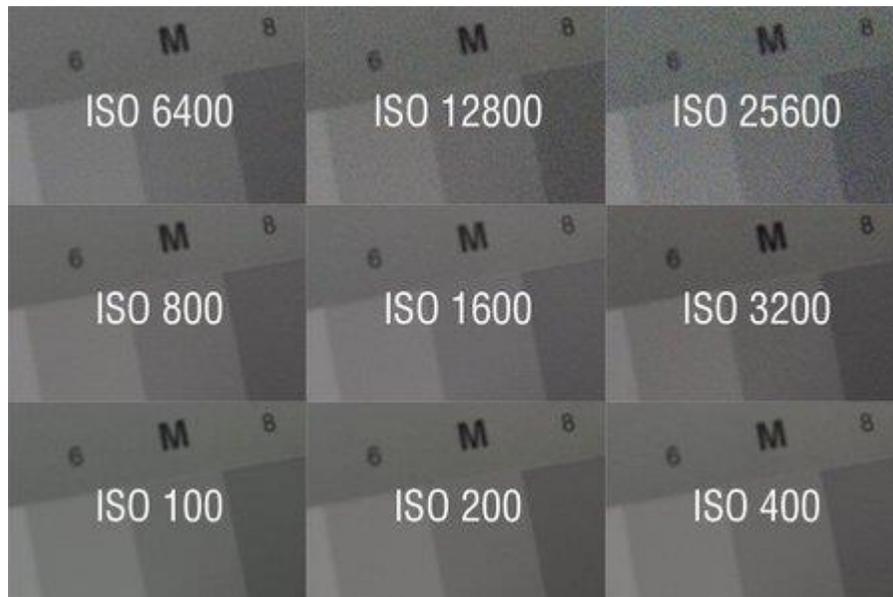


Ilustración 7: Comparación de sensibilidad ISO.

Originalmente, en las cámaras fotográficas análogas, la sensibilidad ISO era una propiedad de la película que se ubicaba al fondo de la cámara, hoy en día, las cámaras digitales simulan esta propiedad por medio de un ajuste en el sensor CCD [8].

b. Apertura del diafragma

El diafragma es una imitación del iris humano, tiene la capacidad de abrirse y cerrarse dejando pasar más o menos luz a la cámara (ver Ilustración 8).



Ilustración 8: Diafragma con distintas aperturas.

c. Velocidad de obturación

El tiempo de exposición (inverso de la velocidad) es el tiempo en que el obturador se abre para que la luz pase al sensor. Si la luz que el diafragma deja pasar es poca, un largo tiempo de exposición puede hacer que la toma no salga oscura, pero si algo se mueve en ese transcurso, la toma saldrá movida.

### 2.2.3 Relación entre píxeles

Para poder definir de manera más acertada algunos términos a utilizar durante el presente trabajo, es necesario dar nombre a ciertas relaciones entre píxeles. Anteriormente se definió un píxel como un punto  $(x, y)$  en una imagen digital, para hacer referencia a los mismos, en esta sección se usarán letras minúsculas tales como  $p$  o  $q$  [4].

#### 2.2.3.1 Píxeles vecinos

Un píxel  $p$  de coordenadas  $(x, y)$  tiene 4 vecinos de forma horizontal y vertical cuyas coordenadas están dadas por

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)$$

Este set de 4 píxeles vecinos se denota  $N_4(p)$ . También se consideran vecinos a los que se encuentran de manera diagonal a  $p$  cuyas coordenadas están dadas por

$$(x + 1, y + 1), (x + 1, y - 1), (x - 1, y + 1), (x - 1, y - 1)$$

La notación de estos píxeles es  $N_D(p)$  y junto con  $N_4(p)$  forman una vecindad de 8 píxeles  $N_8(p)$ . Si  $p$  está al borde de la imagen, algunos de sus vecinos quedarán fuera de la imagen.

#### 2.2.3.2 Distancia entre píxeles

La distancia entre píxeles es importante para definir el área que ocupa una vecindad, pues su magnitud depende de la medida de distancia que se utilice. La distancia  $D_4$  considera que la vecindad  $N_4$  se encuentra a 1 unidad de distancia y ocupa la misma regla para medir distancias más lejanas. Considerando los píxeles  $p$  y  $q$  de coordenadas  $(x, y)$  y  $(s, t)$  respectivamente, esta regla puede ser llevada a la siguiente ecuación:

$$D_4(p, q) = |x - s| + |y - t|$$

En la Ilustración 9 se puede ver al píxel  $p$  al centro representado con un 0 que representa la distancia  $D_4$  hasta el mismo, el resto de los píxeles también son representados con su distancia  $D_4$  hasta  $p$ .

$$\begin{array}{ccccc} & & 2 & & \\ & & 2 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 \\ & & 2 & 1 & 2 \\ & & & & 2 \end{array}$$

Ilustración 9: Ejemplo de distancia  $D_4$  desde  $p$  ubicado al centro.

Otra medida de distancia es llamada  $D_8$  que siguiendo la notación anterior se define por la ecuación:

$$D_8(p, q) = \max(|x - s|, |y - t|)$$

En la Ilustración 10, se puede ver al pixel  $p$  al centro representado con un 0 que representa la distancia  $D_8$  hasta el mismo, el resto de los pixeles también son representados con su distancia  $D_8$  hasta  $p$ .

```

2 2 2 2 2
2 1 1 1 2
2 1 0 1 2
2 1 1 1 2
2 2 2 2 2

```

Ilustración 10: Ejemplo de distancia  $D_8$  desde  $p$  ubicado al centro

#### 2.2.4 Operaciones con imágenes en el dominio espacial

Con el fin de extraer o filtrar información de una o más imágenes, se pueden hacer operaciones sobre ellas. Sea  $f(x, y)$  una imagen digital, tras aplicar un operador  $T$  sobre ella, se puede obtener una imagen procesada  $g(x, y)$  que facilita procesar la información contenida en  $f(x, y)$ . Es importante mencionar que, un operador  $T$  puede aplicarse también sobre un set de imágenes generando una sola imagen  $g(x, y)$  que reúne la información contenida en el set.

Uno de los objetivos principales de utilizar este tipo de operadores en este trabajo es diferenciar el fondo de la imagen con el objeto de interés, técnicamente llamado primer plano, que para este caso será la garra del pollo.

En el presente trabajo se utilizarán operadores lógicos y aritméticos sobre imágenes y, a continuación, se describirán los más importantes.

##### 2.2.4.1 NOT

El operador negación, por su naturaleza, solo puede aplicarse en una imagen y no en un set de ellas pues no establece relación alguna entre 2 imágenes, es usado para obtener el negativo (ver Ilustración 11) de una imagen o para formar operadores más complejos al combinarse con operadores OR y AND [3] [4].

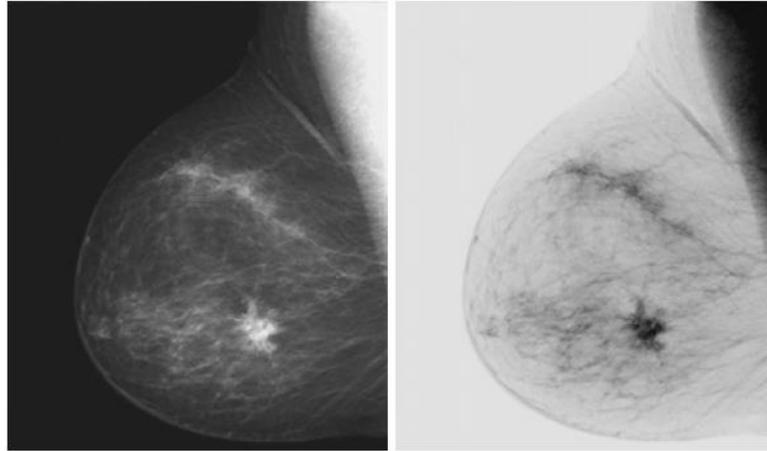


Ilustración 11: Imagen y su negativo tras aplicar el operador NOT.

#### 2.2.4.2 AND

AND es un operador utilizado para seleccionar cierta región de la imagen y eliminar (llevar a 0 en escala de gris) todo el resto de ella, este proceso es llamado enmascarar la imagen. En la Ilustración 12 se puede ver la imagen original a la izquierda, la máscara al centro y el resultado de aplicar el operador AND entre ellas a la derecha [3] [4].

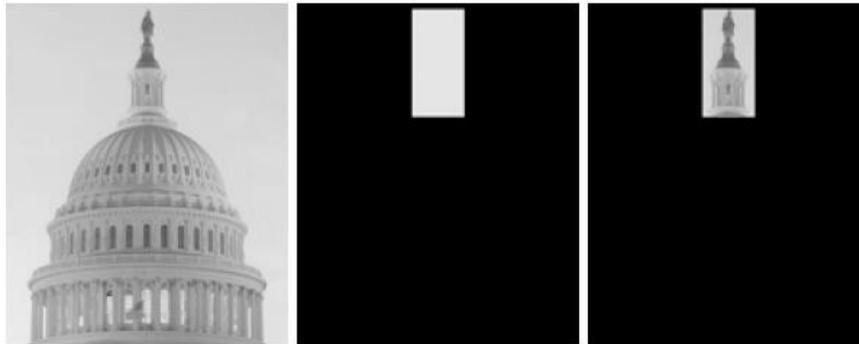


Ilustración 12: Aplicación del operador AND entre una imagen y una máscara.

#### 2.2.4.3 OR

OR también es utilizado para enmascarar imágenes, con la diferencia que la región de la imagen que no es de interés, es rellenada con el máximo en la escala de gris. En la Ilustración 13, se puede ver la imagen original a la izquierda, la máscara al centro y el resultado de aplicar el operador OR entre ellas a la derecha [3] [4].

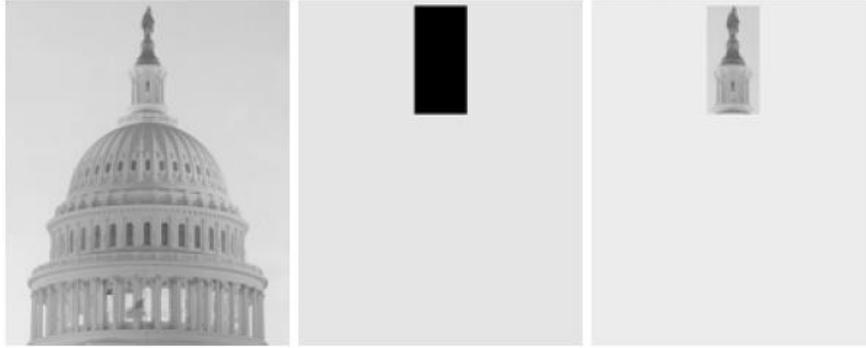


Ilustración 13: Aplicación de operador OR entre una imagen y una máscara.

Los operadores AND y OR son utilizados, generalmente, para seleccionar regiones de interés (ROI por su sigla en inglés *Region Of Interest*) en una imagen para procesamiento morfológico u otro tipo de procesamiento.

#### 2.2.4.4 Modelo de fondo

Es normal que una imagen de fondo tenga pequeñas alteraciones entre una captura y otra (ruido), por lo que se considerará una imagen  $g(x, y)$  que contiene información sobre el fondo real  $f(x, y)$  y sobre el ruido antes mencionado  $\eta(x, y)$

$$g(x, y) = f(x, y) + \eta(x, y)$$

Suponiendo que el ruido no tiene correlación y es de media 0, promediar un set de imágenes tendrá el efecto de neutralizar el ruido y entregar un modelo sobre el fondo real  $h(x, y)$ .

#### 2.2.4.5 Resta de imágenes

Al restar 2 imágenes, se obtiene la diferencia entre ellas, lo que permite resaltar cambios entre una y otra. Esto puede hacer muy efectiva la identificación de objetos de interés en un fondo fijo creando una máscara. En la Ilustración 14 se muestra un diagrama de como restando 2 imágenes y aplicando un umbral, se obtiene una máscara para el objeto [3] [4].

Visto en forma de ecuación, teniendo una imagen  $q(x, y)$  y un modelo de fondo  $h(x, y)$ , un umbral  $T$  sobre la resta da origen a una máscara  $M$  para identificar el primer plano

$$M(x, y) = T(q, h) = q(x, y) - h(x, y)$$

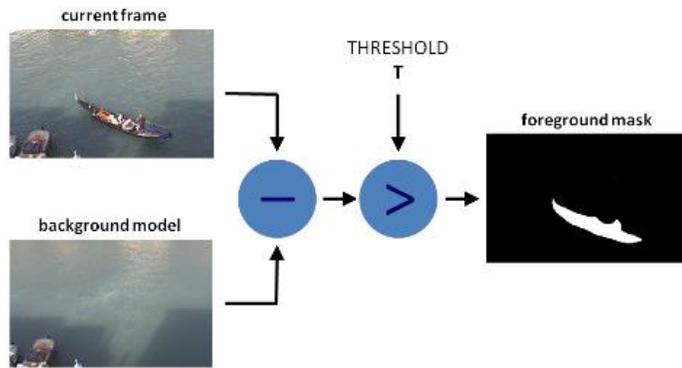


Ilustración 14: Generación de máscara a partir de fondo e imagen.

## 2.2.5 Imágenes a color

Las imágenes digitales a color están compuestas, generalmente, por 3 imágenes en escala de grises que son ponderadas por un espacio de color, de esta manera, el ojo humano puede identificar una mayor cantidad de valores, pero más importante, digitalmente contienen mucha más información.

El espacio de color más utilizado es RGB (por sus siglas en inglés *red*, *green*, *blue*; rojo, verde y azul en español, respectivamente), pero también, existen otros como CMY (*cyan*, *magenta*, *yellow*), YCbCr y HSV [9].

### 2.2.5.1 RGB

El espacio de color RGB se describe como un cubo en coordenadas cartesianas donde cada imagen, en escala de gris, pondera sobre uno de los ejes que corresponden a los colores rojo, verde y azul, quedando en la diagonal espacial los colores grises, tal como se muestra en la Ilustración 15. Este cubo también define al espacio CMY, ya que este es el inverso del espacio RGB [4].

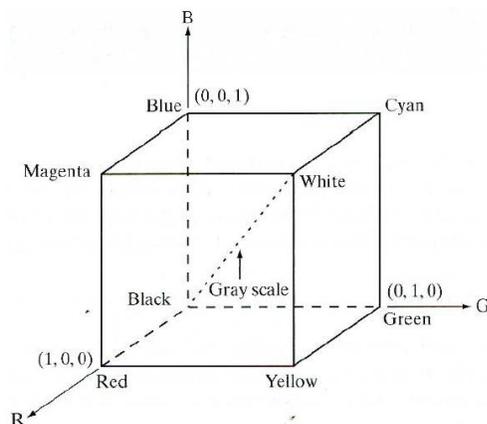


Ilustración 15: Cubo en coordenadas cartesianas normalizadas describiendo el espacio RGB.

### 2.2.5.2 YCbCr

El espacio YCbCr es creada buscando la necesidad de reducir el tamaño de almacenamiento de una imagen para poder comprimir datos y/o aumentar la velocidad de transmisión de una imagen o cadena de ellas. Esto se hace por medio de la descomposición de una imagen RGB en una imagen de luminosidad llamada Y, que debe ser almacenada en buena calidad y 2 componentes de crominancia (Cb para azul y Cr para rojo) que pueden ser comprimidas o sub muestreadas sin perder nivel de detalle en la imagen resultante, y solo se pierden detalles a nivel de colores. La utilidad de este espacio en el presente trabajo no es perder nivel de detalle de los colores, sino que extraer las imágenes de crominancia para hacer la segmentación.

En seguida, se muestran las ecuaciones básicas para convertir una imagen digital RGB, de 8bits de resolución, al espacio YCbCr. Se considera como  $R$ ,  $G$  y  $B$  las intensidades de rojo, verde y azul en su propio espacio, respectivamente:

$$Y = 0.99 * R + 0.587 * G + 0.114 * B$$

$$Cb = -0.172 * R - 0.339 * G + 0.511 * B + 128$$

$$Cr = 0.511 * R - 0.428 * G - 0.083 * B + 128$$

### 2.2.5.3 HSV

El espacio HSV [10] es creado con un cambio de coordenadas cartesianas de RGB a cilíndricas donde H o matiz (en inglés *hue*) es el ángulo (rojo en 0°), S o saturación es el radio y V o valor es la altura. A continuación, se muestra el algoritmo necesario para pasar del espacio RGB a HSV y en la Ilustración 16, se pueden ver los colores del espacio HSV en su forma de coordenadas cilíndricas.

- **Paso 1**, buscar el máximo y mínimo:

$$M = \max(R, G, B) \quad m = \min(R, G, B)$$

- **Paso 2**, normalizar los valores RGB:

$$r = (M - R)/(M - m), \quad g = (M - G)/(M - m), \quad b = (M - B)/(M - m)$$

- **Paso 3**, calcular el valor V:

$$V = \max(R, G, B) / 256$$

- **Paso 4**, calcular la saturación S:

$$\text{Si } M = 0, \text{ entonces } S = 0 \text{ y } h = 180^\circ$$

$$\text{Si } M \neq 0, \text{ entonces } S = (M - m)/M$$

- **Paso 5**, calcular el matiz H:

$$\text{Si } R = M, \text{ entonces } H = 60(b - g)/360$$

Si  $G = M$ , entonces  $H = 60(2 + r - b)/360$   
 Si  $B = M$ , entonces  $H = 60(4 + g - r)/360$   
 Si  $H \geq 1$ , entonces  $H = H - 1$  y si  $H < 0$ , entonces  $H = H + 1$

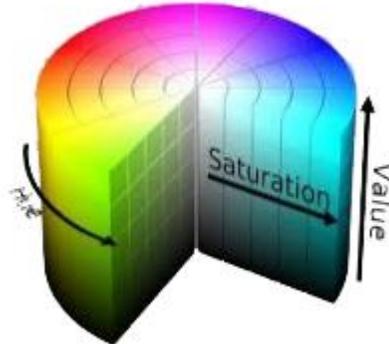


Ilustración 16: Cilindro de colores del espacio HSV.

El espacio HSV es ampliamente utilizado para segmentación en el campo de visión por computador, por lo que puede ser de gran ayuda para éste trabajo.

### 2.2.6 Resolución espacial y de intensidad

Las imágenes digitales tienen ciertas propiedades que son dadas por el CCD que las crea y pueden ser modificadas con cierta pérdida de información. La resolución espacial son las dimensiones de una imagen medida en píxeles, al disminuir la resolución, la imagen se verá más pequeña perdiendo información de detalles (ver Ilustración 17). Al aumentar la resolución, se agregan píxeles cuya información no viene de un muestreo del mundo real, sino que, de operaciones aritméticas entre píxeles vecinos, por lo que la imagen se verá más borrosa (ver Ilustración 18).



Ilustración 17: Imagen de 1024x1024 píxeles submuestreada hasta 32x32 píxeles.

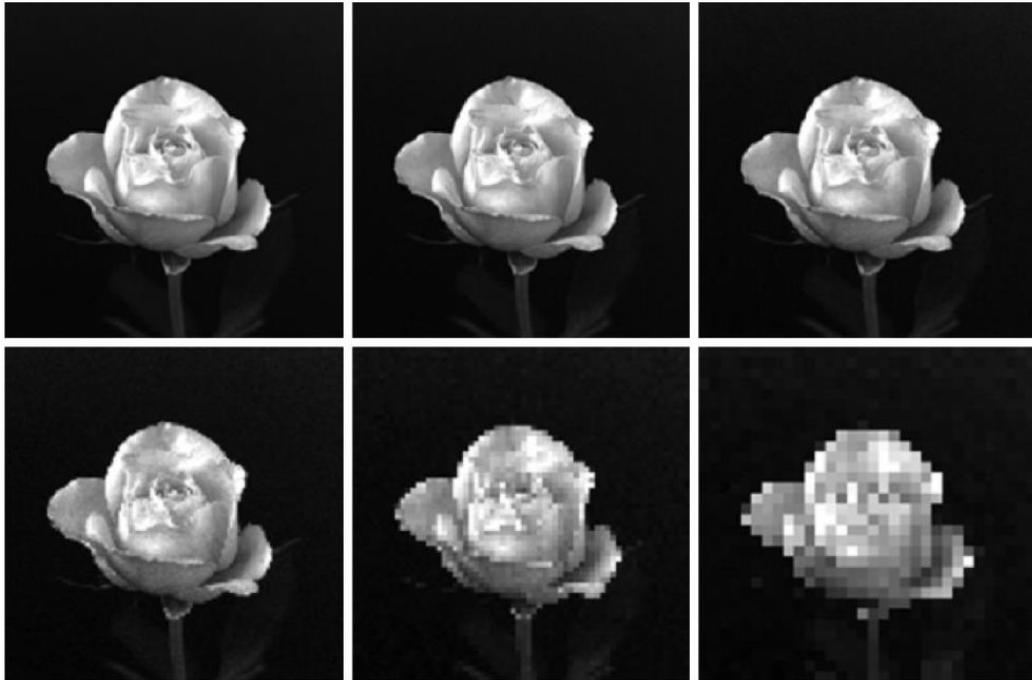


Ilustración 18: Imágenes de Ilustración 17 ampliadas hasta 1024x1024 píxeles.

Se debe tener especial cuidado con la diferencia entre capturar una imagen en alta resolución y aumentar la de una imagen capturada a baja resolución, ya que, en la primera opción, todos los píxeles provienen de un muestreo del mundo real y la imagen debiese ser nítida [3] [4].

Otra propiedad importante es el muestreo de intensidad, el cual está ligado al convertor análogo digital del CCD y se refiere al menor cambio discernible entre una intensidad y otra, también puede entenderse como la cantidad de niveles en que la imagen es representada. En la Ilustración 19, se puede notar la calidad de la imagen al disminuir mucho los niveles de intensidad. Normalmente, se utilizan convertidores de 8-bits, lo que corresponde a 256 niveles de intensidad.

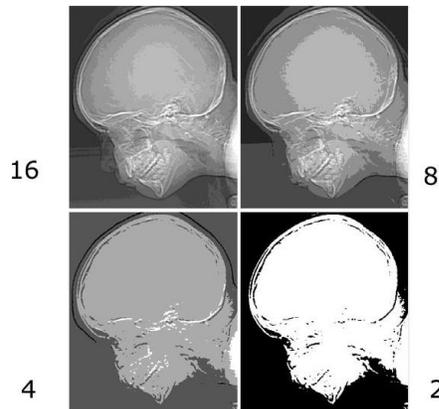


Ilustración 19: Imagen muestreada con 16, 8, 4 y 2 niveles de intensidad.

## 2.2.7 Procesamiento de imágenes

Una vez que se hace adquisición de la imagen por medio del CCD, el computador puede hacer distintos tipos de procesamiento sobre esta, tales como segmentación, procesamiento de color, procesamiento morfológico, compresión, reconocimiento de objetos, entre otros. El presente trabajo centrará su atención en las 3 primeras, ya que estas permitirán extraer suficiente información de la garra presente en cada imagen, las cuales se explican a continuación.

### 2.2.7.1 Segmentación

Consiste en la separación de una imagen en los objetos que la constituyen. En el caso de las garras consiste en separar lo que es el fondo, garra, cayo y cuero como objetos independientes dentro de una misma imagen. Para lograr la segmentación son necesarias otras herramientas de procesamiento tales como detección de bordes o procesamiento de color.

En la Ilustración 20, se puede ver una imagen de frutas a la izquierda y a la derecha el resultado de un proceso de segmentación, resaltando en distintos colores cada fruta.

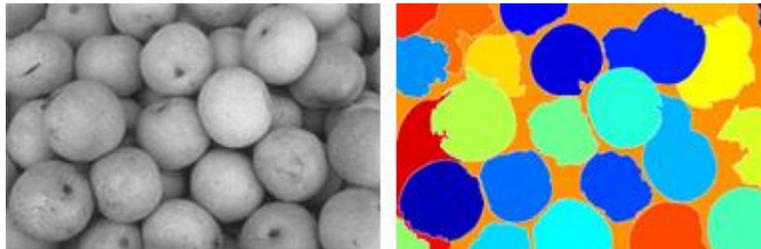


Ilustración 20: Segmentación de frutas en una imagen.

### 2.2.7.2 Procesamiento de color

Para poder hacer procesamiento de color, se debe establecer el espacio a utilizar, para luego, hacer la conversión necesaria. Colores parecidos ocuparán zonas cercanas en cada espacio, por lo que definen zonas en las que se pueden detectar.

Si se dan muestras representativas de los colores de interés, se pueden estimar vectores que den un promedio del color. Al ingresarse un color de prueba, este puede ser segmentado según la menor distancia a los vectores promedio de cada clase o se pueden definir regiones alrededor de dichos vectores promedio dentro de las cuales los nuevos colores de prueba serán segmentados con esa etiqueta (ver Ilustración 21).

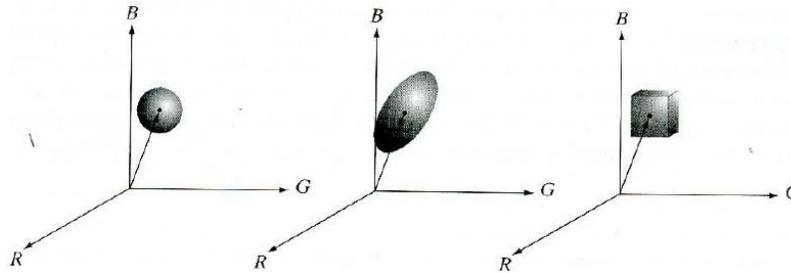


Ilustración 21: Tres formas distintas de definir regiones alrededor de los vectores promedio.

Las 3 dimensiones del color también pueden ser utilizadas como entradas de un algoritmo de clasificación que serán explicados en la sección 2.4.

### 2.2.7.3 Procesamiento morfológico

El procesamiento morfológico consiste en procesar las formas dentro de una imagen. Usando filtros específicos llamados elementos estructurales, se pueden aplicar transformaciones a la imagen a partir de las relaciones entre pixeles vecinos. Las transformaciones más comunes son la erosión y dilatación, descritos a continuación.

#### a. Erosión

La erosión se aplica en imágenes binarias y se usa para contraer las zonas de valor binario 1. En la Ilustración 22 se puede ver que, al aplicar erosión a la imagen de la izquierda, algunos de los cuadrados blancos desaparecen y solo sobreviven los más grandes.

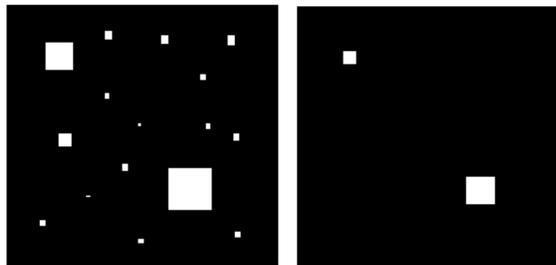


Ilustración 22: Imagen binaria y una erosión de la misma.

Es importante mencionar que existen distintos tipos de elementos estructurales para erosionar y muchos de ellos no conservan las formas de las zonas originales.

#### b. Dilatación

La dilatación se aplica en imágenes binarias y se usa para expandir las zonas de valor binario 1. En la Ilustración 23 se puede ver que, al aplicar dilatación sobre imagen de la izquierda, se obtienen los cuadrados blancos en su tamaño original sin que vuelvan a aparecer los cuadrados pequeños.

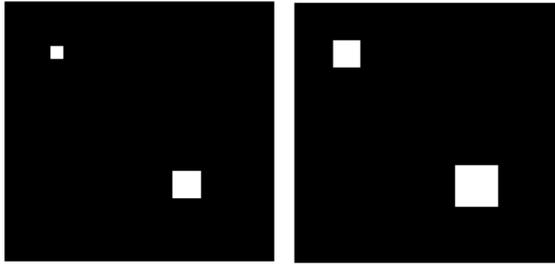


Ilustración 23: Imagen binaria y la dilatación de la misma.

Al igual que los elementos estructurales para erosionar, los utilizados para dilatar, muchas veces provocan una deformación en el objeto.

## 2.3 Extracción de características

La extracción de características consiste en la obtención de información por medio de modelos matemáticos que resuman el contenido de un objeto o imagen, con el fin de simplificar el proceso de aprendizaje de los objetos a reconocer.

Existen diversos tipos de modelos, que tendrán mejor o peor rendimiento en función al tipo de objeto a reconocer y a las condiciones impuestas por el medio en el que se capta la imagen (la luz controlada o no, distancia al objeto a reconocer conocida o no, etc.). Se pueden usar histogramas de color o escala de grises, estadísticos y otras formas de cuantificar variables de la realidad. Durante este trabajo se utilizará una forma de cuantificar defectos por medio del conteo de píxeles en una máscara de segmentación del mismo, ésta cuantificación será utilizada como características de cada muestra para su clasificación.

### 2.3.1 Histogramas de escala de grises

Para definir un histograma de color es mejor partir definiendo lo básico, que es el histograma de escala de grises.

Una imagen en escala de grises que, comúnmente, se encuentra con una resolución de 8 bits, es una matriz donde cada elemento hace referencia a un píxel y el nivel de gris que este representa, mostrando negro para el valor 0 y blanco para 255.

El histograma es un conteo de los valores, agrupando en cada elemento de un arreglo de largo 256 la cantidad de píxeles de cada nivel de gris. Dicho arreglo es normalmente graficado como barras, como se muestra en la Ilustración 24.

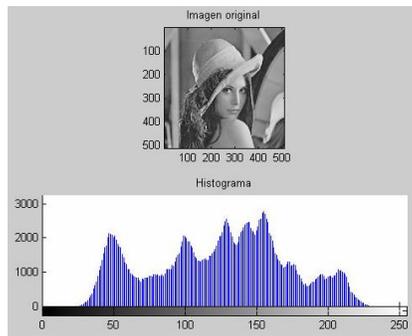


Ilustración 24: Histograma de escala de gris.

### 2.3.2 Histogramas de color

Una imagen de color se representa por 3 imágenes en escala de grises que se ponderan por los colores RGB para formar el color.

El histograma de color es la concatenación de los 3 histogramas de escala de gris que conforman la imagen a color, en la Ilustración 25 se muestran los histogramas de color de una imagen.

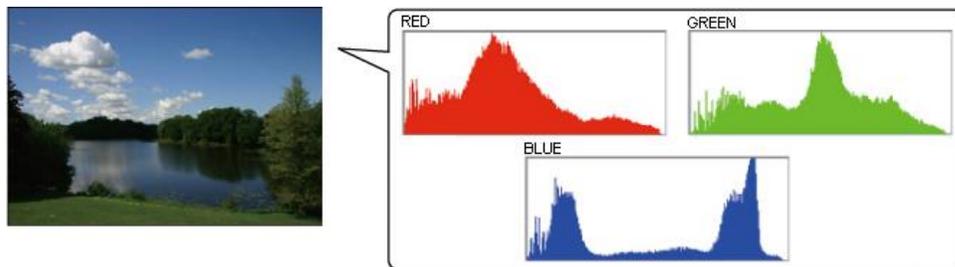


Ilustración 25: Histogramas de color.

### 2.3.3 Entropía de Shannon

En teoría de la información, la entropía mide la incertidumbre de la fuente de información. Pero la definición que aplica mejor al caso en estudio, es la entropía como cantidad de información promedio que aportan los datos, donde los datos más comunes son los que aportan menos información y los menos comunes los que más aportan. En procesamiento de imágenes la intensidad de los píxeles son los datos y las intensidades que más se repiten o que son más probables no aportan tanta información como los píxeles cuya intensidad es poco común.

Sea una imagen  $X$  con  $i$  píxeles de intensidad  $x_i$  y probabilidad  $p(x_i)$ , la entropía de Shannon  $H(X)$  se define como

$$H(X) = - \sum_i p(x_i) \log_2 p(x_i)$$

## 2.4 Inteligencia Computacional

La inteligencia computacional estudia y desarrolla metodologías para que los computadores sean capaces de tomar una decisión en base a un aprendizaje previo sobre el problema en cuestión. El objetivo principal de ésta área, es capacitar máquinas para que se especialicen en la realización de tareas que antes hacían humanos, logrando desarrollarlas con mayor eficiencia, en un menor tiempo y con menor tasa de error. Siempre será necesario contar con un set de información para que el computador pueda tomar una decisión, ejemplos de estas decisiones son, clasificar entre conjuntos, predecir tendencias, caracterizar niveles, o generar patrones según grados de similitud, entre otras.

Los seres humanos, a medida que repiten una experiencia, aprenden de esta y logran obtener mejores resultados en las siguientes iteraciones, la inteligencia computacional copia aquel comportamiento logrado que el resultado final este dado por las iteraciones del proceso de aprendizaje.

Existen 2 tipos de aprendizajes en la inteligencia computacional [11], supervisado y no supervisado. El aprendizaje no supervisado no considera la etiqueta de la muestra para su entrenamiento, en este caso, el sistema tiene que ser capaz de reconocer patrones en los datos para poder separar las nuevas entradas. Estos sistemas están pensados para separar elementos en agrupaciones de características similares y no son capaces de etiquetarlas con una clase definida. A modo de ejemplo de clasificación de perros y gatos, un sistema entrenado sin supervisión solo será capaz de identificar animales de tipo 1 y animales de tipo 2, sin saber cuál de ellos corresponde a gato y cual a perro.

En el aprendizaje supervisado, el algoritmo es entrenado con un conjunto de entradas que contienen tanto las características de una muestra como la clase a la que pertenece. Así, el sistema ajusta parámetros internos que le permiten clasificar otras muestras. Estos son los sistemas utilizados para clasificación y entre ellos se encuentran árboles de decisión, redes neuronales artificiales y otros.

En el caso de las garras de pollo, es de interés que un computador entrenado defina el calibre de una garra y no que distinga que 2 garras son de calibres distintos, por lo que se utilizarán métodos de aprendizaje supervisado.

### 2.4.1 Redes neuronales artificiales

El trabajo en las redes neuronales artificiales ha sido motivado por la idea de imitar el comportamiento del cerebro humano y dejar de lado el

comportamiento digital de la computación convencional. El cerebro es un computador muy complejo, no lineal y de procesamiento paralelo. Tiene la capacidad de organizar sus neuronas para que cumplan funciones determinadas mucho más rápido de lo que lo hace un computador [12] [11].

Una Red Neuronal Artificial (RNA) está diseñada para modelar la forma en que el cerebro realiza determinadas funciones o tareas. Cada neurona hace una separación lineal del problema y al juntar muchas neuronas en una red, estas son capaces de resolver problemas no lineales de alta complejidad.

Los grandes beneficios de utilizar redes neuronales para clasificación, es que funcionan como caja negra, es decir, se estudian las entradas y salidas, quitándole importancia a lo que pasa en su interior. Sin embargo, se explicará brevemente como están formadas las redes neuronales y en que se basa su funcionamiento.

#### 2.4.1.1 Modelo elemental de una neurona

Una neurona es la unidad básica de la red y se modela con pesos sinápticos ( $w_{ki}$ ) para cada entrada ( $x_i$ ), un combinador lineal de las entradas ( $\Sigma$ ), un bias o sesgo ( $b_k$ ) y una función de activación no lineal ( $\varphi$ ) para determinar la salida ( $y_k$ ).

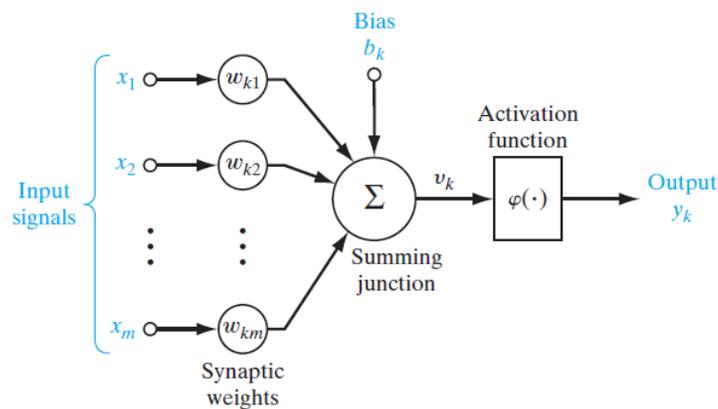


Ilustración 26: Modelo de neurona.

$$u_k = \sum_{j=1}^m w_{kj}x_j \quad v_k = u_k + b_k \quad y_k = \varphi(v_k)$$

##### a. Bias

Tiene el efecto de aplicar una función afín a la salida del combinador lineal, suele agregarse como una entrada extra de peso sináptico  $b_k$ .

b. Función de activación

Es una función no lineal que indica el grado de actividad de la neurona dada las entradas con el fin de establecer una salida.

c. Peso sináptico

Es una constante asociada a la unión entre 2 neuronas. El peso sináptico se escoge al azar y es corregido automáticamente en el proceso de aprendizaje de la red.

#### 2.4.1.2 Arquitectura de Redes Neuronales

Las neuronas se ordenan e interconectan para formar una red. Para darle forma a esta red, se identifican capas de neuronas que cumplen distintas funciones.

a. Capa de entrada

Es la primera capa que se encarga solo de distribuir las entradas a la capa siguiente. Se dice que son neuronas, pero no cumplen el modelo matemático de una.

b. Capa de salida

Es la última capa de neuronas que representan a cada clase de la clasificación. Cuando una neurona de la capa de salida da una salida mayor que las demás significa que la muestra corresponde a la clase asociada a esa neurona.

c. Capa oculta

La capa oculta es la capa o capas que se encuentran entre la capa de entrada y la de salida. Usar muchas neuronas en la capa oculta puede causar un sobre ajuste a la muestra de entrenamiento y muy pocas puede causar una falta de ajuste.

Para saber el número de neuronas de la capa oculta solo existen heurísticas, las cuales dependen de la cantidad de neuronas en la entrada, en la salida y/o la cantidad de muestras, algunas de estas heurísticas son: la pirámide geométrica, algoritmos genéticos y el logaritmo en base 2 de la cantidad de muestras en el conjunto de entrenamiento.

Se sabe que debe haber tantas neuronas en la salida como clases de la clasificación, y tantas neuronas de entrada como dimensiones en el vector de características.

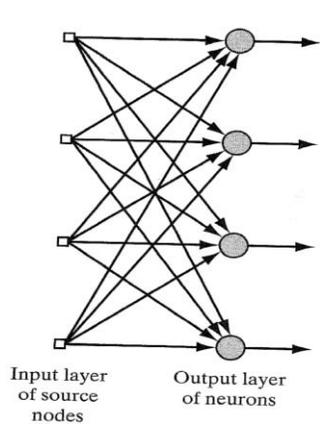


Ilustración 27: Red feed forward de una capa.

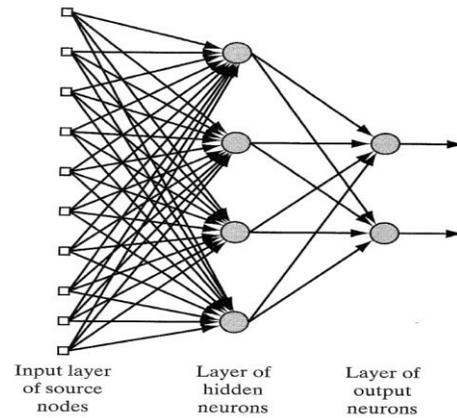


Ilustración 28: Red feed forward multicapa (1 capa oculta).

### 2.4.1.3 Entrenamiento

El procedimiento utilizado para entrenar la red se llama algoritmo *Backpropagation* y es la forma en que se modifican los pesos sinápticos de la red de una forma ordenada para obtener el objetivo deseado. La modificación de pesos sinápticos hace que cada neurona priorice de manera distinta su reacción dentro del proceso global.

El algoritmo *Backpropagation* corrige los pesos sinápticos según una señal de error generada entre la salida de la red y la salida esperada. Dicha señal de error viaja desde la salida a la entrada (propagación hacia atrás) propagándose según la importancia que dio cada neurona a la salida original.

En el presente trabajo se utilizarán redes neuronales ya implementadas en Matlab, por lo que explicar más a fondo el algoritmo de entrenamiento no es una prioridad.

### 2.4.2 Árboles de decisión

Un árbol de decisión es una secuencia de preguntas, con respuesta binaria, dispuestas en forma de árbol invertido. La primera pregunta va en la parte superior y se le llama nodo raíz (*root* en inglés), desde el cual salen 2 ramas o posibles caminos a tomar según la respuesta. Cada rama lleva a un nuevo nodo donde se encontrará otra pregunta. El árbol termina en nodos especiales llamados hojas que no contienen una pregunta, sino que el resultado de clasificación [11].

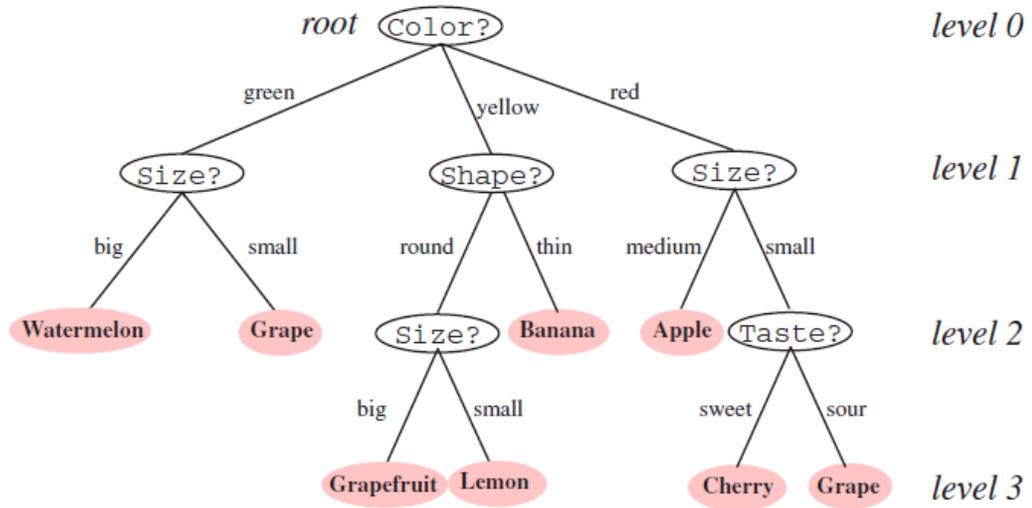


Ilustración 29: Árbol de decisión de 3 niveles.

El entrenamiento de un árbol, se hace minimizando la impureza de Gini. Aquel valor es una medida de la frecuencia con que un elemento al azar dentro de los datos de entrenamiento, es clasificado de forma incorrecta. El algoritmo anterior se encuentra implementado en Matlab y una explicación más profunda de él, escapa del objetivo de este trabajo.

### 2.4.3 K-vecinos más cercanos

A diferencia de los clasificadores anteriores, el algoritmo *K-Nearest Neighbors* (KNN) es un método no paramétrico que basa su predicción en los datos de entrenamiento. La muestra a clasificar busca sus K vecinos más cercanos en un espacio de tantas dimensiones como características tiene el problema. Al nuevo dato se le asignará la clase que más se repita en sus K vecinos [11].

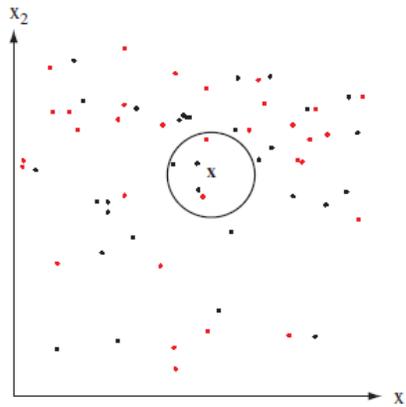


Ilustración 30: Ejemplo bidimensional de KNN.

En la Ilustración 30 se observa una muestra 'x' que marca con una circunferencia sus 5 vecinos más cercanos, 3 de ellos son negros y 2 son rojos; por lo que 'x' será clasificado como negro.

## 2.5 Herramientas de reducción de dimensionalidad

### 2.5.1 Cuantización de histogramas

Como se mencionó en la sección 2.3.1, un histograma es un arreglo de largo 256. Es muy probable que elementos cercanos dentro del mismo, tengan valores parecidos, por lo que hacen que tener un arreglo tan largo sea redundante.

La cuantización de un histograma consiste en agrupar elementos vecinos del arreglo con el fin de obtener uno de menor dimensionalidad. Se obtiene el mismo resultado si se baja la resolución de la imagen de 8 a 7 o menos bits para luego calcular el histograma.

### 2.5.2 Análisis de componentes principales

El análisis de componentes principales, más conocido por sus siglas en inglés (*Principal Components Analysis*), PCA es un procedimiento estadístico que, mediante una transformación ortogonal, convierte variables correlacionadas en no correlacionadas [12].

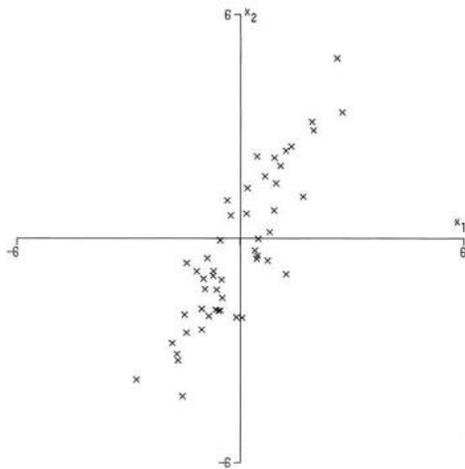


Ilustración 31: 50 observaciones graficadas sobre variables  $x_1, x_2$  correlacionadas.

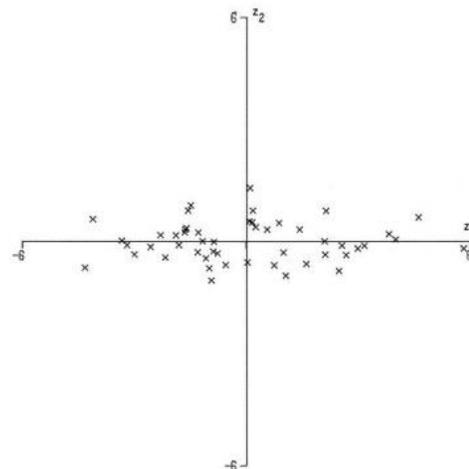


Ilustración 32: 50 observaciones graficadas sobre variables  $z_1, z_2$  no correlacionadas.

Al observar la Ilustración 31 se puede ver que tanto la variable  $x_1$  como  $x_2$  aportan gran información sobre cada observación. En cambio, en la Ilustración 32, la variable  $z_1$  concentra gran parte de la información (mayor varianza), pudiendo incluso, prescindir de la variable  $z_2$  (menor varianza).

Lo anterior, se puede repetir para un caso de  $n$ -dimensiones, creando una transformación ortogonal a  $n$ -nuevas variables que pueden ser ordenadas de mayor a menor varianza.

La reducción de dimensionalidad se aplica al demostrar que solo una parte de las nuevas variables concentran la mayor parte de la información de las observaciones, permitiendo obviar las dimensiones que aportan de manera poco significativa en el análisis.

## 2.6 Herramientas de interpretación de resultados

### 2.6.1 Matriz de confusión

La matriz de confusión contiene información sobre la clase real (columnas) y la clase predicha por el sistema de clasificación (filas), es una forma de visualizar la precisión y exactitud de un sistema de clasificación.

Para construir la matriz, se debe clasificar el conjunto de prueba con el clasificador previamente entrenado. Cada elemento de salida se ubica en la fila que asigna el clasificador y en la columna de su clase real. Así, los elementos de la diagonal serán aciertos y el resto de los elementos son errores.

Sea un problema de predicción de clases, en el que las salidas se etiquetan positivas ( $p$ ) o negativas ( $n$ ). Hay cuatro posibles resultados a partir de un clasificador como el propuesto. Si el producto de una exploración es  $p$  y el valor real también es  $p$ , entonces ese acierto se conoce como un Verdadero Positivo (VP); sin embargo, si el valor real es  $n$  entonces se conoce como un Falso Positivo (FP) o error de tipo 1. De igual modo, tenemos un Verdadero Negativo (VN) cuando tanto la exploración como el valor dado son  $n$ , y un Falso Negativo (FN) cuando el resultado de la predicción es  $n$ , pero el valor real es  $p$ , caso conocido como erro de tipo 2.

		Valor en la realidad		total
		$p$	$n$	
Predicción outcome	$p'$	Verdaderos Positivos	Falsos Positivos	$P'$
	$n'$	Falsos Negativos	Verdaderos Negativos	$N'$
total		$P$	$N$	

Ilustración 33: Matriz de confusión para clasificación binaria.

Esto se puede extender a imágenes binarias, considerando que un pixel blanco o 1 es un caso positivo y un pixel negro o 0 es un caso negativo. Al comparar 2 máscaras se tendrán número distintos de VP, VN, FP y FN, para hacerlos comparables es necesario calcular la tasa de aciertos y errores

- Tasa de verdaderos positivos (TVP) es la sensibilidad de la clasificación, es muy utilizada su sigla en inglés (TPR):

$$TVP = VP / (VP + FN)$$

- Tasa de verdaderos negativos (TVN) es la especificidad de la clasificación:

$$TVN = VN / (FP + VN)$$

- Tasa de falsos negativos (TFN) es la tasa de error tipo 2:

$$TFN = FN / (VP + FN)$$

- Tasa de falsos positivos (TFP) es la tasa de error tipo 1, es muy utilizada su sigla en inglés (FPR):

$$FPR = FP / (FP + VN)$$

Otros valores importantes son la precisión y la exactitud:

- Precisión =  $VP / (VP + FP)$
- Exactitud =  $(VP + VN) / (\text{Población total})$

**Confusion Matrix**

	A	B	C	R	CHOICE	
A	31 19.9%	7 4.5%	0 0.0%	0 0.0%	0 0.0%	81.6% 18.4%
B	0 0.0%	12 7.7%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
C	2 1.3%	14 9.0%	41 26.3%	0 0.0%	2 1.3%	69.5% 30.5%
R	0 0.0%	0 0.0%	0 0.0%	24 15.4%	3 1.9%	88.9% 11.1%
CHOICE	0 0.0%	0 0.0%	1 0.6%	0 0.0%	19 12.2%	95.0% 5.0%
	93.9% 6.1%	36.4% 63.6%	97.6% 2.4%	100% 0.0%	79.2% 20.8%	81.4% 18.6%
	A	B	C	R	CHOICE	
	<b>Target Class</b>					

Ilustración 34: Matriz de confusión.

En la Ilustración 34 se puede ver una matriz de confusión más completa y la que se utilizará para comparar los resultados importantes de este trabajo. Además de mostrar un valor en cada celda, se incluye el porcentaje del total de muestras correspondiente a dicho valor, pero más importante que este número, son los porcentajes que aparecen en las celdas grises y celeste que no corresponden a ninguna clase. En la columna de la derecha, las celdas grises contienen el porcentaje de sensibilidad en verde y

la falta de sensibilidad en rojo; en la fila de abajo, las celdas grises contienen la precisión en verde y la falta de ella en rojo; por último, la celda celeste contiene la exactitud total del clasificador que se está analizando.

## 2.6.2 Curva ROC

Las curvas características de funcionamiento del receptor (*Receiver Operating Characteristic*), son una representación gráfica de la sensibilidad versus la especificidad para un sistema de clasificación binaria.

El análisis de la curva ROC proporciona herramientas para seleccionar modelos posiblemente óptimos y descartar modelos sub óptimos basándose solamente en los resultados de clasificación y obviando el coste ligado a cada modelo.

En el espacio ROC se define por la tasa de falsos positivos y la tasa los verdaderos positivos como ejes X e Y respectivamente (ver Ilustración 35). Se busca que un clasificador tenga una tasa de FP muy pequeña y una tasa de VP muy cercana a 1, por lo que la distancia a la coordenada (0,1) determina la calidad del clasificador (ver Ilustración 36).

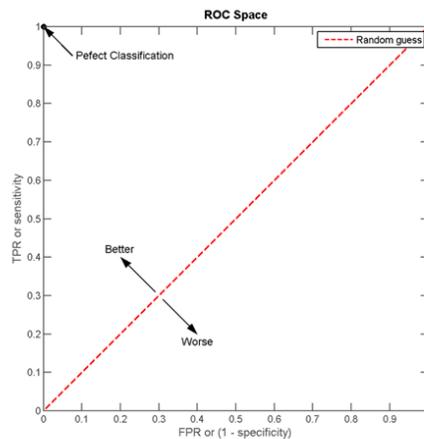


Ilustración 35: Espacio ROC.

Los clasificadores discretos representan su rendimiento en un punto único dentro del espacio ROC. Los clasificadores más complejos, como reglas bayesianas, tienen como salida una medida de probabilidad de pertenencia a una de las 2 clases, por lo que requieren un umbral a partir del cual se toma la decisión de clasificación. Es por esto que clasificadores de este tipo generan curvas en el espacio ROC y el umbral se determina por el punto más cercano al óptimo en el espacio ROC (coordenada (0,1)). Además, el cambio en algún parámetro de un clasificador genera cambios en su rendimiento (reflejado en la curva ROC), estos parámetros pueden ser optimizados observando la curva ROC que genera el clasificador con cada uno de ellos.

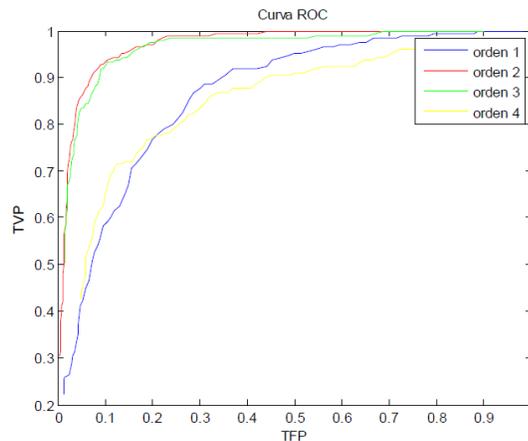


Ilustración 36: Curvas ROC para distintos parámetros de un clasificador.

## 2.7 Estado del arte

Hasta hace pocos años, los países desarrollados no exportaban garras de pollo ya que la demanda no era mucha localmente y no se había descubierto el mercado potencial en Asia del este. Esto se tradujo en una falta de tecnología aplicada a este proceso.

Hoy en día, la clasificación de garras de pollo se hace de forma manual, incluso en las faenadoras más modernas. Este trabajo es uno de los primeros acercamientos para automatizar este proceso.

La clasificación por peso no es de gran complejidad y ya existen máquinas que lo hacen [13], pero de todas maneras se comparará el peso de las muestras con la cantidad de píxeles que ocupan en las imágenes.

## 2.8 Motivación y aporte del Trabajo de Memoria de Título

En la planta faenadora de pollos de Agrosuper ubicada en San Vicente de Tagua Tagua (Región del Libertador Bernardo O'Higgins de Chile), se clasifican las garras de pollo para exportar las de mejor calibre. En este proceso trabajan entre 12 y 15 personas por turno, con 2 turnos diarios, esto significa un gran costo anual para la empresa y, por lo tanto, se hace interesante la automatización de este proceso.

Existe información de estudios orientados a la clasificación de fruta por medio de imágenes digitales, e incluso, hay empresas especializadas en la clasificación automatizada de fruta. Sin embargo, no existe ningún acercamiento en clasificar garras de pollo por medio de imágenes digitales, lo que lo convierte a esta memoria en un desafío interesante.

El objetivo de este trabajo es generar un algoritmo que sea capaz de clasificar garras de pollo por medio de imágenes con la menor tasa de error

posible. Para esto se evaluarán distintas metodologías de procesamiento de imágenes e inteligencia computacional, con el fin de seleccionar la más adecuada para este proceso y, en el futuro, implementarla en una línea automatizada de clasificación. Cabe destacar que, durante el trabajo a realizar se tenga presente que debe ser escalable a una aplicación industrial, por lo que variables como los tiempos de procesamiento, entre otras, son tomadas en cuenta, pero sin dedicarle mayor análisis, debido a los alcances definidos en esta memoria.

### 3 Creación de base de datos

Con el fin de basar este trabajo en información real, esta memoria considera datos de la empresa Agrosuper, la cual proporciona acceso a la línea procesadora de garras en la planta faenadora de San Vicente.

En la planta se crea la base de datos y se dispone de una muestra significativa y representativa de garras etiquetadas con su calibre real, definido por operadores especializados en el proceso de selección.

#### 3.1 Estructura para captura de imágenes

Para crear una base de datos de imágenes se debe tener en cuenta el procesamiento digital que se deben llevar a cabo con las imágenes. Es por esto que se debe controlar la iluminación, el color de fondo y las imágenes deben estar tomadas siempre desde la misma posición, lo que hace que sea necesaria una pequeña estructura que facilite el proceso.

En este tipo de casos, la mejor forma de iluminar el objeto es con una luz en forma de anillo que rodee el foco de la cámara [14], por lo que se construyó un soporte de PVC el cual ubica la cámara fotográfica justo en el centro de un tubo fluorescente en forma de anillo, tal como se puede ver en la Ilustración 37.



Ilustración 37: Estructura para toma de muestras.

En cuanto al fondo, necesita un color que haga contraste con los colores presentes en las garras, por lo que colores como rojo, amarillo o blanco no son buenas elecciones. No así los colores azul y verde, que no están presentes en las garras. Además, por los restos de agua que hay en las garras, el fondo debe ser de un material fácil de limpiar. Se hicieron pruebas con un fondo azul y se logró una segmentación suficientemente buena, por lo que se utilizó ese color de fondo para la creación de la base de datos.

### 3.2 Cámara fotográfica y su configuración

Una cámara fotográfica que no permita controlar la apertura del diafragma ni el tiempo de exposición, puede provocar que algunas imágenes salgan más claras y otras más oscuras (ver Ilustración 38). Lo anterior hace de este tipo de cámaras una mala alternativa para crear una base de datos.



(a)



(b)

Ilustración 38: (a) Imagen capturada con poca luz. (b) Imagen capturada con mucha luz.

Lo visto en la Ilustración 38 se podría corregir ecualizando las imágenes, pero las garras con defectos de color quedarían ecualizadas de manera distinta a las garras sanas (ver Ilustración 39).



(a)



(b)

Ilustración 39: (a) Imagen con poca luz ecualizada. (b) Imagen con mucha luz ecualizada.

Es necesario usar una cámara que permita fijar los parámetros de apertura de diafragma y tiempo de exposición. Para crear la base de datos utilizada en el presente trabajo se utilizó una cámara compacta marca Nikon modelo P90 con un objetivo 4.6-110.4 mm de enfoque automático, la distancia focal se fijó en 4.6 mm para evitar diferencias entre una sesión de muestras y otra.

Considerando la futura implementación del algoritmo, se fijó la exposición más corta permitida por la cámara correspondiente a 1/250 segundos, lo que impide que la imagen salga borrosa cuando la garra se encuentra en movimiento y la apertura del diafragma se ajustó acorde a la velocidad de obturación en F2.8, de manera que imagen salga con la claridad adecuada.

La cámara permite ajustar también el ISO del sensor. La luz que entra en el sensor de la cámara fue ajustada con la apertura del diafragma, por lo que el ISO se configuró en 200, lo más bajo permitido por la cámara, con el fin de no perder los detalles de la imagen.

### **3.3 Toma de muestras y etiquetas**

Las garras de pollo pueden tener defectos tanto en la parte superior como en la inferior, para obtener toda esta información, la base de datos es creada tomando 2 imágenes por cada garra de muestra (una por cada lado). Las garras son volteadas manualmente ya que aún no existe una forma automatizada de hacerlo.

Como se mencionó anteriormente, cada muestra es etiquetada por un operario especializado, definiendo su calibre, peso y comentarios específicos de los defectos de cada muestra, tales como presencia de hematomas, manchas, dimensiones del callo, raspaduras o quebraduras.

## 4 Máscara de segmentación

Para que las muestras puedan ser procesadas para la extracción de un vector de características, estas deben ser pre procesadas para separarlas del fondo por medio de una máscara de segmentación.

### 4.1 Segmentación con modelo de fondo

Una forma efectiva de separar el fondo del primer plano es crear un modelo de fondo promediando imágenes sin garra (ver Ilustración 40a) y restarlas a la imagen que contiene el objeto que se desea detectar (ver Ilustración 40b), de esta manera, se anularán los pixeles de fondo y resaltaran los pixeles que muestran la garra (ver Ilustración 40c).

Aplicar un filtro de media mostró ser de utilidad para eliminar el ruido de las imágenes resultantes, por lo que se aplica una ventana promediadora de 9x9 pixeles sobre imágenes de 960x880 pixeles.

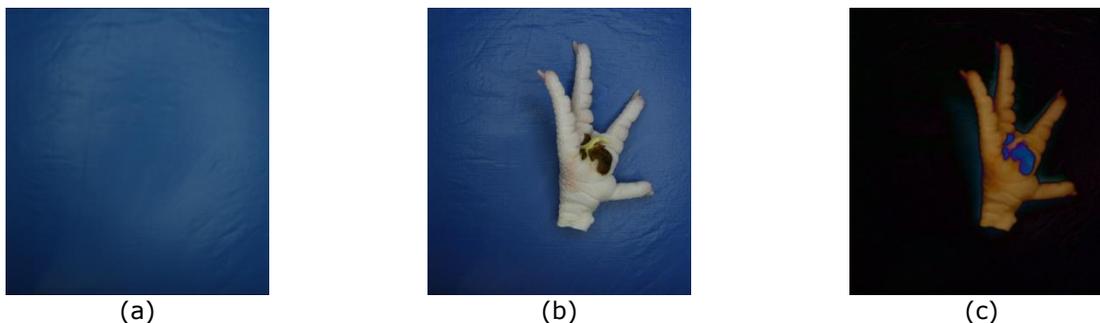


Ilustración 40: (a) Modelo de fondo. (b) Imagen caso I. (c) Resultado de restarle el modelo de fondo a la garra del caso I.

La imagen resultante toma sentido cuando es analizada en sus componentes RGB por separado y se descompone en rojo, verde y azul, como se muestra en la Ilustración 41.

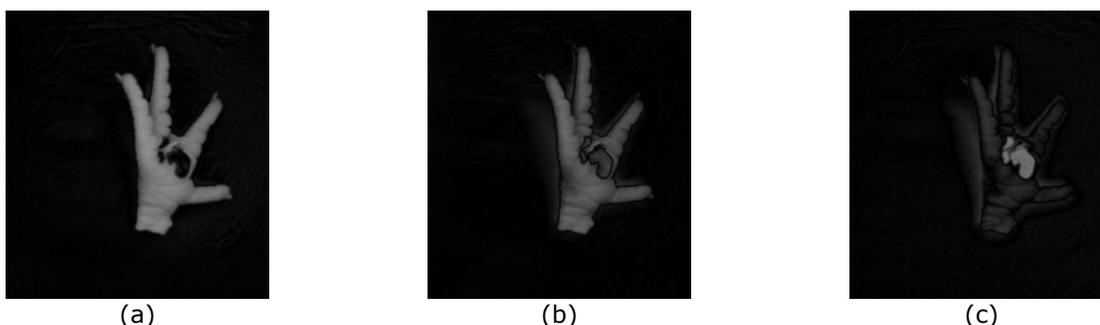


Ilustración 41: Componente rojo (a), verde (b) y azul (c) de imagen de fondo.

Un umbral para la componente roja (Ilustración 41a) y otro para la componente azul (Ilustración 41c) podrían dar lugar a una máscara para

segmentar la garra completa, de tal manera que todos los píxeles cuya intensidad sobrepase el límite en cada componente será considerado garra y aquellos que no lo hagan serán considerados fondo. En la componente roja los píxeles que quedan por debajo de la intensidad de corte son píxeles de fondo o callo, los cuales son diferenciados con la componente azul, donde el callo queda por sobre el umbral y el fondo por debajo.

Por medio de una serie de pruebas y observaciones se llega a un umbral de 50 para la componente roja y de 80 para la azul. La unión de ambas máscaras entregaría una máscara final para la garra completa (ver Ilustración 42). En la sección 6.1 se analizan con mayor profundidad los umbrales fijados.

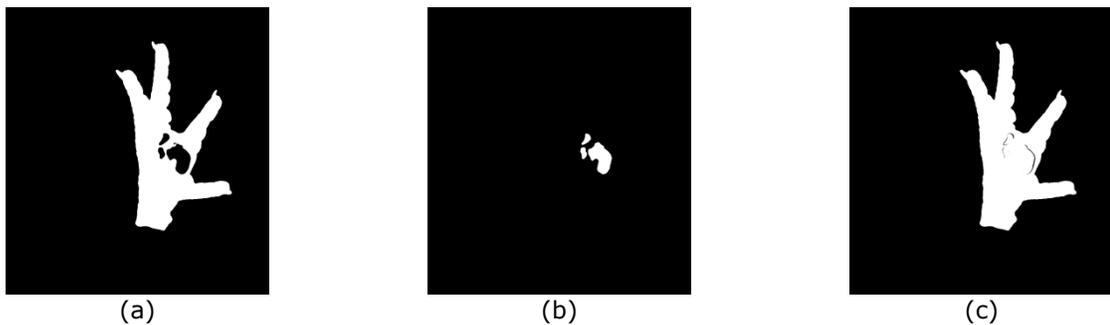


Ilustración 42: Máscara de garra sin callo (a), máscara de callo (b) y unión entre ambas (c).

Cuando la garra proyecta una sombra muy oscura, aumenta la diferencia con el modelo de fondo y se detecta callo donde no lo hay, agregando partes del fondo a la máscara de segmentación. En la Ilustración 43b se puede ver el efecto descrito anteriormente.



Ilustración 43: (a) Imagen caso II. (b) Máscara de segmentación caso II.

Un ajuste en el umbral de la componente azul hace que no se detecte ese tipo de sombras, pero al mismo tiempo hace que no se detecte el callo completamente, haciendo de los umbrales sobre una imagen de diferencia un método poco efectivo.

## 4.2 Segmentación con umbrales RGB

Aprovechando la diferencia de color entre la garra y el fondo se pueden estudiar las componentes RGB sin restar una imagen modelo de fondo (ver Ilustración 44). Al igual que en el método anterior se pueden buscar límites en la intensidad de cada componente que separen los píxeles de garra y de fondo.

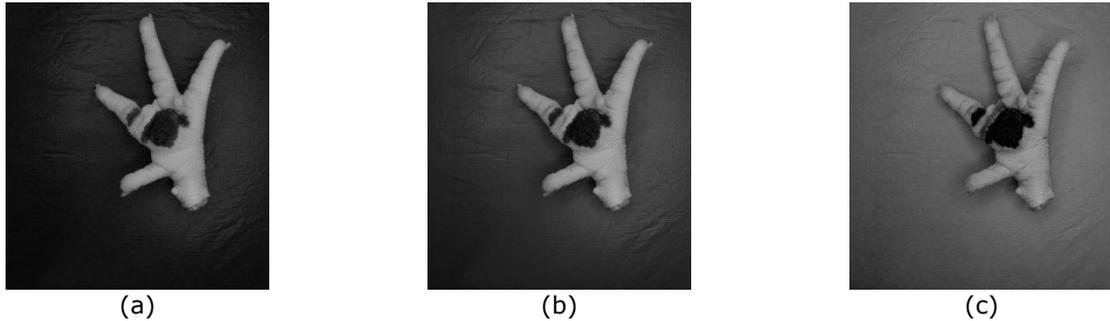


Ilustración 44: Componente roja (a), verde (b) y azul (c) de la imagen del caso II.

Por medio de observación y pruebas se fija un umbral tentativo de 90 sobre la componente roja, de tal manera de que los píxeles que superen esta intensidad sean los de garra sin callo y los que no lo hagan sean los de fondo y callo. La componente verde no es de gran utilidad ya que la garra se encuentra dentro del rango de intensidad ocupado por el fondo. En la componente azul el callo tiene menor intensidad que el resto de los elementos, por lo que se puede establecer un umbral; bajo él los píxeles serán considerados como callo, dejando al resto de la garra y al fondo por sobre él.

En la Ilustración 45a se muestra el resultado de aplicar el umbral indicado a la componente roja y en la Ilustración 45b un umbral de 20 sobre la componente azul (recordar que los casos positivos se encuentran por debajo del umbral). En esta última imagen se puede ver que las sombras siguen apareciendo en intensidades en que el callo comienza a desaparecer tal como muestra la Ilustración 45d que utiliza el mismo umbral. Al igual que en el método anterior, un umbral menor en la componente azul detectaría menos píxeles de callo y uno mayor haría aparece píxeles de fondo en la detección.

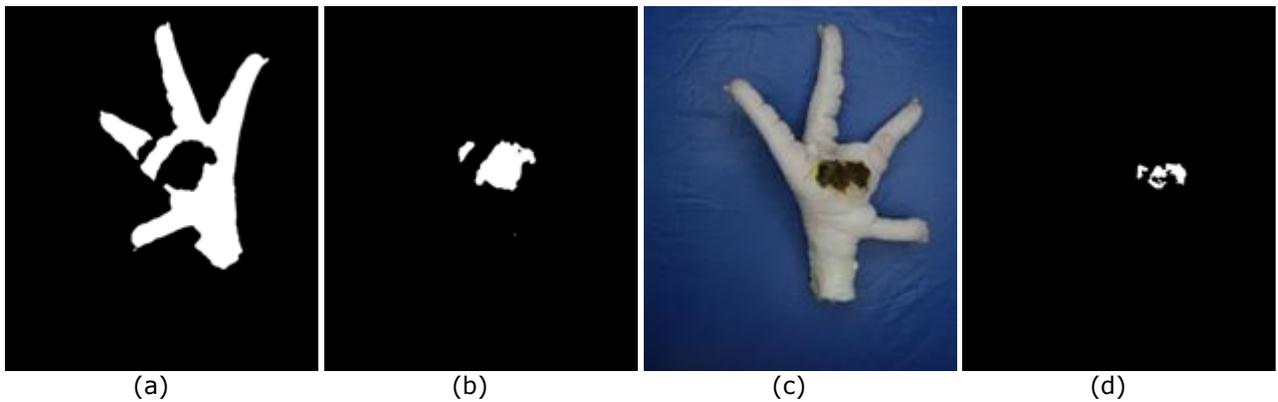


Ilustración 45: (a) y (b) muestran la máscara de garra y callo del caso II. (c) muestra el caso III y (d) su máscara de callo.

Antes de intentar arreglar el resultado aplicando herramientas de procesamiento de imágenes como operadores morfológicos, se estudiarán cambios de espacio de color y se analizarán sus componentes. Si otros espacios de color entregan un buen resultado, no es necesario arreglar este.

### 4.3 Segmentación con espacio de color YCbCr y umbral

En el espacio YCbCr los colores más azules respecto a la imagen en escala de grises se resaltan en la componente Cb, por lo que no es de extrañar que el fondo resalte al estudiar esta componente en una imagen de la base de datos.

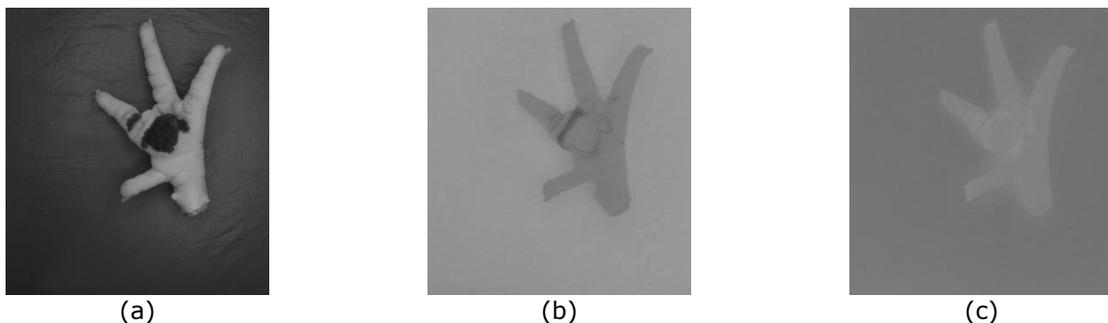


Ilustración 46: Componentes Y (a), Cb (b) y Cr (c) en la garra II.

Un análisis de la componente Cb (ver Ilustración 46b) hace notar que los píxeles de fondo superan la intensidad de 130 y la garra no, por lo que se puede establecer un umbral bajo el cual los píxeles sean considerados garra y definir como fondo aquellos que lo superen. En la Ilustración 47 se pueden ver 3 casos en que este método de segmentación funciona correctamente.

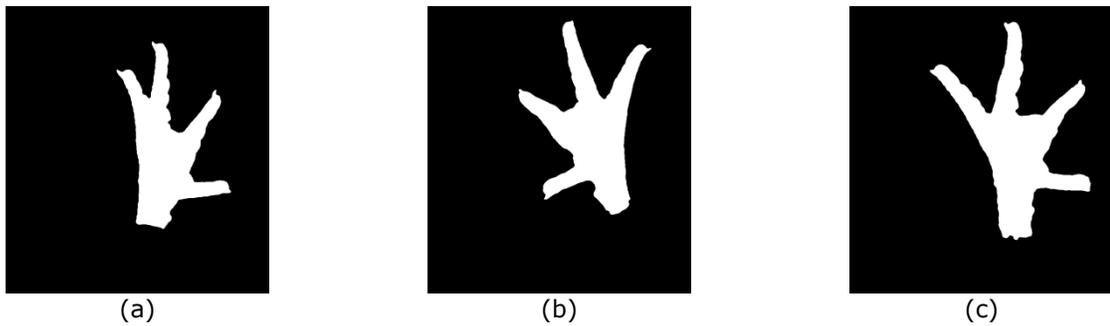


Ilustración 47: Máscara de garra caso I (a), II (b) y III (c).

En la sección 6.1.3 se analiza con más detalle el umbral propuesto para la componente  $C_b$ , ya que fue fijado en una intensidad de 130 en base a simple observación.

#### 4.4 Segmentación con espacio de color HSV y umbral

El espacio de colores HSV también es interesante de estudiar ya que además de ser recomendado por la literatura para hacer segmentación [15], el fondo resalta tanto en la componente de matiz como en la de saturación tal como se puede ver en la Ilustración 48a e Ilustración 48b.

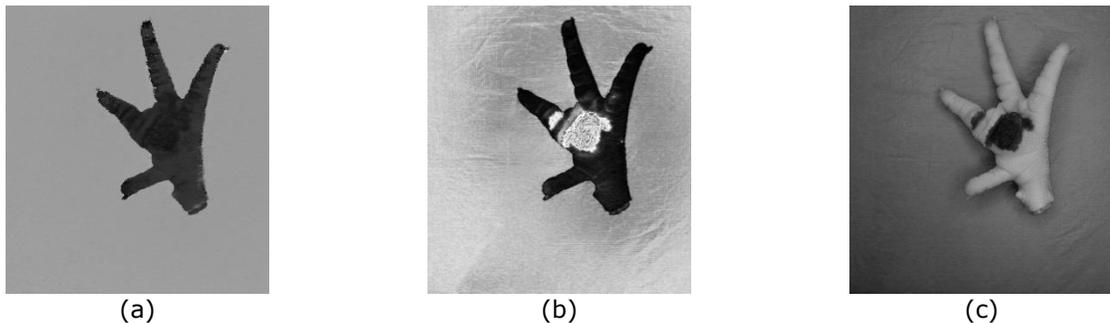


Ilustración 48: Componentes de matiz (a), saturación (b) y valor (c) de garra caso II.

En la componente de saturación, el callo resalta junto con el fondo y lo que se quiere es segmentar la garra completa, por lo que solo es de utilidad la componente de matiz. Se puede aislar el fondo en el rango 0.55-0.65 de esta componente (la intensidad en el espacio HSV se mide en el rango 0-1 y no en el rango 0-255 como otros espacios), por lo que aplicando umbrales en dichos límites se puede aislar el fondo y detectar como garra todo lo que quede fuera de él. En la Ilustración 49 se pueden ver 3 casos en que este método de segmentación funciona correctamente.



(a)



(b)



(b)

Ilustración 49: Máscara de garra caso I (a), II (b) y III (c).

En la sección 6.1.4 se analiza con más detalle los umbrales propuestos para la componente de matiz, ya que fueron fijados en el rango de intensidad 0.55-0.65 en base a simple observación.

## 5 Desarrollo de métodos de clasificación

En esta sección se expone una forma de estimar el peso de una garra a partir de una máscara de segmentación con el fin de comparar con el peso real de la garra y saber si esta estimación es útil o no.

El otro factor que afecta en el calibre son los defectos, se proponen 2 formas de hacer esta clasificación. En la sección 5.2 se muestra el primer enfoque que apunta a la composición de un vector de características redundante para ser utilizado como entrada de una red neuronal, se compararán distintas maneras de formar dicho vector y distintas arquitecturas para la red.

En la sección 5.3 se muestra el segundo enfoque que apunta a una cuantificación de defectos para luego definir los calibres a partir del área que ocupa cada uno de ellos en la garra. Se propone hacer ésta clasificación de dos formas distintas, una con clasificadores multiclase y otra con una serie de umbrales que simulan el criterio de un operario en cuanto a la cantidad de defectos que está permitido en cada calibre.

### 5.1 Clasificación por peso

La forma más simple y precisa de obtener el peso de una garra sin sacarla de la línea de proceso es utilizando un sistema de cinta con báscula que pese cada garra cuando pasa por ahí [13].

Se quiere estudiar la factibilidad de reemplazar el sistema de báscula con una estimación de peso a partir de imágenes, este método consiste en usar una máscara de segmentación y sumar los píxeles que indiquen presencia de garra. Para esto se calcula la correlación existente entre muestras que fueron pesadas una por una, y la cantidad de píxeles obtenidos a partir de la máscara de segmentación. Una alta correlación (mayor a 95%) haría de este método una buena forma de clasificar garras según su peso, pero un valor de correlación menor obliga a seguir utilizando el sistema de cinta con báscula.

Cada muestra está formada por dos imágenes y no basta con calcular la correlación entre el peso y los píxeles de una de las dos máscaras, sino que se debe hacer con las dos imágenes por separado y la suma de píxeles en las máscaras.

## 5.2 Clasificación por defectos. Histogramas de color

Los histogramas son una forma de representar en un arreglo, la información contenida en una fuente de información mucho mayor, es por esto que pueden ser una buena forma de representar los colores contenidos en cada garra para ser usados como entrada de un algoritmo de clasificación para definir el calibre de una garra.

Se crean 4 histogramas que contienen las componentes roja, verde, azul y gris de la garra ya separada del fondo. A cada uno de estos arreglos se les extrae información estadística útil y se crea un vector de características. A continuación, se detallan las estadísticas incluidas

- Moda e intensidad asociada a la moda del histograma
- Ancho del histograma el 75%, 50%, 25%, 12.5% y 6.25% de la moda
- Intensidad media
- Varianza de la intensidad y de la intensidad normalizada
- Entropía de Shannon

Para aumentar los datos entregados al clasificador se incluye en las pruebas los histogramas como variables de entrada, pero agregar 4 arreglos con 256 valores cada uno, para cada imagen, es mucha información y la mayoría innecesaria. Para reducir la cantidad de datos entregados al clasificador perdiendo el mínimo de información, se cuantizaron los histogramas reduciendo el largo de cada arreglo de 256 bins a 128, 64, 32 o 16, lo que significan 5 formas distintas de construir el vector de características.

Otro método de reducción en la dimensionalidad del problema es PCA, el cual permite resumir la información de cientos de variables a unas pocas decenas. Éste método se aplicó a las 5 formas de construir el vector de características por separado, en cada una de ellas se utilizaron las componentes que concentran el 95% de la información, prescindiendo del resto de los datos.

Se cuenta con 521 muestras de distintos calibres, las cuales son separadas en conjuntos de entrenamiento y prueba en razón 7:3, teniendo cuidado de mantener las proporciones de cada calibre en ambos conjuntos. La cantidad de muestras de cada calibre en los conjuntos se detalla en la Tabla 1.

Calibre	TOTAL	ENTRENAMIENTO	PRUEBA
A	111	78	33
B	109	76	33
C	140	98	42
CHOICE	80	56	24
REPROCESO	81	57	24
TOTAL	521	365	156

Tabla 1: Cantidad de muestras en los conjuntos de entrenamiento y prueba según calibre.

En cuanto a la arquitectura de la red neuronal a utilizar, esta puede tener o no una capa oculta, y tanto la capa oculta como la de salida deben tener una función de activación, la cual será tangente-sigmoidea por ser la que entrega mejores resultados. El algoritmo *Backpropagation* de entrenamiento será el del gradiente descendente de segundo orden para todas las pruebas. Detalles de distintas arquitecturas y sus desempeños en la clasificación se muestran en la sección 6.3.

### 5.3 Clasificación por defectos. Cuantificación y clasificación

La idea de este enfoque es cuantificar los defectos según su naturaleza de color, separando los pixeles de la garra en cutícula, callo, hematoma y garra sana, para lo cual se aplicarán distintos métodos y se estudiarán los resultados.

La cuantificación consiste en un conteo de pixeles de una máscara de segmentación (una máscara de segmentación para cada defecto), número que servirá para representar el área afectada por un defecto. Cada muestra está compuesta por 2 imágenes mostrando la parte superior e inferior de la garra, por lo que se tienen 6 variables cuantificando defectos en cada muestra, sin embargo, cada par de variables representan un mismo defecto ubicado en lugares distintos de la garra, permitiendo sumarlos sin perder información y así utilizar solo 3 variables representando la cantidad total de cada defecto presente en la muestra.

Con variables que representen la presencia y tamaño de defectos en cada garra, resta definir el calibre. Para lo anterior se proponen tres métodos, dos de ellos consisten en aplicar inteligencia computacional y el tercero se basa en aplicar la regla que define los calibres, correlacionando el tamaño de los defectos con la cantidad de pixeles que ocupan en la imagen.

#### 5.3.1 Cuantificación de defectos

##### 5.3.1.1 Diferencia con modelo de fondo

Al igual que en la sección 4.1 se calcula una imagen modelo de fondo y la diferencia entre una muestra con dicho modelo. Al observar la componente azul de la imagen de diferencias, se ve que la intensidad del callo resalta por

sobre los hematomas, cutícula y garra sana. Se podría establecer un umbral para detectar los pixeles de callo, de tal manera que los pixeles cuya intensidad supere el umbral, serán considerados como pixeles de callo.

Teniendo una buena máscara de segmentación para la garra completa, las sombras que rodean a la garra no son un problema al momento de segmentar callos ni otros defectos. Por medio de una serie de pruebas y observaciones se llega a un umbral tentativo en una intensidad de 85 sobre la componente azul, lo que haría de este método una forma efectiva de identificar los pixeles de callo (ver Ilustración 50b).



Ilustración 50: (a) Garra II sin fondo y la segmentación de su callo (b)

No todos los cayos son detectados con tanta claridad como el que se muestra en la Ilustración 50, ya que estos pueden no ser tan oscuros, como el que se muestra en la Ilustración 51a. Mantener el umbral en 85 induce a una mala detección (ver Ilustración 51b), que puede ser corregida bajando el umbral, sin embargo, un umbral muy bajo hace que se detecten callos donde no los hay (ver Ilustración 51c cambio de umbral a intensidad de 55). Más adelante en la sección 6.4.1.1 se estudiará con mayor profundidad la sensibilidad de este umbral.

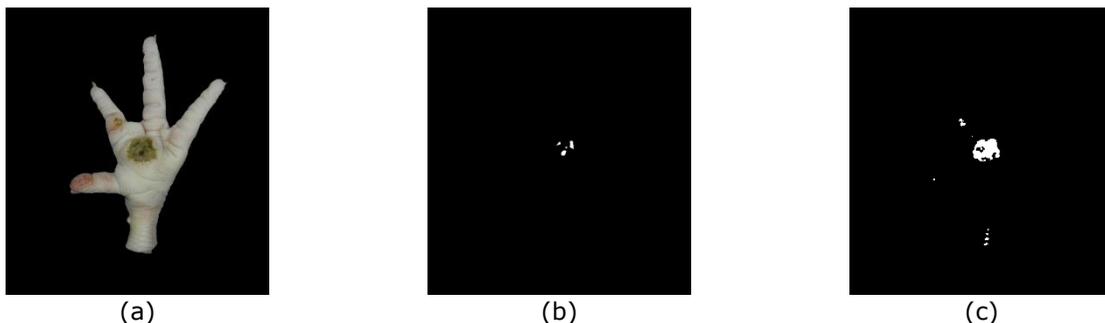


Ilustración 51: (a) Garra caso IV sin fondo. (b) Mala segmentación de callo en garra IV. (c) Cambio de umbral para detectar callo en garra IV.

### 5.3.1.2 Umbrales RGB

Estudiando la naturaleza de los colores de los defectos en las garras, se puede ver que los tonos amarillos son cutículas, cafés y negros son callos y tonos rojos son hematomas, por lo que bastaría con separar estos colores

para detectar cada defecto. En la Ilustración 52 se muestra el caso de una garra que presenta los 3 tipos de defectos a detectar.



Ilustración 52: Garra caso V sin fondo.

a. Cutícula

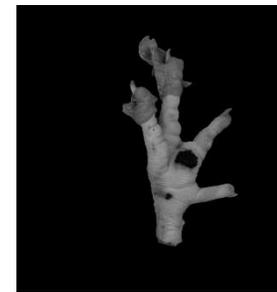
Los colores amarillos son aquellos en que la intensidad de la componente roja y verde es alta, pero que la intensidad de la componente azul es baja.



(a)



(b)



(c)

Ilustración 53: Componente roja (a), verde (b) y azul (c) de imagen del caso V sin fondo.

No es posible encontrar un umbral para la componente azul (Ilustración 53c) que separe los tonos amarillos, y umbrales en las componentes roja (Ilustración 53a) y verde (Ilustración 53b) no son de ayuda, ya que la cutícula alcanza la misma intensidad que el resto de las garras en esta componente. En la Ilustración 54 se ve el resultado de aplicar un umbral de 115 en la componente azul, de tal manera de que los valores bajo el umbral serán considerados como detecciones y los pixeles que superen esa intensidad serán no detecciones. Un umbral más bajo elimina zonas de cutícula y uno más alto agrega más zonas de garra innecesarias.

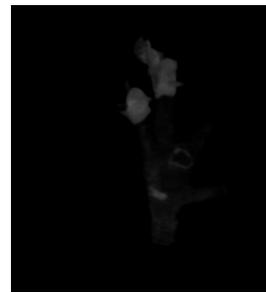


Ilustración 54: Umbral de 115 aplicado a la componente azul de la garra V.

Se puede mejorar este resultado restándole la componente azul a la componente verde o roja para bajar la intensidad de la garra sin bajar demasiado la intensidad de la cutícula.



(a)



(b)

Ilustración 55: (a) Componente roja menos azul. (b) Componente verde menos azul.

Buscando umbrales se llega a que en el ejemplo de la Ilustración 55a, un umbral mayor a 20 deja de detectar correctamente la cutícula mientras se sigue detectando partes de la garra como parte de la detección. En la imagen de la Ilustración 55b, se llega un umbral crítico de 25, pero en este caso, los píxeles de garra detectada como cutícula son notablemente menores. Debido al buen resultado obtenido al aplicar un umbral sobre la diferencia entre las componentes verde y azul, se analizará con mayor profundidad la sensibilidad de este umbral, los resultados se muestran en la sección 6.4.1.2a.



(a)



(b)

Ilustración 56: (a) Umbral de 20 aplicado a la Ilustración 55a. (b) Umbral de 25 aplicado a la Ilustración 55b.

b. Callo

La mayor parte de los callos tienen una intensidad muy baja en las 3 componentes del espacio RGB, lo que haría pensar que es fácil de segmentar bajo un umbral en cualquiera de las componentes, sin embargo, existen callos más claros como el del caso IV (Ilustración 51a) que hacen necesario un umbral mayor para poder detectar. Aquí es donde el método pierde efectividad, ya que un umbral mayor hace que hematomas y sombras del borde de la garra entren en el rango de detección produciendo errores.

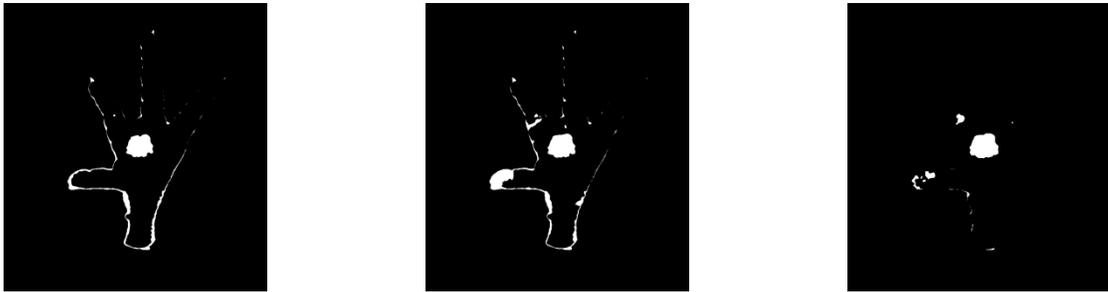


Ilustración 57: Umbral aplicado a la componente roja (a), verde (b) y azul (c) de la garra IV.

En la Ilustración 57 se aplicaron los mínimos umbrales necesarios para detectar todo el callo. En las 2 primeras componentes se generan gran cantidad de pixeles de error detectando callo donde no lo hay, por lo que se pueden descartar estas 2 opciones del método. Sin embargo, un umbral de 75 en la componente azul logra una buena detección de los callos sin los errores incluidos en los otros ejemplos.

En la Ilustración 58 se puede ver cómo el mismo umbral aplicado a la componente azul de la garra del caso II logra una buena detección, pero agrega algunos pixeles de error. Se analizará con mayor profundidad la sensibilidad de este umbral, los resultados se muestran en la sección 6.4.1.2b.



Ilustración 58: Detección de callo en caso II

c. Hematomas

Este método también podría utilizarse para cuantificar los hematomas presentes en la garra, ya que los tonos rojos tienen una alta intensidad en su componente, pero baja intensidad en las componentes verde y azul. Los

colores claros de la garra tienen intensidad alta en las 3 componentes, por lo que el umbral se debe fijar en las componentes en que los hematomas tienen baja intensidad.

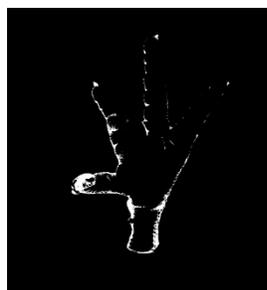
La solución no es tan simple, ya que los colores de la cutícula también tienen baja intensidad en la componente verde y azul, alcanzando la misma intensidad que los hematomas en éstas componentes. Para poder encontrar un umbral para segmentar los hematomas, se debe separar primero la cutícula y el callo (ver Ilustración 59).



Ilustración 59: Garra V sin fondo, cutícula ni callo.

Incluso asumiendo la segmentación propuesta anteriormente para la cutícula y el callo, los errores en dichas segmentaciones afectan directamente la detección de hematoma, ya que la intensidad del hematoma es parecida a la de los errores de segmentación arrastrados hasta este punto.

En la Ilustración 60 se muestra el mejor resultado, obtenido aplicando un umbral de 115 en la componente verde e invirtiendo la intensidad para mostrar los hematomas en blanco y no en negro, en éste resultado se pueden ver los errores arrastrados por la mala segmentación de cutícula en la garra V y una falsa detección de hematoma en los bordes sombreados de ambas garras.



(a)



(b)

Ilustración 60: Segmentación de hematomas en garra IV (a) y V (b).

Debido al mal resultado obtenido con este método, se descartará como una opción para segmentar hematomas.

### 5.3.1.3 Algoritmos de clasificación

Una buena opción para distinguir entre los distintos defectos según su color es entrenar un clasificador que a partir de los valores RGB de un pixel, pueda definir si éste corresponde a cutícula, callo, hematoma o garra sana. Luego, al pasar el clasificador por todos los pixeles de la garra, se puede definir en qué lugares se encuentra cada defecto. Desde este enfoque de clasificación de pixeles, una muestra no es un par de imágenes de una misma garra, sino que una muestra es un pixel dentro de una imagen.

Se toman recortes de defectos y se agrupan en imágenes que servirán de entrenamiento para el clasificador. Las imágenes con recortes utilizadas se muestran en el Anexo A.

#### a. Clasificador multiclase

Clasificadores multiclase como árboles de decisión o redes neuronales podrían ser capaces de separar los defectos simultáneamente considerándolos a cada uno como una clase, incluyendo una clase para garra sana. Para obtener resultados de esta detección de defectos, se debe ingresar un pixel de la garra como si fuera una muestra (intensidades RGB como vector de características) y el algoritmo entregará la clase estimada de dicho pixel.

Una red neuronal puede tener o no una capa oculta en su arquitectura y la cantidad de neuronas en ella no está definida a priori, por lo cual se hacen pruebas con y sin esta capa. Para los casos en que sí se utilice una capa oculta, heurísticas recomiendan seguir una pirámide geométrica para definir la cantidad de neuronas en esta capa, para el caso en estudio esto significa usar entre 3 y 4 neuronas en la capa oculta, por lo que se harán pruebas considerando ambos casos. Otro parámetro importante a considerar en una red neuronal es la función de activación de su capa oculta y de salida. Las funciones recomendadas son: escalón, tangente-sigmoidea, logaritmo-sigmoidea y lineal. Todas las redes que se compararán, serán entrenadas utilizando el algoritmo *Backpropagation* de gradiente descendente de segundo orden *Levenberg-Marquardt* debido a su superior velocidad.

Para efectos de la cuantización, solo es necesario que, tras analizar todos los pixeles de la garra, el método de segmentación determine cuantos pixeles pertenecen a cada clase. Sin embargo, un resultado de este tipo no permite calcular una tasa de error tipo 1 o 2 para comparar con otros métodos, por lo que es necesario reconstruir la imagen mostrando en que pixeles son detectados los defectos.

Para representar las distintas clases en una sola imagen, se representará cada una de ellas con un color distinto. Las clases son: garra sana, cutícula, callo y hematoma, y serán representadas con los colores

verde, amarillo, azul y rojo respectivamente. En la Ilustración 61 se pueden ver 3 casos de garras con defectos y en la Ilustración 62 se muestra la reconstrucción en colores de los resultados de clasificación de los mismos casos.

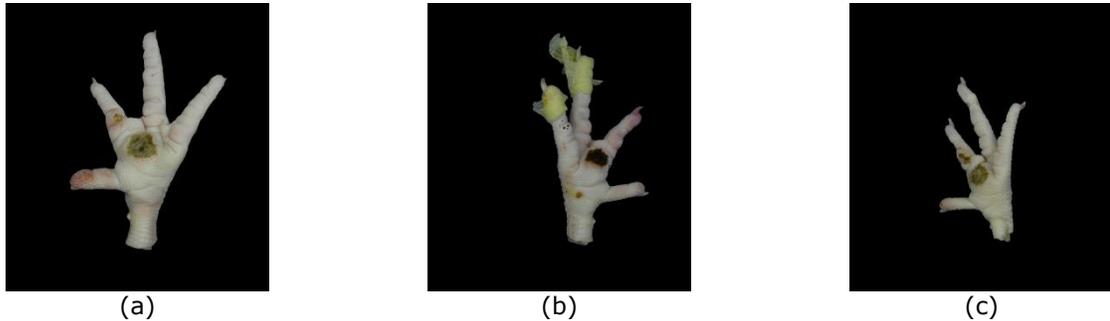


Ilustración 61: Garras casos IV (a), V (b) y VI (c).

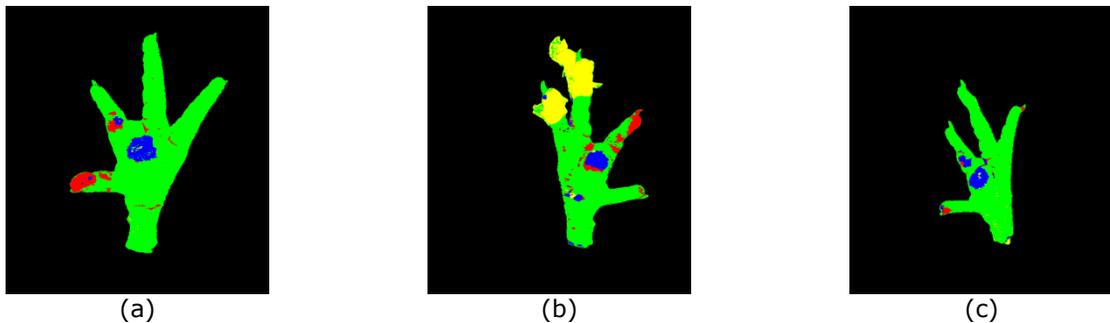


Ilustración 62: Reconstrucción en colores de resultados de clasificación garras casos IV (a), V (b) y VI (c).

En la sección 6.4.1.3 se comparará el desempeño en la detección de defectos de distintas arquitecturas de redes neuronales, árboles de decisión y KNN con distintos valores de K.

#### b. Cascada de clasificadores binarios

Se prueba una forma distinta de configurar clasificadores, de tal manera que sea una cascada de algoritmos configurados de manera especial para que solo clasifiquen de forma binaria en cada nivel (ver Ilustración 63). Se descarta el uso de SVM debido a que, en las pruebas realizadas, el algoritmo tardó más de 2 minutos en entregar resultados para una sola imagen, independiente del kernel utilizado, haciendo imposible su uso en una aplicación industrial. Si bien este no es el tema principal de este trabajo, debe ser considerado si la aplicación de éste es una línea que procesa cientos de garras por minuto.

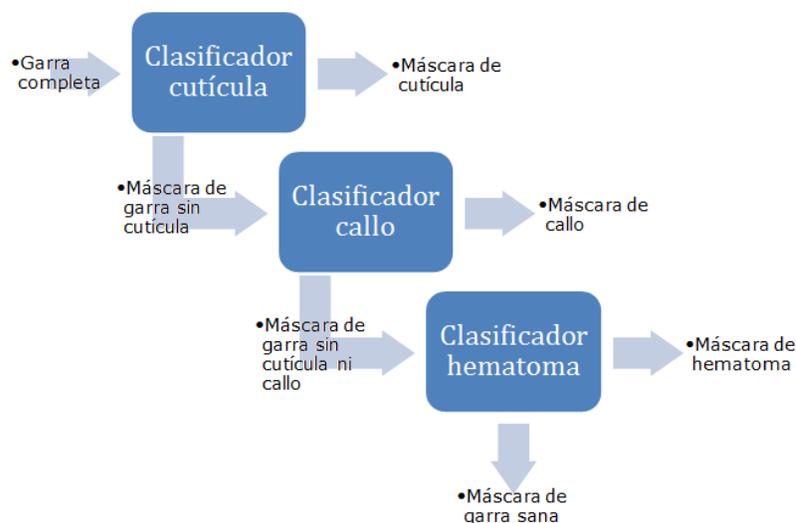


Ilustración 63: Diagrama de cascada de clasificadores binarios.

Los píxeles mal clasificados en cada nivel, se heredan como error a los siguientes niveles, por lo que la configuración propuesta puede cambiar con tal de que los niveles se ordenen según su desempeño.

Al igual que con el clasificador multiclase, en este caso se toman los píxeles como muestras independientes y el algoritmo entrega la clase a la que corresponde dicho píxel, dependiendo del nivel de la cascada en que se encuentre.

Se entrenarán redes neuronales, árboles de decisión y KNN utilizando distintos parámetros. Siguiendo con lo expuesto en 5.3.1.3a, se probarán redes neuronales sin capa oculta y con capa oculta de 3 o 4 neuronas, además, se probarán funciones de activación escalón, tangente-sigmoidea, logaritmo-sigmoidea y lineal, todo lo anterior utilizando un entrenamiento con algoritmo *Backpropagation* de gradiente descendente de segundo orden *Levenberg-Marquardt* debido a su superior velocidad.

A continuación, se detalla el procedimiento a seguir en cada etapa de la cascada. A modo de ejemplo se utilizan árboles binarios, pero se analizan más métodos en la sección 6.4.1.4.

- Cutícula

En el nivel del clasificador de cutícula, la entrada son todos los píxeles de la garra ingresados de manera independiente. La salida es un conjunto de etiquetas binarias que marcan que píxeles son parte de una cutícula y que píxeles no, de esta manera se pueden construir máscaras de segmentación para el defecto analizado y el resto de la garra.

Se tomarán las garras correspondientes a los casos II y V para hacer una comparación de la segmentación, esto se debe a que la primera tiene una cutícula considerada pequeña que no necesariamente va a reproceso y la segunda tiene una suficientemente grande para serlo.



Ilustración 64: Mala segmentación de cutícula en garras casos II (a) y V (b).

En la Ilustración 64 se muestra el resultado de la segmentación con el algoritmo de árboles binarios, donde se puede observar gran cantidad de píxeles no detectados debido a la similitud de colores entre algunas partes de la cutícula y la garra sana, también se ven falsas detecciones en lugares del contorno de la garra, debido a la distorsión de colores por sombras. En la sección 6.4.1.4a se hace una comparación de los árboles binarios mostrados en los ejemplos con otros métodos de clasificación, para saber cual tiene mejor tasa de aciertos versus menor tasa de falsas detecciones.

Se puede mejorar este resultado utilizando operadores morfológicos de erosión para eliminar las falsas detecciones pequeñas y dilatación para rellenar los píxeles no detectados. El resultado de aplicar lo anterior en los casos en cuestión de puede ver en la Ilustración 65.



Ilustración 65: Segmentación de cutículas en garras casos II (a) y V (b.)

La negación de la máscara anterior, intersectada con la máscara de la garra completa entrega la entrada para la siguiente etapa del clasificador (ver Ilustración 66).



Ilustración 66: Máscara de garras casos II (a) y V (b) sin cutícula.

- Callo

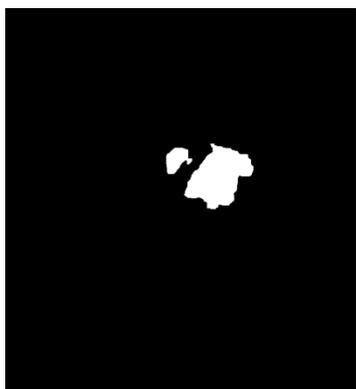
Las máscaras de la garra sin cutícula que se muestran en la Ilustración 66, se utilizan como entrada al segundo nivel de la cascada, nivel donde el clasificador identificará los píxeles de callo permitiendo crear una máscara para el callo y una para la garra sin callo.

Siguiendo con los ejemplos de la etapa anterior y con un procedimiento análogo, se llega al mismo problema de tener pequeños puntos de no detección en el callo, y pequeños puntos de falsa detección fuera del callo (ver Ilustración 67).

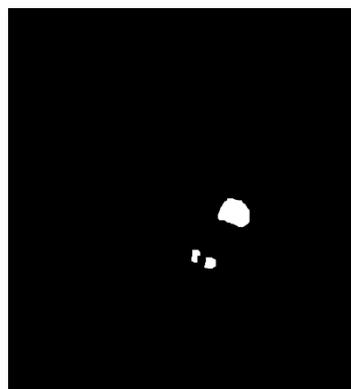


Ilustración 67: Mala segmentación del callo en garras casos II (a) y V (b).

Siguiendo el mismo análisis que para las cutículas, se aplican los operadores de erosión y dilatación para mejorar los resultados, obteniendo las máscaras que se muestran en la Ilustración 68.



(a)



(b)

Ilustración 68: Segmentación mejorada de callo en garras casos II (a) y V (b).

La negación de la máscara anterior, intersectada con la máscara de la garra completa entrega la entrada para la siguiente etapa del clasificador.

- Hematomas

Con la máscara de la garra sin cutícula ni callo (ver Ilustración 69), se entra al tercer nivel de la cascada donde el algoritmo de clasificación identifica los pixeles de hematomas y genera una máscara con todos ellos, permitiendo generar una máscara para los pixeles restantes que corresponden a la garra sana.



(a)



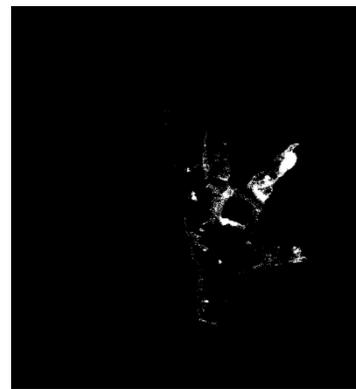
(b)

Ilustración 69: Máscara de garras casos II (a) y V (b) sin cutícula ni callo.

Al pasar por la tercera etapa de clasificación, se obtiene una máscara tentativa para los hematomas, en la Ilustración 70 se muestran dichas máscaras para los casos en estudio.



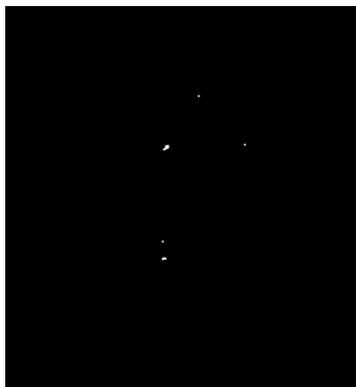
(a)



(b)

Ilustración 70: Segmentación de hematomas en garras casos II (a) y V (b).

El resultado anterior hace parecer que la detección no es buena ya que se detectan muchos puntos aislados donde no hay hematoma, esto al igual que los casos anteriores, se puede solucionar con operadores de erosión y dilatación, pero en este caso debe ser un operador más pequeño. El resultado se muestra en la Ilustración 71.



(a)



(b)

Ilustración 71: Segmentación mejorada de hematomas en garras casos II (a) y V (b).

### 5.3.2 Determinación de calibre con clasificador multiclase

Se quiere estudiar la viabilidad de utilizar inteligencia computacional en la determinación de calibres a partir de las 3 variables obtenidas con la cuantificación de defectos, para esto se debe establecer un rango para las variables. Las variables no tienen relación con el tamaño total de la garra, pero si el tamaño de la imagen, por lo que pasarán de ser cantidad de píxeles, a porcentaje de píxeles en el par de imágenes de cada muestra.

Con el calibre real de cada muestra, sus 3 variables de cuantificación y sus rangos bien definidos, se pueden hacer pruebas utilizando un conjunto de entrenamiento y prueba. Se cuenta con 521 muestras de distintos calibres, las cuales son separadas en conjuntos de entrenamiento y prueba en razón 7:3, teniendo cuidado de mantener las proporciones de cada calibre

en ambos conjuntos. La cantidad de muestras de cada calibre en los conjuntos se detalla en la Tabla 2.

Calibre	TOTAL	ENTRENAMIENTO	PRUEBA
A	111	78	33
B	109	76	33
C	140	98	42
CHOICE	80	56	24
REPROCESO	81	57	24
TOTAL	521	365	156

Tabla 2: Cantidad de muestras en los conjuntos de entrenamiento y prueba según calibre.

En caso de usar una red neuronal, es importante probar distintas arquitecturas para ver cuál entrega un mejor resultado de clasificación. Se probarán redes sin capa oculta y otras con capa oculta con 3, 4 o 5 neuronas. En todas las arquitecturas se probarán las funciones de activación escalón, tangente-sigmoidea, logaritmo-sigmoidea y lineal. Todo lo anterior utilizando un entrenamiento con algoritmo *Backpropagation* de gradiente descendente de segundo orden *Levenberg-Marquardt* debido a su superior velocidad. Se compararán los resultados con los obtenidos utilizando árboles de decisión y KNN con K entre 1 y 4.

Finalmente, un conjunto de muestras de prueba sirve para comparar este método con otras formas de definir el calibre.

### 5.3.3 Determinación de calibres con umbrales

Haciendo una relación entre los tamaños reales de los defectos con la cantidad de píxeles que ocupan en una imagen, se puede implementar un set de reglas que defina los calibres en cuanto a la cantidad de cada defecto. Estas reglas están basadas en las medidas que definen un calibre, medidas que se detallan en la sección 2.1.2, y son cantidades críticas o umbrales sobre la cantidad de píxeles de defecto, en las que una muestra pasa de un calibre a otro.

Se utilizará un 70% de las muestras para hacer observaciones y análisis con tal de probar el método con muestras que no influyen en su creación, mismas muestras de prueba utilizadas para otros métodos, lo que los hace comparables.

Para hacer la relación entre los tamaños reales de los defectos y la cantidad de píxeles, es necesario tener un buen método de cuantificación. Para construir la tabla de umbrales que se describirá a continuación, se utilizaron los mejores métodos conseguidos de la sección 5.3.1, cuya comparación y resultados se muestran en la sección 6.4.1.

Al terminar esta sección se adjunta el resultado final de la construcción de este método, correspondiente a la Tabla 3 y muestra los rangos de cada defecto que un calibre permite tener.

Tras observar y estudiar con atención las muestras considerando el calibre, las imágenes y la cuantificación de defectos, se observa cierta parcialidad en la asignación de calibres hecha por los operarios. Lo anterior se notó con casos en que las imágenes y la cuantificación mostraban garras parecidas, pero estaban etiquetadas con calibres distintos. Estos casos fueron considerados como errores en el conjunto de entrenamiento y se aplicó la regla real para ver a favor de que muestra se desplazaba el umbral.

En la Ilustración 72 se pueden ver 2 muestras que presentan cantidades parecidas de callo y hematoma, tanto según criterio visual como en la cuantización, ya que ambas presentan cerca de 9000 pixeles de callo y 9000 de hematomas. Sin embargo, la muestra de la izquierda fue etiquetada como una garra calibre C y la de la derecha como CHOICE. La medida real del callo no supera los 3 cm, por lo que la cifras de cuantización servirán para determinar los umbrales del calibre C, pero la etiqueta de calibre no será cambiada en este caso ni en ningún otro que se repita esta parcialidad en la clasificación de los operarios.

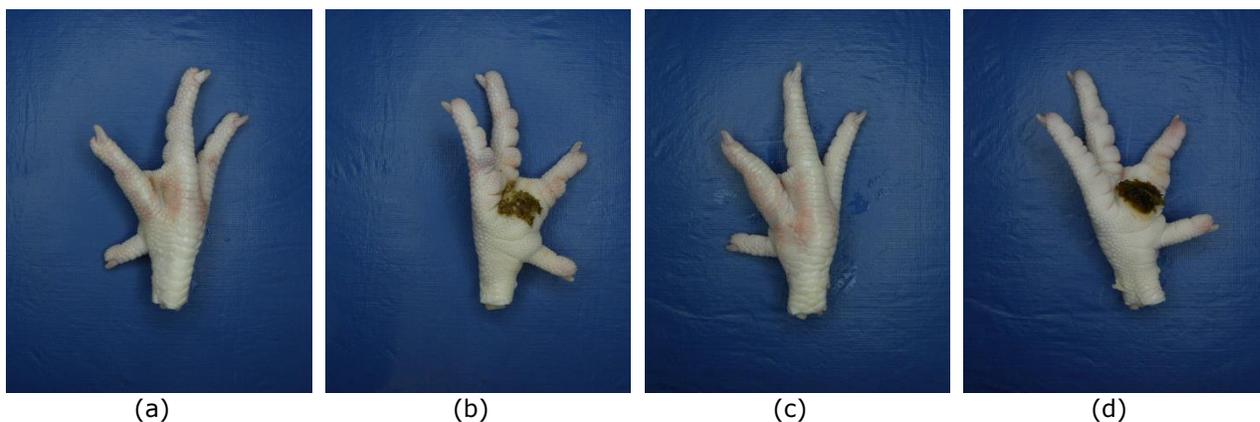


Ilustración 72: Comparación entre garras posiblemente mal clasificadas por operador. (a) con (b) muestran una garra y (c) con (d) muestran la otra.

#### 5.3.3.1 Cutículas

Se pidió especialmente a un operador de la planta faenadora que seleccionara garras con cutículas pequeñas y definiera cuales debían ser enviadas a reproceso y cuales tenían una cutícula suficientemente pequeña como para ser aceptadas en un calibre.



(a)



(b)

Ilustración 73: (a) Garra con cutícula aceptable (a). (b) Garra con cutícula de reproceso.

Se pudo observar que, al aplicar el método detector de cutículas, éste en general no superaba los 3000 píxeles para garras con cantidad de cutículas aceptables (ver Ilustración 73a), umbral que era superado con facilidad en garras que debían ser enviadas a reproceso (ver Ilustración 73b). El operario afirmó que solo en calibres A, B y C se debe ser cuidadoso con la cutícula, ya que el calibre CHOICE puede incluir cutículas si su callo o hematomas la definen en ese calibre.

#### 5.3.3.2 Callos

Por definición, las garras A no deben tener callo, las garras B pueden tener callos de hasta 1.5 cm y las garras C aceptan callos de hasta 3 cm. A estas medidas se les suma las garras de reproceso ya que, si se les detectan callos de más de 3 cm, son directamente clasificadas como CHOICE.

Se creería que una garra A no debe tener ningún píxel de callo después de pasar por el método de detección de los mismos, sin embargo, los operadores especializados pasan por alto los callos que son demasiado pequeños como el que se muestra en la Ilustración 74, al cual se le detectan 550 píxeles de callo, lo que lleva a fijar el umbral de callo para garras A en 600.



Ilustración 74: Acercamiento de pequeño callo en garra calibre A.

Utilizando los comentarios anotados para cada muestra, se buscan garras con callos de 1.5 cm (ver Ilustración 75a) y muestras con callos más grandes (ver Ilustración 75b), para luego aplicarles el detector de callo. De los resultados se observa que las muestras con callos de hasta 1.5 cm no entregan un resultado mayor a 6600 pixeles por lo que el umbral se fijará en ese valor.



(a)



(b)

Ilustración 75: Garras con callo de 1.5 cm (a) y 2 cm (b).

Se hace el mismo procedimiento con los callos de 3 cm (ver Ilustración 76) y se define el umbral en 14000 pixeles.

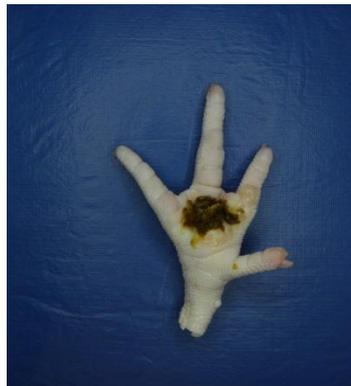


Ilustración 76: Garra con callo de 3 cm.

### 5.3.3.3 Hematomas

Considerando las garras a las que se le detectaron 0 pixeles de callo se encuentran garras con distintas cantidades de pixeles de hematomas y de todos los calibres. Se observa un claro cambio en los 3600 pixeles de hematoma, donde las garras pasan de ser calibre A a B. Lo mismo ocurre a los 9000 pixeles, donde empiezan a clasificarse como C, terminando en los 12000 pixeles, donde el hematoma es tan grande que pasa a ser CHOICE.

Para una cantidad fija de hematoma, el calibre puede cambiar según las cantidades de callo, tal como la regla lo define. El criterio anterior es más estricto para separar entre garras A, B y C, ya que CHOICE se considera

como material descartable. Lo anterior se ve reflejado al observar garras con mayor cantidad de callo detectado. Entre 0 y 600 pixeles de callo los umbrales encontrados para hematomas son de 1600 para separar entre A y B, 4000 entre B y C, terminando con 12000 como límite para que una muestra pase a ser CHOICE.

Siguiendo el mismo tipo de observaciones se crean distintos rangos de callos y hematomas para las garras con callos de entre 0 y 1.5 cm, asignándoles calibres B, C y CHOICE. Terminando con rangos de callos sobre 1.5 cm para separar C y CHOICE. En la Tabla 3 se muestran todos los rangos establecidos, se debe tener en consideración que las garras CHOICE serán todas aquellas que no entren en ningún rango.

CUTÍCULAS [pixeles]	CALLOS [pixeles]	HEMATOMAS [pixeles]	CALIBRE
0	0	0	A
3000	0	3600	
		3601	B
		9000	
		9001	C
		12000	
	1	0	A
	600	1600	
		1601	B
		4000	
		4001	C
		12000	
	601	0	B
	2000	3000	
		3001	C
		12000	
	2001	0	B
	4200	2500	
		2501	C
		12000	
	4201	0	B
	5500	1950	
		1951	C
		12000	
	5501	0	B
	6600	1750	
		1751	C
		12000	
	6601	0	C
	10000	12000	
	10001	0	C
	14000	9000	
3001	0	0	REPROCESO
	15000	12000	

Tabla 3: Conjunto de umbrales para definir calibres según cuantificación de defectos medidos en pixeles.

## 6 Análisis de resultados

### 6.1 Análisis máscara de segmentación

Para poder hacer una comparación más numérica que visual entre los métodos de segmentación de la garra, primero se debe contar con una segmentación ideal. Se tomaron 60 imágenes con garras y utilizando Photoshop se les crearon máscaras de segmentación, las que serán consideradas como caso ideal con el cual comparar los distintos métodos propuestos.

Se considerará el óptimo de una curva ROC, en punto de menor distancia a la coordenada (0,1) del mismo espacio. Sin embargo, en algunos casos se preferirá abandonar esta definición de óptimo para darle más importancia a los verdaderos positivos por sobre los falsos positivos.

#### 6.1.1 Método con modelo de fondo

Se calcula la tasa de aciertos y tasa de falsos positivos para distintos umbrales, fijando un umbral se puede generar una curva ROC con los posibles valores del otro umbral.

Fijando distintos umbrales para la componente roja y haciendo un barrido por umbrales entre 60 y 100 para la componente azul, se construyen distintas curvas ROC; algunas de ella se muestran en la Ilustración 77a. De la misma forma, fijando distintos umbrales para la componente azul y haciendo un barrido por umbrales entre 20 y 60 de la componente roja, se construyen distintas curvas ROC, algunas de ellas se muestran en la Ilustración 77b.

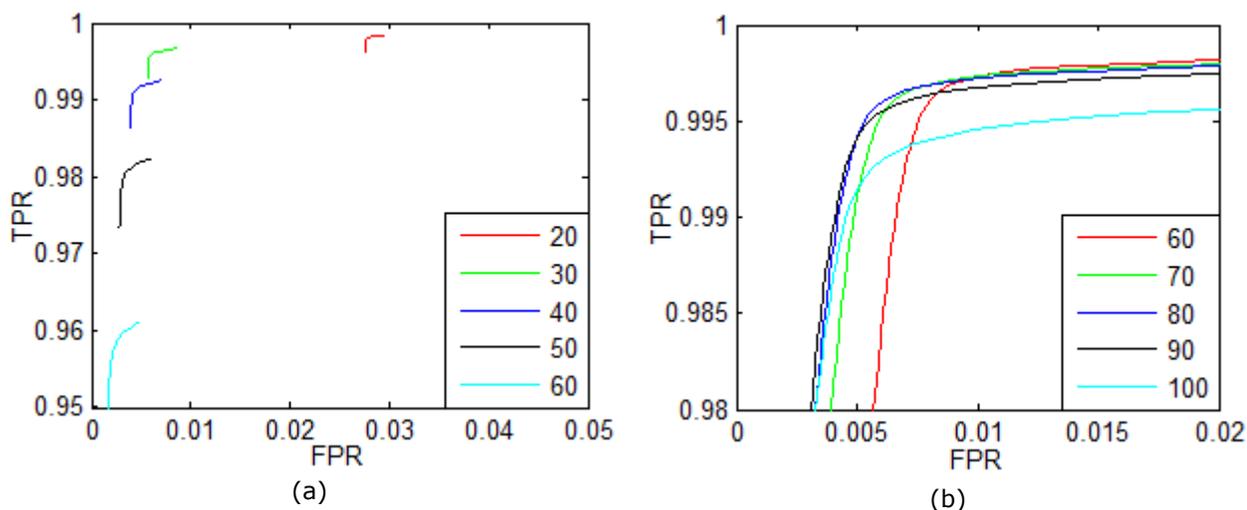


Ilustración 77: Curvas ROC umbrales componentes roja (a) y azul (b).

La menor distancia a la coordenada (0,1) se logra fijando los umbrales en una intensidad de 31 para la componente roja y 82 para la azul, entregando tasas promedio de aciertos y error tipo 1 en 99.56% y 0.57% respectivamente.

### 6.1.2 Método con componentes RGB

Para evaluar el desempeño de este método, se sigue el mismo procedimiento que el realizado con el método del modelo de fondo, ya que la construcción de curvas ROC permite la comparación entre ambos.

En este caso, al fijar distintos umbrales entre 55 y 95 para la componente roja, se hace un barrido entre umbrales de la componente azul y entre 15 y 55 para el caso análogo, se da lugar al conjunto de curvas, algunas de ellas se muestran en la Ilustración 78.

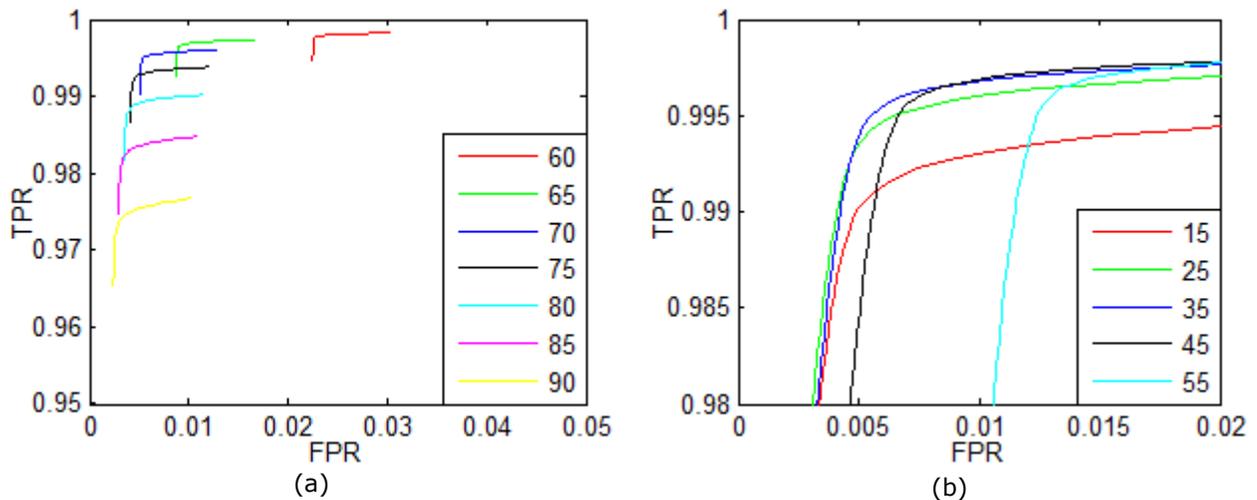


Ilustración 78: Curvas ROC umbrales componentes roja (a) y azul (b).

Se pueden establecer con mayor precisión los umbrales en la intensidad de 70 para la componente roja y en 34 para la componente azul debido a su menor distancia a la coordenada (0,1), entregando tasas promedio de aciertos y error tipo 1 en 99.5% y 0.55% respectivamente.

### 6.1.3 Método con espacio de color YCbCr

Evaluar este método es más sencillo, ya que solo se debe evaluar 1 umbral en la componente Cb y el óptimo de esa curva es el que se compara con los otros métodos.

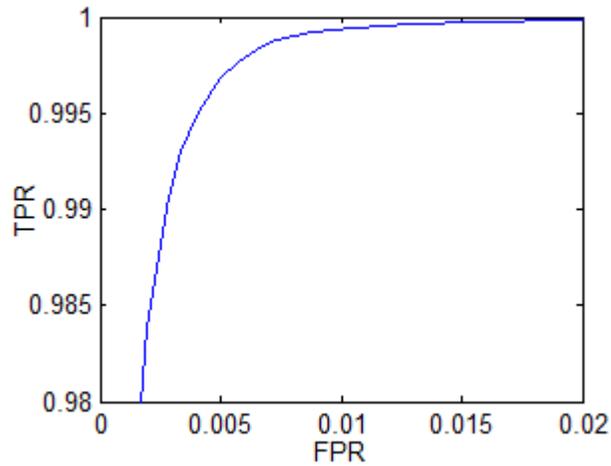


Ilustración 79: Curva ROC umbral en componente Cb.

En este caso, el óptimo (menor distancia a la coordenada (0,1)) se logra fijando un umbral en la intensidad de 134 sobre la componente Cb, dando como resultado una tasa promedio de aciertos y error tipo 1 de 99.68% y 0.5% respectivamente.

#### 6.1.4 Método con espacio de color HSV

Para este caso vuelve a ser necesario evaluar 2 umbrales, pero esta vez es sobre la misma componente de matiz. Se prueba fijando el umbral inferior entre 0.545 y 0.585, haciendo un barrido por varios umbrales superiores dando lugar a un conjunto de curvas, parte de las cuales se muestran en la Ilustración 80a. Se hace un proceso análogo para el umbral superior variándolo entre 0.625 y 0.665, algunas curvas se pueden ver en la Ilustración 80b.

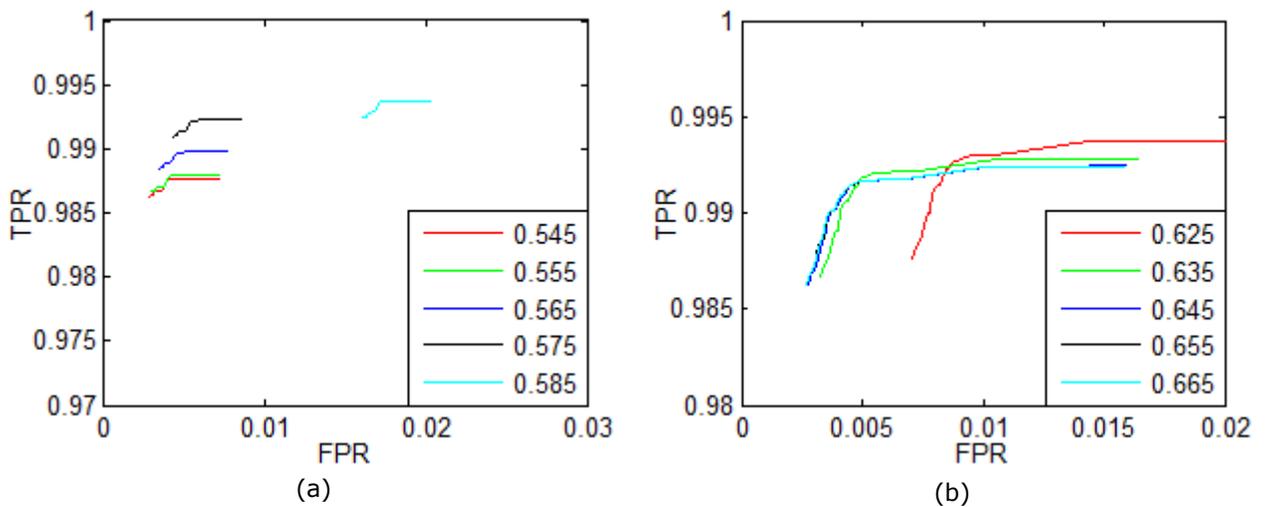


Ilustración 80: Curvas ROC umbral inferior (a) en superior (b) en componente de matiz.

Lo anterior permite ubicar un óptimo para la intensidad de fondo entre 0.552 y 0.656, lo que significa una tasa promedio de aciertos y error tipo 1 de 98.65% y 0.28% respectivamente.

Se resumen los resultados de los distintos métodos de segmentación en la Tabla 4, donde es más fácil notar que el espacio de color HSV tiene la menor tasa de error tipo 1 y el espacio YCbCr la mayor tasa de aciertos.

Métodos	TPR	FPR
Modelo de fondo	0.9955	0.0057
RGB	0.9950	0.0055
YCbCr	0.9968	0.0050
HSV	0.9865	0.0028

Tabla 4: Comparación de métodos de segmentación.

Sin embargo, los pixeles no detectados con HSV son pixeles del contorno de la garra, donde aparecen leves sombras poco relevantes para detectar defectos, que incluso en algunos casos inducen a error. Es por esto que se utilizará éste último método para separar la garra del fondo en lo que sigue del trabajo.

## 6.2 Clasificación por peso

Se toma un set de 140 muestras pesadas y se crean máscaras de segmentación utilizando el espacio de color HSV como se describe en la sección 6.1.4, para luego calcular la correlación entre el peso y las distintas variables obtenidas a partir de las máscaras.

Variables	Pixeles en máscara de lado superior	Pixeles en máscara de lado inferior	Suma de pixeles de ambos lados
Correlación	0.7861	0.7532	0.7894

Tabla 5: Correlaciones entre peso de muestra y pixeles en las máscaras.

Según lo indicado en la Tabla 5, se puede afirmar que es mejor usar la suma de pixeles de ambas máscaras que usar solo los pixeles de un lado de la garra, sin embargo, la correlación que tiene con el peso real de la garra no es suficiente para aceptar este método como una definitivo para estimar el peso de las garras.

## 6.3 Clasificación por defectos. Histogramas de color

Se entrenó una red neuronal con una arquitectura de red igual en todos los casos para hacerlos comparables. Es imposible entrenar una red con una capa oculta en el set más grande de datos ya que el computador no es capaz de procesarlo en un tiempo razonable, por lo que la arquitectura de red no contendrá capa oculta. La función de activación de las neuronas de la

capa de salida será tangente-sigmoidea y el algoritmo *Backpropagation* de entrenamiento será el del gradiente descendente de segundo orden, estos últimos parámetros de la red mostraron un mejor rendimiento general con estos sets de datos.

De las 521 muestras de garras, se utilizó un 70% para entrenamiento y un 30% para pruebas. Los detalles de cómo está formado cada conjunto se puede ver en la Tabla 1.

Con las 5 formas de construir un vector de características sin utilizar PCA y con redes neuronales entrenadas con los parámetros anteriormente mencionados, no se obtienen resultados satisfactorios. Los 2 mejores resultados se obtienen con cuantizaciones a 16 y 64 bins, los que se pueden observar en la Ilustración 81.

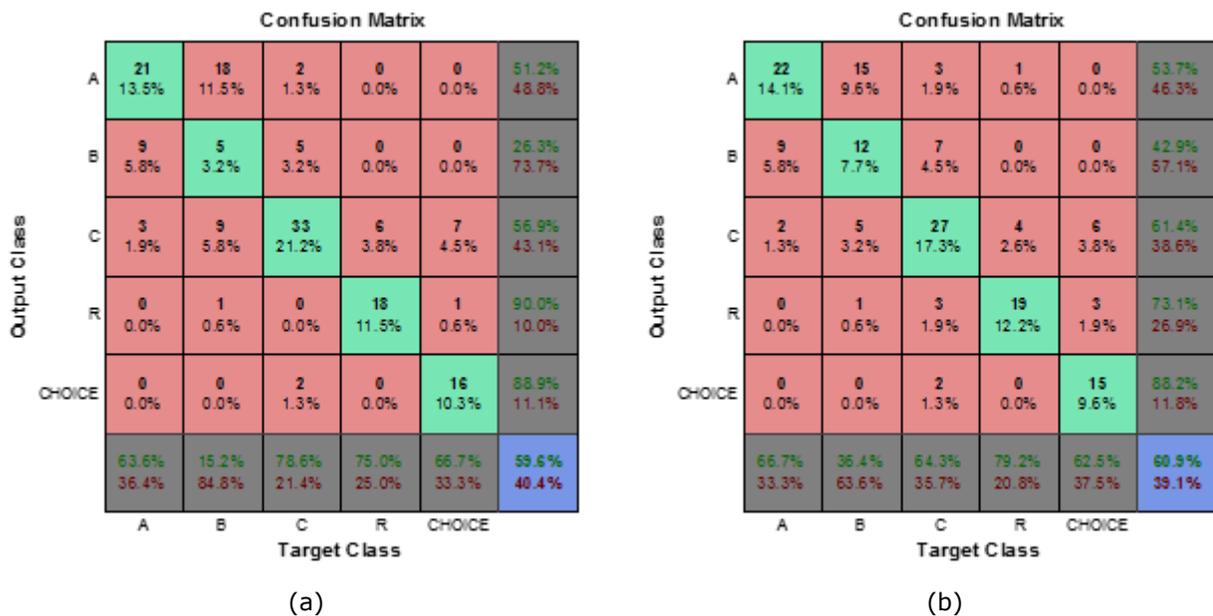


Ilustración 81: Resultados de clasificación sin usar PCA y con cuantización a 16 bins (a) y 64 bins (b).

Aplicar PCA no mejora significativamente el resultado, ya que no se logra una exactitud mayor a un 63%. En la Ilustración 82 se muestran los 2 mejores resultados obtenidos usando PCA en los casos con cuantizaciones a 16 y 32 bins.

**Confusion Matrix**

Output Class	A	18 11.5%	13 8.3%	1 0.6%	0 0.0%	0 0.0%	56.3% 43.8%
	B	11 7.1%	13 8.3%	7 4.5%	0 0.0%	1 0.6%	40.6% 59.4%
	C	4 2.6%	7 4.5%	29 18.6%	5 3.2%	5 3.2%	58.0% 42.0%
	R	0 0.0%	0 0.0%	1 0.6%	19 12.2%	1 0.6%	90.5% 9.5%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	0 0.0%	17 10.9%	81.0% 19.0%
		54.5% 45.5%	39.4% 60.6%	69.0% 31.0%	79.2% 20.8%	70.8% 29.2%	61.5% 38.5%
	Target Class						
	A	B	C	R	CHOICE		

(a)

**Confusion Matrix**

Output Class	A	27 17.3%	21 13.5%	2 1.3%	0 0.0%	0 0.0%	54.0% 46.0%
	B	4 2.6%	4 2.6%	6 3.8%	0 0.0%	0 0.0%	28.6% 71.4%
	C	2 1.3%	7 4.5%	31 19.9%	4 2.6%	5 3.2%	63.3% 36.7%
	R	0 0.0%	0 0.0%	0 0.0%	19 12.2%	2 1.3%	90.5% 9.5%
	CHOICE	0 0.0%	1 0.6%	3 1.9%	1 0.6%	17 10.9%	77.3% 22.7%
		81.8% 18.2%	12.1% 87.9%	73.8% 26.2%	79.2% 20.8%	70.8% 29.2%	62.8% 37.2%
	Target Class						
	A	B	C	R	CHOICE		

(b)

Ilustración 82: Resultados de clasificación con PCA y cuantización a 16 bins (a) y 32 bins (b).

La reducción de la dimensionalidad del problema por el uso de PCA permite la introducción de una capa oculta en la arquitectura en la red. En el mejor caso las componentes principales son solo 19 con una pérdida de información de solo el 5%, por lo que se utiliza la heurística de la media geométrica para establecer en 10 la cantidad de neuronas en la capa oculta para esta nueva configuración. Utilizando la misma función de activación que la capa de salida y el mismo algoritmo de entrenamiento se entrenan 5 nuevas redes neuronales que se suman a la comparación. En la Ilustración 83 se muestran los 2 mejores resultados obtenidos usando PCA y capa oculta en los casos con cuantizaciones a 32 y 256 bins.

**Confusion Matrix**

Output Class	A	27 17.3%	16 10.3%	0 0.0%	0 0.0%	0 0.0%	62.8% 37.2%
	B	2 1.3%	8 5.1%	7 4.5%	1 0.6%	0 0.0%	44.4% 55.6%
	C	3 1.9%	8 5.1%	27 17.3%	2 1.3%	7 4.5%	57.4% 42.6%
	R	1 0.6%	1 0.6%	5 3.2%	21 13.5%	2 1.3%	70.0% 30.0%
	CHOICE	0 0.0%	0 0.0%	3 1.9%	0 0.0%	15 9.6%	83.3% 16.7%
		81.8% 18.2%	24.2% 75.8%	64.3% 35.7%	87.5% 12.5%	62.5% 37.5%	62.8% 37.2%
	Target Class						
	A	B	C	R	CHOICE		

(a)

**Confusion Matrix**

Output Class	A	25 16.0%	19 12.2%	3 1.9%	0 0.0%	0 0.0%	53.2% 46.8%
	B	7 4.5%	8 5.1%	4 2.6%	1 0.6%	0 0.0%	40.0% 60.0%
	C	1 0.6%	6 3.8%	30 19.2%	4 2.6%	4 2.6%	66.7% 33.3%
	R	0 0.0%	0 0.0%	2 1.3%	19 12.2%	2 1.3%	82.6% 17.4%
	CHOICE	0 0.0%	0 0.0%	3 1.9%	0 0.0%	18 11.5%	85.7% 14.3%
		75.8% 24.2%	24.2% 75.8%	71.4% 28.6%	79.2% 20.8%	75.0% 25.0%	64.1% 35.9%
	Target Class						
	A	B	C	R	CHOICE		

(b)

Ilustración 83: Resultados de clasificación con PCA, capa oculta y cuantización a 32 bins (a) y 256 bins (b).

Los resultados obtenidos para los otros casos con distintas cuantizaciones se pueden ver en forma de matrices de confusión en el Anexo B, pero como ningún caso supera el 65% de exactitud, el método es descartado de forma definitiva.

## **6.4 Clasificación por defectos. Cuantificación y clasificación**

### 6.4.1 Cuantificación

En esta sección se compararán los distintos métodos para segmentar cada uno de los defectos para su cuantificación, al igual que en el análisis de la máscara de segmentación de la garra completa expuesto en la sección 6.1, se generarán máscaras ideales con la ayuda de Photoshop en las que el defecto a analizar se encontrará marcado en blanco y el resto de la imagen de negro. Se utilizarán los métodos propuestos para segmentar defectos y estudiar las diferencias con las máscaras ideales para obtener una tasa de aciertos y una de error de tipo 1.

Al igual que en la sección 6.1, se considerará el óptimo de una curva ROC, en punto de menor distancia a la coordenada (0,1) del mismo espacio.

#### 6.4.1.1 Diferencia con modelo de fondo

Este método está pensado para detectar callo y a pesar de que no se plantea como un buen método de segmentación, vale la pena tener un resultado numérico a modo de comparación con otros métodos. Para esto, se toman 24 imágenes de garras mostrando callo, algunas de ella contienen también partes de cutícula y hematomas.

Debido a la incertidumbre sobre el umbral que se debe utilizar para detectar callo con este método, se toman umbrales de intensidades entre 10 y 90 para construir la curva ROC que se muestra en la Ilustración 84.

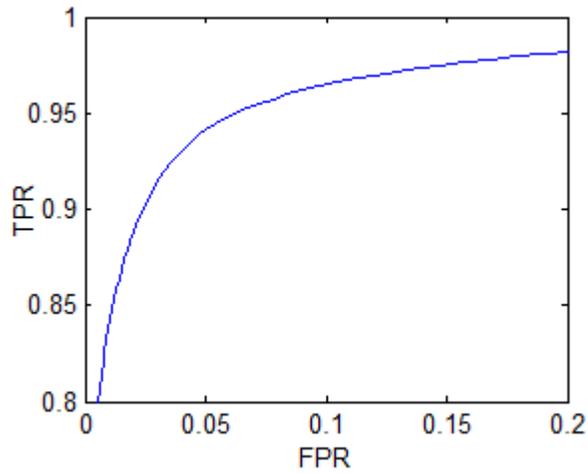


Ilustración 84: Curva ROC de umbrales en método de detección de callo.

Se ubica el óptimo en un umbral de intensidad 57, dando como resultado una tasa promedio de aciertos y error tipo 1 de 94.12% y 5% respectivamente.

#### 6.4.1.2 Umbrales RGB

##### a. Cutículas

Para este caso, solo se estudiará la sensibilidad para el caso de restar la componente azul a la componente verde, ya que el mal resultado de usar la componente roja hace que no valga la pena estudiar la sensibilidad sobre ese umbral.

Se hacen pruebas usando umbrales de intensidades entre 0 y 60 para construir una curva ROC que permita definir el calibre óptimo, la cuales se muestra en la Ilustración 85.

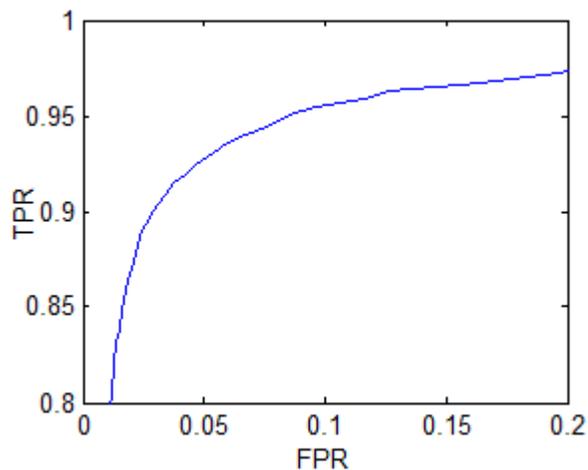


Ilustración 85: Curva ROC de umbrales en método RGB de detección de cutícula.

Lo anterior permite establecer un umbral óptimo de intensidad 26, donde la tasa de aciertos es 92.57% y la de error tipo 1 es 6.01%.

b. Callos

Utilizando el mismo set de muestras y máscaras de la sección 6.4.1.1 se hacen pruebas usando umbrales de intensidad cercanos al propuesto en 75 y calculando la tasa de aciertos y errores para cada umbral. Los resultados se muestran en la Ilustración 86.

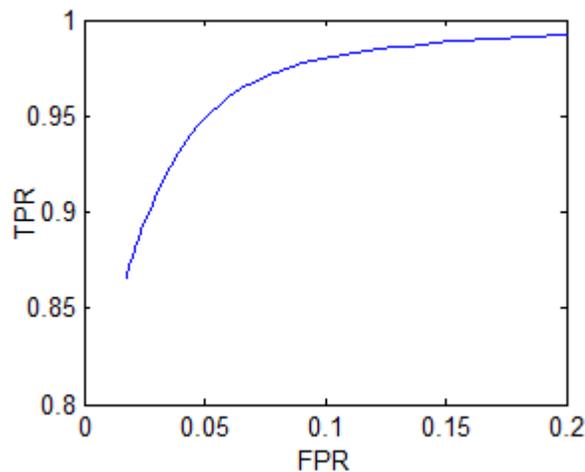


Ilustración 86: Curva ROC de umbrales en método RGB de detección de callo.

Se ubica el óptimo en un umbral de intensidad 62, dando como resultado una tasa promedio de aciertos y error tipo 1 de 95.27% y 5.32% respectivamente.

### 6.4.1.3 Clasificadores multiclase

Una red neuronal puede estar armada de distintas formas en cuanto a capas ocultas y funciones de activación en cada capa, pudiendo lograr un desempeño distinto para una misma aplicación. Algunas arquitecturas de red entregan muy malos resultados que no vale la pena estudiar, como los mostrados en la Ilustración 87b donde los callos y cutículas son clasificados como un mismo defecto y los hematomas son pasados por alto al utilizar el escalón como función de activación. Los resultados que sí se tomaron en cuenta para hacer la comparación, son aquellos que entregan una máscara de segmentación que a la vista concuerda con lo que muestra la garra en la imagen, tal como se puede ver en la Ilustración 87c donde la garra sana se muestra en verde, el callo en azul, los hematomas en rojo y las cutículas en amarillo.

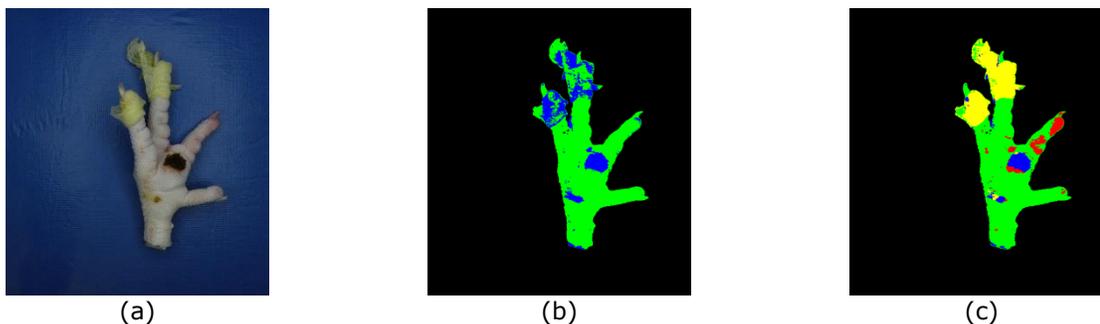


Ilustración 87: (a) Garra del caso V. (b) Mala máscara de segmentación obtenida con arquitectura de red descartada. (c) Buena máscara de segmentación obtenida con arquitectura de red incluida en el estudio de resultados.

Las pruebas incluyen casos sin capa oculta y con capa oculta de 3 o 4 neuronas. Las funciones de activación incluidas son la función escalón, tangente-sigmoidea, logaritmo-sigmoidea y una función lineal. Se probaron 2 algoritmos de entrenamiento *Backpropagation* de gradiente descendente, uno de segundo orden *Levenberg-Marquardt* y uno con momentum, éste último no entregó ningún resultado aceptable.

No solo se consideran las redes neuronales como clasificador multiclase, sino que también se incluyen pruebas con KNN con K entre 1 y 4; y árboles de decisión.

Se estudiará la tasa de aciertos y error tipo 1 para cada defecto por separado con tal de hacerlos comparables con otros métodos.

#### a. Cutículas

Se toma el mismo set de muestras de la sección 6.4.1.2a, formado por un conjunto de imágenes y sus cutículas enmascaradas de forma ideal. Se clasifican los pixeles de cada imagen utilizando los métodos propuestos, pero se considera la detección de cutícula como un caso positivo y la detección de otra cosa como caso negativo.

En el Anexo C, Tabla 12, se muestran los resultados obtenidos al comparar las máscaras ideales con las obtenidas utilizando redes neuronales y en la Tabla 13 con KNN y árboles. Los mejores resultados se obtienen con redes neuronales, en la Tabla 6 se rescatan las arquitecturas que entregaron mejores resultados.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
3	Log-sig	Lineal	0.8978	0.0168
3	Log-sig	Tan-sig	0.8920	0.0356
3	Tan-sig	Lineal	0.8973	0.0167

Tabla 6: Desempeño de cuantificación de cutícula por redes neuronales.

Los resultados anteriores definen como óptima a una red con 3 neuronas en la capa oculta, con logaritmo-sigmoidea como su función de activación y una función lineal como función de activación en las neuronas de la capa de salida, logrando una tasa de aciertos de 89.78% y de error tipo 1 de 1.68%.

b. Callos

Se toma el mismo set de muestras de la sección 6.4.1.1, formado por un conjunto de imágenes y sus callos enmascarados de forma ideal. Se clasifican los pixeles de cada imagen utilizando los métodos propuestos, pero se considera la detección de callos como un caso positivo y la detección de otra cosa como caso negativo.

En el Anexo C, Tabla 14, se muestran los resultados obtenidos al comparar las máscaras ideales con las obtenidas utilizando redes neuronales y en la Tabla 15 con KNN y árboles. Los mejores resultados se obtienen con redes neuronales, en la Tabla 7 se rescatan las arquitecturas que entregaron mejores resultados.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Tan-sig	0.9712	0.0195
3	Lineal	Tan-sig	0.9715	0.0197
4	Lineal	Tan-sig	0.9713	0.0196

Tabla 7: Desempeño de cuantificación de callo por redes neuronales.

Los resultados anteriores definen como óptima a una red con 3 neuronas en la capa oculta, con una función lineal como función de activación y la tangente-sigmoidea como función de activación en la capa de salida, logrando una tasa de aciertos de 97.15% y de error tipo 1 de 1.97%.

c. Hematomas

Evaluar el clasificador de hematomas no es tan fácil como los anteriores, ya que la creación manual de una máscara de segmentación no es tan precisa como la de un callo o las cutículas. Esto se debe a que no existe un límite claro entre un hematoma y la garra sana.

Se tomará un set de 16 imágenes a las cuales se les generará manualmente, con la ayuda de Photoshop, máscaras de segmentación ideales para los hematomas. Es de esperar que la tasa de aciertos y de error se vea afectada, no necesariamente por un mal desempeño del método, sino que, por una máscara de segmentación ideal imprecisa.

En el Anexo C, Tabla 16, se muestran los resultados obtenidos al comparar las máscaras ideales con las obtenidas utilizando redes neuronales y en la Tabla 17 con KNN y árboles. Los mejores resultados se obtienen con redes neuronales, en la Tabla 8 se rescatan las arquitecturas que entregaron mejores resultados.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
3	Tan-sig	Tan-sig	0.7902	0.0210
4	Log-sig	Tan-sig	0.7875	0.0203
4	Tan-sig	Tan-sig	0.7825	0.0191

Tabla 8: Desempeño de cuantificación de callo por redes neuronales.

Los resultados anteriores definen como optima a una red con una capa oculta con 3 neuronas, con función de activación tangente-sigmoidea en la capa oculta y la capa de salida, logrando una tasa de aciertos de 79.02% y de error tipo 1 de 2.1%. Lo anterior no es un buen resultado ya que significa más de un 20% de no detección, sin embargo, parte de él se le atribuye a la manera imprecisa de hacer la máscara ideal.

El gran problema de un único clasificador multiclase, es que solo se debe utilizar una arquitectura y cada defecto es segmentado de mejor manera con un método distinto. A continuación, se presenta una cascada de clasificadores binarios que no solo permite mezclar algoritmos, sino que también abre la posibilidad de aplicar operadores morfológicos entre un nivel y otro para mejorar los resultados.

#### 6.4.1.4 Cascada de clasificadores binarios

Para no incluir falsas detecciones arrastradas por niveles superiores de la cascada, cada uno se evaluará de forma independiente, sin quitar los defectos detectados por el nivel anterior. Las pruebas que se harán en este caso, además de probar los distintos clasificadores, pone a prueba la efectividad de aplicar operadores morfológicos a la máscara de segmentación obtenida con cada uno de ellos.

##### a. Cutículas

Se toma el mismo set de muestras de la sección 6.4.1.2a, formado por un conjunto de imágenes y sus cutículas enmascaradas de forma ideal.

En la Tabla 9 se muestran los mejores resultados, los cuales son obtenidos utilizando operadores morfológicos. En el Anexo D se muestran el resto de los resultados comparando distintos clasificadores y los casos de aplicar o no operadores morfológicos (Tabla 18 a Tabla 21).

Clasificador	TPR	FPR
Árboles	0.9499	0.0292
KNN, K=2	0.9529	0.0308
KNN, K=4	0.9441	0.0271

Tabla 9: Desempeño de mejores clasificadores de cutícula.

Lo anterior muestra que utilizar KNN con K=2 y aplicando operadores morfológicos, se obtiene el mejor resultado, con un 95.29% de aciertos y 3.08% de error tipo 1. El resultado anterior supera a todos los métodos propuestos para segmentar cutículas, por lo que se utilizará como método de cuantificación para la sección 6.4.2.

b. Callos

Se toma el mismo set de muestras de la sección 6.4.1.1, formado por un conjunto de imágenes y sus callos enmascarados de forma ideal.

En la Tabla 10 se muestran los mejores resultados, los cuales son obtenidos utilizando operadores morfológicos. En el Anexo D se muestran el resto de los resultados comparando distintos clasificadores y los casos de aplicar o no operadores morfológicos (Tabla 22 a Tabla 25).

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
2	Log-sig	Tan-sig	0.9943	0.0256
3	Log-sig	Tan-sig	0.9944	0.0248
3	Tan-sig	Lineal	0.9925	0.0228

Tabla 10: Desempeño de mejores clasificadores de callo.

De lo anterior se puede calcular que lo óptimo es utilizar una red neuronal con 3 neuronas en la capa oculta, función de activación tangente-sigmoidea para la capa oculta y lineal para la de salida. Lo que entrega una alta tasa de aciertos de 99.29% y una tasa de error tipo 1 relativamente baja correspondiente a 2.28%. El resultado anterior supera a todos los métodos propuestos para segmentar callos, por lo que se utilizará como método de cuantificación para la sección 6.4.2.

c. Hematomas

Al igual que en la sección 6.4.1.3c, se utilizará un set imágenes de garras con hematomas y una máscara creada manualmente, la cual no es del todo precisa debido a la dificultad de establecer un límite entre un hematoma y la garra sana.

En la Tabla 11 se muestran los mejores resultados, los cuales son obtenidos utilizando operadores morfológicos. En el Anexo D se muestran el

resto de los resultados comparando distintos clasificadores y los casos de aplicar o no operadores morfológicos (Tabla 26 a Tabla 29).

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
2	Log-sig	Log-sig	0.8318	0.0266
2	Log-sig	Tan-sig	0.8633	0.0324
2	Tan-sig	Log-sig	0.8278	0.0240

Tabla 11: Desempeño de mejores clasificadores de hematomas.

Lo anterior ubica como óptimo a una red neuronal con una capa oculta de 2 neuronas, con función de activación logaritmo-sigmoidea y tangente-sigmoidea en las capas oculta y de salida respectivamente. Lo que entrega una tasa de aciertos de 86.33% y una tasa de error tipo 1 de 3.24%. Si bien el resultado anterior no es bueno, se debe considerar el error arrastrado en la creación de máscaras ideales de segmentación de hematomas.

Este es el método con mejor desempeño para la detección de hematomas, por lo que se utilizará como método de cuantificación para la sección 6.4.2.

#### 6.4.2 Clasificación

Una vez que se logra obtener un método efectivo para hacer cuantificación de defectos, se debe determinar el calibre correspondiente a cada muestra. A continuación, se analizan los resultados de las distintas formas propuestas de hacer la clasificación mencionada.

Se debe considerar que la parcialidad en la asignación de calibres por parte de los operarios afecta tanto en el entrenamiento como a las pruebas realizadas en esta sección. El hecho de aplicar criterio en la construcción del método de clasificación con umbrales, contrarresta el error mencionado, hecho que no se aplica a los clasificadores multiclase. Debido a lo anterior, se espera obtener mejores resultados con el método de los umbrales.

##### 6.4.2.1 Clasificadores multiclase

De un total de 521 muestras, se utiliza un 70% para entrenamiento de algoritmos de clasificación multiclase como lo son KNN, árboles de decisión y redes neuronales con diferentes arquitecturas, dejando el 30% restante para pruebas. Detalles de la composición del conjunto de entrenamiento y prueba se puede ver en la Tabla 2.

Los 2 mejores resultados de clasificación se obtienen con KNN ( $K=1$ ), y con una red neuronal de 5 neuronas en la capa oculta y función de activación

tangente-sigmoidea en las capas oculta y de salida. Los resultados se muestran en forma de matriz de confusión en la Ilustración 88. En el Anexo E se pueden ver las matrices de confusión obtenidas de la clasificación del conjunto de prueba con el resto de los clasificadores propuestos. Se omiten los resultados que tienen bajo un 80% de exactitud.

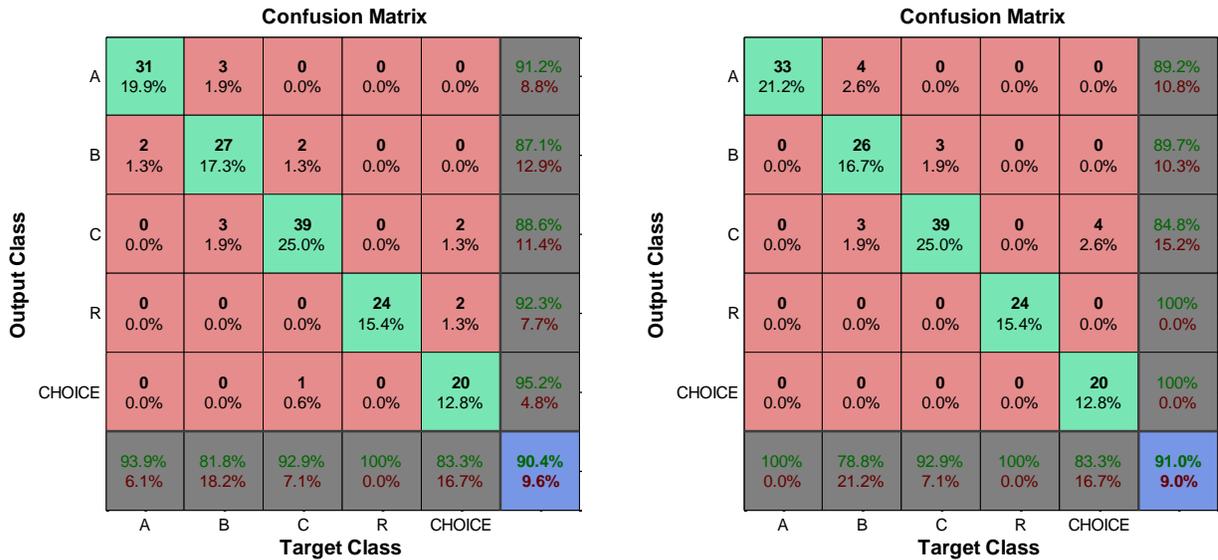


Ilustración 88: Matrices de confusión de los 2 mejores casos. (a) KNN con K=1. (b) Red neuronal con 5 neuronas en la capa oculta y función de activación tangente-sigmoidea en capa oculta y de salida.

Aunque los clasificadores tienen una exactitud parecida, el mejor resultado obtenido corresponde a la red neuronal. Considerando que clasificar una garra A como B tiene un costo distinto a cometer el error en sentido inverso, se podría considerar más adecuado utilizar KNN, ya que el costo para empresa de ese 0.6% adicional de error puede ser menor.

Se pueden hacer cambios en los bias de la red neuronal con tal de ajustar que los errores tienen costos distintos, sin embargo, este proceso no es directo ya que hacer un reajuste para la neurona encargada de detectar un calibre, puede repercutir en la detección de los demás.

#### 6.4.2.2 Umbrales

Se utiliza la tabla construida observando el conjunto de entrenamiento para clasificar el mismo conjunto de prueba de la sección anterior, con el fin de que ambas formas de hacer la clasificación sean comparables. En la Ilustración 89 se muestra la matriz de confusión obtenida.

Si bien el resultado se acerca bastante a la clasificación perfecta, existen errores de alto costo como lo es clasificar una garra de un calibre

bajo como uno mejor (garra CHOICE como C o garra B como A). Ajustar los umbrales es fácil y directo, si algunas garras están siendo clasificadas como A teniendo muchos hematomas (calibre real B), se puede disminuir la cantidad de pixeles de este defecto que puede tener el calibre y esas muestras pasarán a ser clasificadas de manera correcta. Si se aplica lo anterior al resultado obtenido, aumenta mucho el error menos grave de clasificar garras de buen calibre como uno peor (garra A clasificada como B), generando pérdidas mayores que las que se tienen con el sistema como está.

**Confusion Matrix**

<b>Output Class</b>	A	33 21.2%	1 0.6%	0 0.0%	0 0.0%	0 0.0%	97.1% 2.9%
	B	0 0.0%	30 19.2%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	C	0 0.0%	2 1.3%	42 26.9%	0 0.0%	1 0.6%	93.3% 6.7%
	R	0 0.0%	0 0.0%	0 0.0%	24 15.4%	0 0.0%	100% 0.0%
	CHOICE	0 0.0%	0 0.0%	0 0.0%	0 0.0%	23 14.7%	100% 0.0%
			100% 0.0%	90.9% 9.1%	100% 0.0%	100% 0.0%	95.8% 4.2%
		<b>Target Class</b>					
		A	B	C	R	CHOICE	

Ilustración 89: Matriz de confusión obtenida con método de umbrales.

## 7 Conclusiones

La clasificación automatizada de garras de pollo es una necesidad no satisfecha de la industria faenadora de pollos. Se logró encontrar una metodología que permite, con 97.4% de exactitud, clasificar garras de pollo sin defectos de textura en los distintos calibres.

Las principales etapas de este proceso son: La captura de dos imágenes por muestra, mostrando tanto su lado superior como el inferior; el pre procesamiento de las imágenes para separar el fondo y el primer plano; la cuantificación de cutículas, callos y hematomas utilizando una cascada de clasificadores binarios; y finalmente la clasificación de la muestra según la cantidad de cada defecto que tiene en la suma de sus caras.

Tras probar distintas formas de separar el fondo y la garra, se llega a que el mejor método es utilizar la componente de matiz (*Hue*) del espacio de color HSV y aislar el fondo en el rango de intensidad 0.552-0.656, lo que deja como detección de garra todo el resto del rango 0-1.

Con una segmentación correcta, se hicieron pruebas sin éxito para hacer una clasificación por peso, sin embargo, este tipo de separación se puede hacer de manera mucho más precisa utilizando un sistema de báscula en correa transportadora, sistema que ya existe en algunas plantas faenadoras.

Todos los métodos probados para cuantificar los callos y cutículas tuvieron un desempeño aceptable, pero la cascada de clasificadores binarios con operadores morfológicos como medida de mejora de resultados, sobresale sobre las otras formas de cuantificación. Si bien la detección de hematomas no mostró el mismo desempeño, esto se atribuye a la difícil segmentación manual de este defecto, ya que numéricamente sí representan la cantidad de hematomas que tienen las garras. El tiempo de procesamiento en cada nivel es relativamente bajo para un nivel académico, sin embargo, podrían ser demasiado altos para una aplicación industrial, por lo que se tendrían que priorizar métodos que disminuyan exactitud a cambio de velocidad.

Finalmente, la tabla de rangos construida (Tabla 3), es el método más exacto y rápido para hacer la clasificación por calibre en base a la cuantificación, logrando una exactitud de 97.4%. Esta forma de hacer la separación de calibres es fácil de ajustar en caso de que una exigencia estratégica lo requiera, por ejemplo, si en la jornada están procesando muchas garras B y pocas A, permitir un poco más de hematomas en el

calibre A castigaría el valor de esa categoría, pero aumentaría su volumen, significando un aumento en las ganancias de la empresa.

La implementación de una clasificación automática de garras de pollo podría significar grandes beneficios para una empresa como Agrosuper, ya que disminuye su costo fijo de mano de obra al pasar de un gran número de operarios, a una cantidad reducida de personal más capacitado en el funcionamiento de una línea automatizada. No es necesario que todos los operadores sean reubicados, ya que, con capacitación y entrenamiento, algunos de ellos seguirían siendo de utilidad en la operación.

Como conclusión se puede decir que se requiere poder detectar garras con uñas rotas, fracturas y raspaduras antes de poder implementar este algoritmo en una línea de garras de una planta faenadora de pollos. A pesar de las limitaciones impuestas para el desarrollo de la metodología propuesta, se han logrado los objetivos planteados en el proyecto. Cabe destacar que pese a ser éste el primer acercamiento a la automatización de este proceso, los resultados obtenidos son satisfactorios.

## 8 Bibliografía

- 1] «ARGFood,» 7 diciembre 2012. [En línea]. Available: <https://argfood.wordpress.com/2012/12/07/189/>. [Último acceso: 13 septiembre 2016].
- 2] «Agrosuper,» [En línea]. Available: <http://b2b.agrosuperfoods.com/cl/product/garra-a/>. [Último acceso: 13 septiembre 2016].
- 3] A. K. Jain, Fundamentals of Digital Image Processing, T. Kailath, Ed., Englewood Cliffs, New Jersey: Prentice Hall and System Sciences Series, 1989.
- 4] R. E. W. Rafael C. Gonzalez, Digital Image Processing, 2ª ed ed., Upper Saddle River, New Jersey: Prentice Hall, 2002.
- 5] «Torre Tavira,» [En línea]. Available: [http://www.torretavira.com/wp-content/uploads/2015/09/camaras\\_oscuras.pdf](http://www.torretavira.com/wp-content/uploads/2015/09/camaras_oscuras.pdf). [Último acceso: 13 septiembre 2016].
- 6] M. Langford, La fotografía paso a paso, Londres: Dorling Kindersley, 1979.
- 7] E. Atamian, «Blog del Fotógrafo,» [En línea]. Available: <http://www.blogdelfotografo.com/iso/>. [Último acceso: 13 septiembre 2016].
- 8] J. M. M. Martinez, Fotografia digital de alta calidad, 2ª ed ed., Barcelon: Artual, 2006.
- 9] M. M. H. R. Z. K. P. K. M. Noor A. Ibraheem, «Understanding color models: A review,» *ARPN Journal of Science and Technology*, vol. vol 2, nº nº 3, pp. 265-275, 2012.

K. Jack, Video Demystified: a handbook for the digital engineer,  
10] 5ª ed ed., Burlington: Elsevier, 2007, pp. 15-34.

P. E. H. D. G. S. Richard O. Duda, Pattern Classification, Wiley  
11] Interscience, 2000.

S. O. Haykin, Neural networks and learning machines, Prentice  
12] Hall, 2008.

G. Tong, «Weight sorting of chicken feet,» Toupack, 2016. [En  
13] línea]. Available: <https://www.youtube.com/watch?v=aSpudLjiU08>.

J. A. C. O. J. A. C. O. Christian Montoya Holguin, «Sistema  
14] automático de reconocimiento de frutas basado en visión por  
computador,» *Ingeniare. Revista chilena de ingeniería*, vol. 22, nº nº 4,  
oct. 2014.

A. K. G. F. A. K. Dibya Jyoti Bora, «Comparing the performance of  
15] LAB and HSV color spaces with respect to color image segmentation,»  
*International Journal of Emerging Technology and Advanced  
Engineering*, vol. 5, nº Issue 2, february 2015.

# Anexos

## Anexo A

Imágenes utilizadas para entrenamiento de métodos de cuantificación.

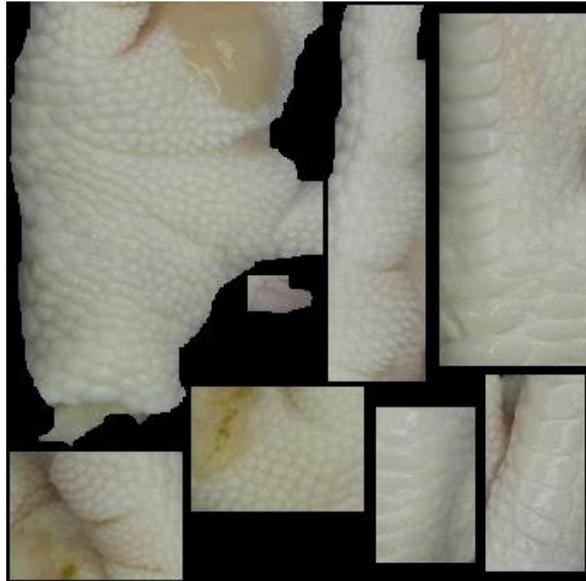


Ilustración 90: Garra sana.



Ilustración 91: Hematomas.

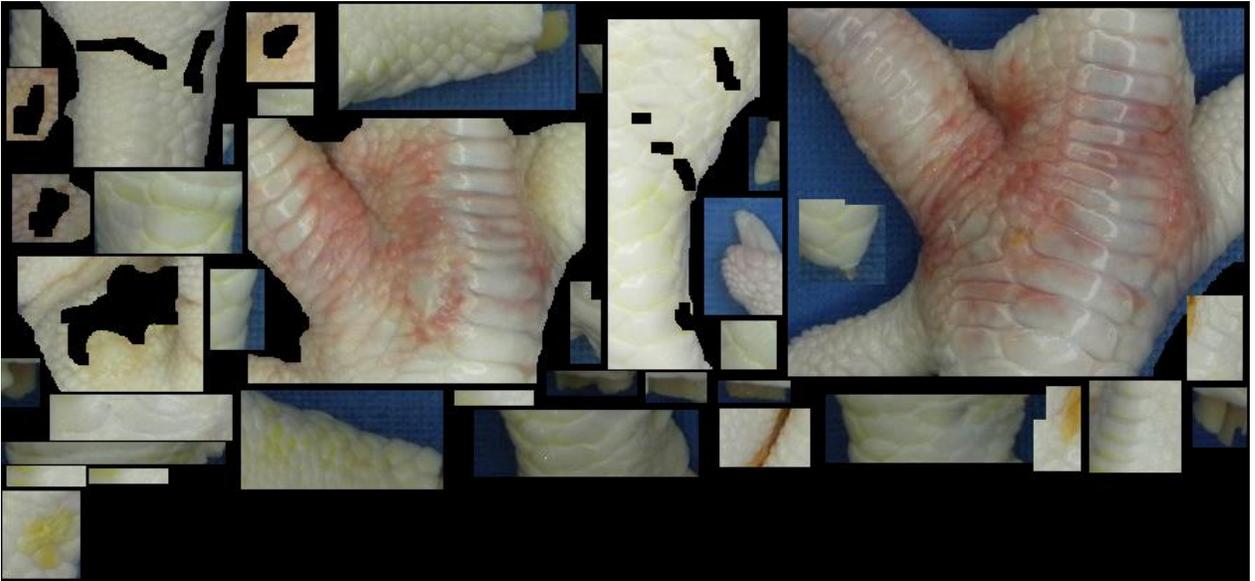


Ilustración 92: Garra sana más hematomas.

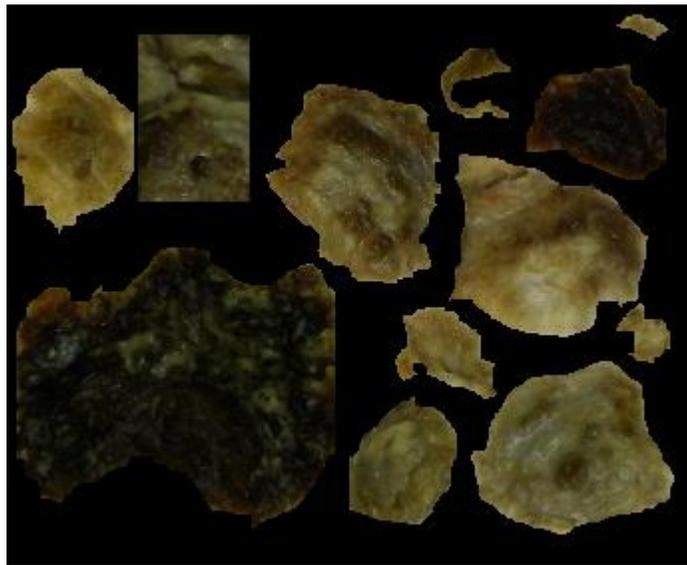


Ilustración 93: Callo

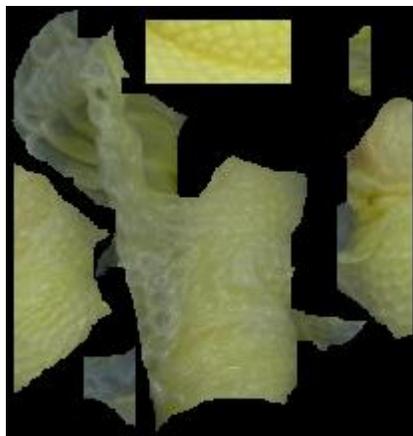


Ilustración 94: Cutículas.

## Anexo B

A continuación, se muestran las matrices de confusión detallando los resultados de la sección 6.3.

**Confusion Matrix**

Output Class	A	19 12.2%	15 9.6%	5 3.2%	0 0.0%	0 0.0%	48.7% 51.3%
	B	13 8.3%	8 5.1%	4 2.6%	0 0.0%	0 0.0%	32.0% 68.0%
	C	0 0.0%	10 6.4%	29 18.6%	7 4.5%	5 3.2%	56.9% 43.1%
	R	1 0.6%	0 0.0%	0 0.0%	17 10.9%	1 0.6%	89.5% 10.5%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	0 0.0%	18 11.5%	81.8% 18.2%
			57.6% 42.4%	24.2% 75.8%	69.0% 31.0%	70.8% 29.2%	75.0% 25.0%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

**Confusion Matrix**

Output Class	A	19 12.2%	16 10.3%	5 3.2%	0 0.0%	0 0.0%	47.5% 52.5%
	B	12 7.7%	11 7.1%	5 3.2%	0 0.0%	0 0.0%	39.3% 60.7%
	C	2 1.3%	6 3.8%	27 17.3%	5 3.2%	8 5.1%	56.3% 43.8%
	R	0 0.0%	0 0.0%	0 0.0%	19 12.2%	1 0.6%	95.0% 5.0%
	CHOICE	0 0.0%	0 0.0%	5 3.2%	0 0.0%	15 9.6%	75.0% 25.0%
			57.6% 42.4%	33.3% 66.7%	64.3% 35.7%	79.2% 20.8%	62.5% 37.5%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 95: Set sin PCA y sin cuantización.

Ilustración 96: Set sin PCA y con cuantización a 128 bins.

**Confusion Matrix**

Output Class	A	22 14.1%	15 9.6%	3 1.9%	1 0.6%	0 0.0%	53.7% 46.3%
	B	9 5.8%	12 7.7%	7 4.5%	0 0.0%	0 0.0%	42.9% 57.1%
	C	2 1.3%	5 3.2%	27 17.3%	4 2.6%	6 3.8%	61.4% 38.6%
	R	0 0.0%	1 0.6%	3 1.9%	19 12.2%	3 1.9%	73.1% 26.9%
	CHOICE	0 0.0%	0 0.0%	2 1.3%	0 0.0%	15 9.6%	88.2% 11.8%
			66.7% 33.3%	36.4% 63.6%	64.3% 35.7%	79.2% 20.8%	62.5% 37.5%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

**Confusion Matrix**

Output Class	A	25 16.0%	19 12.2%	3 1.9%	1 0.6%	2 1.3%	50.0% 50.0%
	B	4 2.6%	9 5.8%	5 3.2%	0 0.0%	2 1.3%	45.0% 55.0%
	C	4 2.6%	5 3.2%	32 20.5%	4 2.6%	15 9.6%	53.3% 46.7%
	R	0 0.0%	0 0.0%	2 1.3%	19 12.2%	5 3.2%	73.1% 26.9%
	CHOICE	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
			75.8% 24.2%	27.3% 72.7%	76.2% 23.8%	79.2% 20.8%	0.0% 100%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 97: Set sin PCA y con cuantización a 64 bins.

Ilustración 98: Set sin PCA y con cuantización a 32 bins.

**Confusion Matrix**

Output Class	A	21 13.5%	18 11.5%	2 1.3%	0 0.0%	0 0.0%	51.2% 48.8%
	B	9 5.8%	5 3.2%	5 3.2%	0 0.0%	0 0.0%	26.3% 73.7%
	C	3 1.9%	9 5.8%	33 21.2%	6 3.8%	7 4.5%	56.9% 43.1%
	R	0 0.0%	1 0.6%	0 0.0%	18 11.5%	1 0.6%	90.0% 10.0%
	CHOICE	0 0.0%	0 0.0%	2 1.3%	0 0.0%	16 10.3%	88.9% 11.1%
			63.6% 36.4%	15.2% 84.8%	78.6% 21.4%	75.0% 25.0%	66.7% 33.3%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 99: Set sin PCA y con cuantización a 16 bins.

**Confusion Matrix**

Output Class	A	17 10.9%	12 7.7%	6 3.8%	0 0.0%	0 0.0%	48.6% 51.4%
	B	11 7.1%	13 8.3%	6 3.8%	0 0.0%	0 0.0%	43.3% 56.7%
	C	5 3.2%	7 4.5%	25 16.0%	6 3.8%	6 3.8%	51.0% 49.0%
	R	0 0.0%	1 0.6%	1 0.6%	18 11.5%	3 1.9%	78.3% 21.7%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	0 0.0%	15 9.6%	78.9% 21.1%
			51.5% 48.5%	39.4% 60.6%	59.5% 40.5%	75.0% 25.0%	62.5% 37.5%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 100: Set con PCA y sin cuantización.

**Confusion Matrix**

Output Class	A	19 12.2%	13 8.3%	5 3.2%	0 0.0%	0 0.0%	51.4% 48.6%
	B	13 8.3%	15 9.6%	6 3.8%	0 0.0%	0 0.0%	44.1% 55.9%
	C	1 0.6%	5 3.2%	26 16.7%	10 6.4%	3 1.9%	57.8% 42.2%
	R	0 0.0%	0 0.0%	2 1.3%	13 8.3%	1 0.6%	81.3% 18.8%
	CHOICE	0 0.0%	0 0.0%	3 1.9%	1 0.6%	20 12.8%	83.3% 16.7%
			57.6% 42.4%	45.5% 54.5%	61.9% 38.1%	54.2% 45.8%	83.3% 16.7%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 101: Set con PCA y con cuantización a 128 bins.

**Confusion Matrix**

Output Class	A	25 16.0%	19 12.2%	4 2.6%	0 0.0%	0 0.0%	52.1% 47.9%
	B	7 4.5%	7 4.5%	8 5.1%	1 0.6%	1 0.6%	29.2% 70.8%
	C	1 0.6%	7 4.5%	25 16.0%	4 2.6%	4 2.6%	61.0% 39.0%
	R	0 0.0%	0 0.0%	1 0.6%	19 12.2%	1 0.6%	90.5% 9.5%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	0 0.0%	18 11.5%	81.8% 18.2%
			75.8% 24.2%	21.2% 78.8%	59.5% 40.5%	79.2% 20.8%	75.0% 25.0%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 102: Set con PCA y con cuantización a 64 bins.

**Confusion Matrix**

Output Class	A	27 17.3%	21 13.5%	2 1.3%	0 0.0%	0 0.0%	54.0% 46.0%
	B	4 2.6%	4 2.6%	6 3.8%	0 0.0%	0 0.0%	28.6% 71.4%
	C	2 1.3%	7 4.5%	31 19.9%	4 2.6%	5 3.2%	63.3% 36.7%
	R	0 0.0%	0 0.0%	0 0.0%	19 12.2%	2 1.3%	90.5% 9.5%
	CHOICE	0 0.0%	1 0.6%	3 1.9%	1 0.6%	17 10.9%	77.3% 22.7%
			81.8% 18.2%	12.1% 87.9%	73.8% 26.2%	79.2% 20.8%	70.8% 29.2%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 103: Set con PCA y con cuantización a 32 bins.

**Confusion Matrix**

Output Class	A	18 11.5%	13 8.3%	1 0.6%	0 0.0%	0 0.0%	56.3% 43.8%
	B	11 7.1%	13 8.3%	7 4.5%	0 0.0%	1 0.6%	40.6% 59.4%
	C	4 2.6%	7 4.5%	29 18.6%	5 3.2%	5 3.2%	58.0% 42.0%
	R	0 0.0%	0 0.0%	1 0.6%	19 12.2%	1 0.6%	90.5% 9.5%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	0 0.0%	17 10.9%	81.0% 19.0%
			54.5% 45.5%	39.4% 60.6%	69.0% 31.0%	79.2% 20.8%	70.8% 29.2%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 104: Set con PCA y con cuantización a 16 bins.

**Confusion Matrix**

Output Class	A	25 16.0%	19 12.2%	3 1.9%	0 0.0%	0 0.0%	53.2% 46.8%
	B	7 4.5%	8 5.1%	4 2.6%	1 0.6%	0 0.0%	40.0% 60.0%
	C	1 0.6%	6 3.8%	30 19.2%	4 2.6%	4 2.6%	66.7% 33.3%
	R	0 0.0%	0 0.0%	2 1.3%	19 12.2%	2 1.3%	82.6% 17.4%
	CHOICE	0 0.0%	0 0.0%	3 1.9%	0 0.0%	18 11.5%	85.7% 14.3%
			75.8% 24.2%	24.2% 75.8%	71.4% 28.6%	79.2% 20.8%	75.0% 25.0%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 105: Set con PCA, con capa oculta y sin cuantización.

**Confusion Matrix**

Output Class	A	26 16.7%	18 11.5%	3 1.9%	1 0.6%	0 0.0%	54.2% 45.8%
	B	6 3.8%	6 3.8%	4 2.6%	1 0.6%	0 0.0%	35.3% 64.7%
	C	1 0.6%	9 5.8%	29 18.6%	2 1.3%	6 3.8%	61.7% 38.3%
	R	0 0.0%	0 0.0%	2 1.3%	18 11.5%	1 0.6%	85.7% 14.3%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	2 1.3%	17 10.9%	73.9% 26.1%
			78.8% 21.2%	18.2% 81.8%	69.0% 31.0%	75.0% 25.0%	70.8% 29.2%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 106: Set con PCA, con capa oculta y con cuantización a 128 bins.

**Confusion Matrix**

Output Class	A	25 16.0%	20 12.8%	3 1.9%	1 0.6%	0 0.0%	51.0% 49.0%
	B	8 5.1%	8 5.1%	7 4.5%	0 0.0%	0 0.0%	34.8% 65.2%
	C	0 0.0%	5 3.2%	26 16.7%	3 1.9%	4 2.6%	68.4% 31.6%
	R	0 0.0%	0 0.0%	4 2.6%	20 12.8%	5 3.2%	69.0% 31.0%
	CHOICE	0 0.0%	0 0.0%	2 1.3%	0 0.0%	15 9.6%	88.2% 11.8%
			75.8% 24.2%	24.2% 75.8%	61.9% 38.1%	83.3% 16.7%	62.5% 37.5%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

**Confusion Matrix**

Output Class	A	27 17.3%	16 10.3%	0 0.0%	0 0.0%	0 0.0%	62.8% 37.2%
	B	2 1.3%	8 5.1%	7 4.5%	1 0.6%	0 0.0%	44.4% 55.6%
	C	3 1.9%	8 5.1%	27 17.3%	2 1.3%	7 4.5%	57.4% 42.6%
	R	1 0.6%	1 0.6%	5 3.2%	21 13.5%	2 1.3%	70.0% 30.0%
	CHOICE	0 0.0%	0 0.0%	3 1.9%	0 0.0%	15 9.6%	83.3% 16.7%
			81.8% 18.2%	24.2% 75.8%	64.3% 35.7%	87.5% 12.5%	62.5% 37.5%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 107: Set con PCA, con capa oculta y con cuantización a 64 bins.

Ilustración 108: Set con PCA, con capa oculta y con cuantización a 32 bins.

**Confusion Matrix**

Output Class	A	27 17.3%	23 14.7%	5 3.2%	2 1.3%	0 0.0%	47.4% 52.6%
	B	3 1.9%	4 2.6%	2 1.3%	0 0.0%	0 0.0%	44.4% 55.6%
	C	3 1.9%	6 3.8%	33 21.2%	5 3.2%	6 3.8%	62.3% 37.7%
	R	0 0.0%	0 0.0%	0 0.0%	17 10.9%	2 1.3%	89.5% 10.5%
	CHOICE	0 0.0%	0 0.0%	2 1.3%	0 0.0%	16 10.3%	88.9% 11.1%
			81.8% 18.2%	12.1% 87.9%	78.6% 21.4%	70.8% 29.2%	66.7% 33.3%
		A	B	C	R	CHOICE	
		<b>Target Class</b>					

Ilustración 109: Set con PCA, con capa oculta y con cuantización a 16 bins.

## Anexo C

A continuación, se muestran los resultados de distintos métodos en la cuantificación de defectos utilizando clasificadores multiclase.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Lineal	0.5830	0.0019
Sin capa	-	Tan-sig	0.8765	0.0156
3	Log-sig	Lineal	0.8978	0.0168
3	Log-sig	Tan-sig	0.8920	0.0356
3	Lineal	Log-sig	0.8253	0.0274
3	Lineal	Tan-sig	0.8747	0.0154
3	Tan-sig	Lineal	0.8973	0.0167
3	Tan-sig	Tan-sig	0.8326	0.0292
4	Log-sig	Lineal	0.5002	0.0402
4	Log-sig	Tan-sig	0.8496	0.0336
4	Lineal	Tan-sig	0.8757	0.0154
4	Tan-sig	Lineal	0.7482	0.0361
4	Tan-sig	Tan-sig	0.8520	0.0324

Tabla 12: Desempeño de redes neuronales en detección de cutícula.

Clasificador	TPR	FPR
Árboles	0.8547	0.0366
KNN, K=1	0.8619	0.0352
KNN, K=2	0.8335	0.0268
KNN, K=3	0.8688	0.0355
KNN, K=4	0.8522	0.0314

Tabla 13: Desempeño de KNN y árboles en detección de cutículas.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Lineal	0.9819	0.0431
Sin capa	-	Tan-sig	0.9712	0.0195
3	Log-sig	Lineal	0.9620	0.0166
3	Log-sig	Tan-sig	0.9577	0.0174
3	Lineal	Log-sig	0.9691	0.0278
3	Lineal	Tan-sig	0.9715	0.0197
3	Tan-sig	Lineal	0.9635	0.0167
3	Tan-sig	Tan-sig	0.9658	0.0215
4	Log-sig	Lineal	0.9527	0.0377
4	Log-sig	Tan-sig	0.9603	0.0183
4	Lineal	Tan-sig	0.9713	0.0196
4	Tan-sig	Lineal	0.9779	0.0363
4	Tan-sig	Tan-sig	0.9616	0.0191

Tabla 14: Desempeño de redes neuronales en detección de callo.

Clasificador	TPR	FPR
Árboles	0.9648	0.0251
KNN, K=1	0.9633	0.0221
KNN, K=2	0.9592	0.0212
KNN, K=3	0.9657	0.0190
KNN, K=4	0.9635	0.0195

Tabla 15: Desempeño de KNN y árboles en detección de callos.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Lineal	0.3922	0.0036
Sin capa	-	Tan-sig	0.7452	0.0140
3	Log-sig	Lineal	0.6588	0.0176
3	Log-sig	Tan-sig	0.7419	0.0148
3	Lineal	Log-sig	0.7465	0.0151
3	Lineal	Tan-sig	0.7437	0.0139
3	Tan-sig	Lineal	0.6750	0.0182
3	Tan-sig	Tan-sig	0.7902	0.0210
4	Log-sig	Lineal	0.5529	0.0240
4	Log-sig	Tan-sig	0.7875	0.0203
4	Lineal	Tan-sig	0.7456	0.0141
4	Tan-sig	Lineal	0.6535	0.0250
4	Tan-sig	Tan-sig	0.7825	0.0191

Tabla 16: Desempeño de redes neuronales en detección de hematomas.

Clasificador	TPR	FPR
Árboles	0.7482	0.0227
KNN, K=1	0.7234	0.0204
KNN, K=2	0.6786	0.0137
KNN, K=3	0.7230	0.0179
KNN, K=4	0.7028	0.0147

Tabla 17: Desempeño de KNN y árboles en detección de hematomas.

## Anexo D

A continuación, se muestran los resultados de distintos métodos en la cuantificación de defectos utilizando clasificadores binarios. Los resultados se muestran de forma independiente según cada defecto y se comparan los casos con y sin operadores morfológicos.

Clasificador	TPR	FPR
Árboles	0.6881	0.0056
KNN, K=1	0.5892	0.0035
KNN, K=2	0.7030	0.0056
KNN, K=3	0.6359	0.0039
KNN, K=4	0.6976	0.0050

Tabla 18: Desempeño de clasificadores de cutícula KNN y árboles sin operadores morfológicos.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Tan-sig	0.7217	0.0028
2	Log-sig	Lineal	0.7113	0.0026
2	Log-sig	Tan-sig	0.7153	0.0038
2	Lineal	Log-sig	0.7008	0.0025
2	Lineal	Tan-sig	0.7219	0.0028
2	Tan-sig	Log-sig	0.6834	0.0028
2	Tan-sig	Lineal	0.7210	0.0028
3	Log-sig	Lineal	0.7211	0.0028
3	Log-sig	Tan-sig	0.7041	0.0031
3	Tan-sig	Lineal	0.7219	0.0028
3	Tan-sig	Tan-sig	0.7144	0.0038

Tabla 19: Desempeño de clasificadores de cutícula tipo redes neuronales sin operadores morfológicos.

Clasificador	TPR	FPR
Árboles	0.9499	0.0292
KNN, K=1	0.9306	0.0250
KNN, K=2	0.9529	0.0308
KNN, K=3	0.9304	0.0248
KNN, K=4	0.9441	0.0271

Tabla 20: Desempeño de clasificadores de cutícula KNN y árboles con operadores morfológicos.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Tan-sig	0.8539	0.0188
2	Log-sig	Lineal	0.8424	0.0182
2	Log-sig	Tan-sig	0.8712	0.0211
2	Lineal	Log-sig	0.8383	0.0176
2	Lineal	Tan-sig	0.8536	0.0188
2	Tan-sig	Log-sig	0.8364	0.0171
2	Tan-sig	Lineal	0.8526	0.0187
3	Log-sig	Lineal	0.8501	0.0187
3	Log-sig	Tan-sig	0.8665	0.0192
3	Tan-sig	Lineal	0.8535	0.0189
3	Tan-sig	Tan-sig	0.8709	0.0212

Tabla 21: Desempeño de clasificadores de cutícula tipo redes neuronales con operadores morfológicos.

Clasificador	TPR	FPR
Árboles	0.9506	0.0157
KNN, K=1	0.9396	0.0148
KNN, K=2	0.9602	0.0197
KNN, K=3	0.9492	0.0149
KNN, K=4	0.9571	0.0172

Tabla 22: Desempeño de clasificadores de callo KNN y árboles sin operadores morfológicos.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Tan-sig	0.9244	0.0486
2	Log-sig	Lineal	0.9170	0.0344
2	Log-sig	Tan-sig	0.9521	0.0111
2	Lineal	Tan-sig	0.9242	0.0485
2	Tan-sig	Log-sig	0.9255	0.0515
2	Tan-sig	Lineal	0.9260	0.0100
2	Tan-sig	Tan-sig	0.9500	0.0134
3	Log-sig	Log-sig	0.9475	0.0127
3	Log-sig	Lineal	0.9260	0.0101
3	Log-sig	Tan-sig	0.9488	0.0107
3	Lineal	Tan-sig	0.9243	0.0487
3	Tan-sig	Lineal	0.9300	0.0080
3	Tan-sig	Tan-sig	0.9499	0.0119

Tabla 23: Desempeño de clasificadores de callo tipo redes neuronales sin operadores morfológicos.

Clasificador	TPR	FPR
Árboles	0.9971	0.0405
KNN, K=1	0.9966	0.0402
KNN, K=2	0.9976	0.0499
KNN, K=3	0.9964	0.0356
KNN, K=4	0.9971	0.0399

Tabla 24: Desempeño de clasificadores de callo KNN y árboles con operadores morfológicos.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Tan-sig	0.9876	0.0721
2	Log-sig	Lineal	0.9908	0.0580
2	Log-sig	Tan-sig	0.9943	0.0256
2	Lineal	Tan-sig	0.9876	0.0720
2	Tan-sig	Log-sig	0.9886	0.0746
2	Tan-sig	Lineal	0.9909	0.0280
2	Tan-sig	Tan-sig	0.9950	0.0292
3	Log-sig	Log-sig	0.9942	0.0295
3	Log-sig	Lineal	0.9908	0.0282
3	Log-sig	Tan-sig	0.9944	0.0248
3	Lineal	Tan-sig	0.9876	0.0722
3	Tan-sig	Lineal	0.9925	0.0228
3	Tan-sig	Tan-sig	0.9945	0.0268

Tabla 25: Desempeño de clasificadores de callo tipo redes neuronales con operadores morfológicos.

Clasificador	TPR	FPR
Árboles	0.7488	0.0247
KNN, K=1	0.7241	0.0225
KNN, K=2	0.6780	0.0142
KNN, K=3	0.7236	0.0196
KNN, K=4	0.7017	0.0152

Tabla 26: Desempeño de clasificadores de hematomas KNN y árboles sin operadores morfológicos.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Tan-sig	0.7661	0.0174
2	Log-sig	Log-sig	0.7795	0.0198
2	Log-sig	Lineal	0.6993	0.0108
2	Log-sig	Tan-sig	0.8183	0.0246
2	Lineal	Tan-sig	0.7621	0.0166
2	Tan-sig	Log-sig	0.7659	0.0172
2	Tan-sig	Lineal	0.7729	0.0186
2	Tan-sig	Tan-sig	0.7345	0.0132
3	Log-sig	Lineal	0.6721	0.0090
3	Log-sig	Tan-sig	0.6907	0.0109
3	Lineal	Tan-sig	0.7720	0.0178
3	Tan-sig	Log-sig	0.7537	0.0164
3	Tan-sig	Lineal	0.7714	0.0184
3	Tan-sig	Tan-sig	0.7722	0.0185

Tabla 27: Desempeño de clasificadores de hematomas tipo redes neuronales sin operadores morfológicos.

Clasificador	TPR	FPR
Árboles	0.7970	0.0243
KNN, K=1	0.7741	0.0211
KNN, K=2	0.7336	0.0158
KNN, K=3	0.7743	0.0208
KNN, K=4	0.7566	0.0175

Tabla 28: Desempeño de clasificadores de hematomas KNN y árboles con operadores morfológicos.

Neuronas en capa oculta	Función de activación capa oculta	Función de activación capa de salida	TPR	FPR
Sin capa	-	Tan-sig	0.8201	0.0237
2	Log-sig	Log-sig	0.8318	0.0266
2	Log-sig	Lineal	0.7659	0.0155
2	Log-sig	Tan-sig	0.8633	0.0324
2	Lineal	Tan-sig	0.8179	0.0228
2	Tan-sig	Log-sig	0.8278	0.0240
2	Tan-sig	Lineal	0.8270	0.0252
2	Tan-sig	Tan-sig	0.7913	0.0183
3	Log-sig	Lineal	0.7419	0.0130
3	Log-sig	Tan-sig	0.7585	0.0158
3	Lineal	Tan-sig	0.8247	0.0243
3	Tan-sig	Log-sig	0.8190	0.0229
3	Tan-sig	Lineal	0.8262	0.0249
3	Tan-sig	Tan-sig	0.8265	0.0250

Tabla 29: Desempeño de clasificadores de hematomas tipo redes neuronales con operadores morfológicos.

## Anexo E

A continuación, se muestran las tablas de confusión obtenidas al clasificar el conjunto de prueba con los clasificadores multiclase que mostraron un mejor desempeño.

**Confusion Matrix**

Output Class	A	B	C	R	CHOICE	Accuracy	Precision
A	31 19.9%	3 1.9%	0 0.0%	0 0.0%	0 0.0%	91.2%	8.8%
B	2 1.3%	27 17.3%	2 1.3%	0 0.0%	0 0.0%	87.1%	12.9%
C	0 0.0%	3 1.9%	39 25.0%	0 0.0%	2 1.3%	88.6%	11.4%
R	0 0.0%	0 0.0%	0 0.0%	24 15.4%	2 1.3%	92.3%	7.7%
CHOICE	0 0.0%	0 0.0%	1 0.6%	0 0.0%	20 12.8%	95.2%	4.8%
	93.9% 6.1%	81.8% 18.2%	92.9% 7.1%	100% 0.0%	83.3% 16.7%	90.4%	9.6%
Target Class	A	B	C	R	CHOICE		

Ilustración 110: KNN con K=1.

**Confusion Matrix**

Output Class	A	B	C	R	CHOICE	Accuracy	Precision
A	33 21.2%	5 3.2%	0 0.0%	0 0.0%	0 0.0%	86.8%	13.2%
B	0 0.0%	26 16.7%	6 3.8%	0 0.0%	0 0.0%	81.3%	18.8%
C	0 0.0%	2 1.3%	36 23.1%	1 0.6%	5 3.2%	81.8%	18.2%
R	0 0.0%	0 0.0%	0 0.0%	23 14.7%	2 1.3%	92.0%	8.0%
CHOICE	0 0.0%	0 0.0%	0 0.0%	0 0.0%	17 10.9%	100%	0.0%
	100% 0.0%	78.8% 21.2%	85.7% 14.3%	95.8% 4.2%	70.8% 29.2%	86.5%	13.5%
Target Class	A	B	C	R	CHOICE		

Ilustración 111: KNN con K=2.

**Confusion Matrix**

Output Class	A	33 21.2%	4 2.6%	0 0.0%	0 0.0%	0 0.0%	89.2% 10.8%
	B	0 0.0%	26 16.7%	2 1.3%	0 0.0%	0 0.0%	92.9% 7.1%
	C	0 0.0%	3 1.9%	39 25.0%	1 0.6%	3 1.9%	84.8% 15.2%
	R	0 0.0%	0 0.0%	0 0.0%	23 14.7%	2 1.3%	92.0% 8.0%
	CHOICE	0 0.0%	0 0.0%	1 0.6%	0 0.0%	19 12.2%	95.0% 5.0%
			100% 0.0%	78.8% 21.2%	92.9% 7.1%	95.8% 4.2%	79.2% 20.8%
		A	B	C	R	CHOICE	
		Target Class					

Ilustración 112: KNN con K=3.

**Confusion Matrix**

Output Class	A	33 21.2%	6 3.8%	0 0.0%	0 0.0%	0 0.0%	84.6% 15.4%
	B	0 0.0%	25 16.0%	4 2.6%	0 0.0%	0 0.0%	86.2% 13.8%
	C	0 0.0%	2 1.3%	37 23.7%	1 0.6%	3 1.9%	86.0% 14.0%
	R	0 0.0%	0 0.0%	0 0.0%	23 14.7%	3 1.9%	88.5% 11.5%
	CHOICE	0 0.0%	0 0.0%	1 0.6%	0 0.0%	18 11.5%	94.7% 5.3%
			100% 0.0%	75.8% 24.2%	88.1% 11.9%	95.8% 4.2%	75.0% 25.0%
		A	B	C	R	CHOICE	
		Target Class					

Ilustración 113: KNN con K=4.

**Confusion Matrix**

Output Class	A	33 21.2%	5 3.2%	0 0.0%	0 0.0%	0 0.0%	86.8% 13.2%
	B	0 0.0%	18 11.5%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	C	0 0.0%	10 6.4%	38 24.4%	0 0.0%	1 0.6%	77.6% 22.4%
	R	0 0.0%	0 0.0%	0 0.0%	24 15.4%	3 1.9%	88.9% 11.1%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	0 0.0%	20 12.8%	83.3% 16.7%
			100% 0.0%	54.5% 45.5%	90.5% 9.5%	100% 0.0%	83.3% 16.7%
		A	B	C	R	CHOICE	
		Target Class					

Ilustración 114: Red Neuronal sin capa oculta, función de activación tangente-sigmoidea en capa de salida.

**Confusion Matrix**

Output Class	A	33 21.2%	3 1.9%	0 0.0%	0 0.0%	0 0.0%	91.7% 8.3%
	B	0 0.0%	29 18.6%	6 3.8%	0 0.0%	0 0.0%	82.9% 17.1%
	C	0 0.0%	1 0.6%	32 20.5%	0 0.0%	5 3.2%	84.2% 15.8%
	R	0 0.0%	0 0.0%	0 0.0%	24 15.4%	2 1.3%	92.3% 7.7%
	CHOICE	0 0.0%	0 0.0%	4 2.6%	0 0.0%	17 10.9%	81.0% 19.0%
			100% 0.0%	87.9% 12.1%	76.2% 23.8%	100% 0.0%	70.8% 29.2%
		A	B	C	R	CHOICE	
		Target Class					

Ilustración 115: Red Neuronal con 4 neuronas en la capa oculta, función de activación lineal en capa oculta y tangente-sigmoidea en capa de salida.

**Confusion Matrix**

Output Class	A	33 21.2%	4 2.6%	0 0.0%	0 0.0%	0 0.0%	89.2% 10.8%
	B	0 0.0%	27 17.3%	3 1.9%	0 0.0%	0 0.0%	90.0% 10.0%
	C	0 0.0%	2 1.3%	38 24.4%	1 0.6%	5 3.2%	82.6% 17.4%
	R	0 0.0%	0 0.0%	0 0.0%	22 14.1%	0 0.0%	100% 0.0%
	CHOICE	0 0.0%	0 0.0%	1 0.6%	1 0.6%	19 12.2%	90.5% 9.5%
		100% 0.0%	81.8% 18.2%	90.5% 9.5%	91.7% 8.3%	79.2% 20.8%	89.1% 10.9%
	A	B	C	R	CHOICE		
	Target Class						

**Confusion Matrix**

Output Class	A	33 21.2%	4 2.6%	0 0.0%	0 0.0%	0 0.0%	89.2% 10.8%
	B	0 0.0%	26 16.7%	3 1.9%	0 0.0%	0 0.0%	89.7% 10.3%
	C	0 0.0%	3 1.9%	39 25.0%	0 0.0%	4 2.6%	84.8% 15.2%
	R	0 0.0%	0 0.0%	0 0.0%	24 15.4%	0 0.0%	100% 0.0%
	CHOICE	0 0.0%	0 0.0%	0 0.0%	0 0.0%	20 12.8%	100% 0.0%
		100% 0.0%	78.8% 21.2%	92.9% 7.1%	100% 0.0%	83.3% 16.7%	91.0% 9.0%
	A	B	C	R	CHOICE		
	Target Class						

Ilustración 116: Red Neuronal con 4 neuronas en la capa oculta, función de activación tangente-sigmoidea en capa oculta y de salida.

Ilustración 117: Red Neuronal con 5 neuronas en la capa oculta, función de activación tangente-sigmoidea en capa oculta y de salida.

**Confusion Matrix**

Output Class	A	32 20.5%	3 1.9%	0 0.0%	0 0.0%	0 0.0%	91.4% 8.6%
	B	1 0.6%	29 18.6%	2 1.3%	0 0.0%	2 1.3%	85.3% 14.7%
	C	0 0.0%	1 0.6%	38 24.4%	0 0.0%	2 1.3%	92.7% 7.3%
	R	0 0.0%	0 0.0%	0 0.0%	22 14.1%	1 0.6%	95.7% 4.3%
	CHOICE	0 0.0%	0 0.0%	2 1.3%	2 1.3%	19 12.2%	82.6% 17.4%
		97.0% 3.0%	87.9% 12.1%	90.5% 9.5%	91.7% 8.3%	79.2% 20.8%	89.7% 10.3%
	A	B	C	R	CHOICE		
	Target Class						

Ilustración 118: Árbol de decisión.