

CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS
DEL INSTITUTO POLITÉCNICO NACIONAL
UNIDAD ZACATENCO

DEPARTAMENTO DE COMPUTACIÓN

**Análisis y clasificación de imágenes a color para la
detección de capacitores y resistencias en tarjetas
de montaje superficial**

Tesis que presenta

Jorge Ortiz García

Para obtener el Grado de
Maestro en Ciencias

En la Especialidad de
Ingeniería Eléctrica
Opción Computación

Directores de la Tesis

Dr. Adriano de Luca Pennacchia
Dr. Luis Gerardo de la Fraga

México, D.F.

Junio 2008

Resumen

La presente tesis tiene como finalidad resolver una parte del problema de verificación de calidad en procesos de ensamblaje de componentes electrónicos sobre tarjetas de montaje de superficie realizados en la industria. Consiste en identificar si un capacitor o resistencia hace falta en determinada región de una tarjeta. Se plantea el uso de una imagen a color de la tarjeta, la cual pasa por un proceso previo, para ser adaptada a las necesidades requeridas por procesos posteriores de análisis. El proceso previo consiste en aplicar técnicas de procesamiento digital como la conversión a tonos de gris, suavizado, binarización, entre otras. Como características para el análisis se ocupan los momentos invariantes, que corresponden a las soldaduras o a los extremos de capacitores y resistencias, además del número de elementos presentes en la imagen binarizada. También se utiliza la intersección de histogramas de brillo y contraste obtenidos de la extracción y análisis de texturas en la imagen a color. Para determinar la existencia de un componente, con base en sus características, se usa el clasificador de distancia mínima. La imagen de la tarjeta de montaje de superficie antes de analizarse debe ser catalogada por el tipo de componente que contienen. Para esta clasificación se ocupa una red neuronal de propagación hacia atrás. Esta red recibe como entrada la suma y diferencia de histogramas obtenidos de la imagen en tonos de gris. La implementación en software esta realizada en lenguaje C sobre GNU/Linux.

Abstract

Currently in the industry, where electronic components assembly is achieved on surface mounted boards, subsists the need to check automatically this process quality. This work intends to solve a little part of this problem which consists to identify if a component such as capacitor or resistance is missing in particular sector of the board. This thesis put forward using color image from board and using three features extraction methods: invariant moments, objects count and texture color image analysis; to achieve the identification. An image preprocessing is applied to improve it for analysis process which consists in digital image processing such as grayscale conversion, smoothing filtering, binarization among others. Invariant moment corresponding to welding or component extremes on a binarized image are used furthermore the objects number there. Also a brightness and contrast intersection obtained from the color image texture extraction and analysis from the color image. To the classification basis features the minimum distance classifier is used, it helps to decide if there is a component or not. Before images being analyzed they have been classified by component type, for this a backpropagation neural network is used. The network input is a histogram's sum and difference obtained from a grayscale image. The software implementation is done in C language on a GNU/Linux.

Agradecimientos

Agradezco y dedico esta tesis a mis papas, José Jorge y Evelia, quienes siempre me han amado, guiado y apoyado incondicionalmente en mi vida y me han enseñado a ser humano.

Agradezco y dedico este trabajo a mi hermana Alejandra Evelia, por haberme escuchado, con quien he crecido y compartido momentos importantes de mi vida, de quien admiro su sabiduría y voluntad de hierro.

A mi gran familia y amigos de toda la vida, que siempre me han animado y aconsejado.

A mis muertos, que cuando mis fuerzas y voluntad flaquean, su recuerdo me alienta a continuar.

A todos mis amigos de la maestría, en especial a Marco, Daniel, Saúl, Cuauhtémoc, Juan, Eduardo, Fernando y Carlos, con quienes luche hombro a hombro en todo momento, en esta etapa de nuestras vidas.

A Sonia, Esperanza, Juan Manuel, Juan Ramón, Jorge Machine, Miguel Alvarado, porque me han enseñado cosas importantes en mi vida.

Al Dr. Adriano de Luca Pennacchia, por haberme aceptado como su tesista, haber tenido fe en mí y haber dirigido esta tesis.

Al Dr. Luis Gerardo de la Fraga, quien me guió y coloco en los pasos correctos para realizar este trabajo.

Al Dr. Francisco Rodríguez Henríquez, quien me brindo su confianza y apoyo.

A la Dra. Xiaoou Li, quien me apoyo en la revisión y corrección de esta tesis.

Al CINVESTAV-IPN, por darme la oportunidad de estudiar, a las autoridades, a los investigadores y al personal administrativo, en especial a la Sofí.

Al CONACyT, por el apoyo económico que recibí durante la realización de mis estudios.

A SONY Tijuana, por haber contribuido con material para el desarrollo de este trabajo.

Finalmente, a todos aquellos que no creyeron, ni confiaron en mí, porque eso me ayudo a saber que no importa lo que pase, solo importa lo que yo haga, y que mi único adversario soy yo mismo.

Índice general

Resumen	III
Abstract	V
Agradecimientos	VII
Índice de figuras	XI
Índice de tablas	XV
1. Introducción	1
1.1. Definición del problema	3
1.2. Organización de la tesis	4
2. Antecedentes	7
2.1. Visión artificial	7
2.1.1. Áreas de aplicación	8
2.1.2. Motivaciones para el uso de la visión artificial en la industria	9
2.1.3. Actividades industriales beneficiadas	10
2.1.4. Clasificación de los sistemas para la inspección de calidad	10
2.2. Métodos de inspección de calidad usados en la industria electrónica	11
2.3. Inspección visual	12
2.4. Inspección visual automática	12
2.4.1. Clasificación de la inspección	13
2.4.2. Secuencia de pasos comunes en un sistema de inspección visual automática	14
2.5. Falsos positivos y falsos negativos: criterios de eficiencia	15
2.6. Defectos en las tarjetas de montaje de superficie	16
2.7. Composición y comportamiento de la soldadura	17
2.8. Estado del arte	18
3. Proceso general	23
3.1. Características de las imágenes a analizar	24
3.2. Elementos de las imágenes considerados para el análisis	25
3.3. Proceso de la clasificación de la imagen	25

3.3.1.	Preprocesamiento	25
3.3.2.	Clasificación del tipo de imagen	27
3.4.	Proceso de análisis e identificación del error	27
3.4.1.	Preprocesamiento	28
3.4.2.	Extracción de características	29
3.4.3.	Clasificación del tipo de error	30
4.	Preprocesamiento	31
4.1.	Almacenamiento en memoria de la imagen a color	32
4.2.	Extracción de la región útil de la imagen	33
4.2.1.	Extracción sin máscara	33
4.2.2.	Extracción con máscara	34
4.3.	Conversión de la imagen a color a tonos de gris	36
4.4.	Suavizado	37
4.5.	Binarización	40
4.6.	Quitar ruido con morfologías	41
4.6.1.	Erosión	42
4.6.2.	Dilatación	42
4.6.3.	Abriendo	42
4.7.	Extracción de objetos	43
4.8.	Media de la suma y diferencia de histogramas de imágenes en tonos de gris	45
4.8.1.	Suma y diferencia de histogramas	46
4.8.2.	Media de la suma y diferencia de histogramas	47
4.8.3.	Otras características de los histogramas	47
5.	Clasificación de la imagen	49
5.1.	Las redes neuronales: definición	50
5.2.	Elementos de una red neuronal	51
5.3.	Arquitectura de la red neuronal	53
5.4.	Algoritmo de la red neuronal de propagación hacia atrás con método de entrenamiento on-line	55
6.	Extracción de características	59
6.1.	Momentos invariantes	59
6.1.1.	Definición de momentos	60
6.1.2.	Definición de momentos centrales	61
6.1.3.	Definición de momentos centrales normalizados	62
6.1.4.	Definición de momentos invariantes	63
6.1.5.	Objetos sobrantes, uso de una línea media y cálculo de los momentos invariantes	64
6.2.	Contabilización de objetos	65
6.2.1.	Región factible	66
6.3.	Intersección de histogramas de brillo y contraste de texturas RGB	69

6.3.1.	Histograma de brillo	71
6.3.2.	Histograma de contraste	72
6.3.3.	Intersección de histogramas	73
6.4.	Otros métodos	74
6.4.1.	Distancia euclidiana	74
6.4.2.	Perímetro de objetos	76
7.	Clasificación del tipo de error	79
7.1.	Definición del clasificador de distancia mínima	80
7.2.	Entrenamiento del clasificador de distancia mínima	82
8.	Implementación y resultados	83
8.1.	Implementación de la clasificación de imágenes	84
8.1.1.	Características de la red neuronal implementada	85
8.1.2.	Clases definidas para el entrenamiento de la red neuronal	86
8.1.3.	Resultados y estadísticas de la clasificación de imágenes	89
8.2.	Implementación del analizador para la identificación del error	91
8.2.1.	Características del analizador	92
8.2.2.	Resultados y estadísticas del análisis e identificación	94
8.3.	Comparación de características y resultados con otro método existente	100
9.	Conclusiones y trabajo futuro	103
9.1.	Conclusiones	103
9.2.	Trabajo futuro	104
9.2.1.	Clasificación	104
9.2.2.	Análisis e identificación	105
A.	Imágenes de ejemplos de intersecciones de histogramas de brillo y contraste	107
B.	Entrenamiento: clases definidas y comprobación	119
B.1.	Clases definidas	119
B.2.	Resultado del entrenamiento	124
B.2.1.	Convergencia del error	124
B.2.2.	Resultados de la comprobación del conjunto de entrenamiento	127
C.	Imágenes clasificadas erróneamente	131
	Referencias	145

Índice de figuras

1.1.	Esta es sólo una de las 8000 imágenes del corpus de análisis, las cuales en su mayoría pertenecen a distintos tipos de tarjetas con diferentes tipos de componentes electrónicos.	2
1.2.	Secuencia de procesos para identificar componentes de tipo rectangular.	3
2.1.	Imagen de la mezcla de soldadura.	17
3.1.	La imagen se encuentra dividida en cuatro partes para facilitar su análisis.	24
3.2.	La imagen de la izquierda es de un capacitor, la imagen del centro es una resistencia y la imagen de la derecha muestra soldaduras de la base sin presencia del elemento.	25
3.3.	Pasos realizados para la clasificación de la imagen.	26
3.4.	Pasos realizados para el análisis e identificación la existencia de capacitores y resistencias.	28
4.1.	Esta imagen es el resultado de la extracción sin máscara de la parte inferior izquierda de la imagen de la figura 1.1.	33
4.2.	Extracción de una parte de una imagen.	34
4.3.	Extracción de la parte inferior derecha de la imagen de la figura 1.1. Es realizada con una máscara circular.	34
4.4.	Extracción con máscara circular de una parte de una imagen	35
4.5.	Estas son las imágenes en tonos de gris de las imágenes extraídas mostradas en la figura 4.1 y 4.3	37
4.6.	Máscara de 3×3	38
4.7.	Posiciones validas de una máscara de 5×5 sobre una imagen	39
4.8.	Filtro espacial pasa-bajas de dimensión 3×3	39
4.9.	Estas imágenes son el resultado del suavizado de las imágenes de la figura 4.5	39
4.10.	La imagen de la izquierda no ha sido suavizada, mientras que la imagen de la derecha ya ha pasado por este proceso.	40
4.11.	Imagen binarizada correspondiente al histograma de la figura 4.12 con un umbral $T = 200$	40
4.12.	Histograma de la imagen derecha de la figura 4.9	41
4.13.	Imagen de la figura 4.11 limpiada con la operación abrir.	43

4.14. Imagen binarizada con tres objetos.	44
4.15. Cada una de estas imágenes pertenece a un objeto extraído de la imagen de la figura 4.14.	44
4.16. Partición de la imagen en 10×10 partes.	45
5.1. Perceptrón	53
5.2. Esquema de la red neuronal	54
6.1. En las imágenes de la izquierda y derecha se muestran objetos con momentos invariantes similares soldaduras de capacitores y resistencias. En la imagen central se describen objetos con momentos invariantes pertenecientes a extremos de capacitores y resistencias.	64
6.2. Estas imágenes muestran su correspondiente línea lógica (color gris) y la intersección con los objetos centrales con respecto a la orientación del componente buscado.	65
6.3. En las imágenes de la parte izquierda se muestran la homogeneidad de los extremos de una resistencia en color y en binario. En la parte derecha, se muestra la heterogeneidad de las soldaduras en imágenes a color y binaria.	67
6.4. Regiones factibles de $m \times n$ en imágenes binarias de soldaduras.	68
6.5. Regiones factibles de $m \times n$ en imágenes binarias de capacitores o resistencias.	69
6.6. Texturas de una tarjeta con capacitor y sin capacitor.	69
6.7. Definición de las secciones de extracción de textura de 20×20 píxeles para un capacitor en posición vertical.	70
6.8. Definición de las secciones de extracción de textura de 20×20 para un capacitor en posición horizontal.	70
6.9. Ventana de 3×3 píxeles para el cálculo de los histogramas de brillo y contraste. Con ocho píxeles vecinos	71
6.10. Ejemplo de la intersección de histogramas. De los histogramas de la imagen a y b se utilizan las barras más pequeñas de cada intervalo, dando como resultado el histograma de la imagen c	74
6.11. La distancia euclidiana entre ambas imágenes es similar.	75
6.12. Entre ambas imágenes, la distancia euclidiana es distinta.	75
6.13. Perímetro de una imagen binarizada de un capacitor o una resistencia.	76
6.14. Perímetro de una imagen binarizada perteneciente a una soldadura.	76
7.1. Límite de decisión del clasificador de distancia mínima.	81
8.1. Imágenes no representativas	87
8.2. Esta imagen es única en el conjunto de análisis.	87
8.3. Existen muy pocas imágenes con una textura similar.	88
8.4. Elementos contenidos en el conjunto de pruebas de 443 elementos.	97
8.5. Componentes contenidos en el conjunto de pruebas de 945 elementos.	98
8.6. Soldaduras en el conjunto de pruebas de 945 elementos.	99

A.1. Histogramas de brillo y contraste de los canales RGB de la sección superior de la imagen con capacitor.	109
A.2. Histogramas de brillo y contraste de los canales RGB de la sección central de la imagen con capacitor.	110
A.3. Histogramas de brillo y contraste de los canales RGB de la sección inferior de la imagen con capacitor.	111
A.4. Histogramas de brillo y contraste de los canales RGB de la sección superior de la imagen sin capacitor.	112
A.5. Histogramas de brillo y contraste de los canales RGB de la sección central de la imagen sin capacitor.	113
A.6. Histogramas de brillo y contraste de los canales RGB de la sección inferior de la imagen con capacitor.	114
A.7. Histogramas de brillo y contraste de los canales RGB resultantes de la intersección de los histogramas de la sección superior con la sección central de la imagen con capacitor.	115
A.8. Histogramas de brillo y contraste de los canales RGB resultantes de la intersección de los histogramas de la sección inferior y sección central de la imagen con capacitor.	116
A.9. Histogramas de brillo y contraste de los canales RGB resultantes de la intersección de los histogramas de la sección central y la sección superior de la imagen sin capacitor.	117
A.10. Histogramas de brillo y contraste del RGB resultado de la intersección de los histogramas de la sección central con la sección inferior de la imagen sin capacitor.	118

Índice de cuadros

8.1.	Resultados de la clasificación de aproximadamente 8000 imágenes. . .	90
8.2.	Resultados del análisis con momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 obtenidos de las formas de la imagen.	95
8.3.	Resultados del análisis contabilizando la cantidad de objetos presentes en la imagen.	95
8.4.	Resultados usando la extracción y el análisis de histogramas de texturas RGB de zonas de la imagen.	95
8.5.	Resultados del análisis, combinando los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 , la contabilización de objetos y, la extracción y análisis de texturas RGB.	96
8.6.	Resultados del análisis usando la distancia euclidiana entre centroides de las masas de objetos.	96
8.7.	Resultados del análisis utilizando el perímetro de las figuras.	96
8.8.	Resultados del análisis de un conjunto de 945 imágenes sin componente (soldaduras en posición horizontal o vertical).	96
8.9.	Esta tabla muestra un a comparación de las características del problema y de la solución propuesta.	101
8.10.	Tabla comparativa de este sistema y del desarrollado por Lin Shih-Chieh	102
B.1.	Error de convergencia por época.	124
B.1.	Error de convergencia por época.	125
B.1.	Error de convergencia por época.	126
B.2.	Resultado de la clasificación del conjunto de entrenamiento.	127
B.2.	Resultado de la clasificación del conjunto de entrenamiento.	128
B.2.	Resultado de la clasificación del conjunto de entrenamiento.	129

Capítulo 1

Introducción

Las empresas que fabrican productos electrónicos necesitan evaluar la calidad de éstos durante las fases de fabricación para tener una producción confiable. Entre estas fases se encuentra el ensamblaje de tarjetas de montaje de superficie donde se realiza la colocación de componentes electrónicos sobre las tarjetas, como son resistencias, capacitores, diodos, circuitos integrados, etc.

En la fase de ensamblaje de tarjetas de montaje de superficie es necesario que cada componente esté colocado correctamente sobre la tarjeta, ya que de no ser así, el producto final funcionara de manera incorrecta. Es importante identificar las tarjetas con error para impedir que avancen a procesos posteriores en la línea de producción.

El problema de la colocación incorrecta de un elemento electrónico sobre una tarjeta comprende que éste se encuentre volteado o desplazado cierta distancia de la región a la que pertenece, que un componente distinto esté en su lugar, o que no esté el componente en el lugar donde debería de estar.

Un método tradicionalmente utilizado para llevar a cabo la identificación de errores en las tarjetas es la inspección visual. Éste consiste en que un ser humano realiza de manera óptica la verificación de la colocación correcta de componentes electrónicos en una tarjeta. El ser humano entrenado para llevar a cabo esta tarea, con facilidad puede identificar un error. Por otro lado, también puede sugestionarse o cansarse, lo cual ocasiona que cometa errores al reconocer desperfectos. Por estas razones se necesita reducir la intervención humana haciendo esta tarea de manera automatizada.

Como caso de estudio se toma a la empresa Sony, la cual utiliza la inspección visual en su línea de producción como último recurso para encontrar errores en el control de calidad de sus productos.

Actualmente en Sony, durante el proceso de ensamblaje de componentes electrónicos se realiza una inspeccion automatica la cual es efectuada por un **sistema de reconocimiento de patrones**. Este sistema puede detectar fallas con cierto nivel

de exactitud sin hacer distinción del defecto en particular. Después de este proceso automático, es adquirida una imagen de la tarjeta catalogada como defectuosa y se le realiza una inspección visual en la que se confirma la existencia del error, el cual puede ser identificado de manera específica en esta fase.

El problema de Sony consiste en confirmar de manera automática la existencia de una falla en la imagen de una tarjeta de montaje de superficie, reduciendo el uso de la inspección visual. Este trabajo de investigación se enfoca a resolver sólo una pequeña parte de este problema, la cual consiste en confirmar (garantizar) la falta de un capacitor o resistencia en una imagen que no contienen etiquetas de números o letras como se muestran en la figura 1.1. La implementación se realizó en lenguaje ANSI C sobre una plataforma GNU/Linux.

Debido a la gran diversidad de casos de tarjetas defectuosas que se pueden presentar, es imposible utilizar un solo método que sea capaz de resolver al 100 % el problema de la identificación de la existencia de capacitores y resistencias. Por lo anterior es necesario utilizar varios métodos los cuales se complementen entre ellos para obtener la eficiencia requerida en la identificación. En muchas ocasiones un método no cubre por completo las deficiencias de otro, por lo que hay que utilizar cuantos sean necesarios hasta obtener la efectividad buscada.

Como material de análisis, Sony proporcionó un conjunto de aproximadamente 8000 imágenes de secciones de tarjetas catalogadas como defectuosas (figura 1.1). Estas imágenes pertenecen a una gran cantidad de componentes electrónicos, de las cuales sólo se analizan las que corresponden a capacitores y resistencias.

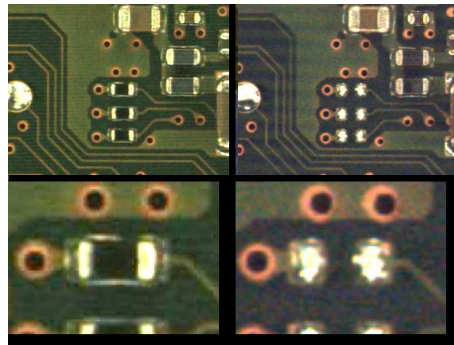


Figura 1.1: Esta es sólo una de las 8000 imágenes del corpus de análisis, las cuales en su mayoría pertenecen a distintos tipos de tarjetas con diferentes tipos de componentes electrónicos.

El proceso de análisis e identificación de la existencia de capacitores y resistencias que se propone es el mostrado en la figura 1.2. Éste es aplicado a cada imagen del conjunto de análisis.

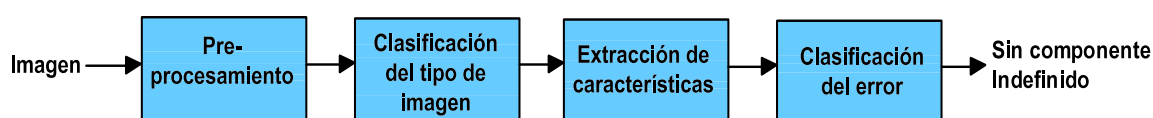


Figura 1.2: Secuencia de procesos para identificar componentes de tipo rectangular.

El primer procedimiento es el *preprocesamiento*, el cual recibe como entrada una imagen a color perteneciente a una tarjeta. Los procesos realizados a la imagen recibida son el almacenamiento, extracción de la región útil, conversión a tonos de gris, suavizado, binarización, extracción de la textura a color y la obtención de la suma y resta de histogramas. Estos procesos son para acondicionar a la imagen para procedimientos posteriores.

En el segundo procedimiento, denominado *clasificación de la imagen*, se hace uso de una red neuronal para determinar a qué tipo de clase pertenece el componente mostrado en la imagen actualmente examinada.

El tercer proceso, nombrado como *extracción de características*, recibe información solo de imágenes pertenecientes al conjunto de capacitores y resistencias. Los datos obtenidos en este procedimiento son trascendentes para la identificación del error. Estos datos son los “momentos invariantes” de la soldadura o de los extremos de capacitores y resistencias, la intersección de “histogramas de brillo y contraste” de algunas regiones de la imagen y la “contabilización de objetos”.

El último proceso es la *clasificación del error*, en la cual se hace uso del clasificador de distancia mínima para determinar si existe o no un capacitor o una resistencia en la imagen analizada, basándose en los momentos invariantes, la contabilización de objetos y la intersección de histogramas de brillo y contraste.

1.1. Definición del problema

Muchas empresas, entre ellas Sony, necesitan reducir el número de componentes electrónicos buenos detectados como defectuosos por un sistema de inspección visual automática (*AVI System*)¹ antes de ser inspeccionados visualmente por un humano².

La respuesta del sistema de inspección cuando detecta una anomalía, es una imagen a color de la región donde se determinó que existe el defecto (sin hacer distinción del problema), de otro modo no retorna nada.

¹La definición y características de los sistemas AVI están mencionadas en la sección 2.4.

²La inspección visual se menciona en la sección 2.3.

Las características de las imágenes resultantes y de los elementos mostrados en ella, son las siguientes:

- Son imágenes a color.
- En una zona se muestra un elemento bueno como referencia, mientras que en otra zona muestra la región donde se detecto el defecto.
- Pueden pertenecer cualquier tipo de elemento.
- El ambiente no está controlado, el color es significativamente diferente entre imágenes.
- No se sabe que elemento se busca, se desconocen los atributos de éste (dimensión, orientación, etc.).
- La posición del elemento que se necesita identificar no es precisa en la imagen.
- No existe más información que la misma imagen.

Las características presentadas anteriormente plantean la necesidad de una metodología o sistema intermedio entre el sistema AVI y la inspección visual humana que tenga las siguientes cualidades:

- Que maneje imágenes a color.
- Que ocupe la misma imagen para catalogar el tipo de elemento y defecto buscado, basándose en la zona buena.
- Que tolere diferencias de color, brillo y contraste de la imagen.
- Que tolere variaciones importantes en la posición de los elementos buscados.
- Que reduzca el número elementos buenos catalogados como malos.

. En este caso, la necesidad primordial es identificar si existe un componente de tipo rectangular con dos terminales, como lo son los capacitores y resistencias, ya que estos cumplen con las características morfológicas y de ensamble buscadas.

1.2. Organización de la tesis

Esta tesis está dividida en los siguientes capítulos:

Capitulo 2. Antecedentes. Este capítulo se refiere a conceptos y características de los sistemas de inspección visual automática. También se menciona el estado del arte.

Capítulo 3. Proceso general. Se explica de manera general el sistema propuesto, la forma en cómo se realiza el proceso de clasificación de imágenes y como se lleva a cabo el análisis y confirmación de la existencia de un capacitor o una resistencia en una imagen.

Capítulo 4. Preprocesamiento. Se describe cada uno de los subprocesos realizados para preparar los datos utilizados en la clasificación de la imagen y la extracción de características.

Capítulo 5. Clasificación de la imagen. Se menciona la utilización de una red neuronal de propagación hacia atrás para clasificar un conjunto de imágenes de análisis. Son descritas las características de la red: elementos, arquitectura y algoritmo de entrenamiento.

Capítulo 6. Extracción de características. En este capítulo se explica qué son, para qué sirven y cómo se obtienen: los momentos invariantes, cómo se realiza la intersección de histogramas de brillo y contraste de imágenes a color, y la contabilización de objetos. Además se describen brevemente algunos otros métodos para obtener información de las imágenes.

Capítulo 7. Clasificación del error. Se describe como haciendo uso del clasificador de distancia mínima se determina la existencia de un capacitor o resistencia, utilizando los momentos invariantes, la contabilización de objetos y la intersección de histogramas de brillo y contraste.

Capítulo 8. Implementación y resultados. En este capítulo se indica cómo se realizó la implementación del sistema clasificador de imágenes y del analizador e identificador de la existencia de capacitores y resistencias utilizando las metodologías descritas a lo largo de este trabajo. Se publican los resultados obtenidos de la implementación utilizando conjuntos de prueba definidos.

Capítulo 9. Conclusiones y trabajo futuro. Se presentan las conclusiones obtenidas y el trabajo futuro que se concibe para el mismo.

Capítulo 2

Antecedentes

En este capítulo se mencionan algunos conceptos y datos de interés referentes a la inspección visual automática. También se mencionan trabajos realizados, sus características y tipo de análisis que realizan. La información referida es útil para ayudar y comprender el problema y las limitaciones de este trabajo.

2.1. Visión artificial

La visión artificial o también llamada visión de máquina (*machine vision*) es la aplicación de la visión por computadora en la industria. Entre sus subáreas se encuentra la inspección visual automática que se utiliza para el control de calidad en la fabricación de productos [1].

La visión por computadora (*computer vision*) es un área muy extensa que tiene como objetivo duplicar o simular el efecto de la visión humana de manera electrónica proveyendo a las computadoras de habilidades de percepción como la de los humanos [1, 2]. Que las computadoras tengan conciencia de su entorno, comprendan los datos percibidos, tomen decisiones apropiadas y aprendan de su experiencia en medida que mejoran su desempeño [2, 3]. Se utilizan varias técnicas como el procesamiento de imagen, el reconocimiento de patrones y la inteligencia artificial [1].

La industria de la visión artificial es un segmento de la “gran industria” que se caracteriza por el uso de imágenes electrónicas. Ofrece soporte tecnológico (generalmente a la producción) para sustituir las funciones visuales humanas. Provee sistemas que incorporan técnicas que realizan decisiones basadas en una funcionalidad equivalente a la visión. Su funcionalidad final es controlar procesos, calidad, máquinas y robots [2].

Los requerimientos para el diseño y desarrollo de un sistema de visión artificial varían dependiendo del dominio de la aplicación y están relacionados con la tarea que

necesita realizar, por lo que no existe un sistema de visión capaz de realizar todas las tareas requeridas en cada una de las áreas [4, 5].

A los sistemas de visión específicamente destinados al entorno industrial se les denomina comúnmente sistemas industriales de visión artificial (*industrial machine vision systems*). Se asume que la escena de estos sistemas es generalmente en dos dimensiones, con conocimiento de sus elementos y frecuentemente en un ambiente visualmente contrastante. La retroalimentación es necesaria ya que no son perfectos y no están libres de errores, por lo que se requiere obtener información para poder tomar decisiones y realizar acciones correctivas para asegurar que la calidad de los resultados y la productividad se mantengan [1].

Con el uso de visión artificial en la producción masiva se intenta mantener el 100 % de calidad en todas sus fases de fabricación (producto intermedio y terminado) [6]. La visión artificial representa una parte del control universal de calidad en la manufactura. El control universal es dirigido por datos del proceso de fabricación, los cuales son de suma importancia para mejorar la operatividad de toda la producción [7].

El problema de la verificación de calidad en la producción es muy extenso, ya que este depende de lo que se considere como un defecto, el cual varía de acuerdo a los estándares de calidad del producto [5].

Existen tipos de inspección que no son por muestreo, ya que es necesario examinar el 100 % de las piezas fabricadas [8].

En la actualidad con los avances tecnológicos en los sistemas de computo (alta velocidad, memoria extensa, bajo costo, etc.), técnicas como el procesamiento de imagen, reconocimiento de patrones y la inteligencia artificial han resultado más fáciles de implementar en equipos de visión artificial para la industria [6].

2.1.1. Áreas de aplicación

Una gran cantidad de aplicaciones de la visión artificial se encuentran en las siguientes áreas [6, 8]:

- *Medicine*. Análisis de enfermedades, sangre, DNA, etc. Operaciones con endoscopio, ayuda en diagnósticos, etc.
- *Milicia*. Detección y seguimiento de objetos, análisis de terreno, armas inteligentes, etc.
- *Robótica*. Guía de robots móviles y de brazos de robot, etc.

- *Agricultura.* Análisis de plantaciones, en algunos casos a través de imágenes vía satélite.
- *Identificación y seguridad.* Identificación automática de huellas dactilares, reconocimiento de caras, etc.
- *Redes de comunicación.* Compresión de imágenes, teleconferencia, etc.
- *Astronomía.* Búsqueda, reconocimiento y seguimiento de objetos en el espacio.
- *Industria en general.* clasificación, búsqueda, control de calidad, procesos, máquinas y robots, etc.

2.1.2. Motivaciones para el uso de la visión artificial en la industria

Las tecnologías de visión artificial mejoran la producción y ofrecen una ventaja competitiva a las industrias que las emplean. Las razones son las siguientes [1]:

- **Seguridad y fiabilidad**

Porque es requerida una inspección del 100% en las partes manufacturadas ya que son de mucha importancia. Facilita la consistencia de la calidad en los estándares de inspección.

- **Calidad del producto**

En producciones de gran volumen que utilizan la inspección visual humana en muy pocas ocasiones facilitan la inspección de todas las partes, mientras que las técnicas de inspección visual automática hacen esto posible. La calidad depende de la complejidad de la tarea y de la efectividad requerida por el sistema de producción. Una consideración particularmente importante es en qué parte de la línea de producción se necesita incorporar el sistema artificial.

- **Automatización flexible**

En entornos donde la calidad es asegurada por un sistema, es posible integrar la inspección dentro del ciclo de manufactura, y permitir que esta proporcione retroalimentación para facilitar el control de la producción.

2.1.3. Actividades industriales beneficiadas

Algunas de las actividades industriales que se ven beneficiadas por la utilización de la visión artificial son [4, 8]:

- Fabricación de circuitos integrados.
- Fabricación de componentes electrónicos delicados.
- Producción textil de calidad.
- Realización de productos metálicos finales.
- Manufactura de lentes.
- Elaboración de partes de máquinas.
- Producción de productos de impresión.
- Inspección de calidad en etiquetas, contenedores, motores, cristales, soldaduras, circuitos integrados, dispositivos electrónicos, granito, etc.

2.1.4. Clasificación de los sistemas para la inspección de calidad

Los sistemas de visión artificial dedicados a la inspección pueden ser clasificados de acuerdo a las características que analizan de la siguiente manera [4]:

- **Dimensional:** Verifica si las dimensiones de un objeto están dentro de los márgenes de tolerancia o si los objetos tienen la forma correcta. Se involucran características geométricas de objetos en dos y tres dimensiones.
- **Superficial:** Se buscan rayones, grietas, desgaste o que la superficie tenga un acabado y textura apropiados.
- **Estructural:** Verifica componentes colocados erróneamente, la presencia de componentes diferentes o objetos extra.
- **Operacional:** Se refiere a la verificación de la funcionalidad de acuerdo a los estándares de producción.

2.2. Métodos de inspección de calidad usados en la industria electrónica

La inspección es el proceso mediante el cual se determina si un producto presenta alguna falla. Las fallas en componentes electrónicos (tarjetas, circuitos integrados, condensadores, resistencias, inductores, diodos, etc.) pueden generar un mal funcionamiento del mismo y del equipo en el que están integrados.

La inspección se puede realizar de manera manual o automática. La manual la realiza una ser humano, mientras que la automática la realiza un equipo electrónico.

De manera general los métodos usados para la inspección de componentes electrónicos pueden ser de dos tipos [6, 9]:

- **Métodos eléctricos o con contacto**

Consiste en aplicar corriente a los dispositivos y verificar su funcionamiento individual. En el caso de la tarjetas de circuito impreso (sin componentes montados) para probar su conexión.

En tarjetas terminadas (con dispositivos ya montados) se busca la presencia de cortos entre terminales de dispositivos con lo que se verifica la unión de terminales con la base de la tarjeta. Sin embargo, si es encontrada una falla, la tarjeta debe ser desechada o reparada, lo cual implica gastos adicionales. Por esta razón es usado sobre todo en tarjetas sin dispositivos.

Sin embargo este método no puede hallar fallas potenciales como una mala colocación de componentes.

- **Métodos no eléctricos o sin contacto**

Este usa la **inspección visual** de las tarjetas. Permite hallar fallas que no es posible detectar con métodos eléctricos, como exceso de soldadura, fallas estéticas, unión deficiente, elementos mal colocados o extraños, etc. Algunos de estos defectos pueden no provocar un mal funcionamiento ya terminada la tarjeta, pero potencialmente pueden hacer que falle en algún otro momento.

Con la **inspección visual** se evita que las tarjetas defectuosas pasen a etapas posteriores del proceso de producción con el fin de evitar que afecten la calidad del producto final.

2.3. Inspección visual

Tradicionalmente, la inspección visual para el control de calidad era realizada por un humano experto en la tarea. Aunque el ser humano se adapta a situaciones nuevas de manera muy rápida y realiza la inspección mejor que una máquina en la mayoría de los casos¹, también es más lento, se cansa, se aburre y sugestiona rápidamente, lo cual lo lleva a cometer errores de apreciación.[4, 6, 9]. En la mayoría de los casos el humano que realiza la inspección clasificará de manera diferente el mismo error en distinto momento [5].

La inspección visual de manera manual es muy lenta en relación a los niveles de producción, es costosa, y no asegura el alto nivel de calidad requerido por la industria. El incremento en el uso de tecnología para la producción automática y la **inspección visual automática** han ido rápidamente substituyendo la inspección visual humana [6].

2.4. Inspección visual automática

La inspección visual automática (*automatic visual inspection*, AVI) o también llamada inspección óptica automática es una de las áreas mas difíciles e importantes de la visión artificial, se encarga de la detección y clasificación de defectos funcionales y cosméticos [1, 5, 6].

En el mundo real, uno de los mayores obstáculos de los procesos de identificación visual es la gestión de la incertidumbre presentada por el entorno. Un sistema de inspección está diseñado para detectar y clasificar defectos de tipo *abc* bajo condiciones de operación *xyz*. Inevitablemente en el mundo real los defectos cambian de *abc* a *def* y las condiciones de operación lo hacen de *xyz* a *rst*. El sistema se encuentra bajo una continua evolución [5, 10].

La incertidumbre se presenta debido a las variaciones en la producción como los cambios en procesos, en la materia prima o el incremento en la demanda y calidad, entre otras.

Debido a la extensa cantidad de posibilidades en las que puede presentarse un producto en el mundo real es prácticamente imposible que un sistema resuelva el problema de la verificación de calidad con un 100% de efectividad. Por esta razón, los requerimientos para el diseño y desarrollo de un sistema AVI (*automatic visual inspection system*) varían dependiendo de las características del problema.

¹Para el humano es muy fácil reconocer un error de manera visual.

La efectividad del reconocimiento y la clasificación depende mucho de la calidad de las imágenes adquiridas. Un alto grado de eficiencia es obtenido bajo ciertas condiciones de iluminación y bajo ruido. Los sistemas de inspección deben de ser capaces de diferenciar entre variaciones o defectos aceptables e inaceptables en los productos. Deben de ser lo suficientemente rápidos con relación a la demanda del entorno de producción [4].

En los sistemas AVI, la iluminación es siempre un factor crítico que debe ser manejado cuidadosamente. Las imágenes frecuentemente utilizadas son binarias². La estrategia de análisis es extraer características sobresalientes (generalmente medidas descriptivas de las formas) y tomar alguna decisión basada en la discriminación [1].

Los resultados obtenidos de un sistema AVI pueden ser utilizados para controlar un proceso de producción o ser propagados a otros dispositivos externos para procesos posteriores. También se puede caracterizar defectos o elementos inválidos y así poder realizar acciones para reportar y corregir estas fallas y remplazar o remover el producto de la línea de producción [4].

Los sistemas de manufactura requieren inspección automática para garantizar la calidad a lo largo de sus procesos de producción [7].

En la actualidad cualquier falla en un sistema AVI es compensada por medio de la inspección visual realizada por un operador humano [8].

2.4.1. Clasificación de la inspección

Las inspección visual automática puede ser clasificada en base a su funcionalidad como [1, 6]:

- **Medición:** Correspondiente a las medidas de las características de dimensión y verificación de tolerancias.
- **Inspección:** Se refiere a la verificación de elementos: si existen partes o secciones erróneas, o algún elemento extraño que no debería estar presente.
- **Clasificación:** Referente a la identificación y reconocimiento de partes o elementos.

²Imágenes en blanco y negro.

2.4.2. Secuencia de pasos comunes en un sistema de inspección visual automática

La inspección visual automática se realiza comúnmente utilizando un método de reconocimiento de patrones el cual comprende los siguientes procesos [4, 8]:

1. Adquisición de la imagen

La imagen se obtiene a través de un dispositivo de captura (cámaras, sensores, digitalizadores, etc.) utilizando alguna fuente de energía que en combinación con el elemento de análisis proporcione información sobre su estado. Los tipos de energía utilizados son: la luz visible para fotografía, los rayos X y los rayos γ para radiografía y tomografía, ultrasonido para la geografía, campos magnéticos para resonancia magnética, calor para la termografía, etc.

2. Preprocesamiento

Una vez adquirida la imagen, se intenta aumentar la calidad de esta y prepararla para procesos posteriores con el fin de obtener mejores datos. Algunas de las técnicas empleadas son la eliminación de ruido mediante filtros digitales. También se aplica restauración para mejorar la calidad.

3. Segmentación

Es el proceso mediante el cual una imagen se subdivide en unidades significativas. Permite clasificar los píxeles como pertenecientes a una categoría en concreto. Algunos de los atributos de la imagen que son considerados para segmentar son: la luminancia (brillo) en imágenes en tonos de gris, los canales RGB en imágenes a color, los bordes y texturas.

4. Extracción de características

Es principalmente la medición de las propiedades como: área, perímetro, forma, posición, texturas en regiones, intensidades de gris y color, etc.

5. Clasificación

Cataloga las regiones segmentadas en clases según las características que posean. Cada región es asignada a un grupo con características similares. Las

clases o grupos representan todos los posibles tipos de regiones que se espera que existan en la imagen.

Comúnmente las clases existentes son dos: falsas y falsas alarmas. La clase falsa corresponde a piezas malas y la de falsas alarmas a piezas buenas catalogadas como malas.

La clasificación se realiza en un sistema automático de inspección luego de una etapa de entrenamiento supervisado. En esta etapa se hace un estudio estadístico de las características que se extraen de objetos cuyas fallas son conocidas a priori.

2.5. Falsos positivos y falsos negativos: criterios de eficiencia

La eficiencia de un sistema de visión artificial está relacionada con los falsos positivos y los falsos negativos.

- **Falsos positivos:** Se refiere al número de productos que son clasificados como defectuosos sin tener fallas. Estos también son llamados *falsas alarmas*.
- **Falsos negativos:** Indica la cantidad de piezas defectuosas que no fueron detectadas por el sistema.

Lo ideal es que las tasas de falsos positivos y negativos sean cero, que el sistema catalogue con un 100% de eficiencia la calidad de un producto, sin embargo en la realidad esta situación difícilmente se consigue.

La implementación de los sistemas AVI y la sintonía de sus parámetros involucran un compromiso entre los falsos positivos y los falsos negativos.

Al querer detectar el 100% de las fallas existentes se tiende a obtener un gran número de falsos positivos (que no tienen defectos). Del mismo modo, si se desea eliminar el total de falsos positivos posiblemente no se detecten algunas de las fallas reales elevando el número de falsos negativos (con errores).

En casos en los que el control de calidad debe de garantizar que los productos clasificados como buenos estén libres de fallas, se ajusta el sistema AVI de tal manera que se garantice una tasa baja de falsos negativos. Sin embargo, el costo económico de esta alternativa puede ser elevado, ya que como se mencionó, muchas veces esta estrategia conlleva a que se eleve la tasa de falsos positivos, lo que implica que se

estarían desechando productos que se encuentran en perfecto estado y que ha sido clasificados como defectuoso [4, 8].

Lo anterior se debe a las características de los elementos inspeccionados y de los métodos utilizados para el análisis. Frecuentemente las características no son útiles para realizar una distinción entre elementos o los métodos no pueden lidiar correctamente con ellas.

2.6. Defectos en las tarjetas de montaje de superficie

Con el uso resiente de tecnología de montaje de superficie (*surface mount technology, SMT*)³ en la industria de la electrónica, los dispositivos de montaje de superficie (*surface mount devices, SMD*) como capacitores, resistencias, circuitos integrados, entre otros, se han hecho más pequeños. Se ha incrementado la densidad de los componentes en tarjetas de circuito impreso (*printed circuit boards, PCB*).

Sin embargo con estos avances tecnológicos han surgido nuevas características en los tipos de defectos, causando que la inspección visual sea más complicada de realizar [11, 12].

Los tipos de defectos que se presentan en tarjetas de montaje de superficie (tarjetas de circuito impreso) son clasificados en cuatro categorías [12, 13]:

- Componentes mal colocados (aproximadamente 10%)⁴
- Componentes con mala polaridad o invertidos (aproximadamente 25%)
- Componentes ausentes (aproximadamente 20%)
- Defectos de soldaduras (aproximadamente 55%)

³SMT es una colección de tecnologías de métodos científicos y tecnológicos para diseñar, construir y probar productos con componentes electrónicos que son montados sobre tarjetas de circuito impreso sin agujeros para plomo [11].

⁴Componentes extraños o en lugares que no deberían de ir, desalineados.

2.7. Composición y comportamiento de la soldadura

La pasta de soldadura contiene bolas microscópicas de soldadura, comúnmente de estaño y plomo, con una cubierta de oxido, y un solvente que sirve para deshacer la cubierta (figura 2.1) [11].

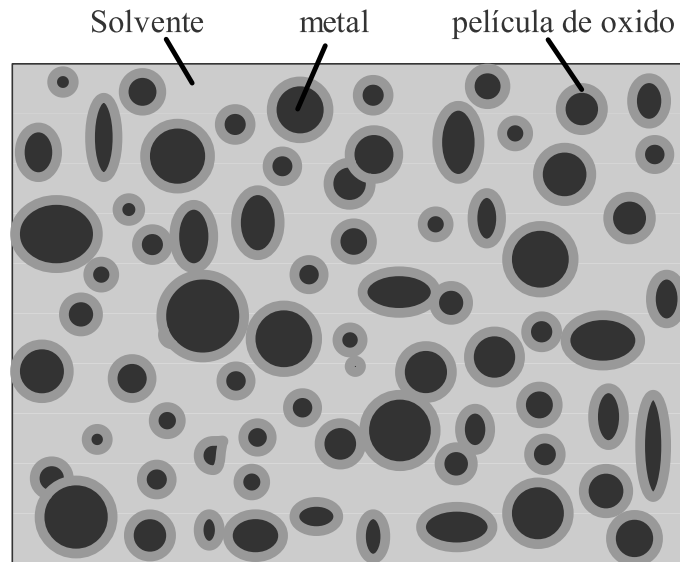


Figura 2.1: Imagen de la mezcla de soldadura.

La capacidad del solvente para reducir el oxido que esta presente en la pasta de soldadura y el incremento en la temperatura y el tiempo durante las fases de precalentamiento y secado, pueden causar una rápida fuga del solvente volátil que está atrapado, dando como resultando la expulsión de pequeñas cantidades de metal que formaran bolas de soldadura indeseables. Estas pequeñas bolas de metal dispersadas alrededor de los empalmes en el substrato [11].

2.8. Estado del arte

El problema de verificación de calidad en la industria es muy importante y necesario de resolver, por lo que se han realizado un sin número de trabajos, tanto de carácter científico como industrial, para dar soluciones adecuadas a algunas de las múltiples facetas que se presentan.

Existe una extensa cantidad de posibilidades en las que puede presentarse el problema de la verificación de calidad en el mundo real. Debido a las particulares que presenta cada problema en particular, generalmente dadas por el entorno en el que habita (tipo de producto, iluminación, proporciones, dimensiones, forma, etc.) y el tipo de resultados que se desean obtener. Esto provoca que no exista un método o sistema que pueda resolver totalmente el problema⁵.

La información y los desarrollos tecnológicos del sector industrial, en la gran mayoría de los casos son inaccesibles por terceros. Por otro lado, el sector científico proporciona soluciones frecuentemente más teóricas en las que el problema presenta condiciones menos apegadas al mundo real.

Cabe mencionar que debido a las características específicas del problema atacado, no existen muchos trabajos que busquen o tengan resultados parecidos en circunstancias similares. En otros casos solo una parte de éstos tienen cosas en común.

Se mencionan a continuación trabajos relacionados al problema de esta tesis, los cuales sirvieron como base y referencia.

■ Diseño en FPGA de un circuito comparador de imágenes

En el 2005, Miguel Ángel Sánchez y Adriano de Luca diseñan un circuito comparador de imágenes, que debe usarse en un sistema automático de inspección visual para tarjetas de circuito impreso con dispositivos de soldadura superficial. Con esta arquitectura se pretende reducir los tiempos y los costos de los sistemas de inspección actuales.

Los tiempos son reducidos ya que el algoritmo de procesamiento de imagen es implementado en hardware, para ello el sistema hace uso de un FPGA. El uso del FPGA permite obtener un sistema compacto y menos costoso que los sistemas de inspección actuales, ya que toda la parte electrónica del control y procesamiento de imagen puede quedar alojado en un solo circuito.

⁵Esto se menciona en las secciones 2.1 (página 7) y 2.4 (página 12).

El algoritmo que utiliza se basa en la comparación de histogramas de dos imágenes, una correspondiente a una tarjeta sin fallas (imagen base), y la otra a la imagen de la tarjeta a analizar (imagen de prueba).

La comparación se realiza tomando sucesivamente segmentos de 40x40 píxeles, para obtener un histograma de brillo y contraste que representa las características de textura del segmento analizado. Este histograma es característico de cada segmento de la imagen, de tal forma que hay un histograma para el circuito impreso, otro para la pasta, otro para la terminal y otros para sus combinaciones. Una variación en el histograma indica diferencias en las imágenes analizadas y por lo tanto fallas en el producto.

La arquitectura hardware emplea múltiples procesadores y un sistema para la distribución de datos con el cual se acelera de forma importante la respuesta del sistema.

Otras aplicaciones del sistema propuesto se limitan a actividades en donde el plano de la cámara se mantenga a una distancia fija del objeto a fotografiar y las condiciones de iluminación puedan ser controladas, situación presente en múltiples aplicaciones industriales [9].

- **Localización de SMD mediante técnicas de visión artificial implementadas en hardware**

En el 2006, Luis Francisco Barbosa y Adriano de Luca diseñan e implementan en FPGA, un sistema de localización SMD (Surface Mounted Device).

Este un sistema de visión en hardware para el proceso industrial automático de colocación de componentes de montaje superficial sobre tarjetas electrónicas con circuito impreso, por medio del cual se pueda corroborar su correcta colocación y garantizar así la calidad del producto.

Este sistema utiliza la técnica *Matching for segmentation / recognition* en combinación con el clasificador KNN (vecino más cercano) mismo que consiste en la realización de múltiples comparaciones de los patrones de la plantilla objeto a localizar a través de los patrones de la imagen tomada por una cámara.

El criterio de localización se basa en un umbral de distancia entre patrones. La arquitectura emplea múltiples procesadores y un sistema para la distribución de datos con el cual se acelera de forma importante la respuesta del sistema.

Otras aplicaciones del sistema se limitan a actividades en donde el plano de la cámara se mantenga a una distancia fija del objeto a fotografiar y las condiciones de iluminación puedan ser controladas, situación presente en múltiples aplicaciones industriales [8].

■ **Un sistema de inspección visual para dispositivos de montaje superficial sobre tarjetas de circuito impreso**

En el 2006, Lin Shih Chieh y Su Chia Hsin desarrollan un sistema de inspección visual confiable y rápido para la inspección de tarjetas de circuito impreso. Esta implementación es de inspección estructural⁶, detecta si un componente de tipo capacitor o resistencia se encuentra en posición correcta, hace falta, está desalineado o invertido.

Para lograrlo, el proceso de la inspección fue dividido en dos etapas.

- La primera etapa es un discriminador de elementos bien definidos como correctos. Solo obtiene una característica o indicador de la imagen. Este indicador es usado para eliminar rápidamente partes (imágenes) buenas del conjunto de análisis y reducir el tiempo de proceso.
- La segunda etapa utiliza una red neurona de propagación hacia atrás para realizar la clasificación. Esta red consiste de 5 nodos (neuronas) de entrada, 9 nodos intermedios y 2 nodos de salida. Los nodos de salida representan los 4 resultados posibles: normal, faltante, desalineado e invertido. Se usa una cantidad mayor de características de la imagen para alcanzar mejores resultados en la clasificación. Las pruebas indicaron que la implementación necesita más de tres indicadores para que identifique defectos correctamente.

Se ocuparon 558 imágenes de para el entrenamiento, catalogadas en 18 tipos. El sistema ya entrenado fue probado con 1949 muestras colectadas de la línea de producción. En el primer etapa la tasa de falsas alarmas⁷ fue de 12.3%. En la segunda etapa la tasa de falsas alarmas es menor del 5%.

El tipo de imágenes que utilizan a pesar de ser distintas, presentan condiciones de iluminación constantes y las posiciones de los componentes electrónicos son

⁶Véase la sección 2.4.1.

⁷Véase la sección 2.5.

constantes. El sistema utiliza características propias de la imagen mas no elementos en ella.

Los indicadores analizados son: la cantidad de pixeles blancos en una imagen binarizada para detectar una mala orientación del componente (este invertido), distribuciones de los niveles de gris a través de histogramas para detectar una mala alineación, coeficiente de correlación para detectar la falta de componente, regiones para intentar reducir el efecto de la iluminación, y el alto contraste [13].

Capítulo 3

Proceso general

El proceso propuesto se enfoca a solucionar el problema de confirmar automáticamente si existe un capacitor o una resistencia en una imagen de una tarjeta de montaje de superficie.

Como es importante asegurar un 100 % de efectividad en la detección de imágenes sin componente, muchas serán clasificadas como que si existe componente. Esto se debe a la relación entre falsos positivos y falsos negativos¹.

La imagen de la tarjeta es la única información que se tiene para resolver el problema. No se posee información acerca de las características de la tarjeta o del componente electrónico con falla, por lo que no se puede saber qué se está buscando en la imagen. Por esta razón es necesario realizar una clasificación de cada una de las imágenes para definir a que conjunto o clase pertenece. En este trabajo se probó una red neuronal para realizar la clasificación.

Este proceso se divide en dos grandes bloques, el primero para clasificar el tipo de imagen y el segundo para análisis y confirmación de la existencia de un capacitor o resistencia en la imagen clasificada.

Los pasos que se llevan a cabo para clasificar el tipo de imagen se analizan en la sección 3.3. Los pasos para el análisis y confirmación de un dispositivo se enlistan en la sección 3.4.

¹Véase su definición en la sección 2.5.

3.1. Características de las imágenes a analizar

Los procedimientos ocupados en este trabajo de investigación están definidos para imágenes con determinadas características.

De manera general, todas las imágenes que se analizan están divididas en cuatro partes de forma tal que permite detectar la falla rápidamente (como se muestra en la figura 3.1). La parte superior izquierda presenta la zona de la muestra de una tarjeta en estado correcto donde se detectó un problema. La parte superior derecha muestra la zona de la tarjeta a la que pertenece la imagen donde se detecto una falla. La parte inferior izquierda exhibe la zona ampliada de la tarjeta de muestra. La parte inferior derecha muestra la zona ampliada donde fue encontrada la falla correspondiente a la tarjeta de la imagen.

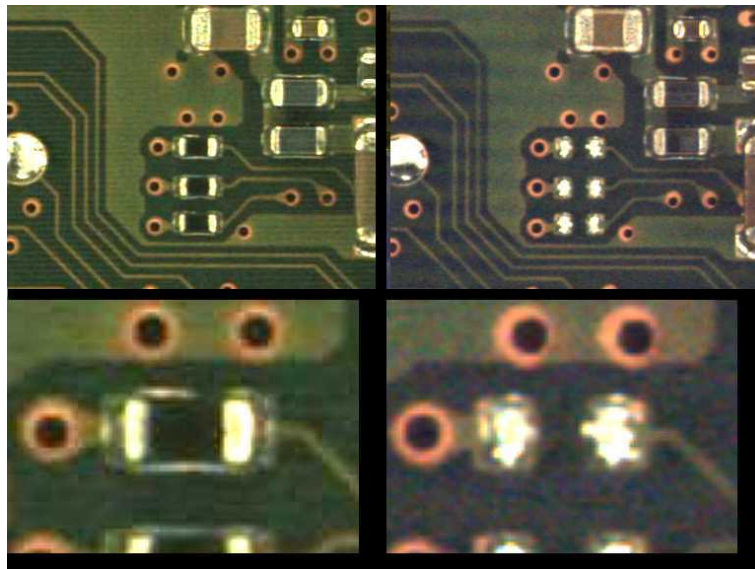


Figura 3.1: La imagen se encuentra dividida en cuatro partes para facilitar su análisis.

De manera particular, el tipo de imágenes de capacitores y resistencias que se analizan son como los mostradas en la figura 3.2. Estas imágenes contienen el componente centrado en la región de análisis y no tienen letras o números.

3.2. Elementos de las imágenes considerados para el análisis

Para realizar el análisis se necesita considerar qué elementos de la imagen son útiles (que aportan información). De manera general, cuando no existe componente, las formas más descriptivas y constantes en la imagen son las soldaduras (imagen derecha de la figura 3.2). Por otro lado cuando existe un capacitor o una resistencia, los elementos más notorios y persistentes son los extremos de estos (imagen izquierda y central de la figura 3.2). Se pueden apreciar las formas distintas entre extremos de capacitores y resistencias y soldadura.



Figura 3.2: La imagen de la izquierda es de un capacitor, la imagen del centro es una resistencia y la imagen de la derecha muestra soldaduras de la base sin presencia del elemento.

Otro elemento muy descriptivo es la textura que se presenta en la zona del componente rectangular y la tarjeta. Cuando existe un componente, hay una diferencia entre las texturas de la tarjeta y la región central donde se encuentra este último. Cuando no existe componente, la región central tiene una textura similar a la tarjeta. Las texturas de ambos casos se pueden ver en la figura 3.2.

3.3. Proceso de la clasificación de la imagen

Este proceso se utiliza para identificar a qué clase pertenece una imagen (figura 3.3). Tiene dos fases: preprocesamiento y clasificación del tipo de imagen. Como entrada se utiliza una imagen sin clasificar y como salida se obtiene la clase a la que pertenece la imagen.

3.3.1. Preprocesamiento

Como primer paso en todos los procesos generales para poder trabajar con la imagen de la tarjeta, se necesita almacenar ésta en disco duro en formato PPM (*Portable Pixmap Format*)², el cual puede representar una imagen a color y es muy fácil de

²Véase la sección 4.1.

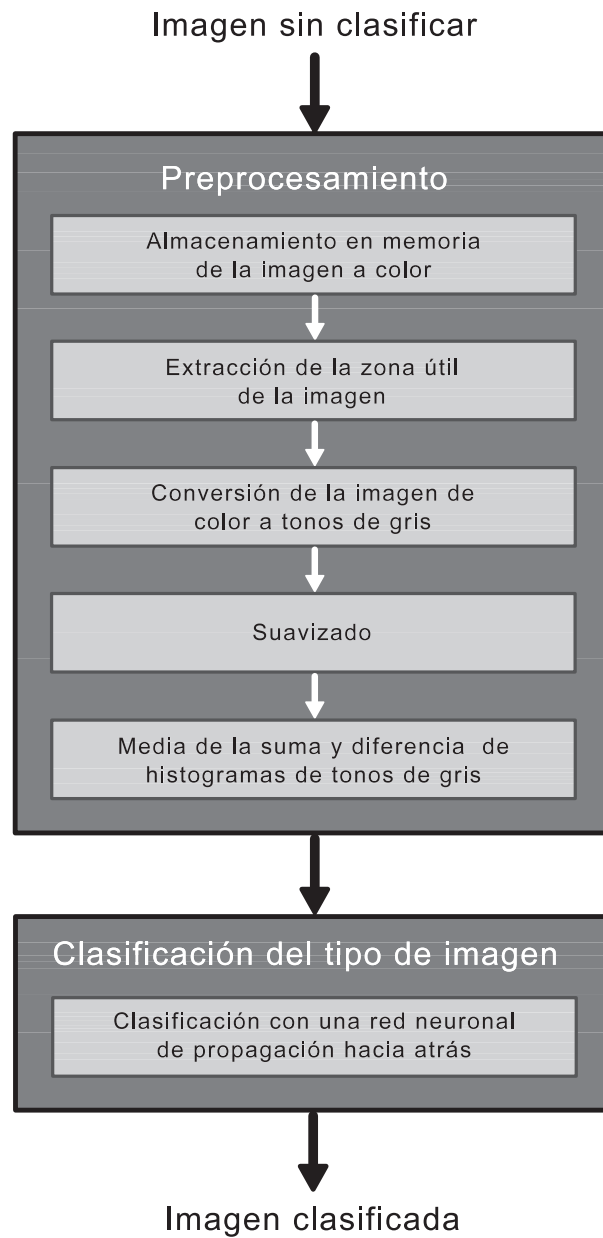


Figura 3.3: Pasos realizados para la clasificación de la imagen.

manejar. En memoria, la imagen se almacena en una matriz bidimensional.

Una vez almacenada la imagen en memoria se realiza una extracción de la región útil de ésta, correspondiente a la parte que describe el componente base o muestra correcta que está en la parte inferior-izquierda de la imagen. La extracción se usa para no procesar la imagen completa.

La imagen a color extraída se convierte a una imagen en tonos de gris la cual ocupa menos espacio en memoria, 1 byte por pixel en vez de tres cuando está en color, y proporciona información suficiente correspondiente a la imagen original.

A la imagen en tonos de gris se le aplica un filtro de suavizado el cual es utilizado para difuminar y reducir ruido. El difuminado sirve para remover pequeños detalles y cubrir pequeños huecos en los trazos de una imagen.

La imagen suavizada en tonos de gris es particionada en $n \times k$ partes. De cada una de las partes es obtenida la media de la suma y diferencia de histogramas. Estas partes son una representación numérica de la textura de la imagen.

3.3.2. Clasificación del tipo de imagen

Las medias de la suma y diferencia de histogramas son utilizadas como vector de características (patrón), el cual es introducido a una red neuronal de propagación hacia atrás (backpropagation) para clasificar a qué clase pertenece la imagen. Esta red es previamente entrenada con medias de la suma y diferencia de histogramas de un conjunto de imágenes de entrenamiento que definen un grupo de clases.

3.4. Proceso de análisis e identificación del error

Este proceso determina si existe un capacitor o resistencia en una imagen correspondiente al componente buscado (figura 3.4). El proceso se divide en tres fases: preprocesamiento, extracción de características y clasificación del tipo de error.

Como entrada del proceso se utiliza una imagen clasificada como de capacitor o resistencia, y como salida se obtiene: *sin componente* o *indefinido*.

Como es importante asegurar un 100 % de efectividad en la detección de imágenes sin componente, muchas serán clasificadas como que si está presente uno (falsas alarmas)³. Debido a esto las imágenes clasificadas como que si tienen componente son

³Véase su definición en la sección 2.5.

ambiguas ya que pueden “si tener” o “no tener”. Por eso, una manera más conveniente de llamar a esta respuesta es: *indefinido*.

Por la razón anterior, la respuesta *sin componente* se refiere a una imagen que coincide con los parámetros correctos de identificación. La respuesta *indefinido* es para imágenes que no cumplen con los parámetros correctos.

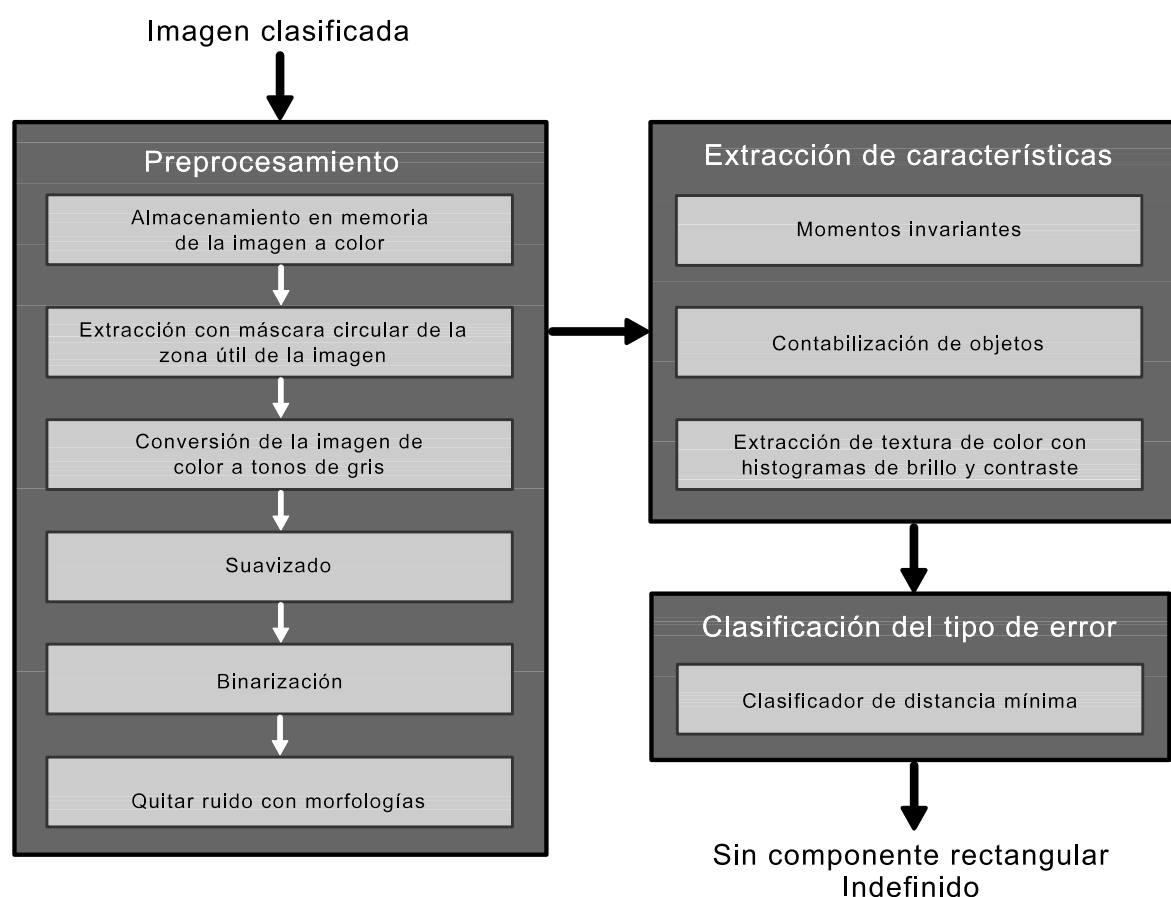


Figura 3.4: Pasos realizados para el análisis e identificación la existencia de capacitores y resistencias.

3.4.1. Preprocesamiento

Una vez almacenada y catalogada la imagen, se procede a realizar su análisis de acuerdo a la clase a la que pertenece. En este caso se analizan las imágenes que son clasificadas como capacitores o resistencias.

Se realiza una extracción de la zona útil de la imagen original correspondiente a la parte que muestra el componente que necesita ser confirmado (parte inferior-derecha). Esta nueva imagen extraída está acotada por una máscara circular, la cual delimita

de manera uniforme la región que rodea al componente. Después ésta es convertida a una imagen en tonos de gris y se le aplica un filtro de suavizado para eliminar ruido.

Se realiza una binarización a la imagen suavizada obtenida anteriormente. La binarización consiste en transformar una imagen en tonos de gris a una imagen de dos tonos (blanco y negro) la cual es utilizada para la extracción de objetos.

La imagen binarizada necesita ser limpiada de pequeños objetos que consideramos como residuos, resultantes del proceso de binarización. Para realizar esto se utilizan morfologías como son la erosión y dilatación⁴.

3.4.2. Extracción de características

En este proceso es donde se obtienen los datos necesarios para realizar la clasificación del error. Estos datos son los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 correspondientes a la forma de la soldadura y los extremos de los componentes, el número de objetos en la imagen binarizada y los histogramas de brillo y contraste de regiones específicas de la tarjeta.

Los momentos invariantes de una imagen son cuantificadores de distribución de puntos. En el caso de los capacitores y resistencias, cuantifican la distribución de puntos de éstos o la soldadura en la imagen binarizada.

Debido a la heterogeneidad o salpicado⁵ que a veces se presenta en la soldadura o a la etapa del proceso en el que se encuentra la tarjeta, pueden apreciarse varios elementos en una imagen binarizada. Empíricamente esto se presenta comúnmente en tarjetas sin componente. Entonces se considera que, si existe una cantidad de objetos distinta de dos (haciendo referencia a dos terminales de componente), no existe componente. Esto se abarca en la sección (6.2).

La variación en el brillo y el contraste suelen ocasionar un cambio visible en una imagen. Existe una diferencia entre la textura de la tarjeta y el centro del componte, en caso contrario, cuando no hay elemento ambas texturas son semejantes. Para detectar estas diferencias se extraen las texturas de la imagen a color con histogramas de brillo y contraste de la superficie de la tarjeta y el centro del componente.

⁴Las morfologías son herramientas para la extracción de elementos de una imagen. Véase la sección 4.6.

⁵Véase la sección 2.7

3.4.3. Clasificación del tipo de error

Se realiza la validación de los datos obtenidos de la imagen en el proceso de extracción de características (momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 , el número de objetos y los histogramas de brillo de las texturas de la imagen a color) utilizando el clasificador de distancia mínima. Para entrenar este clasificador se necesitan dos vectores prototipo, los cuales corresponden a estos mismos datos, pero pertenecientes a un conjunto de imágenes de entrenamiento.

Se tienen dos respuestas posibles, las cuales son “sin componente” o “indefinido”⁶.

⁶Véase el inicio de la sección 3.4 (página 27).

Capítulo 4

Preprocesamiento

Se define al preprocesamiento como el paso donde se llevan a cabo procedimientos necesarios para incrementar la probabilidad de éxito en procesos posteriores [14].

En este proceso se realizan procedimientos para la obtención de datos y preparación de información utilizada en el proceso de extracción de características y en el de clasificación del tipo de error. Los procedimientos realizados son los siguientes:

1. Almacenamiento en memoria de la imagen a color
2. Extracción de la región útil de la imagen
3. Conversión de la imagen a color a tonos de gris
4. Suavizado
5. Binarización
6. Quitar ruido con morfologías
7. Obtención de la media de la suma y diferencia de histogramas de imágenes de tonos de gris

La definición y la utilización de estos procedimientos en este trabajo son descritos en el resto de este capítulo.

La implementación de algunos de estos subprocedimientos está fundamentada en la biblioteca para procesamiento de imagen Scimagen¹ [15].

¹Es una librería de procesamiento de imagen de código libre, escrita en lenguaje C. Las funciones básicas con las que cuenta son: rotación, espejeo (rotación a ± 90 y ± 180), cálculo del histograma, convolución espacial (detección de bordes, filtro levanta altas (high boost), etc.), normalización del histograma, igualación del histograma, umbralización global. Las funciones complejas son: transformaciones de Fourier, la discreta coseno y la Hartley, filtrado en frecuencia (pasa-bajas y pasa-altas con perfiles: ideal, coseno alzado, Gauss, Fermi y Butterworth), filtrado homomórfico, creación de la

4.1. Almacenamiento en memoria de la imagen a color

Para poder procesar una imagen es necesario guardarla de alguna forma en un dispositivo de almacenamiento. El formato de almacenaje la imagen debe de permitir un fácil empleo de ésta para que los procesos que requieren información la puedan obtener y procesar sin complicaciones.

En nuestro trabajo ocupamos tres tipos de imágenes: color, tonos de gris, y blanco y negro. Por esta razón el formato de imagen utilizado es el PNM (*Portable Any Map*) el cual es una abstracción de los formatos PBM, PGM y PPM. Podemos representar cualquiera de estos tres formatos con uno solo, lo cual nos ofrece la versatilidad buscada para todos los procesos [16].

Las características de los formatos PBM, PGM y PPM son las siguientes:

- El formato PPM (*Portable Pixel Map*) representa imágenes a color (RGB). Utiliza 24 bits (1 byte por cada canal del RGB) para representar un solo pixel, esto es: 8 bits son para representar el rojo, 8 para el verde y 8 para el azul.
- El formato PGM (*Portable Gray Map*) representa imágenes en tonos de gris (256 escalas de gris que van del negro al blanco). Utiliza 8 bits (1 byte) para representar un pixel.
- El formato PBM (*Portable Bit Map*) representa imágenes en blanco y negro (binarias). Un bit para un pixel.

Cuando se trabaja con la imagen en blanco y negro en memoria, se utiliza 1 byte por pixel a diferencia de cuando esta sin usar (solo almacenada).

imagen logaritmo. Presenta un algoritmo para la extracción de objetos en imágenes binarias y cuenta con una rutina para el adelgazamiento binario (para la representación de una forma a través de su esqueleto). Esta puede ser obtenida de "<http://delta.cs.cinvestav.mx/~fraga/SoftwareLibre/>".

4.2. Extracción de la región útil de la imagen

El análisis sólo se realiza en ciertas partes de la imagen original, por lo cual no tiene caso aplicar procesos a la imagen completa o a partes que no se utilizan. Por esta razón se realiza una extracción de la región de interés. De esta manera el proceso de análisis se reduce en tiempo y costo computacional.

La región útil corresponde a la zona donde se considera que se encuentra el componente buscado. El centro de la región útil debe de corresponder al centro del componente, para que este último sea extraído de manera simétrica.

La extracción de la región útil se presenta de dos maneras dependiendo la fase del problema que se quiera resolver: con máscara o sin máscara.

4.2.1. Extracción sin máscara

Esta extracción es utilizada sólo para obtener la región de interés y puede adaptarse a cualquier dimensión xy . Se realiza recortando de manera rectangular como se muestra en la figura 4.1.

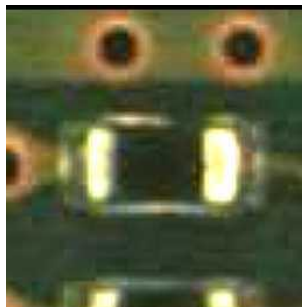


Figura 4.1: Esta imagen es el resultado de la extracción sin máscara de la parte inferior izquierda de la imagen de la figura 1.1.

La extracción de $I \times J$ de la región útil perteneciente a una imagen $M \times N$, nosotros la definimos como:

$$P_E(s, t) = P_O(k + s, l + t) \quad (4.1)$$

donde P_O es la imagen original. P_E es la nueva imagen extraída con $0 \leq s < I$ y $0 \leq t < J$. El desplazamiento del origen de la imagen completa al origen de la imagen extraída se encuentra definido por k y l con $0 \leq k < M - I$ y $0 \leq l < N - J$ (figura 4.2).

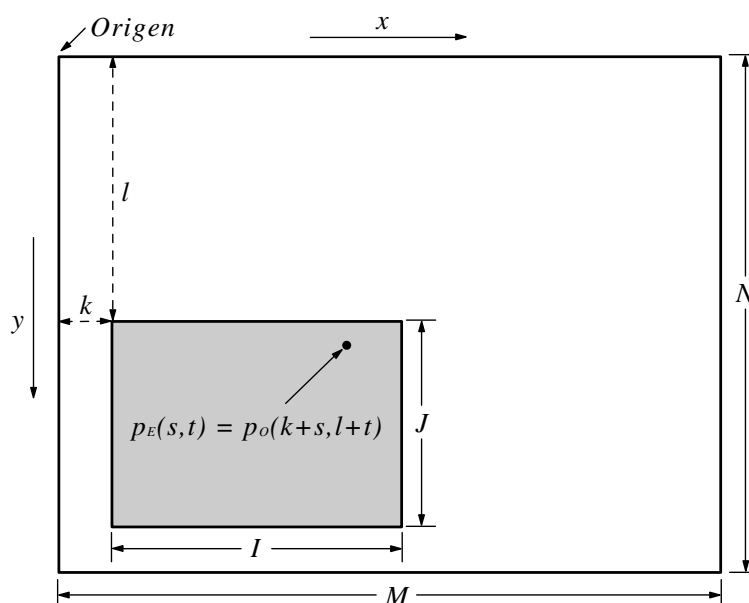


Figura 4.2: Extracción de una parte de una imagen.

4.2.2. Extracción con máscara

Se utiliza una máscara circular para acotar de manera uniforme el área que rodea al componente (es útil para capacitores y resistencias en posiciones horizontales y verticales) y de esta manera eliminar ruido que pueda ser introducido por la presencia de objetos en la imagen los cuales son irrelevantes para el análisis.

Se recorta de manera circular la región de interés como se muestra en la figura 4.3. La parte externa del círculo (acotada por la forma rectangular de la imagen) es de color negro.



Figura 4.3: Extracción de la parte inferior derecha de la imagen de la figura 1.1. Es realizada con una máscara circular.

La extracción con máscara circular de $D \times D$ de la región útil de una imagen de tamaño $M \times N$, nosotros la definimos como:

$$P_E(s, t) = \begin{cases} P_O(k + s, l + t) & \text{si } r^2 \geq (s - \frac{D}{2})^2 + (t - \frac{D}{2})^2 \\ 0 & \text{si } r^2 < (s - \frac{D}{2})^2 + (t - \frac{D}{2})^2 \end{cases} \quad (4.2)$$

donde P_O es la imagen original. P_E es la nueva imagen extraída con $0 \leq s, t < D$ y $r = D/2$. El desplazamiento del origen de la imagen completa al origen de la imagen extraída se encuentra definido por k y l con $0 \leq k < M - D$ y $0 \leq l < N - D$ (figura 4.4).

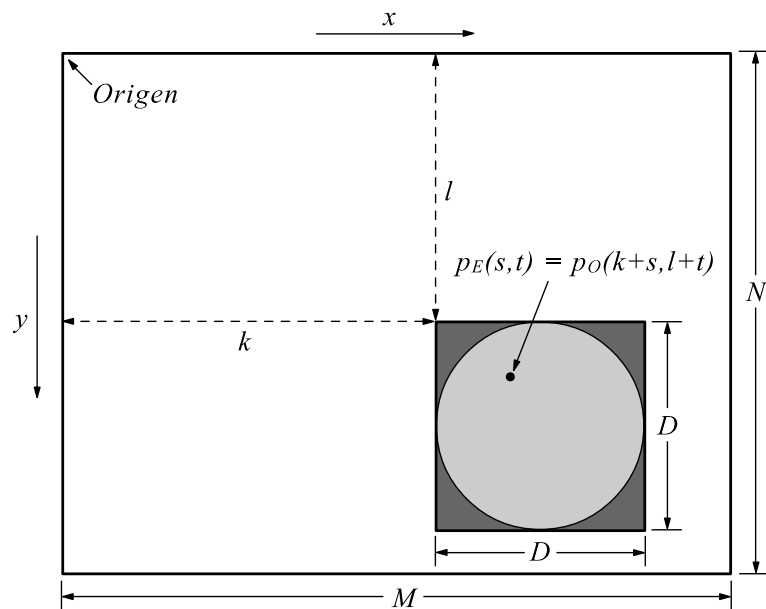


Figura 4.4: Extracción con máscara circular de una parte de una imagen

4.3. Conversión de la imagen a color a tonos de gris

En el análisis de imágenes, algunos procedimientos necesitan la representación de las imágenes en tonos de gris, en este caso se utiliza para la binarización. La conversión de color a tonos de gris transforma los tres canales RGB en un solo canal en tonos de gris preservando gran parte de la información visual.

Un modelo de color es la especificación de un sistema coordinado 3D de color y un subconjunto visible en este sistema coordinado, en el cual todos los colores se encuentran en un espectro de color particular. El modelo de color RGB es el cubo unitario (un subconjunto) del sistema coordinado Cartesiano 3D. El propósito de un modelo de color es permitir la especificación conveniente de los colores en algún espectro de color.

El modelo de color RGB (*red, green, blue*) es utilizado en los monitores CRT² e imágenes rasterizadas³ a color que emplean el sistema coordinado cartesiano. Las contribuciones individuales de cada canal son agregadas juntas para construir un color resultante. La gama⁴ del color cubierta por el modelo del RGB es definida por las cromaticidades⁵ del fosforo del CRT [17].

Para realizar la conversión del modelo de color RGB a tonos de gris se ocupa el modelo de color YIQ, el cual era usado antiguamente por el estándar de televisión NTSC (*National Television Systems Committee*). La transformación se realiza con la siguiente transformación lineal [18, 17, 19]:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.253 & -0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (4.3)$$

con $0 \leq R, G, B \leq 1$, ya que el modelo RGB está definido en el cubo unitario [17, 20, 19]. Los valores de la matriz de transformación de 3×3 son los pesos requeridos para hacer la conversión, los cuales representan la proporción de fosforo del CRT

²Tubo de rayos catódicos (*cathode ray tube*, CRT) es empleado principalmente en monitores, televisiones y osciloscopios.

³Es un archivo de datos que representa una rejilla rectangular de pixeles o puntos de color, denominada raster, que se puede visualizar en un monitor de computadora, papel u otro dispositivo que permitan su visualización.

⁴Espectro.

⁵Son los lugar geométrico de los puntos en un diagrama de cromaticidad. Un diagrama cromático o de cromaticidad es una aproximación para especificar los colores en función de x e y . Para cada valor de x e y , el correspondiente valor de z se obtiene de la ecuación $z = 1 - (x + y)$.

[17]. La componente Y es la información de luminancia, I es la fase y Q la cuadratura [18, 19].

La relación que existe entre una imagen de color y una en tonos de gris es la cantidad de luz por lo que se utiliza la componente Y para realizar la conversión de color a tonos de gris [18]. Esta relación no es igual entre cada canal del RGB, razón por la cual los valores de la matriz de transformación son distintos. En las imágenes de color y las de tonos de gris, los colores claros son los más luminosos (cercaos al blanco), mientras que los colores oscuros representan los menos luminosos (cercaos al negro) como se puede observar en la figura 4.5.

Como sólo se utiliza la componente Y no es necesario resolver toda la matriz de transformación, por lo que podemos reducir la conversión a:

$$Y = (0.299)R + (0.587)G + (0.114)B \quad (4.4)$$

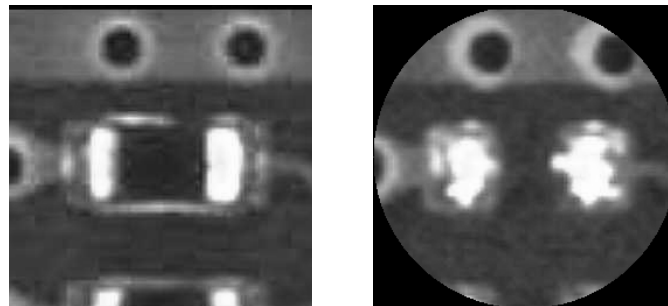


Figura 4.5: Estas son las imágenes en tonos de gris de las imágenes extraídas mostradas en la figura 4.1 y 4.3

4.4. Suavizado

El suavizado (smoothing) es un filtrado espacial de tipo pasa-bajas que es utilizado para aplicar difuminado y reducir ruido. El difuminado sirve para remover pequeños detalles considerados como imperfecciones y cubrir pequeños huecos en los trazos de una imagen.

En una imagen vista desde el dominio de Fourier, las altas frecuencias caracterizan esquinas, bordes y detalles irregulares, mientras que las bajas frecuencias representan características como el contraste y la intensidad.

Un filtro pasa-bajas (lowpass filter) elimina las frecuencias altas dejando intactas las frecuencias bajas. La aplicación de un filtro pasa-bajas a una imagen ayuda a eliminar los bordes y los trazos rígidos (considerados como ruido).

Se le llama filtrado espacial al uso de una máscara espacial en el procesamiento de imágenes.

Definimos una máscara como en la figura 4.6 donde W_i es un coeficiente. La dimensión mínima de una máscara es de 3×3 , pero pueden ser más grandes.

W_1	W_2	W_3
W_4	W_5	W_6
W_7	W_8	W_9

Figura 4.6: Máscara de 3×3

El resultado R de usar una máscara espacial de 3×3 , es la suma de los productos obtenidos entre los coeficientes de la máscara W_i y los valores de la intensidad de los pixeles bajo la máscara Z_i en una posición específica de la imagen. El valor de R es:

$$R = W_1Z_1 + W_2Z_2 + \dots + W_9Z_9 \quad (4.5)$$

R será el nuevo valor correspondiente al pixel de la imagen que se encuentra bajo el centro de la máscara.

La máscara se va moviendo a través de la imagen, de manera que cada coeficiente de la máscara empate con un pixel de la imagen como se muestra en la figura 4.7.

El filtro de suavizado que se utiliza es de tamaño 3×3 , el cual tiene un valor de 1 en todos sus coeficientes (figura 4.8). Como el resultado es la suma de los niveles de gris de los nueve pixeles de la imagen, se sobrepasa el rango de la escala de grises, por lo que se necesita dividir entre 9 (número total de coeficientes de la máscara) para regresar el valor dentro de la escala de grises [14].

En la figura 4.9 se muestra la aplicación del suavizado a imágenes en tonos de gris. La figura es un acercamiento a la región de soldadura de una imagen.

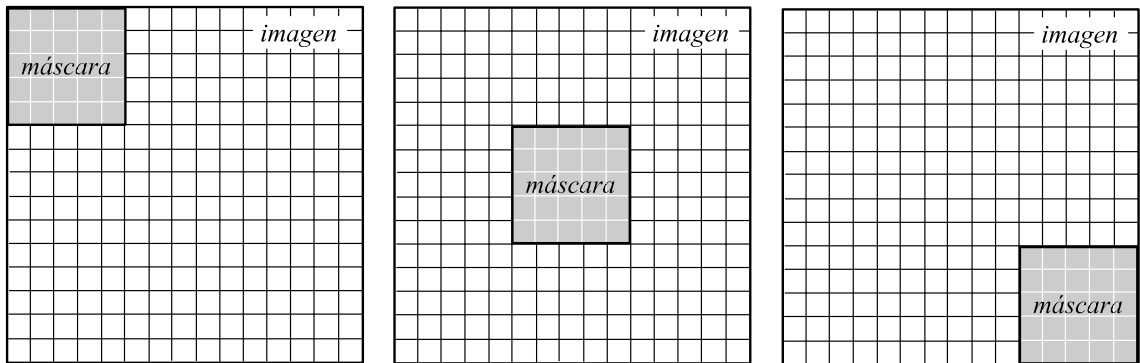


Figura 4.7: Posiciones validas de una máscara de 5×5 sobre una imagen

$$\frac{1}{9} \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Figura 4.8: Filtro espacial pasa-bajas de dimensión 3×3

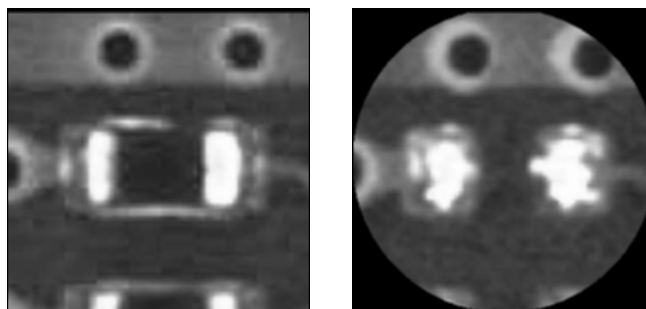


Figura 4.9: Estas imágenes son el resultado del suavizado de las imágenes de la figura 4.5

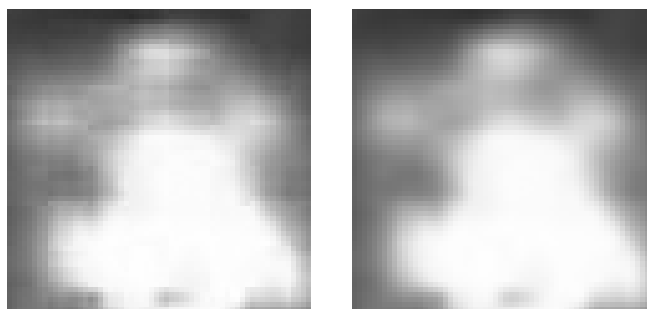


Figura 4.10: La imagen de la izquierda no ha sido suavizada, mientras que la imagen de la derecha ya ha pasado por este proceso.

4.5. Binarización

La binarización es el proceso de convertir una imagen en tonos de gris a una imagen en blanco y negro. Con una imagen binarizada se busca facilitar la búsqueda y extracción de objetos. Comúnmente los elementos de tonos claros son descritos con el color blanco, mientras que los oscuros con el color negro (figura 4.11).

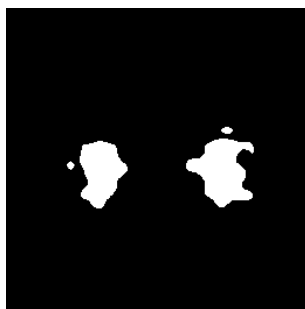


Figura 4.11: Imagen binarizada correspondiente al histograma de la figura 4.12 con un umbral $T = 200$.

La gran mayoría de las imágenes de tarjetas de soldadura de superficie tienen un histograma con una distribución de intensidades de luz (tonos de gris) parecida a la de la figura 4.12. En el histograma 4.12 predominan las intensidades de luz pequeñas (parte izquierda) las cuales representan al cuerpo de componentes electrónicos y a la tarjeta de montaje. Las soldaduras y extremos de los componentes rectangulares son representados por las intensidades altas de luz (parte derecha).

Para realizar la binarización se usa un umbral T el cual separa las intensidades de cada pixel en dos grupos (blanco y negro). Cada pixel con intensidad menor o igual a T es considerado como fondo de la imagen y se le asocia el color negro. Si la intensidad del pixel es mayor a T , es considerado como un objeto y se le asocia el

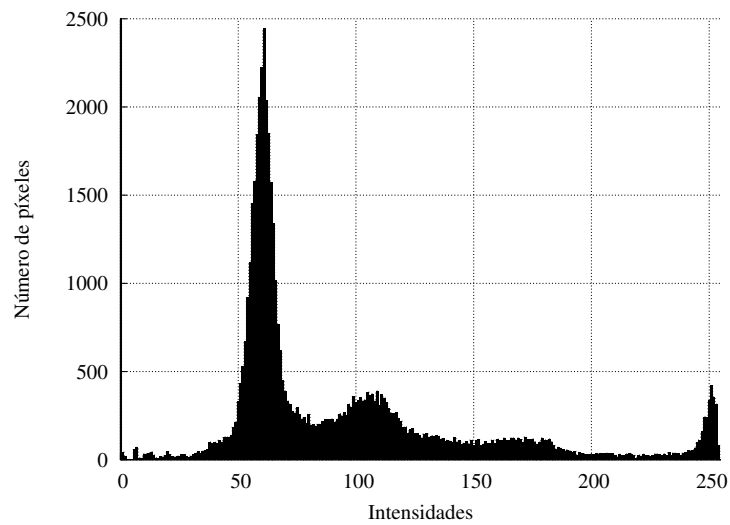


Figura 4.12: Histograma de la imagen derecha de la figura 4.9

color blanco. La binarización se define como:

$$g(x, y) = \begin{cases} 1 & \text{si } f(x, y) \geq T \\ 0 & \text{si } f(x, y) < T \end{cases} \quad (4.6)$$

con $0 \leq T \leq 255$, donde $f(x, y)$ es la imagen de entrada y $g(x, y)$ la imagen de salida [14].

Analizando el grupo de intensidades pertenecientes a las soldaduras y extremos de componentes rectangulares del histograma de la figura 4.12 y buscando de manera empírica un umbral T que proporcionara mayor información en una imagen binarizada se obtuvo que la mejor opción es $T = 200$.

4.6. Quitar ruido con morfologías

Las morfologías son una herramienta para la extracción de elementos de una imagen. Son utilizadas en la representación y descripción de la forma de una región. Las operaciones básicas e importantes que son realizadas en este trabajo son: erosión, dilatación y abrir. Las definiciones de estas operaciones son descritas a continuación [14].

4.6.1. Erosión

Ésta adelgaza los objetos, preservando su forma principal. La erosión del objeto A dada por el objeto B (con origen en el centro), está definida como:

$$A \ominus B = \{x | (B)_x \subseteq A\} \quad (4.7)$$

donde $A, B \in Z^2$ (objetos en el plano entero). $(B)_x$ es el desplazamiento de B desde su origen realizado por $x = (x_1, x_2)$.

La erosión de A dada por B es el conjunto de todas las x , tal que B es trasladada x posiciones y se encuentra contenida en A [14].

4.6.2. Dilatación

Ésta engrosa los objetos, conservando su apariencia primordial. Definimos la dilatación del objeto A por el objeto B (con origen en el centro) como:

$$A \oplus B = \{x | (B)_x \cap A \neq \emptyset\} \quad (4.8)$$

donde $A, B \in Z^2$ (objetos en el plano entero). $(B)_x$ es el desplazamiento de B desde su origen proporcionado por $x = (x_1, x_2)$.

La dilatación de A por B son todas las x que desplazan a B , tal que exista la intersección de A y B [14].

4.6.3. Abriendo

Esta operación generalmente suaviza los contornos de una imagen, descomponiendo líneas angostas y adelgazando protuberancias delgadas.

La abertura de un objeto A por un objeto B se define como:

$$A \circ B = (A \ominus B) \oplus B \quad (4.9)$$

esto es la erosión A por B , seguida de la dilatación del resultado por B [14].

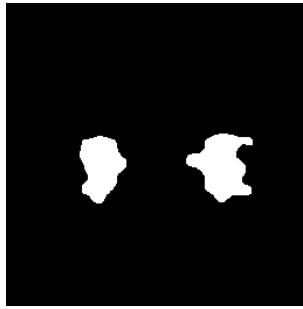


Figura 4.13: Imagen de la figura 4.11 limpiada con la operación abrir.

En la imagen de la figura 4.11 se pueden apreciar dos pequeñas formas, cada una junto a una forma grande. Estas pequeñas formas están consideradas como ruido, debido a sus dimensiones, además de que no están unidas a las formas representativas. Con el uso de la operación *abrir* estas manchas son eliminadas como se muestra en la figura 4.13.

4.7. Extracción de objetos

Al momento de binarizar una imagen esta puede contar varias formas, las cuales representan los objetos que por su intensidad de luz se encuentran de un lado del umbral de binarización (en este caso por encima)

Estos objetos pueden ser soldaduras, extremos de capacitores y resistencia, o en su defecto ruido. Para poder obtener las características de estos objetos deben de separarse de la imagen para poder facilitar su análisis.

La manera más fácil de lograr esto es separar cada objeto (concentración consecutiva o continua de pixeles blancos) en imágenes diferentes. A esto lo llamamos extracción de objetos.

La extracción de objetos en una imagen binarizada como la mostrada en la figura 4.14 (con objetos en color blanco y fondo de color negro) se obtiene realizando los siguientes pasos:

1. Se barre la imagen binarizada de izquierda a derecha y de arriba a abajo. Una vez que se localiza un pixel blanco (el más arriba a la izquierda) se considera que se encontró un objeto.
2. Se crea una imagen nueva de fondo negro en la cual sólo se pintan todos los

pixeles blancos consecutivos a otro pixel blanco, partiendo del pixel blanco localizado inicialmente.

3. De la imagen original binarizada se borra el objeto encontrado (cambiando los pixeles blancos consecutivos a color negro).
4. Se repiten los pasos del 1 al 3 hasta que ya no existen pixeles blancos en la imagen original.

De esta manera se obtienen todos los objetos por separado (figura 4.15) para su análisis posterior.



Figura 4.14: Imagen binarizada con tres objetos.

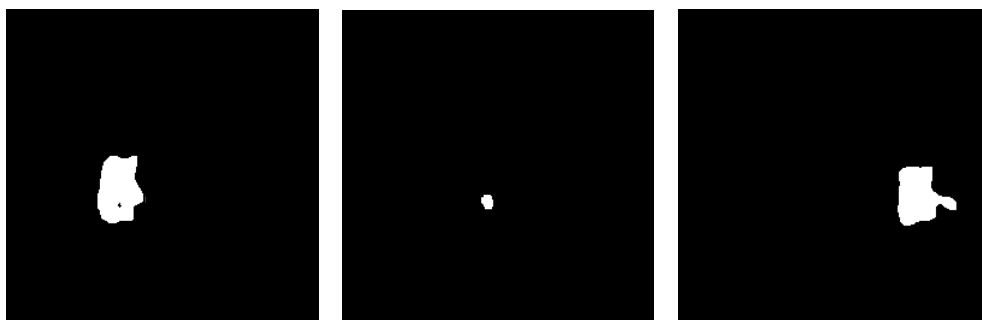


Figura 4.15: Cada una de estas imágenes pertenece a un objeto extraído de la imagen de la figura 4.14.

La función de extracción de objetos esta implementada en la biblioteca para procesamiento de imagen Scimagen [15].

4.8. Media de la suma y diferencia de histogramas de imágenes en tonos de gris

El histograma es una herramienta de descripción muy poderosa por lo que es una de las mejores técnicas para representar una imagen numéricamente. Un histograma especifica la frecuencia con la que se presentan intensidades de color en una imagen.

Para la clasificación del tipo de imagen, la red neuronal necesita como entrada datos de la imagen analizada, por lo que se utiliza histogramas de ésta. Se usa la imagen suavizada en escala de grises, la cual se particiona en $n \times n$ partes como se aprecia en la figura 4.16.

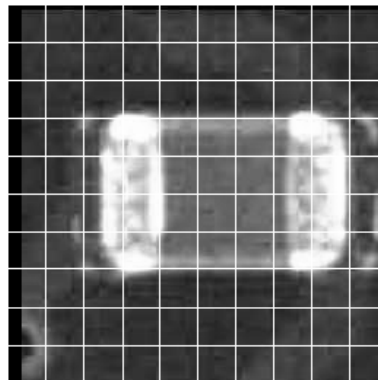


Figura 4.16: Partición de la imagen en 10×10 partes.

De cada una de estas particiones se obtiene un histograma de 256 niveles de intensidad de luz.

Es necesario reducir la dimensión del histograma usado a un solo valor representativo de este, ya que cada valor corresponde a un nodo de entrada de la red neuronal. Para lograr esto se hace uso de características (medidas) que describen con un valor numérico a una imagen basándose en su histograma.

Se probaron algunas características de la imagen, donde la que representó mejor la textura de ésta es la media de la suma y diferencia de histogramas.

4.8.1. Suma y diferencia de histogramas

Con la suma y diferencia de histogramas se establece una relación numérica entre dos puntos distintos en una imagen utilizando sus intensidades de luz, separados por una distancia d . Esta relación se describe a continuación.

La textura de imagen es definida de manera discreta como una malla rectangular de $K \times L$, donde cada elemento de ésta es $y_{k,l}$ (k y l definen la posición del elemento en la malla) con $k = 0, \dots, K - 1$ y $l = 0, \dots, L - 1$. En una imagen se define el valor de cada pixel como $G = 1, 2, \dots, N_g$, y el histograma es representado como $h = h(1), h(2), \dots, h(N_g)$.

La suma y diferencia de histogramas consiste en obtener dos histogramas $h_s = \{h_s(i)\}$ y $h_d = \{h_d(j)\}$ con $i = s_{k,l}$ y $j = d_{k,l}$. Definimos a $s_{k,l}$ y $d_{k,l}$ como:

$$s_{k,l} = y_{k,l} + y_{k+d_1, l+d_2} \quad (4.10)$$

$$d_{k,l} = y_{k,l} - y_{k+d_1, l+d_2} \quad (4.11)$$

donde d_1 y d_2 son desplazamientos. Los dominios de h_s y h_d son $G_s = 2, \dots, 2N_g$ y $G_d = -N_g + 1, \dots, N_g - 1$ respectivamente.

Los histogramas $h_s(i)$ y $h_d(j)$ normalizados están definidos respectivamente como:

$$\hat{h}_s(i) = h_s(i)/N \quad (4.12)$$

$$\hat{h}_d(j) = h_d(j)/N \quad (4.13)$$

con

$$N = \sum_i h_s(i) = \sum_j h_d(j) \quad (4.14)$$

ambos tienen la misma cantidad de puntos [21].

La suma total de puntos de los histogramas normalizados es \hat{h}_s y \hat{h}_d tienen:

$$\sum_i \hat{h}_s(i) = \sum_j \hat{h}_d(j) = 1 \quad (4.15)$$

Con este método los histogramas no solo representan la cantidad de tonos en una imagen en un dominio con el doble de tamaño, sino que también establecen una co-ocurrencia, entre puntos de la imagen. Esto proporciona una mayor cantidad de información de la textura.

4.8.2. Media de la suma y diferencia de histogramas

La media es una característica de textura. Ésta proporciona valores más distintivos (heterogéneos) para este problema en relación a otras características de una textura. La media μ está definida como:

$$\mu = \frac{1}{2} \sum_i i \cdot \hat{h}_s(i) \quad (4.16)$$

donde i es un elemento del histograma, $\hat{h}_s(i)$ es el histograma normalizado resultante de la suma de los histogramas. Con $0 \leq \mu < 256$ [21].

4.8.3. Otras características de los histogramas

La principal causa de que estas características del histograma no sean útiles para el tipo de imágenes analizadas, se debe a que el intervalo de los valores de estas es difícil de definir. Los valores son grandes y con poca variación entre ellos, lo cual no asegura que sean constantes para todas las muestras, y en algún momento estos valores sean pequeños o mucho más grandes y con gran variación.

Las características probadas sin mucho éxito son las siguientes:

- Energía E de la suma y diferencia de histogramas

$$E = \sum_i \hat{h}_s(i) + \sum_j \hat{h}_d(j) \quad (4.17)$$

- Varianza σ^2 de la suma y diferencia de histogramas

$$\sigma^2 = \frac{1}{2} \left[\sum_i (i - 2\mu)^2 \cdot \hat{h}_s(i) + \sum_j j^2 \cdot \hat{h}_d(j) \right] \quad (4.18)$$

- Correlación r de la suma y diferencia de histogramas

$$r = \frac{1}{2} \left[\sum_i (i - 2\mu)^2 \cdot \hat{h}_s(i) - \sum_j j^2 \cdot \hat{h}_d(j) \right] \quad (4.19)$$

En las ecuaciones (4.17), (4.18) y (4.19), la variable i es un elemento del histograma, $\hat{h}_s(i)$ es el histograma normalizado resultante de la suma de los histogramas y $\hat{h}_d(i)$ es el histograma normalizado resultante de la diferencia de los histogramas. Estas características fueron obtenidas de [21], donde se pueden revisar otras más.

Sin embargo el hecho de que estas medidas no sean útiles de manera primordial para representar una imagen en nuestra aplicación, no significa que no puedan ser útiles como complemento para mejorar la precisión de la clasificación.

Capítulo 5

Clasificación de la imagen

Cada imagen dependiendo de a qué tipo de componente hace referencia, necesita un procesamiento distinto, por lo que es necesario identificar qué componente se va a analizar, para ello hay que realizar una clasificación. Cuando se está trabajando con una gran cantidad de imágenes es necesario automatizar este proceso de clasificación, razón por la cual se implementó una red neuronal.

Debido a la variedad de imágenes causada por diferentes condiciones como: la clase de componentes, la diversidad de características en la tarjeta, detalles e imprecisiones; es necesario ocupar información común en todas estas y que sea descriptiva de manera individual.

Se utilizan los histogramas porque proporcionan información referente a la intensidad de color de toda la imagen. Existe la gran posibilidad de que dos imágenes muy diferentes proporcionen como resultado un histograma muy similar y también puede ser que dos imágenes muy similares produzcan un histograma muy diferente.

Para disminuir este problema, se secciona la imagen en $m \times n$ partes y se obtiene el histograma de cada una de estas. Entre más grande sea el número de partes en las que se divide una imagen mayor será el número de valores representativos.

Esto da lugar al problema de obtener una gran cantidad de información la cual requiere muchos recursos para ser almacenada y procesada. En una imagen de color seccionada $m \times n$ partes, cada parte tiene un histograma de 256 niveles por canal de color (3 canales). En total se obtendría de una imagen $m \times n \times 256 \times 3$ datos, lo cual es demasiada información para ser procesada en un tiempo corto. Una red neuronal necesitaría $m \times n \times 256 \times 3$ nodos de entrada para poder procesar la información de la imagen.

La solución a este exceso de información, es ocupar imágenes en tonos de gris, las cuales solo proporcionan las intensidades de luz en un solo canal producidas por los canales RGB de las imágenes a color, reduciendo a un 33.34% la cantidad de datos.

La imagen se secciona $m \times n$ partes y de cada una de estas se obtiene la media de la suma y diferencia de histogramas. La media de la suma y diferencia de histogramas es un solo valor que describe al histograma de manera significativa. De esta manera se obtienen $m \times n$ datos de la imagen original, los cuales ya son más fáciles de procesar. La obtención de la media de la suma y diferencia de histogramas es explicada en la sección 4.8, página 45.

Como entrada, la red neuronal recibe las medias de la suma y diferencia de histogramas de una parte de una imagen seccionada. Cada una de estas medias es el valor de cada uno de los nodos de entrada de la red neuronal.

La red neuronal clasifica las imágenes a través de la distribución de sus intensidades de luz, por lo que no identifica formas.

Las imágenes con características visuales similares bajo el criterio de la existencia de un componente, fueron agrupadas a mano en clases que las describen. Estas clases son utilizadas para indicarle a la red neuronal en el entrenamiento, a que categoría pertenece una imagen con determinadas características.

La explicación de cómo entrenar y usar la red neuronal está en el capítulo 8, en el que se menciona la implementación de la misma.

5.1. Las redes neuronales: definición

Las redes neuronales (*artificial neural network*, ANN) son una herramienta computacional de modelado que ha emergido y encontrado una extensa aceptación en muchas disciplinas para modelar problemas complejos del mundo real. Las redes neuronales han sido definidas como estructuras que comprenden elementos adaptativos de procesamiento simple interconectados densamente (llamados neuronas artificiales o nodos) que son capaces de realizar el computo paralelo masivo de procesamiento de datos y conocimiento.

Las ANN son abstracciones radicales de las redes neuronales biológicas. La idea de una ANN, no es replicar las operaciones de los sistemas biológicos, pero sí hacer uso de lo que se conoce acerca de la funcionalidad de las redes biológicas para resolver problemas complejos.

El atractivo de las ANN viene de las notables características de procesamiento de información de los sistemas biológicos como lo son la no linealidad, alto paralelismo, robustez, tolerancia a defectos y fracasos, aprendizaje, la habilidad para manipular información confusa e imprecisa y la capacidad de generalizar.

Existe un gran número de ANNs, nuevas y modificaciones de las ya existentes que están en constante desarrollo. Algunas aplicaciones pueden ser resueltas por diferentes tipos ANN, sin embargo otras solo pueden ser resueltas con ANN específicas. El tipo de ANN más utilizados son: *Hopfield*, *adaptive resonance theory* (ART), Kohonen, Backpropagation, recurrent, *counterpropagation*, *radial basis function* (RBF), entre otras.

Las ANN *backpropagation* (redes neuronales de propagación hacia atrás) son las más utilizadas de todas las redes y son consideradas las más eficientes para casi cualquier tipo de problema. Estas son redes multicapa (multi-layer perceptron, MLP).

Las ANN frecuentemente superan a otras herramientas computacionales en la solución de una variedad de problemas catalogados como: clasificación de patrones (*pattern classification*), agrupamiento (*clustering*), aproximación de funciones (*function approximation*), pronóstico (*forecasting*), optimización (*optimization*), asociación (*association*) y control (*control*) [22].

5.2. Elementos de una red neuronal

Los siguientes elementos son comunes en todos los modelos de redes multicapa [22, 23]:

- **Señal de entrada:** Es la señal que recibe una neurona enviada a través de su conexión con otra neurona de la capa inferior.
- **Señal de la red:** Es la señal resultante de todas las señales de entrada recibidas por el perceptrón. Es denotada por ξ y también llamada *efecto de la red*.
- **Peso:** Es el valor asociado a la conexión entre nodos de distintas capas. Es denotado por w .
- **Conexiones:** Es la relación que existe a través un peso w , entre un nodo i de la capa intermedia l y un nodo j de la capa intermedia $l - 1$ (figura 5.2).
- **Corrección de pesos:** Definido como Δw es el cambio incremental (+/-) aplicado a los pesos para mejorar la precisión de la red.
- **Bias:** Es el umbral de activación. Cuando el efecto de la red sobrepasa el umbral la neurona, esta se activa. El bias se considera como un nodo con valor de activación 1.0 y un peso w igual al valor del umbral que representa (figura 5.2).
- **Grado de aprendizaje:** Definido como η , controla la proporción de la corrección del peso.

- **Coefficiente de momento:** Definido como μ , ayuda a orientar la búsqueda del óptimo y a escapar de mínimos locales.
- **Funcion de transferencia:** Determina la activación x de la red a través de la conversión de la señal de la red ξ . Existen varias funciones de activación, sin embargo en las redes de propagación hacia atrás, la más común y que mejor se adapta a casi cualquier tipo de problemas es la *sigmoidal* σ , la cual está definida en la ecuación (5.2).
- **Neurona:** Es cada nodo de la red neuronal, en la cual se procesa y guarda la señal de la red (figura 5.2).
- **Perceptrón:** Es un sistema que consiste en una neurona principal interconectada con neuronas de la capa previa, de las cuales recibe el valor de activación x_j , donde cada una de estas interconexiones tiene un pesos w asignado. El valor de activación y de la neurona principal es obtenido a través de la función de transferencia del efecto de la red (figura 5.1).
- **Capa:** Es la agrupación de neuronas que se encuentran a un mismo nivel (no existe interconexión entre ellas) como se ve en la figura 5.2.
- **Capa intermedia:** Es el conjunto de componentes que comprende las neuronas de la capa actual y las interconexiones con la capa intermedia anterior. Una red neuronal multicapa contiene L capas intermedias. Para la capa intermedia $l = 1, \dots, L$ existen N_l nodos y $N_l \times N_{l-1}$ conexiones de enlace (N_{l-1} son el número de nodos de la capa intermedia anterior) con un peso w por cada una. N_l y N_{l-1} son el número de nodos incluyendo los bias (figura 5.2).
- **Época:** Es el número de iteraciones que se entrena la red (ciclos de entrenamiento).
- **Modo de entrenamiento:** Es la manera en la que se actualizan los pesos de la red. Existen dos modos de entrenamiento: *off-line* (*batch*) y *on-line*.
 - *off-line*

La corrección de los incrementos de los pesos Δw se realiza por cada patrón. Una vez realizadas las Δw de todos los patrones, se realiza la actualización de cada peso w .
 - *on-line*

El cálculo de valores de la red hacia adelante, y la corrección de los incrementos de los pesos Δw y la actualización de los pesos w en la propagación hacia atrás, se realizan n cantidad de veces o hasta que el error converja, para cada uno de los patrones de entrenamiento.

Cada uno de estos modos se realiza por cada época definida para el entrenamiento.

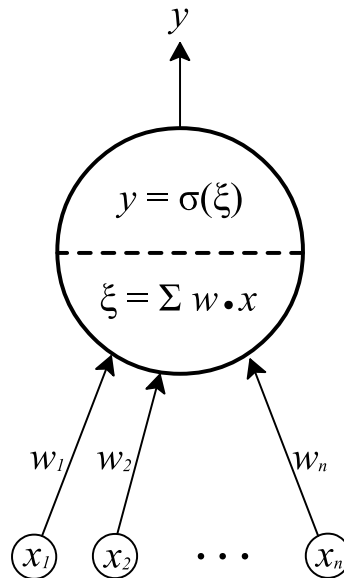


Figura 5.1: Perceptrón

5.3. Arquitectura de la red neuronal

La red neuronal utilizada es la de propagación hacia atrás (Backpropagation), la cual es considerada como una red muy versátil y poderosa. Se basa en la búsqueda de una superficie de error (ésta como una función de pesos de la red) usando el gradiente descendente. Se utilizan para el modelado, clasificación, pronóstico, control, datos y compresión de imágenes y reconocimiento de patrones. Una red de propagación hacia atrás es un Multi-Capa de perceptrones, la cual consiste en:

- Una capa de entrada con nodos que representan variables de entrada.
- Una capa de salida con nodos que representan las variables dependientes.
- Una o más capas ocultas que contienen nodos que ayudan a obtener la no-linealidad de los datos.

El termino **propagación hacia atrás** se refiere a la forma en la que es computado el error obtenido en la salida de la red neuronal. Éste se propaga de la capa de salida hacia la capa(s) oculta(s) y finalmente hacia la capa de entrada. Estas redes pueden

tener sus nodos completa o parcialmente interconectados [22].

Para las redes neuronales, no existen valores exactos para los parámetros de esta, solo hay rangos y recomendaciones dependiendo del problema que se requiera resolver. Los valores de los parámetros son buscados de manera empírica en los intervalos indicados en la literatura [22]. El modo de entrenamiento on-line proporcionó mejores resultados en la implementación que el modo off-line ¹.

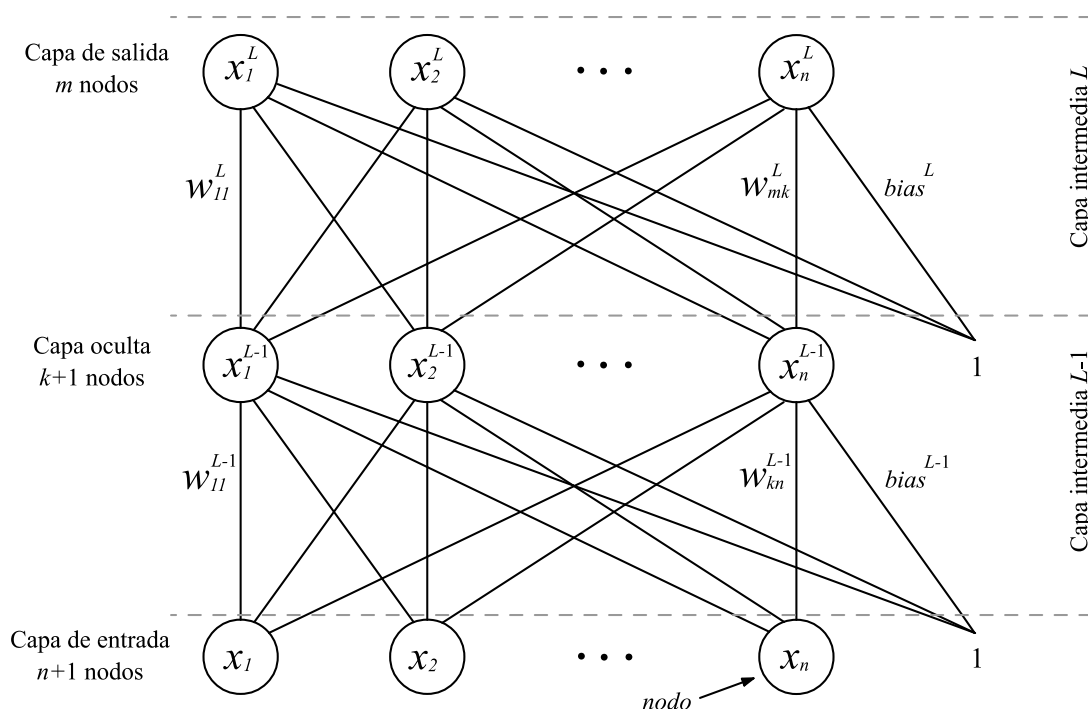


Figura 5.2: Esquema de la red neuronal

¹Los métodos de entrenamiento on-line y offline son mencionados en la sección 5.2.

5.4. Algoritmo de la red neuronal de propagación hacia atrás con método de entrenamiento on-line

La Figura 5.2 muestra la estructura de la red neuronal. Cada iteración del algoritmo de propagación hacia atrás constituye dos pasos: la activación hacia adelante en cada capa (*feedforward*) y la propagación hacia atrás del error computado para ajustar los pesos de los nodos (*backpropagation*). El modo de entrenamiento de la red es on-line² [22, 23].

Los bias³ de las neuronas de una capa se establecen como un nodo extra en esta última (excepto la capa de salida), con el valor de activación igual 1.0. Al ser un nodo más en la capa, se le trata de la misma manera que cualquier otra neurona [22].

Los valores de entrada y salida de la red neuronal están normalizados en el intervalo [0.0, 1.0].

El algoritmo es el siguiente [22, 23]:

1. Se generan valores aleatorios para cada peso w (generalmente entre 0.0 y 1.0).
2. El efecto de la red ξ_j^l del nodo j de la capa intermedia actual l , con $l = \{1 \dots L\}$, es obtenido a través de la suma de los productos punto de los valores de activación x_i^{l-1} de los nodos i de la capa intermedia anterior $l - 1$, con sus pesos de interconexión w_{ji}^l correspondientes a la capa intermedia actual l . Esto se define como:

$$\xi_j^l = \sum_{i=1}^{N_{l-1}} w_{ji}^l x_i^{l-1} \quad (5.1)$$

con $-\infty < \xi_j^l < \infty$, N_{l-1} es el número de nodos en la capa anterior $l - 1$.

En el caso de que la capa intermedia anterior $l - 1$ corresponda a la capa de entrada de la red neuronal, los valores x_i^l son los correspondientes a los datos de entrada.

La activación x_j^l de la neurona j es determinada usando la *sigmoide* σ como función de transferencia, definida como:

²Véase modos de entrenamiento en la sección 5.2.

³Umbrales, véase en la sección 5.2.

$$x_j^l = \sigma(\xi_j^l) = \frac{1}{1 + \exp^{-\xi_j^l}} \quad (5.2)$$

con $0.0 \leq \sigma(\xi_j^l) \leq 1.0$. Este valor de activación representa la nueva señal que será transferida a las capas siguientes.

Este procedimiento se realiza por cada capa intermedia, partiendo de la capa de entrada hacia la capa de salida (pasando por las n capas ocultas).

3. Una vez calculados los valores de activación x_j^l de cada neurona, se calcula la corrección de los pesos Δw_{ji}^l en orden inverso al flujo de la red (partiendo de la capa de salida en dirección a la capa de entrada, pasando por las capas ocultas) como sigue:

$$\Delta w_{ji}^l = -\eta \delta_j^l x_i^{l-1} - \mu \Delta w_{ji}^{l(\text{previo})} \quad (5.3)$$

donde η es el grado de aprendizaje, x_i^{l-1} es el valor de activación de la neurona i de la capa anterior $l - 1$, μ el coeficiente de momento y $\Delta w_{ji}^{l(\text{previo})}$ es la corrección de peso previo.

En la capa intermedia $l = L$ (capa de salida), se define el δ_j^L de cada neurona como:

$$\delta_j^L = (x_j^L - y_j) x_j^L (1 - x_j^L) \quad (5.4)$$

donde x_j^L es el valor de activación y y_j es el valor ideal (en la mayoría de los casos es 1.0, aunque depende de los resultados que se quieran obtener). x_j^L y y_j pertenecen a la neurona j de la capa de salida.

Si la capa intermedia es $l < L$ (capas ocultas), el δ_j^l de cada neurona se define como:

$$\delta_j^l = x_j^l (1 - x_j^l) \left(\sum_{k=1}^{N_{l+1}} \delta_k^{l+1} w_{kj}^{l+1} \right) \quad (5.5)$$

donde x_j^l es el valor de activación de la neurona j de la capa oculta l , δ_k^{l+1} es la corrección de pesos obtenido de la capa siguiente $l + 1$ y w_{kj}^{l+1} es el peso correspondiente a la interconexión con la neurona k . N_{l+1} es el número de neuronas de la capa siguiente.

4. Se actualizan los pesos w con los valores de las correcciones de pesos Δw como:

$$w_{ji}^l(t) = w_{ji}^l(t-1) + \Delta w_{ji}^l(t) \quad (5.6)$$

donde w_{ji}^l es el peso de la interconexión de la neurona j la capa intermedia actual l con la neurona i de la capa intermedia anterior $l-1$, t es la época⁴ o iteración del entrenamiento y Δw_{ji}^l es la corrección correspondiente al peso w_{ji}^l .

5. Se ejecuta n veces del paso 2 al paso 4 con el mismo patrón. Con esto se obtiene una corrección de error más precisa.
6. Se realiza una época del aprendizaje, la cual consiste en realizar el paso 5 por cada patrón del conjunto de entrenamiento.
7. Se realiza la suma de los cuadrados del error ρ , como criterio de convergencia del error. Éste consiste en obtener por cada época, la media de los cuadrados del error adquiridos de cada patrón de entrenamiento. Se defina a ρ como:

$$\rho = \frac{1}{N} \sum_{p=1}^N \sum_{i=1}^M (t_{pi} - O_{pi})^2 \quad (5.7)$$

donde N es el número de patrones de entrenamiento, M el número de nodos de salida, t_{pi} y O_{pi} son el valor objetivo y el valor de salida resultante respectivamente pertenecientes al nodo i del patrón p . Si $\rho \leq \varepsilon$ entonces finaliza el algoritmo, ε debe de ser un error pequeño.

8. Los pasos 6 y 7 son realizan tantas veces como épocas están definidas.

⁴Es el número de iteraciones que se entrena la red (ciclos de entrenamiento).

Capítulo 6

Extracción de características

Las características de una imagen son los rasgos que hacen la distinción de ésta entre otras imágenes, como lo son: tamaños, texturas, colores y formas.

Se utilizan tres tipos de características para realizar la identificación de capacitores y resistencias en la imagen. Estas características son los momentos invariantes de las formas de los objetos, los histogramas de brillo y contraste de la textura a color, y la cantidad de objetos en la imagen.

Se probó la distancia euclidiana entre los centroides de los objetos, y también los perímetros de estos. Estas características no proporcionaron información significativa para el proceso de identificación.

Para la obtención de los momentos invariantes y la contabilización de objetos, es necesario realizar la extracción de los objetos que aparecen en la imagen binarizada.

6.1. Momentos invariantes

Los momentos son cuantificadores de la distribución de muestras en un espacio (en este caso puntos en una imagen). Son utilizados para describir las formas de los objetos en una imagen.

Se les denomina invariantes porque no son afectados por la rotación, traslación y escalamiento de las formas en la imagen. Para definir los momentos invariantes, hay que determinar los momentos centrales y los momentos centrales normalizados.

Dado que los extremos de los capacitores y las resistencias tienen cierta distribución de puntos similar entre ellos, y una distribución distinta cuando se trata de soldadura, los momentos invariantes se presentan como una buena opción para cuantificar estas distribuciones y poder realizar una distinción entre los componentes.

La obtención de los momentos invariantes de las formas de los objetos en una imagen, se realiza a partir de la imagen binarizada, puesto que ésta representa un objeto con un solo valor (blanco).

En un caso ideal, una imagen con soldadura o con extremos de capacitores y resistencias, sólo debe de tener dos objetos, por lo que los momentos invariantes son calculados dos veces.

De manera empírica se determinó que son útiles solo los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 . Estos fueron los que más información útil aportaron para caracterizar un objeto en una imagen binarizada.

6.1.1. Definición de momentos

Para una función continua de distribución de densidades en dos dimensiones $\rho(x, y)$, los momentos de orden $(p + q)$ están definidos en términos de las integrales de Reimann, como [14, 24]:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q \rho(x, y) dx dy \quad (6.1)$$

para $p, q = 0, 1, 2, \dots$

Si se asume que $\rho(x, y)$ es continua por trozos (por lo tanto limitada) y tiene valores no cero sólo en una parte del plano finito xy , existen momentos de todos los órdenes. Por el teorema de la unicidad tenemos que la secuencia de momentos (m_{pq}) es únicamente determinada por $\rho(x, y)$, e inversamente, (m_{pq}) determina únicamente a $\rho(x, y)$ [14, 24].

Si determinamos a una imagen como una función discreta en dos dimensiones $f(x, y)$ con valores no cero, se definen los momentos de ésta en forma discreta como [25]:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad (6.2)$$

donde $f(x, y)$ es el valor de cada pixel de la imagen. En el caso de una imagen binarizada los valores de $f(x, y)$ son:

$$f(x, y) = \begin{cases} 1 & \text{si el pixel es blanco} \\ 0 & \text{si el pixel es negro} \end{cases} \quad (6.3)$$

6.1.2. Definición de momentos centrales

Los momentos centrales cuantifican la distribución de puntos a partir del centro de masa de la forma de un objeto. Estos son expresados en su forma discreta como [25]:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (6.4)$$

\bar{x} y \bar{y} son desplazamientos del origen hacia el centro de masa en x y y respectivamente, los cuales son definidos como [24, 25]:

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad (6.5)$$

$$\bar{y} = \frac{m_{01}}{m_{00}} \quad (6.6)$$

donde m_{10} es la longitud máxima en el eje x y m_{01} es la longitud máxima en el eje y del objeto, m_{00} es la masa.

Los momentos centrales son invariantes a la traslación, debido a que su referencia son ellos mismos, no importa si cambian de lugar [24].

Es menos costoso calcular los momentos centrales a partir de los momentos que evaluar la ecuación (6.4) directamente. Los momentos centrales de hasta orden 3 son los siguientes [14]:

$$\begin{aligned} \mu_{00} &= m_{00} \\ \mu_{10} &= m_{10} - \frac{m_{10}}{m_{00}}(m_{00}) = 0 \\ \mu_{01} &= m_{01} - \frac{m_{01}}{m_{00}}(m_{00}) = 0 \\ \mu_{11} &= m_{11} - \frac{m_{10}m_{01}}{m_{00}} \end{aligned}$$

$$\begin{aligned}\mu_{20} &= m_{20} - \frac{2m_{10}^2}{m_{00}} + \frac{m_{10}^2}{m_{00}} = m_{20} + \frac{m_{10}^2}{m_{00}} \\ \mu_{02} &= m_{02} - \frac{2m_{01}^2}{m_{00}} + \frac{m_{01}^2}{m_{00}} = m_{02} + \frac{m_{01}^2}{m_{00}} \\ \mu_{12} &= m_{12} - 2\bar{y}m_{11} - \bar{x}m_{02} + 2\bar{y}^2m_{10} \\ \mu_{21} &= m_{21} - 2\bar{x}m_{11} - \bar{y}m_{20} + 2\bar{x}^2m_{01} \\ \mu_{30} &= m_{30} - 3\bar{x}m_{20} + 2\bar{x}^2m_{10} \\ \mu_{03} &= m_{03} - 3\bar{y}m_{02} + 2\bar{y}^2m_{01}\end{aligned}$$

Los momentos centrales de hasta orden 3 son utilizados para calcular los momentos centrales normalizados.

6.1.3. Definición de momentos centrales normalizados

Dado que μ_{00} es la masa del objeto, podemos normalizar los momentos centrales para tener una descripción independiente del tamaño (invariantes al escalamiento). Los momentos centrales normalizados η_{pq} , son definidos como [14]:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \quad (6.7)$$

donde:

$$\gamma = \frac{p+q}{2} + 1 \quad (6.8)$$

para $p+q = 2, 3, \dots$

Como se comento anteriormente, es mucho menos costoso calcular los momentos centrales normalizados a partir de los momentos centrales, que evaluar la ecuación (6.4) directamente. Los momentos centrales normalizados de hasta orden 3 son [14]:

$$\begin{aligned}\eta_{00} &= \frac{\mu_{00}}{\mu_{00}} = \frac{m_{00}}{m_{00}} \\ \eta_{10} &= \frac{\mu_{10}}{\mu_{00}^{\frac{3}{2}}} = \frac{m_{10} - \frac{m_{10}}{m_{00}}(m_{00})}{m_{00}^{\frac{3}{2}}} \\ \eta_{01} &= \frac{\mu_{01}}{\mu_{00}^{\frac{3}{2}}} = \frac{m_{01} - \frac{m_{01}}{m_{00}}(m_{00})}{m_{00}^{\frac{3}{2}}}\end{aligned}$$

$$\begin{aligned}
 \eta_{11} &= \frac{\mu_{11}}{\mu_{00}^2} = \frac{m_{11} - \frac{m_{10}m_{01}}{m_{00}}}{m_{00}^2} \\
 \eta_{20} &= \frac{\mu_{20}}{\mu_{00}^2} = \frac{m_{20} - \frac{2m_{10}^2}{m_{00}} + \frac{m_{10}^2}{m_{00}}}{m_{00}^2} = m_{20} + \frac{m_{10}^2}{m_{00}} \\
 \eta_{02} &= \frac{\mu_{02}}{\mu_{00}^2} = \frac{m_{02} - \frac{2m_{01}^2}{m_{00}} + \frac{m_{01}^2}{m_{00}}}{m_{00}^2} = m_{02} + \frac{m_{01}^2}{m_{00}} \\
 \eta_{12} &= \frac{\mu_{12}}{\mu_{00}^{\frac{5}{2}}} = \frac{m_{12} - 2\bar{y}m_{11} - \bar{x}m_{02} + 2\bar{y}^2m_{10}}{m_{00}^{\frac{5}{2}}} \\
 \eta_{21} &= \frac{\mu_{21}}{\mu_{00}^{\frac{5}{2}}} = \frac{m_{21} - 2\bar{x}m_{11} - \bar{y}m_{20} + 2\bar{x}^2m_{01}}{m_{00}^{\frac{5}{2}}} \\
 \eta_{30} &= \frac{\mu_{30}}{\mu_{00}^{\frac{5}{2}}} = \frac{m_{30} - 3\bar{x}m_{20} + 2\bar{x}^2m_{10}}{m_{00}^{\frac{5}{2}}} \\
 \eta_{03} &= \frac{\mu_{03}}{\mu_{00}^{\frac{5}{2}}} = \frac{m_{03} - 3\bar{y}m_{02} + 2\bar{y}^2m_{01}}{m_{00}^{\frac{5}{2}}}
 \end{aligned}$$

Los momentos centrales normalizados de hasta orden 3 son utilizados para conseguir los momentos invariantes.

6.1.4. Definición de momentos invariantes

Estos son derivados de los momentos centrales normalizados de segundo y tercer orden. Son denotados por ϕ y están definidos como [14]:

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

$$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

El momento invariante ϕ_7 es utilizado para distinguir objetos de su reflejo [24].

6.1.5. Objetos sobrantes, uso de una línea media y cálculo de los momentos invariantes

En algunas imágenes binarizadas correspondientes a capacitores y resistencias, se presentan objetos que no pertenecen a sus extremos, o a soldaduras (como se muestra en la imagen superior izquierda de la figura 4.14). Estos pueden ser elementos como: componentes que no son capacitores o resistencias, soldaduras de otros componentes distintos al buscado o marcas en la tarjeta como números, letras o zonas de ensamble.

Los momentos invariantes de estos objetos sobrantes, en su gran mayoría, no corresponden a los requeridos para realizar la identificación de la existencia del componente. Por otro lado, algunos objetos sobrantes tienen momentos invariantes similares a los buscados para el análisis, pero su posición y cantidad en la imagen no es la correcta, lo que ocasiona una mala identificación (figura 6.1). Por las razones anteriores, es necesario no considerar los objetos sobrantes.



Figura 6.1: En las imágenes de la izquierda y derecha se muestran objetos con momentos invariantes similares soldaduras de capacitores y resistencias. En la imagen central se describen objetos con momentos invariantes pertenecientes a extremos de capacitores y resistencias.

Para descartar a los objetos sobrantes, sólo se toman en cuenta los objetos cuya masa se intersecta en algún punto con una línea media en posición horizontal o vertical, la cual depende la de orientación del componente analizado (figura 6.2).

La línea media debe pasá por el centro de la imagen debido a que el componente analizado se encuentra en el centro de ésta. Si la imagen pertenece a un dispositivo en posición vertical, la línea lógica será vertical. De manera similar, si el componente está orientado horizontalmente, la línea lógica será horizontal. La intención es que la línea lógica pase por los dos objetos pertenecientes a las manchas de soldadura o a



Figura 6.2: Estas imágenes muestran su correspondiente línea lógica (color gris) y la intersección con los objetos centrales con respecto a la orientación del componente buscado.

los extremos de capacitores y resistencias, excluyendo los objetos sobrantes.

Definimos a un objeto O con coordenadas iniciales (O_{x_i}, O_{y_i}) y con coordenadas finales (O_{x_f}, O_{y_f}) , y definamos a la línea media como L con coordenadas centrales (L_{x_c}, L_{y_c}) . La intersección con una línea media vertical se realiza si $O_{x_i} \leq L_{x_c} \leq O_{x_f}$. Si la línea media es horizontal, entonces la intersección se hace con $O_{y_i} \leq L_{y_c} \leq O_{y_f}$.

Se calculan los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 de los primeros dos objetos extraídos que su masa interseca a la línea media correspondiente a su orientación.

6.2. Contabilización de objetos

La soldadura en algunas ocasiones presenta cierta heterogeneidad en su composición¹ o en su presentación, cosa que no sucede con los extremos de capacitores y resistencias, ya que éstos son homogéneos visualmente, como se puede ver apreciar en parte superior de la figura 6.3. Estas características son más fáciles de identificar en las imágenes binarizadas como se muestra en la parte inferior de la figura 6.3.

La heterogeneidad se puede presentar por varias situaciones, la más común es que no se le ha montado un componente y no ha pasado por el horneado para fundir la soldadura o por un mal deposito de pasta de soldadura, durante el proceso de colocación de soldadura y el componente no alcanza a cubrir con alguno de sus extremos. Empíricamente se ha distinguido que muchas de las tarjetas sin componente, presentan manchas heterogéneas.

Después del proceso de binarización y de limpieza con morfologías a veces subsisten marcas del esparcimiento de la soldadura por lo que se plantea que si existe

¹Véase la sección 2.7

componente este esparcimiento no sería visible.

Las características que se analizan para el conteo de objetos son las siguiente:

- Si en una imagen binarizada se encuentran más de dos objetos, se dice que se trata de una soldadura, debido a su heterogeneidad.
- Si existe solo soldadura en la imagen, entonces no puede estar un capacitor o resistencia.

Dicho lo anterior, se realiza la contabilización de objetos que aparecen en la imagen. Si se presentan más de dos objetos, se considera que se trata de soldaduras, por lo que no existe componente. En el caso en el que existen dos elementos, queda la incertidumbre acerca de la existencia de un capacitor o resistencia debido a que puede tratarse de dos soldaduras o los dos extremos del un componente. En el caso de que aparezcan menos de dos elementos, también se considera que no existe componente, debido a que puede existir una sola soldadura, más no un solo extremo de capacitor o resistencia.

El problema que se presenta, al igual que en otros métodos, es el ruido en la imagen (objetos no deseados) los cuales elevan la probabilidad de realizar una identificación errónea. Para reducir esta probabilidad de error se hace uso de regiones factibles.

6.2.1. Región factible

Para disminuir este ruido se plantean dos regiones factibles, cada una correspondiente a un extremo de un capacitor o resistencia (figura 6.5), o en otro caso, a una soldadura (figura 6.4). Si el centro de masa de un objeto se encuentra dentro de una zona factible, se considera éste para el conteo de elementos en la imagen.

De esta manera limitamos el área donde se pueden encontrar objetos pertenecientes a soldaduras o a extremos de capacitores y resistencias.

Definamos a la región factible como F con coordenadas iniciales (F_{x_i}, F_{y_i}) y coordenadas finales (F_{x_f}, F_{y_f}) , y a un objeto O con centro de masa en las coordenadas (O_{x_c}, O_{y_c}) . Para determinar si un objeto esta dentro de una región factible, se tiene que cumplir que:

$$F_{x_i} \leq O_{x_c} \leq F_{x_f} \tag{6.9}$$

$$F_{y_i} \leq O_{y_c} \leq F_{y_f} \tag{6.10}$$

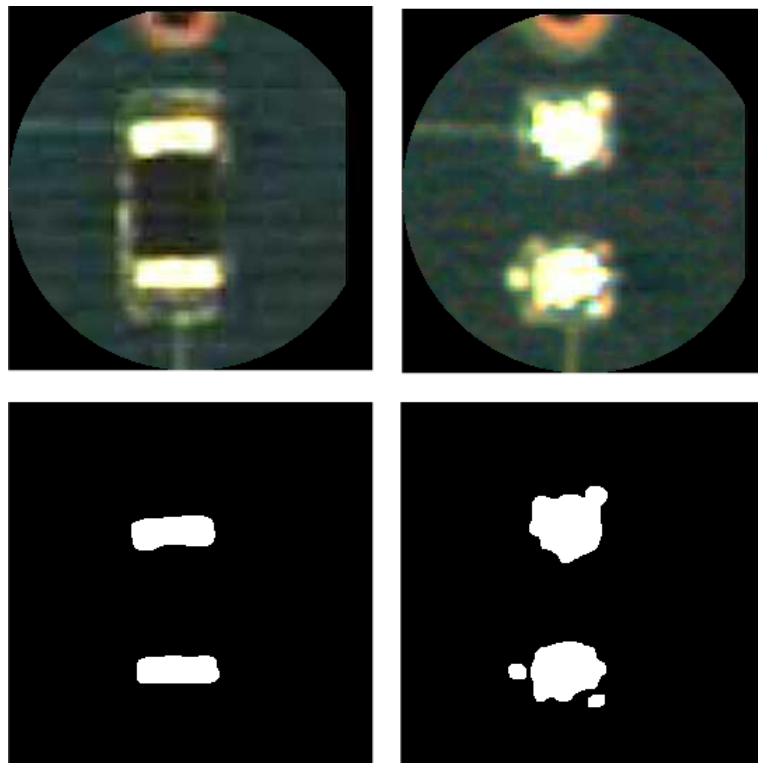


Figura 6.3: En las imágenes de la parte izquierda se muestran la homogeneidad de los extremos de una resistencia en color y en binario. En la parte derecha, se muestra la heterogeneidad de las soldaduras en imágenes a color y binaria.

Esto es que, las coordenadas del centro de masa del objeto O_{x_c} y O_{y_c} , se encuentren entre las coordenadas iniciales F_{x_i} y F_{y_i} , y las coordenadas finales F_{x_f} y F_{y_f} de la región factible respectivamente.

En la figura 6.4 se muestran las regiones factibles en imágenes de soldaduras en posición horizontal y vertical, en las cuales se distinguen más de dos objetos. La imagen de la izquierda se puede apreciar que sólo hay un objeto por región factible, lo cual sugiere que no se trata de una soldadura. La imagen central presenta dos objetos en la parte inferior, pero sólo el más grande se encuentra en la zona factible, esto advierte que no es una soldadura. La imagen derecha presenta dos objetos en la región factible de la izquierda, lo cual indica que se trata de una soldadura.

Las imágenes de la figura 6.5 pertenecen a capacitores o resistencias en posiciones verticales y horizontales donde se muestran sus respectivas regiones factibles. Todas las imágenes presentan dos objetos, cada uno de éstos, contenido en una región factible, lo cual indica que no se trata de una soldadura.

Usando las regiones factibles, afianzamos la probabilidad de identificar la falta de capacitor o resistencia en una imagen.

El principal problema del uso de regiones factibles es cuando el componente no está centrado en la imagen, ya que los objetos de interés muy probablemente no tengan sus centroides dentro de las regiones.



Figura 6.4: Regiones factibles de $m \times n$ en imágenes binarias de soldaduras.

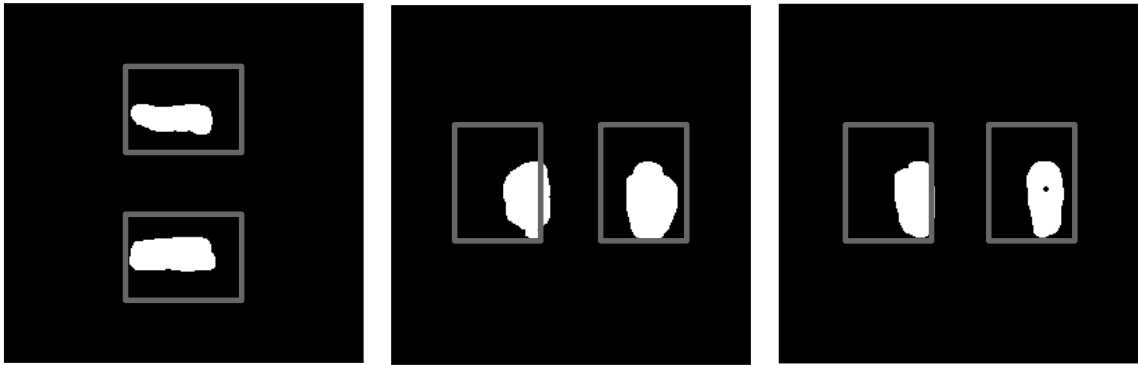


Figura 6.5: Regiones factibles de $m \times n$ en imágenes binarias de capacitores o resistencias.

6.3. Intersección de histogramas de brillo y contraste de texturas RGB

Este método consiste en obtener una medida de similitud representativa entre dos texturas a través de la semejanza de sus histogramas de brillo y contraste correspondientes a cada canal RGB. La semejanza de los histogramas se obtiene a partir del número de puntos de cada uno de estos. Las texturas utilizadas son la del centro del componente y la de alguna otra parte de la tarjeta. Los datos obtenidos serán utilizados en la clasificación para determinar si existe o no, un capacitor o resistencia en una tarjeta.

Cuando se encuentra un capacitor o una resistencia, hay una diferencia entre la textura de la tarjeta y la textura en la región central del componente. En caso contrario, cuando no existe un componente, las texturas del centro de la zona del componente y la tarjeta son similares. Esto se puede apreciar en la figura 6.6.



Figura 6.6: Texturas de una tarjeta con capacitor y sin capacitor.

La variación en el brillo y el contraste suelen ocasionar un cambio visible en una imagen, por lo que estas son las características usadas para definir a la textura. Se hace uso de histogramas de estas características, debido a que éstos representan la frecuencia de valores de una imagen.

Se definen tres secciones en la imagen a color, las cuales se localizan alineadas en forma perpendicular a la orientación del supuesto componente, pasando por el centro de la imagen. Para el caso de un componente en posición vertical, las secciones de análisis son las mostradas en la figura 6.7. Las secciones de análisis de un componente en posición horizontal son descritas en la figura 6.8.

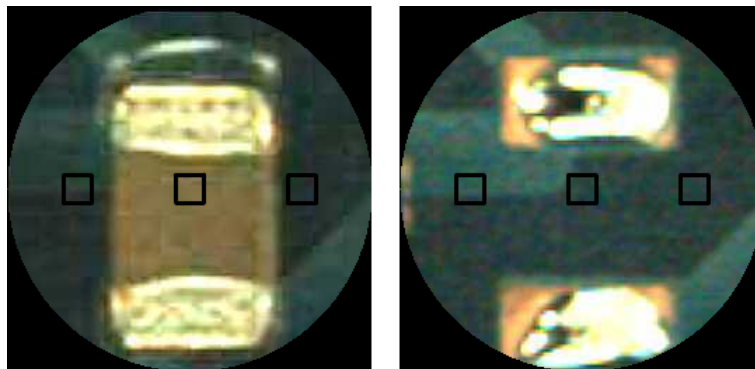


Figura 6.7: Definición de las secciones de extracción de textura de 20x20 píxeles para un capacitor en posición vertical.

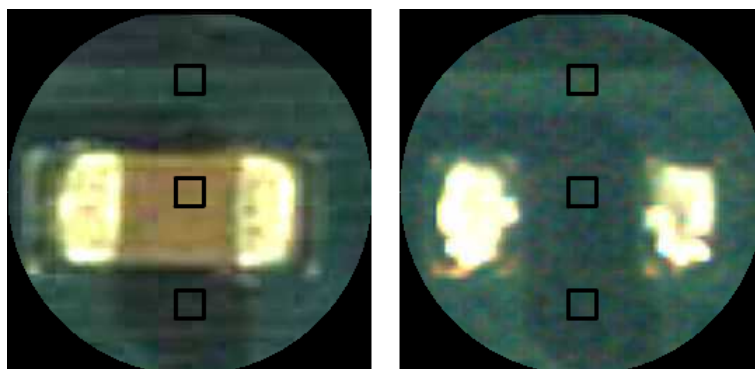


Figura 6.8: Definición de las secciones de extracción de textura de 20x20 para un capacitor en posición horizontal.

Una vez definidas las secciones, de cada una de las texturas correspondientes a éstas, se obtienen los histogramas de brillo y contraste de los canales del RGB. Los histograma obtenidos de la zona del centro (donde supuestamente se encuentra un

componente) se intersectan con sus correspondientes histogramas de cada una de las secciones de los extremos. De esta manera se obtiene un nuevo conjunto de histogramas RGB descriptor de similitud de la sección del centro con las de los extremos.

Para este método no se necesita una iluminación controlada debido a que las secciones de las texturas utilizadas pertenecen a la misma imagen.

En el apéndice A se encuentran imágenes de los histogramas de brillo y contraste de las secciones de las imágenes de la figura 6.8, así como los histogramas resultantes de su intersección.

6.3.1. Histograma de brillo

Un histograma de brillo es la representación de la frecuencia de la luminosidad u oscuridad de una imagen, donde ésta tiene un intervalo de $[0,255]$ tonos.

Para calcular el histograma se considera que cada pixel está rodeado por ocho pixeles, como se muestra en la figura 6.9. El pixel del centro P_C es del cual se quiere obtener el brillo, y los pixeles vecinos P_{N_k} son los que lo rodean.

P_{N_1}	P_{N_2}	P_{N_3}
P_{N_4}	P_C	P_{N_5}
P_{N_6}	P_{N_7}	P_{N_8}

Figura 6.9: Ventana de 3x3 píxeles para el cálculo de los histogramas de brillo y contraste. Con ocho pixeles vecinos

Para obtener el brillo del pixel central B_{P_C} , se calcula la media de las intensidades de los nueve pixeles (los ocho pixeles vecinos y el pixel central). Esto se define como:

$$B_{P_C} = \frac{(\sum_{k=1}^8 P_{N_k}) + P_C}{9} \quad (6.11)$$

donde $0 \leq B_{P_C} \leq 255$. Cada valor B_{P_C} corresponde a un elemento del histograma de brillo h_b [9].

Considerando la figura 6.8, los histogramas de brillo de los canales de color RGB de las secciones de análisis de la imagen con capacitor son mostrados en las imágenes de la izquierda de las figuras A.1, A.2 y A.3 respectivamente del apéndice A.

Los histogramas de brillo de los canales RGB de las secciones de la imagen sin capacitor de la figura 6.8 son las imágenes de la parte izquierda de las figuras A.4, A.5 y A.6 del apéndice A.

6.3.2. Histograma de contraste

El histograma de contraste representa la frecuencia de la variabilidad de los tonos de la imagen. Estos tonos se encuentran en el intervalo $[0,255]$.

Para obtener el histograma de contraste se consideran de manera similar los píxeles como en el cálculo del histograma de brillo (figura 6.9).

El contraste del pixel central C_{P_C} es calculado a través de la resta del promedio de los valores de todos los píxeles vecinos P_{N_k} mayores al pixel central P_C menos el promedio de los píxeles vecinos P_{N_k} menores al pixel central P_C . Esto es definido como:

$$M_s = \frac{(\sum_{k=1}^8 P_{N_k}) + P_C}{n_s + 1} \forall P_{N_k} > P_C \quad (6.12)$$

$$M_i = \frac{(\sum_{k=1}^8 P_{N_k}) + P_C}{n_i + 1} \forall P_{N_k} < P_C \quad (6.13)$$

$$C_{P_C} = M_s - M_i \quad (6.14)$$

con $0 \leq n_s, n_i \leq 8$ y $0 \leq M_s, M_i, C_{P_C} \leq 255$, donde n_s es el número de píxeles vecinos mayores o iguales al pixel central P_C y n_i es el número de píxeles vecinos menores a P_C , con sus medias respectivas M_s y M_i .

Cada uno de los valores C_{P_C} corresponde a un elemento del histograma de contraste h_c [9].

En las ecuaciones () y () se agrega el termino P_C con el objeto de considerar el valor del pixel del centro y se suma 1 a n_s para balancear.

Los histogramas de contraste de cada canal del RGB de la imagen con capacitor de la figura 6.8 corresponden a las imágenes de la derecha de las figuras A.1, A.2 y A.3 del apéndice A respectivamente.

Los histogramas de de contraste del RGB de la imagen sin capacitor de la figura 6.8 son los mostrados en las imágenes de la derecha de las figuras A.4, A.5 y A.6 respectivamente, las cuales se encuentran en el apéndice A.

6.3.3. Intersección de histogramas

Con la intersección de histogramas se obtiene la medida de similitud de dos histogramas en base a su cantidad de puntos. Entre más cercanos se encuentran la cantidad de puntos del nuevo histograma (resultado de la intersección) con cualquiera de los puntos de los histogramas base, la semejanza es mayor [9].

La descripción de esta técnica utiliza la figura 6.10. Esta técnica consiste en:

1. Seleccionar dos histogramas, a los cuales se les denominara histogramas base. Por ejemplo los histogramas a y b .
2. Se contabilizan los puntos de los histogramas base seleccionados (la cantidad de puntos es la misma para cada histogramas). En el caso de los histogramas a y b se puede observar que ambos tienen 30 puntos.
3. Generar un nuevo histograma como resultado de tomar los valores más pequeños de entre los dos histogramas seleccionados, como se muestra en el histograma c .
4. Se contabilizan los puntos del nuevo histograma (la cantidad de puntos siempre será menor o igual al número total de puntos de cualquiera de los histogramas base). La cantidad de puntos del histograma c es de 23.

Considerando como ejemplo la figura 6.8, las imágenes de los histogramas resultantes de la intersección de las secciones superior-central e inferior-central de las imágenes con capacitor y sin capacitor, son las mostradas en las figuras A.7, A.8, A.9 y A.10 respectivamente, en el apéndice A.

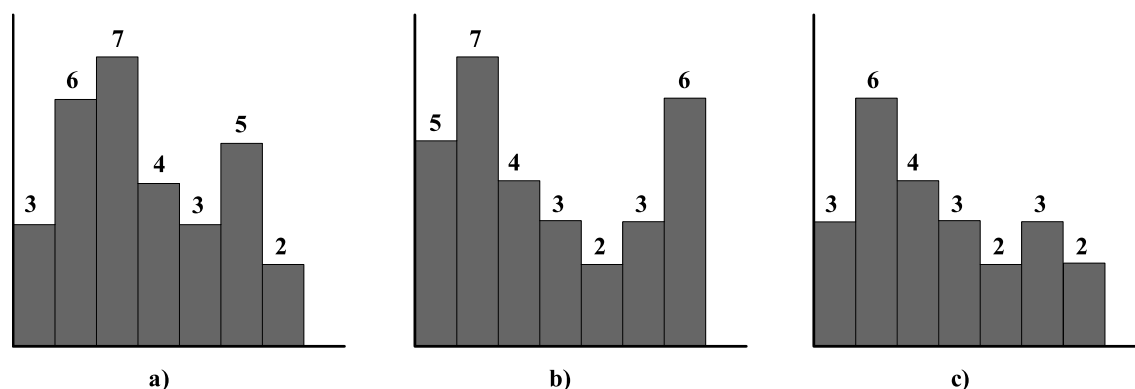


Figura 6.10: Ejemplo de la intersección de histogramas. De los histogramas de la imagen *a* y *b* se utilizan las barras más pequeñas de cada intervalo, dando como resultado el histograma de la imagen *c*.

6.4. Otros métodos

Para el tipo de imágenes analizadas, los siguientes métodos no proporcionan información importante a nivel del conjunto al que pertenecen los objetos, puesto que entre clases los valores fluctúan demasiado. Lo anterior significa que parte importante de los objetos presentan características diferentes a las de la clase a la que pertenecen, lo cual produce resultados con poca precisión en la clasificación del error.

Sin embargo lo anterior no quiere decir que estas características no sean útiles en otros tipos de componentes electrónicos. Estas características podrían ser representativas en procedimientos de otros objetivos requeridos en la solución de este problema.

6.4.1. Distancia euclidiana

La distancia euclidiana se lleva a cabo entre los dos centroides de la concentración de objetos de cada lado de la imagen binarizada. Los lados se definen de manera perpendicular a la orientación del componente en la imagen.

En las imágenes de las figuras 6.11 y 6.12 se pueden ver las líneas divisoras (continuas) de los lados, las cuales pasan por el centro de la imagen y las líneas que denotan la distancia euclidiana entre los centroides de la concentración de objetos de cada lado.

En la figura 6.11, ambas distancias euclidianas son parecidas a pesar de que la imagen de la izquierda pertenece a extremos de un capacitor o resistencia y la imagen de la derecha a soldaduras. El hecho de que las distancias sean similares indica que se trata de imágenes del mismo tipo.

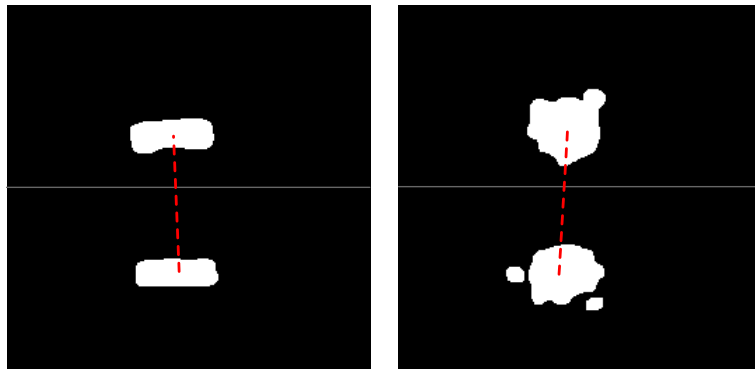


Figura 6.11: La distancia euclidiana entre ambas imágenes es similar.

La figura 6.12 se muestran distancias euclidianas diferentes, sin embargo las imágenes contienen muchos objetos, lo cual provoca medidas erróneas. La imagen de la izquierda podría considerarse útil, ya que la distancia mostrada es similar a la de una imagen con componente. La imagen de la derecha es completamente errónea, pues se observa una distancia muy grande, la cual sería catalogada por un clasificador como correspondiente a una imagen con componente.

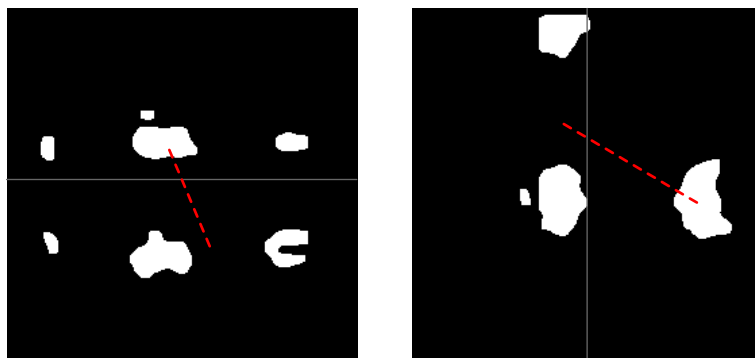


Figura 6.12: Entre ambas imágenes, la distancia euclidiana es distinta.

Se puede observar que el utilizar la distancia euclidiana como método de análisis no es adecuado, debido a lo expuesto en las figuras 6.11 y 6.12. Además este método se ve limitado a las dimensiones de los objetos mostrados, ya que si se tratara de un componente más pequeño o grande, estas distancias cambiarían, lo cual sería muy común en un conjunto real.

6.4.2. Perímetro de objetos

El cálculo del perímetro sólo se lleva a cabo en imágenes binarizadas con dos objetos porque este se cataloga directamente con el clasificador. En el caso de existir más de dos objetos, los sobrantes son limpiados con una línea medía como en el caso de los momentos o con las morfologías.

El perímetro P_A de un objeto A lo obtenemos como:

$$P_A = A - A \ominus B \quad (6.15)$$

donde $A \ominus B$ es la erosión del objeto A con respecto a un objeto B .

En la imagen derecha de las figuras 6.13 y 6.14 se pueden apreciar los perímetros resultantes de las imágenes de la izquierda de cada figura respectivamente.

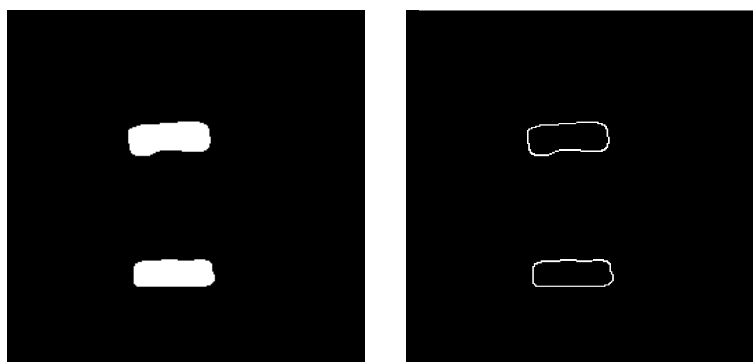


Figura 6.13: Perímetro de una imagen binarizada de un capacitor o una resistencia.



Figura 6.14: Perímetro de una imagen binarizada perteneciente a una soldadura.

Los perímetros de ambas imágenes (figuras 6.13 y 6.14) son similares a pesar de ser de distinto tipo. Sin importar si la imagen es de un componente o de soldadura, los objetos presentan frecuentemente perímetros similares. Por esta razón es que esta característica no fue útil para hacer la distinción del tipo de imagen.

Capítulo 7

Clasificación del tipo de error

La clasificación del tipo de error consiste en determinar si existe o no, un componente electrónico (específicamente un capacitor o una resistencia) en determinada área de una imagen de una tarjeta de montaje de superficie. En el caso de que no se encuentre un componente, se considera que existe la soldadura utilizada para montarlo.

La clasificación se lleva a cabo utilizando como descriptores a los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 obtenidos de los extremos de capacitores y resistencias, o de las soldaduras de estos. También se usa la cantidad de puntos resultantes de la intersección de histogramas RGB de las texturas de secciones definidas en la imagen.

Para cada método se definieron dos clases patrón como sigue:

- Para los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 , se definió una clase con la media de los valores obtenidos de un conjunto de momentos invariantes de los extremos de capacitores y resistencias, y la otra clase se define con la media de momentos invariantes pertenecientes a un conjunto de soldaduras.
- En el caso de la intersección de histogramas, las dos clases patrón se definen utilizando las medias de los valores de la cantidad de puntos. Una clase pertenece a imágenes con componente y la otra a imágenes sin componente.

Conceptualmente entre cada clase, los valores de los patrones son evidentemente distintos, mientras que entre patrones del mismo conjunto, los valores en su mayoría son similares entre sí.

Para incrementar la efectividad de esta clasificación del error, se realizó la combinación de la ambos métodos.

7.1. Definición del clasificador de distancia mínima

Un clasificador es un método basado en el uso de funciones de decisión o discriminación, para poder determinar la clase a la que pertenece un elemento analizado.

Un patrón x es un vector de descriptores denotado como:

$$\mathbf{x} = [x_1, x_2, \dots, x_n]^T \quad (7.1)$$

donde cada x_i es una característica.

Definimos a una clase patrón como una familia de patrones que comparten propiedades en común, y son denotadas como w_1, w_2, \dots, w_M .

Un vector prototipo m_j (o vector de medias) es un vector representativo de una clase patrón j , y se define como:

$$\mathbf{m}_j = \frac{1}{N_j} \sum_{x \in w_j} \mathbf{x} \quad j = 1, 2, \dots, M \quad (7.2)$$

donde N_j es el número de patrones x pertenecientes a la clase patrón w_j . La cantidad de clases patrón está representada por M .

La manera para determinar a qué clase pertenece un patrón desconocido x es asignarlo al vector prototipo con valores más parecidos.

Lo anterior se obtiene evaluando la función:

$$d_j(\mathbf{x}) = \mathbf{x}^T \mathbf{m}_j - \frac{1}{2} \mathbf{m}_j^T \mathbf{m}_j \quad (7.3)$$

donde el d_j que produce el mayor valor determina la clase w_j a la que pertenece el patrón \mathbf{x} .

El límite de decisión de entre a cual clase pertenece el patrón \mathbf{x} , si a w_i o w_j , lo obtenemos a través de:

$$d_{ij}(\mathbf{x}) = d_i(\mathbf{x}) - d_j(\mathbf{x}) \quad (7.4)$$

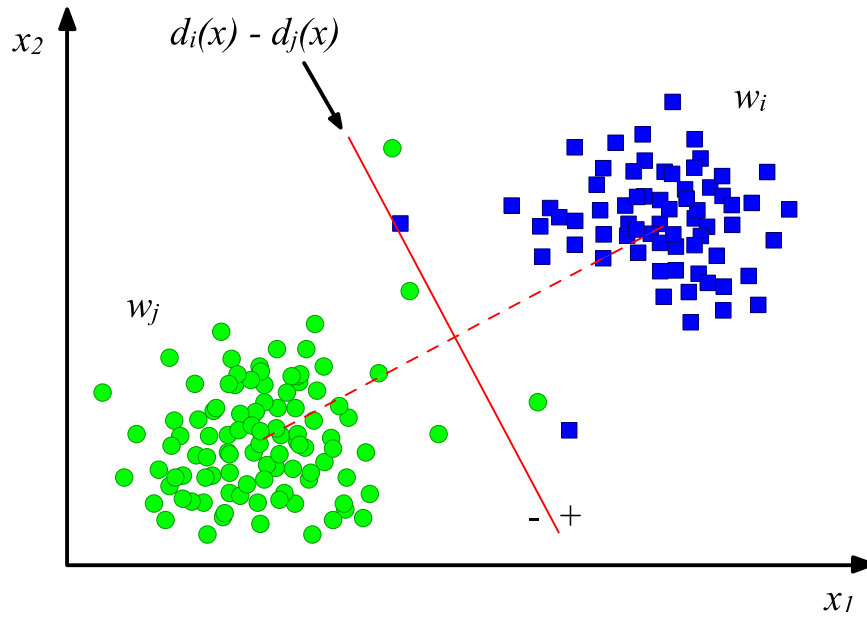


Figura 7.1: Limite de decisión del clasificador de distancia mínima.

si $d_{ij}(\mathbf{x}) \geq 0$, entonces pertenece a la clase w_i , de ser $d_{ij}(\mathbf{x}) < 0$, entonces pertenece a la clase w_j [14].

En la figura 7.1 se muestran dos clases w_i y w_j representadas por cuadros y círculos respectivamente. Cada cuadro y círculo representa un patrón de dos características (x_1 y x_2) de cada clase en un espacio $2D$. Se observa gráficamente el límite de decisión $d_{ij}(\mathbf{x}) = d_i(\mathbf{x}) - d_j(\mathbf{x})$ en posición perpendicular, justamente a la mitad de la distancia entre los centros de masa de las clases w_i y w_j .

Considerando como ejemplo a los patrones de la clase w_j (círculos) en la figura 7.1, se puede observar como la mayoría se encuentran concentrados en una área específica. Algunos otros patrones se encuentran separados de la mayor concentración, lo que significa que son elementos que a pesar de pertenecer a la clase w_j , no comparten mucha similitud en sus características con la principal densidad de patrones.

Un caso extremo, también mostrado en la figura 7.1, es cuando patrones de la clase w_j se encuentran del otro lado del límite de decisión, como si fuera un elemento de la clase w_i . Este patrón que pertenece a la clase w_j presenta todas las características de la clase w_i .

7.2. Entrenamiento del clasificador de distancia mínima

Para entrenar al clasificador, se consideran todas las clases que se quieren para realizar la comparación y se realiza lo siguiente:

1. Se selecciona una clase patrón (conjunto de muestras o patrones).
2. Se obtienen las características (descriptores) de las muestras seleccionadas.
3. Se obtienen los valores medios de los descriptores y se crea un vector prototipo con ellos, el cual es considerado como el resultado del entrenamiento. El clasificador de distancia mínima utiliza este vector de medias para caracterizar la clase.

En el caso de este trabajo se definen dos clases por método, una para imágenes con capacitores y resistencias, y otra para imágenes con soldadura.

Capítulo 8

Implementación y resultados

Este trabajo de investigación se enfoca a implementar un sistema en software que identifique si un componente (específicamente capacitores y resistencias) existe en una imagen a color perteneciente a una sección de una tarjeta de montaje de superficie la cual fue previamente identificada como defectuosa.

El conjunto proporcionado por Sony es de aproximadamente 8000 imágenes (conjunto de análisis), de las cuales sólo tenemos interés en las que contienen resistencias (color negro) y capacitores (color café o gris), estos elementos se presentan en varios tamaños y posiciones en la imagen. Por la razón anterior, se realizó la implementación del sistema clasificador de imágenes que determinará cuales incluyen resistencias o capacitores y las posiciones de estos.

Las características funcionales de las implementaciones están definidas de acuerdo a las necesidades del problema que se está resolviendo. Ambas implementaciones comparten métodos de preprocesamiento, los cuales son descritos en el capítulo 4. La especificación de la red neuronal utilizada para el sistema de clasificación de imágenes se encuentran mencionados en el capítulo 5. Los procedimientos usados en el analizador que recibe una imagen y determina si existe un componente electrónico (capacitor o resistencia) o no, se mencionan en los capítulos 6 y 7.

Las implementaciones fueron realizadas en ANSI C debido a que es un lenguaje estándar (con las funciones más básicas), lo cual permite que se pueda compilar y ejecutar en cualquier sistema operativo que cuente con un compilador de C. Algunos compiladores de C tienen diferencias en su modo de funcionar, por ejemplo en Linux un dato de tipo entero es representado por 4 bytes, mientras que Windows los representa con 2 bytes, esto provoca desde problemas de precisión hasta un mal funcionamiento total. El sistema operativo de desarrollo fue Linux ya que este es de los más estables. Otra razón importante por la que se utilizó Linux es porque es de código abierto.

8.1. Implementación de la clasificación de imágenes

La clasificación de imágenes es necesaria para determinar si estas contienen resistencias o capacitores, ya que la implementación del análisis e identificación solo trabaja con imágenes de este tipo.

Para realizar la clasificación, se hace uso de una red neuronal la cual se encarga de catalogar a que clase pertenece cada una de estas imágenes. El funcionamiento de esta implementación se divide en dos partes: entrenamiento y clasificación.

El entrenamiento consiste en adiestrar la red neuronal con un conjunto de imágenes, las cuales definen las características de las clases en las que se organizaran las imágenes analizadas. La manera en la que se implemento en software es la descrita en el procedimiento (1).

En la clasificación se cataloga una imagen de la cual se desconoce su categoría (clase a la que pertenece). Esta se lleva a cabo una vez que la red neuronal fue entrenada. El procedimiento (2) muestra la implementación en software.

Procedimiento 1 Entrenamiento de la Red Neuronal para la clasificación de imágenes

Entrada: Lista de imágenes RGB de entrenamiento

Salida: Red neuronal entrenada

Inicialización de variables

for imagen = 1 **to** total de imágenes **do**

if (img1 \Leftarrow Lectura y almacenamiento (imagen)) = TRUE **then**

if (img2 \Leftarrow Extracción de la zona útil (img1)) = TRUE **then**

 img3 \Leftarrow Conversión RGB a escala de grises (img2)

 img4 \Leftarrow Suavizado (img3)

 medias \Leftarrow Media de la suma y diferencia de histogramas (img4)

 datos \Leftarrow Almacenamiento de Datos para la red (medias)

end if

end if

end for

Red neuronal entrenada \Leftarrow Red neuronal en modo de entrenamiento (datos)

Procedimiento 2 Clasificación del tipo de imagen

Entrada: Lista de imágenes RGB para clasificar

Salida: Clase a la que pertenece la imagen

Inicialización de variables

for imagen = 1 **to** total imágenes **do**

if ($img1 \leftarrow$ Lectura y almacenamiento (imagen)) = TRUE **then**

if ($img2 \leftarrow$ Extracción de la zona útil (img1)) = TRUE **then**

$img3 \leftarrow$ Conversión RGB a escala de grises (img2)

$img4 \leftarrow$ Suavizado (img3)

$medias \leftarrow$ Media de la suma y diferencia de histogramas (img4)

 Clase a la que pertenece la imagen \leftarrow Red neuronal en modo de prueba (medias)

end if

end if

end for

8.1.1. Características de la red neuronal implementada

La red neuronal utilizada para resolver el problema de la clasificación consiste de 3 capas:

- Capa de entrada

Tiene 100 nodos de entrada, debido a que la imagen está dividida en 100 partes (figura 4.16). El valor x_i recibido por cada nodo de entrada es un número real que se encuentra normalizado entre 0.0 y 1.0, el cual es obtenido de la media de la suma y diferencia de histogramas de una sección $y_{k,l}$ correspondiente a una imagen en tonos de gris (sección 4.8, pagina 45).

- Capa oculta

La determinación del número apropiado de capas ocultas y nodos ocultos es una tarea difícil en el desarrollo de redes neuronales. Una red con muy pocos nodos ocultos es incapaz de diferenciar entre patrones complejos conduciendo sólo a una estimación lineal de los datos, en contraste si la red tiene muchos nodos ocultos, ésta seguirá el ruido en los datos debido a la sobre-parametrización, conduciendo a una generalización de los datos [22].

La red utilizada contiene una sola capa oculta con 40 nodos. Este número de nodos fue determinado de manera empírica. Se hicieron pruebas con 10, 20, 30 y 40 nodos. Con cada una de estas cantidades de nodos, se realizaron ejecuciones, en todas, al final se introdujo el mismo conjunto de entrenamiento para verificar la calidad del aprendizaje. Varios elementos no fueron debidamente clasificados con 10 y 20 nodos ocultos, con 30 nodos ocultos se obtuvieron resultados regulares (pocos elementos sin clasificar) y con 40 nodos ocultos todos los elementos del conjunto de entrenamiento fueron clasificados correctamente (en casi todas

las pruebas). Las pruebas realizadas indican que una cantidad grande de nodos ocultos ayuda a que la identificación de objetos sea más precisa pero la teoría nos dice que demasiados nodos afectan la clasificación.

- Capa de salida

Se tienen 31 nodos de salida, donde cada uno de estos es representante de una clase de imagen. De esta manera cuando la red clasifica y determina un nodo de salida también está determinando a qué clase de imagen se asemeja más la imagen introducida. Si una nueva clase se define también se debe de agregar un nodo de salida.

- Los valores iniciales de los pesos están en el intervalo $[-0.3, 0.3]$.
- Grado de aprendizaje utilizado para la red es de $\eta = 0.11$, valores más grandes provocan una convergencia prematura del error, mientras que valores más pequeños tardan mucho en converger.
- El coeficiente de momento es $\mu = 0.07$, un número grande provoca una inestabilidad en la convergencia del error.
- El error utilizado es de $\varepsilon = 0.001$. De manera empírica se determino que un valor superior a $\varepsilon = 0.003$ provoca un mal entrenamiento y valores menores a mejor desempeño.
- El número de iteraciones internas para mejorar la precisión de la corrección del error de cada patrón está definido como $n = 20$. Valores más pequeños no ofrecen una buena convergencia, mientras que valores más grandes provocan que la búsqueda quede atrapada en óptimos locales.
- El número aproximado de épocas necesario para obtener un error cercano a $\varepsilon = 0.001$ es de 15000.

8.1.2. Clases definidas para el entrenamiento de la red neuronal

Debido a la enorme variedad de texturas, es difícil definir pocas clases que alberguen a todas las imágenes. Aunado a lo anterior, muchas de las imágenes no son representativas de algún elemento, por lo que no resulta conveniente crear una clase con ellas (figura 8.1).

Por otro lado si existen texturas representativas pero éstas son únicas por lo que especificar una clase para una o dos imágenes no es útil (figura 8.2 y figura 8.3).

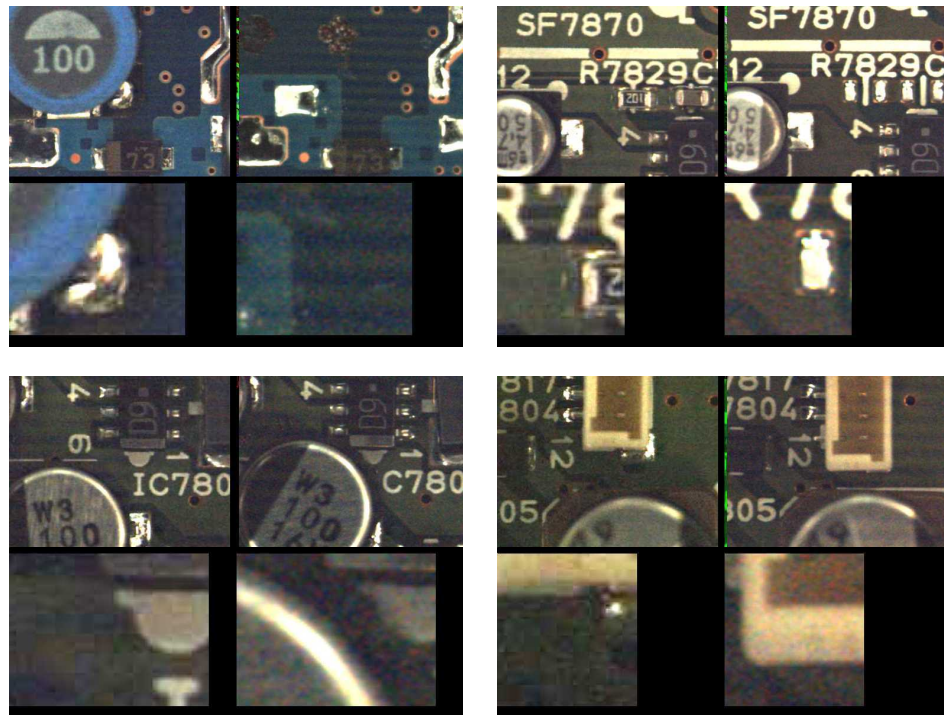


Figura 8.1: Imágenes no representativas



Figura 8.2: Esta imagen es única en el conjunto de análisis.

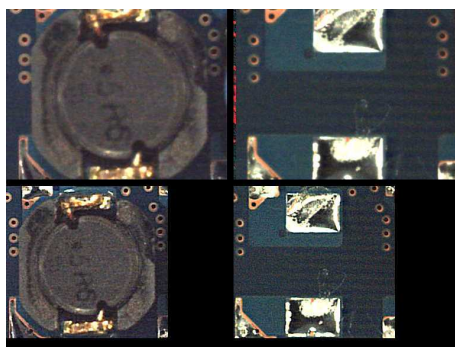


Figura 8.3: Existen muy pocas imágenes con una textura similar.

El conjunto de entrenamiento seleccionado consta de 381 imágenes (elegidas del conjunto de 8000) y de éstas se especificaron 31 clases distintas (de la clase 0 a la clase 30), las cuales catalogan entre: resistencias y capacitores rectangulares de distintos tamaños y orientaciones (horizontal y vertical), condensadores redondos rotados 0, 90, 180 y 270 grados y circuitos integrados (algunas partes, debido a sus tamaños, por lo que en algunas imágenes solo se muestran terminales).

También se especificó un conjunto nombrado como *indefinido* donde se agrupan imágenes con figuras de distintas formas, que no pudieron ser catalogadas en otras clases. La definición de esta clase es con el fin de que la red clasifique como *desconocida* cualquier imagen que no empate de manera significativa en alguna otra clase.

Si el conjunto de clases de entrenamiento fuera muy pequeño, los pesos de la red neuronal no serían más precisos, lo que ocasionaría que una gran cantidad de imágenes con características distintas (digamos componentes diferentes) serían catalogadas erróneamente.

Por otro lado una gran cantidad de clases definiría muchas imágenes de manera incorrecta solo porque los valores de sus características estarían fuera de la tolerancia obtenida debido a la gran cantidad de categorías con que se entreno la red.

Lo correcto es definir un número de clases en base a la tolerancia que se requiere para catalogar las características de una imagen.

El criterio que se toma para definir las clases es correspondiente a las formas, tamaños, colores y posiciones de los elementos electrónicos en las imágenes.

Ejemplos representativos de las 31 clases ocupadas para entrenar la red neuronal se muestran en la sección B.1.

8.1.3. Resultados y estadísticas de la clasificación de imágenes

La tabla 8.1 es la estadística de la clasificación de aproximadamente 8000 imágenes con la red neuronal. La red fue adiestrada con un conjunto de entrenamiento de 381 imágenes que describen 31 clases distintas.

En el apéndice B, específicamente en la sección B.2, se pueden consultar un ejemplo exitoso de convergencia del error (B.2.1) en el cual se puede observar su desempeño y la comprobación de la calidad de entrenamiento a través del conjunto de adiestramiento (B.2.2). En el apéndice C se muestran ejemplos de imágenes mal clasificadas.

Una razón importante por la que se obtienen errores en la clasificación y una gran varianza entre los resultados, se debe a la distribución de valores de la medias de la suma y diferencia de los histogramas.

Estos valores no pueden ser más precisos debido a que la textura de las imágenes es muy variada. La calidad de la imagen es muy mala, el ambiente en el que se obtuvo no es controlado y los componentes contenidos no se encuentran en posiciones constantes (están movidos).

Las clases que mejor fueron clasificadas, son las que contienen componentes en posiciones constantes, con poca variación de en la iluminación (color). Las dimensiones de los objetos no han presentado problema debido a que se conservan dentro de los rangos de tolerancia impuesto por las clases de entrenamiento.

Resultados de la clasificación de 8000 imagenes					
clase	total de elementos clasificados	Elementos clasificados correctamente	Elementos clasificados incorrectamente	porcentaje de correctos	Porcentaje de incorrectos
clase 0	53	19	34	35.85 %	64.15 %
clase 1	56	46	10	82.14 %	17.86 %
clase 2	80	72	8	90.00 %	10.00 %
clase 3	255	209	46	81.96 %	18.04 %
clase 4	416	366	50	87.98 %	12.02 %
clase 5	248	197	51	79.44 %	20.56 %
clase 6	57	57	0	100.00 %	0.00 %
clase 7	48	27	21	56.25 %	43.75 %
clase 8	161	101	60	62.73 %	37.27 %
clase 9	648	455	193	70.22 %	29.78 %
clase 10	91	85	6	93.41 %	6.59 %
clase 11	107	80	27	74.77 %	25.23 %
clase 12	67	37	30	55.22 %	44.78 %
clase 13	91	75	16	82.42 %	17.58 %
clase 14	135	25	110	18.52 %	81.48 %
clase 15	166	98	68	59.04 %	40.96 %
clase 16	231	151	80	65.37 %	34.63 %
clase 17	89	39	50	43.82 %	56.18 %
clase 18	17	8	9	47.06 %	52.94 %
clase 19	143	131	12	91.61 %	8.39 %
clase 20	7	5	2	71.43 %	28.57 %
clase 21	47	24	23	51.06 %	48.94 %
clase 22	22	8	14	36.36 %	63.64 %
clase 23	17	13	4	76.47 %	23.53 %
clase 24	345	288	57	83.48 %	16.52 %
clase 25	147	125	22	85.03 %	14.97 %
clase 26	251	249	2	99.20 %	0.80 %
clase 27	103	92	11	89.32 %	10.68 %
clase 28	4	3	1	75.00 %	25.00 %
clase 29	6	4	2	66.67 %	33.33 %
clase 30	3950	3790	160	95.95 %	4.05 %
Totales:	8058	6879	1179	71.22 %	28.78 %

Cuadro 8.1: Resultados de la clasificación de aproximadamente 8000 imágenes.

8.2. Implementación del analizador para la identificación del error

El analizador implementado examina imágenes que contienen capacitores o resistencias (componentes que Sony denomina como *tipo chip*) e identificar la existencia de estos.

La implementación arroja dos resultados: *sin componente* o *indefinido*. La razón por la cual se manejan estos dos resultados se debe que a Sony le interesa el detectar la falta de componente con un alto nivel de eficiencia¹. El procedimiento (3) muestra la implementación para realizar la identificación.

El analizador consiste en tres partes las cuales son: pre-procesamiento, extracción de características y clasificación del tipo de error, las cuales son mostradas en la figura 3.4.

Procedimiento 3 Análisis e Identificación de la existencia de un capacitor o resistencia

Entrada: Ruta de la imagen RGB para análisis, clase de la imagen

Salida: *sin componente* / *indefinido*

Inicialización de variables

if clase de la imagen = capacitor **or** clase de la imagen = resistencia **then**

if (*img1* \Leftarrow Lectura y carga de la imagen) = TRUE **then**

if (*img2* \Leftarrow Extracción de la zona útil con mascara (*img1*)) = TRUE **then**

img3 \Leftarrow Conversión RGB a escala de grises (*img2*)

img4 \Leftarrow Suavizado (*img3*)

img5 \Leftarrow Binarización de la imagen (*img4*)

img6 \Leftarrow Erosión (*img5*)

img7 \Leftarrow Dilatación (*img6*)

obj \Leftarrow Extracción de objetos (*img7*)

mi \Leftarrow Calculo de momentos invariantes (*img8*)

pih \Leftarrow Intersección de histogramas RGB (*img2*)

nobj \Leftarrow Contabilización de objetos (*obj*)

sin componente / *indefinido* \Leftarrow Clasificador de distancia mínima (*mi*, *pih*, *nobj*)

end if

end if

end if

¹Véase falsos positivos y falsos negativos en la sección 2.5.

8.2.1. Características del analizador

Las características finales elegidas para la implementación y en las que se basan los resultados reportados, son producto de varias pruebas, las cuales tuvieron como objeto encontrar los valores más eficientes. Los valores definidos para cada característica son los más universales, considerando la variedad de imágenes.

Las características consideradas para el análisis y sus valores son los siguientes:

- las imágenes analizadas tienen una dimensión de 768×576 pixeles. Imágenes de otras medidas no son procesadas.
- Las coordenadas de extracción utilizadas son $(560, 425)$, pues en la mayoría de las imágenes pertenecientes a capacitores o resistencias, los elementos buscados tienen centro en estas coordenadas.
- El radio de la máscara circular utilizada es de $r = 130$ pixeles. Con esta longitud de radio se enmascaran casi de manera completa, elementos no centrados en la imagen
- El umbral utilizado para la binarización de la imagen es $T = 200$. Este valor de umbral separa eficientemente los extremos de componente y las soldaduras (ambos colores claros) del resto de elementos en la imagen de la tarjeta.
- En la suma y diferencia de histogramas, las distancias que define la relación entre puntos son $d_1 = 10$ y $d_2 = 10$.
- Se usaron los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 ya que éstos proporcionan datos más útiles para la identificación del error. Se definen dos vectores prototipo con los momentos invariantes de la forma $v = (\phi_1, \phi_2, \phi_3)$.

Los mejores valores de momentos obtenidos del entrenamiento del clasificador de distancia mínima son:

El vector prototipo cuando existe capacitor o resistencia es:

$$v_1 = (0.198716745336752, 0.008134169712974, 0.001168128417321)$$

En el caso de soldadura, el vector es:

$$v_2 = (0.253716779486000, 0.038767945551026, 0.000443554966773)$$

- Para la contabilización de objetos, tomando como referencia a la imagen extraída, las coordenadas de la zona factible están definidas con los valores:
 - Para zona Z^1 : $Z_i^1 = (20, 60)$ y $Z_f^1 = (70, 140)$
 - Para zona Z^2 : $Z_i^2 = (130, 60)$ y $Z_f^2 = (180, 140)$
- En el caso de extracción y análisis de texturas RGB, el vector prototipo perteneciente a la cantidad de puntos obtenidos de los histogramas RGB cuando existe componente es $v_1 = (1250)$, en el caso de no existir componente el vector es $v_2 = (786)$.
- Las zonas de extracción de textura para elementos en posición vertical están definidas como:
 - Coordenadas de la Zona 1: $Z_i^1 = (40, 90)$ y $Z_f^1 = (60, 110)$
 - Coordenadas de la Zona 2: $Z_i^2 = (90, 90)$ y $Z_f^2 = (110, 110)$
 - Coordenadas de la Zona 3: $Z_i^3 = (140, 90)$ y $Z_f^3 = (160, 110)$

las coordenadas para elementos en posición horizontal son:

- Coordenadas de la Zona 1: $Z_i^1 = (90, 40)$ y $Z_f^1 = (110, 60)$
- Coordenadas de la Zona 2: $Z_i^2 = (90, 90)$ y $Z_f^2 = (110, 110)$
- Coordenadas de la Zona 3: $Z_i^3 = (90, 140)$ y $Z_f^3 = (110, 160)$

Se puede apreciar entre las coordenadas Z horizontales y verticales, que sus valores x y y se encuentran invertidos, esto se debe a que en la imagen extraída, las zonas de texturas son similares si se hace una rotación de 90° de la imagen con origen en el centro.

8.2.2. Resultados y estadísticas del análisis e identificación

Se realizaron pruebas con dos tipos de conjuntos de análisis, el primero de ellos de con 443 elementos y el segundo con 945 elementos.

- Conjunto de 443 elementos

Este conjunto se caracteriza por albergar imágenes de capacitores y resistencias de tamaño pequeño, posicionados aproximadamente en el centro de la imagen (haciendo referencia a la región útil extraída) con orientaciones horizontal y vertical. Ejemplos de imágenes de este conjunto son mostrados en la figura 8.4.

Los resultados y estadísticas mostrados en las tablas 8.2, 8.3, 8.4, 8.5, 8.6 y 8.7 fueron obtenidos de métodos de manera individuales y en conjunto, con el objeto de conocer su efectividad. También se presentan valores de métodos poco eficientes debido a las características del problema. Estos valores fueron clasificados con el clasificador de distancia mínima.

- Conjunto de 945 elementos

Este conjunto contiene 945 imágenes sin componentes y con variso tamaños de soldadura de varios tamaños, en orientaciones horizontal y vertical. Gran cantidad de los elementos buscados se encuentran en posiciones diferentes a las del centro de la imagen, como se puede ver en las figuras 8.5 y 8.6.

Los resultados en la tabla 8.8 fueron obtenidos con el clasificador de distancia mínima, el cual utilizo los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 , la contabilización de objetos y la extracción y análisis de histogramas de textura de regiones de la tarjeta.

Los resultados se ven afectados debido a la distribución de valores de las características en el dominio. Los valores de las características de las imágenes con componente se encuentran muy cercanos a los valores de las imágenes sin componente. Incluso hay imágenes que al momento de analizar sus características, estas presentan las del otro conjunto.

Otras causas de error en la clasificación son: la variación de la posición de los componentes, la calidad e iluminación de las imágenes.

Con imágenes con mayor nitidez y en un ambiente controlado, los resultados seguramente serian mejores.

La **eficiencia** de la detección de un componente en la imagen se mide de acuerdo a la cantidad de imágenes *sin componente* que son *clasificadas correctamente*, reduciendo lo más posible la cantidad de imágenes *sin componente clasificadas incorrectamente*. Las imágenes clasificadas con componente pasan a tener una importancia secundaria².

Momentos Invariantes ϕ_1 , ϕ_2 y ϕ_3		
Conjunto de imágenes	N° de clasificados como <i>sin componente</i>	Porcentaje de efectividad
Con componente	216 / 443	48.758465 %
Sin componente	424 / 443	95.711061 %

Cuadro 8.2: Resultados del análisis con momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 obtenidos de las formas de la imagen.

Contabilización de objetos		
Conjunto de imágenes	N° de clasificados como <i>sin componente</i>	Porcentaje de efectividad
Con componente	179 / 443	40.406321 %
Sin componente	179 / 443	40.406321 %

Cuadro 8.3: Resultados del análisis contabilizando la cantidad de objetos presentes en la imagen.

Texturas RGB		
Conjunto de imágenes	N° de clasificados como <i>sin componente</i>	Porcentaje de efectividad
Con componente	130 / 443	29.345372 %
Sin componente	434 / 443	97.968397 %

Cuadro 8.4: Resultados usando la extracción y el análisis de histogramas de texturas RGB de zonas de la imagen.

²Véase falsos positivos y falsos negativos en la sección 2.5.

M. I. ϕ_1 , ϕ_2 y ϕ_3 , contabilización de objetos y texturas RGB		
Conjunto de imágenes	N° de clasificados como <i>sin componente</i>	Porcentaje de efectividad
Con componente	403 / 443	90.970655 %
Sin componente	443 / 443	100.000000 %

Cuadro 8.5: Resultados del análisis, combinando los momentos invariantes ϕ_1 , ϕ_2 y ϕ_3 , la contabilización de objetos y, la extracción y análisis de texturas RGB.

Distancia Euclidiana		
Conjunto de imágenes	N° de clasificados como <i>sin componente</i>	Porcentaje de efectividad
Con componente	144 / 443	32.505643 %
Sin componente	427 / 443	96.388262 %

Cuadro 8.6: Resultados del análisis usando la distancia euclidiana entre centroides de las masas de objetos.

Perímetro		
Conjunto de imágenes	N° de clasificados como <i>sin componente</i>	Porcentaje de efectividad
Con componente	229 / 443	51.693002 %
Sin componente	361 / 443	81.489842 %

Cuadro 8.7: Resultados del análisis utilizando el perímetro de las figuras.

Resultados del análisis de 945 imágenes	
N° de clasificados como <i>sin componente</i>	Porcentaje de efectividad
873 / 945	92.380952 %

Cuadro 8.8: Resultados del análisis de un conjunto de 945 imágenes sin componente (soldaduras en posición horizontal o vertical).

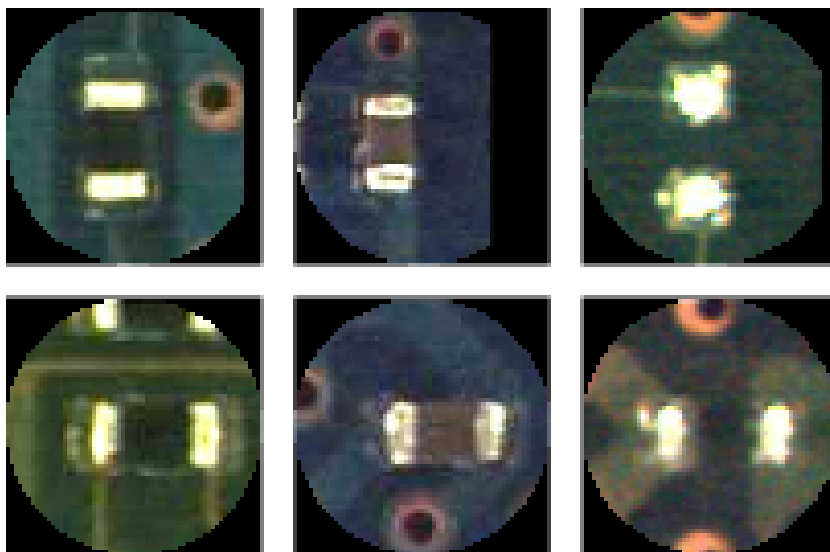


Figura 8.4: Elementos contenidos en el conjunto de pruebas de 443 elementos.

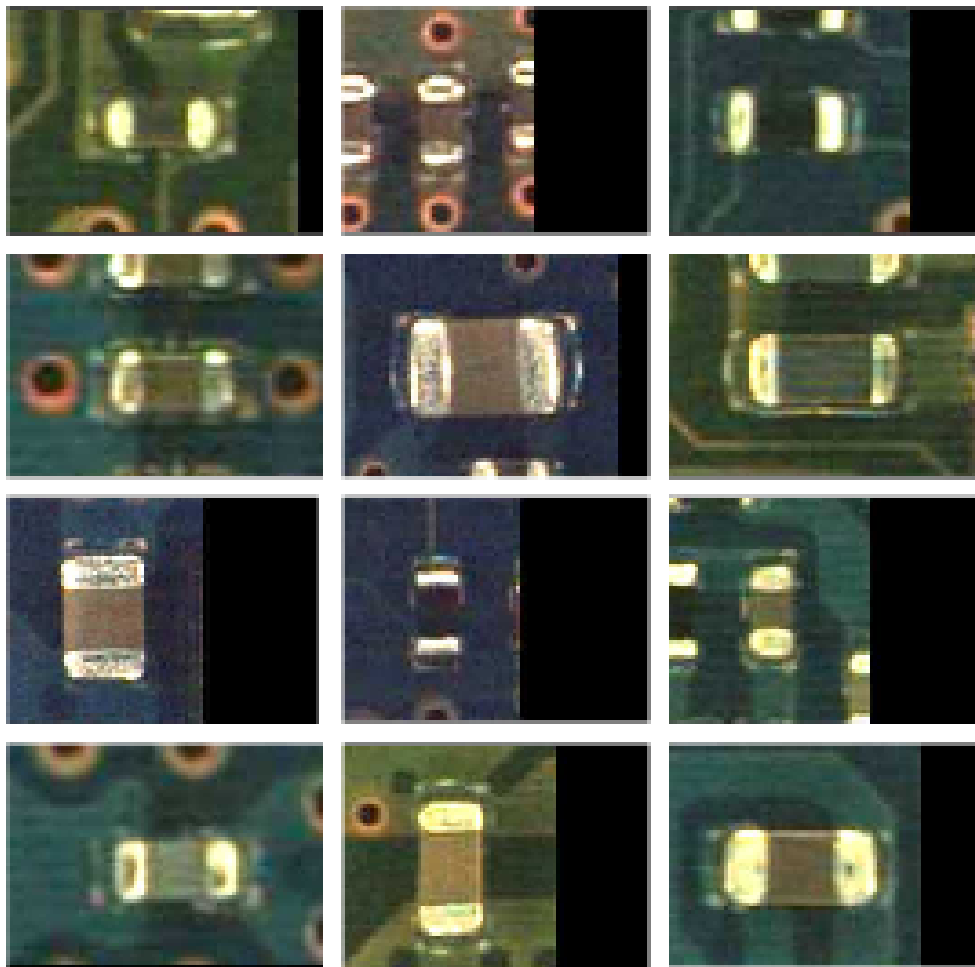


Figura 8.5: Componentes contenidos en el conjunto de pruebas de 945 elementos.



Figura 8.6: Soldaduras en el conjunto de pruebas de 945 elementos.

8.3. Comparación de características y resultados con otro método existente

Con este trabajo pretende proporcionar una alternativa a un problema específico de la industria. Por lo que esta implementación considera una clasificación correcta al caso de que elementos correctos se consideren incorrectos (falsas alarmas)³. Existen muchos trabajos relacionados, sin embargo bajo distintas características primarias y secundarias como lo son la calidad de las imágenes utilizadas.

Lin Shih-Chieh y Su Chia-Hsin⁴ presentan un sistema de inspección visual para dispositivos de montaje de superficie.

La razón por la que se hace la comparación con el trabajo de Shih-Chieh, es que realiza un tipo de inspección⁵ similar y el problema es parecido (detectar la falta un componente de tipo capacitores o resistencia).

La tabla 8.9 se presentan las características del problema y la manera en la que se le da solución. Se puede apreciar que las características del entorno y la funcionalidad propuesta son diferentes. Esto ocasiona una diferencia enorme entre los resultados obtenidos de cada trabajo, sin embargo estos se justifican por las mismas características.

La comparación de resultados obtenidos de acuerdo a cada análisis realizado por las implementaciones son mostrados en la tabla 8.10. El termino “NN-5-9-2” se refiere a la red neuronal de propagación hacia atrás con 5 entradas (conteo de pixeles blancos, regiones, histogramas, coeficientes de correlación y alto contraste), 9 nodos ocultos y 2 nodos de salida.

Para este análisis, las falsas alarmas se refieren a los componente que fueron clasificados como “incorrectos” siendo “correctos”. Por otro lado, una mala clasificación seria que elementos “incorrectos” se consideran como “correctos”.

Aunque los resultados de Shih-Chieh resultan más atractivos, hay recordar que trabaja con elementos con características de la imagen más deseables.

Con la combinación de los métodos aseguramos de manera importante la inspección, sin embargo se ven mucho más comprometidas las falsas alarmas. Esto se debe a que cada método clasifica de manera distinta a la imagen, de tal modo que con uno de ellos que detecte un defecto, se cataloga como incorrecta.

³Véase la sección 2.5

⁴La descripción de su trabajo está en la sección 2.8.

⁵Véase la sección 2.4.1.

Comparación de características			
	Característica	Shih-Chieh	Esta implementación
1	Tipo de inspección	Estructural	Estructural
2	Entorno	Está controlado	No está controlado
3	Escena de la imagen	Conocida	Desconocida
4	Calidad de la imagen	Constante	variable entre media y mala
5	Canales de color	1 (escala de grises)	3 (RGB)
6	Constancia en los colores de las imágenes	Constante	Muy variable
7	Posiciones validas de los componentes para el análisis	Se mantienen constantes entre muestras de un mismo tipo	Totalmente variables
8	Componentes analizados	Capacitores y resistencias en su mayoría	Capacitores y resistencias
9	Elementos de análisis	Textura de la imagen	Texturas de la imagen. Medidas de las formas en la imagen
10	Características analizadas	Conteo de pixeles blancos, regiones, histogramas, coeficientes de correlación y alto contraste	Momentos invariantes de las formas, cantidad de objetos en la imagen e histogramas de texturas.
11	Método de clasificación	Red neuronal (backpropagation)	Clasificador de distancia mínima
12	Respuesta	normal, faltante, invertido y desalineado	sin componente indefinido

Cuadro 8.9: Esta tabla muestra un a comparación de las características del problema y de la solución propuesta.

	Lin Shih-Chieh	Esta implementación			
Categoría	NN-5-9-2	Momentos invariantes	Contabilización de objetos	Texturas RGB	Combinación
Clasificación correcta	95 % +	95.71 %	40.40 %	97.96 %	100.00 %
Falsas alarmas	5 % -	48.75 %	40.40 %	29.34 %	90.96 %

Cuadro 8.10: Tabla comparativa de este sistema y del desarrollado por Lin Shih-Chieh

Capítulo 9

Conclusiones y trabajo futuro

9.1. Conclusiones

La cantidad de defectos presentados en tarjetas de montaje de superficie son bastante extensos, los cuales van desde la falta de un componente electrónico hasta defectos físicos de la tarjeta, por lo que sistemas de inspección actuales no son capaces de abarcar todos estos problemas. Estos sistemas reconocen desperfectos de manera general, sin determinarlo específicamente.

Este trabajo de investigación intenta resolver una pequeña parte de este problema, la cual consiste en determinar la presencia de un capacitor o resistencia en una región determinada de una tarjeta de montaje de superficie. Para esto, se realizó la implementación y combinaron métodos de análisis de imagen, entre los que se encuentran: aplicación de filtros, uso de umbrales, cálculo de características, empleo de texturas y redes neuronales.

El sistema implementado sólo trabaja con imágenes que contienen capacitores o resistencias, por lo que se necesita identificar previamente cuáles de estas son candidatas para este análisis. Para poder identificar el tipo de imagen, se implementa una red neuronal, la cual realizará la clasificación.

Los resultados de la implementación son aceptables considerando las características de la información de entrada. Hay que recordar el conjunto de imágenes de prueba utilizado es muy limitado y no representa toda la variedad de posibilidades que existen en la realidad.

Esta implementación es útil, ya que puede aminorar la carga de trabajo si se es implementado entre un sistema de inspección automática y la inspección visual humana.

Con la implementación y los resultados obtenidos, se pueden conocer los alcances y limitaciones de estos mismos para la solución del problema. Al conocer las limita-

ciones se puede saber las fallas de un procedimiento y mejorarlo.

Reafirmamos que no existe un método único capaz de resolver en su totalidad el problema de la identificación de componentes en la imagen de una tarjeta de montaje de superficie, por lo que es necesario implementar varias metodologías para mejorar la eficiencia en la solución.

Las principales contribuciones de este trabajo son: la descripción de procedimientos de análisis de imágenes que sirvan de base para el desarrollo de metodologías y la construcción de sistemas útiles para el control de calidad de productos; el desarrollo de una aplicación para la clasificación de imágenes y reconocimiento de objetos (componentes existentes e inexistentes).

La ejecución de esta implementación no requiere mucho tiempo de procesamiento, ni recursos de hardware y se conserva debajo del tiempo promedio requerido por la industria. Estos tiempos corresponden al proceso mecánico general correspondiente al ensamblaje.

9.2. Trabajo futuro

Se pueden probar otros métodos o mejorar los implementados, tanto en la clasificación de imágenes como el análisis e identificación de componentes, así como también utilizar un conjunto de imágenes de mejor calidad para alcanzar mejores resultados.

Existe mucho trabajo aún por desarrollar, ya que lo presentado es solo una parte de la solución general. En las siguientes subsecciones se mencionan cambios y mejoras que se pueden realizar en la clasificación y la identificación.

9.2.1. Clasificación

- Implementar una red con más de una capa oculta. El problema es que entre más nodos, mayor será el tiempo de computo.
- Implementar en la red neuronal el ajuste dinámico de el grado de aprendizaje η y el coeficiente de momento μ utilizados en la corrección de pesos, en base a la convergencia del error.
- Podría hacerse uso de varias características de los histogramas de la textura extraída (por ejemplo varianza, energía, correlación, entropía, contraste, homogeneidad, etc) y utilizarlos junto con la media de la textura para proporcionar

más información para la clasificación. Esto probablemente mejoraría los resultados de la clasificación notablemente. El problema es que por cada característica utilizada, se tendría que hacer uso de un conjunto más de nodos de la misma cardinalidad que la media.

- Utilizar los tres canales de color (RGB) para la extracción de características obtenidas de la suma y diferencia de histogramas de la textura. Se requeriría utilizar tres veces más nodos de entrada para la red neuronal.
- Para clasificar a través de las características de un histograma de suma y diferencia, hacer uso del clasificador Bayesiano en vez de la red neuronal.

9.2.2. Análisis e identificación

- Utilizar otro clasificador del error en vez o además del clasificador de distancia mínima, tal como el clasificador Bayesiano o una red neuronal. Estos dos últimos son más útiles cuando la respuesta puede tomar más de dos valores.
- Disponer de las características de los histogramas de suma y diferencia (media, varianza, contraste, etc.) junto con los métodos empleados.
- Extender el tipo de componentes analizados a condensadores y circuitos integrados, ya que el trabajo realizado proporciona las bases para ello.

Apéndice A

Imágenes de ejemplos de intersecciones de histogramas de brillo y contraste

Los histogramas de este apéndice pertenecen a las imágenes de la figura 6.8 del capítulo 6.

En todo los histogramas de contraste de las secciones definidas (superior, central e inferior) de las imágenes con capacitor y sin capacitor se puede apreciar una mínima frecuencia de variación de tonos (la concentración de puntos esta cerca del cero). Esto indica que cada pixel de las texturas analizadas pertenecientes a las secciones tiene un tono similar a sus pixeles vecinos. Por lo tanto todos los histogramas tienen la característica de que los colores de las texturas de sus secciones definidas son muy homogéneos (figuras A.1, A.2,A.3, A.4, A.5 y A.6).

La similitud de la homogeneidad que presentan las secciones es constatada por los histogramas resultantes de la intersección de histogramas de contraste, debido a que estos conservan una distribución de puntos similar a la de los histogramas base (figuras A.7, A.8, A.9 y A.10).

En el caso de los histogramas de brillo, los que pertenecen a las tres secciones de la imagen sin capacitor tienen una distribución de puntos similar entre sus canales RGB, como se muestra las figuras A.4, A.5, A.6. En la imagen con capacitor, los histogramas de brillo de la secciones superior e inferior tienen una distribución de puntos similar (figuras A.1 y A.3), mientras que la sección central tiene una distribución notoriamente distinta de las otras dos, como se ve en la figura A.2.

Los histogramas de brillo resultantes de la intersección de las secciones superior-central e inferior-central de la imagen sin resistencia muestran una cantidad de puntos de similitud evidente entre las secciones intersectadas (figuras A.9 y A.10), lo cual

afirma que existe parecido entre las zonas analizadas. En el caso de la imagen con capacitor, el histograma de brillo del canal B perteneciente a la intersección superior-central es el único que tiene elementos, mientras que los pertenecientes a los canales R y G están en blanco (figura A.7). En los histogramas de brillo de la sección inferior-central no existe ni un solo elemento que denote una similitud entre las secciones intersectadas (figura A.8). La escases o falta de elementos describe un nivel de semejanza pequeño o nulo, ya que no existen muchos puntos del histograma en común.

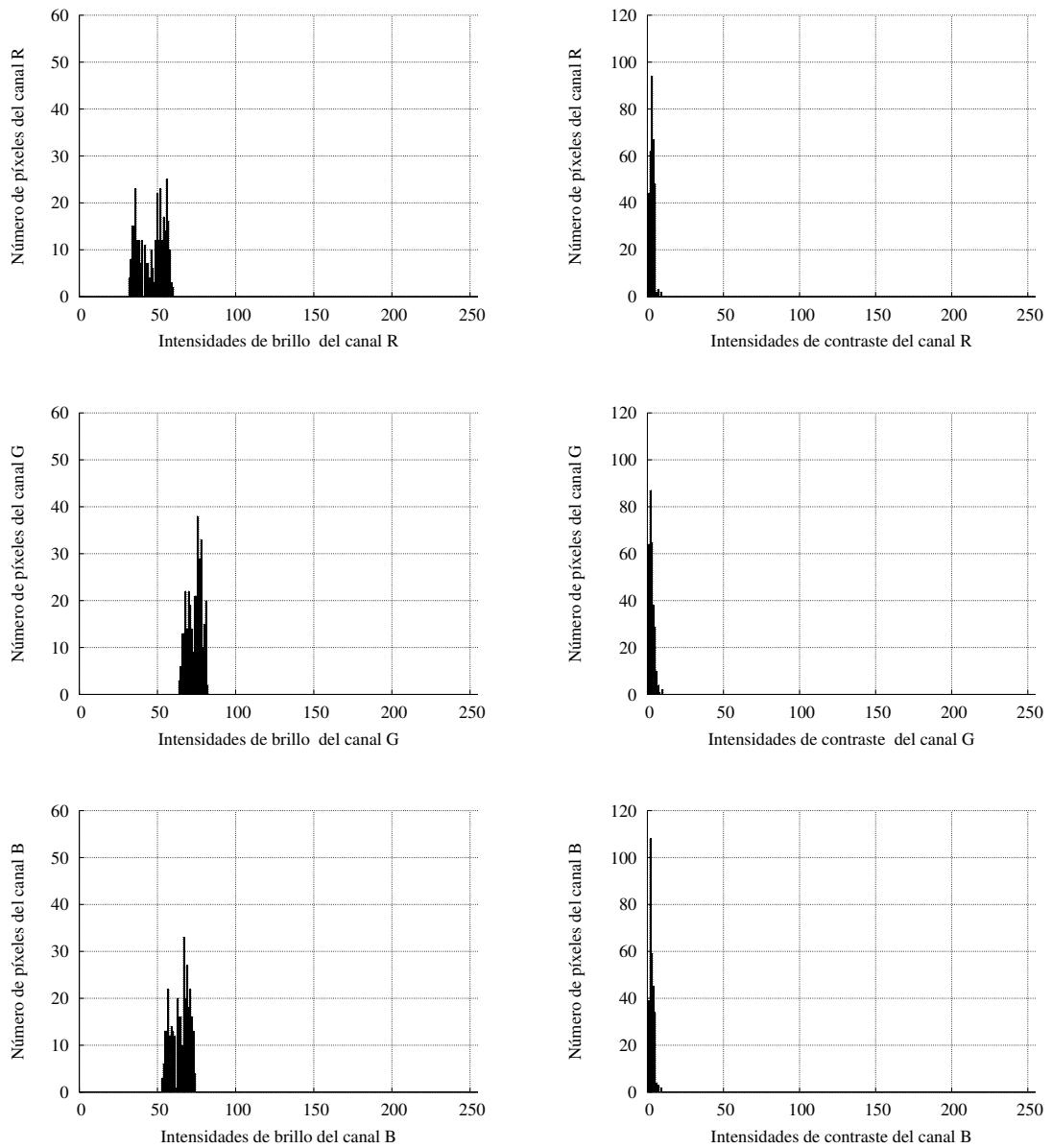


Figura A.1: Histogramas de brillo y contraste de los canales RGB de la sección superior de la imagen con capacitor.

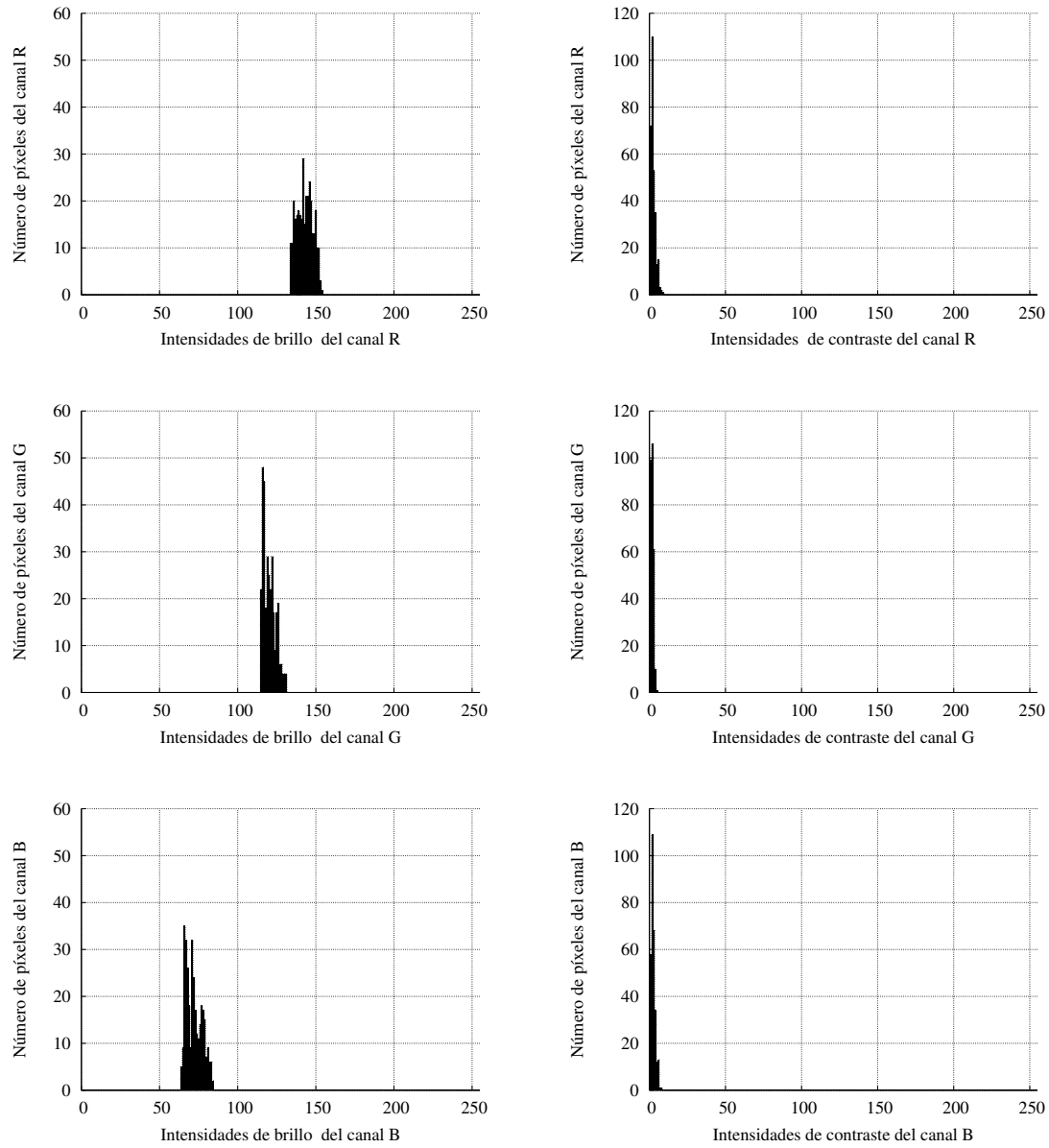


Figura A.2: Histogramas de brillo y contraste de los canales RGB de la sección central de la imagen con capacitor.

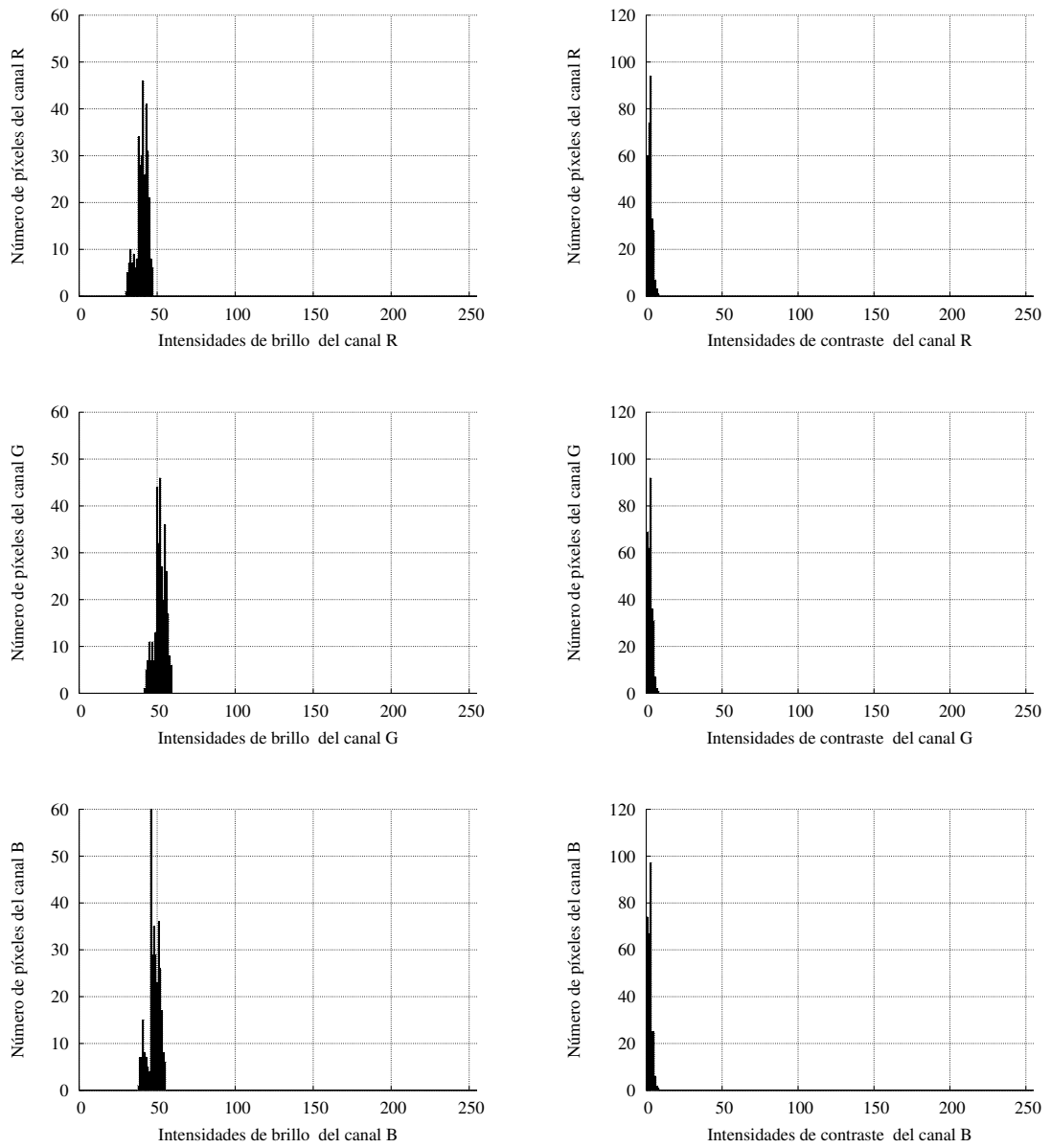


Figura A.3: Histogramas de brillo y contraste de los canales RGB de la sección inferior de la imagen con capacitor.

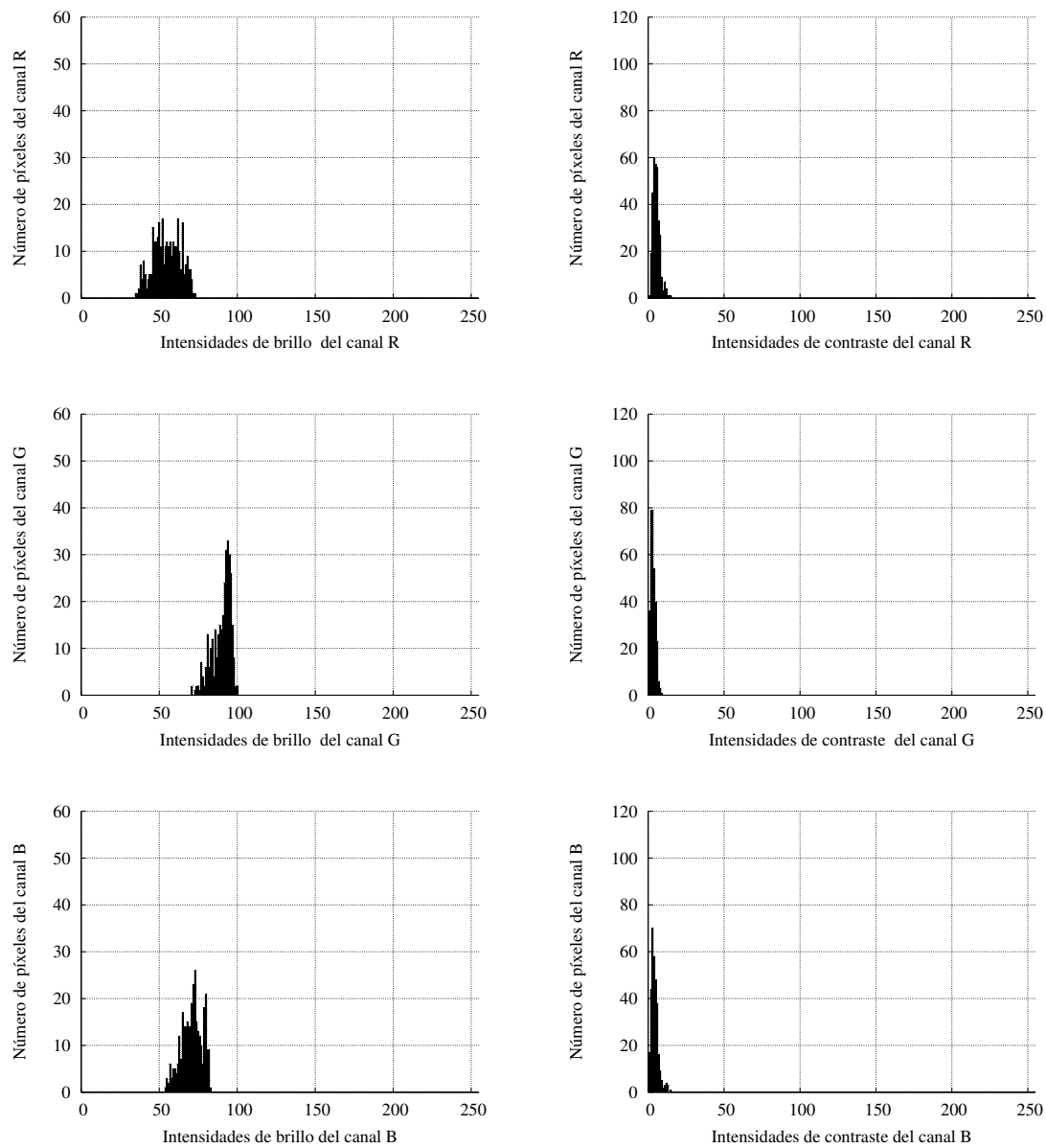


Figura A.4: Histogramas de brillo y contraste de los canales RGB de la sección superior de la imagen sin capacitor.

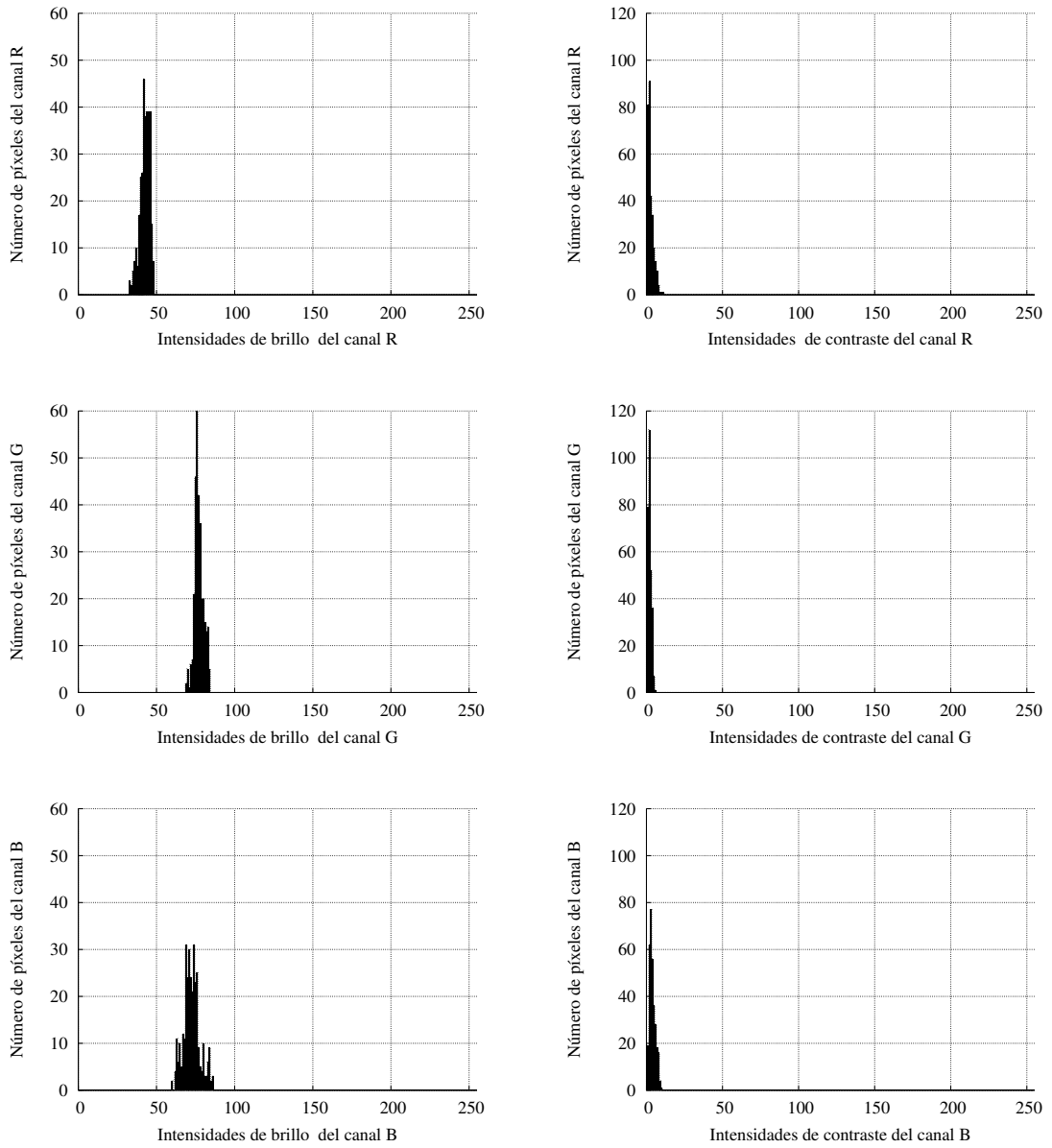


Figura A.5: Histogramas de brillo y contraste de los canales RGB de la sección central de la imagen sin capacitor.

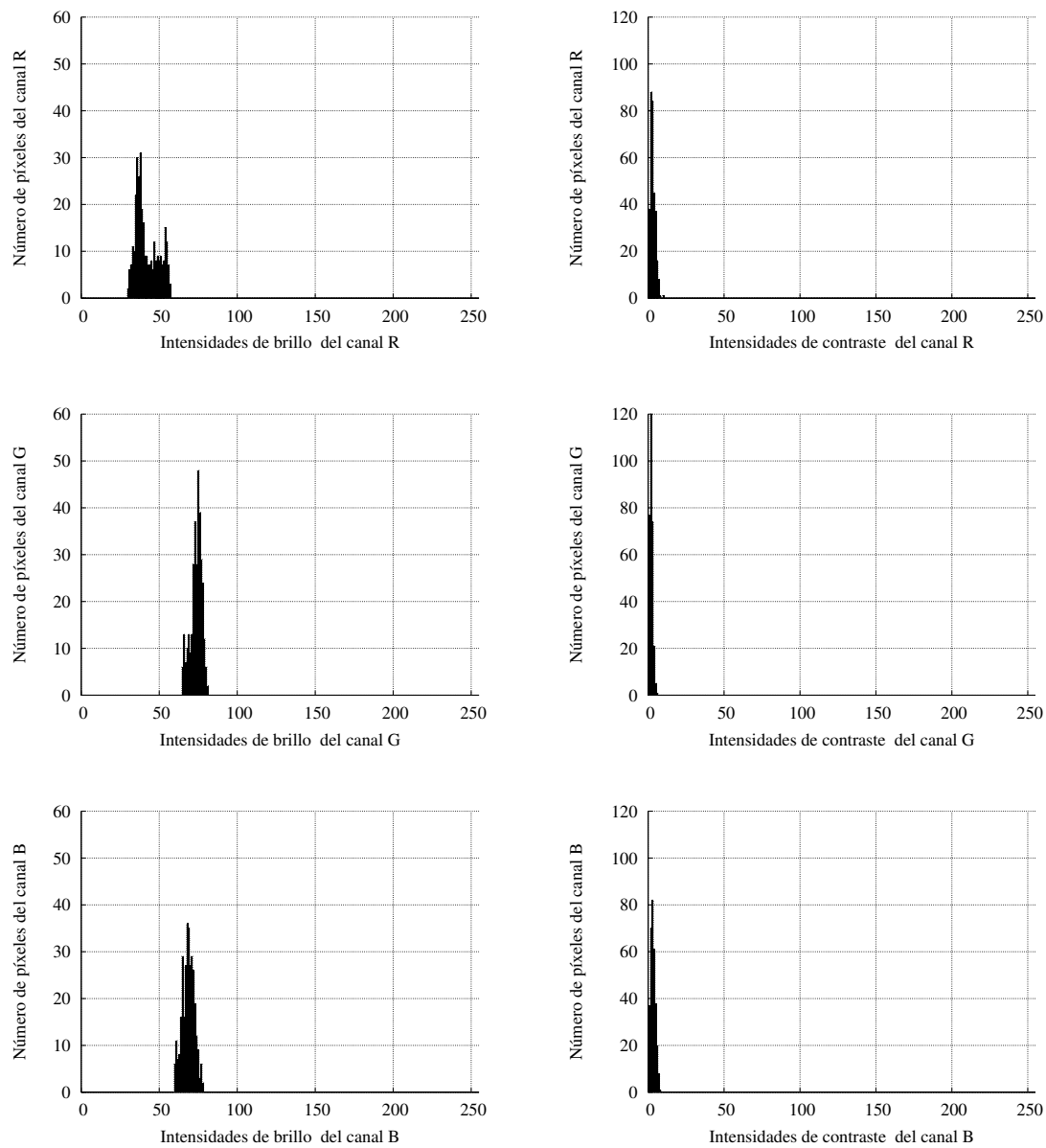


Figura A.6: Histogramas de brillo y contraste de los canales RGB de la sección inferior de la imagen con capacitor.

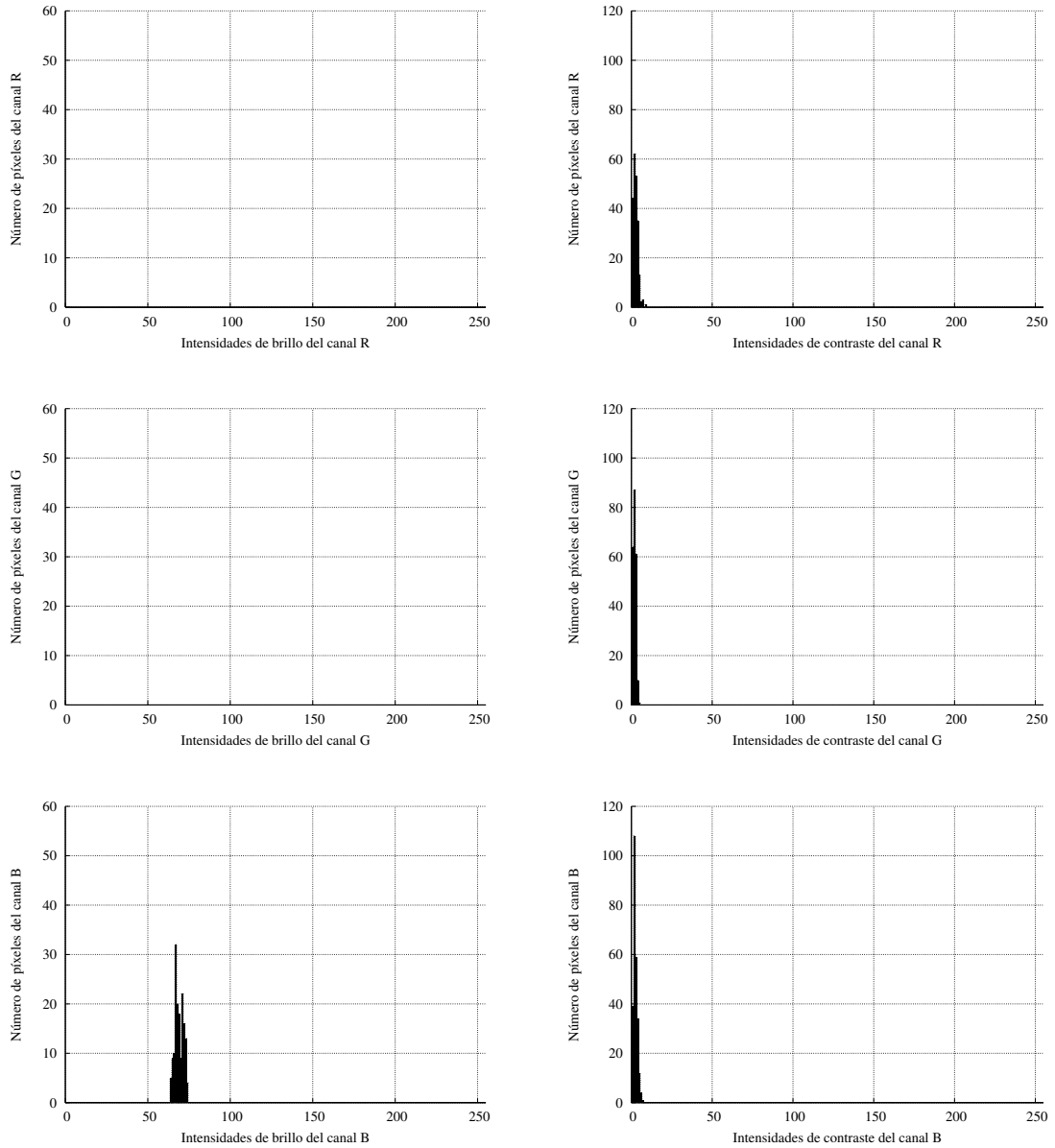


Figura A.7: Histogramas de brillo y contraste de los canales RGB resultantes de la intersección de los histogramas de la sección superior con la sección central de la imagen con capacitor.

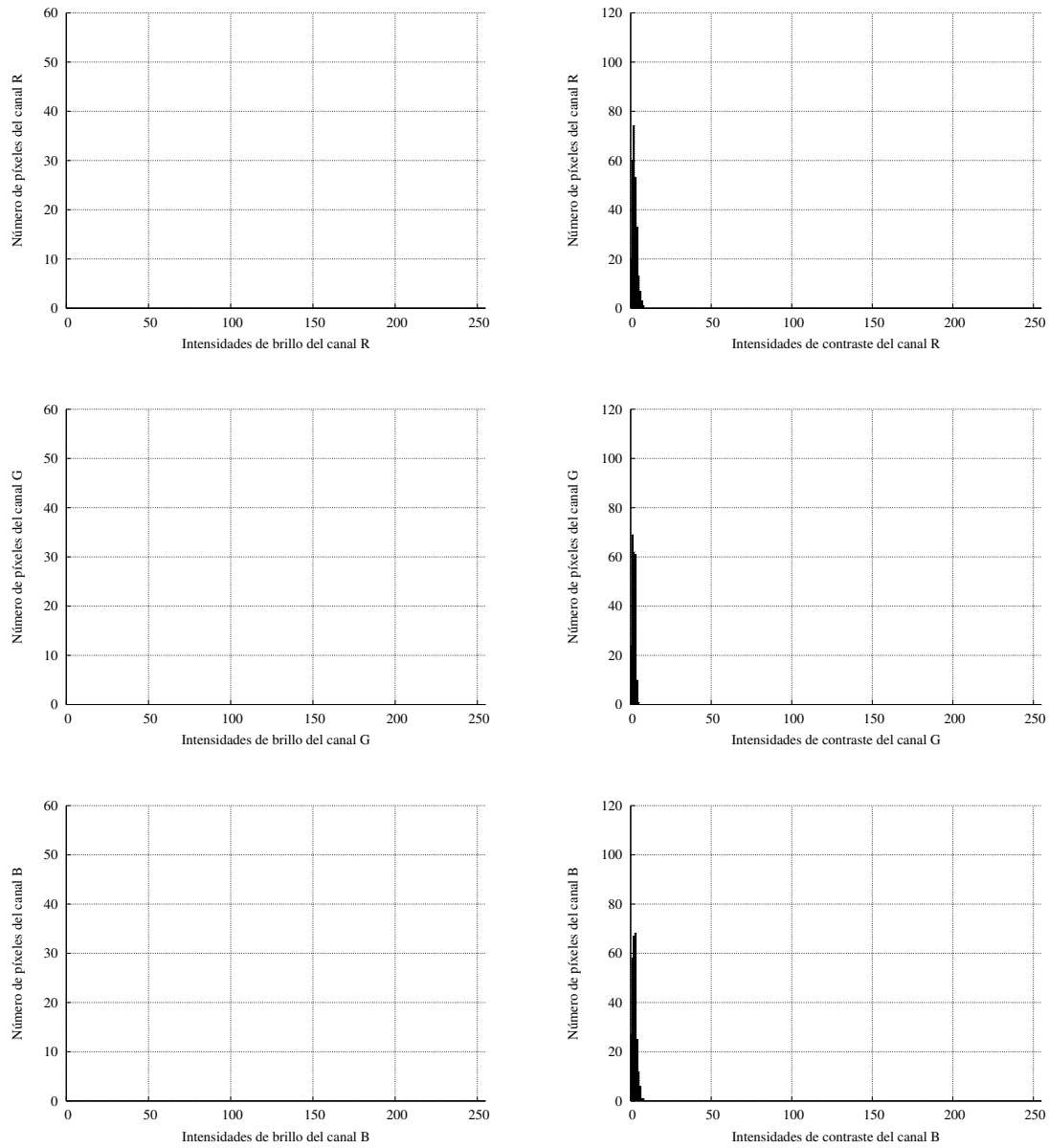


Figura A.8: Histogramas de brillo y contraste de los canales RGB resultantes de la intersección de los histogramas de la sección inferior y sección central de la imagen con capacitor.

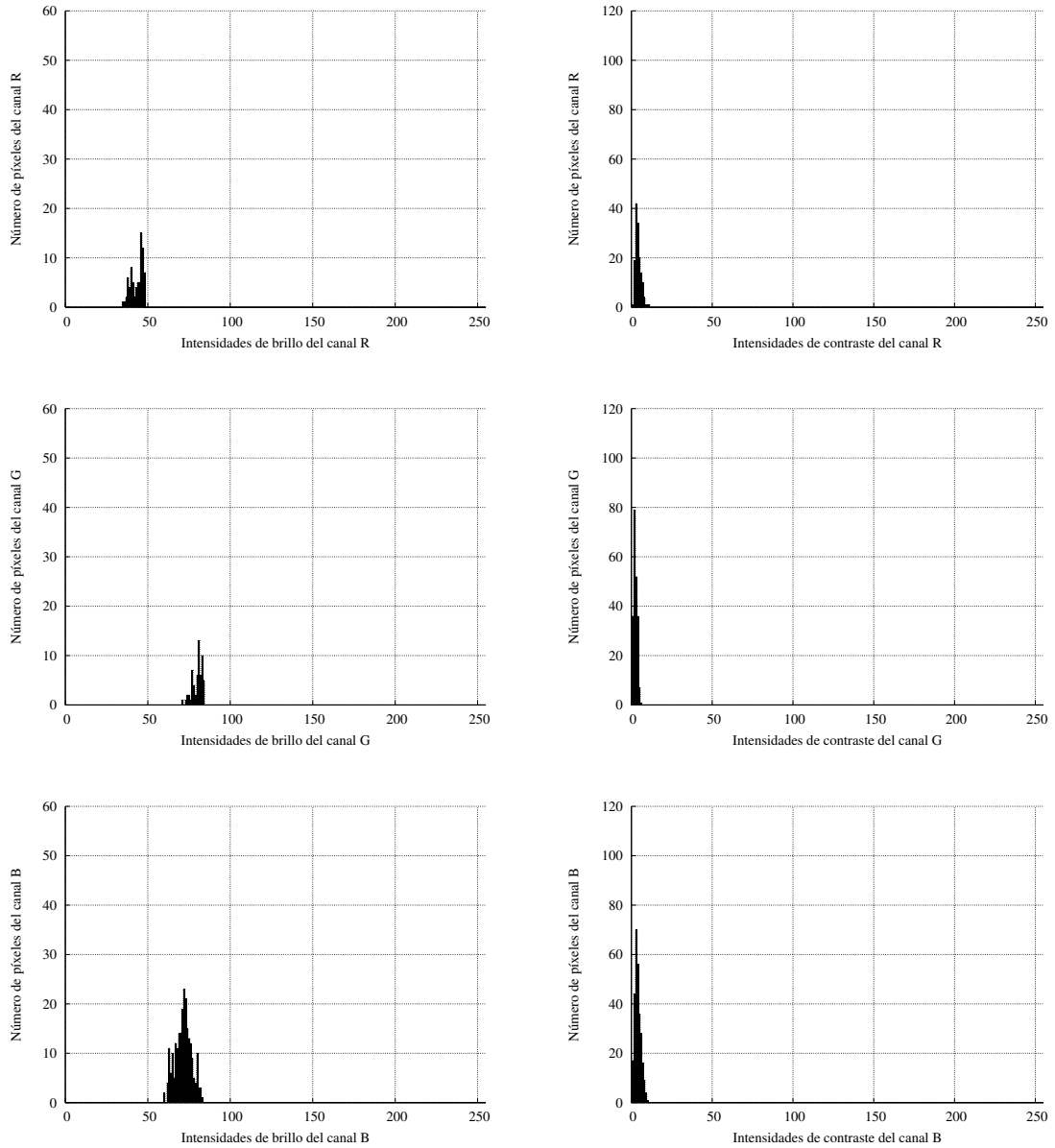


Figura A.9: Histogramas de brillo y contraste de los canales RGB resultantes de la intersección de los histogramas de la sección central y la sección superior de la imagen sin capacitor.

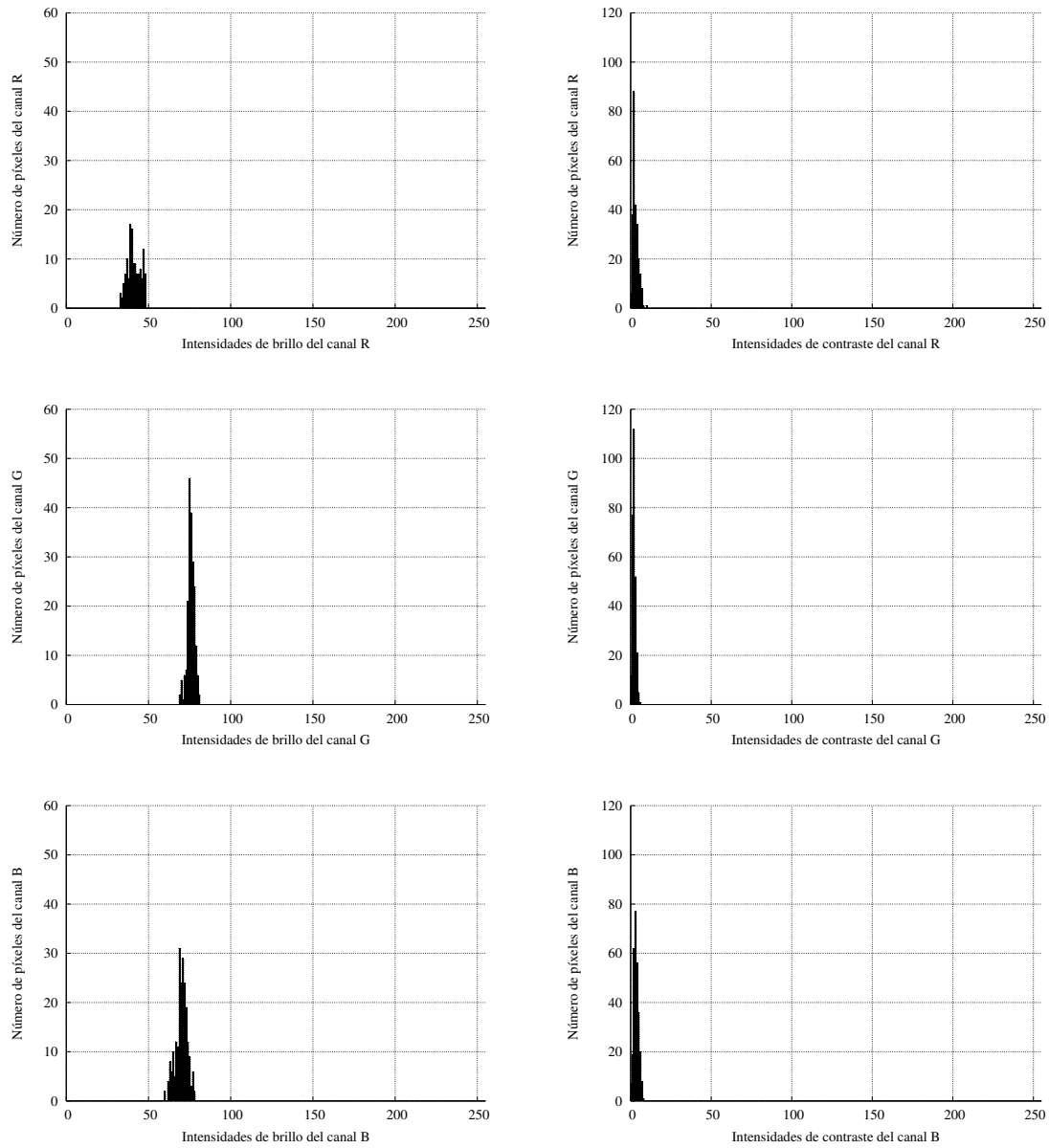


Figura A.10: Histogramas de brillo y contraste del RGB resultado de la intersección de los histogramas de la sección central con la sección inferior de la imagen sin capacitor.

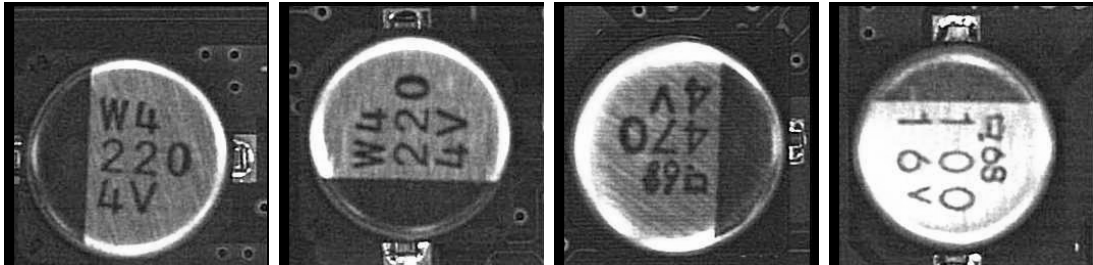
Apéndice B

Entrenamiento: clases definidas y comprobación

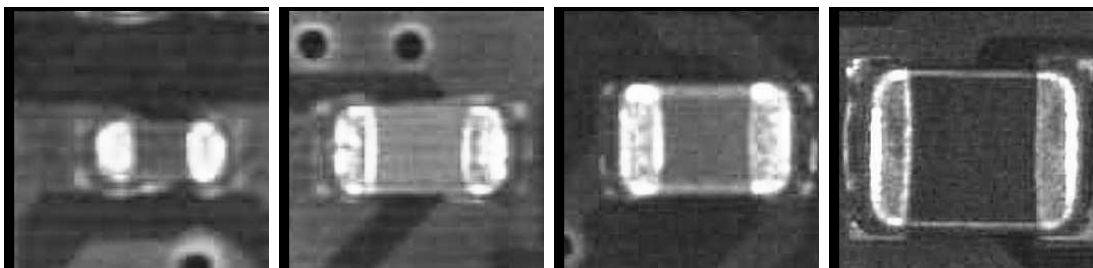
B.1. Clases definidas

Las 31 clases están definidas como:

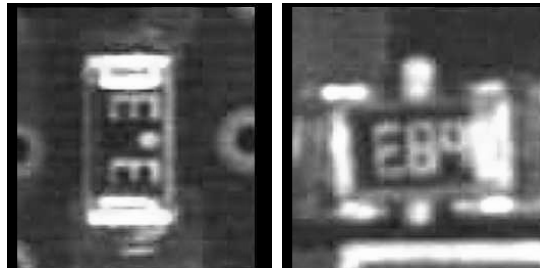
- Las clases 0, 1, 2 y 3 agrupan imágenes de condensadores rotados 0, 90, 180 y 270 grados respectivamente.



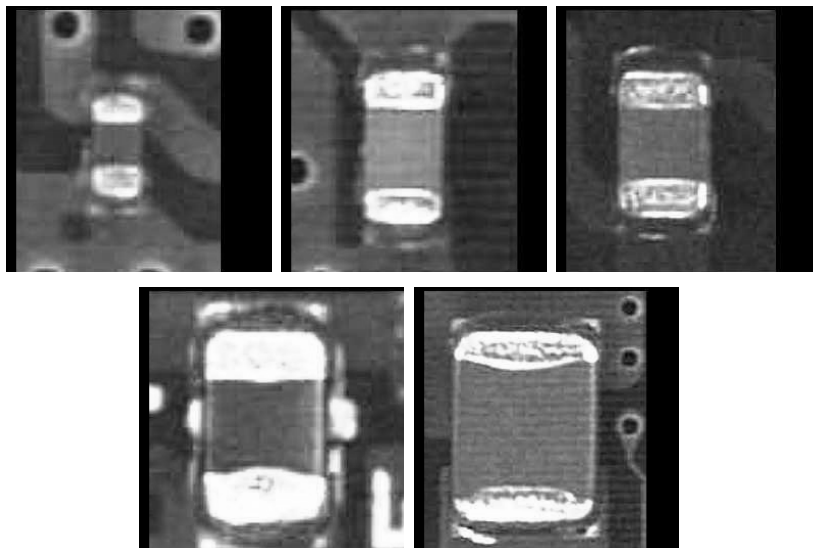
- Las clases 4, 5, 6 y 7 agrupan capacitores en posición horizontal en 4 tamaños diferentes, ordenados del más pequeño al más grande respectivamente.



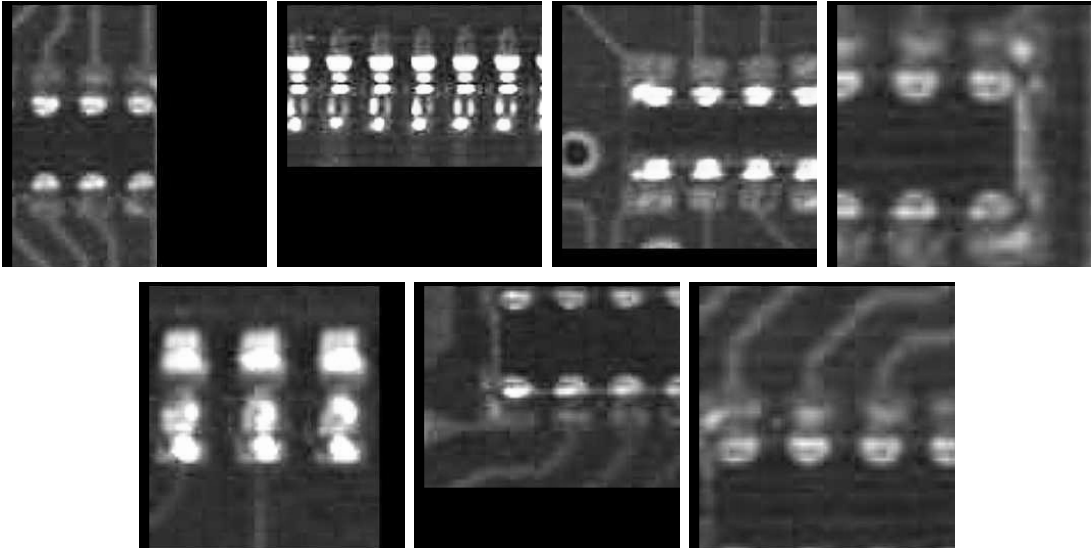
- Componentes rectangulares son agrupados como clase 8 (posición horizontal) y clase 14 (posición vertical), ambos contienen marcas.



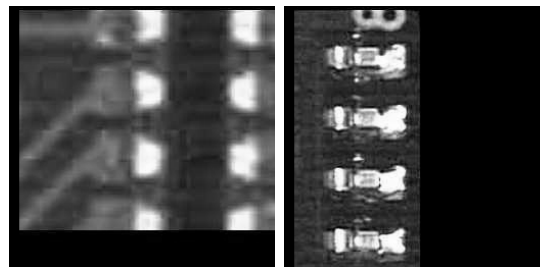
- Las clases 9, 10, 11, 12 y 13 son de capacitores en posición vertical de 5 tamaños distintos, ordenados del más pequeño al más grande respectivamente.



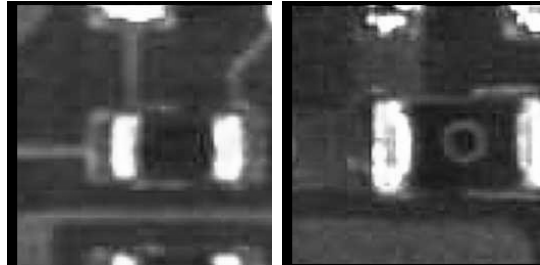
- Las clases 15, 16, 17, 18, 19, 20 y 21 agrupan circuitos integrados (o sus terminales) en el orden respectivo de izquierda a derecha..



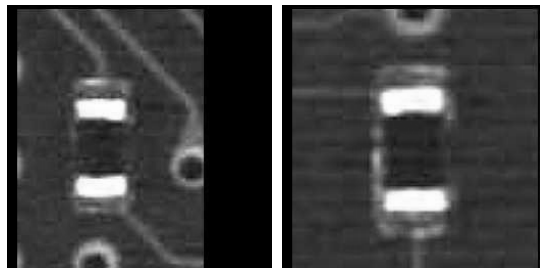
- Clases 22 y 23 son de terminales de circuitos integrados en posición vertical.



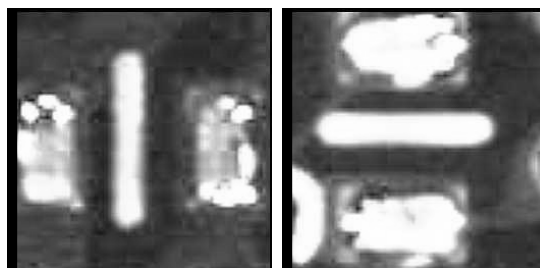
- Las clases 24 y 25 agrupan resistencias en posición horizontal en 2 tamaños diferentes, ordenados del más pequeño al más grande respectivamente.



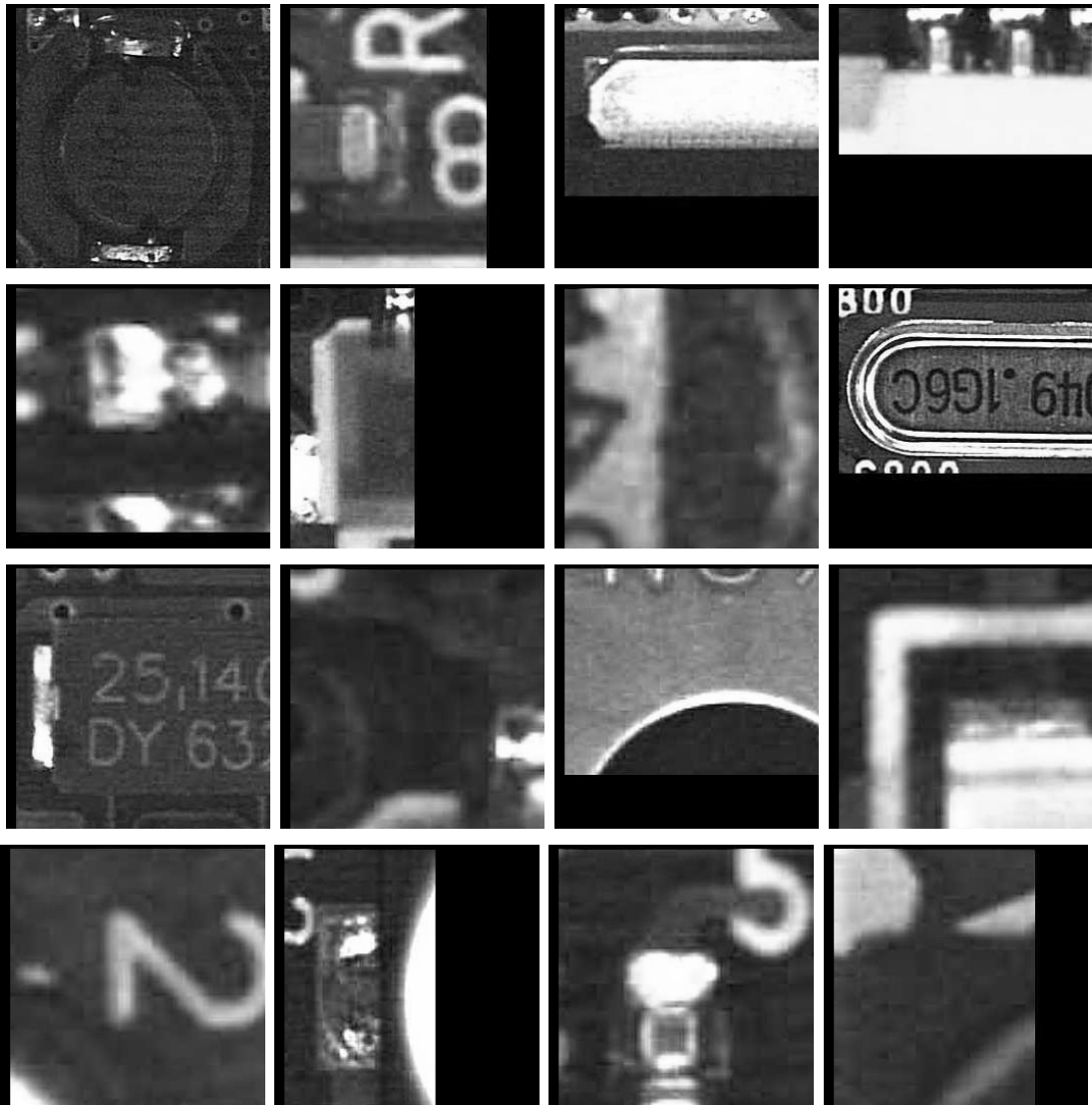
- Las clases 26 y 27 agrupan resistencias en posición vertical de 2 tamaños.



- Las clases 28 y 29 son de soldaduras para componentes rectangulares en posición vertical y horizontal.



- La clase 30 que agrupa imágenes sin definir como las mostradas a continuación.



B.2. Resultado del entrenamiento

B.2.1. Convergencia del error

El error mostrado en la tabla B.1, corresponde al error de convergencia de la solución por cada época del entrenamiento. Estos valores pertenecen a un ejemplo (ejecución del entrenamiento) exitosa.

Epoca	Error	Epoca	Error	Epoca	Error	Epoca	Error	Epoca	Error
0	0.107232	1	0.174469	2	0.165507	3	0.157589	4	0.150109
5	0.145216	6	0.140018	7	0.134492	8	0.127666	9	0.122041
10	0.117038	11	0.113057	12	0.109074	13	0.1053	14	0.102157
15	0.099317	16	0.094864	17	0.089928	18	0.085789	19	0.081864
20	0.076927	21	0.071827	22	0.067941	23	0.064311	24	0.062358
25	0.059302	26	0.056376	27	0.052205	28	0.048888	29	0.046174
30	0.044229	31	0.041793	32	0.040321	33	0.037668	34	0.036019
35	0.03401	36	0.032502	37	0.029641	38	0.026633	39	0.025163
40	0.02454	41	0.024211	42	0.025082	43	0.025148	44	0.025472
45	0.026261	46	0.027248	47	0.02696	48	0.026771	49	0.026556
50	0.026167	51	0.02441	52	0.023933	53	0.023738	54	0.023583
55	0.02344	56	0.023303	57	0.023163	58	0.02299	59	0.022665
60	0.022017	61	0.021579	62	0.020695	63	0.020417	64	0.020313
65	0.020222	66	0.020128	67	0.020029	68	0.019922	69	0.0198
70	0.019617	71	0.018795	72	0.018273	73	0.017941	74	0.016901
75	0.016755	76	0.016644	77	0.016477	78	0.015848	79	0.01525
80	0.015147	81	0.015079	82	0.015018	83	0.01496	84	0.014902
85	0.014846	86	0.014791	87	0.014735	88	0.01468	89	0.014628
90	0.014576	91	0.014527	92	0.014479	93	0.014431	94	0.014379
95	0.01432	96	0.014242	97	0.014086	98	0.013504	99	0.010491
100	0.009609	101	0.009132	102	0.008991	103	0.008751	104	0.008096
105	0.007767	106	0.007676	107	0.007619	108	0.007571	109	0.007522
110	0.007468	111	0.007383	112	0.00679	113	0.006113	114	0.006073
115	0.006041	116	0.006013	117	0.005986	118	0.005958	119	0.005931
120	0.005905	121	0.005879	122	0.005853	123	0.005829	124	0.005805
125	0.005781	126	0.005757	127	0.005732	128	0.005707	129	0.005681
130	0.005651	131	0.00561	132	0.004944	133	0.003952	134	0.003
135	0.002963	136	0.002944	137	0.002926	138	0.00291	139	0.002894
140	0.002878	141	0.002862	142	0.002847	143	0.002833	144	0.002819
145	0.002805	146	0.002791	147	0.002778	148	0.002765	149	0.002751
150	0.002738	151	0.002725	152	0.002712	153	0.002699	154	0.002687
155	0.002674	156	0.002662	157	0.002649	158	0.002637	159	0.002624
160	0.002613	161	0.002601	162	0.00259	163	0.002579	164	0.002568
165	0.002557	166	0.002547	167	0.002536	168	0.002526	169	0.002515
170	0.002505	171	0.002495	172	0.002486	173	0.002476	174	0.002466
175	0.002456	176	0.002446	177	0.002436	178	0.002426	179	0.002417

Cuadro B.1: Error de convergencia por época.

Epoca	Error	Epoca	Error	Epoca	Error	Epoca	Error	Epoca	Error
180	0.002409	181	0.0024	182	0.002391	183	0.002383	184	0.002374
185	0.002366	186	0.002358	187	0.00235	188	0.002342	189	0.002335
190	0.002327	191	0.002319	192	0.002312	193	0.002304	194	0.002296
195	0.002289	196	0.002282	197	0.002275	198	0.002267	199	0.00226
200	0.002254	201	0.002247	202	0.002241	203	0.002234	204	0.002228
205	0.002222	206	0.002215	207	0.002209	208	0.002203	209	0.002197
210	0.002191	211	0.002185	212	0.002179	213	0.002173	214	0.002167
215	0.002162	216	0.002156	217	0.002151	218	0.002146	219	0.002141
220	0.002135	221	0.00213	222	0.002126	223	0.002121	224	0.002116
225	0.002111	226	0.002107	227	0.002102	228	0.002098	229	0.002093
230	0.002089	231	0.002085	232	0.00208	233	0.002076	234	0.002073
235	0.002069	236	0.002065	237	0.002061	238	0.002057	239	0.002054
240	0.00205	241	0.002047	242	0.002043	243	0.00204	244	0.002037
245	0.002033	246	0.00203	247	0.002027	248	0.002024	249	0.002021
250	0.002018	251	0.002015	252	0.002012	253	0.002009	254	0.002006
255	0.002003	256	0.002	257	0.001997	258	0.001995	259	0.001992
260	0.00199	261	0.001988	262	0.001985	263	0.001983	264	0.00198
265	0.001978	266	0.001975	267	0.001973	268	0.00197	269	0.001968
270	0.001965	271	0.001963	272	0.001961	273	0.001958	274	0.001956
275	0.001953	276	0.001951	277	0.001948	278	0.001946	279	0.001944
280	0.001941	281	0.001939	282	0.001936	283	0.001934	284	0.001932
285	0.001929	286	0.001927	287	0.001925	288	0.001924	289	0.001922
290	0.00192	291	0.001918	292	0.001916	293	0.001914	294	0.001912
295	0.001911	296	0.001909	297	0.001907	298	0.001905	299	0.001904
300	0.001902	301	0.0019	302	0.001898	303	0.001896	304	0.001894
305	0.001892	306	0.00189	307	0.001888	308	0.001887	309	0.001885
310	0.001883	311	0.001882	312	0.00188	313	0.001878	314	0.001876
315	0.001875	316	0.001873	317	0.001871	318	0.001869	319	0.001868
320	0.001866	321	0.001864	322	0.001862	323	0.001861	324	0.001859
325	0.001858	326	0.001856	327	0.001855	328	0.001853	329	0.001852
330	0.00185	331	0.001849	332	0.001847	333	0.001846	334	0.001845
335	0.001844	336	0.001842	337	0.001841	338	0.00184	339	0.001839
340	0.001837	341	0.001836	342	0.001835	343	0.001834	344	0.001833
345	0.001832	346	0.001831	347	0.00183	348	0.001829	349	0.001827
350	0.001826	351	0.001825	352	0.001824	353	0.001823	354	0.001822
355	0.001821	356	0.00182	357	0.001819	358	0.001818	359	0.001817
360	0.001816	361	0.001815	362	0.001813	363	0.001812	364	0.001811
365	0.00181	366	0.001809	367	0.001808	368	0.001806	369	0.001806
370	0.001804	371	0.001803	372	0.001802	373	0.001801	374	0.0018
375	0.001799	376	0.001798	377	0.001797	378	0.001796	379	0.001795
380	0.001794	381	0.001793	382	0.001792	383	0.001791	384	0.00179
385	0.001789	386	0.001788	387	0.001786	388	0.001786	389	0.001785
390	0.001784	391	0.001783	392	0.001782	393	0.001781	394	0.00178
395	0.001779	396	0.001778	397	0.001777	398	0.001776	399	0.001775
400	0.001774	401	0.001773	402	0.001772	403	0.001771	404	0.00177
405	0.00177	406	0.001769	407	0.001768	408	0.001767	409	0.001766

Cuadro B.1: Error de convergencia por época.

Epoca	Error	Epoca	Error	Epoca	Error	Epoca	Error	Epoca	Error
410	0.001765	411	0.001764	412	0.001764	413	0.001763	414	0.001762
415	0.001761	416	0.00176	417	0.001759	418	0.001759	419	0.001758
420	0.001757	421	0.001756	422	0.001755	423	0.001754	424	0.001754
425	0.001753	426	0.001752	427	0.001751	428	0.00175	429	0.001749
430	0.001748	431	0.001747	432	0.001746	433	0.001745	434	0.001744
435	0.001744	436	0.001743	437	0.001741	438	0.001741	439	0.00174
440	0.001739	441	0.001738	442	0.001737	443	0.001736	444	0.001735
445	0.001734	446	0.001733	447	0.001732	448	0.001731	449	0.00173
450	0.001729	451	0.001728	452	0.001727	453	0.001726	454	0.001725
455	0.001723	456	0.001723	457	0.001722	458	0.001721	459	0.00172
460	0.001719	461	0.001718	462	0.001717	463	0.001716	464	0.001715
465	0.001713	466	0.001713	467	0.001711	468	0.001711	469	0.00171
470	0.001709	471	0.001708	472	0.001707	473	0.001706	474	0.001705
475	0.001704	476	0.001703	477	0.001702	478	0.001701	479	0.0017
480	0.0017	481	0.001699	482	0.001698	483	0.001697	484	0.001697
485	0.001696	486	0.001695	487	0.001694	488	0.001693	489	0.001692
490	0.001691	491	0.001691	492	0.001689	493	0.001688	494	0.001688
495	0.001686	496	0.001685	497	0.001684	498	0.001683	499	0.001682
500	0.001681	501	0.00168	502	0.001679	503	0.001678	504	0.001677
505	0.001676	506	0.001675	507	0.001674	508	0.001673	509	0.001671
510	0.001671	511	0.001669	512	0.001668	513	0.001667	514	0.001666
515	0.001666	516	0.001665	517	0.001664	518	0.001662	519	0.001662
520	0.00166	521	0.00166	522	0.001658	523	0.001658	524	0.001656
525	0.001655	526	0.001654	527	0.001653	528	0.001652	529	0.001651
530	0.001649	531	0.001647	532	0.001645	533	0.00164	534	0.001617
535	0.000318								

Cuadro B.1: Error de convergencia por época.

B.2.2. Resultados de la comprobación del conjunto de entrenamiento

Estos resultados son el producto de la comprobación (análisis) del conjunto de imágenes de entrenamiento, una vez que la red ya fue entrenada. Esto es una forma de comprobar que la red fue adiestrada de manera correcta, ya que de no clasificar correctamente este conjunto es evidente que hubo un error en el entrenamiento.

Los datos presentados en la tabla ?? pertenecen a la ejecución de un entrenamiento exitoso

Patrón	Clase	Valor	Patrón	Clase	Valor	Patrón	Clase	Valor	Patrón	Clase	Valor
0	0	0.972018	1	0	0.983260	2	0	0.980697	3	0	0.921528
4	1	0.982343	5	1	0.974887	6	1	0.995764	7	1	0.974922
8	1	0.978432	9	2	0.967429	10	2	0.994699	11	2	0.991790
12	2	0.950680	13	2	0.979659	14	2	0.985246	15	2	0.932314
16	3	0.996654	17	3	0.997902	18	3	0.989867	19	3	0.970545
20	3	0.982387	21	3	0.965489	22	3	0.960605	23	4	0.999819
24	4	0.981409	25	4	0.996601	26	4	0.998726	27	4	0.997413
28	4	0.986011	29	4	0.999625	30	4	0.981441	31	4	0.986228
32	4	0.999501	33	4	0.988603	34	4	0.991813	35	4	0.967699
36	4	0.968758	37	4	0.999112	38	4	0.977074	39	4	0.975552
40	4	0.993529	41	4	0.999935	42	4	0.959025	43	5	0.996117
44	5	0.990125	45	5	0.974135	46	5	0.999895	47	5	0.999895
48	5	0.968865	49	5	0.969289	50	5	0.999217	51	5	0.986043
52	5	0.996925	53	5	0.974098	54	5	0.985740	55	5	0.979112
56	5	0.961676	57	5	0.978971	58	5	0.972795	59	5	0.988754
60	5	0.964241	61	5	0.965933	62	6	0.997060	63	6	0.981149
64	6	0.996928	65	6	0.969601	66	6	0.997787	67	7	0.980048
68	7	0.987747	69	7	0.986323	70	7	0.988618	71	7	0.978932
72	7	0.988018	73	7	0.986951	74	7	0.964099	75	7	0.998083
76	8	0.961644	77	8	0.993386	78	8	0.998951	79	8	0.999956
80	8	0.976581	81	8	0.959725	82	8	0.971459	83	8	0.981908
84	8	0.969749	85	8	0.999289	86	8	0.981738	87	8	0.962935
88	9	0.984674	89	9	0.989490	90	9	0.970999	91	9	0.998773
92	9	0.997680	93	9	0.999910	94	9	0.997656	95	9	0.991063
96	9	0.969257	97	9	0.987169	98	9	0.999867	99	9	0.987313
100	9	0.995942	101	9	0.998962	102	9	0.970155	103	9	0.992782
104	9	0.998165	105	9	0.991227	106	9	0.999509	107	9	0.999509
108	10	0.998991	109	10	0.999641	110	10	0.996822	111	10	0.998926
112	10	0.992289	113	10	0.997647	114	10	0.994605	115	10	0.987941
116	10	0.996517	117	10	0.998416	118	10	0.990115	119	10	0.977574
120	10	0.983409	121	11	0.986601	122	11	0.984065	123	11	0.977535
124	11	0.984213	125	11	0.984213	126	11	0.987957	127	11	0.985078
128	11	0.985080	129	11	0.996430	130	12	0.970751	131	12	0.989941
132	12	0.981856	133	12	0.990076	134	12	0.981856	135	13	0.985383

Cuadro B.2: Resultado de la clasificación del conjunto de entrenamiento.

Patrón	Clase	Valor	Patrón	Clase	Valor	Patrón	Clase	Valor	Patrón	Clase	Valor
136	13	0.992178	137	13	0.962024	138	13	0.968470	139	13	0.968484
140	13	0.985383	141	13	0.970640	142	13	0.971236	143	13	0.929767
144	14	0.976738	145	14	0.975980	146	14	0.976398	147	14	0.966176
148	14	0.995007	149	15	0.989886	150	15	0.998815	151	15	0.969896
152	15	0.995714	153	15	0.998794	154	15	0.991156	155	15	0.999866
156	15	0.969418	157	15	0.974294	158	15	0.992431	159	15	0.965513
160	16	0.999271	161	16	0.999662	162	16	0.974667	163	16	0.978451
164	16	0.990033	165	16	0.998608	166	16	0.998467	167	16	0.981795
168	16	0.973153	169	16	0.966455	170	16	0.986888	171	17	0.986822
172	17	0.990482	173	17	0.975787	174	17	0.976702	175	17	0.973736
176	17	0.955149	177	17	0.973736	178	18	0.984419	179	18	0.974020
180	18	0.976555	181	19	0.953857	182	19	0.974422	183	19	0.999943
184	19	0.999111	185	19	0.992091	186	19	0.983369	187	19	0.964735
188	19	0.926732	189	19	0.935853	190	19	0.968872	191	19	0.994501
192	20	0.974325	193	20	0.987964	194	20	0.976973	195	21	0.976526
196	21	0.963371	197	21	0.992015	198	21	0.976526	199	21	0.961293
200	21	0.990713	201	22	0.996323	202	22	0.964720	203	22	0.998705
204	22	0.983354	205	22	0.936968	206	23	0.971919	207	23	0.972731
208	23	0.994062	209	23	0.994062	210	23	0.979908	211	23	0.978496
212	24	0.953637	213	24	0.993484	214	24	0.926886	215	24	0.858501
216	24	0.995010	217	24	0.908573	218	24	0.998278	219	24	0.988002
220	24	0.872737	221	24	0.966474	222	24	0.729743	223	24	0.869612
224	24	0.999515	225	24	0.994599	226	24	0.996615	227	24	0.969519
228	24	0.993836	229	25	0.990822	230	25	0.975761	231	25	0.989464
232	25	0.992188	233	25	0.996062	234	25	0.967976	235	25	0.974875
236	25	0.987382	237	26	0.993736	238	26	0.998987	239	26	0.997510
240	26	0.999960	241	26	0.999185	242	26	0.997174	243	26	0.996690
244	26	0.998055	245	26	0.982521	246	26	0.977687	247	26	0.999888
248	26	0.978823	249	26	0.997986	250	26	0.986640	251	26	0.999170
252	26	0.999790	253	26	0.978650	254	27	0.992824	255	27	0.996995
256	27	0.996754	257	27	0.990184	258	27	0.953668	259	27	0.995589
260	27	0.979679	261	27	0.977307	262	27	0.953636	263	27	0.980785
264	27	0.977864	265	27	0.880884	266	28	0.991189	267	28	0.799943
268	29	0.936199	269	30	0.936039	270	30	1.000000	271	30	1.000000
272	30	1.000000	273	30	0.999889	274	30	1.000000	275	30	1.000000
276	30	1.000000	277	30	1.000000	278	30	0.999999	279	30	1.000000
280	30	1.000000	281	30	1.000000	282	30	0.999500	283	30	1.000000
284	30	0.999159	285	30	0.999996	286	30	0.999967	287	30	0.999983
288	30	1.000000	289	30	0.999102	290	30	0.999515	291	30	0.999971
292	30	1.000000	293	30	0.999783	294	30	0.996115	295	30	1.000000
296	30	1.000000	297	30	0.999916	298	30	0.999832	299	30	0.999972
300	30	0.999963	301	30	0.999760	302	30	1.000000	303	30	0.999825
304	30	0.999991	305	30	0.999999	306	30	0.999981	307	30	0.999725
308	30	0.997209	309	30	0.999782	310	30	0.999195	311	30	0.998879
312	30	0.998194	313	30	0.999999	314	30	0.999863	315	30	0.999999
316	30	0.999973	317	30	0.993079	318	30	0.999746	319	30	0.999995

Cuadro B.2: Resultado de la clasificación del conjunto de entrenamiento.

Patrón	Clase	Valor	Patrón	Clase	Valor	Patrón	Clase	Valor	Patrón	Clase	Valor
320	30	0.998955	321	30	0.999999	322	30	0.999854	323	30	0.999829
324	30	0.999853	325	30	0.999697	326	30	0.999489	327	30	0.999878
328	30	0.999977	329	30	0.993435	330	30	1.000000	331	30	0.999983
332	30	0.999927	333	30	0.999930	334	30	0.999702	335	30	0.997472
336	30	0.989350	337	30	1.000000	338	30	0.998281	339	30	0.999977
340	30	0.999670	341	30	0.995055	342	30	0.994180	343	30	0.993959
344	30	0.999768	345	30	0.995836	346	30	0.995160	347	30	0.999975
348	30	0.999515	349	30	0.998148	350	30	0.998053	351	30	0.997390
352	30	0.992987	353	30	0.998204	354	30	0.997013	355	30	0.997971
356	30	0.986861	357	30	0.994327	358	30	0.999616	359	30	0.998708
360	30	0.999332	361	30	0.999999	362	30	0.993117	363	30	0.995966
364	30	0.999710	365	30	0.993790	366	30	0.976208	367	30	0.992743
368	30	0.999458	369	30	0.998384	370	30	0.981789	371	30	0.986861
372	30	0.995703	373	30	0.972896	374	30	0.992502	375	30	0.997838
376	30	0.989789	377	30	0.999993	378	30	0.995544	379	30	0.997575
380	30	0.998934									

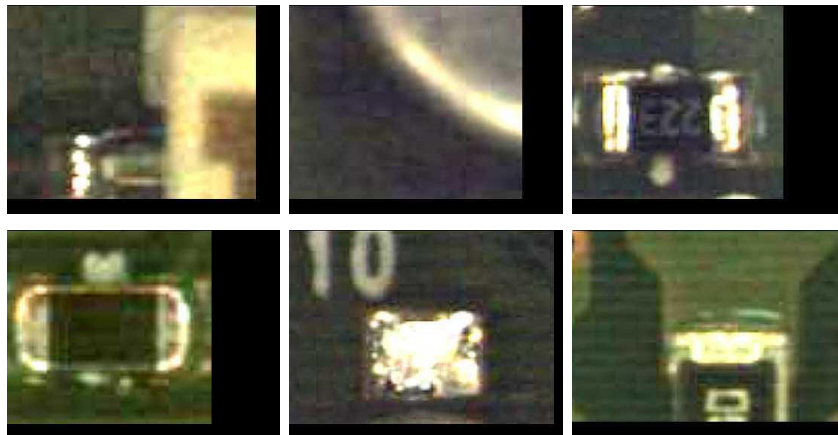
Cuadro B.2: Resultado de la clasificación del conjunto de entrenamiento.

Apéndice C

Imágenes clasificadas erróneamente

En la clasificación se catalogaron erróneamente imágenes en clases a las que no pertenecen. Algunas de estas imágenes son las siguientes:

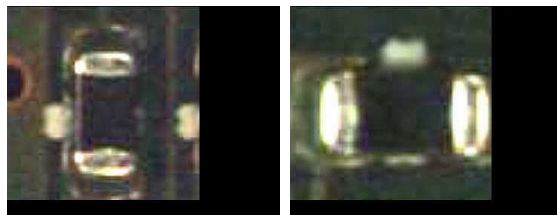
- Clase 0



- Clase 1



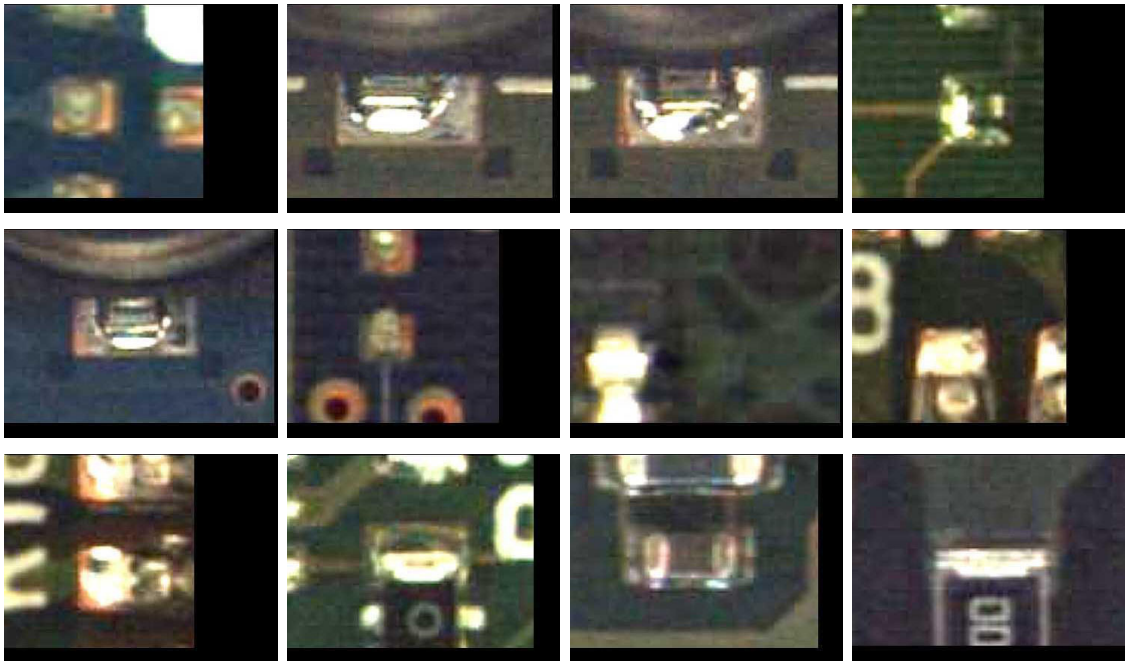
- Clase 2



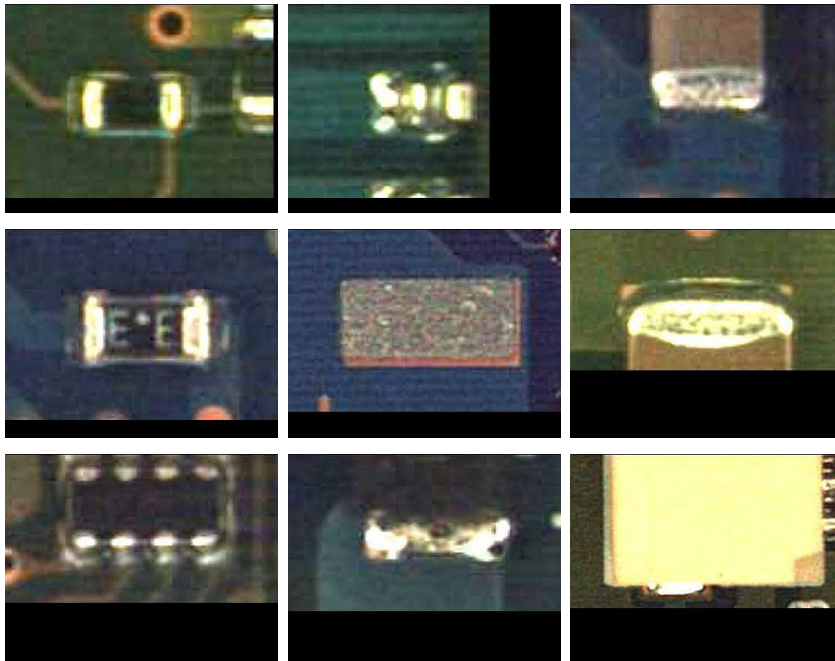
- Clase 3



■ Clase 4



■ Clase 5



■ Clase 6



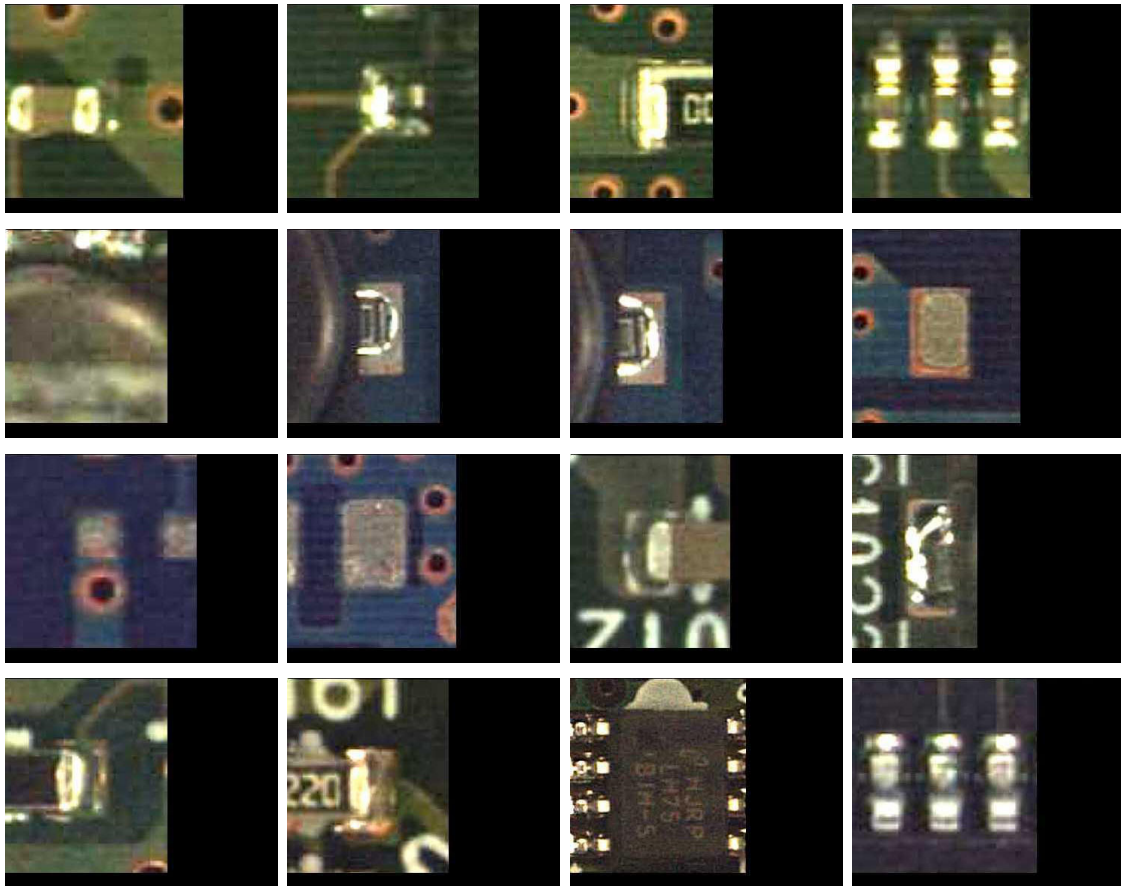
- Clase 7



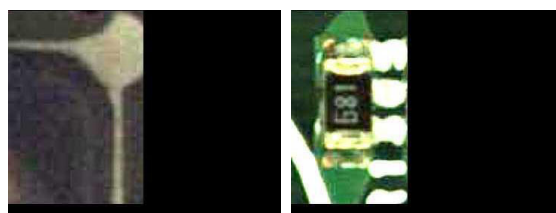
- Clase 8



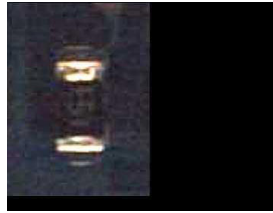
■ Clase 9



■ Clase 10



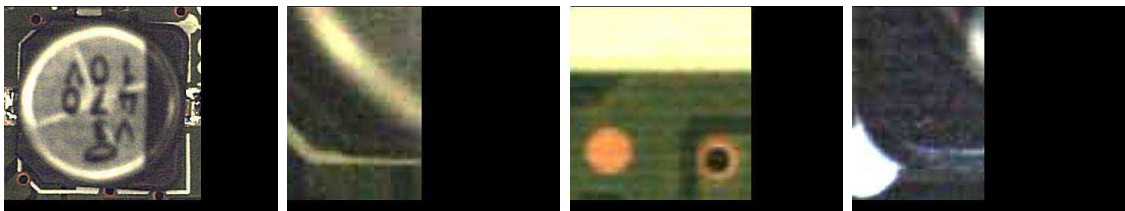
- Clase 11



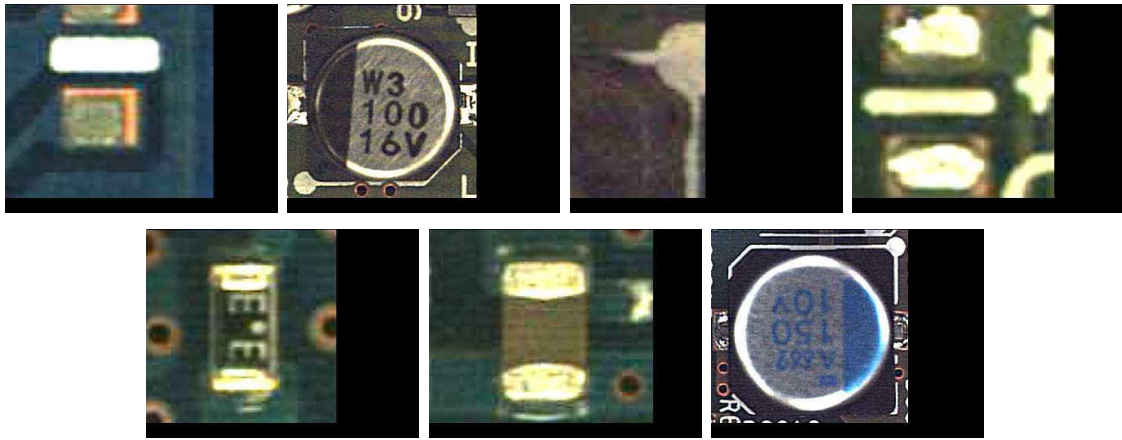
- Clase 12



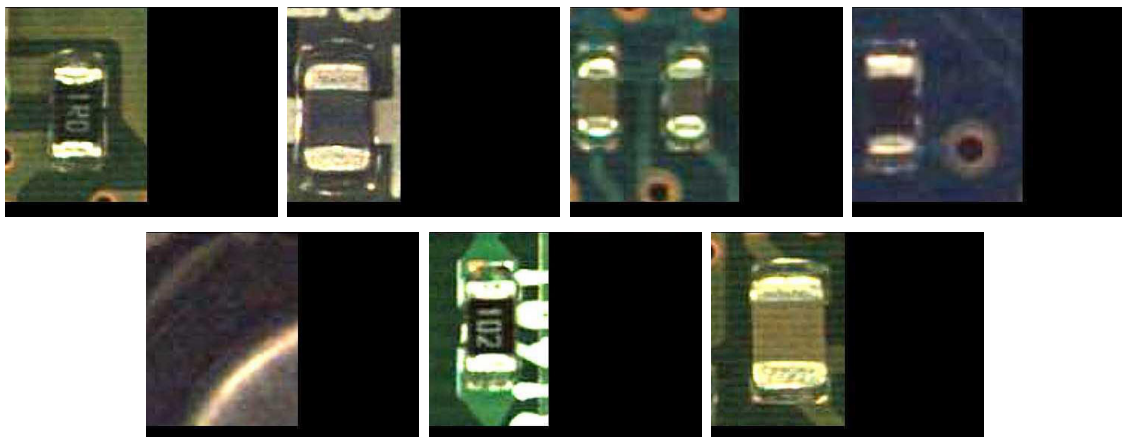
- Clase 13



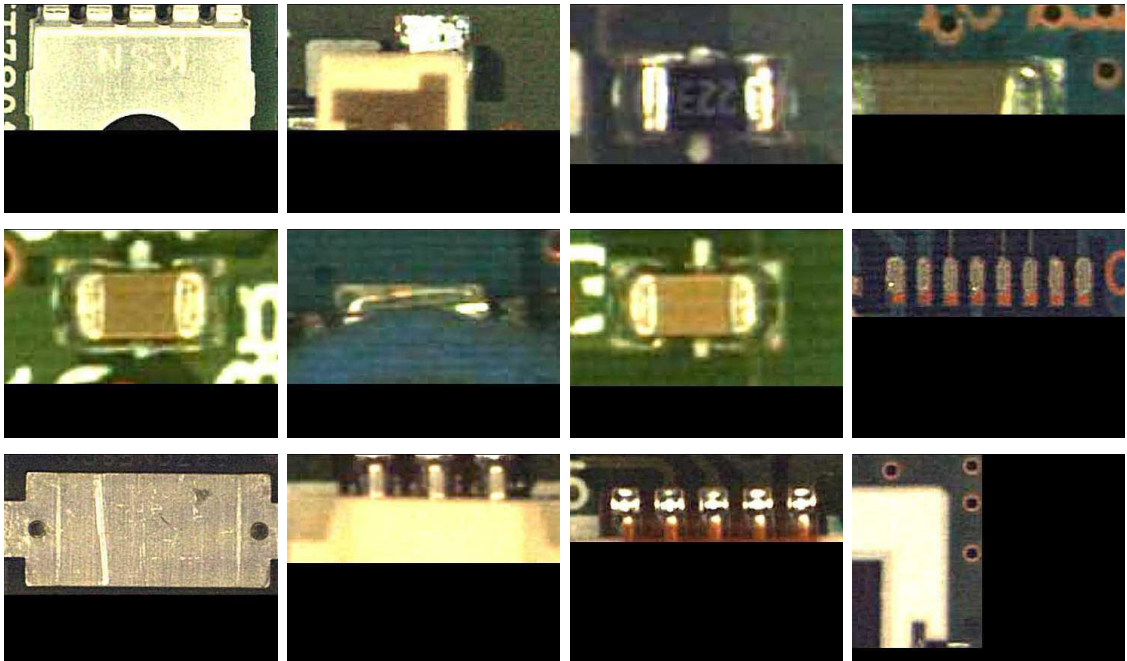
■ Clase 14



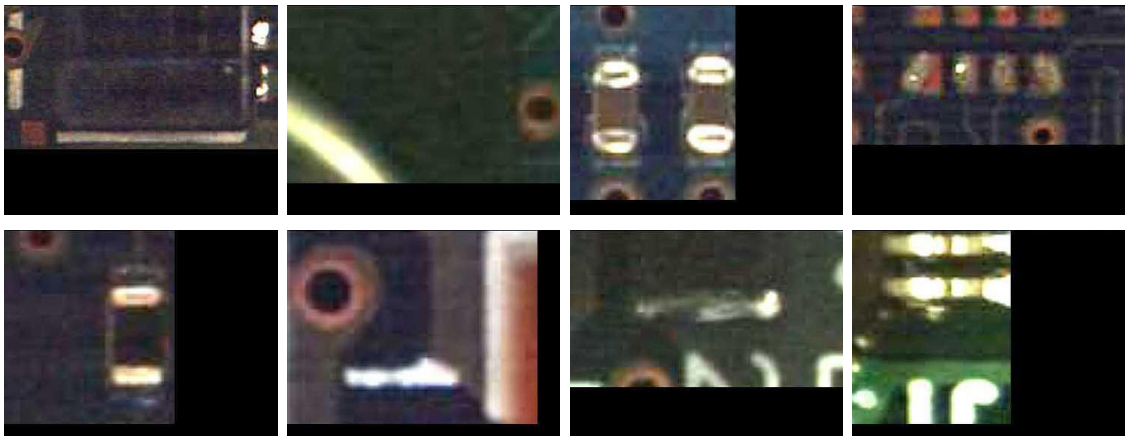
■ Clase 15



■ Clase 16



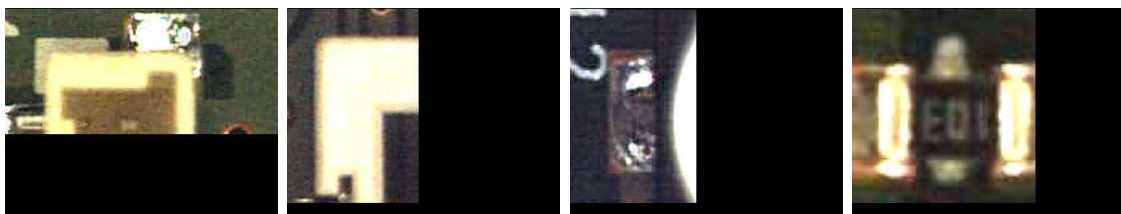
■ Clase 17



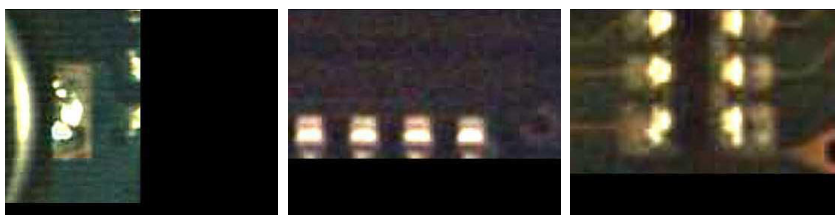
■ Clase 18



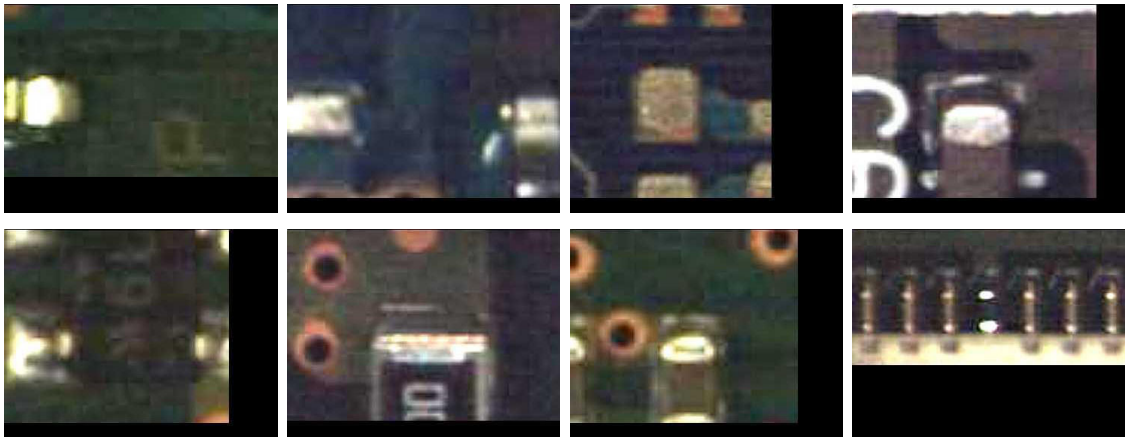
■ Clase 19



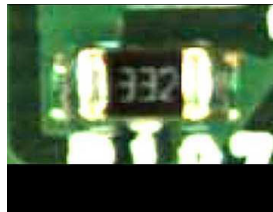
■ Clase 20



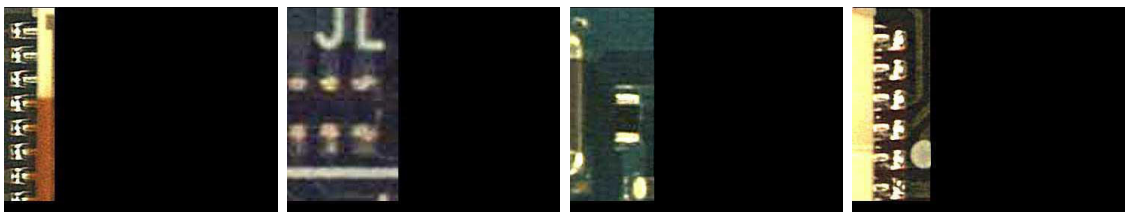
- Clase 21



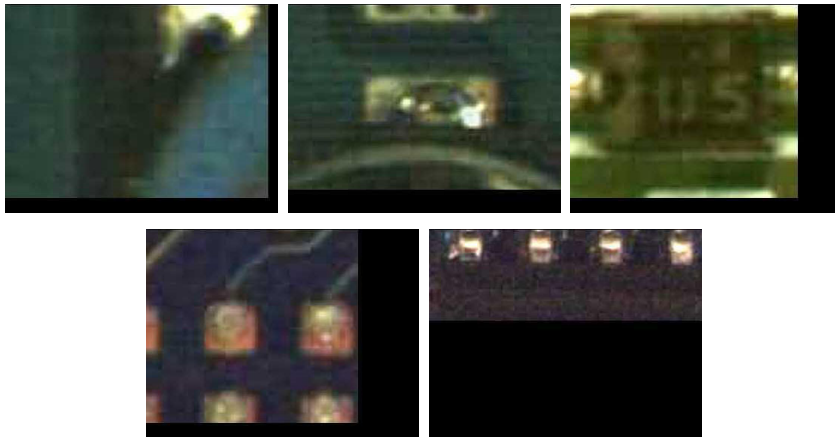
- Clase 22



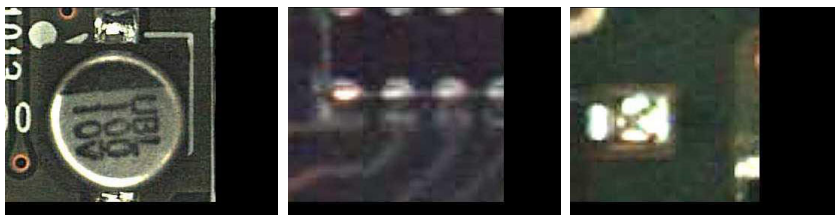
- Clase 23



■ Clase 24



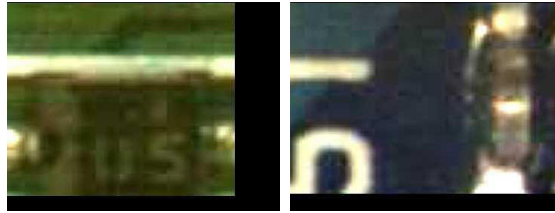
■ Clase 25



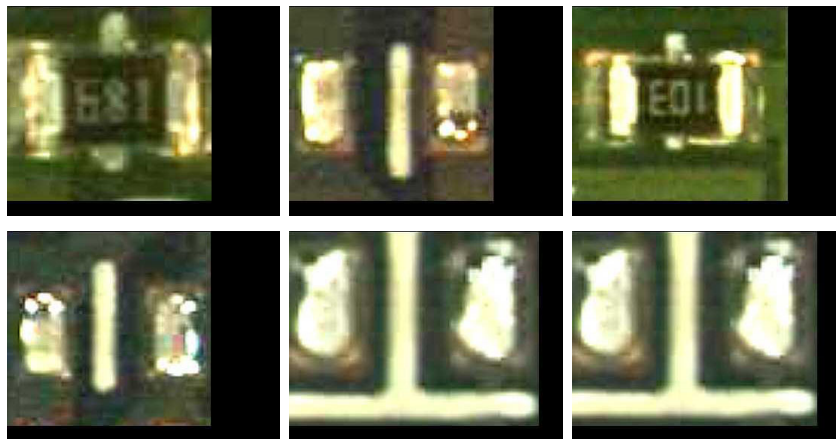
■ Clase 26



- Clase 27



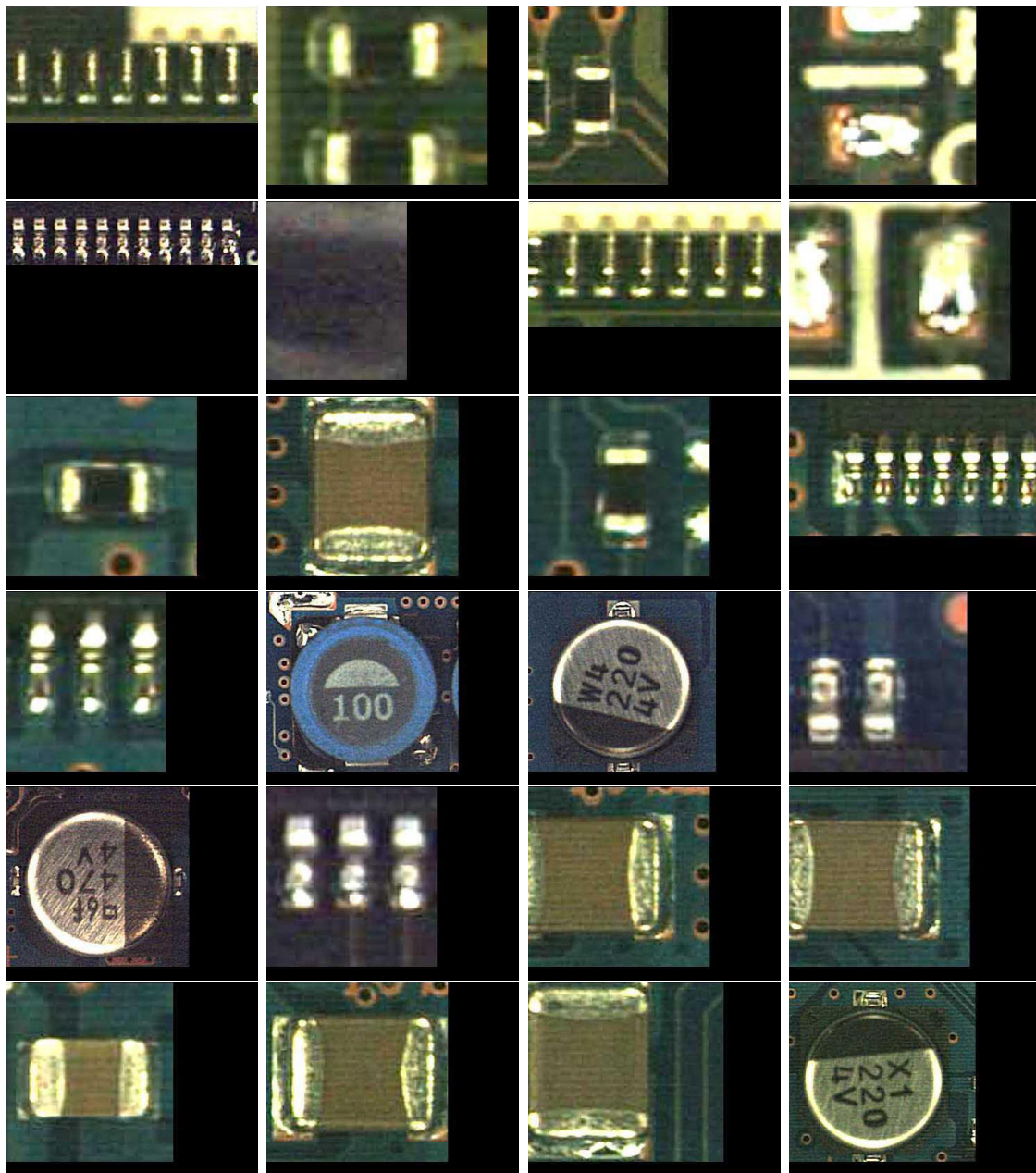
- Clase 28



- Clase 29



■ Clase 30



Bibliografía

- [1] David Vernon. *Machine Vision: Automated Visual Inspection and Robot Vision*. Prentice Hall International, 1991.
- [2] Nicu Sebe, Ira Cohen, Ashutosh Garg, and Thomas S. Huang. *Machine Learning in Computer Vision (Computational Imaging and Vision)*. Springer, 2005.
- [3] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson Learning, 2007.
- [4] Elias N. Malamas, Euripides G. M. Petrakis, Michalis Zervakis, Laurent Petit, and Jean-Didier Legat. A survey on industrial vision systems, applications and tools. *Image and Vision Computing*, 21(2):171–188, February 2003.
- [5] Duncan Wilson, Alistair Greig, John Gilby, and Robert Smith. Uncertainty in automated inspection, the problems in the real world.
- [6] Madhav Moganti, Fikret Ercal, Cihan H. Dagli, and Shou Tsunekawa. Automatic PCB inspection algorithms: A survey. *Computer Vision and Image Understanding: CVIU*, 63(2):287–313, 1996.
- [7] Nello Zuech. *Understanding and Applying Machine Vision, Second Edition, Revised and Expanded*. Marcel Dekker, Inc., 2000.
- [8] Luis Francisco Barbosa Santillán and Adriano de Luca Pennacchia. Localización de smd mediante técnicas de visión artificial implementadas en hardware. Master's thesis, Sección Mecatronica. Departamento de Ingeniería Eléctrica, CINVESTAV-IPN, Av. Instituto Politécnico Nacional 2508. 07300 México, D.F., 2006.
- [9] Miguel Ángel Sanchez Martinez and Adriano de Luca Pennacchia. Diseño en FPGA de un circuito comparador de imágenes. Master's thesis, Sección de Computación. Departamento de Ingeniería Eléctrica, CINVESTAV-IPN, Av. Instituto Politécnico Nacional 2508. 07300 México, D.F., 2005.
- [10] Robert Sablatnig. A flexible concept for automatic visual inspection.
- [11] G.R. Blackwell. *The Electrical Engineering Handbook*, chapter 26. CRC Press LLC, 2000.

- [12] Loh Horng Hai and Lu Ming Sing. Printed circuit board inspection using image analysis. *IEEE Transactions on Industry Applications*, 35(2):426–431, April 1999.
- [13] Lin Shih Chieh and Su Chia Hsin. A visual inspection system for surface mounted devices on printed circuit board. *IEEE Conference on Cybernetics and Intelligent Systems*, pages 1–4, June 2006.
- [14] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley, 1992.
- [15] Julio Cornejo Herrera, Adriana Lara López, Ricardo Landa Becerra, and Luis Gerardo de la Fraga. Una librería para procesamiento de imagen: scimagen. *VIII Conferencia de Ingeniería Eléctrica*, Septiembre 2002.
- [16] Jef Poskanzer and Bryan Henderson. Netpbm. <http://netpbm.sourceforge.net/doc/>, 1999.
- [17] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice in C (2nd Edition)*. Addison-Wesley, 1997.
- [18] H. D. Cheng, X. H. Jiang, Y. Sun, and Jingli Wang. Color image segmentation: advances and prospects. *Pattern Recognition*, 34(6):2259–2281, 2001.
- [19] J.J. Báez Rojas, M.L. Guerrero, J. Conde Acevedo, A. Padilla Vivanco, and G. Urcid Serrano. Segmentación de imágenes a color. *Revista Mexicana de Física*, 50(6), Diciembre 2004.
- [20] Abigail Martínez Rivas and Luis Gerardo de la Fraga. Reconstrucción de volúmenes a partir de sus mapas de contornos. Master's thesis, Sección de Computación del CINVESTAV-IPN, Av. Instituto Politécnico Nacional 2508. 07300 México, D.F., 2005.
- [21] Michael Unser. Sum and difference histograms for texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(1):118–125, Enero 1986.
- [22] I.A. Basheer and M. Hajmeer. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43:3–31, 2000.
- [23] Raul Rojas. *Neural Networks: A Systematic Introduction*. Springer-Verlag, Berlín, 1996.
- [24] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, pages 179–187, 1962.

- [25] Mohamed Rizon, Haniza Yazid, Puteh Saad, Ali Yeon Md Shakaff, Abdul Rahman Saad, Mohd Rozailan Mamat, Sazali Yaacob, Hazri Desa, and M. Karthi-gayan. Object detection using geometric invariant moment. *American Journal of Applied Sciences*, 2(6):1876–1878, 2006.