



SUBSECRETARÍA DE EDUCACIÓN SUPERIOR
DIRECCIÓN GENERAL DE EDUCACIÓN SUPERIOR TECNOLÓGICA
INSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ



Instituto Tecnológico de Tuxtla Gutiérrez

División de Estudios de Posgrado e Investigación

Detección de deformaciones en superficies curvas mediante visión por computadora, con aplicación en el control de calidad de huevos avícolas

Tesis que presenta

Julio César Torres Peña

Como requisito para obtener el grado de
Maestro en Ciencias en Ingeniería Mecatrónica

Director de Tesis: **Dr. Rafaél Mota Grajales**
Codirector de Tesis: **Dr. Elías Neftalí Escobar Gómez**
Tutor: **Dr. Alejandro Medina Santiago**

Tuxtla Gutiérrez, Chiapas. México
Marzo, 2014

Dedicatoria

A dios por darme la vida, salud y fuerza en los momentos más difíciles.

A mis padres, hermanos y amigos quienes fueron un gran apoyo emocional e incondicional con los que comparto todo lo que aprendo y de quienes también sigo aprendiendo.

A Diana mi compañera de vida quien me apoyo y alentó para continuar, cuando parecía que me iba a rendir.

A mis maestros, quienes con sus comentarios y retos me hicieron ver mis errores y contribuyeron en alcanzar esta meta.

“En el rudo luchar de tu existencia, guarda en el corazón la calma. Lleva la imagen de Dios en tu conciencia, y mi retrato consévalo en tu alma.”

Anónimo

Resumen

El presente trabajo de investigación describe un método para la detección de deformaciones en superficies curvas; está enfocado de manera particular en el estudio de deformaciones en huevos avícolas, para la identificación de afectaciones en el cascaron causadas por huevos de cáscara fina o en fáfara. El sistema se basa en la detección de defectos, mediante el barrido de un patrón de luz estructurada láser y la adquisición de imágenes, resaltando las alteraciones en su geometría como consecuencia de la deformación del láser en las transiciones generadas por el barrido sobre la superficie del huevo, posteriormente se analizan las imágenes para la obtención de puntos equidistantes a lo largo de la curva evaluada y aplicando una interpolación mediante spline cúbicos es posible la extracción de características métricas descriptivas para observar la disparidad entre las curvas evidenciando las deformaciones al realizar una interposición gráfica; la información métrica adquirida, se utiliza para clasificar las muestras defectuosas mediante el desarrollo de un algoritmo empleando una red neuronal artificial, entrenada con una base de datos compuesta por 200 imágenes, obteniendo una eficiencia de 97.5% durante la evaluación de 150 muestras de huevos.

Abstract

In this work is presented a method for detecting deformations on curved surfaces, this research is particularly focused on the study of deformations in poultry eggs, for identification of damages in the shell caused by thin shelled eggs. The system is based on the detection of defects, by sweeping a pattern of laser structured light and imaging, highlighting the changes in geometry as a result of deformation of the laser transitions generated by scanning the surface of egg, then the images are analyzed to obtain equidistant points along the curve and evaluated by generating a cubic spline interpolation allows the extraction of descriptive metric characteristics to observe the disparity between curves showing deformation by performing a graph interposition, the metric information acquired is used to classify the defective samples by developing an algorithm using an artificial neural network, trained with a database composed of 200 images, achieving 97.5% efficiency during the evaluation of 150 samples of eggs.

Índice general

| | | |
|----------|---|-----------|
| 1 | Introducción..... | 1 |
| 1.1 | Estado del arte..... | 3 |
| 1.2 | Contexto del trabajo de investigación | 5 |
| 2 | Visión Artificial..... | 6 |
| 2.1 | Modelo Físico de la luz..... | 6 |
| 2.1.1 | Características Físicas..... | 7 |
| 2.2 | Visión Biológica | 8 |
| 2.2.1 | Percepción acromática..... | 9 |
| 2.2.2 | Percepción cromática..... | 10 |
| 2.3 | Visión Artificial | 11 |
| 2.3.1 | Representación de la realidad | 11 |
| 2.3.2 | Elementos de un sistema de visión | 12 |
| 2.3.3 | Etapas de un sistema de visión | 12 |
| 2.3.4 | Arquitectura de un sistema de visión..... | 13 |
| 2.3.5 | Dispositivos para visión..... | 15 |
| 2.3.6 | Adquisición y representación de imágenes..... | 15 |
| 2.3.7 | Procesamiento de la imagen | 25 |
| 3 | Redes Neuronales Artificiales..... | 42 |
| 3.1 | Neurona biológica..... | 43 |
| 3.1.1 | Descripción del funcionamiento y actividad neuronal..... | 43 |
| 3.2 | Neurona Artificial | 47 |
| 3.2.1 | Definición de una red neuronal artificial | 47 |
| 3.2.2 | Funciones de activación..... | 49 |
| 3.2.3 | Topologías de las redes neuronales | 54 |
| 3.2.4 | Entrenamiento de las redes neuronales | 55 |
| 3.2.5 | Redes de una capa..... | 56 |
| 3.2.6 | Redes multicapa..... | 62 |
| 3.2.7 | Redes de retropropagación (backpropagation) | 63 |
| 4 | Interpolación Polinómica Segmentaria..... | 68 |
| 4.1 | Interpolación con Splines..... | 69 |
| 4.2 | Splines Cúbicos..... | 71 |
| 5 | Propuesta para la detección de deformaciones | 73 |
| 5.1 | Etapas del Método..... | 74 |
| 5.2 | Adquisición de imagen | 76 |
| 5.2.1 | Arreglo experimental..... | 76 |
| 5.2.2 | Registro de Imágenes..... | 77 |

| | | |
|----------|---|------------|
| 5.3 | Pre procesamiento de la imagen..... | 79 |
| 5.3.1 | Imagen de entrada..... | 80 |
| 5.3.2 | Escala de Gris | 81 |
| 5.3.3 | Región de interés | 82 |
| 5.3.4 | Binarización..... | 83 |
| 5.3.5 | Filtro media | 84 |
| 5.3.6 | Morfología matemática..... | 85 |
| 5.3.7 | Esqueletización..... | 86 |
| 5.4 | Análisis de la imagen | 88 |
| 5.4.1 | Representación gráfica de curvas | 88 |
| 5.4.2 | Suavizado por Spline Cúbico..... | 91 |
| 5.5 | Extracción de características..... | 94 |
| 5.5.1 | Longitud curva real..... | 96 |
| 5.5.2 | Longitud curva spline | 97 |
| 5.5.3 | Área entre curvas | 99 |
| 5.6 | Clasificación de forma | 100 |
| 5.6.1 | Arquitectura de la Red Neuronal Artificial..... | 100 |
| 5.6.2 | Entrenamiento de la Red Neuronal Artificial | 101 |
| 6 | Resultados del método sobre superficies deformadas | 106 |
| 6.1 | Relación de adaptación en Spline Cúbico | 107 |
| 6.1.1 | Spline Cúbico 3 puntos..... | 108 |
| 6.1.2 | Spline Cúbico 4 puntos..... | 109 |
| 6.1.3 | Spline Cúbico 5 puntos..... | 110 |
| 6.1.4 | Spline Cúbico 6 puntos..... | 111 |
| 6.1.5 | Spline Cúbico 7 puntos..... | 112 |
| 6.2 | Evaluación de la Red Neuronal Artificial | 113 |
| 6.2.1 | Resultado RNA con Spline Cúbico de 4 puntos | 113 |
| 6.2.2 | Resultado RNA con Spline Cúbico de 5 puntos | 114 |
| 6.2.3 | Resultado RNA con Spline Cúbico de 6 puntos | 115 |
| 6.3 | Análisis de Resultados | 116 |
| 6.4 | Discusión de resultados..... | 117 |
| 7 | Conclusiones y Recomendaciones | 120 |
| | Referencias..... | 122 |

Índice de Figuras

| | |
|--|----|
| Figura 1.1 Huevos con cascara fina o en fáfara..... | 2 |
| Figura 2.1 Espectro electromagnético | 7 |
| Figura 2.2 Representación del Ojo Humano. (a) sección del ojo humano. (b) visión esquemática de las células fotorreceptoras | 9 |
| Figura 2.3 Representación acromática y cromática de una imagen | 11 |
| Figura 2.4 Etapas de un sistema de visión | 13 |
| Figura 2.5 Sistema de Visión..... | 14 |
| Figura 2.6 Elementos de una cámara fotográfica analógica | 16 |
| Figura 2.7 Sensor CCD (Coupled Charge Device)..... | 17 |
| Figura 2.8 Sensor CCD/CMOS | 18 |
| Figura 2.9 Esquema de un CCD que utiliza un filtro Bayer | 18 |
| Figura 2.10 Cámara Progresiva | 20 |
| Figura 2.11 Cámara Lineal | 21 |
| Figura 2.12 Cámaras Inteligentes | 22 |
| Figura 2.13 Organización matricial de una imagen digital..... | 23 |
| Figura 2.14 Esquema del fichero de una imagen..... | 24 |
| Figura 2.15 Histograma de una imagen..... | 26 |
| Figura 2.16 Representación matricial de las componentes RGB..... | 30 |
| Figura 2.17 Conversión a nivel de gris | 31 |
| Figura 2.18 Representación Binaria de una imagen | 34 |
| Figura 2.19 Tipos de ruido en una imagen. (a) Imagen original. (b) Escala de gris. (c) Ruido tipo Gaussiano. (d) Ruido tipo "Sal y Pimienta" | 35 |
| Figura 2.20 Eliminación de ruido gaussiano aplicando filtro de mediana. (a) Ruido Gaussiano. (b) Filtro de la mediana..... | 36 |
| Figura 2.21 Detección de manchas en hojas mediante segmentación | 37 |
| Figura 2.22 Erosión de una Imagen. (a) Imagen original. (b) Escala de gris. (c) Binarizado. (d) Erosión..... | 39 |
| Figura 2.23 Dilatación de una imagen. (a) Imagen original. (b) Escala de gris. (c) Binarización. (d) Dilatación | 40 |
| Figura 3.1 Neurona Biológica | 43 |
| Figura 3.2 Botón Sináptico..... | 44 |
| Figura 3.3 Potencial de acción..... | 46 |
| Figura 3.4 Función de activación escalón..... | 50 |
| Figura 3.5 Función de activación lineal o mixta..... | 51 |
| Figura 3.6 Función tangente hiperbólica | 52 |
| Figura 3.7 Función de activación sigmoidal..... | 53 |
| Figura 3.8 Función de activación gaussiana | 53 |
| Figura 3.9 Esquema básico de una Red Neuronal Artificial..... | 54 |
| Figura 3.10 Red de una capa | 56 |
| Figura 3.11 Función de nodo..... | 57 |
| Figura 3.12 Separación lineal de los conjuntos A y B | 58 |
| Figura 3.13 Perceptrón clásico propuesto por McCulloch - Pitts | 58 |
| Figura 3.14 Clasificación de patrones con el perceptrón..... | 61 |

| | |
|--|-----|
| Figura 3.15 Red neuronal de dos capas | 62 |
| Figura 3.16 Red de dos capas | 64 |
| Figura 3.17 Propagación de las señales en las neuronas..... | 65 |
| Figura 3.18 Retropropagación del error..... | 66 |
| Figura 3.19 Retropropagación del error hacia capas anteriores..... | 66 |
| Figura 3.20 Actualización de los pesos | 67 |
| Figura 4.1 Spline de Grado 0..... | 70 |
| Figura 4.2 Spline de Grado 1..... | 70 |
| Figura 5.1 Etapas del método | 74 |
| Figura 5.2 Diagrama esquemático del sistema de visión propuesto | 76 |
| Figura 5.3 Huevos sin deformaciones | 78 |
| Figura 5.4 Huevos con deformaciones | 78 |
| Figura 5.5 Imágenes de las muestras, posterior al ajuste del obturador..... | 78 |
| Figura 5.6 Diagrama a bloques del pre procesamiento de una imagen..... | 79 |
| Figura 5.7 Imágenes de entrada..... | 80 |
| Figura 5.8 Imagen de una curva irregular, obtenida de un huevo deformado | 80 |
| Figura 5.9 Representación de imágenes. (a) RGB. (b) Escala de Gris | 81 |
| Figura 5.10 Extracción de Región de Interés (ROI). (a) Delimitación ROI. (b) Imagen resultante | 82 |
| Figura 5.11 Binarización de imagen de una muestra deformada. (a) Escala de gris. (b) Binarizada..... | 83 |
| Figura 5.12 Filtro de media | 84 |
| Figura 5.13 Identificación de cortes en una curva | 84 |
| Figura 5.14 Resultado de la operación morfológica (cierre) | 86 |
| Figura 5.15 Esqueleto de la curva en una imagen. (a) Real. (b) Esqueleto | 87 |
| Figura 5.16 Imágenes de curvaturas presentes en huevos. (a, b) Huevos regulares. (c, d) Huevos deformados | 88 |
| Figura 5.17 Distribución de pixeles en un plano, obtenidos de la imagen de la muestra (a) | 89 |
| Figura 5.18 Distribución de pixeles en un plano, obtenidos de la imagen de la muestra (b)..... | 89 |
| Figura 5.19 Distribución de pixeles en un plano, obtenidos de la imagen de la muestra (c) | 90 |
| Figura 5.20 Distribución de pixeles en un plano, obtenidos de la imagen de la muestra (d)..... | 90 |
| Figura 5.21 Representación de curvaturas. (a) Huevo regular. (b) Huevo deforme. | 92 |
| Figura 5.22 Spline Cúbico curva sin deformaciones | 93 |
| Figura 5.23 Spline Cúbico curva deformada | 93 |
| Figura 5.24 Interposición de una curvatura real sin deformaciones y su representación mediante Spline Cúbicos a partir de 5 puntos geométricos muestreados | 94 |
| Figura 5.25 Interposición de una curvatura real deformada y su representación mediante Spline Cúbicos a partir de 5 puntos geométricos muestreados | 95 |
| Figura 5.26 Cálculo de longitud en una curva real | 97 |
| Figura 5.27 Cálculo de longitud en una curva spline | 98 |
| Figura 5.28 Cálculo del área entre curvas | 99 |
| Figura 5.29 Modelo matemático de una RNA..... | 100 |
| Figura 5.30 Red Neuronal Artificial propuesta | 101 |
| Figura 5.31 Imágenes de entrenamiento de huevos sin deformaciones | 102 |
| Figura 5.32 Imágenes de entrenamiento de huevos deformados | 103 |
| Figura 6.1 Muestras para la evaluación de la adaptación de Spline Cúbico..... | 107 |
| Figura 6.2 Spline cúbico de 3 Puntos obtenido de la muestra (b)..... | 108 |
| Figura 6.3 Spline cúbico de 3 Puntos obtenido de la muestra (c)..... | 108 |
| Figura 6.4 Spline cúbico de 4 Puntos obtenido de la muestra (b)..... | 109 |
| Figura 6.5 Spline cúbico de 4 Puntos obtenido de la muestra (c)..... | 109 |
| Figura 6.6 Spline cúbico de 5 Puntos obtenido de la muestra (b)..... | 110 |
| Figura 6.7 Spline cúbico de 5 Puntos obtenido de la muestra (c)..... | 110 |
| Figura 6.8 Spline cúbico de 6 Puntos obtenido de la muestra (b)..... | 111 |
| Figura 6.9 Spline cúbico de 6 Puntos obtenido de la muestra (c)..... | 111 |

| | |
|--|-----|
| Figura 6.10 Spline cúbico de 7 Puntos obtenido de la muestra (b)..... | 112 |
| Figura 6.11 Spline cúbico de 7 Puntos obtenido de la muestra (c)..... | 112 |
| Figura 6.12 Eficiencia Spline Cúbico 4 Puntos | 113 |
| Figura 6.13 Eficiencia Spline Cúbico 5 Puntos | 114 |
| Figura 6.14 Eficiencia Spline Cúbico 6 Puntos | 115 |
| Figura 6.15 Eficiencia óptima en la evaluación de 200 Huevos, representando el Error Cuadrático Medio (MSE) / Épocas..... | 116 |
| Figura 6.16 Gráfica de eficiencia de la red neuronal artificial | 116 |

Índice de Tablas

| | |
|--|-----|
| Tabla 2.1 Formatos para ficheros gráficos y características principales..... | 24 |
| Tabla 2.2 Píxeles de imagen en escala de gris..... | 29 |
| Tabla 6.1 Resultados Spline Cúbico 4 Puntos | 113 |
| Tabla 6.2 Resultados Spline Cúbico 5 Puntos | 114 |
| Tabla 6.3 Resultados Spline Cúbico 6 Puntos | 115 |
| Tabla 6.4 Comparación de Investigaciones | 119 |

Capítulo 1

Introducción

La necesidad de lograr una mejor calidad en los productos hace que la inspección de defectos ocupe un lugar de gran importancia dentro de los procesos de producción. De forma general, la inspección se puede definir como el proceso mediante el cual se determina si una pieza o producto se desvía de un conjunto de especificaciones dadas (Newman & Jain, 1995). En la actualidad los procesos industriales de producción y manufactura valoran las características que aseguren la calidad del producto a comercializar.

Hoy día, la producción de alimentos mantiene patrones preestablecidos y estandarizados de las características de sus productos; algunas de las más importantes son: tamaño, forma, peso, color, textura, etc., asegurando de esta manera la calidad final del producto.

En este contexto, la industria de producción de huevos avícolas es un sector con una gran demanda de consumo, esto ha ocasionado diversas investigaciones científicas enfocadas en desarrollar sistemas automatizados para el control de calidad; algunas de las principales características evaluadas son; el peso (Asadi & Raoufat, 2010), tamaño (Rashidi & Keshavarzpour, 2011; Omid & Soltani, 2013), fertilidad (Shan, 2010; Arivazhagan & Newlin, 2013; Dehrouyeh & Omid, 2010; Omid & Soltani, 2013), manchas o suciedad (Ibrahim & Mohd Zin, 2012; Arivazhagan & Newlin, 2013; Dehrouyeh & Omid, 2010), fracturas (Arivazhagan & Newlin, 2013; Omid & Soltani, 2013), entre otros.

No obstante, la identificación de deformaciones en huevos avícolas es una interesante área de investigación; con frecuencia en la producción se presentan huevos con deformaciones en el cascarón, son poco atractivos para el consumidor y susceptibles a dañarse; entre los que destacan, huevos sin cáscara en toda la membrana o “fárfara” y con cáscaras muy delgadas (Coutts & Wilson, 2007) (Figura 1.1). La incidencia de estas afectaciones en los huevos varía aproximadamente de 0,5 a 6%, y entre un 12 a 15% de huevos no son factibles para incubar (Coutts & Wilson, 2007; King’ori, 2012; Leeson & Summers, 2009; Pescatore & Jacob, 2011; Valbuena, 2010). Esto se debe comúnmente a un útero inmaduro o defectuoso, producidos por gallinas que entran en puesta prematuramente, ocasionando que el huevo sea puesto antes de que haya terminado la calcificación completa de la cáscara. Se asocia de igual manera estas deformaciones a enfermedades como Bronquitis infecciosa y el síndrome de caída de puesta (Coutts & Wilson, 2007).



Figura 1.1 Huevos con cascara fina o en fárfara

Los métodos actuales de evaluación en la calidad del huevo respecto a defectos relacionados a su forma se basan en algunos casos en una inspección manual por personal humano, sujetos a diversos aspectos fisiológicos como la fatiga ocular; además de implicar un alto grado de subjetividad en la determinación de la calidad de este producto.

Los sistemas automatizados basados en técnicas de inteligencia artificial posibilitan a la industria avícola la supervisión de sus líneas de producción con poca intervención humana; debido a esto, el objetivo de la investigación fue desarrollar un método para detectar

deformaciones en superficies curvas, aplicándolo de manera específica en la detección de defectos en el cascaron de huevos avícolas.

El método propuesto se basa en el principio de detección de defectos mediante la generación de un barrido de luz estructurada laser en forma de línea y su proyección sobre la superficie del cascaron de un huevo, registrando imágenes del haz laser deformado al entrar en contacto con la superficie analizada. Tras la adquisición de las imágenes del barrido laser, se realiza un procesamiento que permite detectar y clasificar las muestras deformes.

La clave de la identificación de los defectos reside en la obtención de puntos distribuidos en la curvatura de una imagen, para realizar una interpolación por spline cúbicos; estimando de esta manera, una curva representativa de como debiera ser la estructura del huevo analizado si no presentara deformaciones. Una posterior comparación de ambas curvas de manera gráfica permite identificar diferencias geométricas sobre la superficie de un huevo, realizando mediciones en base a pixeles de la longitud y área entre dichas curvaturas; de esta manera, es posible aplicar una red neuronal artificial para obtener una clasificación de defectos en las muestras analizadas.

1.1 Estado del arte

En particular las deformaciones son un inconveniente presentado en diversos productos, esto ha originado investigaciones centradas en detectar defectos o clasificar con base en la forma, utilizando técnicas de visión por computadora e inteligencia artificial para desarrollar robustos sistemas automatizados de control de calidad industrial. Algunas de las áreas de interés, donde se han desarrollado estas investigaciones son: Automotriz (Torneró & Armesto, 2012), Metalúrgica (Luque, Ruiz, & Hogert, 2011), Piscicultura (Storbeck & Daan, 2001), Agrícola (Shouche & Rastogi, 2001; ElMasry & Cubero, 2012; Kataoka & Al-Mallahi, 2010; Liming & Yanchao, 2010; Beyer & Hahn, 2002; Jahns & Nielsen, 2001; Rokunuzzaman & Jayasuriya, 2013), Alimentos (Brosnan & Sun, 2003; Belongie & Malik, 2002; Du & Sun, 2004).

Tornero (2012) desarrolló un sistema para la detección de defectos en carrocerías de vehículos. Proyectó un patrón de luz estructurada, mediante lámparas fluorescentes sobre la carrocería de automóviles, con una capa inicial de pintura; posteriormente, mediante un sistema de visión compuesto por una cámara, previamente calibrada a una resolución de 0.896 mm/px, situada sobre el automóvil, capturó una serie de imágenes, realizando mediciones del ancho de las líneas en píxeles, reflejadas sobre los paneles de la carrocería; de esta manera determinó las variaciones contenidas en la imagen; detectando alteraciones de 0.2, 0.3 y 0.6 mm, con una eficiencia del 100 % (Tornero & Armesto, 2012).

ElMasry (2012) desarrolló un clasificador de papas según su forma y tamaño, implementando un registro de imágenes con papas de diferentes formas y tamaños. Su método se basó en la obtención de contorno de una papa y el cálculo de parámetros como: el eje mayor, eje menor, centroide y momento de inercia; de esta manera, realizó una firma, de cada papa evaluada, al graficar la magnitud del radio del centroide hacia cada uno de los puntos del contorno; posteriormente realizó una clasificación de estos parámetros mediante una red neuronal artificial (RNA), obteniendo una clasificación promedio de 96.5% de un grupo de aprendizaje compuesto por 228 papas (ElMasry & Cubero, 2012).

Luque (2011) detectó deformaciones en tuberías, utilizando una técnica óptica denominada triangulación activa con luz estructurada. Proyectó un patrón laser en forma de anillo y analizó las imágenes registradas; realizó un procesamiento de la imagen y obtuvo el contorno del círculo proyectado, a partir de este contorno y el centro, calculó la magnitud del radio; comparando el ángulo de muestreo, respecto a la longitud del radio, para detectar variaciones, le fue posible detectar ovalidad, fisuras y corrosión en ductos (Luque, Ruiz, & Hogert, 2011).

Storbeck (2001) clasificó especies de peces, mediante la medición del ancho y largo de una línea laser proyectada a lo largo del cuerpo de cada pez; aplicando redes neuronales artificiales, evaluó ventanas de diferentes tamaños en la imagen del reflejo del haz laser sobre la superficie del pez; de esta manera, entrenó una red neuronal, como clasificador de patrones. Finalmente, analizó los patrones de la variación de la concentración de píxeles en la imagen, en relación con la forma de la línea; de esta manera obtuvo una eficiencia del 95 % en la correcta clasificación de peces (Storbeck & Daan, 2001).

1.2 Contexto del trabajo de investigación

La inspección de deformaciones es un objeto de estudio en los modernos procesos de control de calidad industriales, debido a la necesidad de contar cada vez con métodos y técnicas que aseguren una alta calidad productiva. La industria avícola es un área de interés para este tipo de análisis, debido a la necesidad de generar productos que soporten estrictos puntos calidad.

Los huevos, productos derivados del sector avícola, son afectados por diversos problemas que generan variaciones en las formas y características geométricas permisibles para obtener un producto de calidad; la producción de huevos con problemas en la estructura del cascaron como es el caso de las deformaciones, comúnmente son consecuencia de problemas fisiológicos y congénitos en las aves. La producción de huevos con estas características impacta de manera puntual en el sector productivo teniendo pérdidas de entre 12 – 15% en la incubación de productos deformes (Leeson & Summers, 2009; Rey, 2008; Coutts & Wilson, 2007); por otro lado, el sector comercial rechaza el consumo de huevos defectuosos.

Debido a la importancia que con lleva el estudio de deformaciones en la producción de huevos en el sector avícola y conforme a lo expuesto anteriormente, el trabajo de investigación consistió en desarrollar un algoritmo computacional para la inspección de deformaciones en superficies curvas, como las encontradas en el cascarón de huevos avícolas.

Capítulo 2

Visión Artificial

La visión puede ser considerada como un proceso que permite adquirir una gran cantidad de información presente en el mundo físico. El intento por tratar de emular el proceso de visión mediante un sistema computacional dio origen a lo que hoy conocemos como visión artificial, generando una de las áreas de investigación y desarrollo tecnológico de mayor importancia en los últimos años.

En este capítulo abordaremos el concepto de visión artificial, comenzando con una introducción a la visión, la conceptualización y características de los sistemas de visión artificial, así como una serie de conceptos utilizados en el procesamiento de imágenes, que serán empleados en la presente investigación.

2.1 Modelo Físico de la luz

En general, desde el punto de vista del procesado digital de imágenes, basta considerar la luz como una onda. Según el modelo ondulatorio las características de un rayo de luz vienen completamente determinadas por dos propiedades: su amplitud y su longitud de onda (Raymond, 1994; Sears, 1996; Robert, 1991), siendo una forma de energía que emiten los cuerpos luminosos y que percibimos mediante el sentido de la vista.

2.1.1 Características Físicas

Aunque las ondas luminosas constituyen una parte muy pequeña del conjunto de ondas electromagnéticas, son especialmente interesantes porque tienen la particularidad de que son captadas por los ojos y procesadas en el cerebro (Figura 2.1). El ojo humano es capaz de distinguir radiaciones de longitudes de onda comprendidas entre 400 y 700 nanómetros (1 nanómetro = 10^{-9} metros) (Sears, 1996; Robert, 1991).

Nuestro sistema sensorial visual, interpreta las diferentes amplitudes y longitudes de onda de la luz, produciendo las sensaciones que conocemos como brillo y color respectivamente. Así por ejemplo, una onda electromagnética que viaja por el vacío con una longitud de onda predominante de 650 nanómetros se interpreta en el cerebro como la sensación del color rojo.

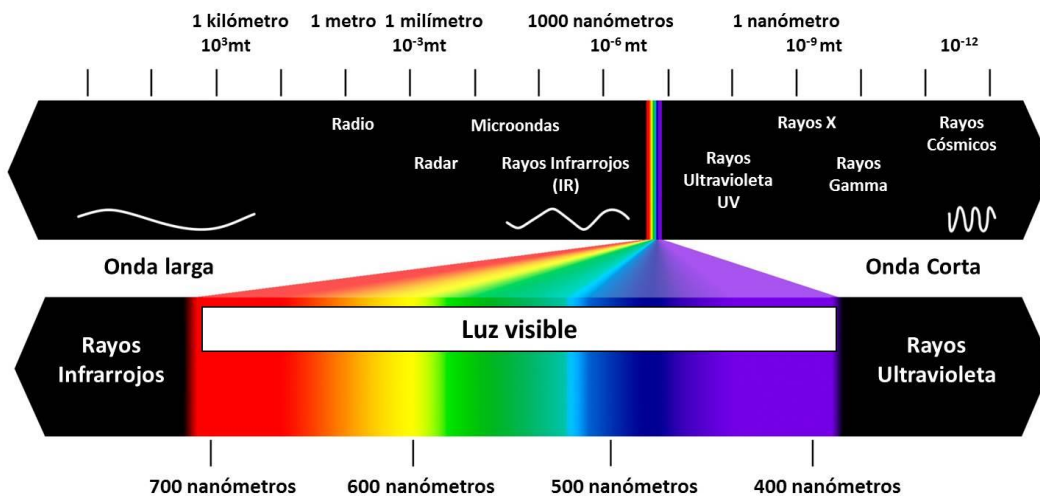


Figura 2.1 Espectro electromagnético

Flujo luminoso

El flujo luminoso es la parte del flujo radiante detectada por el ojo, expresada por la unidad de medida lumen (lm). Un lumen corresponde al flujo luminoso procedente de una abertura de $1/60 \text{ cm}^2$ en un cilindro de material refractario que contiene un material patrón que radia a través de un cono de radiación de un estereorradián (Robert, 1991). El flujo luminoso se puede medir con un fotómetro y se representa con el símbolo Φ .

Luminancia o brillo

La luminancia o brillo se define como la densidad angular y superficial de flujo luminoso que incide, atraviesa o emerge de una superficie siguiendo una dirección determinada (Robert, 1991); sirve para expresar el brillo de los objetos iluminados y es la que determina la sensación visual producida por dichos objetos. Esta magnitud es de gran importancia para evaluar el grado de deslumbramiento.

2.2 Visión Biológica

El conocimiento del mundo circundante, la percepción de los objetos y la estimación de distancias, actividades básicas para el ser humano, está dado principalmente por el sentido de la vista, considerado en conjunto como la visión. El fenómeno de la visión es un proceso complejo, que consiste, a grandes rasgos, en captar mediante los ojos la energía procedente de una fuente emisora, transmisora o reflectora de radiaciones luminosas. La energía puede ser cuantificada en esta etapa mediante el uso de la óptica y la fotometría. Esta energía luminosa es convertida en señales neurobiológicas desde la retina hasta el córtex y termina con la interpretación y la comprensión de la información, es decir, la percepción del mundo que nos rodea.

El ojo es un órgano que captura la luz y la transforma en un impulso neuronal que se transmite al cerebro para su procesamiento. La luz, tras atravesar un lente llamado cristalino, incide en una capa de células receptoras llamada retina, que está situada en la parte anterior del ojo. Estas células en presencia de luz generan impulsos neuronales que se envían al cerebro mediante el nervio óptico (ER, 2000; Goldstein, 1992).

El cerebro procesa la información que recibe y genera sensaciones, en lo que se conoce como el proceso de percepción visual. A continuación se describe la percepción acromática (sólo teniendo en cuenta el brillo, es decir la cantidad de energía), y después la percepción cromática (que tiene en cuenta el color, esto es, la forma de distribuirse la energía en distintas longitudes de onda).

2.2.1 Percepción acromática

La percepción del brillo de una imagen la realizan en el ojo los bastones (Figura 2.2). Los bastones son unas células especializadas que tenemos en la retina, en un número superior a 100 millones, que son capaces de detectar y medir el brillo de los haces luminosos que les llegan (Goldstein, 1992; Kaufman, 2002). La sensación de brillo está relacionada con dos fenómenos (ER, 2000; Kaufman, 2002):

- La sensibilidad a la intensidad.
- La inhibición lateral.

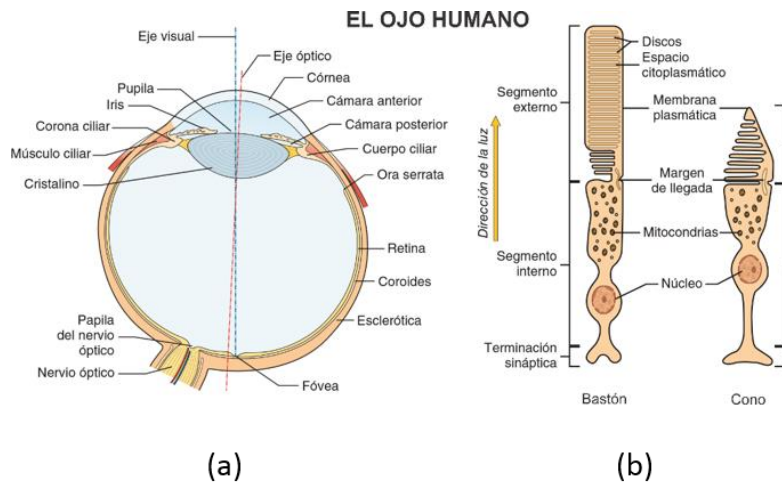


Figura 2.2 Representación del Ojo Humano. (a) sección del ojo humano. (b) visión esquemática de las células fotorreceptoras

Sensibilidad a la intensidad luminosa y el contraste

La sensibilidad a la intensidad, es lo que dota al ojo de la capacidad de distinguir un nivel de intensidad de otro. La diferencia de intensidad se denomina contraste. Se dice que una escena tiene gran contraste si las diferencias de intensidad que contiene son pronunciadas.

La sensibilidad a la intensidad en el ser humano es alta siempre que los elementos que se comparan son pocos. Cuando el número de intensidades involucradas simultáneamente es

superior a unos 24 tonos se pierde la mayor parte de esta sensibilidad. Esto implica que, en la mayoría de casos prácticos, sea suficiente el uso de 32 ó 64 niveles de intensidad para representar una imagen (Kaufman, 2002).

Los seres humanos son capaces de distinguir un rango muy amplio de intensidades. Sin embargo la relación entre la intensidad real de la luz reflejada por un pigmento y la intensidad percibida por un humano no es lineal (Goldstein, 1992).

Inhibición lateral

El otro fenómeno presente en la percepción acromática es la inhibición lateral, se origina en el hecho de que las células de la retina, al detectar un nivel de intensidad, inhiben las células vecinas, produciendo perturbaciones en las fronteras de cambio de intensidad (Kaufman, 2002).

2.2.2 Percepción cromática

La percepción del color de una imagen la realizan los conos (Figura 2.2). Son unas células especializadas, dispuestas en la retina en un número cercano a los 6 millones, que son capaces de variar su comportamiento ante cambios en la longitud de onda de una radiación electromagnética (Kaufman, 2002; Goldstein, 1992). Basándose en la información aportada por los conos el cerebro construye la sensación de color.

Los conos del ojo humano tienen una sensibilidad menor que los bastones, es así que con poca luz sólo los bastones captan suficiente energía para activarse.

Estudios fisiológicos han revelado que existen tres tipos de conos, que se han denominado mediante las letras S, L, y M. Los conos de tipo S (short) son más sensibles a las radiaciones con longitud de onda corta (azules), los M (medium) a las radiaciones de longitud media (verdes), y los L (large) a las de longitud larga (rojos). Así, la sensación de color que percibimos está relacionada con la energía que tiene a diferentes longitudes de onda una radiación electromagnética (Goldstein, 1992).

2.3 Visión Artificial

La visión artificial tiene como finalidad la extracción de información del mundo físico a partir de imágenes, la información adquirida es procesada utilizando sistemas basados en computadoras, los cuales ejecutan algoritmos complejos que determinan y extraen características relevantes de la escena y generan una respuesta como salida del procesamiento.

2.3.1 Representación de la realidad

Un sistema de Visión Artificial actúa sobre una representación de una realidad que le proporciona información sobre brillo, colores, formas, etcétera. Estas representaciones suelen estar en forma de imágenes estáticas, escenas tridimensionales o en movimiento.

Imágenes

Una imagen bidimensional es una función que a cada par de coordenadas (x, y) asocia un valor relativo a alguna propiedad del punto que representa (por ejemplo su brillo o su matiz). Una imagen acromática, sin información de color, en la que a cada punto se le asocia información relativa al brillo, se puede representar como una superficie, en la cual la altura de cada punto indica su nivel de brillo (Figura 2.3). Una imagen en color RGB¹ se puede representar asociando a cada punto un conjunto de valores que indica la intensidad de tres componentes (una roja, otra verde y otra azul). (Figura 2.3) (M. Sonka, 1999; Castleman, 1996).



Figura 2.3 Representación acromática y cromática de una imagen

¹ RGB (en inglés Red, Green, Blue, en español rojo, verde y azul) es la composición del color en términos de la intensidad de los colores primarios de la luz presentes en una imagen.

Escenas 3D

Otra forma de representar la realidad consiste en asignar a cada punto del espacio que pertenece a un objeto (x, y, z) una propiedad del punto (su existencia, su intensidad, su matiz, etcétera.). Al trabajar con imágenes 3D², como se tiene la forma de los objetos, la información de brillo y color puede no ser tan relevante (Verri, 1998).

Secuencias animadas

Un punto con un brillo suficiente que parpadee con una frecuencia superior a 25 pulsos por segundo será percibido como un punto brillante fijo por nuestros sentidos (M. Sonka, 1999). Éste efecto es el usado en cine y televisión para crear la ilusión del movimiento. Así, cuando una cámara cinematográfica toma sucesiones de imágenes estáticas que se capturan a una frecuencia determinada, si estas sucesiones de imágenes se presentan luego a una frecuencia superior a 25 imágenes por segundo, el sistema visual humano no es capaz de distinguir el cambio e interpreta movimiento.

2.3.2 Elementos de un sistema de visión

Se ha visto que el ser humano captura la luz a través de los ojos, y que esta información circula a través del nervio óptico hasta el cerebro donde se procesa. El primer paso de este procesado consiste en encontrar elementos más simples en que descomponer la imagen (como segmentos y arcos). Después el cerebro interpreta la escena y actúa en consecuencia.

2.3.3 Etapas de un sistema de visión

La visión artificial, en un intento de reproducir el comportamiento del sistema de percepción visual del ser humano de manera artificial, este proceso comprende 4 fases

² 3D hace referencia a la geometría, análisis matemático tridimensional o la presentación de objetos tridimensionales realizados por gráficos de computadora.

principales (Castleman, 1996; Umbaugh, 1998; Chen, 2010), como se observa en la Figura 2.4 y se describen a continuación:

La primera fase, que es sensorial, consiste en la captura o adquisición de las imágenes digitales mediante algún tipo de sensor.

La segunda etapa consiste en el tratamiento digital de las imágenes, con objeto de facilitar las etapas posteriores. En esta etapa de procesamiento previo es donde, mediante filtros y transformaciones geométricas, se eliminan partes indeseables de la imagen o se realzan partes interesantes de la misma.

La siguiente fase se conoce como segmentación, y consiste en aislar los elementos que interesan de una escena para comprenderla.

Por último se llega a la etapa de reconocimiento o clasificación. En ella se pretende distinguir los objetos segmentados, gracias al análisis de ciertas características que se establecen previamente para diferenciarlos.

Estas cuatro fases no se siguen siempre de manera secuencial, sino que en ocasiones deben realimentarse hacia atrás. Así, es normal volver a la etapa de segmentación si falla la etapa de reconocimiento, o a la de preproceso, o incluso a la de captura, cuando falla alguna de las siguientes.

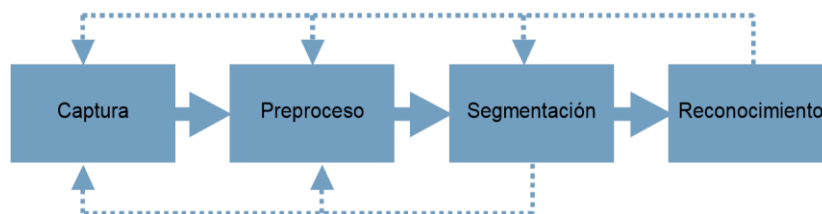


Figura 2.4 Etapas de un sistema de visión

2.3.4 Arquitectura de un sistema de visión

Un sistema típico de visión por computadora, además de un dispositivo de captura, cuenta con al menos otros 4 elementos: un dispositivo de conversión de analógico a digital (A/D),

una memoria de video, un elemento de procesamiento y un monitor (Davies, 1997; Szeliski, 2010; Sonka, Hlavac, & Boyle, 1998). En la Figura 2.5 se muestra la arquitectura básica de un sistema de visión. A continuación se describen los principales elementos:

Dispositivo de captura: Dispositivo físico que es sensible a una determinada banda del espectro electromagnético. El dispositivo produce una señal eléctrica proporcional al nivel de energía detectado.

Conversión A/D: Convierte la señal obtenida del dispositivo de captura en una señal digital.

Memoria de video: Memoria semiconductora (RAM³) en la que se almacena la imagen digitalizada. Normalmente la conversión A/D y la memoria de video se agrupan en un módulo conocido como *frame grabber*⁴ (captura de imágenes).

Procesador: La memoria de video se acopla a un procesador de propósito general que permite operar sobre la imagen. Opcionalmente puede existir otro procesador dedicado para captura y procesamiento de imágenes.

Monitor: Generalmente se tiene un monitor que permita visualizar las imágenes adquiridas.

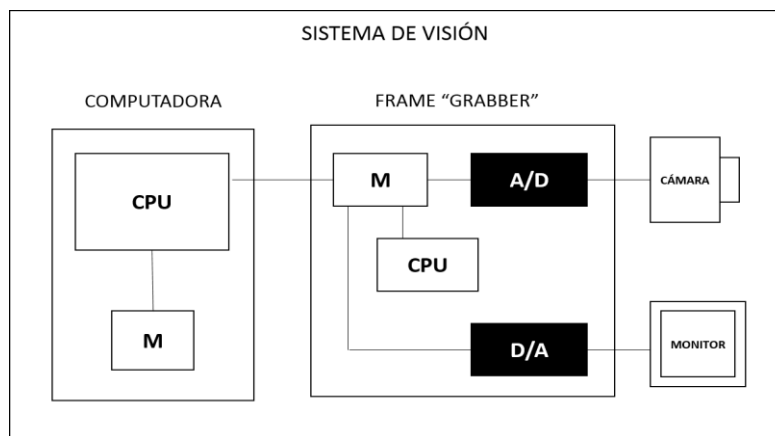


Figura 2.5 Sistema de Visión

³ La memoria de acceso aleatorio (en inglés: random-access memory) se utiliza como memoria de trabajo para el sistema operativo, los programas y la mayoría del software computacional.

⁴ Los frame grabbers han sido utilizados desde hace mucho tiempo para poder capturar imágenes a una alta velocidad.

2.3.5 Dispositivos para visión

Existen diferentes dispositivos para la captura de imágenes. Dichas imágenes son digitalizadas y almacenadas en la memoria de la computadora. Una vez en la computadora, o en ocasiones desde el mismo dispositivo de captura, la imagen puede ser ya procesada.

Para la adquisición de la imagen se requiere de un dispositivo físico que sea sensible a una determinada banda del espectro electromagnético. El dispositivo produce una señal eléctrica proporcional al nivel de energía detectado, la cual es posteriormente digitalizada (B. Jahne, 1999; Chen, 2010). Entre los dispositivos de captura o sensores se encuentran:

- Cámaras fotográficas (CCD⁵)
- Digitalizadores (scanners)
- Sensores de rango (franjas de luz, laser)
- Sensores de ultrasonido (sonares)
- Rayos X
- Imágenes de tomografía
- Imágenes de resonancia magnética

2.3.6 Adquisición y representación de imágenes

Una vez digitalizada una imagen bidimensional digital está constituida por un conjunto de elementos llamados píxeles⁶. Cada píxel ofrece cierta información sobre una región elemental de la imagen. En imágenes en niveles de gris esta información es el brillo. En imágenes en color, la información corresponde a la intensidad de cada una de las componentes de una base de color (por ejemplo RGB) (Sonka, Hlavac, & Boyle, 1998; Davies, 1997).

⁵ Un charge-coupled device o CCD (en español “dispositivo de carga acoplada”) es un circuito integrado que contiene un número determinado de condensadores enlazados o acoplados, diseñado para la captura de una imagen digital.

⁶ Un pixel (acrónimo del inglés picture element, “elemento de imagen”) es la menor unidad homogénea en color que forma parte de una imagen digital, ya sea esta una fotografía, un fotograma de vídeo o un gráfico.

2.3.6.1 Dispositivos de captura

La cámara constituye uno de los elementos más importantes en cualquier sistema de comunicación visual. Su función básica es convertir la escena tridimensional en una señal temporal, analógica o digital.

2.3.6.1.1 Cámara fotográfica analógica

La Figura 2.6, muestra los elementos básicos de una cámara fotográfica analógica; básicamente, una cámara fotográfica está constituida por un espacio oscuro (la cámara), en la que se monta un objetivo. El objetivo está compuesto por un conjunto de lentes que tiene el propósito de comportarse como una única lente que forma la imagen luminosa en el interior de la cámara, donde hay una superficie sensible a la luz llamada película. Entre el objetivo y la superficie sensible se encuentra el obturador, que sólo deja pasar la luz en el momento de captura de la imagen. Este momento lo determina el fotógrafo presionando el disparador, para permitir al usuario encuadrar el objeto el dispositivo dispone de un visor (Jackson & Townsend, 1991).

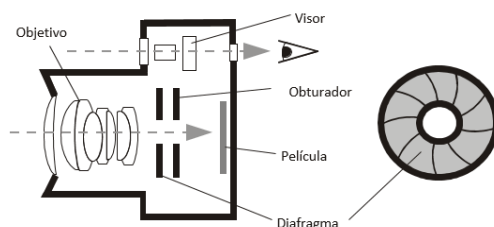


Figura 2.6 Elementos de una cámara fotográfica analógica

Normalmente las cámaras fotográficas permiten variar la distancia a la que se encuentra la lente ideal del plano de formación de la imagen (donde está la película). Este proceso permite el enfoque, es decir el ajuste de la definición de la imagen de los objetos que se encuentren a una distancia determinada de la cámara.

Una cámara también suele permitir variar la cantidad de luz que entra en ella mediante un dispositivo conocido como diafragma. Cuando se abre el diafragma, ingresan suficientes rayos de luz por cada punto de la escena, lo que permite que los elementos que estén a cierta distancia de la cámara aparezcan enfocados. Por el contrario, cuanto menor es la apertura del

diafragma menos rayos de luz entran por cada punto de la escena. Cuando la apertura es pequeña el tiempo que debe dejarse abierto el obturador (tiempo de exposición) debe ser grande, pues en otro caso no entraría suficiente luz como para impresionar la película. Sin embargo, mantener mucho tiempo abierto el obturador puede ocasionar que la imagen resultante aparezca borrosa si durante este tiempo los objetos de la escena se mueven respecto a la cámara.

2.3.6.1.2 Cámara fotográfica digital

El esquema de una cámara fotográfica digital es idéntico al de su correspondiente analógica, con la diferencia de que el dispositivo sensible es un componente electrónico constituido por una matriz de elementos fotosensibles denominado CCD (Coupled Charge Device), que se sitúa en el mismo lugar que el plano de formación de la imagen, de manera que se forma la imagen sobre él (Figura 2.7).

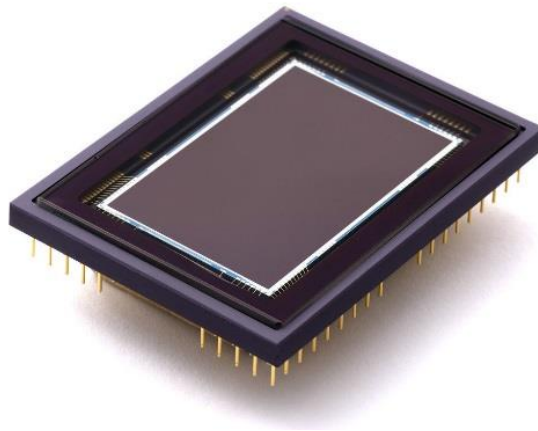


Figura 2.7 Sensor CCD (Coupled Charge Device)

El funcionamiento de estos elementos fotosensibles tiene su base en el efecto fotoeléctrico. Consisten en materiales que de manera espontánea emiten electrones en una cantidad proporcional al número de fotones que reciben. El número de elementos fotosensibles, junto con el área que ocupan, definen la resolución espacial del dispositivo. El primer CCD comercial, constaba de 120.000 elementos y tenía un tamaño de 0'5x0'25 pulgadas (Sawyer) (Figura 2.8).

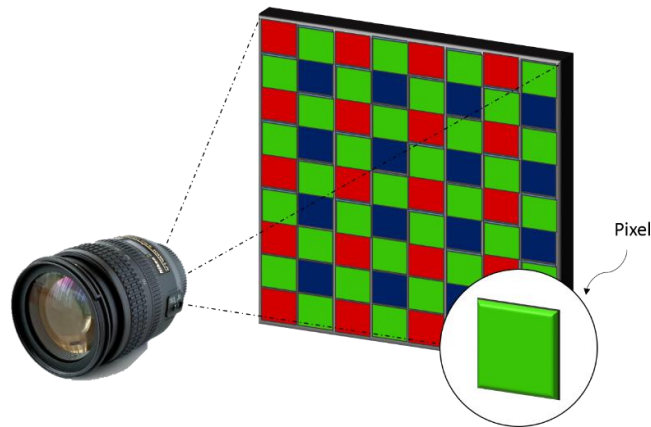


Figura 2.8 Sensor CCD/CMOS

Por otro lado, cada uno de estos elementos fotosensibles es capaz de obtener una carga eléctrica proporcional a la intensidad de la luz que le incide. Después, la carga eléctrica de cada elemento se transmite a un amplificador eléctrico. El tiempo que tarda esta operación determina el número de imágenes por segundo que puede tomar el dispositivo.

Así, desde el punto de vista de la resolución espacial estas matrices son dispositivos digitales, mientras que desde el de la resolución radiométrica pueden considerarse dispositivos analógicos. Aunque finalmente, esta información de carga analógica es discretizada mediante un conversor analógico digital, fijándose en este punto la resolución radiométrica. Originalmente estos dispositivos registran únicamente la intensidad de luz incidente. Una solución para conseguir una imagen en color consiste en cubrir la retícula de celdas con filtros que sólo permitan el paso de cada una de las componentes RGB denominado filtro *Bayer* (Sawyer) (Figura 2.9).

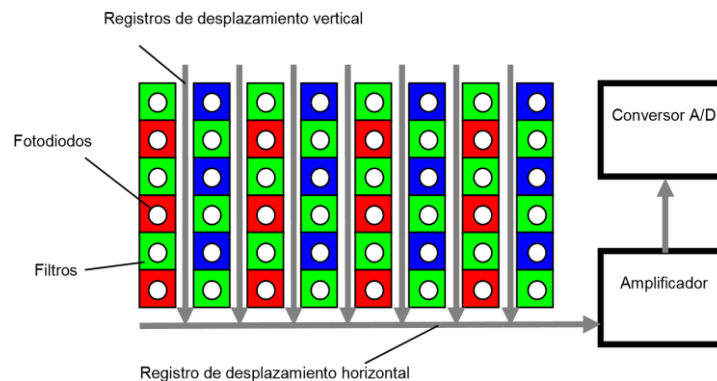


Figura 2.9 Esquema de un CCD que utiliza un filtro Bayer

Actualmente existen dos tecnologías para estos dispositivos: CMOS⁷ y CCD. La tecnología CMOS es más barata al basarse en elementos semiconductores clásicos. Sin embargo la resolución y la respuesta conseguida con la tecnología CCD ha sido hasta ahora mejor.

2.3.6.1.3 Cámaras industriales y de visión

Se engloban dentro de este apartado todas las cámaras que se utilizan en aplicaciones de reconocimiento automático de formas y visión por computador. Las características de las cámaras dependen del tipo de aplicación y pueden encontrarse tipos muy distintos. Normalmente la cámara está conectada a una tarjeta de digitalización de imagen que puede realizar la adquisición de la secuencia de vídeo en tiempo real o adquirir un único fotograma cada vez que se produzcan las condiciones de la imagen que debe reconocerse. Este segundo caso es el más extendido en la mayoría de aplicaciones, utilizándose tarjetas de digitalización que se denominan capturadoras de imagen (*frame grabber*), que generalmente aceptan distintos formatos de señal de cámara y que disponen de puertos para controlar algunos de los parámetros de la cámara.

En las aplicaciones más simples pueden utilizarse cámaras de vídeo vigilancia que proporcionan una señal PAL⁸ estándar en color o, en la mayoría de los casos, en blanco y negro. La adquisición de la imagen se realiza cuando se producen las condiciones en las que se requiere realizar un reconocimiento de la imagen. Generalmente estas condiciones son controladas por circuitos externos que producen una señal de inicio de captura. Una vez la imagen ha sido digitalizada, se realiza la identificación o reconocimiento de los objetos produciendo señales externas de control y quedando el sistema en condiciones de capturar una nueva imagen.

Aparte de las cámaras convencionales, las más utilizadas son las progresivas, las lineales y las denominadas cámaras inteligentes (Sawyer; Chen, 2010). En cualquier caso, es posible que la señal proporcionada sea analógica o directamente digital. Las señales digitales suelen

⁷ Complementary metal-oxide-semiconductor o CMOS (semiconductor complementario de óxido metálico) es una de las familias lógicas empleadas en la fabricación de circuitos integrados.

⁸ PAL es la sigla de Phase Alternating Line (línea de fase alternada). Es el nombre con el que se designa al sistema de codificación utilizado en la transmisión de señales de televisión analógica en color en la mayor parte del mundo.

entregarse mediante un cable específico en el que todos los bits del elemento de imagen se suministran en paralelo, además de existir versiones que utilizan protocolos serie estándar.

Las cámaras progresivas proporcionan todas las líneas que constituyen la imagen en un único campo (Figura 2.10). Con ello, aumenta la calidad y resolución de la imagen, sobre todo cuando se realiza la captura de objetos móviles. Existen modelos de alta definición que pueden entregar imágenes de hasta 2000 líneas. Generalmente, estas cámaras pueden controlarse mediante una interfaz digital en la que es posible variar el tiempo de exposición, la apertura del diafragma, el enfoque y el ángulo de visión. Como generalmente no es necesario que la cámara proporcione señales de vídeo en tiempo real, también es posible controlar el instante en el que se realiza la captura de la imagen en el sensor CCD. Asimismo, el tiempo de exposición puede superar los 1/50 segundos de las cámaras convencionales, ya que la frecuencia de imagen puede controlarse externamente e incluso, trabajar basándose exclusivamente en fotogramas. En estos modelos, el tiempo mínimo entre dos fotogramas viene determinado por el tiempo de transferencia de la imagen desde la cámara a la tarjeta digitalizadora.



Figura 2.10 Cámara Progresiva

Las cámaras lineales están basadas en una CCD de línea en vez de una CCD matricial (Figura 2.11). La ventaja principal de este tipo de cámaras es que pueden obtener un gran número de elementos de imagen por línea (hasta 4000 píxeles). No obstante, la captura de la imagen debe realizarse línea a línea, por lo que es preciso que exista un desplazamiento relativo entre la cámara y el objeto que se está capturando. La velocidad de desplazamiento entre ellos determina la resolución de la imagen en el sentido ortogonal a la línea de CCD.

Es importante que esta velocidad sea lineal para reducir las posibles distorsiones geométricas de imagen que obtendríamos si el movimiento relativo no es uniforme. Generalmente, el movimiento del objeto se realiza con sistemas de posicionamiento mecánicos de gran precisión que son controlados directamente por el sistema encargado del reconocimiento de imagen. Los formatos de señal que se obtienen en la salida de la cámara pueden ser analógicos o digitales, aunque estos últimos son los que más predominan en los modelos actuales. La tasa de transmisión de bits entre la cámara y la tarjeta de digitalización depende del modelo de cámara, el número de píxeles, si es en color o en blanco y negro, etc. Las tarjetas de digitalización suelen poder configurarse para adaptarse a las características de las cámaras.



Figura 2.11 Cámara Lineal

Algunas cámaras incorporan la posibilidad de realizar tratamiento de datos interno, por lo que se denominan cámaras inteligentes (Figura 2.12). Entre las operaciones que pueden realizar estas cámaras destaca la estimación de gradiente, el ajuste digital de contraste o la transformada de Fourier de la imagen. La ventaja de estos sistemas es que proporcionan una información previamente tratada, por lo que simplifican la tarea del procesador durante el reconocimiento de formas. En algunos casos el tratamiento puede realizarse directamente por métodos ópticos (transformada de Fourier) utilizando lentes especialmente diseñadas para que proporcionen este tipo de información. En general se trata de cámaras de precio muy elevado cuyo uso sólo se ha popularizado en aplicaciones muy concretas.



Figura 2.12 Cámaras Inteligentes

2.3.6.2 Representación de una imagen digital

Aunque el sistema de visión humano tiene mayor resolución en la fovea y menos en la periferia, se ha observado que a pesar de que la distribución de los fotorreceptores no es uniforme, la percepción visual sí lo es (Goldstein, 1992; Kaufman, 2002). Los humanos percibimos con una única resolución. Estas circunstancias han conducido a la utilización de sensores con matrices de resolución uniforme. Por tanto, la organización corresponde a una matriz 2D⁹ uniforme.

Las imágenes para ser procesadas en una computadora deben ser adquiridas a través una cámara de vídeo y puestas en su memoria empleando las tarjetas de adquisición de vídeo. Esta señal es de carácter bidimensional y emplea variables discretas. Los elementos de la matriz se llaman píxeles. El acceso a esta elemental información se hace indicando la fila y la columna que ocupa. El origen de coordenadas de la imagen se encuentra en la esquina superior izquierda. El eje horizontal corresponde con las columnas y el eje vertical con las filas. Se emplearán índices enteros para posicionar el píxel. Se denotará el valor del píxel a través de una función, del tipo $f(x,y)$, siendo x el índice de la fila e y de la columna (Umbaugh, 1998; Ruíz, 2012).

Si la imagen es acromática, sólo se presenta la luminancia, esto es, los niveles de grises. La función $f(x,y)$ retornará el nivel de gris del píxel mencionado. En caso de que la imagen sea en color, $f(x,y)$ devolverá un vector. Normalmente suele expresarse como una

⁹ 2D. Hace referencia a un plano bidimensional como la generación de imágenes digitales por computadora bidimensionales (modelos geométricos, texto e imágenes digitales 2D).

proyección del color sobre el sistema RGB (Red-Green-Blue) (Figura 2.13) (B. Jahne, 1999; Ruíz, 2012).

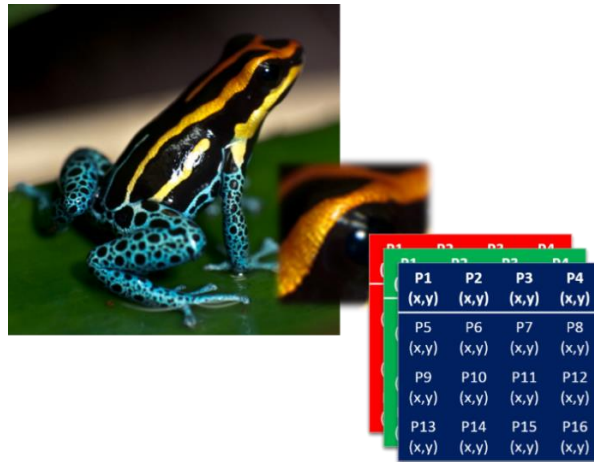


Figura 2.13 Organización matricial de una imagen digital

La imagen puede ser de tipo 3D, por ejemplo, en resonancia magnética, luz estructurada, etc. Éstas se presentan como una pila de imágenes 2D. Se les añade otro índice, denominado k o z , que indica el orden de la rodaja de imagen 2D dentro de la pila (Verri, 1998).

La secuencia temporal de imágenes estáticas da lugar al vídeo. En el cine se emplea 25 fotogramas por segundo, gracias a la inercia visual del ojo humano, da sensación de continuidad en la escena (Jackson & Townsend, 1991).

2.3.6.3 Representación de la imagen y estructura de datos

Las imágenes digitales suelen almacenarse en los ordenadores en forma de ficheros o archivos, para su posterior uso en el procesamiento de imágenes. Se analizarán las estructuras que se usan a tal efecto, los métodos utilizados para optimizar el espacio requerido y algunos de los diferentes formatos estándar (TIFF, GIF, BMP, JPG...) (Gibson & Berger, 1998; Netravali & Haskell, 1995; Solari, 1997).

2.3.6.4 Estructura del fichero de la imagen

Generalmente una imagen almacenada en un ordenador está constituida por un mapa de bits (píxeles), precedido por una cabecera que describe sus características (Gibson & Berger,

1998; Solari, 1997; Netravali & Haskell, 1995) (tamaño de la imagen, modo de color, paleta, resolución de la imagen, etc.) (Figura 2.14). Frecuentemente, cuando la imagen se encuentra en la memoria principal del ordenador la cabecera y el mapa de bits no están contiguos.

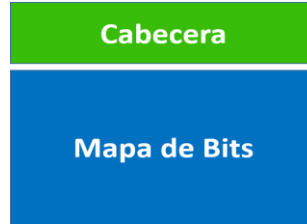


Figura 2.14 Esquema del fichero de una imagen

2.3.6.5 Formatos comerciales de representación

Existen multitud de formatos de ficheros de imágenes de tipo mapa de bits. Se puede hablar de ficheros tipo BMP, TIFF, GIF, JFIF, PGM, entre otros, Cada uno ofrece ciertas ventajas que otros formatos pueden no contemplar. La Tabla 2.1 muestra las principales características de algunos de estos formatos.

Tabla 2.1 Formatos para ficheros gráficos y características principales

| Formato | Color Real | Paleta | Grises | Bi-tonal | Compresión | Origen | Multi-Imagen |
|----------------|------------|--------|--------|----------|--|----------|--------------|
| Bitmap | SI | SI | SI | SI | Run-Length | Windows | NO |
| TIFF | SI | SI | SI | SI | JPG, LZW, Runs, CCITT4, CCITT3, PackBits | Estándar | SI |
| JFIF | SI | NO | SI | NO | JPEG | Estándar | NO |
| JPG2000 | SI | NO | SI | NO | JPEG 2000 | Estándar | NO |
| PCX | NO | SI | NO | NO | Propia | Windows | NO |
| PGM | NO | NO | SI | NO | NO | Unix | NO |
| GIF | NO | SI | SI | SI | LZW | Estándar | SI |

A continuación se muestran algunos de los formatos más comunes y las posibilidades que brinda cada uno (Netravali & Haskell, 1995; Solari, 1997):

BMP.- El formato BMP (Bit Map) es el formato de las imágenes en bitmap de Windows. Aunque muy extendido, tiene la dificultad de la escasa compresión que realiza en los archivos por lo que ocupan mucho espacio.

TIFF.- El formato TIFF, que corresponde a las siglas Tagged-Image File Format, se utiliza cuando se van a realizar impresiones en papel de la imagen. Es un formato que admite una compresión muy baja, por lo que la pérdida en la calidad de imagen es prácticamente nula. Se conoce como formato de compresión sin pérdida. Una desventaja es que los archivos .tiff son de gran tamaño.

GIF.- Estas tres letras, representan las siglas de "Graphics Interchange Format". La tecnología del GIF fue desarrollada por la empresa CompuServe, destacando en sus características la posibilidad de trabajar, con un máximo de 256 colores y con más de una imagen, el GIF es un formato ideal para utilizar en la web en imágenes pequeñas o de pocos colores y no es recomendable para utilizar en impresión, ya que la calidad es limitada.

JPEG o JPG. - El formato JPEG (Joint Photographic Experts Group) está diseñado para realizar compresión de imágenes, permitiendo reducir la cantidad de información de las mismas, con una consecuente reducción de peso del archivo final. La resolución de este formato es baja y no es recomendable para la impresión.

2.3.7 Procesamiento de la imagen

Cuando se adquiere una imagen mediante cualquier sistema de captura, por lo general ésta no se utiliza directamente. La aparición de variaciones en intensidad debidas al ruido, por deficiencias en la iluminación o la obtención de imágenes de bajo contraste hace necesario un pre procesamiento de la imagen con el objetivo fundamental de corregir estos problemas, además de aplicar aquellas transformaciones a la imagen que acentúen las características que se deseen extraer, de manera que se facilite las operaciones de las etapas posteriores.

2.3.7.1 Histograma de una imagen

La intensidad de un histograma muestra como los niveles de brillo están ocupados en la imagen; el contraste de la imagen se mide con el rango de brillo en la imagen. El histograma grafica el número de pixeles con un nivel de brillo particular en contra del nivel de brillo. Otra de las características que se pueden detectar con el histograma es la existencia de ruido en la imagen, esto mediante el conocimiento del histograma ideal. La eliminación de este ruido puede mejorar la calidad de la imagen, lo que en consecuencia facilitara que las técnicas

que se le apliquen a dicha imagen puedan realizarse de manera más fácil y con mejores resultados.

El histograma de una imagen es una función discreta que representa el número de píxeles en la imagen en función de los niveles de intensidad g . La probabilidad $P(g)$ de ocurrencia de un determinado nivel g se define como (Pajares Martinsanz & de la Cruz García, 2008; Cuevas & Zaldivar, 2004; Parker, 2010):

$$P(g) = \frac{N(g)}{M} \quad (2.1)$$

donde M es el número de píxeles en la imagen y $N(g)$ es el número de píxeles en el nivel de intensidad g . Como con cualquier distribución de probabilidad todos los valores de $P(g)$ son menor o iguales que 1 y la suma de todos los valores de $P(g)$ es 1 (M. Sonka, 1999; Parker, 2010; Cuevas & Zaldivar, 2004). En la Figura 2.15 se muestra el resultado del obtener el histograma de una imagen.

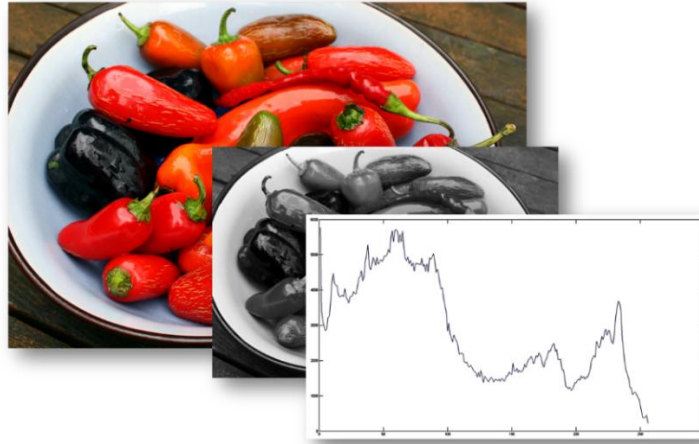


Figura 2.15 Histograma de una imagen

2.3.7.1.1 Propiedades estadísticas del histograma

En la representación del histograma de una imagen, es posible determinar diversas propiedades estadísticas que dan información sobre la distribución de los niveles de gris (Pajares Martinsanz & de la Cruz García, 2008; Ruíz, 2012; Sucar & Gómez, 2010):

Media: es el valor medio de los niveles de gris y nos informa sobre el brillo general de la imagen, está definida por.

$$\bar{g} = \sum_{g=0}^{L-1} gP(g) = \sum_i \sum_j \frac{I(i,j)}{M} \quad (2.2)$$

L es el número total de niveles de gris, así para una imagen con valores de gris entre 0 y 255: L sería 256. Una imagen brillante tendría una media alta y viceversa

Varianza: mide la dispersión de los alrededores de la media, está definida por

$$\bar{\sigma} = \sum_{g=0}^{L-1} (g - \bar{g})^2 P(g) \quad (2.3)$$

Una varianza alta corresponde a una imagen con contraste alto y viceversa.

Asimetría: sobre la media en la distribución de los niveles de gris,

$$a = \sum_{g=0}^{L-1} (g - \bar{g})^3 P(g) \quad (2.4)$$

Un valor absoluto alto de a indica una gran asimetría y un valor bajo representa una simetría.

Energía: nos indica sobre la distribución de los niveles de gris en la imagen y está dada por la siguiente ecuación:

$$E = \sum_{g=0}^{L-1} (P(g))^2 \quad (2.5)$$

La energía tiene valor máximo 1 para una imagen con un único nivel de gris y disminuye a medida que aumenta el número de niveles de gris, dado todos los valores de $P(g) \leq 1$.

2.3.7.1.2 Ecualización del histograma

La ecualización del histograma consiste en una expansión del histograma de la imagen, dotando al mismo de mayor linealidad y haciendo que éste ocupe el ancho del espectro de tonalidades grises por completo, ocasionando las siguientes mejoras en la imagen (R. González, 1996; BRANCH & OLAGUE, 2001):

- *Una mayor utilización de los recursos disponibles:* al ecualizar el histograma, los tonos que antes estaban más agrupados, ahora se separan, ocupando todo el rango de grises, por lo que la imagen se enriquece al tener niveles de gris más distintos entre sí, mejorando, por tanto, la apariencia visual de la imagen.
- *Un aumento del contraste:* esta ventaja es derivada del punto anterior, debido a que si hacemos que el histograma de la imagen ocupe todo el rango de grises, estamos aumentando la distancia entre el tono más claro y el más oscuro, convirtiendo a éstos, en blanco y negro y consecuentemente aumentando el contraste de la imagen.
- *Regulación óptima y automática del contraste de la imagen:* De esta manera son evitados los ajustes manuales con los que no se consigue un equilibrio óptimo entre el blanco y el negro en la representación de una imagen.

A su vez, aparecen algunos inconvenientes que surgen a la hora de ecualizar la imagen, algunos de ellos se detallan a continuación:

- *Pérdida de información:* puede ocurrir que a algunos píxeles que en la imagen original tenían distintos niveles de gris, se les asigne tras la ecualización global, el mismo nivel de gris. Por otro lado, hay casos en los que dos niveles de gris muy próximos se separen, dejando huecos en el histograma.
- *Error indeseado:* En ocasiones, las bandas horizontales, consecuencia de una deficiente digitalización pueden resultar intensificadas, resaltando aún más este error.

Suponiendo que se dispone de una imagen de 8 bits. Partiendo del histograma original, el primer paso para trabajar con el mismo es la normalización, para ello hay que normalizar

tanto el eje horizontal (niveles de gris) como el vertical (aparición del nivel de gris en cuestión), de manera que ambas magnitudes queden comprendidas entre 0 y 1.

Para normalizar el eje horizontal, es decir, el que indica el nivel de gris, dividimos cada magnitud entre 255, con lo que queda: $0, 1/255, 2/255 \dots 255/255=1$, obteniendo un rango de niveles de gris comprendido entre 0 y 1. Para la normalización del eje vertical, se debe dividir cada componente entre el número total de píxeles, así en el caso más extremo, es decir en el que toda la imagen fuese de un mismo color, la representación sería una única barra de altura igual a 1 (Umbaugh, 1998; Cuevas & Zaldivar, 2004; Ruíz, 2012). El siguiente paso consiste en la acumulación del histograma, esto consiste en sustituir cada magnitud, por el valor de ella misma más el total de la suma de las anteriores, con ello se consigue que el histograma (que reflejaba la función "densidad de probabilidad" para cada color) se transforme en una representación de la función de distribución. Quedando una función creciente, cuyo valor máximo será siempre 1.

Aplicando la siguiente ecuación, se obtiene los nuevos niveles de gris (haciendo que el histograma se extienda hasta el negro), además se consigue que éstos vuelvan a encontrarse entre 0 y 255:

$$S^* = \text{Int} \left[\frac{S - S_{\min}}{1 - S_{\min}} 255 + 0.5 \right] \quad (2.6)$$

donde *Int* representa la operación de tomar el entero más cercano por defecto, *S_{min}* representa el menor valor de *S* distinto de cero; en este caso, 255 representa el número de niveles de gris menor uno.

Suponiendo que se dispone de una imagen de 9 píxeles cuyos niveles de gris son los mostrados en la Tabla 2.2.

Tabla 2.2 Píxeles de imagen en escala de gris

| | | |
|-----|-----|-----|
| 253 | 253 | 255 |
| 253 | 255 | 254 |
| 253 | 254 | 254 |

2.3.7.2 Imagen RGB

Una imagen bidimensional es una región en el plano en el cual un brillo en cada punto es definido como una función $f(x, y)$ donde x y y son las coordenadas de dicho punto en el plano. La amplitud de f en cualquier par de coordenadas se conoce como la *intensidad* de la imagen en un punto (Umbaugh, 1998; Jähne & HauBecker, 2000; Szeliski, 2010). Cuando los valores de las coordenadas y de la intensidad son cantidades discretas finitas, la imagen se considera una *imagen digital*. Los elementos que constituyen una imagen digital se les conocen como *píxeles*.

Una imagen RGB se define como un arreglo de $3 \times M \times N$ píxeles donde cada pixel corresponde a los componentes rojo, verde y azul de una imagen de color (R-red, G-green, B-blue) (Figura 2.16) (M. Sonka, 1999). El propósito principal del modelo RGB es el sensado, representación y muestra de imágenes en aparatos electrónicos tal como televisores, computadoras, etc.

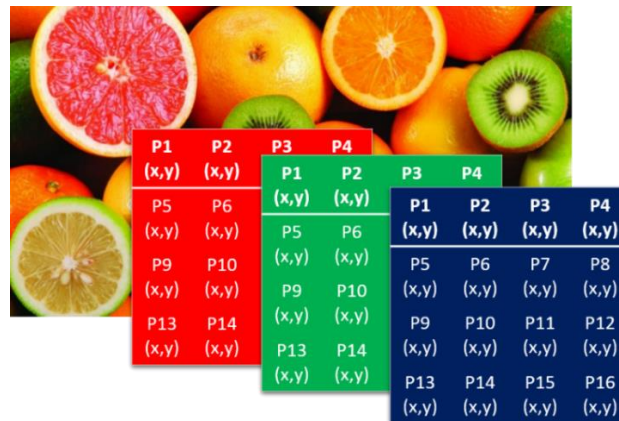


Figura 2.16 Representación matricial de las componentes RGB

2.3.7.3 Escala de gris

Las imágenes digitales generalmente están compuestas por un amplio rango de valores de intensidad. A estas imágenes se les ha denominado imágenes de nivel de gris. Aunque el rango de niveles de gris utilizado para representar la imagen comúnmente es de 256 valores (8 bits por pixel), este es variable y depende de la aplicación. La tendencia natural, debido al

aumento de la potencia computacional y de la calidad de los sensores es aumentar este rango para dotar de mayor fidelidad a la imagen.

Las imágenes en escala de grises o escala monocromática (Figura 2.17), están compuestas por una sola matriz de $M \times N$ valores en donde cada pixel es una sola muestra que contiene la información de la intensidad de la imagen. Las tres características con que se puede definir un color son el tono o matiz (color actual), el valor (luminosidad u oscurecimiento) y la saturación (pureza del color) (R. González, 1996; Jähne & HauBecker, 2000). Por lo tanto la conversión de una imagen de color a una imagen en escala de grises no se realiza de una forma única, sin embargo en su acercamiento más común, se trata de retener la información de la luminosidad y descartar los valores de tono y saturación (Umbaugh, 1998; B. Jahne, 1999).

Asumiendo que los colores, rojo, verde y azul son señales de luminosidad, la aproximación de una imagen en escala de grises a partir de una imagen de color está dada por (2.7) en donde 0 es el valor de menos intensidad, refiriéndose al color negro y 1 es el valor de mayor luminosidad o el color blanco.

$$GRAY = (0.30 * R) + (0.59 * G) + (0.11 * B) \quad (2.7)$$



Figura 2.17 Conversión a nivel de gris

2.3.7.4 Binarización

Las imágenes binarias siempre se obtienen a partir de imágenes de niveles de gris. El proceso de conversión de una imagen de nivel de gris a una imagen formada solo por dos valores o etiquetas (0 para el negro y 1 para el blanco) se conoce como binarización (B. Jahne, 1999; Chen, 2010; Rodríguez, 2012).

Trabajar con imágenes binarias resulta muy interesante dadas las siguientes características:

- Se reduce al mínimo los datos necesarios para representar la imagen y ello permite un máximo aprovechamiento de la potencia computacional.
- Desde el punto de vista computacional, las imágenes binarias se procesan mucho más rápidamente.
- Las propiedades geométricas y topológicas de los objetos presentes en la imagen, en las que se basan un gran número de aplicaciones industriales, puede obtenerse rápida y fácilmente a partir de las imágenes binarias.

La binarización selecciona los píxeles que tienen un valor particular que se encuentren dentro de un rango específico. Esto sirve para encontrar objetos que se encuentren dentro de la imagen si se conoce su nivel de brillo.

2.3.7.4.1 Binarización uniforme

En la binarización uniforme, se buscan los píxeles que se encuentran debajo de un nivel específico, se consideraran como píxeles color blanco y los píxeles que se encuentran por arriba del nivel, se consideran como píxeles color negro. Para utilizar este método se necesita tener conocimiento acerca del nivel de grises que contiene la imagen, pero si el nivel no es conocido, el histograma puede ser de mucha utilidad.

Una forma de aplicar este tipo de binarización es la extracción de los objetos del entorno, estableciendo un nivel T que separe los dos tonos de intensidad. De esta forma, un píxel

(x, y) para el cual $f(x, y) > T$ será un pixel del objeto; en caso contrario, el punto será del entorno (Pajares Martinsanz & de la Cruz García, 2008; Sucar & Gómez, 2010).

Basándonos en lo anterior, podemos considerar la fijación del umbral como una operación que implica un análisis con respecto a una función T de la forma,

$$T = T[x, y, p(x, y), f(x, y)] \quad (2.8)$$

donde $f(x, y)$ es la intensidad en el punto (x, y) y $p(x, y)$ es alguna propiedad local del punto; por ejemplo, la intensidad media de un entorno de una pixel adyacente centrado en (x, y) .

De esta manera se creará una imagen binaria $g(x, y)$ definiendo,

$$g(x, y) = \begin{cases} 0 & \text{si } f(x, y) > T \\ 1 & \text{si } f(x, y) \leq T \end{cases} \quad (2.9)$$

Examinando $g(x, y)$, se puede determinar que los pixeles a los que se les asigna valor 0 corresponden a los objetos, mientras que los correspondientes al entorno tienen el valor 1.

Cuando T depende sólo de $f(x, y)$, el umbral se llama global. Si T depende tanto de $f(x, y)$ como de $p(x, y)$, entonces el umbral se llama local. Si T depende de las coordenadas espaciales x e y , se denomina umbral dinámico.

La Figura 2.18, muestra el resultado de aplicar una binarización a una imagen RGB a partir del ajuste de umbral mediante la obtención del histograma de una imagen en escala de gris.



Figura 2.18 Representación Binaria de una imagen

2.3.7.5 Filtros

El ruido es una información no deseada que contamina la imagen. El ruido aparece en imágenes procedente de una gran variedad de fuentes. El proceso de adquisición de la imagen digital, que convierte una imagen óptica en una señal eléctrica continua que luego es muestreada, es el primer proceso por el cual el ruido aparece en imágenes digitales. En cada paso en el proceso hay fluctuaciones originadas por fenómenos naturales que añaden un valor aleatorio al valor exacto de la intensidad para un determinado pixel. En las imágenes típicas el ruido puede modelarse bien como una distribución *gaussiana (normal)*, *uniforme* o “*sal y pimienta*” (*impulso*) (Pajares Martinsanz & de la Cruz García, 2008; R. González, 1996). En la Figura 2.19 se observa dos posible representaciones de ruido del tipo *Gaussiano* y “*Sal y Pimienta*” contenidos en una imagen.

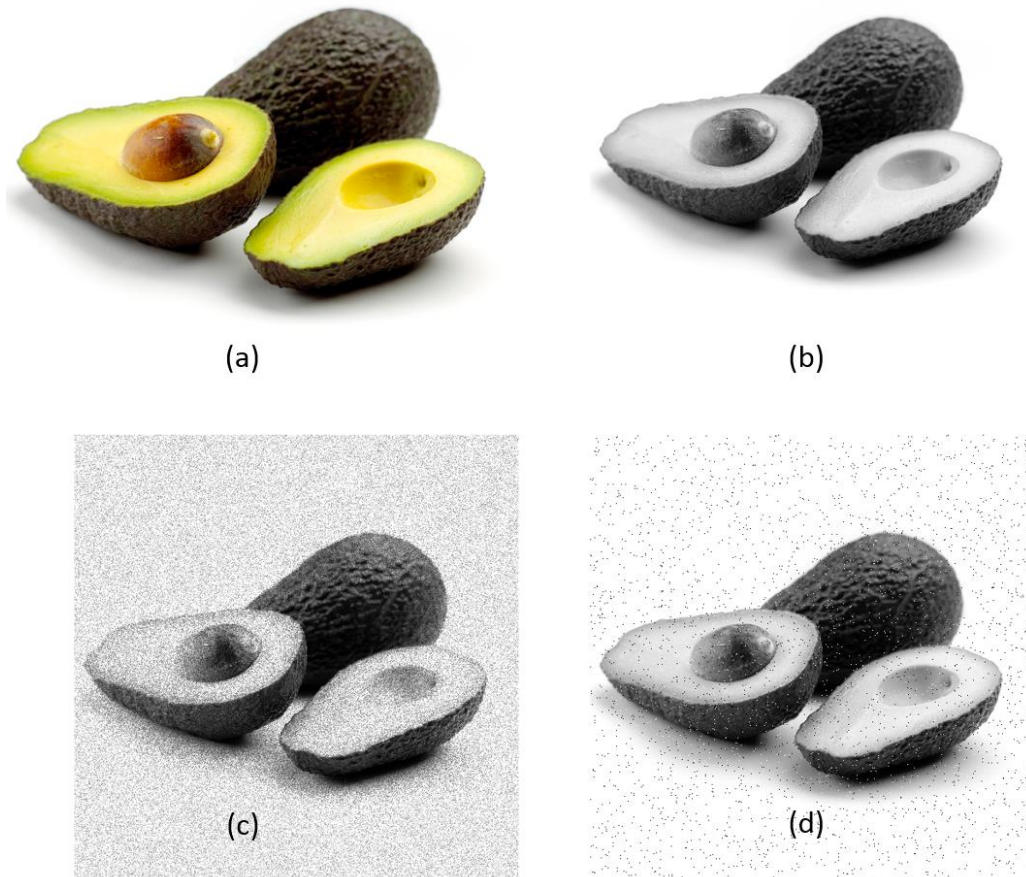


Figura 2.19 Tipos de ruido en una imagen. (a) Imagen original. (b) Escala de gris. (c) Ruido tipo Gaussiano. (d) Ruido tipo "Sal y Pimienta"

Debido a la presencia de este tipo de alteraciones en las imágenes, es necesario la aplicación de filtros que contribuyan en eliminar de una imagen la presencia de los diferentes tipos de ruido.

Un tipo de filtro usado comúnmente son los filtros de orden, están basados en un tratamiento específico de la estadística de la imagen llamado estadísticas de orden. Estos filtros operan en una vecindad de un determinado pixel, denominada *ventana* y rempazan al valor del pixel central. La estadística de *orden* es una técnica que organiza todos los pixeles de la ventana en orden secuencial, basándose en el valor del nivel de gris de los mismos. El lugar de un determinado valor dentro de un conjunto ordenado es referido como el *rango* (Pajares Martinsanz & de la Cruz García, 2008).

Dada una ventana V de dimensión $n \times n$, los valores de los píxeles pueden ordenarse de menor a mayor como siguen:

$$f_1 \leq f_2 \leq f_3 \leq \dots \leq f_{n^2} \quad (2.10)$$

donde $\{f_1, f_2, f_3, \dots, f_{n^2}\}$ son los valores de intensidad o nivel de gris del subconjunto de píxeles en la imagen.

2.3.7.5.1 Filtro de la mediana

La mediana M de un conjunto de valores es tal que la mitad de los valores del conjunto son menores que M y la mitad de los valores son mayores que M . Con el objeto de realizar un filtrado de la mediana en el entorno de la vecindad, ordenamos las intensidades de la vecindad como se propone en la ecuación (2.10), determinamos la mediana y asignamos esta última a la intensidad del píxel. La principal función del filtrado de la mediana es hacer que los puntos con intensidades muy distintas se hagan muy parecidos a sus píxeles vecinos, eliminando así los picos de intensidad que aparezcan aislados en el área de la máscara del filtro (R. González, 1996; Pajares Martinsanz & de la Cruz García, 2008). En la Figura 2.20 se observa el resultado de aplicar un filtro de orden tipo *mediana* a una imagen en escala de gris sometida a ruido Gaussiano.



(a)



(b)

Figura 2.20 Eliminación de ruido gaussiano aplicando filtro de mediana. (a) Ruido Gaussiano. (b) Filtro de la mediana

2.3.7.6 Segmentación

En las técnicas de procesamiento de imágenes, cada píxel es transformado de un valor a otro (escala de gris, binarización, etc). Estos cambios, mayoritariamente, se realizan para facilitar la partición de la imagen en áreas de píxeles con significado. La segmentación trata de agrupar los píxeles, por algún criterio de homogeneidad, para particionar la escena en regiones de interés. Estas áreas deben de tener algún significado físico. Por tanto, la segmentación de una imagen es un proceso de extracción de los objetos de interés insertados en la escena capturada (Figura 2.21). La agrupación de los píxeles se hace a razón de que sus vecinos sean similares en criterios como de luminancia, color, bordes, texturas, movimientos, entre otros (Rodríguez, 2012; R. González, 1996; Pajares Martinsanz & de la Cruz García, 2008). Una vez que la imagen ha sido particionada, la unidad dejara de ser el píxel para ser la agrupación de píxeles que constituye el objeto, de esta forma, la imagen estará definida por un conjunto de objetos y la información estará preparada para el reconocimiento e interpretación de la imagen.

Para la segmentación de las imágenes se usan tres conceptos básicos:

Similitud: los píxeles agrupados del objeto deben ser similares respecto algún criterio (nivel de gris, color, borde, textura, etc).

Conectividad: los objetos corresponden a áreas de píxeles con conectividad. Las particiones corresponden con regiones continuas de píxeles.

Discontinuidad: los objetos tienen formas geométricas que definen unos contornos. Estos bordes delimitan unos objetos de otros.

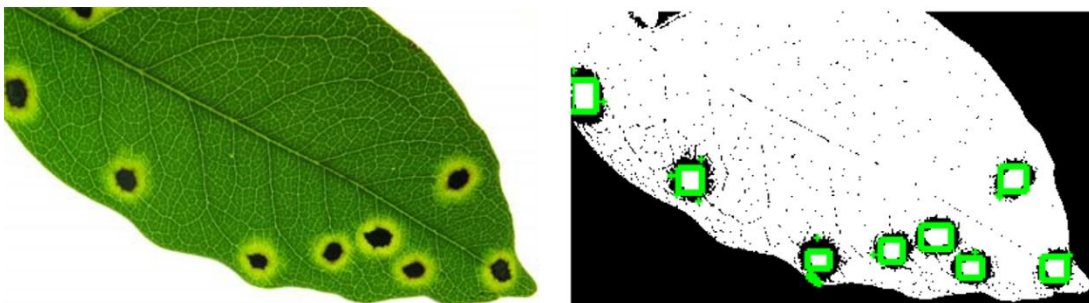


Figura 2.21 Detección de manchas en hojas mediante segmentación

2.3.7.7 Morfología matemática

La palabra morfología denota una rama de la Biología que se ocupa de la forma y estructura de los animales y plantas. La morfología matemática está basada en la geometría y la forma. Las operaciones morfológicas simplifican las imágenes y preservan las formas principales de los objetos.

En visión artificial es frecuente utilizar la morfología para el tratamiento de regiones en el sentido de determinar cómo se pueden cambiar, contar o evaluar; pueden emplearse en los siguientes casos:

- Suavizar los bordes de una región. Esto es útil para mejorar un borde, ya que usando técnicas de segmentación estándar, los bordes se presentan generalmente ruidosos debido tanto al proceso de captura como a los procesos de segmentación apareciendo sobre ellos pequeñas protuberancias, que pueden suprimirse mediante transformaciones morfológicas.
- Separar determinadas regiones que el proceso de segmentación las presenta unidas.
- Unir regiones que han sido separadas durante la segmentación.
- Facilita el cómputo de regiones en una imagen.

El conjunto más simple de transformaciones morfológicas cuantitativas está formado por: dilatación, erosión, apertura y cierre.

2.3.7.7.1 Erosión

La transformación de la erosión es el resultado de comprobar si el elemento estructurante B está completamente incluido dentro del conjunto X (R. González, 1996; Sucar & Gómez, 2010; Pajares Martinsanz & de la Cruz García, 2008). Cuando no ocurre, el resultado de la erosión es el conjunto vacío:

$$\varepsilon_B(X) = \{x | B_x \subseteq X\} \quad (2.11)$$

Cuando los objetos de la escena sean menores que el elemento estructurante, éstos desaparecerán. Otra interpretación de la erosión supone tomar el valor mínimo de la imagen en el entorno de vecindad definido por el elemento estructurante.

Su utilidad consiste en definir una geometría determinada por el elemento estructurante y pasarlo sobre la imagen. Los objetos de menor tamaño que el elemento estructurante no aparecerán en la imagen resultante. Los objetos que queden de la transformación habrán sido degradados. Por tanto, la erosión supone una degradación de la imagen. La aplicación iterativa de esta transformación hará que se eliminen todos los objetos existentes en la imagen (Figura 2.22). La erosión es una transformación antiextensiva:

$$\varepsilon_B(X) \subseteq X \quad (2.12)$$

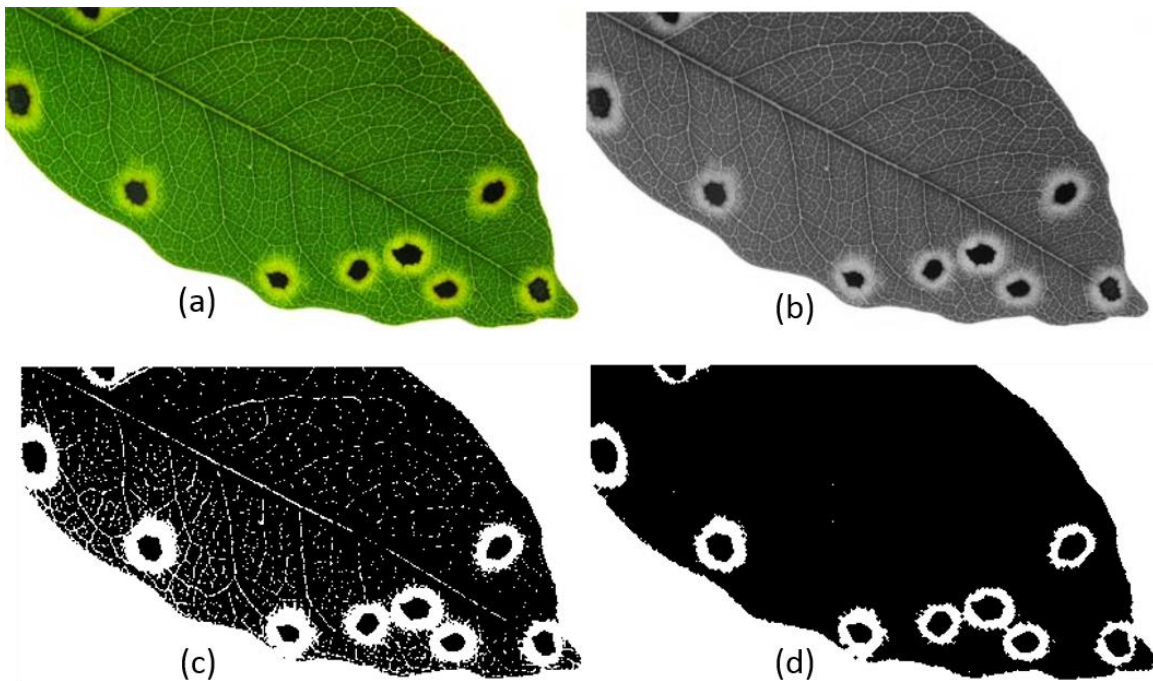


Figura 2.22 Erosión de una Imagen. (a) Imagen original. (b) Escala de gris. (c) Binarizado. (d) Erosión

2.3.7.7.2 Dilatación

La dilatación es la transformación dual a la erosión. El resultado de la dilatación es el conjunto de elementos tal que al menos algún elemento del conjunto estructurante B está contenido en el conjunto X, cuando B se desplaza sobre el conjunto X (R. González, 1996; Pajares Martinsanz & de la Cruz García, 2008):

$$\delta_B(X) = \{x | X \cap B_x \neq \emptyset\} \quad (2.13)$$

Esta operación representa un crecimiento progresivo del conjunto X. Al pasar el elemento estructurante dentro del conjunto, éste no se modificará. Sin embargo, en la frontera del conjunto X, al desplazar a B, el conjunto resultado se expansionará. La aplicación iterada de este operador haría degradar la imagen, haciendo coincidir el conjunto dilatado con la totalidad del dominio de la imagen (Figura 2.23) (R. González, 1996; Pajares Martinsanz & de la Cruz García, 2008). La dilatación es una transformación extensiva:

$$X \subseteq \delta_B(X) \quad (2.14)$$

La dilatación también se interpreta como el valor máximo del entorno de vecindad definido por el elemento estructurante.

Las aplicaciones de las operaciones de erosión seguida con una dilatación no son conmutativas. Los resultados son diferentes dando paso a las aperturas y cierres morfológicos.

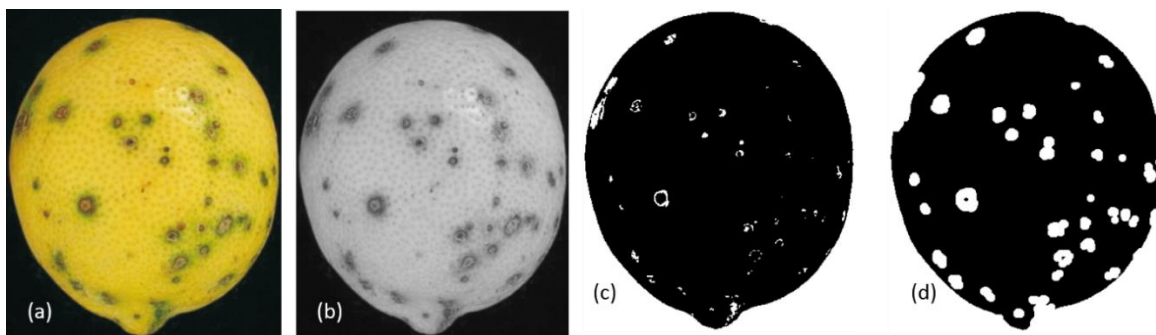


Figura 2.23 Dilatación de una imagen. (a) Imagen original. (b) Escala de gris. (c) Binarización. (d) Dilatación

2.3.7.7.3 Apertura y cierre

Usando los operadores elementales de erosión y dilatación se pueden diseñar operaciones de realce de las formas de los objetos. La erosión binaria suele utilizarse para eliminar pequeños objetos, práctica que suele ser utilizada en la fase de post-procesado, después de una segmentación inicial. Sin embargo, tiene el inconveniente de disminuir el tamaño del resto de los objetos. Este efecto puede ser subsanado con una aplicación en cascada de erosión y dilatación binaria con igual elemento estructurante. A esta operación se la llama apertura (*opening*) (R. González, 1996; Umbaugh, 1998; Parker, 2010):

$$\gamma_B(X) = \delta_B(\varepsilon_B(X)) \quad (2.15)$$

La apertura binaria elimina todos los objetos que no están completamente contenidos en el elemento estructurante, pero además no disminuye el tamaño a los objetos que superen la erosión. Sin embargo, la imagen resultante no recupera la misma forma de los objetos filtrados de la imagen de entrada. Esta operación puede ser ideal para la eliminación de ruido, aunque no preserva la geometría de los objetos. Los bordes de los objetos serán suavizados.

Por el contrario, la dilatación binaria opera aumentando el tamaño de los objetos y cerrando los agujeros y las grietas. El ensanchamiento de los objetos puede ser reducido mediante la aplicación seguida de una erosión. La operación combinada de dilatación y erosión es llamada cierre (*closing*) (R. González, 1996; Parker, 2010; Umbaugh, 1998):

$$\varphi_B(X) = \varepsilon_B(\delta_B(X)) \quad (2.16)$$

El cierre binario morfológico produce que la dilatación rellene las estructuras que la erosión no puede separar. Los contornos de los objetos también serán suavizados, pero habiendo rellanado las fisuras.

Capítulo 3

Redes Neuronales Artificiales

El aprendizaje del ser humano, desde su nacimiento, se basa en experiencias percibidas del medio ambiente a través de los sentidos como son el oído, la vista, el tacto, el gusto y el olfato. A través de los años aprendemos a reaccionar a los estímulos generados desde el exterior. Si nuestro sistema nervioso detecta una cierta cantidad de percepciones y además sus intensidades son altas o sobrepasa cierto umbral generarán una acción inmediata o bien quedarán almacenados en nuestra memoria por algún tiempo para futuras reacciones. El umbral al cual un ser humano reacciona dependerá directamente de sus experiencias previas.

Existen algunas diferencias básicas entre el tipo de procesamiento que se lleva a cabo en una computadora digital con respecto al sistema nervioso central de un ser vivo. La diferencia más importante consiste en la forma de procesar los datos de entrada. Una computadora digital convencional lleva a cabo un proceso secuencial de los datos mientras que el cerebro realiza un proceso en paralelo, lo cual aumenta las capacidades de procesamiento y reconocimiento de éste.

Otro factor importante a considerar es que a pesar de la diferencia desproporcionada en velocidad de transmisión de datos de los componentes electrónicos de una computadora digital con respecto a los componentes biológicos de un cerebro humano, este tiene una respuesta más rápida a las labores de reconocimiento de patrones.

En este capítulo se presentarán los conceptos básicos del funcionamiento y entrenamiento de una red neuronal implementada en software basado en su análogo biológico. Se analizarán y se expondrán los fundamentos biológicos así como su interpretación y modelación matemática.

3.1 Neurona biológica

En esta sección se explica de una manera simple y breve el funcionamiento celular en cuanto al aspecto excitatorio eléctrico que sufre la neurona para llevar a cabo la propagación de un potencial de acción.

3.1.1 Descripción del funcionamiento y actividad neuronal

Uno de los factores más importante en el funcionamiento neuronal es la conexión entre las neuronas (Figura 3.1). A cada lugar de conexión entre neuronas se le llama sinapsis. Los botones sinápticos están localizados sobre toda la superficie de la neurona, los cuales pueden sufrir eventos excitatorios o inhibitorios, dependiendo de las características del impulso nervioso (Guyton, 2011).

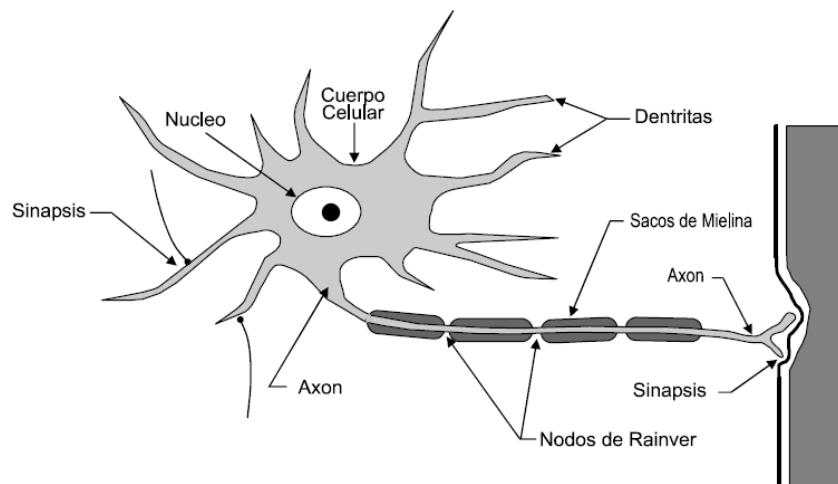


Figura 3.1 Neurona Biológica

Existe una separación de 200 a 300 Angstroms entre el botón sináptico y el cuerpo de la neurona (hendidura sináptica). Dentro del botón sináptico se localizan las mitocondrias y las vesículas sinápticas quienes llevan a cabo las acciones de excitación e inhibición (Figura 3.2). Las sustancias transmisoras para realizar las excitaciones y las inhibiciones se localizan en las vesículas. Mientras que las mitocondrias contienen el trifosfato de adenosina (ATP) requerido para sintetizar estas sustancias. El ATP es regenerado en cada momento, ya que su acción dura sólo unos segundos. Estas vesículas son vaciadas a través de la hendidura cuando ocurre un potencial de acción. Tal situación genera un cambio en la permeabilidad de la membrana celular, lo que genera una excitación y una inhibición dependiendo del tipo de transmisor (Guyton, 2011).

La proporción de iones de sodio y calcio extracelular genera un flujo similar de neurotransmisores que van hacia la neurona, lo cual desencadena un flujo excitatorio. En el caso de un impulso inhibitorio ocurre algo similar cuando el flujo de iones de magnesio se incrementa. La permeabilidad de la membrana celular es permitida sólo para los iones de sodio y potasio. El sodio es enviado al exterior celular, mientras que el potasio es captado dentro de la neurona, es decir funciona como una bomba de sodio-potasio, de tal forma que fija una concentración pequeña de sodio y una alta de potasio dentro de la neurona, que mantiene en un estado de voltaje estable (período de descanso) a la neurona.

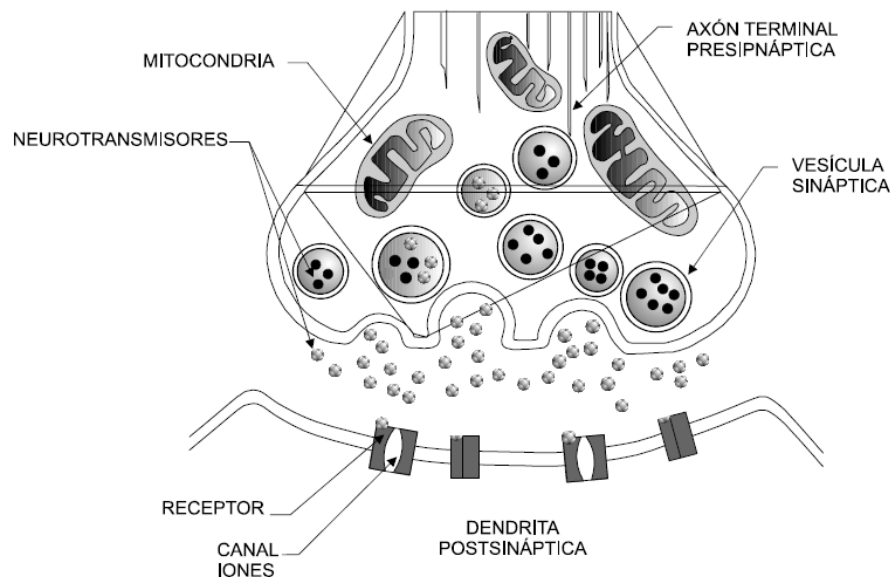


Figura 3.2 Botón Sináptico

Dado que los iones orgánicos son demasiado grandes en comparación que los poros de la membrana celular, estos no pueden fluir al exterior celular. Esto genera que la carga interna sea preferencialmente negativa en estado de reposo, lo cual favorece que se acumule una mayor concentración de iones de sodio y potasio en el interior celular. Esta situación hace desfavorable la concentración de iones negativos de cloro. De tal forma que el potencial de descanso quedará en un rango de voltaje de -70 a -100 microvolts (potencial de descanso) (Guyton, 2011). Este voltaje es menor que el presente en tejido esquelético o nervioso periférico, pero el hecho de que se mantenga en un nivel bajo hace posible un control positivo y negativo del nivel de excitación neuronal. Ya que si disminuimos el voltaje a un valor positivo la membrana neuronal se excita con mayor facilidad y caso contrario se inhiba rápidamente.

La comunicación entre las neuronas se lleva a cabo por medio de extensiones nerviosas llamadas axones. Estos están recubiertos de mielina, la cual es particionada por los nodos de Ranvier, los cuales sirven para llevar a cabo la amplificación de la señal. Las neuronas sólo pueden secretar un sólo tipo de sustancia transmisora, sea esta inhibitoria o excitadora. Entonces cualquier impulso nervioso inhibitorio o excitador será transmitido a través de los axones procedentes de la neurona a todas las sinapsis de otras neuronas hasta donde se extienda la conexión a través de los axones.

La diferencia de potencial entre el interior de la neurona y su exterior es disminuido con cada entrada excitadora (proceso de sumación) a la neurona. Esto modifica las características permeables de la membrana en el caso de los iones de sodio, permitiéndolos fluir con facilidad. Esto genera una despolarización, lo que a su vez genera un potencial de acción (Figura 3.3) si un umbral es sobrepasado y a su vez un efecto en cadena que es propagado a las demás neuronas con las que se tiene conexión a través de los axones. Sin embargo, el potencial de acción no inicia en la membrana somática vecina de los botones excitadores, sino que empiezan en la parte inicial del axón, lo cual es debido a que cualquier factor que aumenta el potencial dentro del soma en un lugar aislado incrementa este potencial en todo el resto del soma al mismo tiempo. Además, debido a las diferencias físicas de la membrana y la disposición geométrica de la membrana en diferentes localidades de la neurona, el voltaje en el interior celular que desencadenará un potencial de acción también difiere según los

puntos de la membrana neuronal. Las primeras 100 micras del axón representan la parte más excitable de la neurona. El potencial posterior a la sinapsis es cercano a los 10 milivolts (Guyton, 2011). De esta manera el potencial de acción será propagado a través del axón. Ya que el potencial de acción ha iniciado, éste se propagará en dirección periférica a lo largo del axón, y también en sentido retrógrado al soma de la neurona.

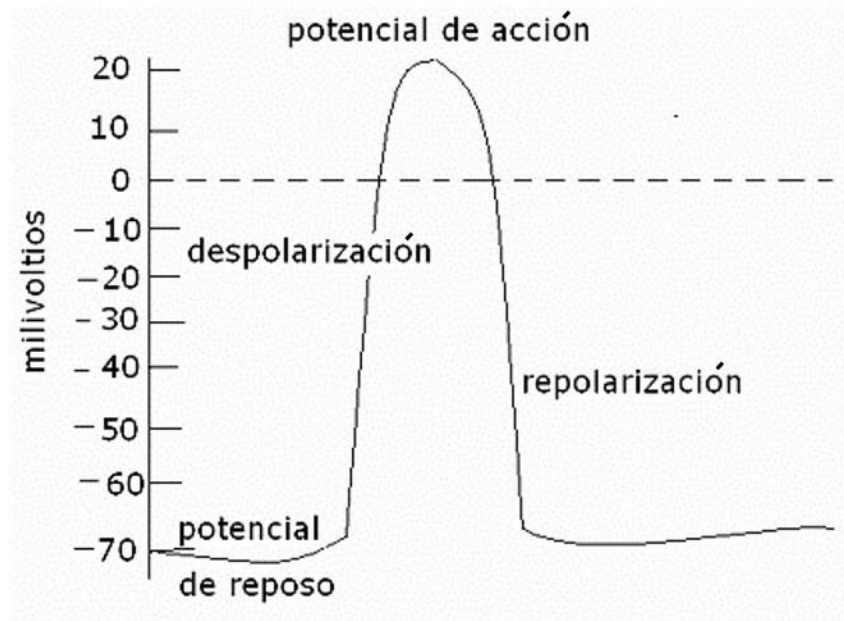


Figura 3.3 Potencial de acción

Es importante resaltar que una sola excitación en alguna sinapsis neuronal no será capaz de provocar un potencial de acción. Para que este pueda ocurrir una buena cantidad de neurotransmisores excitadores deben de ocurrir en un momento determinado en diferentes puntos sinápticos. A esto se le llama el proceso de sumación espacial. Después de que un potencial de acción ocurre debe esperarse un período refractario para que otro potencial de acción pueda ocurrir.

El análisis y entendimiento del funcionamiento biológico de una neurona son primordiales para la modelación artificial de una neurona, así como el de una red neuronal, en las siguientes secciones se expone la modelación de una red neuronal artificial y sus características.

3.2 Neurona Artificial

Algunos científicos con el interés de entender el funcionamiento neuronal y el comportamiento humano, entre los que se encontraban psicólogos, físicos, médicos y matemáticos, se enfocaron en estudiar modelos artificiales simplificados de redes neuronales biológicas. Los primeros modelos fueron desarrollados por Mc. Culloch & Pitts (1943), Rosenblatt (1958), Hebb (1949), Minsky (1969), Widrow & Hoff (1960) y otros más.

En un modelo típico de una neurona artificial se presenta las entradas, que representan los valores binarizados que codifican dependiendo del problema los atributos que se quieren reconocer, o bien pueden ser las salidas de neuronas en capas anteriores. La salida será un valor por lo general limitado por una función de activación, el cual representa la clase reconocida dados las entradas (Haykin, 1994; Kröse & Van der Smagt, 1996).

3.2.1 Definición de una red neuronal artificial

Las RNA¹⁰ se definen como sistemas de mapeos no lineales cuya estructura se basa en principios observados en los sistemas nerviosos de humanos y animales. Constan de un número grande de procesadores simples ligados por conexiones con pesos. Las unidades de procesamiento se denominan neuronas. Cada unidad recibe entradas de otros nodos y genera una salida simple escalar que depende de la información local disponible, guardada internamente o que llega a través de las conexiones con pesos. Pueden realizarse muchas funciones complejas dependiendo de las conexiones.

Las neuronas artificiales simples fueron introducidas por McCulloch y Pitts en 1943. Una red neuronal se caracteriza por los siguientes elementos (Rojas, 1996; Matich, 2001):

- Un conjunto de unidades de procesamiento o neuronas.
- Un estado de activación para cada unidad, equivalente a la salida de la unidad.

¹⁰ RNA. Red Neuronal Artificial (en inglés ANN: Artificial Neural Network).

- Conexiones entre las unidades, generalmente definidas por un peso que determina el efecto de una señal de entrada en la unidad.
- Una regla de propagación, que determina la entrada efectiva de una unidad a partir de las entradas externas.
- Una función de activación que actualiza el nuevo nivel activación basándose en la entrada efectiva y la activación anterior.
- Una entrada externa que corresponde a un término determinado como bias para cada unidad.
- Un método para reunir la información, correspondiente a la regla de aprendizaje.
- Un ambiente en el que el sistema va a operar, con señales de entrada e incluso señales de error.

En muchas redes las unidades de proceso tiene respuesta de la forma:

$$y = f\left(\sum_k \omega_k x_k\right) \quad (3.1)$$

donde:

x_k : señales de salida de otros nodos o entradas externas.

ω_k : pesos de las ligas de conexión.

$f(\cdot)$: función no lineal simple.

La función f puede ser sigmoideal, tangente hiperbólica, escalón, entre otras.

Cada unidad de proceso tiene una tarea simple: recibir la entrada de otras unidades o de fuentes externas y procesa la información para obtener una salida que se propaga a otras unidades.

Una red puede tener una estructura arbitraria, pero las capas que contienen estas estructuras están definidas de acuerdo con su ubicación en la topología de la red neuronal. Las entradas externas son aplicadas en la primera capa, y las salidas se consideran la última capa. Las capas internas que no se observan como entradas o salidas se denominan capas ocultas. Por convención, las entradas no se consideran como capa por que no realizan procesamiento (Kleffors, 1998; Basogain, 2008; Dongare, Kharde, & Kachare, 2012).

La entrada total u de una unidad k es la suma de los pesos de las entradas conectadas, más un bias θ :

$$u = \sum_j \omega_j x_j + \theta \quad (3.2)$$

Si el peso ω_j es positivo se habla de una *excitación* y si el peso es negativo se considera una *inhibición* de la entrada. Si consideramos a las entradas como funciones del tiempo, la expresión anterior se convierte en:

$$u_t = \sum_j \omega_j(t) x_j(t) + \theta(t) \quad (3.3)$$

3.2.2 Funciones de activación

La regla que logra establecer el efecto de la entrada total $\omega(t)$ en la activación de unidad k se denomina función de activación (F_k) (Dayhoff, 1990; Wasserman, 1989):

$$y(t + 1) = F_k(y(t), \theta(t)) \quad (3.4)$$

En muchas ocasiones esta función es de la forma no decreciente respecto a la entrada total de la unidad:

$$y(t + 1) = F_k \left[\sum_j \omega_j(t) x_j(t) + \theta(t) \right] \quad (3.5)$$

Algunas de las funciones de activación más usadas son las siguientes:

3.2.2.1 Función escalón

La función escalón se asocia a neuronas binarias en las cuales, cuando la suma de las entradas es mayor o igual que el umbral de la neurona, la activación es 1; si es menor, la activación es 0 (o -1) (Hagan & Dcmuth, 1996; Barrera, 2003) (Figura 3.4).

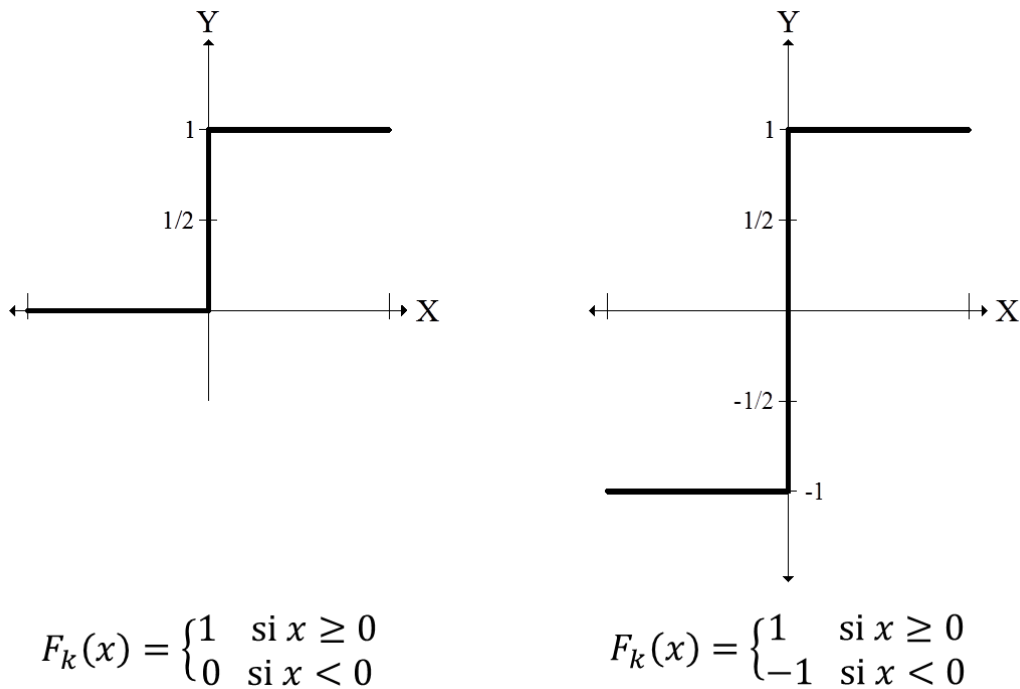


Figura 3.4 Función de activación escalón

3.2.2.2 Función lineal o mixta

La función lineal o identidad responde a la expresión $F_k(u) = u$. En las neuronas con función mixta, si la suma de las señales de entrada es menor que el límite inferior, la función se define como 0 (o -1). Si dicha suma es mayor o igual que el límite superior, entonces la activación es 1. Si la suma de entrada está comprendida entre los ambos límites, superior e inferior, entonces la activación se define como una función lineal de la suma de las señales de entrada (Hagan & Dcmuth, 1996; Schalkoff, 1997) (Figura 3.5)

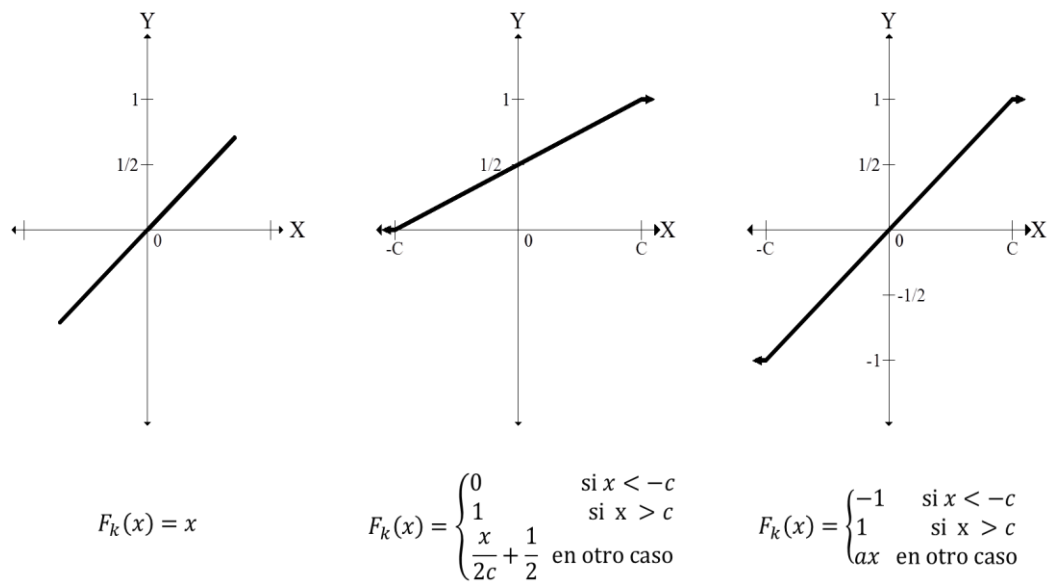


Figura 3.5 Función de activación lineal o mixta

3.2.2.3 Función tangente hiperbólica

La función de activación tangente hiperbólica se emplea en los casos que presentan variaciones suaves de valores positivos y negativos de la señal a clasificar. Como se puede ver en su descripción en la Figura 3.6, es una de las funciones más empleadas en entrenamientos supervisados, como en el caso del entrenamiento de retropropagación del error.

Debe tenerse cuidado de emplear esta figura entre los umbrales positivos y negativos antes de la saturación, de otra forma la salida siempre genera valores saturados iguales a 1 y -1 (Hagan & Dcmuth, 1996; Schalkoff, 1997).

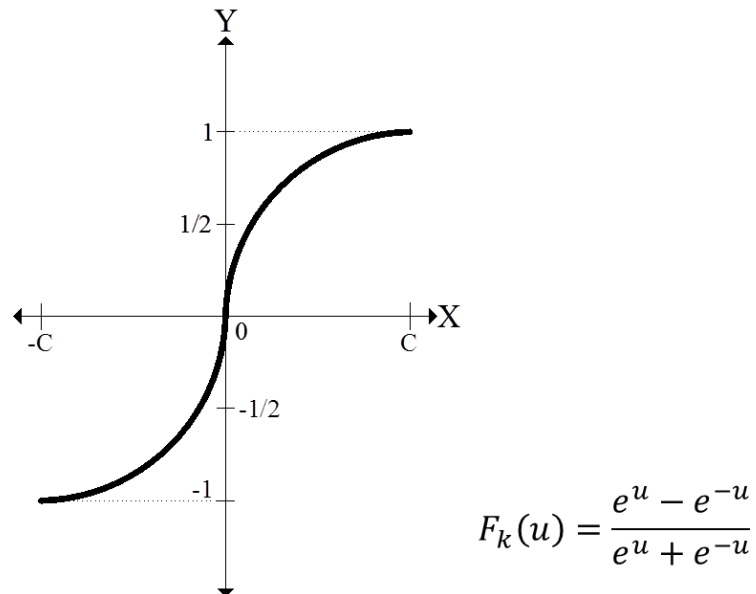


Figura 3.6 Función tangente hiperbólica

3.2.2.4 Función sigmoideal

Con la función sigmoideal el valor dado por la función es cercano a uno de los valores asintóticos. Esto hace que en la mayoría de los casos, el valor de salida esté comprendido en la zona alta o baja del sigmoide. De hecho, cuando la pendiente es elevada, esta función tiende a la función escalón. Sin embargo, la importancia de la función sigmoideal es que su derivada siempre es positiva y cercana a cero para los valores grandes positivos o negativos; además, toma su valor máximo cuando $x = 0$. Esto hace que se puedan utilizar reglas de aprendizaje definidas para las funciones escalón, con la ventaja, respecto a esta función, de que la derivada está definida en todo el intervalo (Figura 3.7) (Hagan & Dcmuth, 1996; Schalkoff, 1997).

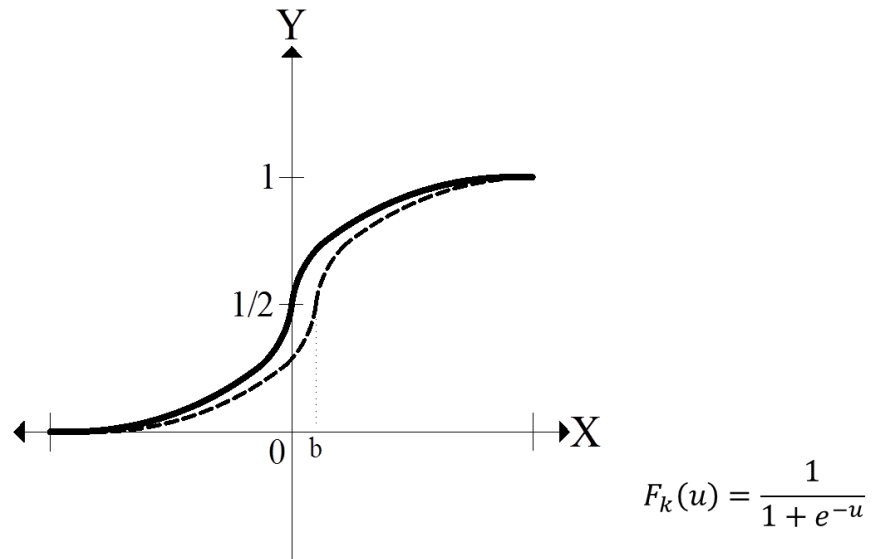


Figura 3.7 Función de activación sigmoideal

3.2.2.5 Función de Gauss

Los mapeos ocultos algunas veces pueden realizarse con un solo nivel de neuronas mediante el uso de funciones de activación de Gauss, en lugar de funciones tipo sigmoideales (Figura 3.8) (Hagan & Dcmuth, 1996; Schalkoff, 1997).

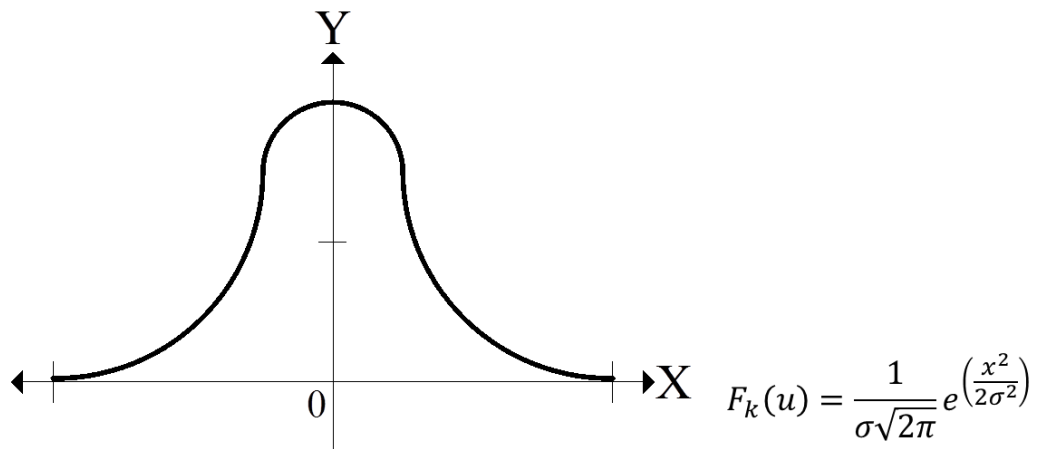


Figura 3.8 Función de activación gaussiana

3.2.3 Topologías de las redes neuronales

Dos de las topologías más usadas, de acuerdo con las diferencias en la manera de realizar las conexiones, son (Hagan & Dcmuth, 1996; Gershenson, 2003; Schalkoff, 1997; Wasserman, 1989):

Redes de propagación hacia adelante (feed-foward): el flujo de la información de las entradas a las salidas es exclusivamente hacia delante, extendiéndose por capas múltiples de unidades, pero no hay ninguna conexión de retroalimentación.

Redes recurrentes: contiene conexiones de retroalimentación, lo que puede derivarse en un proceso de evolución hacia un estado estable en el que no hay cambios en el estado de activación de las neuronas.

3.2.3.1 Elementos de una red neuronal

Una RNA consta de un conjunto de elementos de procesamiento conectados entre sí y entre los que se envían información a través de conexiones. Un esquema básico de una red neuronal artificial se observa en la Figura 3.9, la cual presenta las diferentes capas que tiene esta topología, definiéndose como una estructura que se conoce con el nombre de feed-forward (hacia delante) debido al flujo de la información (Hagan & Dcmuth, 1996; Fyfe, 2000; Freeman & Skapura, 1991).

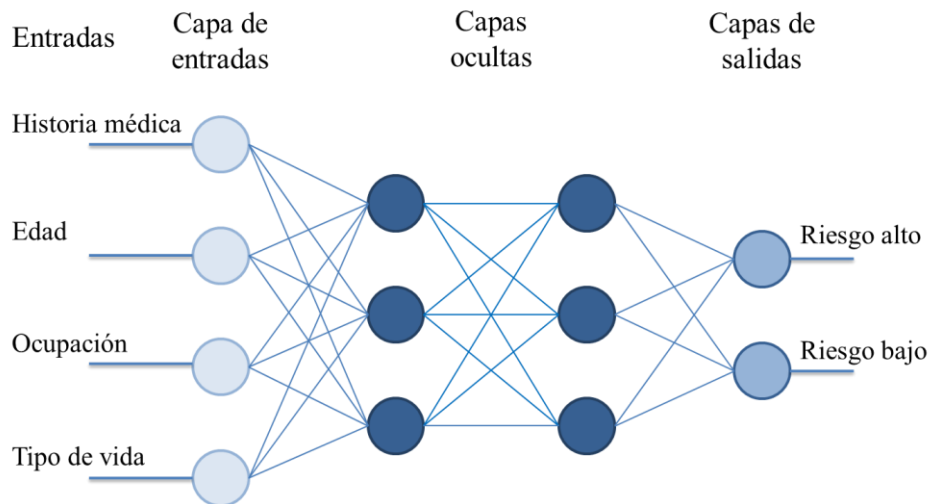


Figura 3.9 Esquema básico de una Red Neuronal Artificial

Los elementos básicos de una RNA son:

- Conjuntos de unidades de procesamiento (neuronas).
- Conexiones entre unidades (asociado a cada conexión un peso o valor).
- Funciones de salida o activación para cada unidad de procesamiento.

El modelo de la neurona y la arquitectura de la red describen cómo esta última transforma sus entradas en las salidas. Los elementos individuales de cálculo que forman la mayoría de los modelos de sistemas neurales artificiales, se conocen como neuronas artificiales.

3.2.4 Entrenamiento de las redes neuronales

Se denomina entrenamiento al proceso de configuración de una red neuronal para que las entradas produzcan las salidas deseadas a través del fortalecimiento de las conexiones. Una forma de llevar esto a cabo es a partir del establecimiento de pesos conocidos con anterioridad y otro método implica el uso de técnicas de retroalimentación y patrones de aprendizaje que cambian los pesos hasta encontrar los adecuados.

Además, el aprendizaje puede dividirse en *superioridad o asociativo* y *no supervisado o auto-organizado* (Haykin, 1994; Hagan & Dcmuth, 1996; Klefors, 1998). En el primer caso se introducen entradas que corresponden a determinadas salidas, ya sea por un agente externo o por el mismo sistema. En el segundo caso el entrenamiento se enfoca a encontrar características estadísticas entre agrupamientos de patrones en las entradas.

Un tipo de regla que se usa para el entrenamiento mediante el ajuste de pesos es la *Hebbiana*, propuesta por Hebb en 1949 y que ha dado pie a diversas variantes propuestas en el transcurso del tiempo. Si dos unidades j y k están activas al mismo tiempo, la conexión entre las dos debe ser fortalecida mediante la modificación del peso (Hagan & Dcmuth, 1996; Kröse & Van der Smagt, 1996):

$$\Delta\omega_{jk} = \gamma y_j y_k \quad (3.6)$$

donde γ es una constante de proporcionalidad positiva que representa la tasa de aprendizaje.

Otra regla usada comúnmente implica el ajuste de los pesos a través de la diferencia entre la activación actual y la deseada; se le conoce como *Regla Delta*:

$$\Delta\omega_{jk} = \gamma y_j (d_k - y_k) \quad (3.7)$$

donde d_k es la activación deseada.

3.2.5 Redes de una capa

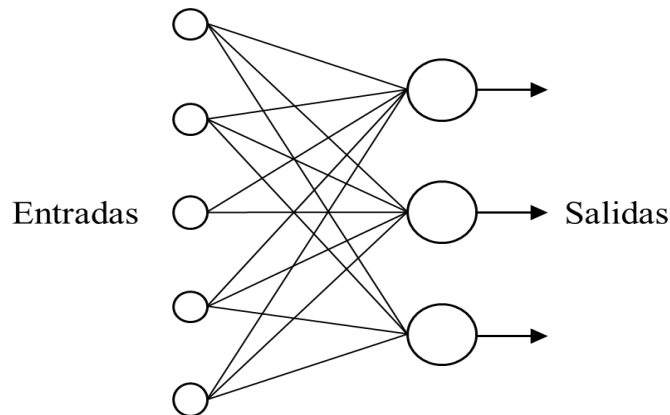


Figura 3.10 Red de una capa

Las redes de una capa, como la que se muestra en la Figura 3.10, presentan las entradas directamente conectadas a las salidas mediante pesos. Las salidas no interactúan, por lo que una red con N_{out} salidas puede ser analizada como N_{out} redes separadas. Cada unidad como

las que muestra la Figura 3.11 produce salidas de la forma (Hagan & Dcmuth, 1996; Krogh, 2008; Rojas, 1996).

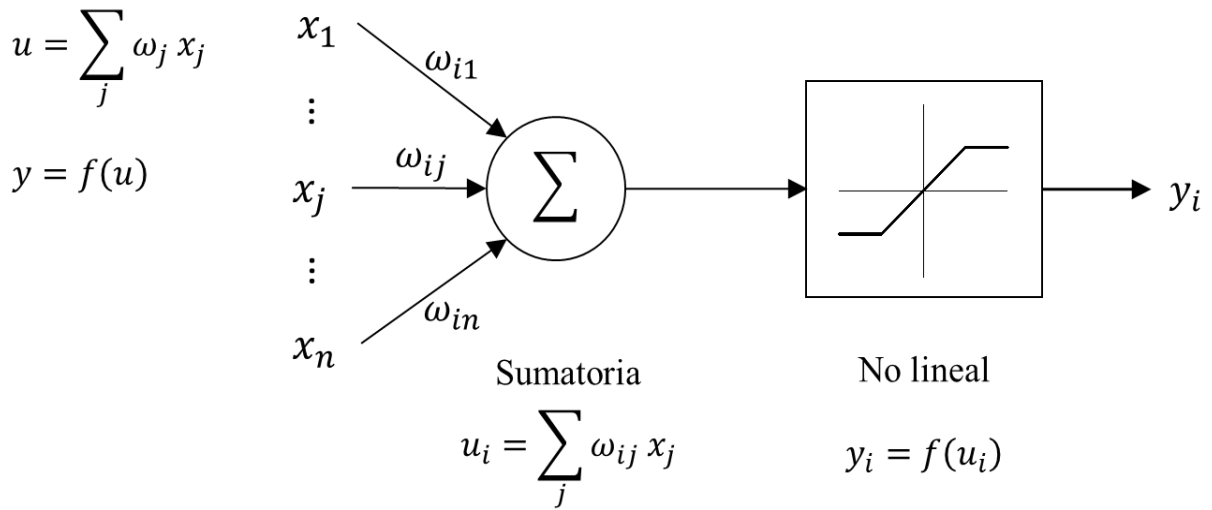


Figura 3.11 Función de nodo

En notación vectorial se expresa como:

$$y(\vec{x}) = f(\vec{\omega}^T \vec{x}) \tag{3.8}$$

Los puntos $\vec{\omega}$ de con una suma constante $\sum_j \omega_j x$ definen un hiperplano perpendicular al vector $\vec{\omega}$. La orientación del hiperplano se determina por la dirección de $\vec{\omega}$, dependiendo de los pesos pero no de la magnitud de $\vec{\omega}$.

La suma $\sum_j \omega_j x$ define un hiperplano que cruza el origen. La introducción de un término θ denominado bias en la sumatoria el hiperplano tiene mayor movilidad y permite la separación de algunos conjuntos de datos, con lo que se obtiene mejores clasificaciones.

3.2.5.1 Perceptrón

Los perceptrones de una capa pueden clasificar correctamente los conjuntos de datos que son linealmente separables, esto es, que pueden ser separados por un hiperplano. En dos dimensiones esto puede verse como una línea recta separada por dos clases (Figura 3.12) (Hagan & Dcmuth, 1996; Matich, 2001).

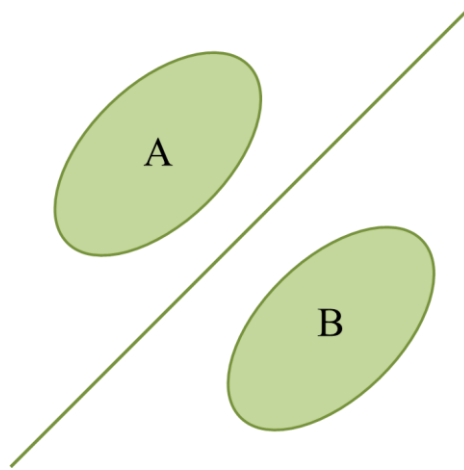


Figura 3.12 Separación lineal de los conjuntos A y B

Un perceptrón convencional tiene una función de no linealidad binaria y la topología que muestra la Figura 3.13 (Hagan & Dcmuth, 1996; Serrano, Soria, & Martín, 2010; Shan, 2010).

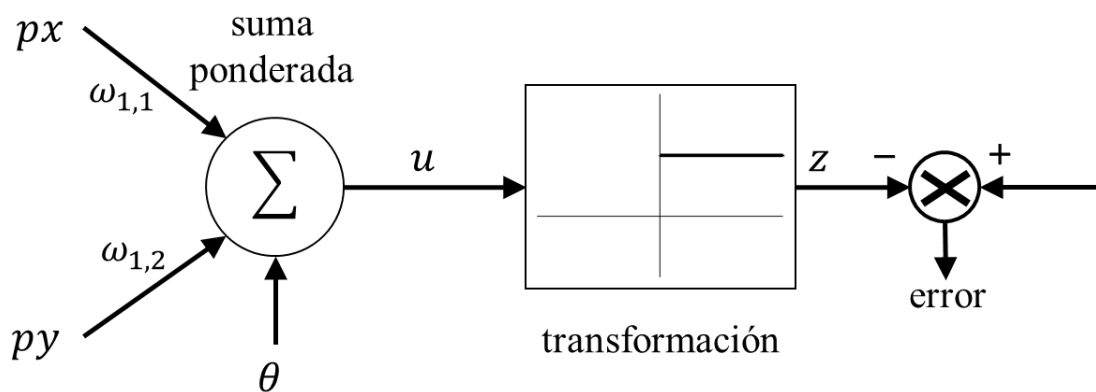


Figura 3.13 Perceptrón clásico propuesto por McCulloch - Pitts

El algoritmo de aprendizaje del perceptrón funciona para aprender funciones binarias linealmente separables, ya que de otra manera el algoritmo no convergería ni produciría la mejor solución. Debido a que las salidas son binarias, se emplean unidades lineales de umbral. Cada unidad calcula la suma u con pesos de las N entradas $x_j, j = 1 \dots N$, y genera una salida binaria y :

$$u = \sum_{j=0}^N \omega_j x_j = \vec{\omega}^T \vec{x} \quad (3.9)$$

$$y = \begin{cases} -1 & u \leq 0 \\ +1 & u > 0 \end{cases}$$

De una forma alterna, se tiene en un intervalo de cero a uno,

$$u = \sum_{j=0}^N \omega_j x_j = \vec{\omega}^T \vec{x} \quad (3.10)$$

$$y = \begin{cases} -1 & u \leq 0 \\ 0 & u > 0 \end{cases}$$

La constante de bias es 1, siendo las entradas, salidas y valores objetivo asumidos como ± 1 o cero. Los pesos son actualizados por un número de reglas simples, comparando las salidas $y(\vec{x})$ con los objetivos $t(\vec{x})$. Los pesos son adaptados con la ecuación:

$$\Delta \vec{\omega} = \begin{cases} 2\eta t \vec{x} & \text{si } t \neq y \\ 0 & \text{otros casos} \end{cases} \quad (3.11)$$

donde: η es una constante positiva pequeña que controla la tasa de aprendizaje, usualmente entre 0 y 1. Al mejorar la exactitud de la clasificación, el sistema comete menos errores y los cambios en los pesos se vuelven menos frecuentes. Una tasa efectiva de aprendizaje puede alentar un proceso, por lo que lograr la clasificación perfecta puede tomar un largo tiempo.

Las redes de una capa y con función de activación tipo sigmoideal se llama normalmente perceptrones, aunque el perceptrón original de Rosenblatt consistía de una familia de redes, de más de una capa, y con conexiones recurrentes (Haykin, 1994; Klefors, 1998).

La forma de entrenamiento se puede analizar de una manera recursiva empleando la siguiente expresión.

$$\Delta \vec{\omega} = \omega(k+1) - \omega(k) = \eta x(k)e(k) \quad (3.12)$$

donde $e(k) = y_d(k) - y(k)$

Se define el error como $e(k)$, el valor deseado $y_d(k)$ y valor actual de salida de la neurona como $y(k)$.

El algoritmo se define como:

1. Fijar pesos iniciales, con valores aleatorios
2. Establecer los valores de entradas x_1, x_2, \dots, x_n
3. Calcular la salida de la neurona

$$y(k) = f\left(\sum x_i \omega_i - \theta\right)$$

4. Actualizar los pesos

$$\Delta \vec{\omega} = \omega(k+1) + \eta x(k)e(k)$$

$$\text{donde: } e(k) = yd(k) - y(k)$$

5. Continuar hasta

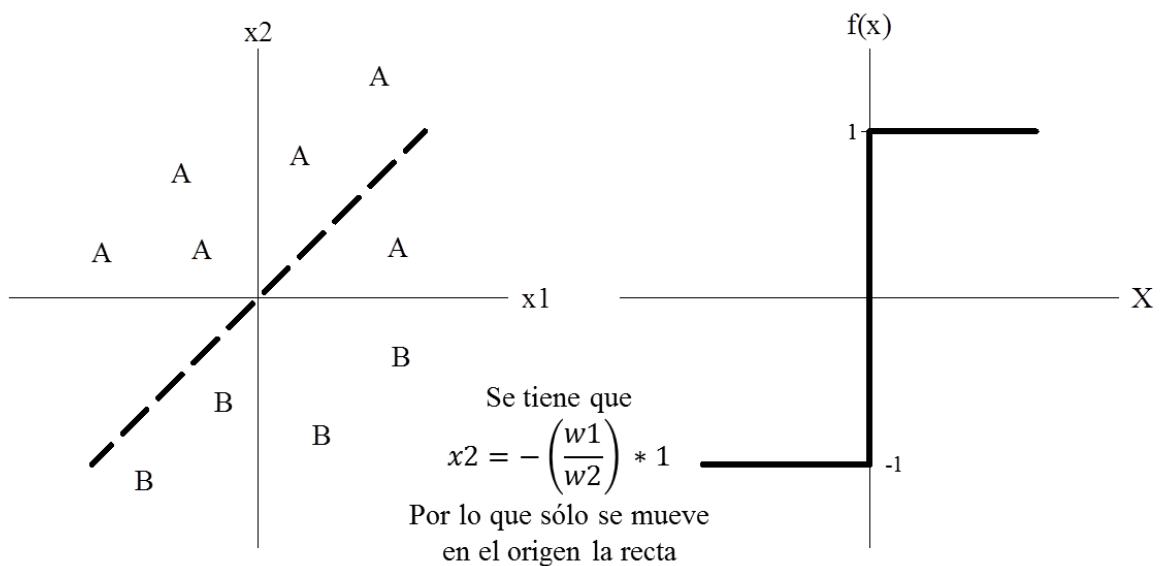
$$e(k) \leq \xi$$

$$\xi = \text{Tolerancia}$$

En caso de no cumplir con la condición del paso 5, se regresa de nuevo al punto 3.

3.2.5.2 Separación de variables linealmente separables con el perceptrón

Un perceptrón de dos entradas puede separar un plano en dos secciones, ya que su ecuación forma una línea recta, como muestra la Figura 3.14 (Hagan & Dcmuth, 1996; Sharma, Rai, & Dev, 2012; Kröse & Van der Smagt, 1996).



El perceptrón puede clasificar un vector analógico de entrada en dos clases A y B

Figura 3.14 Clasificación de patrones con el perceptrón

Considerando las entradas x_1 y x_2 , con sus respectivos pesos ω_1 y ω_2 y un bias θ , se obtiene la ecuación:

$$\omega_1 x_1 + \omega_2 x_2 + \theta = 0 \tag{3.13}$$

Despejando resulta una ecuación donde los pesos determinan la pendiente de la línea y el bias determina el desplazamiento vertical de la clasificación lineal.

$$x_2 = \frac{\omega_1}{\omega_2} x_1 - \frac{\theta}{\omega_2} \tag{3.14}$$

Esta recta se denomina frontera de decisión y es ortonormal al vector de los pesos.

3.2.6 Redes multicapa

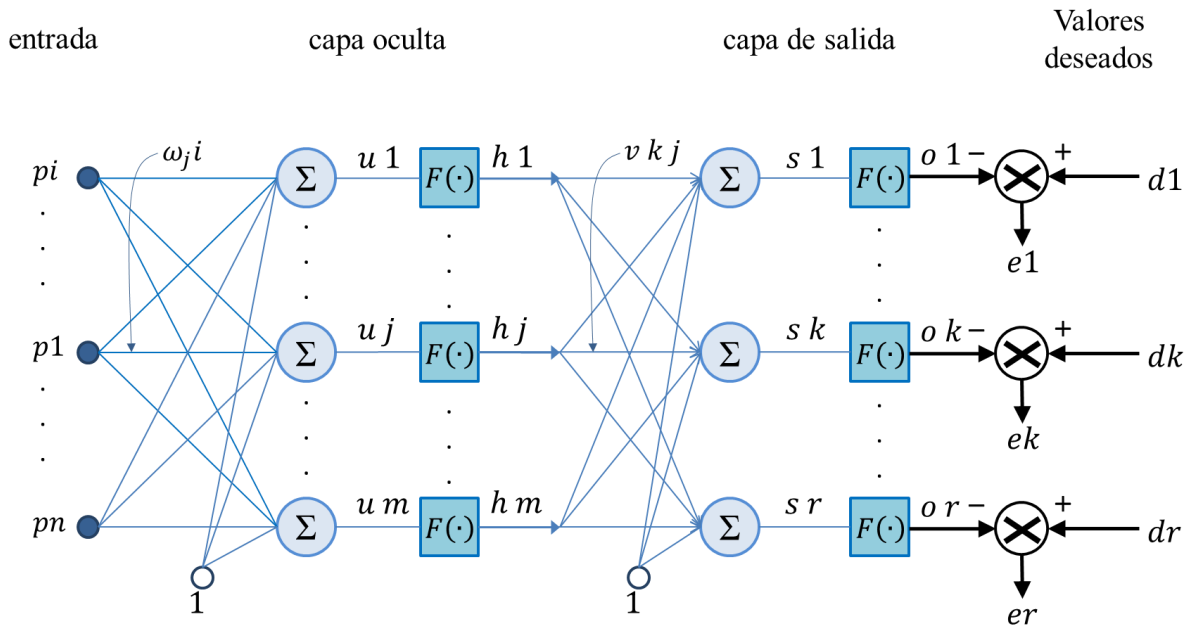


Figura 3.15 Red neuronal de dos capas

3.2.6.1 Perceptrón multicapa

Es uno de los primeros esfuerzos por representar el aprendizaje supervisado, en donde se emplean funciones de activación elementales de forma binaria en la mayoría de las ocasiones. Este elemento es un clasificador lineal con dos entradas y un punto de ajuste, siendo la salida el clasificador.

3.2.7 Redes de retropropagación (backpropagation)

La retropropagación del error es un entrenamiento supervisado que se emplea para redes multicapa, donde se ajusta el valor de los pesos en función del error generado. Esta técnica es muy empleada ya que permite tener un método de optimización que se encuentra al definir el gradiente del error y minimizarlo con respecto a los parámetros de la red neuronal (Hagan & Dcmuth, 1996; Freeman & Skapura, 1991).

Las redes Backpropagation han demostrado su capacidad de trabajar con éxito en un amplio rango de aplicaciones incluyendo clasificación de imágenes, síntesis de voz, clasificación de ecos de sonar, sistemas de base de conocimiento, codificación de información y muchos otros problemas de clasificación y problemas de percepción.

La principal ventaja de la Backpropagation es su capacidad genérica de mapeo de patrones. La red es capaz de aprender una gran variedad de relaciones de mapeo de patrones. No requiere un conocimiento matemático de la función que relaciona los patrones de la entrada y los patrones de salida. La Backpropagation sólo necesita ejemplos de mapeo para aprender. La flexibilidad de esta red es aumentada con la posibilidad de elegir número de capas, interconexiones, unidades procesadoras, constante de aprendizaje y representación de datos. Como resultado de estas características la red Backpropagation es capaz de participar con éxito en una amplia gama de aplicaciones.

3.2.7.1 Principios para entrenar una red multicapa empleando el algoritmo de retropropagación

Si se considera la red de tres capas con dos entradas y una salida de la Figura 3.16, es posible apreciar que cada neurona está compuesta de dos unidades, donde la primera suma los productos de las entradas por sus respectivos pesos, y la segunda unidad contiene la

función de activación. La señal correspondiente a la salida de la suma $y = f(e)$ es la señal de salida del elemento no lineal de la función de activación, así como la salida de la neurona.

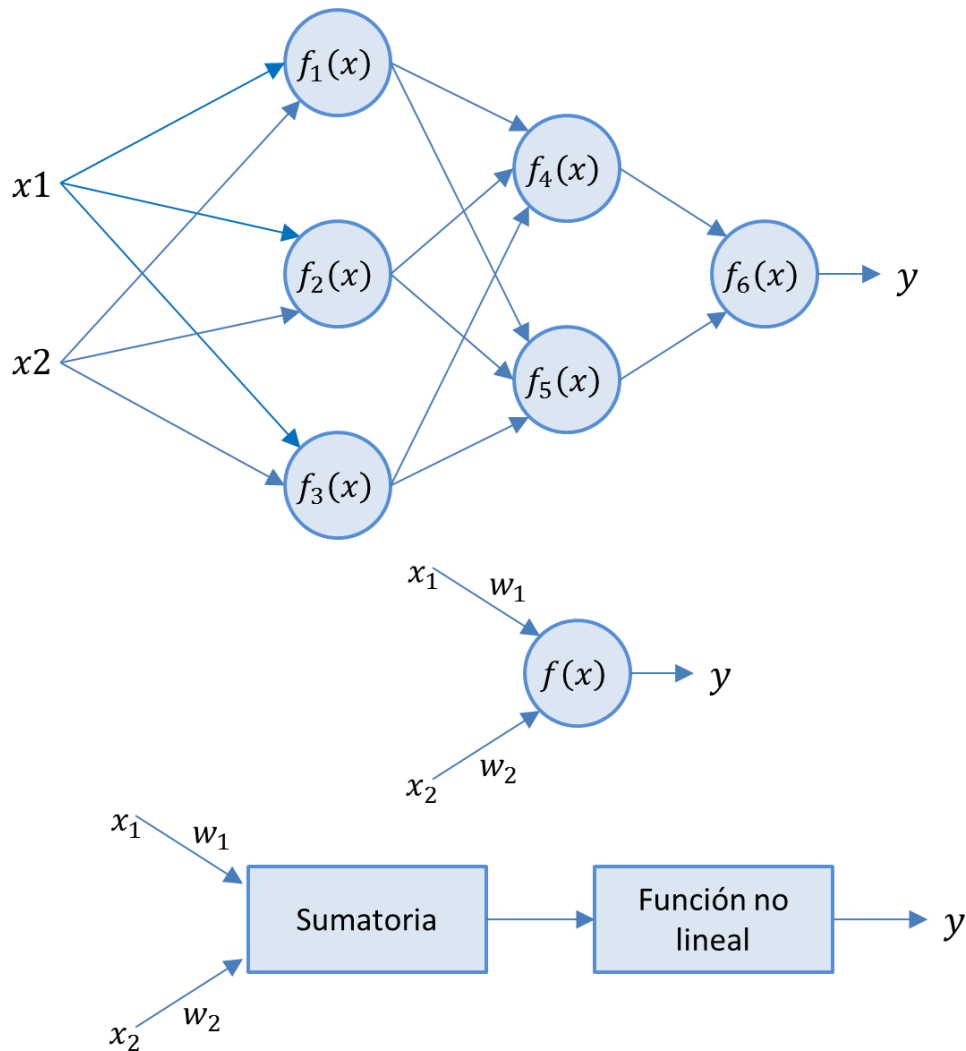


Figura 3.16 Red de dos capas

Para enseñarle a la red neuronal es necesario entrenar un conjunto de datos, el cual consta de señales de entrada x^1 y x^2 asignadas con objetivos correspondientes (salidas deseadas) denominadas z . El entrenamiento es un proceso iterativo. En cada iteración los pesos de los nodos se modifican usando nuevos datos del conjunto para el entrenamiento. Las modificaciones de los pesos se calculan empleando el siguiente algoritmo que se explica a continuación.

Cada paso del entrenamiento comienza al forzar ambas entradas de salida del conjunto de entrenamiento. Después es posible determinar los valores de salida de las señales de cada neurona en cada capa de la red.

La Figura 3.17 muestra dos ejemplos de cómo se propaga la señal a través de la red, donde los pesos ω_{mn} corresponden a la conexión de la salida de la neurona m con la entrada de la neurona n en la capa siguiente.

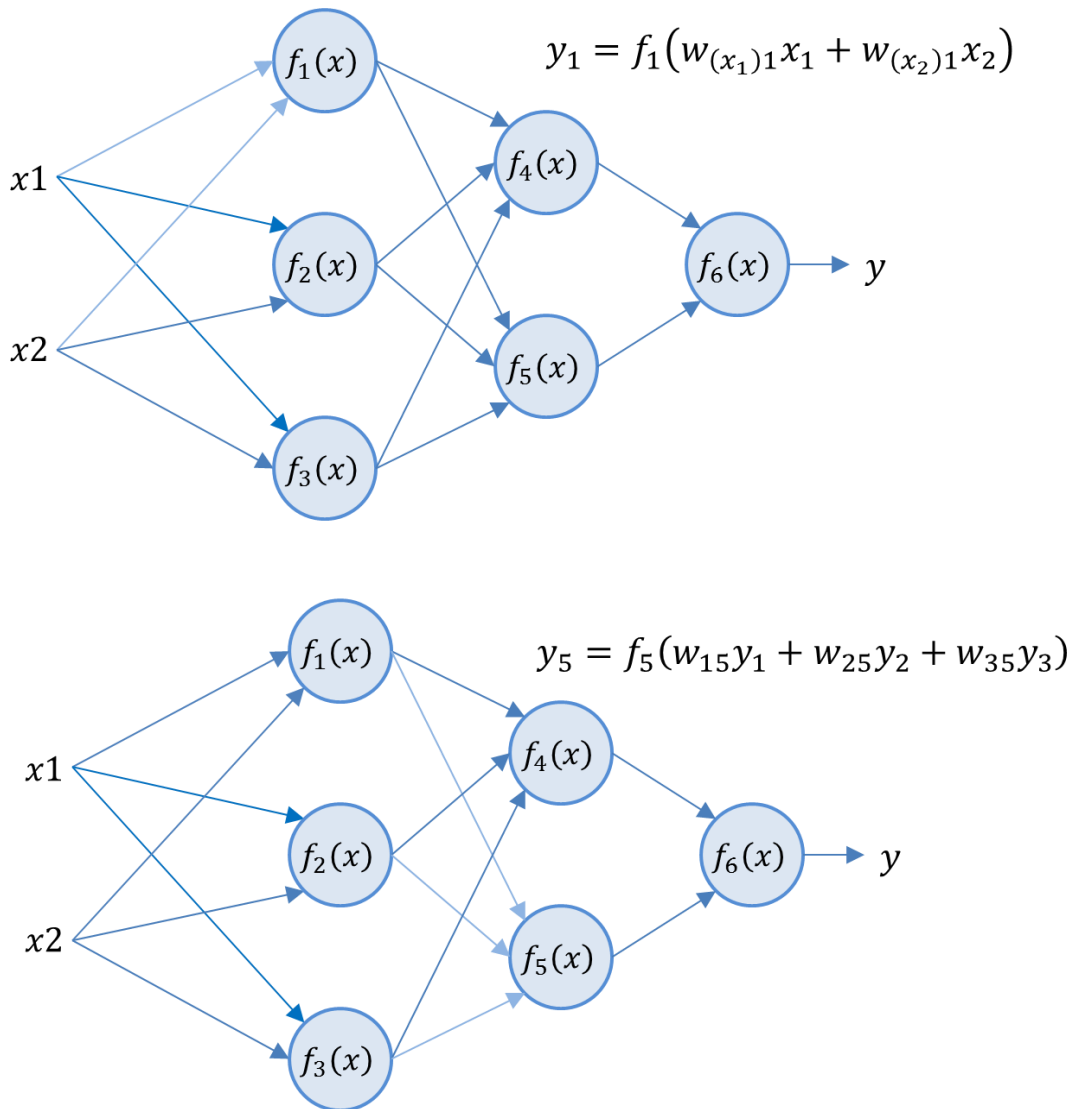


Figura 3.17 Propagación de las señales en las neuronas

En el siguiente paso del algoritmo, la salida de la red es comparada con el valor objetivo deseado. La diferencia se denomina error de la señal (δ). Es imposible conocer el error de

las neuronas de las capas internas directamente, debido a que se desconocen los valores de salida de estas neuronas. El algoritmo de retropropagación propaga el error de riesgo a todas las neuronas, cuya salida fue la entrada de la última neurona (Haykin, 1994; Klefors, 1998), como se muestra en la Figura 3.18.

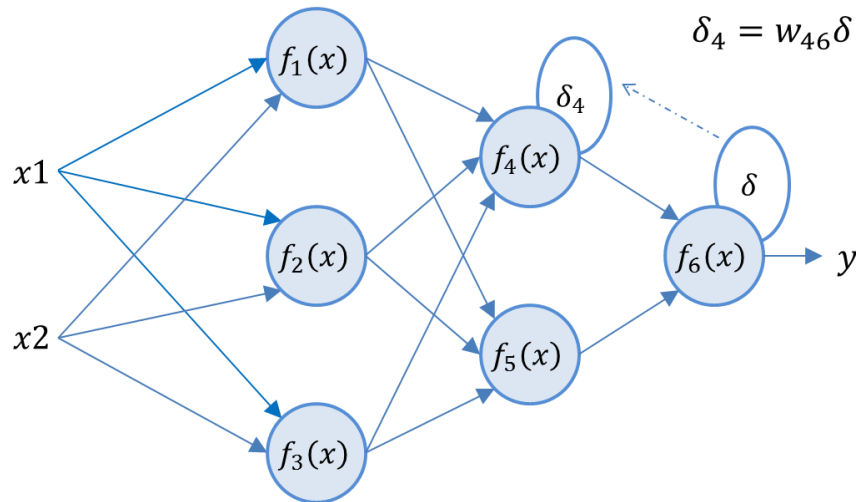


Figura 3.18 Retropropagación del error

Posteriormente el error se va propagando a las neuronas de capas anteriores, considerando los pesos de las conexiones, según se aprecia en la Figura 3.19.

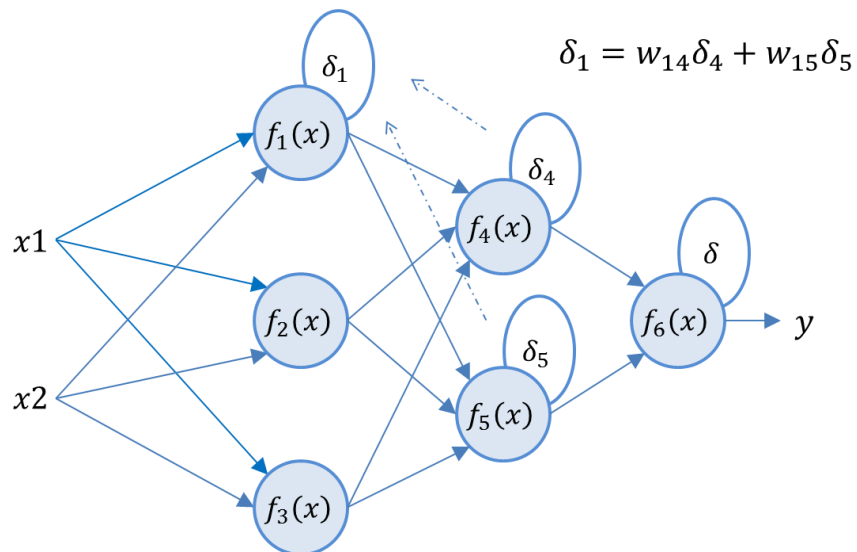


Figura 3.19 Retropropagación del error hacia capas anteriores

Cuando se calcula el error para cada neurona, los pesos de entrada pueden modificarse según los ejemplos de la Figura 3.20. Los coeficientes η afectan la velocidad de aprendizaje y pueden seleccionarse por distintos métodos (Freeman & Skapura, 1991; Haykin, 1994). Uno de ellos implica que al inicio del proceso de entrenamiento se elige un valor grande, el cual va descendiendo gradualmente conforme avanza el proceso. Otro método comienza con parámetros pequeños que aumentan a medida que el proceso avanza y nuevamente disminuye en la etapa final. Comenzar el proceso con un parámetro pequeño permite el establecimiento de los signos de pesos.

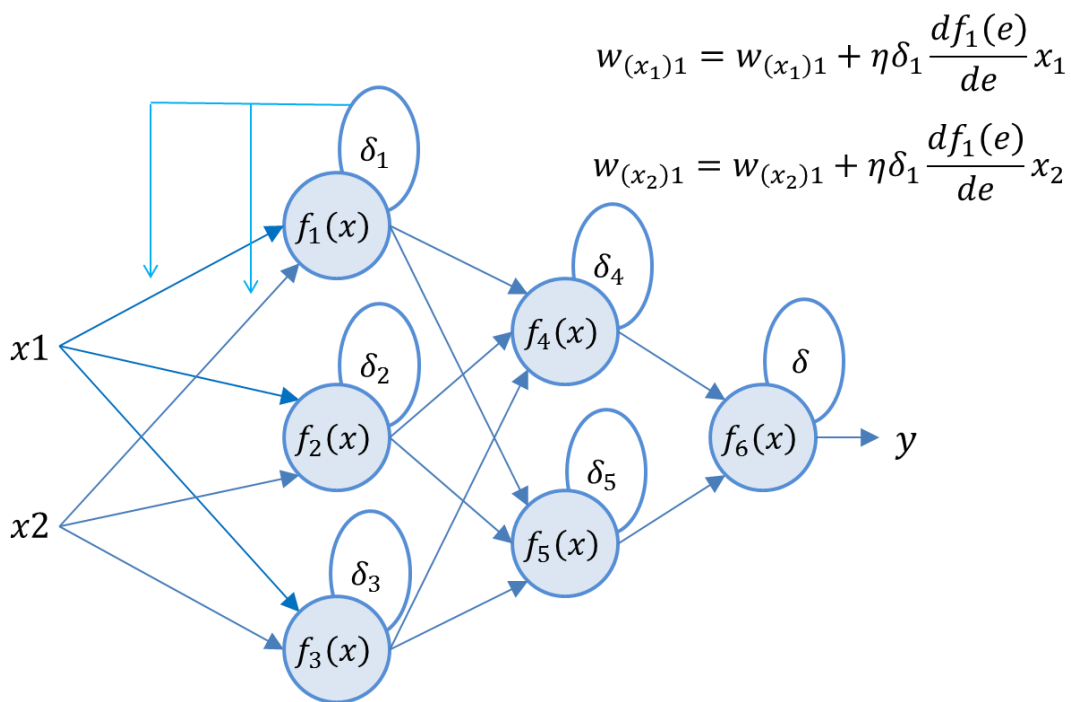


Figura 3.20 Actualización de los pesos

Capítulo 4

Interpolación Polinómica Segmentaria

La construcción de polinomios de interpolación de grado alto aunque es justificable teóricamente, plantea muchos problemas. Por un lado, la forma de la función polinómica de grado alto a menudo no responde al fenómeno debido al gran número de extremos e inflexiones. Por otro lado, su cálculo es muy complicado, lo que limita su utilidad en análisis numérico. Es a menudo más conveniente dividir el intervalo de interés en subintervalos más pequeños y usar en cada subintervalo polinomios de grado relativamente bajo, tratando de que la función a trozos definida de este modo tenga un aspecto final adecuado al fenómeno que estamos representando. La idea central es que en vez de usar un solo polinomio para interpolar los datos, podemos usar segmentos de polinomios y unirlos adecuadamente para formar nuestra interpolación. Podemos decir, que una función *spline* está formada por varios polinomios, cada uno definido en un intervalo y que se unen entre si bajo ciertas condiciones de continuidad.

En este capítulo veremos la teoría referente a la interpolación por spline, su clasificación y profundizaremos en las características de Spline Cúbico como método de interpolación segmentaria.

4.1 Interpolación con Splines

Una *función spline* está formada por varios polinomios, cada uno definido sobre un subintervalo, que se unen entre sí obedeciendo a ciertas condiciones de continuidad (Chapra & Canale, 1988; Maron & López, 1999; Nieves & Domínguez, 2002).

Supongamos que disponemos de $n + 1$ puntos, a los que denominaremos *nodos*, tales que:

$$t_0 < t_1 < \dots < t_n \tag{4.1}$$

Supongamos además que se ha fijado un entero $k \geq 0$. Decimos entonces que una función spline de grado k con nodos en t_0, t_1, \dots, t_n es una función S que satisface las condiciones (Sheid, 1991):

En cada intervalo $[t_{i-1}, t_i)$, S es un polinomio de grado menor o igual a k .

S tiene una derivada de orden $(k - 1)$ continua en $[t_0, t_n]$

Los splines de grado 0 son funciones constantes por zonas. Una forma explícita de presentar un spline de grado 0 es la siguiente:

$$S(x) = \begin{cases} S_0(x) = c_0, & x \in [t_0, t_1) \\ S_1(x) = c_1, & x \in [t_1, t_2) \\ \cdot & \cdot \\ \cdot & \cdot \\ S_{n-1}(x) = c_{n-1}, & x \in [t_{n-1}, t_n) \end{cases} \tag{4.2}$$

Los intervalos $[t_{i-1}, t_i)$ no se intersectan entre sí, por lo que no hay ambigüedad en la definición de la función en los nodos. Un spline de grado 1 se puede definir por (Burden, Faires, & Reynolds, 1991; Henrici, 1972):

$$S(x) = \begin{cases} S_0(x) = a_0x + b_0, & x \in [t_0, t_1) \\ S_1(x) = a_1x + b_1, & x \in [t_1, t_2) \\ \vdots & \vdots \\ S_{n-1}(x) = a_{n-1}x + b_{n-1}, & x \in [t_{n-1}, t_n) \end{cases} \quad (4.3)$$

En la Figura 4.1 se muestra la gráfica correspondiente a un spline de grado cero y en la Figura 4.2 una gráfica correspondiente a un Spline de grado 1.

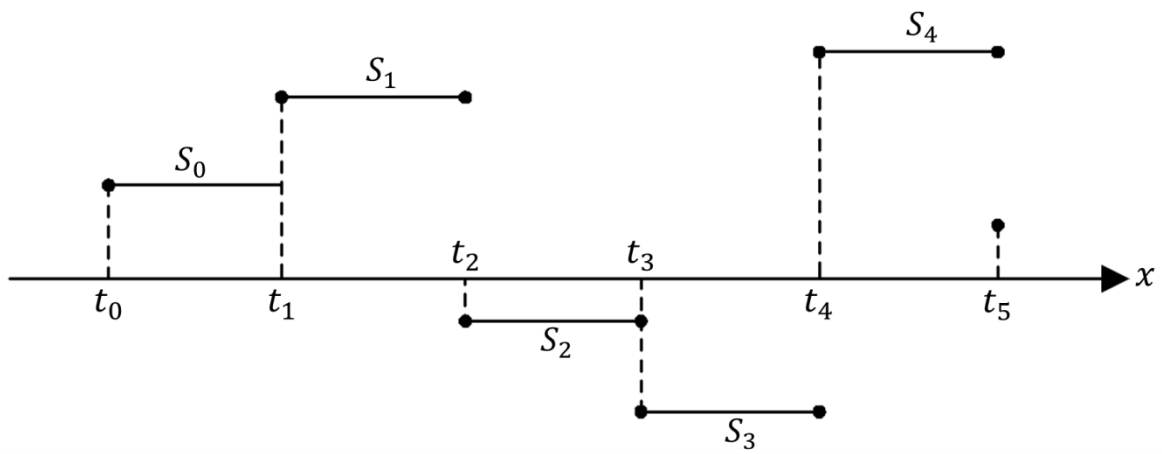


Figura 4.1 Spline de Grado 0

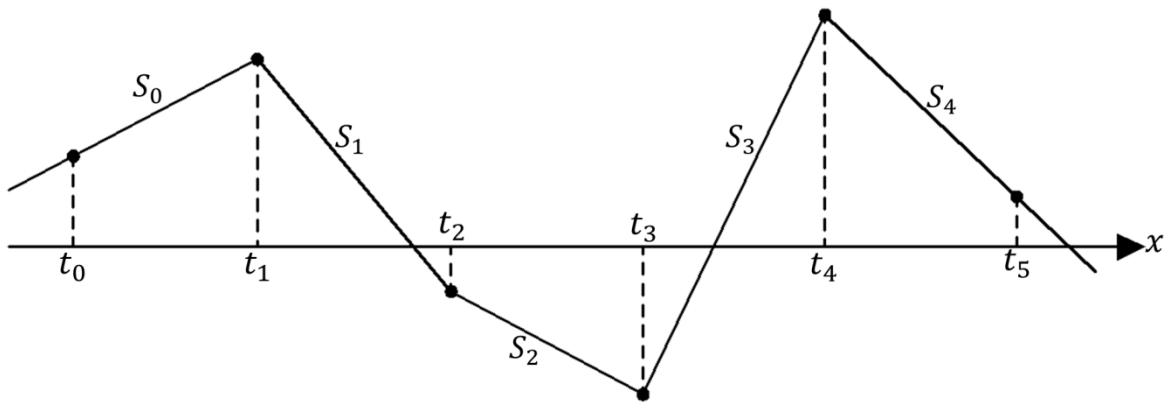


Figura 4.2 Spline de Grado 1

4.2 Splines Cúbicos

El spline cúbico ($k = 3$) es el spline más empleado, debido a que proporciona un excelente ajuste a los puntos tabulados y su cálculo no es excesivamente complejo (Chapra & Canale, 1988; Hildebrand, 1974; Kincaid & Cheney, 1994; Nieves & Domínguez, 2002).

Sobre cada intervalo $[t_0, t_1], [t_1, t_2], \dots, [t_{n-1}, t_n]$, S está definido por un polinomio cúbico diferente. Sea S_i el polinomio cúbico que representa a S en el intervalo $[t_i, t_{i+1}]$, por tanto:

$$S(x) = \begin{cases} S_0(x) = S_0, & x \in [t_0, t_1) \\ S_1(x) = S_1, & x \in [t_1, t_2) \\ \cdot & \cdot \\ \cdot & \cdot \\ S_{n-1}(x) = S_{n-1}, & x \in [t_{n-1}, t_n) \end{cases} \quad (4.4)$$

Los polinomios S_{i-1} y S_i interpolan el mismo valor en el punto S_i , es decir, se cumple:

$$S_{i-1}(t_i) = y_i = S_i(t_i) \quad (1 \leq i \leq n-1) \quad (4.5)$$

por lo que se garantiza que S es continuo en todo el intervalo. Además, se supone que S' y S'' son continuas, condición que se emplea en la deducción de una expresión para la función del spline cúbico (Conte & de Boor, 1980; Curtis & Patrick, 2003).

Aplicando las condiciones de continuidad del spline S y de las derivadas primera S' y segunda S'' , es posible encontrar la expresión analítica del spline. La expresión resultante es:

$$h_{i-1}z_{i-1} + 2(h_i + h_{i-1})z_i + h_i z_{i+1} = \frac{6}{h_{i-1}}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}) \quad (4.6)$$

En la expresión anterior, $h_i = t_{i+1} - t_i$ y z_0, z_1, \dots, z_n son incógnitas. Para determinar sus valores, utilizamos las condiciones de continuidad que deben cumplir estas funciones. El resultado es:

$$h_{i-1}z_{i-1} + 2(h_i + h_{i-1})z_i + h_i z_{i+1} = \frac{6}{h_{i-1}}(y_{i+1} - y_i) - \frac{6}{h_{i-1}}(y_i - y_{i-1}) \quad (4.7)$$

La ecuación anterior, con $i = 1, 2, \dots, n - 1$ genera un sistema de $n - 1$ ecuaciones lineales con $n + 1$ incógnitas z_0, z_1, \dots, z_n . Podemos elegir z_0 y z_1 de forma arbitraria y resolver el sistema de ecuaciones resultante para obtener los valores de z_1, z_2, \dots, z_{n-1} . Una elección especialmente adecuada es hacer $z_0 = z_1 = 0$. La función spline resultante se denomina *spline cúbico natural* y el sistema de ecuaciones lineal expresado en forma matricial es:

$$\begin{pmatrix} u_1 & h_1 & & & & & & & & & \\ h_1 & u_2 & h_2 & & & & & & & & \\ & h_2 & u_3 & h_3 & & & & & & & \\ & & \ddots & \ddots & \ddots & & & & & & \\ & & & h_{n-3} & u_{n-2} & h_{n-2} & & & & & \\ & & & & h_{n-2} & u_{n-1} & & & & & \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_{n-2} \\ z_{n-1} \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{n-2} \\ v_{n-1} \end{pmatrix} \quad (4.8)$$

en donde:

$$\begin{aligned} h_i &= t_{i+1} - t_i \\ u_i &= 2(h_i + h_{i-1}) - \frac{h_{i-1}^2}{u_{i-1}} \\ b_i &= \frac{6}{h_i}(y_{i+1} - y_i) \\ v_i &= b_i - b_{i-1} - \frac{h_{i-1}v_{i-1}}{u_{i-1}} \end{aligned} \quad (4.9)$$

Capítulo 5

Propuesta para la detección de deformaciones

En este apartado, se describe el método propuesto para la detección de deformaciones en superficies curvas, abordando de manera puntual cada una de las etapas que lo conforman; realizando la evaluación sobre superficies curvas, como las encontradas en el cascaron de huevos avícolas irregulares.

El método propuesto se basa en el principio de detección de defectos mediante la generación de un barrido de luz estructurada laser en forma de línea y su proyección sobre la superficie del cascaron de un huevo, registrando imágenes del haz laser deformado al entrar en contacto con la superficie analizada. Tras la adquisición de las imágenes del barrido laser, se realiza un procesamiento que permite detectar y clasificar las muestras deformes.

La clave de la identificación de los defectos, reside en la obtención de puntos distribuidos en la curvatura de una imagen para realizar una interpolación por spline cúbicos, estimando de esta manera una curva representativa “ideal” de como debiera ser la estructura del huevo analizado si no presentara deformaciones; una posterior comparación de ambas curvas de manera gráfica permite identificar diferencias geométricas sobre la superficie de un huevo, realizando mediciones en base a pixeles de la longitud y área entre dichas curvaturas, haciendo posible la aplicación de una red neuronal artificial para obtener una clasificación de defectos en las muestras analizadas.

5.1 Etapas del Método

El proceso de evaluación y análisis de superficies curvas para la detección de deformaciones, está conformado mediante una serie de etapas que intervienen en la adquisición de la imagen, procesamiento y análisis de los patrones geométricos adquiridos.

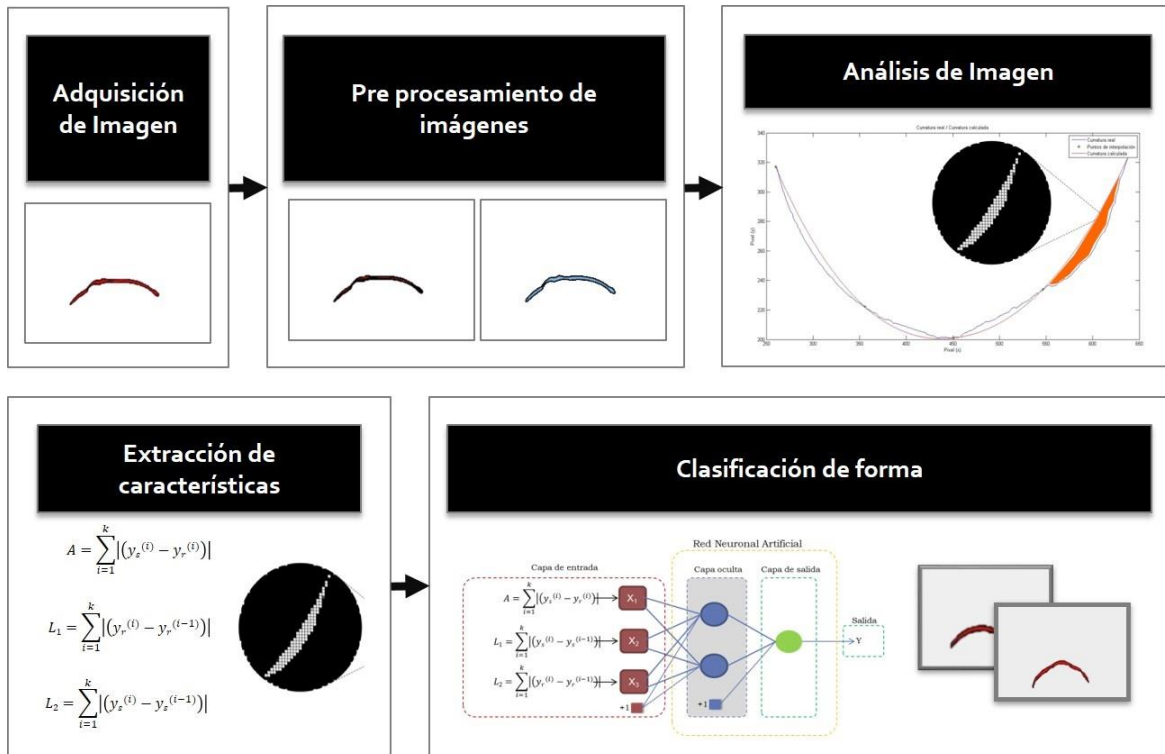


Figura 5.1 Etapas del método

El proceso de evaluación de las muestras de superficies curvas está dividido en 5 etapas como se muestra en la Figura 5.1. A continuación se describe brevemente las etapas del método y en las secciones subsecuentes profundizaremos en cada una de ellas.

Adquisición de imágenes: En esta etapa inicial se realiza la adquisición de imágenes utilizadas para el análisis, este proceso se lleva a cabo empleando un sistema de visión conformado por un patrón de luz estructurada laser en forma de línea, el cual se proyecta sobre la superficie del cascaron de un huevo y mediante una cámara fotográfica digital se capturan las imágenes resultantes de la adaptación del láser sobre la curvatura del huevo.

Pre procesamiento de la imagen: El pre procesamiento de imágenes en la aplicación de técnicas de visión artificial es crucial para corregir y resaltar características importantes en una imagen. En esta etapa del método se realiza una serie de ajustes en las imágenes capturadas previamente, resaltando los rasgos característicos que definen la curvatura de un huevo como consecuencia de proyectar el patrón laser lineal.

Análisis de la imagen: Esta etapa parte del pre procesamiento de la imagen y es la encargada de resaltar los defectos en las curvaturas registradas. La aplicación de este análisis está basado en la obtención de una curvatura “ideal” de la superficie analizada si esta no presentara irregularidades, partiendo del principio de interpolación polinómica segmentaria utilizando Spline Cúbicos; mediante esta técnica es posible obtener una curva suavizada a partir de puntos adquiridos de la curvatura analizada; de esta manera, hace posible realizar una comparación gráfica entre ambas curvas.

Extracción de características: En el procesamiento de imágenes, es necesario determinar características más significativas que definan las propiedades de cada objeto de estudio, con el propósito de disminuir la cantidad de información empleada para el análisis matemático o estadístico, de esta manera, se disminuye en gran medida la carga computacional que esto conlleva. Partiendo de estas consideraciones y de la etapa anterior, se efectúa la extracción de características de interés en la imagen, las cuales son definidas como mediciones en pixeles de *longitud de curva real*, *longitud de curva interpolada* y *área entre curvas*.

Clasificación de forma: La etapa final del método para la detección de deformaciones en superficies curvas, está basado en la aplicación de una técnica de inteligencia artificial denominada *redes neuronales artificiales*; esta técnica es una analogía con el modelo biológico humano. En esta etapa se emplea una red neuronal artificial como clasificador; el funcionamiento radica en entrenar a la RNA¹¹ con una base de conocimiento, que está compuesta por una serie de imágenes de huevos normales y con deformaciones, de esta manera la mediciones adquiridas en la etapa previa son evaluadas por la *red* para realizar la clasificación de las muestras en dos conjuntos según sus características geométricas.

¹¹ RNA (en español: *Red Neuronal Artificial*. en ingles ANN: *Artificial Neural Network*)

5.2 Adquisición de imagen

En esta etapa inicial se realiza la adquisición de imágenes utilizadas para el análisis de las deformaciones en superficies curvas, este proceso se lleva a cabo empleando un arreglo experimental del sistema de visión, conformado por un patrón de luz estructurada láser en forma de línea, el cual se proyecta sobre la superficie del cascarón de un huevo y mediante una cámara fotográfica digital CCD se capturan las imágenes resultantes de la adaptación del láser sobre la curvatura del huevo.

5.2.1 Arreglo experimental

La Figura 5.2 muestra el arreglo experimental del sistema de visión empleado durante el análisis de los defectos en las superficies curvas y a continuación se describen cada una de las partes que lo conforman.

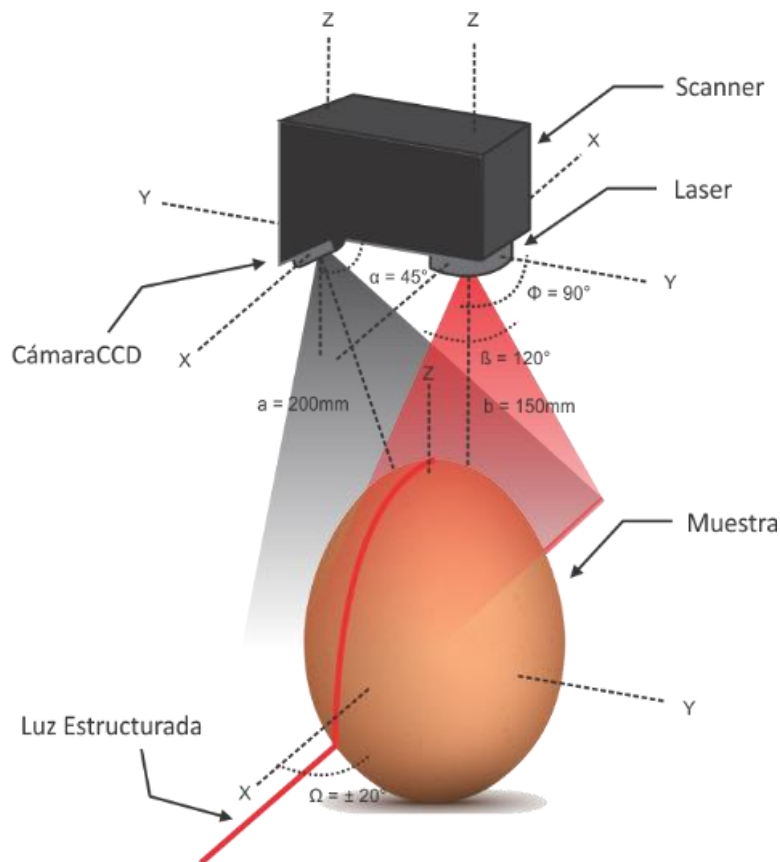


Figura 5.2 Diagrama esquemático del sistema de visión propuesto

Scanner: Cuerpo del sistema que contiene el láser y la cámara CCD para las capturas de las imágenes.

Laser: Elemento óptico que genera un patrón de luz estructurada en forma de línea a 650nm, con una apertura óptica de proyección de 120° a una distancia de 150mm sobre la muestra analizada.

Luz estructurada: Es el patrón óptico resultante proyectado por el láser, mediante el cual se resaltan y se hacen visibles las posibles deformaciones que estén distribuidas sobre la superficie del huevo.

Cámara CCD: Mediante esta cámara es posible capturar imágenes en formato RGB, que muestran las posibles deformaciones del haz laser al entrar en contacto con la superficie deformada, está equipada con una lente de 38mm de distancia focal, una resolución de 1024 x 768 pixeles y ubicada a una distancia de 200mm respecto al objeto analizado.

5.2.2 Registro de Imágenes

Aplicando el arreglo experimental del sistema de visión descrito previamente, se efectúa la proyección superior del patrón de luz estructurada laser en forma de línea a 650nm con una apertura de 120° a una distancia de 150 mm sobre la superficie de un huevo, manteniendo las condiciones de iluminación del espacio de pruebas en concentraciones menores a 1 lux. Una vez proyectado el patrón laser se procede a realizar un barrido longitudinal a través de la superficie del cascaron; al entrar en contacto el láser con la muestra, este es deformado y se adapta a la curvatura del huevo, si la muestra evaluada presenta deformaciones, el patrón proyectado se deforma generando una curvatura irregular, resaltando de esta manera las posibles imperfecciones en el huevo.

En cada sección analizada de la superficie del huevo se realizan capturas de imágenes mediante una cámara fotográfica situada a 200mm y dispuesta en un ángulo de 45°. Las imágenes resultantes que exponen la forma geométrica de la muestra analizada debido a la proyección del patrón estructurado, son empleadas en las etapas subsecuentes del método para obtener la información relevante que describa los defectos de las curvas analizadas.

En la Figura 5.3, se observa el resultado de proyectar el patrón laser sobre la superficie de un huevo normal o sin deformaciones, donde se aprecia una curva suave resultante de la adaptación de la proyección laser.

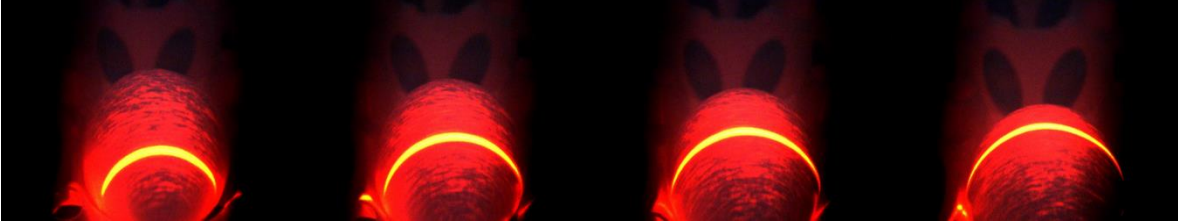


Figura 5.3 Huevos sin deformaciones

Por el contrario en la proyección del patrón laser sobre una superficie deformada, se observa como el patrón es deformado y sobresalta la irregularidad en la curvatura del huevo (Figura 5.4).

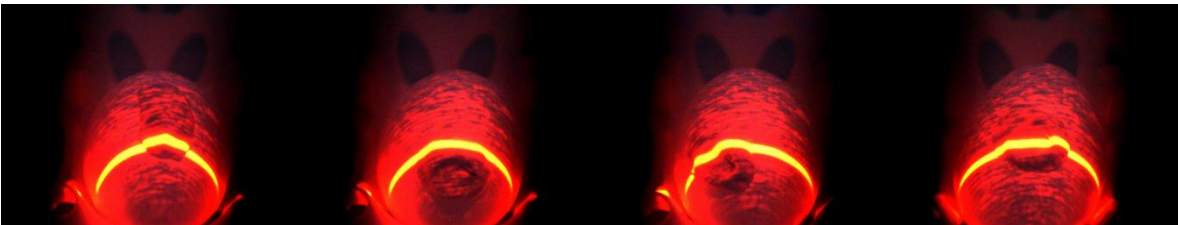


Figura 5.4 Huevos con deformaciones

En el procesamiento de imágenes es necesario capturar una imagen con las mejores características y que contengan la menor cantidad de información que afecte su análisis; por tal motivo se realiza un ajuste del obturador para disminuir la cantidad de luz que ingresara al CCD, de esta manera se elimina parte del resplandor generado por el láser, dando como resultado una imagen clara, donde se enfatiza únicamente la curva de interés (Figura 5.5).



Figura 5.5 Imágenes de las muestras, posterior al ajuste del obturador

5.3 Pre procesamiento de la imagen

En esta etapa del método se realiza un pre procesamiento de la imagen con el objetivo de eliminar alteraciones o ruido en la escena, de esta manera se resaltan los rasgos característicos que definen la curvatura de un huevo como consecuencia de proyectar el patrón laser lineal.

Las imágenes digitales de las muestras sin deformaciones y con deformaciones como las presentadas en la Figura 5.5, fueron adquiridas por la cámara del sistema de visión y almacenadas en formato .jpg. El pre procesamiento de las imágenes se realiza mediante el software MATLAB® (MathWorks, Inc., USA) empleando una computadora Hewlett Packard Pavilion dm4 (Windows® 7 Professional, Intel® Core™ i5, 8 GB, Intel® HD, Graphics 3000, HD 500 GB).

A continuación se presenta en un diagrama a bloques las etapas del algoritmo desarrollado para el pre procesamiento de la imagen (Figura 5.6), utilizando el software MATLAB® como entorno de programación para el tratamiento de las imágenes.

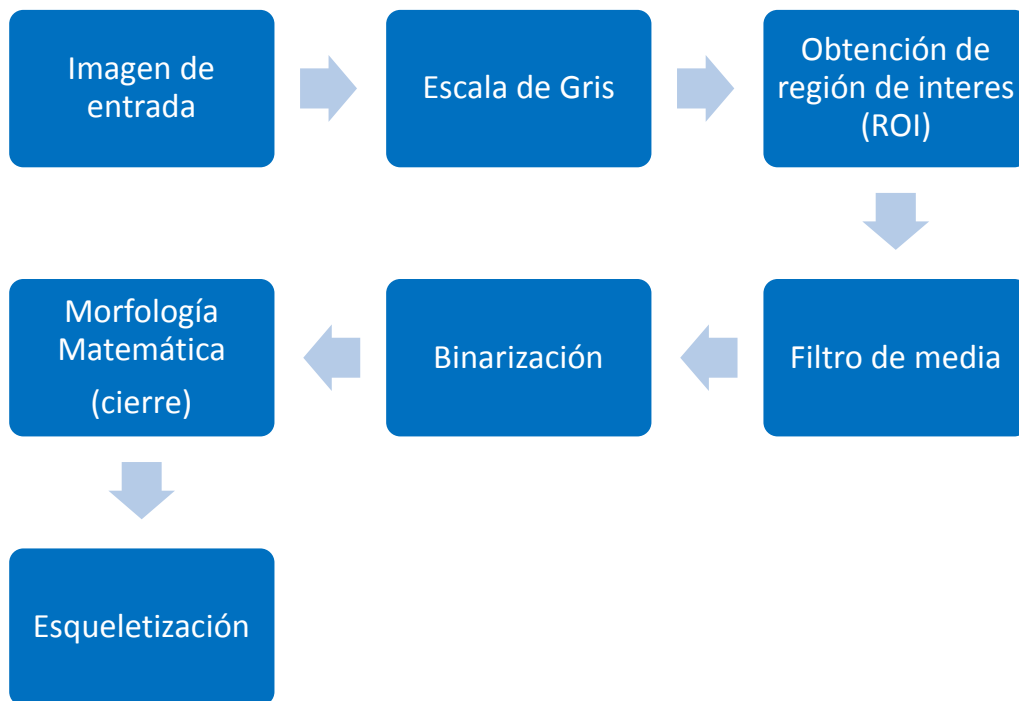


Figura 5.6 Diagrama a bloques del pre procesamiento de una imagen

5.3.1 Imagen de entrada

En esta etapa del pre procesamiento de imágenes, las imágenes son leídas del dispositivo de almacenamiento al ambiente de MATLAB® usando la función *imread*. La sintaxis empleada es la siguiente:

```
img = imread('HB1.jpg');
```

donde el nombre del archivo “HB” determina la imagen del grupo sin deformaciones, “HM” el grupo compuesto por huevos con deformaciones, el dígito contiguo “1” hace referencia al índice o número de imagen y finaliza con el tipo o extensión de la imagen “.jpg”.



Figura 5.7 Imágenes de entrada

La Figura 5.7 muestra el tipo de imágenes adquiridas por el algoritmo para el análisis de las deformaciones. Las imágenes registradas son almacenadas como matrices de píxeles que contienen los valores de las componentes de color RGB para su uso en las etapas posteriores. En lo sucesivo del pre procesamiento de imágenes, tomaremos de referencia la imagen de la curva de un huevo con deformaciones (Figura 5.8) para presentar las diversas etapas de ajuste y observar los resultados obtenidos en cada caso.



Figura 5.8 Imagen de una curva irregular, obtenida de un huevo deformado

5.3.2 Escala de Gris

En el procesamiento de imágenes, una escala de grises es una escala empleada en la imagen digital en la que el valor de cada píxel posee un valor equivalente a una graduación de gris. Las imágenes representadas de este tipo están compuestas de sombras de grises, que van desde el negro más profundo variando gradualmente en intensidad de grises hasta llegar al blanco. MATLAB® dispone de funciones especiales para convertir imágenes de color a escala de grises.

En esta etapa del pre procesamiento de la imagen, se realiza un ajuste de la imagen de entrada en formato RGB (Figura 5.8) a una representación a escala de grises, empleando la función *rgb2gray*.

```
img = imread('HM36.jpg');  
imagen_gray = rgb2gray(img);
```

La función *rgb2gray*, recibe como parámetro la matriz de una imagen en formato RGB, regresando una matriz de píxeles en escala de grises comprendida en un rango de intensidad de 0 a 255.

En la Figura 5.9 se realiza una comparación de la imagen de entrada en formato RGB y su equivalente en escala de grises posterior a la aplicación de la función *rgb2gray*; esta imagen resultante será utilizada en las etapas posteriores del pre procesamiento.



Figura 5.9 Representación de imágenes. (a) RGB. (b) Escala de Gris

5.3.3 Región de interés

Normalmente, la información que deseamos extraer de una imagen no se encuentra aleatoriamente por toda la imagen, se encuentra concentrada en un región de la imagen. Esta región se denomina ROI (region of interest) o región de interés. De esta manera, conseguimos reducir el tamaño de la imagen, teniendo menos píxeles a recorrer. En la Figura 5.10a se muestra marcada por un recuadro la región de interés para nuestro caso de estudio.



Figura 5.10 Extracción de Región de Interés (ROI). (a) Delimitación ROI. (b) Imagen resultante

La extracción de esta región de interés ROI, parte de la imagen en escala de gris obtenida en la etapa previa; se definen las coordenadas de los límites de la región que se desea extraer mediante c y r , finalmente se crea una nueva imagen contenida en las coordenadas de los puntos definidos y se guarda en una variable.

```
imagen_gray = rgb2gray(img);
c = [200 800];
r = [400 750];
imagen_gray = imagen_gray(r(1):r(2),c(1):c(2));
```

Al extraer de la imagen inicial nuestra ROI, obtenemos una nueva imagen reducida (Figura 5.10b) que contiene la parte donde se localiza la curvatura registrada por la proyección laser que analizaremos posteriormente.

5.3.4 Binarización

Una imagen binaria es una imagen en la cual cada píxel puede tener solo uno de dos valores posibles 1 o 0. Como es lógico suponer en una imagen con esas condiciones es mucho más fácil encontrar y distinguir características estructurales. En visión computacional el trabajo con imágenes binarias es muy importante para realizar segmentación por intensidad de la imagen, para generar algoritmos de reconstrucción o reconocer estructuras.

La forma más común de generar imágenes binarias es mediante la utilización del valor umbral de una imagen a escala de grises; es decir se elige un valor límite (o bien un intervalo) a partir del cual todos los valores de intensidades mayores serán codificados como 1, mientras que los que estén por debajo serán codificados a 0, obteniendo un umbral óptimo y una binarización adecuada a la imagen. El proceso de binarización realizado mediante la aplicación de MATLAB® se lleva a cabo en el siguiente fragmento de código:

```
umb = graythresh(imagen_gray);  
imagen_bin = im2bw(imagen_gray,umb);
```

En la primer línea se realiza la obtención de un umbral óptimo adquirido de la propia imagen con la función *graythresh*, enviando de parámetro la imagen resultante de la conversión a escala de gris, regresando el valor correspondiente al umbral *umb*; una vez obtenido este punto de ajuste se realiza una binarización de la imagen, utilizando la función *im2bw*, a esta función se le asigna de parámetros de entrada la matriz de la imagen en escala de gris y el valor umbral, regresando una matriz *imagen_bin* en formato binario. La Figura 5.11, muestra el resultado obtenido al aplicar una binarización.



Figura 5.11 Binarización de imagen de una muestra deformada. (a) Escala de gris. (b) Binarizada

5.3.5 Filtro media

Debido a que el ruido es una información no deseada que contamina la imagen, se aplicó un filtro de media para eliminar los posibles puntos dispersos contenidos en algunas imágenes que no fueron retirados al realizar el ajuste del obturador de la cámara.

La siguiente función MATLAB® *imfilter* es utilizada para realizar el filtrado de la imagen:

```
imagen_bin = im2bw(imagen_gray,umb);
imagen_bin = imfilter(imagen_bin ,fspecial('average'));
```

En la función *imfilter*, es necesario enviar como parámetros de entrada la matriz binaria de la imagen *imagen_bin* y la definición del tipo de filtro que se desea aplicar a la imagen *imagen_bin,fspecial('average')*; esta función regresa una nueva imagen binaria filtrada.



Figura 5.12 Filtro de media

Como resultado de la aplicación del filtro y las características iniciales en la curva resultante de la proyección del patrón laser sobre la superficie del cascaron del huevo, es necesario cerrar posibles espacios que se presente en la imagen como el marcado en la Figura 5.13, esto fue solucionado en la etapa siguiente, aplicando morfología matemática.



Figura 5.13 Identificación de cortes en una curva

5.3.6 Morfología matemática

La palabra morfología denota una rama de la Biología que se ocupa de la forma y estructura de los animales y plantas. La morfología matemática está basada en la geometría y la forma. Las operaciones morfológicas simplifican las imágenes y preservan las formas principales de los objetos.

Usando los operadores elementales de erosión y dilatación se pueden diseñar operaciones de realce de las formas de los objetos. La dilatación binaria opera aumentando el tamaño de los objetos, cerrando los agujeros y grietas. El ensanchamiento de los objetos puede ser reducido mediante la aplicación seguida de una erosión. La operación combinada de dilatación y erosión es denominada cierre (*closing*).

En esta etapa se lleva a cabo la aplicación de una operación de cierre, para compensar la posible ausencia de píxeles de la imagen con el propósito de conseguir una curvatura más definida, conservando las principales características del objeto (Figura 5.13). El comando de MATLAB® utilizado para este caso es el expresado a continuación:

```
H = strel('disk', 20);
imagen_close = imclose(imagen_bin, H);
```

En la aplicación de la etapa de cierre, primeramente es necesario definir el tipo de elemento estructurante denominado STREL¹², con el cual se realizarán las operaciones de dilatación y erosión de la imagen; mediante pruebas realizadas a la imagen con diferentes estructuras y la naturaleza de su forma, se determinó que el tipo de elemento estructurante óptimo para este caso es uno tipo disk (disco), de radio 20. Una vez definido el STREL por la función $H = \text{strel}('disk', 20)$, se procede a la aplicación de la función correspondiente para el cierre de la imagen, definida por $\text{imagen_close} = \text{imclose}(\text{imagen_bin}, H)$, a esta función se envían de parámetros la imagen binaria *imagen_bin* y las características del STREL definido previamente, obteniendo por resultado una imagen “cerrada”, como la observada en la Figura 5.14.

¹² STREL (en inglés: Structuring Element. en español: Elemento estructurante).

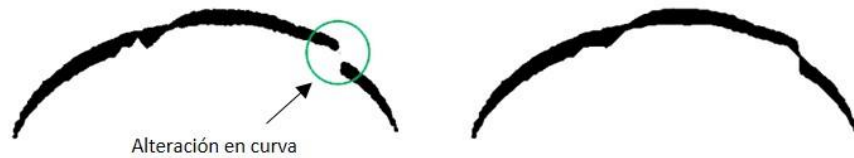


Figura 5.14 Resultado de la operación morfológica (cierre)

5.3.7 Esqueletización

Por definición, la esqueletización es el proceso iterativo de quitar la mayor cantidad de píxeles de una imagen, sin afectar a la forma general del patrón, es decir, una vez eliminados los píxeles, el patrón debe ser reconocido.

En esta última etapa del pre procesamiento de la imagen, se aplica una esqueletización para obtener una mínima representación de la imagen original. Esta evaluación se realiza mediante la programación de un algoritmo en MATLAB®.

El procedimiento que realiza el algoritmo inicia a partir de una matriz binaria de entrada que representa la imagen con una curvatura, una vez adquirida la matriz, se procede a realizar un análisis iterativo, haciendo un barrido de columnas y filas, de aquí la importancia de obtener una región de interés ROI para minimizar el número de recorridos en la matriz; por cada columna que incrementa el apuntador son evaluadas cada una de las filas en búsqueda de un píxel con valor 1 (blanco), una vez localizado se guarda en una nueva matriz la posición de este píxel en relación a sus coordenadas x y y en la imagen; la iteración de la matriz se realiza por cada columna subsecuente y sus respectivas filas que la compongan. Finalmente se evalúa la matriz de coordenadas resultante y se realiza el promedio de las coordenadas x y y de los píxeles blancos de cada columna, el valor obtenido se almacena como una nueva coordenada, creando de esta manera una nueva matriz que contiene un patrón continuo con la menor cantidad de datos posibles, pero que mantiene un mínimo rastro del objeto original.

A continuación se presenta el fragmento de código en MATLAB® utilizado para obtener la mínima representación de la curva en la imagen. La Función *esqueleto*, recibe como parámetro de entrada una matriz binaria de la imagen original, retornando una nueva matriz compuesta por las coordenadas x y y , así como el valor de cada pixel que conforma a la nueva imagen esqueletizada.

```
[pix,xdata,ydata] = esqueleto(imagen_bin);

function [pix,xdata,ydata] = esqueleto(x)
    pix = zeros(size(x));
    [R,C] = find(x > 0);
    xdata = [];
    ydata = [];
    for c = min(C):max(C)
        r = find(x(:,c) > 0);
        if length(r) > 0
            xdata = [xdata,c];
            r1 = mean(r);
            ydata = [ydata,r1];
            pix(round(r1),c) = 1;
        end
    end
end
```

La Figura 5.15, muestra el resultado final de las etapas de pre procesamiento en una imagen; se obtuvo una esqueletización como mínima representación de la curva evaluada, la cual es utilizada en la extracción de la matriz de pixeles empleada para el análisis mediante interpolación polinómica segmentaria por splines cúbicos, que trataremos en las siguientes secciones de este capítulo.



Figura 5.15 Esqueleto de la curva en una imagen. (a) Real. (b) Esqueleto

5.4 Análisis de la imagen

La dificultad del análisis de datos numéricos que no posean una secuencia definida hace necesario tener un método matemático para calcular las formas geométricas correspondientes a cierto agrupamiento de puntos que describan un objeto. Es decir, en este caso particular la obtención de las imágenes que muestran las curvaturas resaltadas por el patrón de luz estructurado laser, hace necesario emplear un método que a partir de la obtención de puntos tomados de la gráfica de la curva registrada se pueda generar la construcción de una nueva curva suavizada que describa la forma uniforme de la curvatura de un huevo sin irregularidades.

De acuerdo a lo anterior, para solucionar el problema presentado en el análisis de las deformaciones de los huevos, se utiliza una interpolación cúbica segmentaria ("spline cúbico"), asegurando la continuidad de la curva en los extremos de los intervalos al utilizar polinomios cúbicos entre cada par de datos. Este tipo de polinomios posee cuatro constantes, lo cual le confiere suficiente flexibilidad, logrando una unión suave entre los segmentos que forman la curva.

5.4.1 Representación gráfica de curvas

En esta etapa del método, partiremos del grupo de imágenes que se observan en la Figura 5.16, compuesto por dos representaciones de curvas regulares (a, b) y dos curvas irregulares (c, d), resultado de proyectar el patrón de luz estructura lineal sobre las superficie del huevo. Cada una de las imágenes del conjunto fue pre procesada hasta obtener su esqueleto.



Figura 5.16 Imágenes de curvaturas presentes en huevos. (a, b) Huevos regulares. (c, d) Huevos deformados

Graficando el conjunto de imágenes descritas previamente en un rango de píxeles, obtenemos la mínima representación, que usaremos como base para el análisis matemático.

Las siguientes gráficas, representan los pixeles distribuidos de las curvas sin deformaciones, presentes en las imágenes a y b de la Figura 5.16.

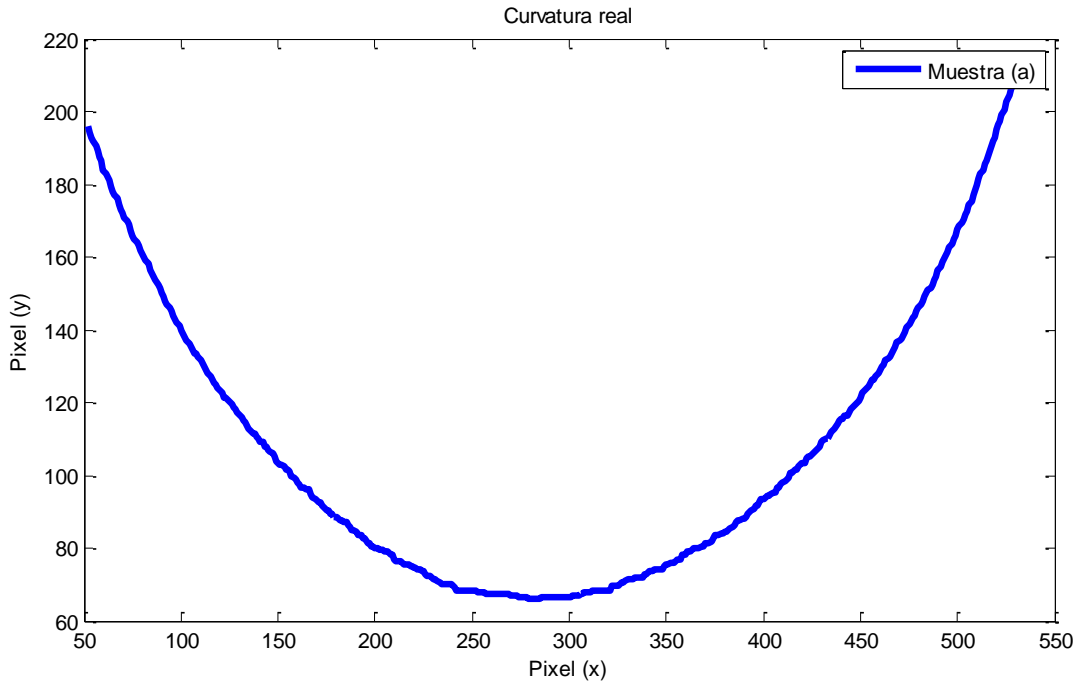


Figura 5.17 Distribución de pixeles en un plano, obtenidos de la imagen de la muestra (a)

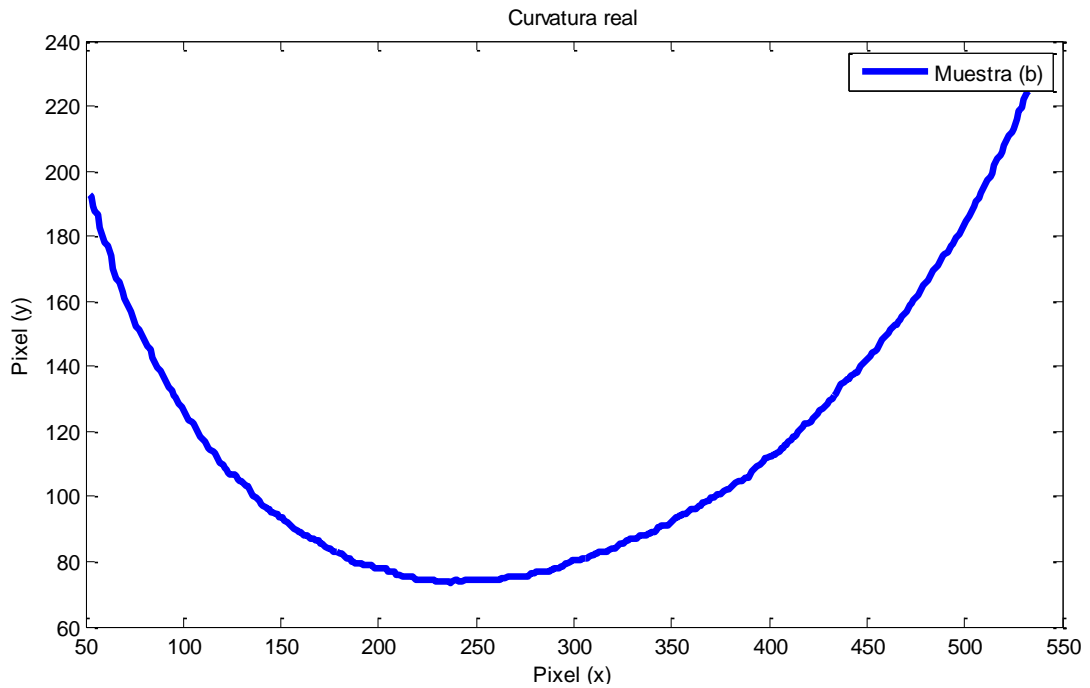


Figura 5.18 Distribución de pixeles en un plano, obtenidos de la imagen de la muestra (b)

A continuación se representan los pixeles distribuidos de las curvas deformadas, presentes en las imágenes *c* y *d* de la Figura 5.16, por medio de las siguientes gráficas.

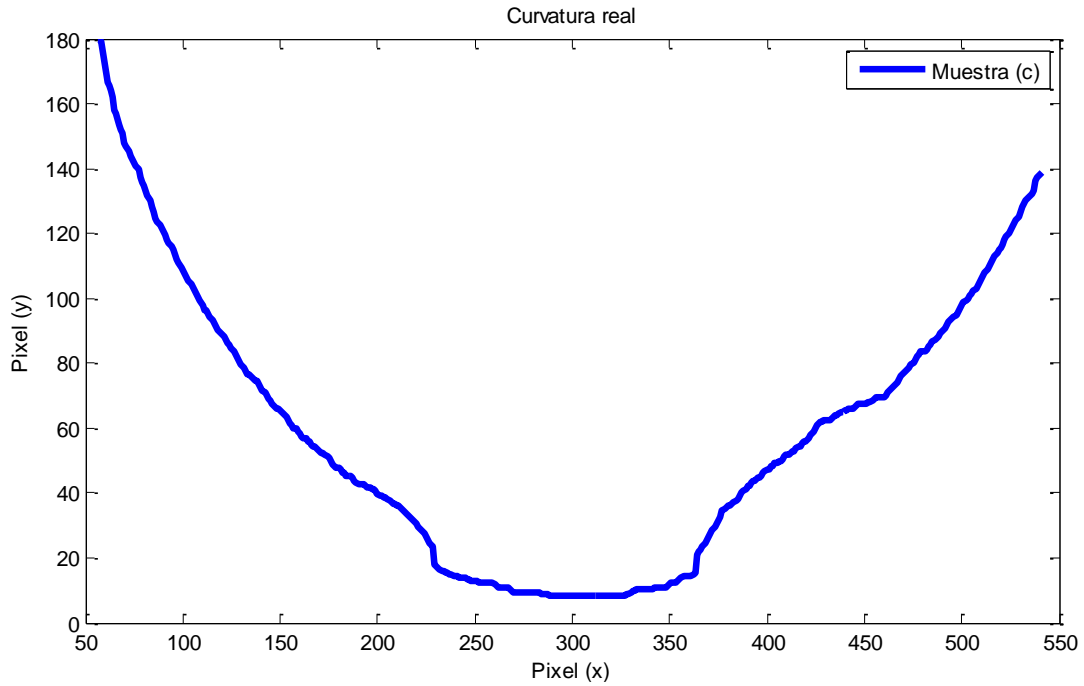


Figura 5.19 Distribución de píxeles en un plano, obtenidos de la imagen de la muestra (c)

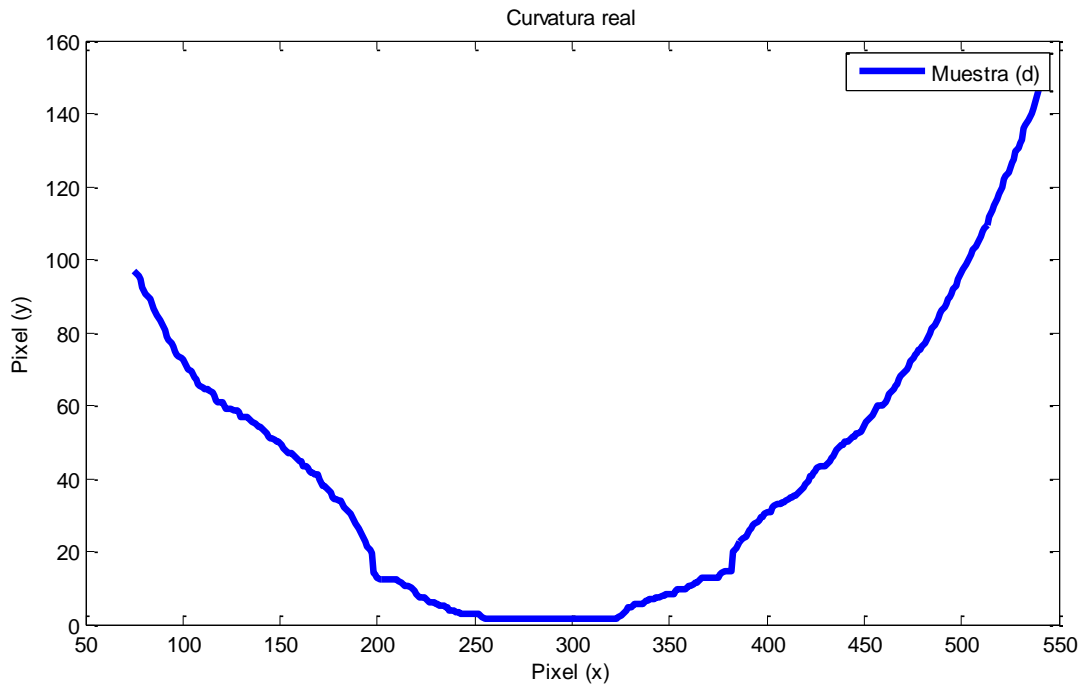


Figura 5.20 Distribución de píxeles en un plano, obtenidos de la imagen de la muestra (d)

5.4.2 Suavizado por Spline Cúbico

En la presente etapa se aplica una interpolación segmentaria mediante Splines Cúbicos, a cada una de las muestras graficadas anteriormente. Debido a que un spline cubico está basado en la unión de segmentos representados por polinomios, es necesario definir la cantidad de puntos que crearan estos segmentos utilizados en la interpolación.

Mediante el uso de MATLAB® como herramienta de análisis matemático, se desarrolla un algoritmo para obtener una interpolación por Splines Cúbicos a partir de puntos muestreados en la curvatura de cada una de las imágenes captadas por la cámara.

```
[pix, xdata, ydata] = esqueleto(imagen_bin);
cluster = 5;
xx = linspace(min(xdata), max(xdata), cluster);
yy = interp1(xdata, ydata, xx);
yspline = spline(xx, yy, xdata);
```

Partiendo de la matriz de pixeles obtenida aplicando la función *esqueleto* previamente descrita, tomaremos las coordenadas de cada uno de los pixeles y definiremos los puntos que usaremos como base para determinar los segmentos del spline; esto se realiza mediante el comando *linspace* que determina una relación lineal de espaciado entre elementos evaluados, donde $\min(xdata)$, $\max(xdata)$, son el mínimo y máximo valor encontrado en el eje x , mientras que *cluster* es una constante que contiene la cantidad de puntos a particionar y *xx* es un vector que contiene las coordenadas en x de los puntos resultantes del espaciado lineal. Una vez obtenidas las coordenadas en x de los puntos del espaciado es necesario calcular su respectivo punto en y , para realizar esta operación se utiliza el comando *interp1* donde *xdata* y *ydata* son los puntos referentes a las coordenadas en curva esqueletizada de la imagen original, mediante estos parámetros el comando *interp1* calcula el vector de puntos *yy* para cada punto de *xx*.

Finalmente es calculada la curva suavizada mediante Splines Cúbicos, esta se realiza aplicando el comando *spline*, tomando como parámetros de entrada los vectores *xx* y *yy* que contienen los puntos del espaciado que definen las cantidad de segmentos del spline y el

vector $xdata$ compuesto por todas las coordenadas en x de cada uno de los píxeles que conforman la curva esqueletizada; de esta manera la función spline regresa un vector de coordenadas y , de la nueva curva suavizada.



Figura 5.21 Representación de curvaturas. (a) Huevo regular. (b) Huevo deforme.

A continuación tomaremos las muestras (a, c) (Figura 5.21) del conjunto de imágenes (Figura 5.16) graficadas previamente y aplicaremos el fragmento del algoritmo para la obtención de una curva suavizada mediante splines cúbicos. Para calcular las curvas interpoladas y observar el resultado de esta etapa del método, tomaremos 5 puntos para el espaciado lineal y aplicaremos la función *spline*, de esta manera se obtiene una nueva curva uniforme y suavizada que representa la forma geométrica que debiera tener la muestra analizada si no presentara defectos sobre su superficie.

Es importante recalcar que en esta etapa del análisis de la imagen, al aplicar un Spline Cúbico, se está interpolando una nueva curva que intenta adaptarse lo mejor posible a la distribución de puntos de la imagen analizada, como consecuencia la variación de los puntos para el espaciado lineal determina cual semejante pueda ser la curva resultante respecto a la real, generando de esta manera variaciones o diferencias entre ellas.

Las gráficas que se presenta a continuación son un ejemplo del resultado de una interpolación con 5 puntos, evaluadas en la muestra de un huevo sin deformaciones (Figura 5.22) y uno con deformaciones (Figura 5.23), observándose las diferencias que pueden coexistir entre curvas respecto a la irregularidad de la curvatura real en la imagen.

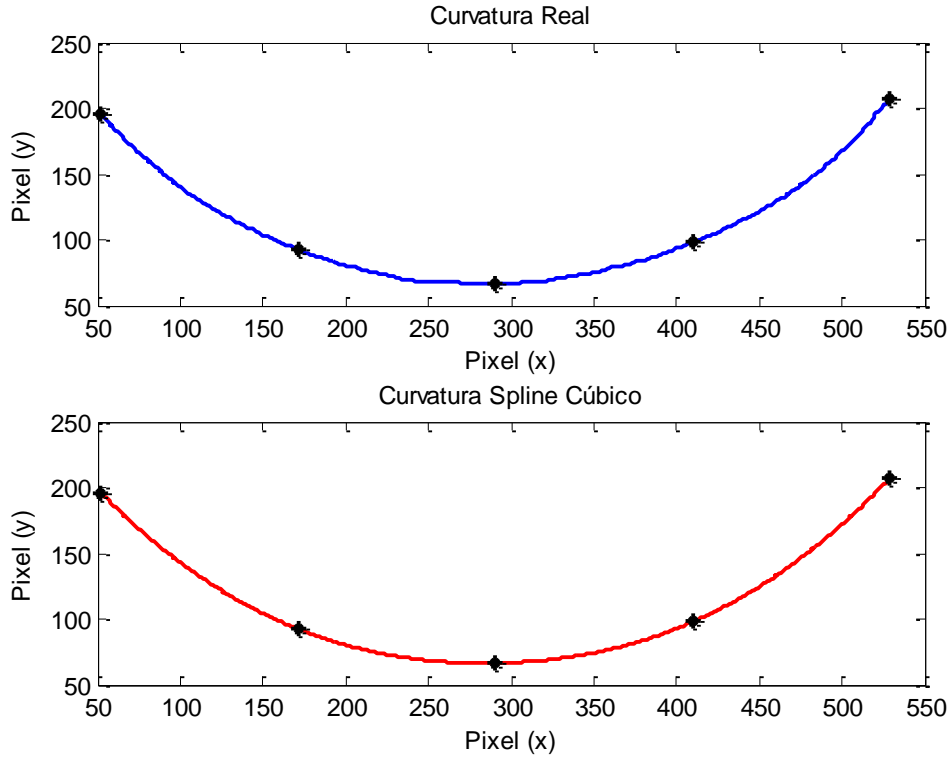


Figura 5.22 Spline Cúbico curva sin deformaciones

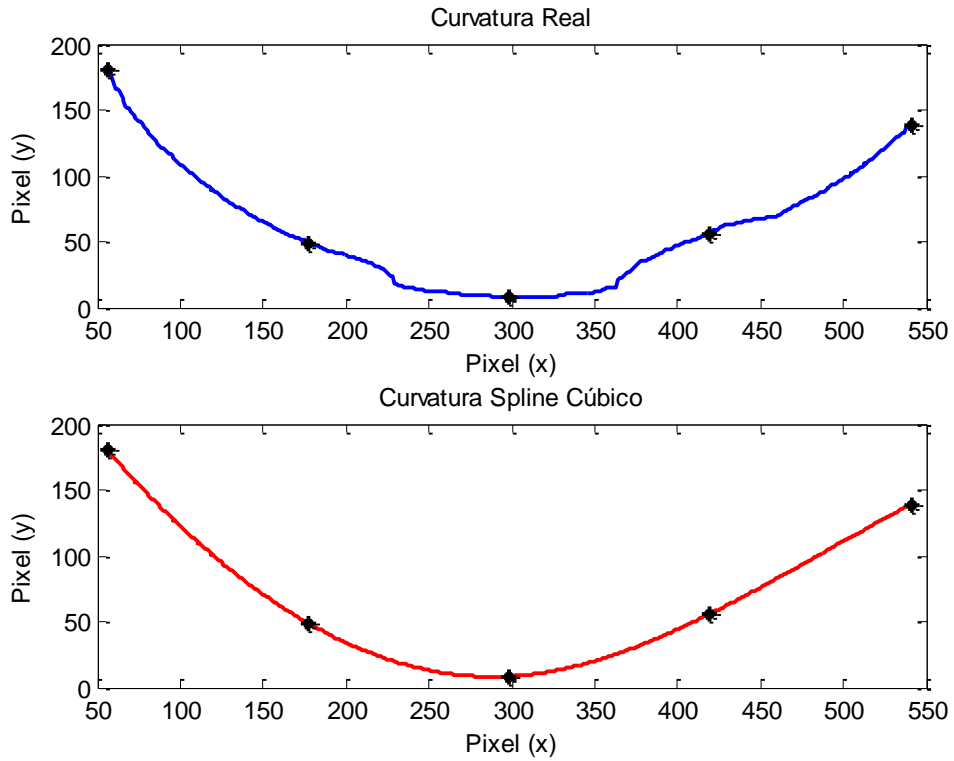


Figura 5.23 Spline Cúbico curva deformada

5.5 Extracción de características

Hasta esta etapa del análisis, hemos obtenido la representación gráfica de una nueva curva suavizada a partir de puntos distribuidos en la curva original, esto nos brinda la capacidad de realizar una comparación en las gráficas resultantes para observar las diferencias que puedan existir debido a las deformaciones en un huevo.

En la Figura 5.24 y Figura 5.25, se observa que al interponer las dos gráficas, se hace notoria la disparidad entre ambas curvas; de esta manera, cuando un huevo no presenta deformaciones, da como resultado una fuerte similitud entre la curvatura real y su spline, debido a flexibilidad de adaptación que caracteriza a los splines cúbicos; por el contrario, si existe alguna deformación en la geometría del huevo, obtendremos diferencias entre las curvas, debido a que el spline no logra adaptarse totalmente como consecuencia de las irregularidades.

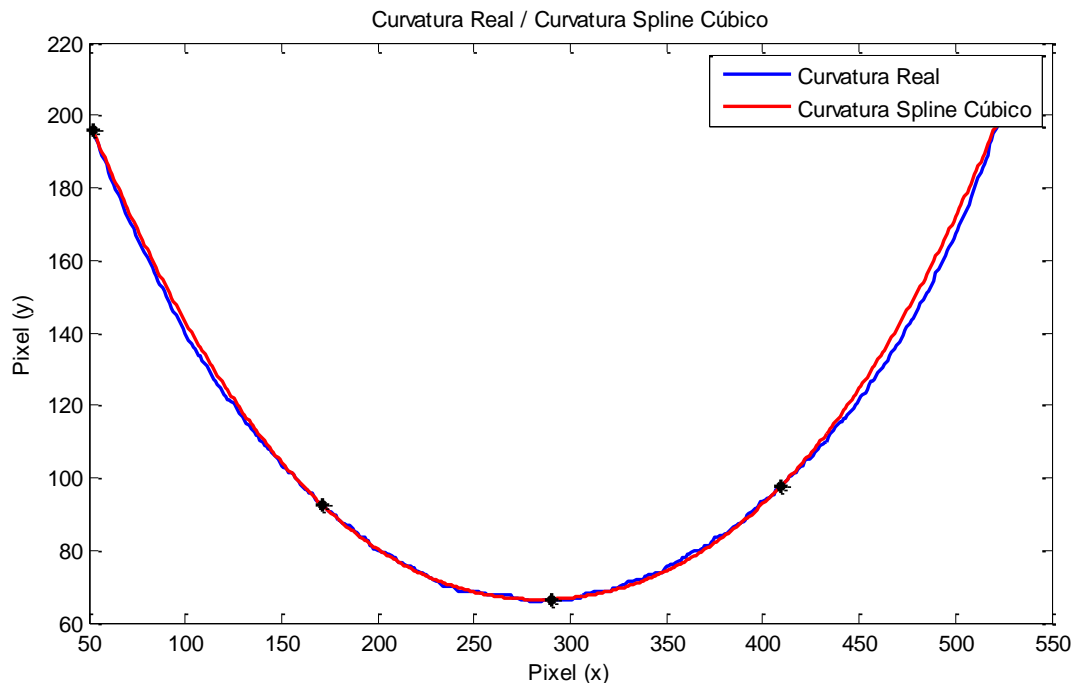


Figura 5.24 Interposición de una curvatura real sin deformaciones y su representación mediante Spline Cúbicos a partir de 5 puntos geométricos muestreados

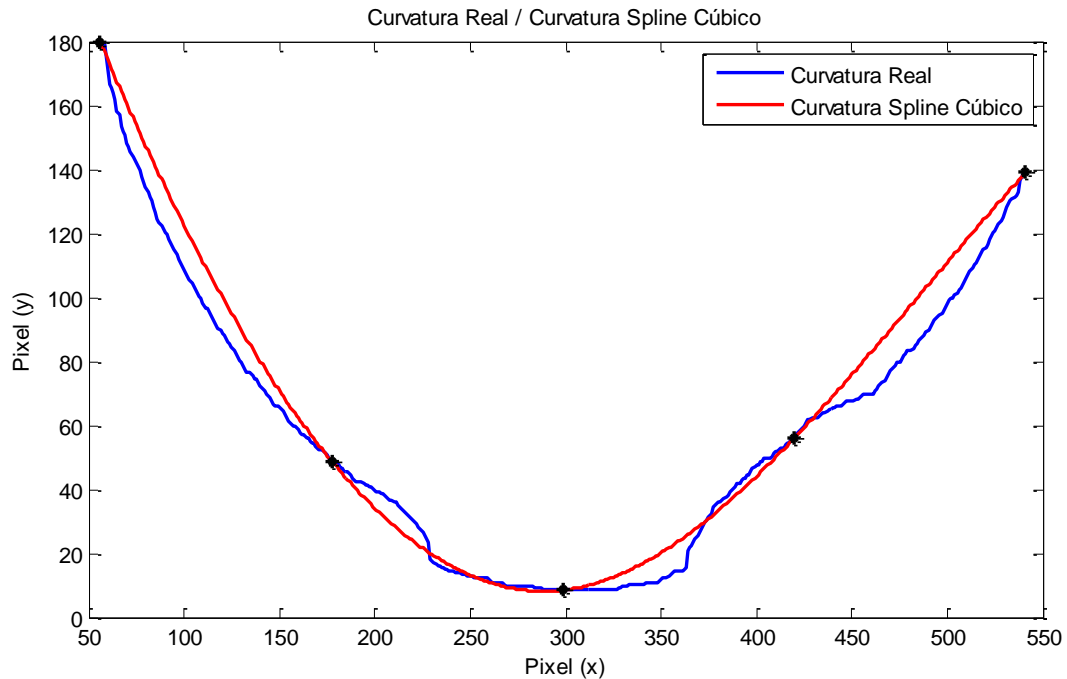


Figura 5.25 Interposición de una curvatura real deformada y su representación mediante Spline Cúbicos a partir de 5 puntos geométricos muestreados

Al efectuar técnicas de procesamiento de imágenes para resolver problemas complejos, es necesario extraer características que represente a la imagen analizada con el fin de tener menos información para su procesamiento y con ello disminuir el tiempo computacional empleado para la operación de los algoritmos.

Al observar las diferencias encontradas al realizar la interpolación, se aprecia que existe una notable variación entre la longitud de la curva real y el spline, debido a las irregularidades de la curvaturas; de igual manera, se hace notoria un área entre las curvas que resalta las zonas donde existen deformaciones. De esta manera se realiza las mediciones en pixeles de los siguientes parámetros de interés:

- Longitud de curva real
- Longitud de curva interpolada
- Área entre curvas

El siguiente conjunto de ecuaciones matemáticas, definen cada uno de los puntos de a evaluar.

5.5.1 Longitud curva real

La longitud lineal en píxeles de la curvatura real está estimada por la sumatoria de las distancias en píxeles entre el conjunto de puntos que definen la curva respecto a su ubicación en el eje de las ordenadas:

$$L1 = \sum_{i=1}^k |(yr^{(i)} - yr^{(i-1)})| \quad (5.1)$$

Donde yr es la matriz de puntos obtenidas de la curvatura real, a partir de una imagen.

Esta longitud de la curvatura real es calculada mediante el siguiente fragmento de código programado en MATLAB®:

```
function l = longitud(M1)
    l = [];
    for i = 1:length(M1)-1
        l = abs(M1(i) - M1(i+1));
    end
    l = sum(l);
end
```

La función *Longitud*, recibe como parámetro de entrada el vector de coordenadas y de la curva real y realiza una iteración para cada uno de los elementos, restando de la posición actual evaluada, el valor absoluto de la coordenada y del valor anterior en el vector. En cada iteración se realiza una sumatoria de los valores calculados para obtener el resultado final de la longitud de píxeles de la línea evaluada.

En la Figura 5.26 se observa la relación de las medidas X_n , estas determinan las diferencia que existe entre cada par de coordenadas y de la gráfica de una curvatura real. De esta manera es posible obtener una magnitud que represente todas las variaciones en la longitud y que es parte importante para la clasificación mediante la Red Neuronal Artificial que proponemos en la siguiente etapa del análisis.

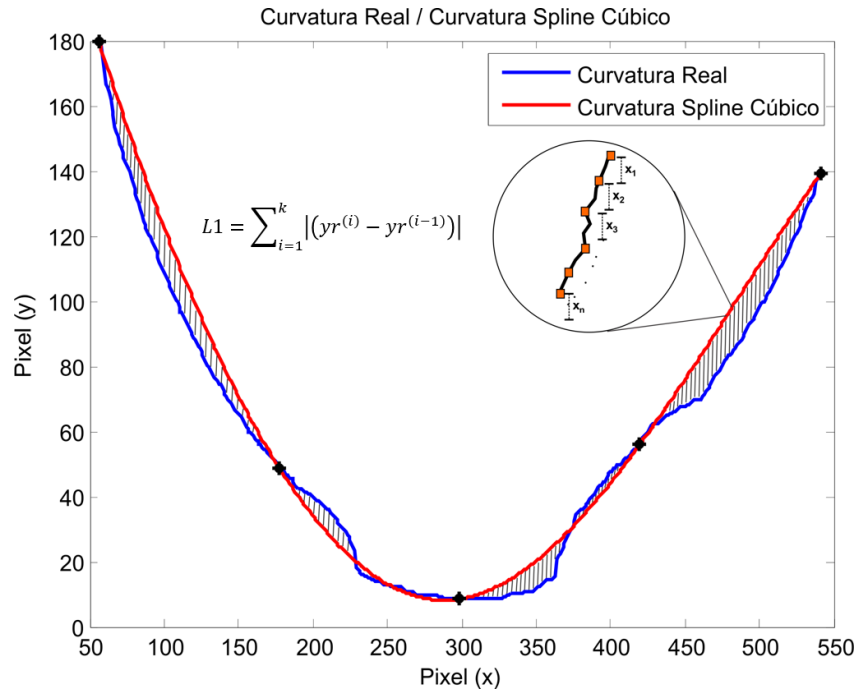


Figura 5.26 Cálculo de longitud en una curva real

5.5.2 Longitud curva spline

La longitud lineal en pixeles de la curvatura interpolada mediante spline cúbicos está determinada por la sumatoria de las distancias en pixeles entre el conjunto de puntos que definen la curva respecto a su ubicación en el eje de las ordenadas:

$$L2 = \sum_{i=1}^k |(y_s^{(i)} - y_s^{(i-1)})| \tag{5.2}$$

Donde y_s es la matriz de puntos obtenidas de la curvatura interpolada, a partir de la curvatura real.

Esta longitud de la curvatura interpolada es calculada mediante el siguiente fragmento de código programado en MATLAB®:

```

function l = longitud(M1)
    l = [];
    for i = 1:length(M1)-1
        l = abs(M1(i) - M1(i+1));
    end
    l = sum(l);
end
    
```

La función *Longitud*, recibe como parámetro de entrada el vector de coordenadas y de la curva interpolada y realiza una iteración para cada uno de los elementos, restando de la posición actual evaluada, el valor absoluto de la coordenada y del valor anterior en el vector. En cada iteración se realiza una sumatoria de los valores calculados para obtener el resultado final de la longitud de pixeles de la línea evaluada.

En la Figura 5.27 se observa la relación de las medidas X_n , esta determina la diferencia que existe entre cada par de coordenadas y de la gráfica de una curvatura interpolada. De esta manera, es posible obtener una magnitud que represente todas las variaciones en la longitud generada al aplicar el Spline Cúbico.

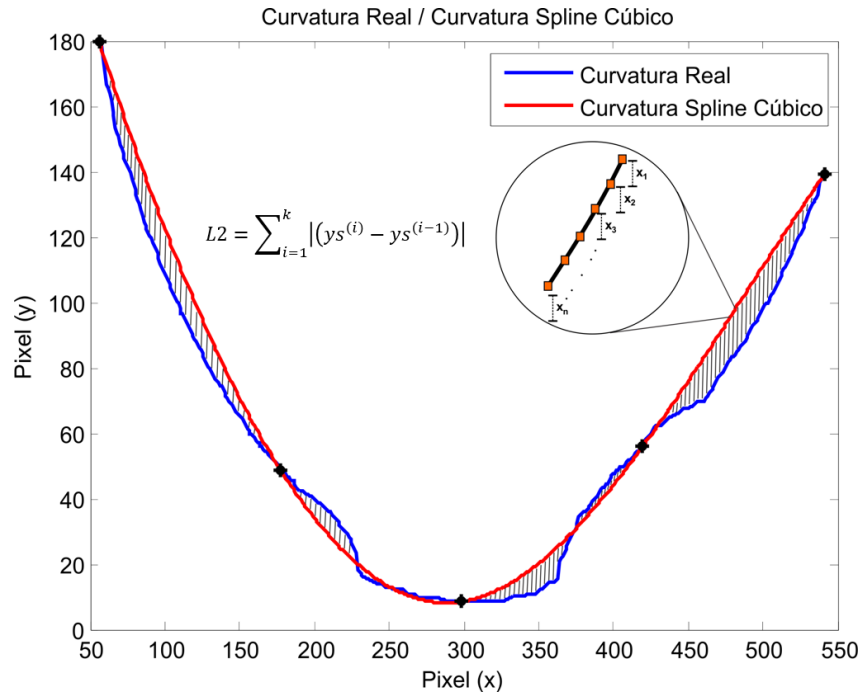


Figura 5.27 Cálculo de longitud en una curva spline

5.5.3 Área entre curvas

El área en píxeles entre las dos curvas al realizar una interposición gráfica de la curvatura real y la interpolada está determinada por la sumatoria de las diferencias entre las distancias del conjunto de puntos que definen ambas curvas respecto a su ubicación en el eje de las ordenadas (Figura 5.28), expresada por la siguiente ecuación:

$$A = \sum_{i=1}^k |(y_s^{(i)} - y_r^{(i)})| \tag{5.3}$$

Donde y_r , y_s son las matrices de puntos obtenidas a partir de una imagen de la curva real e interpolada respectivamente.

El área entre las curvas es calculada utilizando el comando de MATLAB® *sum* para realizar una sumatoria del valor absoluto de las diferencias en la posición de las coordenadas en y de la curva interpolada *ydatab* respecto a la real *ydata*.

```
sum(abs(ydatab - ydata));
```

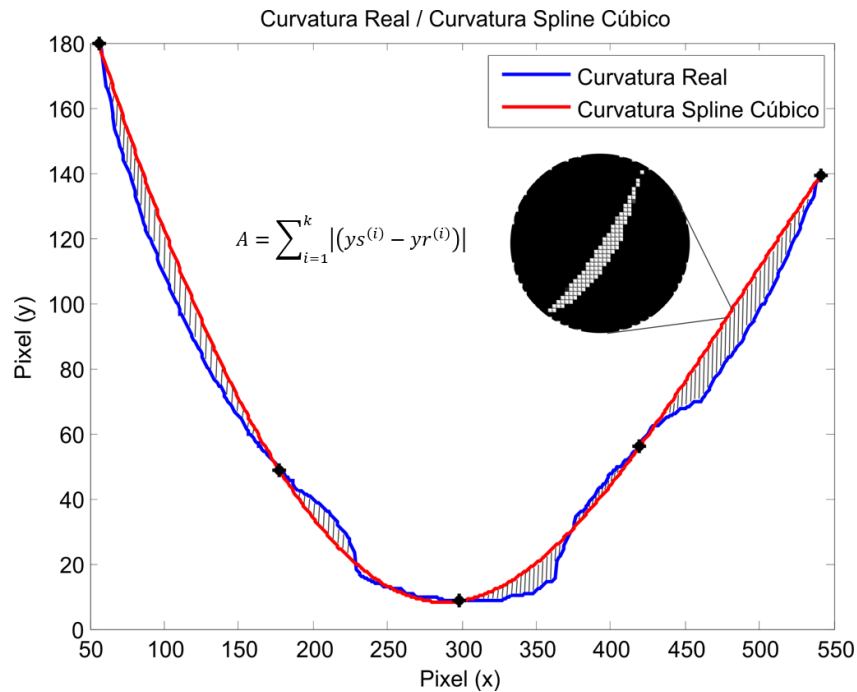


Figura 5.28 Cálculo del área entre curvas

5.6 Clasificación de forma

En esta etapa de la investigación, a partir de las ecuaciones matemáticas que caracterizan las diferencias entre las curvas analizadas, se propone el uso de una Red Neuronal Artificial como clasificador. Para realizar este proceso se utiliza las herramientas de nos proporciona el toolbox de MATLAB® destinado a redes neuronales artificiales. Este paquete de herramientas nos permite definir, entrenar y simular un modelo de red neuronal.

5.6.1 Arquitectura de la Red Neuronal Artificial

Partiendo de la teoría de Redes Neuronales Artificiales, su modelo matemático (Figura 5.29) y las cualidades como clasificadores, se propone el uso de este tipo de herramienta de inteligencia artificial para evaluar la información de los parámetros obtenidos en la etapa previa, esto con la finalidad de interpretar la relación que coexiste entre ellos y generar una clasificación que nos indique la presencia de deformaciones en la muestras de huevos evaluadas.

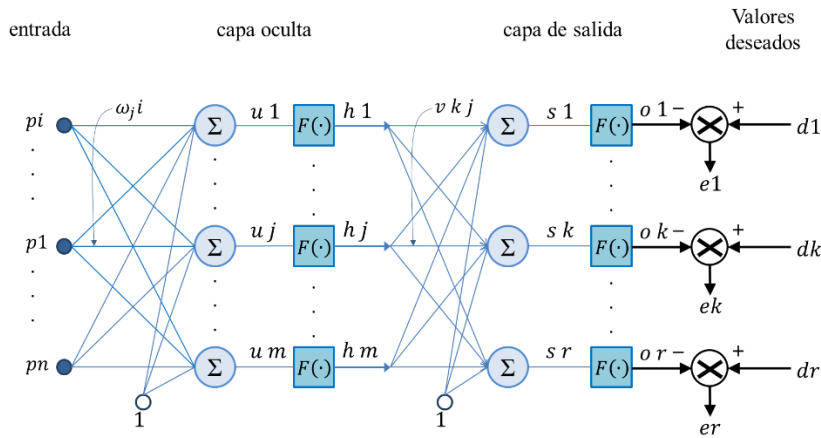


Figura 5.29 Modelo matemático de una RNA

Se propone la Red Neuronal Artificial la Figura 5.30, del tipo Levenberg-Marquardt backpropagation, para realizar el proceso de clasificación de las irregularidades geometrías en los huevos; compuesta por 3 neuronas en la capa de entrada donde se ingresan los resultados obtenidos de las ecuaciones (5.1), (5.2), (5.3), activadas por una función del tipo

logsig; 2 neuronas en la capa oculta activada por una función *logsig* y 1 neurona en la capa de salida con una función de activación del tipo *purelin*.

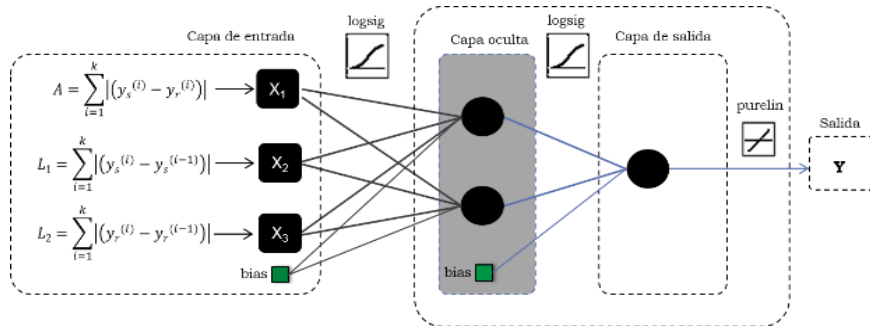


Figura 5.30 Red Neuronal Artificial propuesta

5.6.2 Entrenamiento de la Red Neuronal Artificial

Una RNA al igual que su análogo biológico, requiere de una base de conocimiento que contenga información relevante del caso de estudio, este tipo de entrenamiento se le denomina supervisado.

El aprendizaje supervisado es una técnica para deducir una función a partir de datos de entrenamiento. Los datos de entrenamiento consisten de pares de objetos (normalmente vectores): una componente del par son los datos de entrada y el otro, los resultados deseados. La salida de la función puede ser un valor numérico (como en los problemas de regresión) o una etiqueta de clase (como en los de clasificación). El objetivo del aprendizaje supervisado es el de crear una función capaz de predecir el valor correspondiente a cualquier objeto de entrada válida después de haber visto una serie de ejemplos, los datos de entrenamiento. Para ello, tiene que generalizar a partir de los datos presentados a las situaciones no vistas previamente.

Para el entrenamiento de nuestro modelo de RNA, creamos una base de datos compuesta por dos grupos; El primer agrupamiento integrado por 100 imágenes de huevos sin deformaciones como las apreciadas en la Figura 5.31 y El segundo conformado por 100 imágenes de huevos que presentan deformaciones en la superficie del cascaron (Figura 5.32).

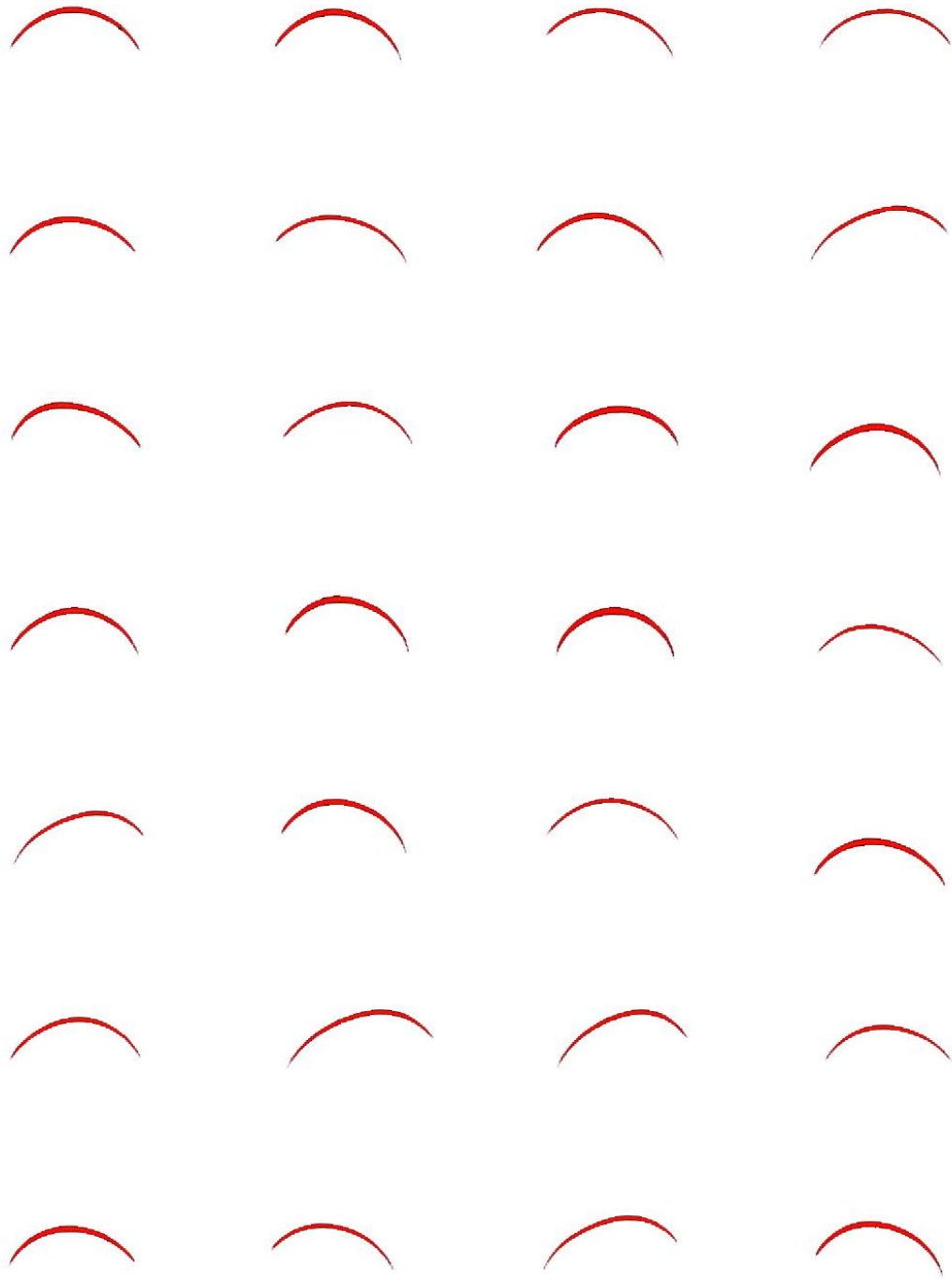


Figura 5.31 Imágenes de entrenamiento de huevos sin deformaciones



Figura 5.32 Imágenes de entrenamiento de huevos deformados

Se desarrolló un algoritmo utilizando MATLAB®, para definir, entrenar, simular y evaluar la Red Neuronal Artificial propuesta.

Tomando como referencia la base de datos compuestas por 200 imágenes, se realizó un entrenamiento de la RNA, como se muestra en la siguiente sección de código.

```
function red = RedNeuronal
    tic;
    clear all;
    clc;
    names = {'HB1.jpg', 'HB2.jpg', 'HB3.jpg', . . . 'HB100.jpg',
            'HM1.jpg', 'HM2.jpg', 'HM3.jpg', . . . 'HM100.jpg'};
    t = [1,1,1, . . . ,1,0,0,0 . . . ,0];
    e = zeros(3,1);
    for k = 1:length(t)
        e(:,k) = deformacion;
    end
end
```

En la función *RedNeuronal* programada, inicialmente se define el vector de imágenes *names* que conforman la base de datos de entrenamiento; posteriormente, se crea un nuevo vector *t* que en este caso, contiene los targets u objetivos que deseamos obtener asociados a la forma de cada uno de los huevos, 1 para huevos sin deformaciones y 0 para huevos deformes. Se crea un vector de *zeros* y se evalúa cada una de las imágenes, registrando los 3 parámetros obtenidos a partir de las ecuaciones (5.1), (5.2), (5.3). Finalmente se configuran cada una de las propiedades de la red neuronal que definen la arquitectura propuesta como veremos a continuación.

Mediante la función *feedforwardnet* se define el método propuesto de entrenamiento *trainlm* (Levenberg-Marquardt), las 3 neuronas de la capa de entrada y las 2 neuronas para la capa oculta.

```
red = feedforwardnet([3,2], 'trainlm');
```

Se configura la red neuronal tomando como referencia los parámetros obtenidos por las ecuaciones extraídas de las curvaturas *e* y el los targets *t* asociados a cada imagen.

```
red = configure(red, e, t);
```

Se configuran algunas propiedades usadas para el entrenamiento de la red como son el máximo número de épocas, error esperado y las funciones de activación para cada capa.

```
red.divideMode = 'none';
red.trainParam.goal = 0.00001; %error
red.trainParam.epochs = 3000;
red.trainParam.show = 100;
red.layers{1}.transferFcn = 'logsig';
red.layers{2}.transferFcn = 'logsig';
red.layers{3}.transferFcn = 'purelin';
```

Finalmente se entrena la red mediante el comando *train* tomando como referencia, la arquitectura de la red, los resultados obtenidos por la ecuaciones matemáticas (5.1), (5.2), (5.3) y los targets u objetivos esperados.

```
[red,tr] = train(red,e,t);
```

Hasta este momento hemos descrito cada una de las etapas que conforman la metodología empleada para el análisis y clasificación de huevos avícolas. En el próximo capítulo veremos los resultados obtenidos durante la ejecución de la red neuronal y validaremos los parámetros calculados por las ecuaciones matemáticas propuestas.

Capítulo 6

Resultados del método sobre superficies deformadas

En este capítulo expondremos los resultados obtenidos en el desarrollo de la investigación.

Primeramente analizaremos las diferencias encontradas al variar el espaciado lineal empleado para definir los segmentos interpolados y cómo influye esto en la adaptación de las curvas creadas por el Spline cúbico.

Posteriormente examinaremos la relación que existe entre el espaciado lineal utilizado en la interpolación con respecto al desempeño de la Red Neuronal Artificial, encontrando de esta manera los puntos necesarios para obtener el mejor desempeño del algoritmo.

Finalmente concluiremos validando si el uso de Spline cúbicos como herramienta para aproximar curvas a partir de puntos distribuidos y Redes Neuronales Artificiales coadyuva a la detección de deformaciones en superficies curvas.

6.1 Relación de adaptación en Spline Cúbico

En este apartado analizaremos la relación que existe en la cantidad de puntos para el espaciado lineal y las curvas generadas por el Spline cúbico tomando como referencia las muestra b y c de la Figura 6.1.



Figura 6.1 Muestras para la evaluación de la adaptación de Spline Cúbico

El análisis en la relación de ajuste del Spline cúbico, está determinado por la cantidad de puntos utilizados en la interpolación, por tal motivo, evaluaremos los efectos de la adaptación del Spline para 3, 4, 5, 6 y 7 puntos.

En la Figura 6.2 y Figura 6.3 se observa que a partir de 3 puntos tomados para realizar la interpolación por Spline cúbicos se obtiene una baja adaptación en las curvas sin deformaciones, siendo poco factible la evaluación de características. Por otra parte, En la Figura 6.4 y Figura 6.5 se observa una mejor adaptación para la interpolación por 4 puntos, sin embargo, las diferencias encontradas entre una curva sin deformaciones, respecto a una deforme; no son lo suficientemente espaciadas para poder realizar una correcta clasificación.

Al evaluar el spline con 5 puntos (Figura 6.6, Figura 6.7) se aprecia que ya existe una buena adaptación de la curva sin deformaciones y una deformada, respecto a la curva suaviza mediante el Spline; en este punto se hace evidente una diferencia en la adaptación del spline cuando se examina una curva con deformaciones (Figura 6.7).

Evaluando el Spline con 6 y 7 puntos (Figura 6.8, Figura 6.9, Figura 6.10, Figura 6.11) se comienza a notar una mejor adaptación para una curva sin deformaciones, sin embargo, el spline de igual manera tiene una mejor adaptación en las curvaturas con deformaciones, dando como resultado una estrecha diferencia, ocultando algunas irregularidades en la forma.

6.1.1 Spline Cúbico 3 puntos

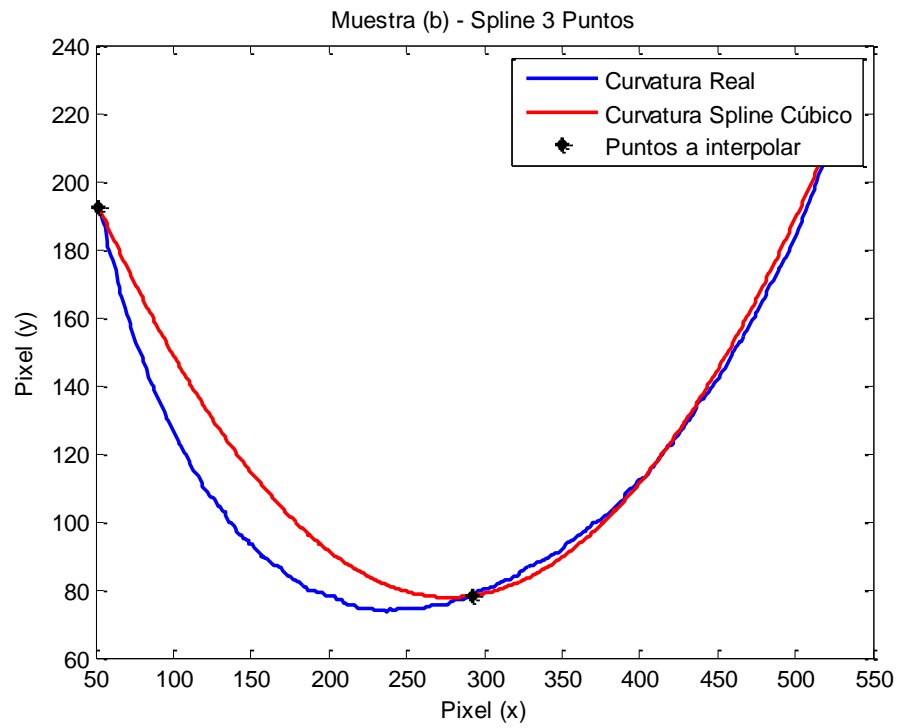


Figura 6.2 Spline cúbico de 3 Puntos obtenido de la muestra (b)

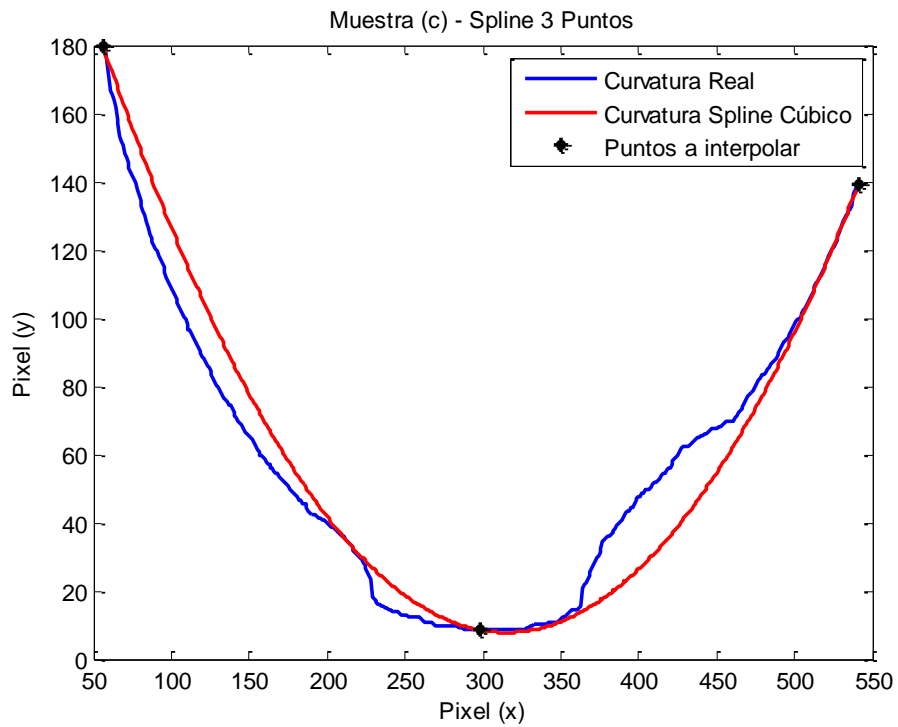


Figura 6.3 Spline cúbico de 3 Puntos obtenido de la muestra (c)

6.1.2 Spline Cúbico 4 puntos

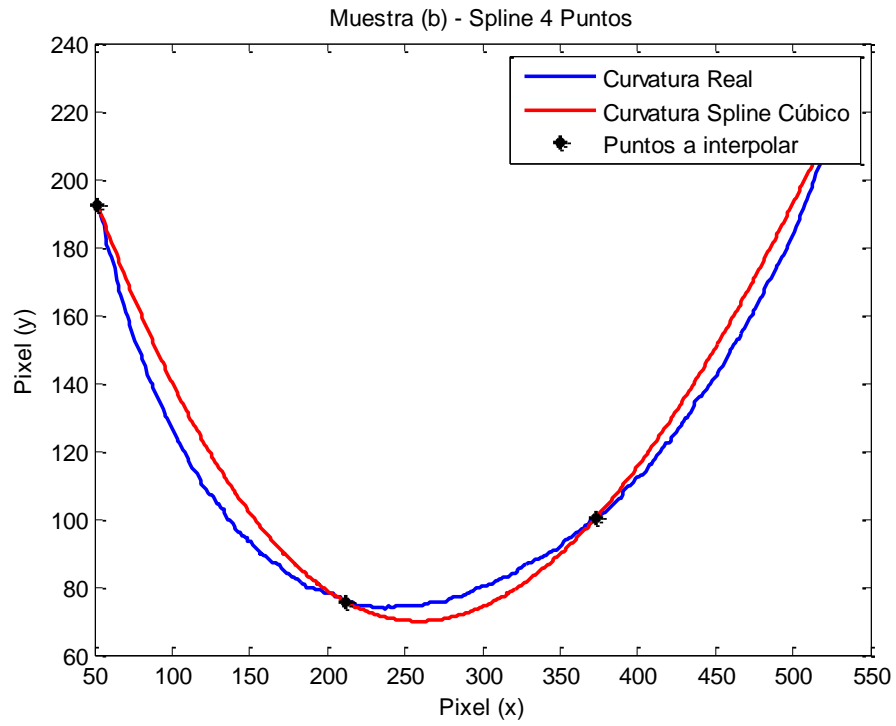


Figura 6.4 Spline cúbico de 4 Puntos obtenido de la muestra (b)

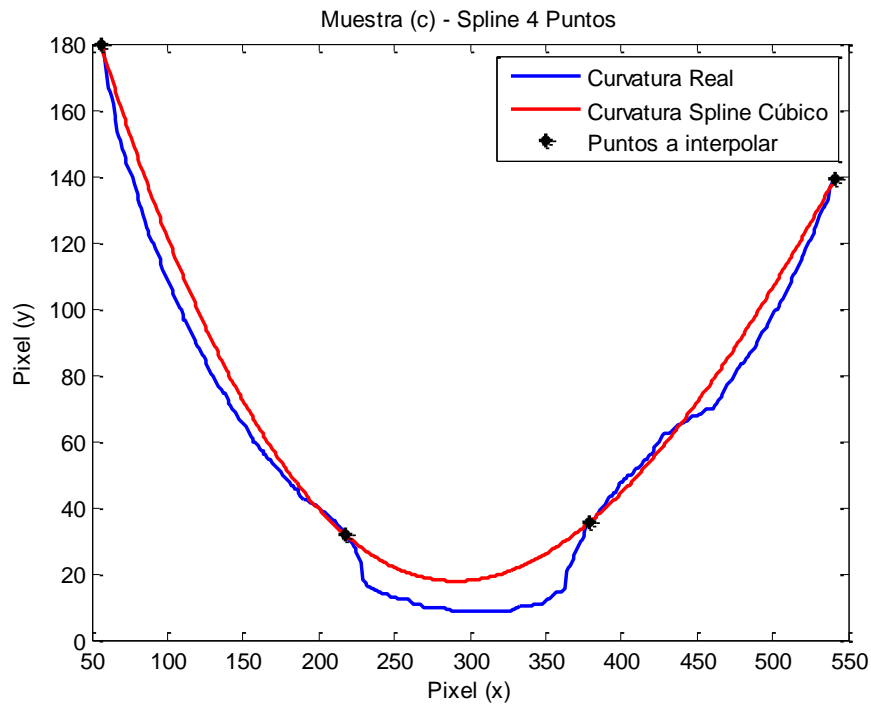


Figura 6.5 Spline cúbico de 4 Puntos obtenido de la muestra (c)

6.1.3 Spline Cúbico 5 puntos

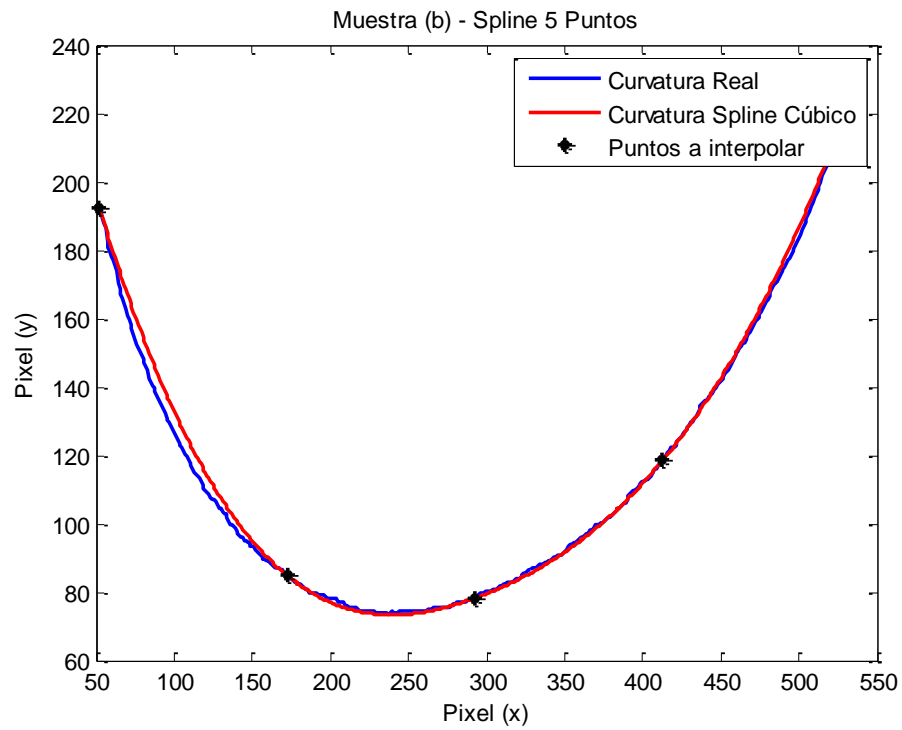


Figura 6.6 Spline cúbico de 5 Puntos obtenido de la muestra (b)

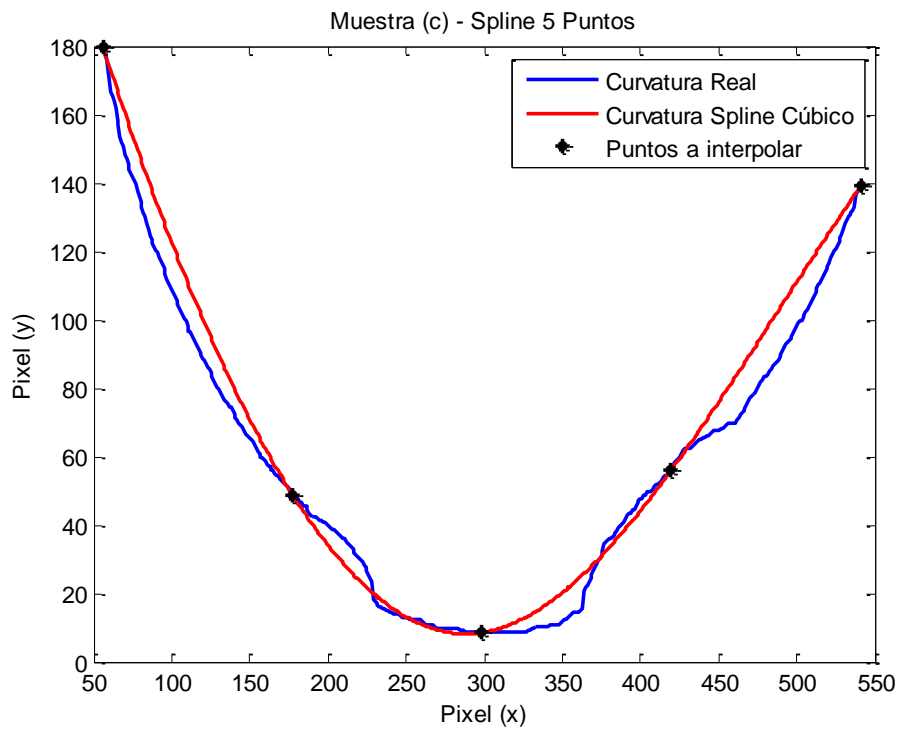


Figura 6.7 Spline cúbico de 5 Puntos obtenido de la muestra (c)

6.1.4 Spline Cúbico 6 puntos

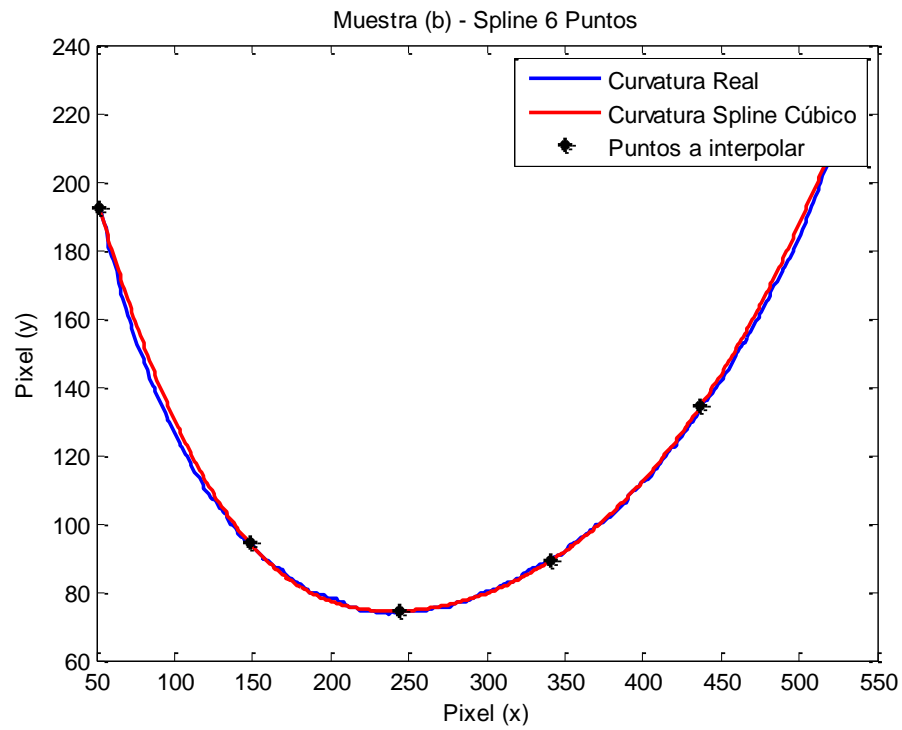


Figura 6.8 Spline cúbico de 6 Puntos obtenido de la muestra (b)

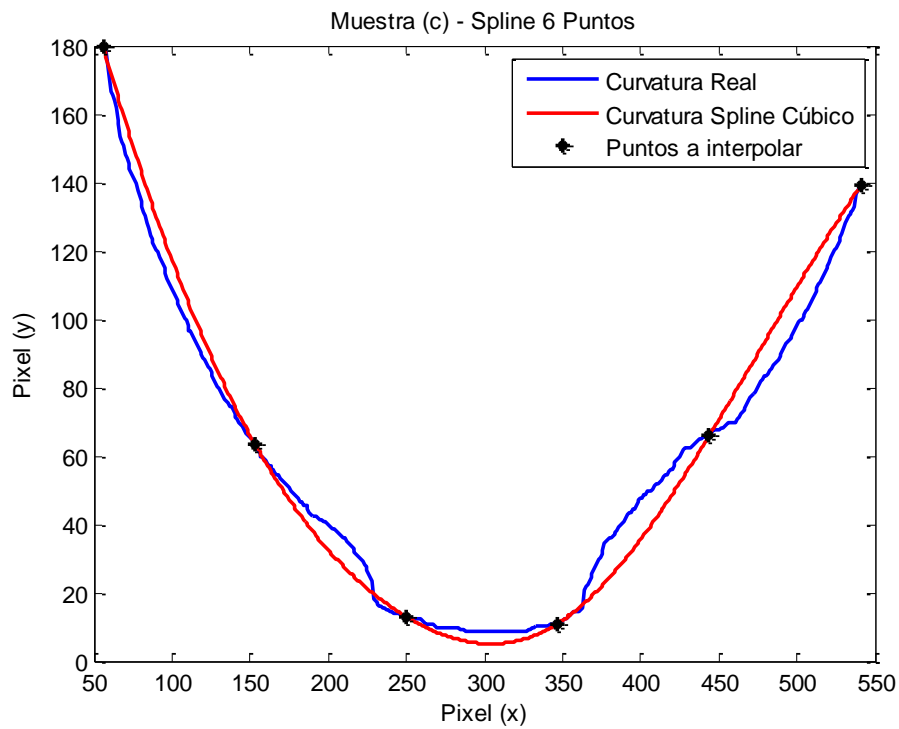


Figura 6.9 Spline cúbico de 6 Puntos obtenido de la muestra (c)

6.1.5 Spline Cúbico 7 puntos

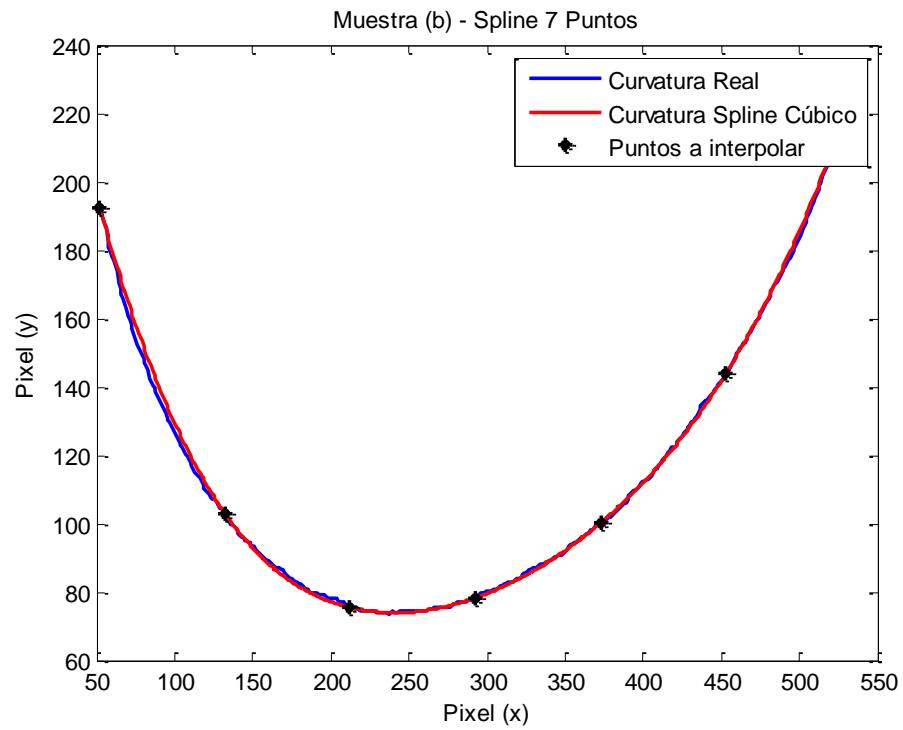


Figura 6.10 Spline cúbico de 7 Puntos obtenido de la muestra (b)

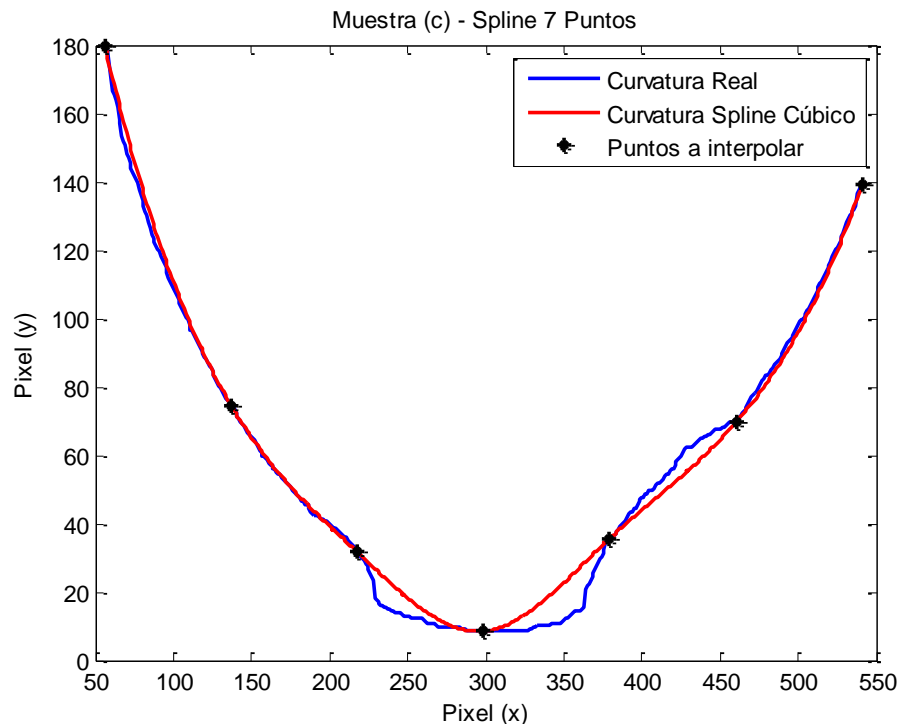


Figura 6.11 Spline cúbico de 7 Puntos obtenido de la muestra (c)

6.2 Evaluación de la Red Neuronal Artificial

En base al análisis realizado a las curvas, para identificar un rango aceptable en la definición del espaciado lineal utilizado al realizar una interpolación por Spline Cúbicos, se determinó que los puntos viables para la evaluación del algoritmo de la RNA está comprendido por 4, 5 y 6 puntos. A partir de esto, se efectuaron 3 simulaciones del algoritmo de la red a diferentes puntos tomados para el Spline Cúbico, realizando en cada prueba una variación de las imágenes de entrenamiento, generando los siguientes resultados.

6.2.1 Resultado RNA con Spline Cúbico de 4 puntos

En esta primera evaluación de la red observamos que existe una relación entre la cantidad de imágenes de entrenamiento, respecto la eficiencia de la red, teniendo la mayor tendencia de errores por debajo de las 100 imágenes, incrementando la tasa de error; esto debido a la poca adaptación del spline en situaciones donde las curvas presenta una inclinación debido a una mala posición del huevo al ser evaluado en el filler.

Tabla 6.1 Resultados Spline Cúbico 4 Puntos

| SPLINE CÚBICO 4 PUNTOS | | | | | | | | |
|------------------------|---------------------------|--------------------|----------|---------|--------|----------|-------------------------|--------------------------|
| Prueba | Imágenes de entrenamiento | Imágenes de prueba | Aciertos | Errores | Épocas | ECM | Tiempo de procesamiento | Porcentaje de eficiencia |
| 1 | 50 | 200 | 183 | 17 | 39 | 8.24E-06 | 250 ms | 91.50% |
| 2 | 100 | 200 | 187 | 13 | 80 | 8.44E-06 | 250 ms | 93.50% |
| 3 | 150 | 200 | 194 | 6 | 134 | 7.11E-06 | 250 ms | 97.00% |

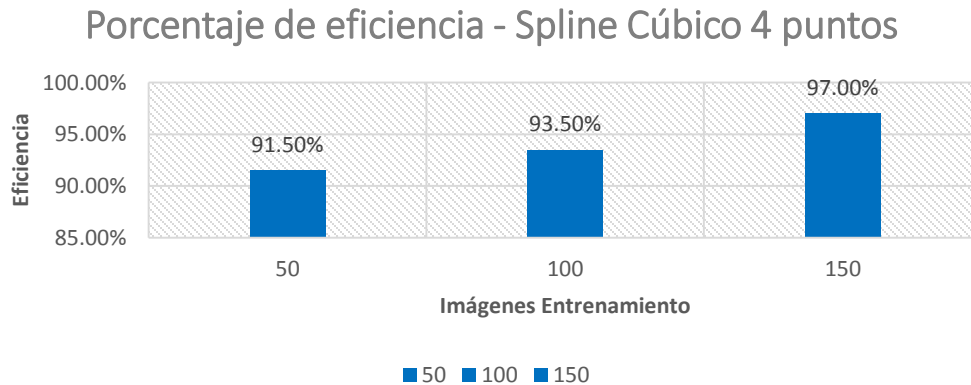


Figura 6.12 Eficiencia Spline Cúbico 4 Puntos

6.2.2 Resultado RNA con Spline Cúbico de 5 puntos

En la segunda evaluación de la red, observamos que se mantiene una relación entre la cantidad de imágenes de entrenamiento, respecto la eficiencia de la red, teniendo la menor tendencia de errores por arriba de las 100 imágenes, disminuyendo la tasa de error; esto debido a una mejor adaptación del spline en situaciones donde las curvas presenta una inclinación, ocasionado por una mala posición del huevo al ser evaluado en el filler y de igual manera aún se mantiene notoria las variaciones en la gráfica de la curva real y la interpolada para los casos donde se presentan deformaciones. Otro punto importante a señalar, es que se conserva un amplio margen de la eficiencia de la prueba 3 respecto a la 1 y 2, debido al bajo ECM¹³ en el entrenamiento de la RNA.

Tabla 6.2 Resultados Spline Cúbico 5 Puntos

| SPLINE CÚBICO 5 PUNTOS | | | | | | | | |
|------------------------|---------------------------|--------------------|----------|---------|--------|----------|-------------------------|--------------------------|
| Prueba | Imágenes de entrenamiento | Imágenes de prueba | Aciertos | Errores | Épocas | ECM | Tiempo de procesamiento | Porcentaje de eficiencia |
| 1 | 50 | 200 | 184 | 16 | 10 | 8.15E-06 | 250 ms | 92.00% |
| 2 | 100 | 200 | 187 | 13 | 33 | 8.60E-06 | 250 ms | 93.50% |
| 3 | 150 | 200 | 195 | 5 | 61 | 2.60E-06 | 250 ms | 97.50% |

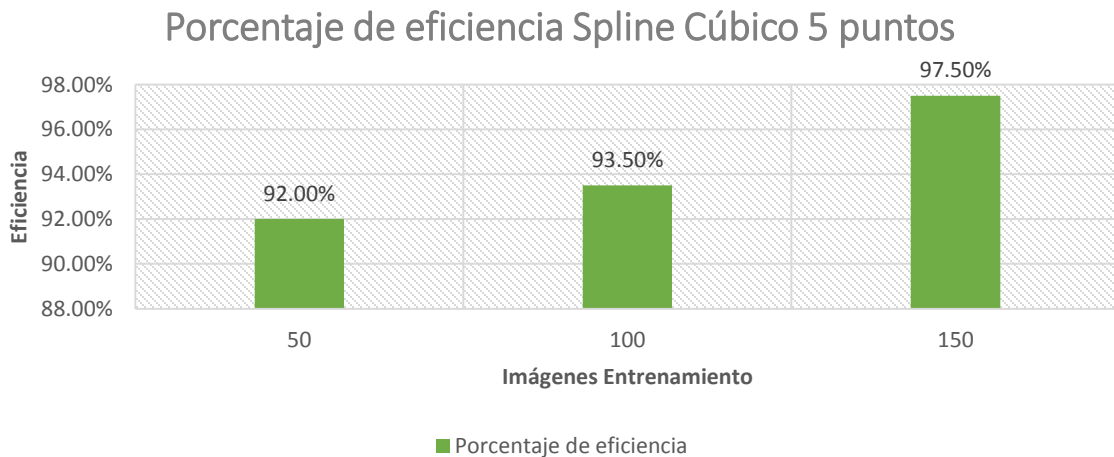


Figura 6.13 Eficiencia Spline Cúbico 5 Puntos

¹³ ECM (en español: Error Cuadrático Medio. en ingles: (MSE) Mean Squared Error)

6.2.3 Resultado RNA con Spline Cúbico de 6 puntos

En esta última prueba, observamos que la relación que existía anteriormente entre las imágenes de entrenamiento y la eficiencia disminuye sustancialmente, haciéndose homogénea, esto es debido a una mejor adaptación del spline en situaciones donde las curvas presenta una inclinación, ocasionado por a una mala posición del huevo al ser evaluado en el filler; sin embargo, al tener más puntos para la evaluación del Spline, en un huevo con deformaciones, la curva interpolada tiene una mejor adaptación, causando que las diferencias extraídas por la ecuaciones matemáticas ya no sean representativas. Por tal motivo ya no se aprecian diferencia entra cada prueba realizada en esta simulación, mantenido una estrecha similitud en la cantidad de aciertos, errores y eficiencias de la red.

Tabla 6.3 Resultados Spline Cúbico 6 Puntos

| SPLINE CÚBICO 6 PUNTOS | | | | | | | | |
|------------------------|---------------------------|--------------------|----------|---------|--------|----------|-------------------------|--------------------------|
| Prueba | Imágenes de entrenamiento | Imágenes de prueba | Aciertos | Errores | Épocas | ECM | Tiempo de procesamiento | Porcentaje de eficiencia |
| 1 | 50 | 200 | 190 | 10 | 19 | 5.15E-06 | 250 ms | 95.00% |
| 2 | 100 | 200 | 191 | 9 | 17 | 7.27E-07 | 250 ms | 95.50% |
| 3 | 150 | 200 | 193 | 7 | 347 | 8.62E-06 | 250 ms | 96.50% |

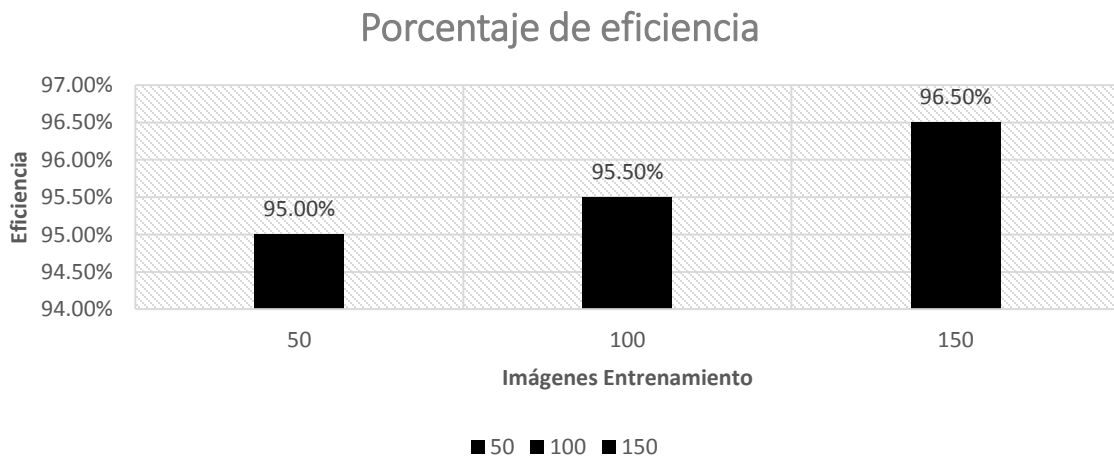


Figura 6.14 Eficiencia Spline Cúbico 6 Puntos

6.3 Análisis de Resultados

El resultado de las pruebas efectuadas, contribuyó a localizar la mayor eficiencia del algoritmo, al determinar que el mejor espaciado lineal para el Spline Cúbico se mantiene en las 5 unidades; conservando una diferencia entre el ajuste de una curva uniforme respecto a una deformada. De esta manera se obtuvo el punto óptimo de operación, dando como resultado una eficiencia del 97.5% y un ECM de 2.60E-06 (Figura 6.15), durante la evaluación de 200 huevos, a partir de 150 imágenes de entrenamiento.

La Figura 6.16, muestra de manera gráfica la eficiencia final respecto a la cantidad de imágenes de entrenamiento y la relación que mantiene con la cantidad de puntos empleados en la interpolación por Spline Cúbicos.

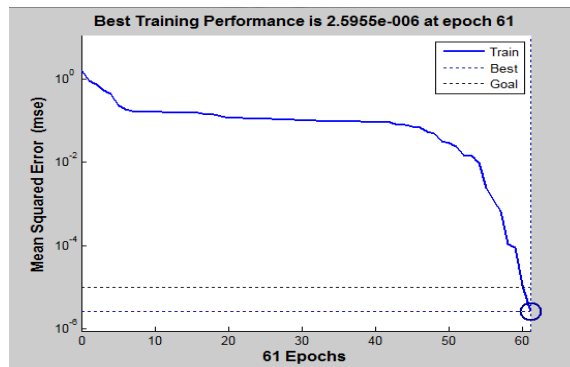


Figura 6.15 Eficiencia óptima en la evaluación de 200 Huevos, representando el Error Cuadrático Medio (MSE) / Épocas

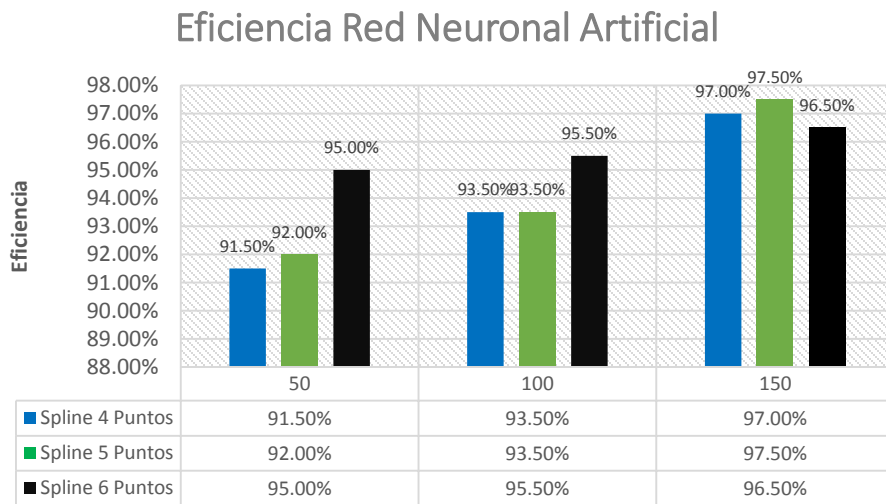


Figura 6.16 Gráfica de eficiencia de la red neuronal artificial

6.4 Discusión de resultados

Partiendo de las aportaciones realizadas en las investigaciones documentadas en el estado del arte y la evaluación de métodos para identificar o clasificar aspectos diversos en la calidad de productos; realizaremos un análisis de los métodos, eficiencia y técnicas utilizadas.

Josep Tornero (2012), en su investigación intitulada “*Detección de defectos en carrocerías de vehículos Basado en Visión Artificial*”, obtuvo una eficiencia del 100% en la evaluación de las deformaciones; registrando el grosor de las líneas capturadas por el reflejo de lámparas fluorescentes sobre las carrocerías de los automóviles; detectando alteraciones de 0.2, 0.3 y 0.6 mm. Esta alta eficiencia se debió a una previa calibración de las cámaras del sistema de visión empleado, resultando una resolución de 0.896 mm/px; sin embargo, presenta el inconveniente de depender de una estricta calibración, una homogeneidad en la iluminación y que el elemento muestreado permanezca estático (Tornero & Armesto, 2012).

Por otra parte, en la investigación “*In-line sorting of irregular potatoes by using automated computer-based machine vision system*”, desarrollada por Gamal ElMasry en 2012, se propone la clasificación de papas en base a su forma y tamaño, empleando una RNA; obteniendo una eficiencia del 96.5% en la correcta clasificación, partiendo de un grupo de aprendizaje conformado por 228 papas. Este resultado favorable, se determinó al capturar una imagen de cada papa, aplicando una segmentación y obteniendo su contorno; calculando de esta manera la longitud del eje mayor, eje menor, centroide y momento de inercia. En este método, la cantidad de parámetros a evaluar aumenta la sobre carga computacional, al realizar cálculos sobre la matriz de la imagen en la obtención de cada una de las magnitudes; otro punto importante radica en el análisis únicamente del contorno de la papa, al efectuar la segmentación de una imagen en 2D, omitiendo deformaciones que podrían encontrarse en zonas no expuesta; finalmente, un inconveniente más se sitúa en la necesidad de obtener el momento de inercia, como parámetro de interés para conocer la posible inclinación de la papa y se tenga la suficiente información en la detección de los defectos (ElMasry & Cubero, 2012).

Nelson Luque (2011), en su investigación denominada, “*cabecal óptico para determinar anomalías en pared interna de tubos o ductos*”, detecta irregularidades en las paredes internas de tubos, al aplicar un patrón de luz estructurada laser en forma de anillo en su interior; basándose en un punto de referencia ubicado en el centro del patrón laser, obtiene la magnitud del radio respecto al contorno del anillo, a partir del procesamiento de una imagen capturada por un sistema de visión; al efectuar este análisis obtuvo como resultado la identificación de ovalidad, fisuras y corrosión. La ejecución de este método, presenta como inconveniente la necesidad de depender de un punto de referencia fijo, correctamente centrado, de esta manera, al realizar la evaluación de la magnitud del radio no se localizarían variaciones en ductos regulares (Luque, Ruiz, & Hogert, 2011).

Frank Storbeck, en 2001, realizó una investigación nombrada, “*Fish species recognition using computer vision and a neural network*”, donde clasificó especies de peces, mediante la medición del ancho y largo de una línea laser proyectada a lo largo del cuerpo de cada pez; aplicando redes neuronales artificiales, obteniendo una eficiencia en la correcta clasificación del 95%. El método empleado por Storbeck, está basado en la identificación de patrones obtenidos en la evaluación de ventanas de diferente tamaños en una imagen, donde es registrado el reflejo de un patrón laser en forma de línea al entrar en contacto sobre el cuerpo del pez; de esta manera entrenó una red neuronal, identificando los patrones de la concentración de pixeles en la imagen en relación con el ancho y largo de la línea. El método propuesto generó un resultado aceptable; sin embargo, para realizar una correcta clasificación requiere evaluar pequeñas ventanas de la imagen identificando una concordancia con algún patrón previamente aprendido por la red; debido a esto, una variación en la ubicación del pez en la imagen, podría causar que los patrones obtenidos no presenten una correcta relación con la sección analizada, generando una incorrecta clasificación (Storbeck & Daan, 2001).

Finalmente el desarrollo de nuestra investigación que hemos denominado “*Detección de deformaciones en superficies curvas, aplicado en el control de calidad de huevos avícolas*”. Generó una eficiencia del 97.5%; resultado de la evaluación de 200 muestras en una red neuronal compuesta por 150 imágenes de entrenamiento. El método propuesto se basa en la estimación de una curva suavizada a partir de una curva muestreada, mediante interpolación polinómica segmentaria por Spline cúbicos; esto da como resultado la obtención de una

nueva representación “ideal”, de como debiera ser la curvatura de un huevo si no presentara irregularidades en su forma, posteriormente se realiza una comparación gráfica entre ambas curvas y se obtiene la longitud de cada una de las curvas y el área entre ellas, para su evaluación mediante una red neuronal artificial.

La aportación de nuestra investigación, respecto a las analizadas previamente, se centra en la identificación de las deformaciones en superficies curvas, estimando una nueva curvatura al aplicar Spline cúbicos; esto confiere una interesante diferencia respecto a los otros métodos, al no depender de estrictas calibraciones como las realizadas por *Tornero (2012)*; por otro lado, se omite la necesidad de requerir un punto de referencia exacto para evaluar magnitudes de los radios como lo propuesto por *Luque (2011)*.

Otro punto importante es el hecho, de que la mediciones realizadas a las curvas interpoladas por el spline, parten de la misma curva extraída de la imagen, lo que facilita la evaluación de las muestras sin requerir el cálculo del momento de inercia como el empleado por *ElMasry (2012)*, un punto a considerar está relacionado a la evaluación de secciones de las curvas de un huevo para encontrar variaciones en toda la superficie, en contra parte a lo realizado por *ElMasry (2012)* que únicamente evalúa el contorno de una imagen en 2D, omitiendo variaciones importantes ubicadas en otras secciones de la muestra. La extracción de características como las longitudes de las curvas y el área entre ellas, facilita el análisis computacional al emplear vectores o matrices más pequeñas en la evaluación de la red neuronal como las ventanas empleadas por *Storbeck (2001)*.

En la siguiente tabla comparativa se muestran los métodos y resultados obtenidos en cada una de las investigaciones analizadas.

Tabla 6.4 Comparación de Investigaciones

| Industria | Estudio | Método utilizado | Eficiencia | Autor |
|--------------|-------------------------------------|--|------------|---------------------------|
| Piscicultura | Clasificación de Peces | Análisis de ventanas en Imagen | 95% | Storbeck F. et al. (2001) |
| Metalúrgica | Defectos en Ductos | Evaluación de uniformidad radial. | | Luque N. et al. (2011) |
| Agrícola | Defectos en Papas | Evaluación de la firma del contorno | 96.5% | ElMasry G. et al. (2012) |
| Avícola | Deformaciones en Huevos | Análisis de forma aplicando Spline Cúbicos | 97.5% | Torres J. et al. (2013) |
| Automotriz | Defectos en Carrocerías Automóviles | Análisis de píxeles en proyección estructurada | 100% | Tornero J. et al. (2012) |

Conclusiones y Recomendaciones

Los resultados obtenidos indican que una combinación de visión por computadora, spline cúbicos y una red neuronal artificial del tipo Levenberg-Marquardt backpropagation como clasificador, hace posible detectar deformaciones en superficies curvas como las encontradas en huevos avícolas, al realizar una estimación de la forma “ideal” de una curvatura deforme a partir de la reconstrucción de una nueva curva suavizada.

La mayor eficiencia en la detección fue de 97.5% con un ECM de 2.60E-06 y un tiempo de procesamiento de 250 ms, obtenido mediante 200 imágenes de prueba y 150 imágenes para el entrenamiento, utilizando una red neuronal artificial del tipo Levenberg-Marquardt backpropagation.

Por otro lado, se concluyó que existe una fuerte dependencia en la eficiencia del algoritmo respecto a la cantidad de puntos empleados para la interpolación, una disminución en la cantidad de puntos genera variaciones de concordancia entre las curvas analizadas a pesar de no presentar variaciones notorias, en contra parte, un mayor aumento de puntos de interpolación ocasiona que los polinomios cúbicos formados realicen una mejor adaptación a la curva real, lo que origina que se omitan importantes deformaciones en muestras con alteraciones en su geometría; determinando de esta manera que el mejor resultado obtenido, se localizó en función a una representación por spline cúbicos de 5 puntos.

Por otra parte, la aplicación de spline cúbicos como técnica para la obtención de modelos representativos de curvas, favorece en la reducción de etapas de procesamiento, disminuyendo la sobrecarga computacional, el tiempo empleado para el análisis y clasificación, haciendo posible evaluar imágenes sin depender de la posición o rotación de la imagen.

De acuerdo a lo anterior, se deriva como recomendación –para trabajos futuros- proyectar un patrón de luz estructurada laser con líneas paralelas, capturando una sola imagen para realizar una segmentación y aislar cada una de las curvas obtenidas; esto puede contribuir en

disminuir el tiempo utilizado para el procesamiento del algoritmo en el muestreo y análisis de un objeto completo.

La utilización de una proyección laser en otras longitudes de onda, resaltarían las curvaturas en casos donde el cascarón de los huevos analizados sean de otras tonalidades o presenten manchas.

Es posible continuar la evaluación de este algoritmo en la inspección de irregularidades en otro tipo de superficies curvas encontradas en algunos alimentos como: limones, tomates, papas, naranjas, chiles, entre otros.

Referencias

- Arivazhagan, S., & Newlin, R. (2013). External and Internal Defect Detection of Egg using Machine Vision. *Journal of Emerging Trends in Computing and Information Sciences*, 4(3).
- Asadi, V., & Raoufat, M. H. (2010). Egg Weight Estimation by Machine Vision and Neural Network Techniques (A case study Fresh Egg). *International Journal of Natural and Engineering Sciences*, 4(2).
- B. Jahne, H. H. (1999). *Computer Vision and Applications Handbook*. USA: Academic.
- Barrera, G. (2003). *Redes Neuronales Artificiales: Como son y como funcionan*. Ciencia Nicolaita.
- Basogain, X. (2008). *Redes Neuronales Artificiales y sus Aplicaciones*. Escuela Superior de Ingeniería de Bilbao, EHU.
- Belongie, S., & Malik, J. (2002). Shape Matching and Object Recognition Using Shape Contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(24).
- Beyer, M., & Hahn, R. (2002). Analysing fruit shape in sweet cherry. *Scientia Horticulturae*, 96.
- BRANCH, J. W., & OLAGUE, G. (2001). La visión por computador. Una aproximación al estado del arte. 16.
- Brosnan, T., & Sun, D.-W. (2003). Improving quality inspection of food products by computer vision: a review. *Journal of Food Engineering*, 61.
- Burden, R. L., Faires, J. D., & Reynolds, A. C. (1991). *Análisis Numérico*. Boston: Prindle, Weber y Schmidt.
- Castleman, K. (1996). *Digital image processing*. New Jersey: Prentice Hall.
- Chapra, S. C., & Canale, R. P. (1988). *Numerical Methods For Engineers*. New York: McGraw Hill.
- Chen, C. H. (2010). *Handbook of pattern recognition and computer vision*. World Scientific.
- Conte, S. D., & de Boor, C. (1980). *Elementary Numerical Analysis: An Algorithmic Approach*. New York: McGraw Hill.
- Coutts, J. A., & Wilson, G. C. (2007). *Optimum Egg Quality: A Practical Approach*. 5M.
- Cuevas, V. E., & Zaldivar, D. (2004). *Visión por computador utilizando MatLAB y el Toolbox de procesamiento digital de imágenes*. México.
- Curtis, G., & Patrick, W. (2003). *Numerical analysis*. Lavoisier.
- Davies, E. R. (1997). *Machine vision*. London: Academic.

- Dayhoff, J. (1990). *Neural network architectures: an introduction*. New York: Van Nostrand Reinhold.
- Dehrouyeh, M. H., & Omid, M. (2010). Grading and Quality Inspection of Defected Eggs Using Machine Vision. *International Journal of Advance Science and Technology*, 16.
- Dongare, A. D., Kharde, R. R., & Kachare, A. D. (2012). Introduction to Artificial Neural Network. *International Journal of Engineering and Innovative Technology (IJEIT)*, 2(1).
- Du, C.-J., & Sun, D.-W. (2004). Recent developments in the applications of image processing techniques for food quality evaluation. *Food Science and Technology*, 15.
- ElMasry, G., & Cubero, S. (2012). In-line sorting of irregular potatoes by using automated computer-based machine vision system. *Journal Food Engineering*, 112.
- ER, K. (2000). *Principles of neural science*. New York: McGraw-Hill.
- Freeman, J., & Skapura, D. (1991). *Neural Networks*. Addison-Wesley.
- Fyfe, C. (2000). *Artificial Neural Networks and Information Theory*. University of Paisley.
- Gershenson, C. (2003). *Artificial Neural Networks for Beginners*.
- Gibson, J., & Berger, T. (1998). *Digital compression for multimedia. Principles and Standards*. Morgan Kaufmann.
- Goldstein, B. (1992). *Sensación y percepción*. Madrid: Debate.
- Guyton, A. (2011). *Tratado de Fisiología Médica*. Interamericana.
- Hagan, T., & Dcmuth, B. (1996). *Neural Network Design*. China Machine.
- Haykin, S. (1994). *Neural Networks*. New York: Nacmillan College.
- Henrici, P. (1972). *Elementos de Análisis Numérico*. México: Trillas.
- Hildebrand, F. B. (1974). *Introduction to Numerical Analysis*. New York: McGraw Hill.
- Ibrahim, R., & Mohd Zin, Z. (2012). Egg's Grade Classification and Dirt Inspection Using Image Processing Techniques. *Proceedings of the World Congress on Engineering*, 2.
- Jackson, K., & Townsend, G. (1991). *Television and video engineer's reference book*. London: Butterworth-Heinemann.
- Jähne, B., & HauBecker, H. (2000). *Computer Vision and Applications: A Guide for Students and Practitioners*. USA: Academic Press.
- Jahns, G., & Nielsen, H. M. (2001). Measuring image analysis attributes and modelling fuzzy consumer aspects for tomato quality grading. *Computers and Electronics in Agriculture*, 31.
- Kataoka, T., & Al-Mallahi, A. (2010). An image processing algorithm for detecting in-line potato tubers without singulation. *Computers and Electronics in Agriculture*, 70.
- Kaufman. (2002). *Physiology of the eye*. USA: Saunders.

- Kincaid, D., & Cheney, W. (1994). *Análisis Numérico: las matemáticas del cálculo científico*. Addison-Wesley Iberoamericana.
- King'ori, A. M. (2012). Egg Quality Defects: Types, Causes and Occurrence: a Review. *Journal of Animal Production Advances*, 2(8).
- Kleffors. (1998). *Artificial Neural Networks*.
- Krogh, A. (2008). What are artificial neural networks? *Computational Biology*, 26(2).
- Kröse, B., & Van der Smagt, P. (1996). *An introduction Neural Networks*.
- Leeson, S., & Summers, J. D. (2009). *Broiler Breeder Production*. Canada: Nottingham University Press.
- Liming, X., & Yanchao, Z. (2010). Automated strawberry grading system based on image processing. *Computers and Electronics in Agriculture*, 71.
- Luque, N., Ruiz, M. F., & Hogert, E. (2011). Cabezal óptico para determinar anomalías en pared interna de tubos o ductos. *Asociación Argentina de Ensayos no Destructivos y Estructurales*.
- M. Sonka, V. H. (1999). *Image Processing, Analysis and Machine Vision*. USA: PWS.
- Maron, M. J., & López, R. J. (1999). *Análisis Numérico: Un enfoque práctico*. México: CECSA.
- Matich, J. (2001). *Redes Neuronales: Conceptos Básicos y Aplicaciones*. Argentina.
- Netravali, A., & Haskell, B. (1995). *Digital Pictures. Representation, compression and standards*. New Jersey: Plenum.
- Newman, T., & Jain, A. (1995). *A survey of automated visual inspection. Computer vision and image understanding*. USA.
- Nieves, A., & Domínguez, F. C. (2002). *Métodos Numéricos: Aplicados a la Ingeniería*. México: CECSA.
- Omid, M., & Soltani, M. (2013). An expert egg grading system based on machine vision and artificial intelligence techniques. *Journal of Food Engineering*, 118.
- Pajares Martinsanz, G., & de la Cruz García, J. (2008). *Visión por computador*. México: Alfaomega.
- Parker, J. R. (2010). *Algorithms for Image Processing and Computer Vision*. John Wiley & Sons.
- Pescatore, T., & Jacob, J. (2011). *Grading Table Eggs*. Lexington, Kentucky: University Of Kentucky.
- R. González, R. W. (1996). *Tratamiento digital de imágenes*. Wilmington: AddisonWesley Iberoamericana.
- Rashidi, M., & Keshavarzpour, F. (2011). Classification of Egg Size and Shape Based on Mass and Outer Dimensions Analysis. *Libyan Agriculture Research Center Journal International*, 2(5).
- Raymond, S. (1994). *Física*. México: McGraw-Hill.

- Rey, M. (2008). *Alteraciones de la cáscara, clara y yema de huevo*. España: Universidad Europea Miguel de Cervantes.
- Robert, H. D. (1991). *Física*. México: Continental.
- Rodríguez, R. (2012). *Procesamiento Y Análisis Digital De Imágenes*. México: Alfaomega.
- Rojas, R. (1996). *Neural Networks: A Systematic Introduction*. Berlin: Springer-Verlag.
- Rokunuzzaman, M., & Jayasuriya, H. (2013). Development of a low cost machine vision system for sorting of tomatoes. *CIGR Journal*, 15(1).
- Ruíz, A. (2012). *Sistemas de percepción y visión por computador*. España: Universidad de Murcia.
- Sawyer, P. (s.f.). *Cámaras digitales*. Madrid: Paraninfo.
- Schalkoff, R. (1997). *Artificial Neural Networks*. Toronto: McGraw-Hill.
- Sears, F. W. (1996). *Física*. México: Addison Wesley Longman.
- Serrano, A. J., Soria, E., & Martín, J. D. (2010). *Redes Neuronales Atificiales*.
- Shan, B. (2010). Fertility Detection of Middle-stage Hatching Egg in Vaccine Production Using Machine Vision. *International Workshop on Education Technology and Computer Science*, 2.
- Sharma, V., Rai, S., & Dev, A. (2012). A Comprehensive Study of Artificial Neural Networks. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(10).
- Sheid, F. (1991). *Métodos numéricos*. México: McGraw Hill.
- Shouche, S. P., & Rastogi, R. (2001). Shape analysis of grains of Indian wheat varieties. *Computers and Electronics in Agriculture*, 33.
- Solari, S. (1997). *Digital video and audio compression*. McGraw-Hil.
- Sonka, M., Hlavac, V., & Boyle, R. (1998). *Image Processing, analysis and machine vision*. Chapman and Hall.
- Storbeck, F., & Daan, B. (2001). Fish species recognition using computer vision and a neural network. *Fisheries Research*, 51.
- Sucar, E. L., & Gómez, G. (2010). *Visión Computacional*. México: INAOE.
- Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer.
- Tornero, J., & Armesto, L. (2012). Detección de Defectos en Carrocerías de Vehículos Basado en Visión Artificial: Diseño e Implantación. *Revista Iberoamericana de Automática e Informática industrial*, 9.
- Umbaugh, S. (1998). *Computer Vision and Image Processing*. Prentice Hall.
- Valbuena, D. A. (2010). *Calidad Huevo*. Colombia: Avicol.

Verri, E. T. (1998). *Introductory Techniques for 3-D Computer Vision*. USA: Prentice Hall.

Wasserman, P. (1989). *Neural Computing: Theory and Practice*. New York: Van Nostrand Reinhold.