



INSTITUTO POLITÉCNICO NACIONAL

Centro de Investigación en Computación

Secretaría de Investigación y Posgrado

**Visión artificial para detección automática de
fallas estructurales en botellas de vidrio.**

T E S I S

**QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS EN INGENIERÍA DE CÓMPUTO
CON OPCIÓN EN SISTEMAS DIGITALES**

PRESENTA

ING. FERNANDO GABINO RAMÍREZ NEYRA

Directores de Tesis:

Dr. Norberto Flores Guzmán

Dr. Luis Pastor Sánchez Fernández





Índice general

Índice general	ii
Índice de figuras	iv
Índice de tablas	vi
Glosario	viii
Capítulo 1. Introducción	1
1.1 Antecedentes.....	1
1.2 Planteamiento del problema	2
1.3 Justificación	3
1.4 Hipótesis.....	3
1.5 Objetivos	3
1.5.1 Objetivo general	3
1.5.2 Objetivos específicos	4
1.6 Alcances del trabajo	4
1.7 Contribuciones	4
1.8 Organización del documento	5
Capítulo 2. Estado del arte	6
2.1 Trabajos relacionados con el problema planteado	6
2.2 Resultados reportados en trabajos relacionados	9
2.3 Uso de los trabajos desarrollados	12
Capítulo 3. Marco teórico	13
3.1 Etapas fundamentales del procesamiento digital de imágenes.....	13
3.1.1 Captura de imágenes.....	13
3.1.2. Preprocesamiento	13
3.1.3 Segmentación	14
3.1.4 Representación y descripción.....	17
3.1.5 Reconocimiento e interpretación	17
3.2 Imágenes a color y en niveles de gris	19
3.3 Índices visuales	20
3.3.1 Bordes y su detección	20
3.3.2 Rasgos puntuales	24
3.4 Morfología matemática	28
3.4.1 Transformaciones morfológicas elementales	28



3.4.2 Dilatación	29
3.4.3 Erosión	30
3.4.4 Apertura	31
3.4.5 Clausura	31
3.5 Redes neuronales artificiales	32
3.5.1 Red Neuronal de función de base radial (RBF).....	34
3.6 La cámara CCD	39
3.7 Módulo NI Vision Development	40
Capítulo 4. Metodología propuesta y experimentación	41
4.1 Metodología propuesta.....	41
4.2 Descripción general del las etapas de la metodología de solución propuesta.....	42
4.3 Desarrollo de las etapas de la metodología de solución propuesta.....	43
4.3.1 Adquirir imágenes	43
4.3.2 Filtrar	44
4.3.3 Detectar índices visuales.....	46
4.3.4 Binarizar.....	49
4.3.5 Extraer zona de interés.....	50
4.3.6 Morfología matemática.....	56
4.3.7 Clasificar	58
Capítulo 5. Resultados	65
5.1. Resultados en pruebas de detectores de esquinas: Harris y SUSAN	65
5.2. Resultados en pruebas realizadas con la transformada de Hough	65
5.3. Resultados en pruebas de clasificación de botellas con la red neuronal de función de base radial.....	66
Capítulo 6. Conclusiones y trabajos futuros	69
6.1 Conclusiones	69
6.2 Trabajos futuros.....	70
Referencias bibliográficas.....	71
Anexos	73
Anexo A. Diagrama a bloques del software desarrollado en la plataforma LabVIEW® 2009.	
Anexo B. Código fuente de la librería de enlace dinámico de la transformada de Hough, desarrollada en NI LabWindows CVI® 2009.	
Anexo C. Código fuente de la librería de enlace dinámico de la red neuronal de función de base radial, desarrollada en NI LabWindows CVI® 2009.	



Índice de figuras

	No. Página
Figura 1. Histograma bimodal.	15
Figura 2. (a) Imagen binaria, (b), (c) y (d) Secuencia de etiquetado.	17
Figura 3. (a) Puntos colineales (A, B y C) y (b) Transformada de Hough de (a).....	26
Figura 4. Formas básicas de elementos de estructura planos.	28
Figura 5. (a) Conjunto A, (b) Elemento de estructura y (c) Dilatación de (A) con (B).	29
Figura 6. (a) Imagen binaria, (b) Elemento de estructura y (c) Dilatación de (a) con (b). .	29
Figura 7. (a) Conjunto A, (b) Elemento de estructura y (c) Erosión de (A) con (B).	30
Figura 8. (a) Imagen binaria, (b) Elemento de Estructura y (c) Erosión de (a) con (b).	30
Figura 9. (a) Conjunto A, (b) Elemento de estructura, (c) Erosión de (a) con (b) y (d)	31
Figura 10. (a) Conjunto A, (b) Elemento de estructura, (c) Dilatación de (a) con (b) y (d) Clausura de (a) con (b).....	32
Figura 11. Estructura de una red de base radial.	35
Figura 12. Representación de función de base radial.	36
Figura 13. Diagrama a bloques de la metodología de solución propuesta.	41
Figura 14. Ubicación espacial de los dispositivos utilizados para adquirir imágenes.	44
Figura 15. Textura y ruido presente en las imágenes adquiridas.	45
Figura 16. Imágenes obtenidas después de emplear el filtro mediana de 5x5 píxeles	45
Figura 17. Imágenes binarias obtenidas después de detectar bordes con el operador de Roberts.	46
Figura 18. Imágenes binarias obtenidas después de realzar bordes con operador de Roberts con valores de umbral $u = 20$ y $u = 180$	47
Figura 19. Imágenes binarias obtenidas después de detectar bordes con la máscara 1 de Sobel.	47
Figura 20. Imágenes binarias obtenidas después de detectar bordes con la máscara 2 de Sobel.	48
Figura 21. Realce de bordes con elemento de estructura Fila de 1x3 píxeles.	48
Figura 22. Realce de bordes con elemento de estructura cuadrado de 5x5 píxeles.	49
Figura 23. Realce de bordes con elemento de estructura circular de 7x7 píxeles.	49



Figura 24. (a), (b) y (c) Imágenes binarias obtenidas con valores de umbral de 70,100 y 130.	50
Figura 25. (a) Botella con inclinación en el cuello y (b) Botella sin inclinación en el cuello.	51
Figura 26. Ubicación del primer píxel válido detectado.	51
Figura 27. Imágenes de la botella vista desde diferentes ángulos.	52
Figura 28. Imágenes binarias de botellas de vidrio con línea superior.	53
Figura 29. Imágenes binarias de botellas de vidrio, con las cinco líneas de mayor tamaño detectadas con la transformada de Hough.	53
Figura 30. Líneas de mayor tamaño detectadas con la transformada de Hough.	54
Figura 31. Líneas de mayor tamaño detectadas con la transformada de Hough.	54
Figura 32. Extracción de la zona de interés.	55
Figura 33. Adelgazamiento de bordes que representan el defecto de la botella de vidrio. ..	56
Figura 34. Bordes no deseados presentes en el área de interés.	56
Figura 35. Separación de bordes que representan el defecto de la botella de vidrio.	56
Figura 36. Bordes creados por molde de fabricación de las botellas.	57
Figura 37. Imagen binaria con borde creado por el molde.	57
Figura 38. Imagen binaria después de emplear operación de apertura morfológica.	58
Figura 39. Resultado obtenido al procesar una imagen de una botella sin defecto, con la metodología propuesta.	61
Figura 40. Resultado obtenido al procesar una imagen de una botella sin defecto, con la metodología propuesta.	62
Figura 41. Resultado obtenido al procesar una imagen de una botella con defecto, con la metodología propuesta.	62
Figura 42. Resultado obtenido al procesar una imagen de una botella sin defecto, con la metodología propuesta.	63
Figura 43. Resultado obtenido al procesar una imagen de una botella con defecto, con la metodología propuesta.	63
Figura 44. Resultado obtenido al procesar una imagen de una botella sin defecto, con la metodología propuesta.	64
Figura 45. Resultado obtenido al procesar una imagen de una botella con defecto, con la metodología propuesta.	64



Índice de tablas

No. Página

Tabla 1. Resultados reportados en el artículo, “Development of A Computerized Method to Inspect Empty Glass Bottle”, en la clasificación de botellas de vidrio, con y sin defectos en el cuerpo.....	9
Tabla 2. Resultados reportados en el artículo, “Development of A Computerized Method to Inspect Empty Glass Bottle”, en la clasificación de botellas de vidrio, con y sin defectos en la corona.....	9
Tabla 3. Resultados reportados en el artículo “Fault detection by a gaussian neural network with reject options in glass bottle production”.	10
Tabla 4. Resultados reportados en el artículo, “Automated Visual Inspection of Glass Bottles Using Adapted Median Filtering ”.....	10
Tabla 5. Resultados reportados en el artículo “Detección de fallas en envases de vidrio no cilíndricos utilizando localización de bordes mediante la herramienta Matlab”. ..	11
Tabla 6. Resultados reportados en el artículo “A glass bottle defect detection system without touching”.....	11
Tabla 7. Resultados reportados en el trabajo: “Análisis y detección de fallas estructurales en envases de vidrio utilizando procesamiento digital de imágenes”.	12
Tabla 8. Máscara Gy	22
Tabla 9. Máscara Gx	22
Tabla 10. Vecindad 8 entre píxeles.....	22
Tabla 11. Máscara no. 2 de Sobel.....	23
Tabla 12. Máscara no. 1 de Sobel.....	23
Tabla 13. Máscara no.2 de Prewitt.	23
Tabla 14. Máscara no. 1 de Prewitt.....	23
Tabla 15. Máscara 2 (Gy).....	46
Tabla 16. Máscara 1 (Gx).....	46
Tabla 17. Máscara 2 (Gy).....	47
Tabla 18. Máscara 1 (Gx).....	47



Tabla 19. Elemento de estructura para eliminar bordes no deseados	56
Tabla 20. Elemento de estructura para eliminar línea vertical	57
Tabla 21. Detectores de esquinas de Harris y SUSAN	65
Tabla 22. Resultados de prueba con transformada de Hough.	65
Tabla 23. Prueba 2 con transformada de Hough.....	66
Tabla 24. Resultados de prueba de clasificación con red neuronal de base radial.	66
Tabla 25. Resultado de prueba de clasificación por medio del cálculo del área ocupada por el defecto.	67
Tabla 26 Porcentaje de efectividad de trabajos desarrollados para detectar defectos en botellas de vidrio.....	68

Resumen

En el presente trabajo se propone y desarrolla una metodología a nivel de software, enfocada a dar solución al problema de detección de defectos físicos en la corona inferior de botellas de vidrio, y para ello se hace uso de algunas técnicas de procesamiento digital de imágenes clásico como: filtros mediana, el detector de bordes de Sobel y la Transformada de Hough. Además durante ésta investigación se han probado y empleado técnicas más actuales como redes neuronales y morfología matemática, para observar sus posibles beneficios como técnicas de reconocimiento de patrones y filtrado.

Dicha metodología está integrada de siete etapas en las que se realiza la adquisición de la imagen (etapa 1) mediante hardware firewire e iluminación controlada tipo Backlight, (etapa 2) se elimina ruido de la imagen mediante un filtro de mediana, se obtiene el gradiente de la imagen empleando el operador de Sobel y después (etapa 3) se binariza la imagen, enseguida (etapa 4) con la transformada de Hough se detectan líneas con ángulos de inclinación dentro del intervalo discreto de -3 a 3 grados y en base a su ubicación dentro de la imagen, se determina cuál esas líneas es la que representa el borde de la corona superior de la botella.

Posteriormente se segmenta la imagen (etapa 5) extrayendo el área en donde es posible encontrar los bordes que representan el defecto de la botella y para ello se realiza un desplazamiento vertical a partir de la posición de línea que representa el borde de la corona superior de la botella y hasta llegar a la zona de interés. Para lograr que dicho desplazamiento se realice de forma adecuada sin importar la variación de escala de las imágenes, se programaron algunas formulas para: normalizar el tamaño de la imagen de entrada, calcular el área de la zona de interés y obtener el número de píxeles que se deben recorrer para llegar a la zona de interés.

Después (etapa 6) se eliminan los bordes que no representan ningún defecto, se unen los bordes separados que si representan algún defecto y además se aumenta el tamaño de estos, mediante las operaciones morfológicas: apertura, cerradura, dilatación y erosión.

Finalmente, (etapa 7) se calcula el momento de orden cero (área) como rasgo descriptor del defecto y este dato es comparado con uno previamente preestablecido, y con base al resultado de ésta comparación la imagen es clasificada dentro de alguna de las clases: “botella con defecto” y “botella sin defecto”.

Abstract

In this work, a software level methodology is proposed and developed, focused on give a solution to the physical defect detection problem, with regards to defects located in the inferior collar of glass bottles. To achieve this, some techniques of classical digital image processing are used, like: Median Filter, Sobel Edges Detector and Hough Transform. Moreover, recent techniques as neural networks and mathematical morphology have been proved and used during this research to observe their possible benefits as pattern recognition and filtering techniques.

Such methodology is formed by seven stages where (stage 1) Acquisition is done by means of firewire hardware and controlled backlight illumination; (stage 2) noise is eliminated from the image through a median filter, the image gradient is obtained by using the Sobel operator and then, (stage 3) the image is binarized, then (stage 4) using the Hough Transform lines with a discrete defined slope from -3 to 3 grades are detected, so the line representing the top collar of the bottle is determined.

Subsequently, the image is segmented (stage 5) by extracting the area of interest where edges representing the bottle defect can be found, to do that a vertical displacement is performed from the line position representing the top collar to the zone of interest. To achieve that this movement is done correctly, regardless of image scale variation, some formulas were programmed to: normalize input image size, calculate zone of interest area and obtain the pixels numbers that must be jumped to reach the correct zone of interest.

Afterwards, (stage 6) edges not representing any defect are vanished, separated failure edges are unified and its size is increased as well, by means of morphological operations: aperture, closure, dilatation and erosion.

Finally, (stage 7) the zero order moment (area) is calculated as feature descriptor of the defect and this data is compared to a setpoint previously identified and, depending on the comparison result, the image is classified as member of one class: “defective bottle” and “bottle without defects”.

Capítulo 1

En el presente capítulo se describen los antecedentes y el planteamiento del problema, se establece: el objetivo general, los objetivos particulares que se pretenden alcanzar, y además se describen: las contribuciones, los alcances del trabajo y el contenido de cada capítulo que conforma este trabajo.



Capítulo 1. Introducción

1.1 Antecedentes

El descubrimiento del vidrio fue trascendental para el desarrollo de la botella tal como la conocemos en la actualidad, pero pasaron muchos años antes de que empezara a utilizarse para la elaboración de envases.

Hasta la primera mitad del siglo XVII, la mayor parte de los recipientes empleados para almacenar y transportar líquidos estaban elaborados en cerámica o loza, hasta que los egipcios le dieron un mayor uso al vidrio como material decorativo, y de uso doméstico para la conservación y almacenamiento de determinados productos alimenticios.

Hasta la Edad Media el uso del vidrio estuvo en manos de unos pocos privilegiados, que mantenían en secreto su composición y fabricación. El vidrio se convirtió en un objeto de lujo, empleado tanto en la decoración como en la conservación, transporte y almacenaje de alimentos, medicinas, aceites, etc. De hecho, el oficio de vidriero fue el único al que la nobleza podía entregarse en Francia durante ese tiempo.

En los primeros años del siglo XX, después de numerosas investigaciones, se crea la primera máquina para la fabricación automática y producción en serie de envases de vidrio.

Existen dos técnicas para la fabricación de envases:

Soplado-soplado: utilizado habitualmente para la fabricación de botellas. Esta técnica se basa en el uso de un punzón que forma una pequeña cavidad en la masa de vidrio fundido (gota de vidrio); esta cavidad es posteriormente expandida mediante aire comprimido. Al expandirse, la gota es forzada a adoptar una determinada configuración que corresponde al molde preliminar que, por su diseño, dará el parto ideal del vidrio creándose así una botella semi-formada, llamada parison o preliminar. A continuación, este parison es transferido al molde terminador que dará la forma final al envase, usando aire comprimido.

Prensado-soplado: desarrollado inicialmente para los envases de boca ancha (tarros) actualmente también se utiliza para los envases de boca estrecha (botellas) al obtenerse un mejor control en el reparto del vidrio. El prensado-soplado comienza por el uso de un punzón relativamente largo, que forma una cavidad interior en la gota, prensando el vidrio contra la superficie del molde preliminar de tal forma que el espacio ocupado por el punzón, corresponde a la burbuja de aire formada por el primer soplo del sistema tradicional. Esta forma del parison es igualmente transferida al molde terminador donde, con la ayuda del aire comprimido, es expandida de una forma simple y más uniforme hasta la forma final del envase.

Las máquinas utilizadas habitualmente son longitudinales, constituidas por varias secciones que funcionan de forma independiente, y que pueden pararse una o varias de ellas, mientras



que las demás siguen produciendo. Normalmente, constan de 6, 8, 10, 12, 16 ó 20 secciones y, cada una de ellas, puede trabajar en S.G. (simple gota o un envase por sección), D.G. (doble gota o dos envases por sección) y T.G. (triple gota o tres envases por sección).

Una vez fabricada la botella o tarro de vidrio, aún a una temperatura elevada, es introducido en un túnel (archa) de recocido para evitar la formación de tensiones internas debidas a un enfriamiento rápido con temperaturas controladas. Los envases pasan por este túnel lentamente siendo recalentados y posteriormente enfriados de una manera predeterminada, así el vidrio adquiere un mayor grado de resistencia.

A la salida del archa de recocido, los envases son tratados en su cara externa para disminuir el coeficiente de rozamiento entre ellos, y permitir un fácil deslizamiento por las líneas de inspección y embalado automático. Al final del recocido, un separador distribuye los envases a las diferentes líneas de inspección y transporte. Sobre estas líneas personal calificado y una serie de máquinas electrónicas supervisan los envases y eliminan aquellos que no cumplen las exigencias de calidad previstas. Habitualmente son máquinas optoelectrónicas que se encargan de realizar controles dimensionales y funcionales del envase.

1.2 Planteamiento del problema

Durante la producción de botellas de vidrio, en ocasiones surgen defectos en éstas, ya sea por error humano, por la calidad de la materia prima o por algún mal funcionamiento de la máquina productora. Entre los defectos más sobresalientes están: las fracturas en el cuello de la botella, piedras, ampollas y defectos en las dimensiones de la botella. En diversas ocasiones esos defectos se detectan al final del ciclo de producción, lo que puede traer consecuencias graves tanto a la línea de producción como al cliente.

Actualmente compañías como: EMHART INES, KRONES, Glasstec, VISIGLASS y SGCC, han desarrollado diversos sistemas de inspección visual que pueden adaptarse a una gran cantidad de procesos industriales, donde se requiere inspeccionar de manera automática, ciertas características físicas y/o funcionales de diversos productos de vidrio o plástico (PET). Los sistemas de inspección visual como: SGCC SEALCAM, VISIGLAS Astra Light System y VISIGLAS ARGOS, tienen la capacidad de inspeccionar específicamente botellas de vidrio para el envasado de productos cosméticos, cada uno de esos sistemas puede inspeccionar, una o más partes de la botella como por ejemplo: el sistema SGCC SEALCAM, emplea cámaras digitales para capturar imágenes en color del cuerpo y corona de la botella, para detectar defectos como: burbujas, líneas, piedras, fracturas, y defectos en las dimensiones de la botella, la capacidad de inspección de este sistema es de 1 a 400 botellas por minuto.

El sistema **VISIGLAS Astra Light System**, emplea 3 cámaras digitales, distribuidas de tal forma que cubren 360° para detectar defectos físicos en el cuerpo de botellas de vidrio, la capacidad de inspección de este equipo es de 1 a 200 botellas por minuto.

El sistema **VISIGLAS ARGOS**, emplea hasta 8 cámaras digitales, situadas alrededor de la botella de vidrio para inspeccionar el cuello de esta y detectar defectos físicos como:

fracturas, burbujas, líneas, piedras y defectos dimensionales. La capacidad de inspección de este sistema es de 1 a 500 botellas por minuto.

Aunque el sistema VISIGLAS ARGOS y otros, entregan buenos resultados al inspeccionar y detectar defectos como: fracturas, burbujas, ampollas, líneas, y dimensiones erróneas en botellas de vidrio lisas o de diseño simple, omiten la detección de algunos de los defectos anteriores en modelos de botellas más elaborados, es decir aquellos que presentan algún grabado, una rosca y/o corona inferior en el cuello de la botella. Es por eso que las fallas localizadas en esa zona de la botella son complicadas de detectar y representan un área, abierta a la investigación.

1.3 Justificación

Los sistemas de producción de botellas de vidrio, tienen etapas susceptibles a fallos que ocasionan defectos físicos en los productos. Dichas fallas se presentan como errores en el ensamble o en los procesos de transformación y se deben a factores como materia prima anómala, fallas en la automatización o a algún error humano.

Existen procesos de inspección que consisten en la observación humana sobre las líneas de producción. Sin embargo en muchos casos, por la velocidad de producción, la exactitud involucrada y/o las condiciones no ergonómicas, resulta imposible confiar la operación de inspección a una persona, por lo que resulta necesario desarrollar un sistema de inspección visual automático, que detecte y clasifique botellas de vidrio con fallas estructurales en el cuello, empleando técnicas, métodos y/o metodologías de visión artificial, procesamiento digital de señales e imágenes y de reconocimiento de patrones que tenga un costo relativamente bajo en comparación a los sistemas de inspección visual existentes en el mercado, que debido a que fueron diseñados para que puedan adaptarse a diversos procesos de inspección visual, tienen un precio mucho mayor, ya que éstos tienen un gran número de características y funciones configurables que en determinados procesos no son necesarias.

1.4 Hipótesis

Mediante técnicas de procesamiento digital de imágenes, como: filtros espaciales, detectores de bordes, y morfología matemática, aplicada a imágenes binarias, es posible detectar fallas estructurales en botellas de vidrio presentes en la corona inferior.

1.5 Objetivos

1.5.1 Objetivo general

Desarrollar y poner en operación una metodología basada en el análisis de imágenes digitales, para clasificar de forma automática botellas de vidrio, que presenten o no fallas estructurales en la zona del cuello conocida como corona inferior.

1.5.2 Objetivos específicos

- Utilizar algunas de las técnicas y metodologías ya propuestas sobre procesamiento digital de imágenes, para detectar fallas estructurales en el cuello de botellas de vidrio y analizar los resultados obtenidos, para decidir si son convenientes en la solución total o parcial del problema que se pretende solucionar en esta investigación.
- Proponer una metodología a nivel de software, con la que sea posible detectar defectos estructurales en la corona inferior de botellas de vidrio, para clasificarlas como defectuosas en caso de existir algún defecto ó como no defectuosa en caso contrario.

1.6 Alcances del trabajo

En este trabajo se desarrolla e implementa una metodología a nivel de software con la cual es posible clasificar botellas de vidrio, ya sea como buenas o malas, cuando éstas hayan o no sufrido alguna alteración física en su estructura durante el proceso de fabricación. Para poder obtener esos resultados se implementaron algunas técnicas de procesamiento digital de imágenes, morfología matemática y de reconocimiento de patrones.

1.7 Contribuciones

- La metodología propuesta para resolver el problema de clasificación de botellas de vidrio en esta investigación es diferente a las encontradas en los trabajos [1],[2],[3],[4] y [6], ya que en ésta, además de utilizar algunas operaciones morfológicas para eliminar y aumentar el tamaño de algunos bordes deseados y no deseados dentro de las imágenes, se emplea la transformada de Hough para detectar líneas en la parte superior de la imagen, de las cuales una sirve como referencia para ubicar el objeto de interés y extraer de la imagen el área donde es posible encontrar los píxeles que representan el defecto de la botella de vidrio.
- La metodología propuesta en esta investigación presenta una solución alternativa a algunas ya existentes, las cuales están limitadas a trabajar solo con botellas de vidrio lisas, es decir, aquellas que no tienen rosca ni corona inferior en el cuello de la botella.
- La segmentación y extracción del área de la imagen, en donde es posible localizar los píxeles que representan el defecto de las botellas, son invariantes ante el cambio de escala y translación horizontal de la botella, ya que se consideró que durante el traslado de las botellas sobre la banda transportadora estas no están fijas, lo cual provoca que estas puedan estar más cercanas o alejadas de la cámara, y al ser fotografiadas pueda haber variaciones en el tamaño de la botella dentro de la imagen.



1.8 Organización del documento

El presente trabajo está estructurado en seis capítulos y a continuación se describe el contenido de cada uno de ellos:

- En el primer capítulo denominado Introducción, se describen los antecedentes y el planteamiento del problema, se establece: el objetivo general, los objetivos particulares que se pretenden alcanzar, y además se describen: las contribuciones, los alcances del trabajo y el contenido de cada capítulo que conforma este trabajo.
- En el segundo capítulo, se describe el estado actual y los resultados de los trabajos reportados en la literatura que están relacionados con la clasificación de botellas de vidrio.
- En el tercer capítulo, se exponen los conceptos, definiciones, métodos, algoritmos y técnicas de: procesamiento de imágenes, de visión computacional y de reconocimiento de patrones en las que se fundamenta este trabajo.
- En el cuarto capítulo, se propone la metodología con la que se pretende dar solución al problema planteado en el capítulo 1, y se describen los experimentos realizados en cada una de las etapas que conforman dicha metodología.
- En el quinto capítulo, se describen los resultados de las pruebas realizadas en las etapas de la metodología de solución propuesta.
- Y finalmente, en el sexto capítulo se dan las conclusiones del trabajo y se describen las modificaciones que se pretenden realizar en el futuro.

Capítulo 1

En el presente capítulo se describen los antecedentes y el planteamiento del problema, se establece: el objetivo general, los objetivos particulares que se pretenden alcanzar, y además se describen: las contribuciones, los alcances del trabajo y el contenido de cada capítulo que conforma este trabajo.



Capítulo 1. Introducción

1.1 Antecedentes

El descubrimiento del vidrio fue trascendental para el desarrollo de la botella tal como la conocemos en la actualidad, pero pasaron muchos años antes de que empezara a utilizarse para la elaboración de envases.

Hasta la primera mitad del siglo XVII, la mayor parte de los recipientes empleados para almacenar y transportar líquidos estaban elaborados en cerámica o loza, hasta que los egipcios le dieron un mayor uso al vidrio como material decorativo, y de uso doméstico para la conservación y almacenamiento de determinados productos alimenticios.

Hasta la Edad Media el uso del vidrio estuvo en manos de unos pocos privilegiados, que mantenían en secreto su composición y fabricación. El vidrio se convirtió en un objeto de lujo, empleado tanto en la decoración como en la conservación, transporte y almacenaje de alimentos, medicinas, aceites, etc. De hecho, el oficio de vidriero fue el único al que la nobleza podía entregarse en Francia durante ese tiempo.

En los primeros años del siglo XX, después de numerosas investigaciones, se crea la primera máquina para la fabricación automática y producción en serie de envases de vidrio.

Existen dos técnicas para la fabricación de envases:

Soplado-soplado: utilizado habitualmente para la fabricación de botellas. Esta técnica se basa en el uso de un punzón que forma una pequeña cavidad en la masa de vidrio fundido (gota de vidrio); esta cavidad es posteriormente expandida mediante aire comprimido. Al expandirse, la gota es forzada a adoptar una determinada configuración que corresponde al molde preliminar que, por su diseño, dará el parto ideal del vidrio creándose así una botella semi-formada, llamada parison o preliminar. A continuación, este parison es transferido al molde terminador que dará la forma final al envase, usando aire comprimido.

Prensado-soplado: desarrollado inicialmente para los envases de boca ancha (tarros) actualmente también se utiliza para los envases de boca estrecha (botellas) al obtenerse un mejor control en el reparto del vidrio. El prensado-soplado comienza por el uso de un punzón relativamente largo, que forma una cavidad interior en la gota, prensando el vidrio contra la superficie del molde preliminar de tal forma que el espacio ocupado por el punzón, corresponde a la burbuja de aire formada por el primer soplo del sistema tradicional. Esta forma del parison es igualmente transferida al molde terminador donde, con la ayuda del aire comprimido, es expandida de una forma simple y más uniforme hasta la forma final del envase.

Las máquinas utilizadas habitualmente son longitudinales, constituidas por varias secciones que funcionan de forma independiente, y que pueden pararse una o varias de ellas, mientras



que las demás siguen produciendo. Normalmente, constan de 6, 8, 10, 12, 16 ó 20 secciones y, cada una de ellas, puede trabajar en S.G. (simple gota o un envase por sección), D.G. (doble gota o dos envases por sección) y T.G. (triple gota o tres envases por sección).

Una vez fabricada la botella o tarro de vidrio, aún a una temperatura elevada, es introducido en un túnel (archa) de recocido para evitar la formación de tensiones internas debidas a un enfriamiento rápido con temperaturas controladas. Los envases pasan por este túnel lentamente siendo recalentados y posteriormente enfriados de una manera predeterminada, así el vidrio adquiere un mayor grado de resistencia.

A la salida del archa de recocido, los envases son tratados en su cara externa para disminuir el coeficiente de rozamiento entre ellos, y permitir un fácil deslizamiento por las líneas de inspección y embalado automático. Al final del recocido, un separador distribuye los envases a las diferentes líneas de inspección y transporte. Sobre estas líneas personal calificado y una serie de máquinas electrónicas supervisan los envases y eliminan aquellos que no cumplen las exigencias de calidad previstas. Habitualmente son máquinas optoelectrónicas que se encargan de realizar controles dimensionales y funcionales del envase.

1.2 Planteamiento del problema

Durante la producción de botellas de vidrio, en ocasiones surgen defectos en éstas, ya sea por error humano, por la calidad de la materia prima o por algún mal funcionamiento de la máquina productora. Entre los defectos más sobresalientes están: las fracturas en el cuello de la botella, piedras, ampollas y defectos en las dimensiones de la botella. En diversas ocasiones esos defectos se detectan al final del ciclo de producción, lo que puede traer consecuencias graves tanto a la línea de producción como al cliente.

Actualmente compañías como: EMHART INES, KRONES, Glasstec, VISIGLASS y SGCC, han desarrollado diversos sistemas de inspección visual que pueden adaptarse a una gran cantidad de procesos industriales, donde se requiere inspeccionar de manera automática, ciertas características físicas y/o funcionales de diversos productos de vidrio o plástico (PET). Los sistemas de inspección visual como: SGCC SEALCAM, VISIGLAS Astra Light System y VISIGLAS ARGOS, tienen la capacidad de inspeccionar específicamente botellas de vidrio para el envasado de productos cosméticos, cada uno de esos sistemas puede inspeccionar, una o más partes de la botella como por ejemplo: el sistema SGCC SEALCAM, emplea cámaras digitales para capturar imágenes en color del cuerpo y corona de la botella, para detectar defectos como: burbujas, líneas, piedras, fracturas, y defectos en las dimensiones de la botella, la capacidad de inspección de este sistema es de 1 a 400 botellas por minuto.

El sistema **VISIGLAS Astra Light System**, emplea 3 cámaras digitales, distribuidas de tal forma que cubren 360° para detectar defectos físicos en el cuerpo de botellas de vidrio, la capacidad de inspección de este equipo es de 1 a 200 botellas por minuto.

El sistema **VISIGLAS ARGOS**, emplea hasta 8 cámaras digitales, situadas alrededor de la botella de vidrio para inspeccionar el cuello de esta y detectar defectos físicos como:

fracturas, burbujas, líneas, piedras y defectos dimensionales. La capacidad de inspección de este sistema es de 1 a 500 botellas por minuto.

Aunque el sistema VISIGLAS ARGOS y otros, entregan buenos resultados al inspeccionar y detectar defectos como: fracturas, burbujas, ampollas, líneas, y dimensiones erróneas en botellas de vidrio lisas o de diseño simple, omiten la detección de algunos de los defectos anteriores en modelos de botellas más elaborados, es decir aquellos que presentan algún grabado, una rosca y/o corona inferior en el cuello de la botella. Es por eso que las fallas localizadas en esa zona de la botella son complicadas de detectar y representan un área, abierta a la investigación.

1.3 Justificación

Los sistemas de producción de botellas de vidrio, tienen etapas susceptibles a fallos que ocasionan defectos físicos en los productos. Dichas fallas se presentan como errores en el ensamble o en los procesos de transformación y se deben a factores como materia prima anómala, fallas en la automatización o a algún error humano.

Existen procesos de inspección que consisten en la observación humana sobre las líneas de producción. Sin embargo en muchos casos, por la velocidad de producción, la exactitud involucrada y/o las condiciones no ergonómicas, resulta imposible confiar la operación de inspección a una persona, por lo que resulta necesario desarrollar un sistema de inspección visual automático, que detecte y clasifique botellas de vidrio con fallas estructurales en el cuello, empleando técnicas, métodos y/o metodologías de visión artificial, procesamiento digital de señales e imágenes y de reconocimiento de patrones que tenga un costo relativamente bajo en comparación a los sistemas de inspección visual existentes en el mercado, que debido a que fueron diseñados para que puedan adaptarse a diversos procesos de inspección visual, tienen un precio mucho mayor, ya que éstos tienen un gran número de características y funciones configurables que en determinados procesos no son necesarias.

1.4 Hipótesis

Mediante técnicas de procesamiento digital de imágenes, como: filtros espaciales, detectores de bordes, y morfología matemática, aplicada a imágenes binarias, es posible detectar fallas estructurales en botellas de vidrio presentes en la corona inferior.

1.5 Objetivos

1.5.1 Objetivo general

Desarrollar y poner en operación una metodología basada en el análisis de imágenes digitales, para clasificar de forma automática botellas de vidrio, que presenten o no fallas estructurales en la zona del cuello conocida como corona inferior.

1.5.2 Objetivos específicos

- Utilizar algunas de las técnicas y metodologías ya propuestas sobre procesamiento digital de imágenes, para detectar fallas estructurales en el cuello de botellas de vidrio y analizar los resultados obtenidos, para decidir si son convenientes en la solución total o parcial del problema que se pretende solucionar en esta investigación.
- Proponer una metodología a nivel de software, con la que sea posible detectar defectos estructurales en la corona inferior de botellas de vidrio, para clasificarlas como defectuosas en caso de existir algún defecto ó como no defectuosa en caso contrario.

1.6 Alcances del trabajo

En este trabajo se desarrolla e implementa una metodología a nivel de software con la cual es posible clasificar botellas de vidrio, ya sea como buenas o malas, cuando éstas hayan o no sufrido alguna alteración física en su estructura durante el proceso de fabricación. Para poder obtener esos resultados se implementaron algunas técnicas de procesamiento digital de imágenes, morfología matemática y de reconocimiento de patrones.

1.7 Contribuciones

- La metodología propuesta para resolver el problema de clasificación de botellas de vidrio en esta investigación es diferente a las encontradas en los trabajos [1],[2],[3],[4] y [6], ya que en ésta, además de utilizar algunas operaciones morfológicas para eliminar y aumentar el tamaño de algunos bordes deseados y no deseados dentro de las imágenes, se emplea la transformada de Hough para detectar líneas en la parte superior de la imagen, de las cuales una sirve como referencia para ubicar el objeto de interés y extraer de la imagen el área donde es posible encontrar los píxeles que representan el defecto de la botella de vidrio.
- La metodología propuesta en esta investigación presenta una solución alternativa a algunas ya existentes, las cuales están limitadas a trabajar solo con botellas de vidrio lisas, es decir, aquellas que no tienen rosca ni corona inferior en el cuello de la botella.
- La segmentación y extracción del área de la imagen, en donde es posible localizar los píxeles que representan el defecto de las botellas, son invariantes ante el cambio de escala y translación horizontal de la botella, ya que se consideró que durante el traslado de las botellas sobre la banda transportadora estas no están fijas, lo cual provoca que estas puedan estar más cercanas o alejadas de la cámara, y al ser fotografiadas pueda haber variaciones en el tamaño de la botella dentro de la imagen.



1.8 Organización del documento

El presente trabajo está estructurado en seis capítulos y a continuación se describe el contenido de cada uno de ellos:

- En el primer capítulo denominado Introducción, se describen los antecedentes y el planteamiento del problema, se establece: el objetivo general, los objetivos particulares que se pretenden alcanzar, y además se describen: las contribuciones, los alcances del trabajo y el contenido de cada capítulo que conforma este trabajo.
- En el segundo capítulo, se describe el estado actual y los resultados de los trabajos reportados en la literatura que están relacionados con la clasificación de botellas de vidrio.
- En el tercer capítulo, se exponen los conceptos, definiciones, métodos, algoritmos y técnicas de: procesamiento de imágenes, de visión computacional y de reconocimiento de patrones en las que se fundamenta este trabajo.
- En el cuarto capítulo, se propone la metodología con la que se pretende dar solución al problema planteado en el capítulo 1, y se describen los experimentos realizados en cada una de las etapas que conforman dicha metodología.
- En el quinto capítulo, se describen los resultados de las pruebas realizadas en las etapas de la metodología de solución propuesta.
- Y finalmente, en el sexto capítulo se dan las conclusiones del trabajo y se describen las modificaciones que se pretenden realizar en el futuro.

Capítulo 2

En este capítulo se describe el estado actual y los resultados de los trabajos reportados en la literatura que están relacionados con la clasificación de botellas de vidrio..

Capítulo 2. Estado del arte

2.1 Trabajos relacionados con el problema planteado

Los trabajos que se han desarrollado y reportado en la literatura en los últimos años, que tienen alguna relación con el problema que se pretende resolver en ésta investigación, son los siguientes:

El trabajo [1] desarrollado por: Firmin C., Hamad D., Postaire J., G. Zhang, y Ruo Dan, en la Univ. des Sciences et Technologies de Lille, Francia en el año 1996, que tiene el título “Fault detection by a gaussian neural network with reject options in glass bottle production”. En este trabajo proponen el uso de una red neuronal de tipo gaussiana, para clasificar botellas de vidrio en dos clases que son: “botellas defectuosas” y “botellas sin defecto”, en base a los defectos presentes o no, en el cuello de éstas. Para lograr tal objetivo, adquieren imágenes del cuello de botellas de vidrio con y sin defecto, empleando una cámara CCD y un lámpara colocada frente a la botella para resaltar el defecto, después definen una región y un área de interés dentro de la imagen, y extraen cuatro de las características más discriminantes dentro de esa región, para entrenar a una red neuronal gaussiana con cuatro neuronas de entrada (una neurona por característica), una capa de salida con dos neuronas (una neurona para identificar botellas con defecto y otra para las botellas sin defecto).

En el trabajo [2] “A glass bottle defect detection system without touching”, desarrollado por: Yun-Yan Wang, Guang-Da Su y Hui-Min Ma, en la Universidad de Beijing, China en el año 2002, presentan el diseño de un sistema de inspección visual automático, para clasificar y separar botellas de vidrio que están sobre una línea de producción que tienen algún defecto físico en el cuello y/o corona. Dicho sistema cuenta con ocho cámaras de tipo CCD, de las cuales, seis están ubicadas en ambos lados de la línea de producción, separadas uniformemente de tal forma que permiten adquirir imágenes de todo el cuello de la botella y las otras dos cámaras están ubicadas en la parte superior de la línea de producción para capturar imágenes de la corona de la botella. Para reducir el tiempo de procesamiento de las imágenes, el sistema está integrado por dos computadoras, de las cuales una procesa las imágenes del cuello, y la otra las imágenes de la corona. La metodología que realizan para procesar las imágenes de cuello de la botella es la siguiente:

Determinan el área de interés de la imagen que será procesada, después definen y aplican un filtro promedio a la imagen para eliminar el ruido provocado por la iluminación y mejorar la visualización de los defectos (si es que estos existen), enseguida transforman la imagen capturada en niveles de gris a binaria, después hacen un barrido total de la región de interés y cuentan el número total de píxeles blancos presentes, y definen las condiciones que se deben cumplir para que la botella pueda ser clasificada como defectuosa o no defectuosa.

El trabajo [3] desarrollado por Domingo Mery y Olaya Medina, en el departamento de Ciencias de la Computación de la Universidad Católica de Chile, en el año 2004, que tiene el título: “Automated Visual Inspection of Glass Bottles Using Adapted Median Filtering”.

En este trabajo proponen el uso de técnicas de procesamiento digital de imágenes clásico, para automatizar el proceso de inspección y detección de defectos estructurales en el cuello de botellas de vidrio de tipo vinero. Esta propuesta se basa en una comparación que realizaron entre imágenes botellas de vidrio con defectos estructurales, adquiridas con una cámara digital e imágenes de rines de aluminio con algún defecto, adquiridas con rayos X.

Tal comparación les permitió observar la semejanza entre la representación visual de los defectos en ambas imágenes y tomar la decisión de emplear el filtro de mediana adaptada en la inspección de botellas de vidrio para detectar los defectos estructurales que éstas tengan.

La idea de ese método es generar filtros de mediana adaptable a la estructura del objeto que está bajo inspección (cuello de la botella de vidrio), para estimar una imagen de referencia “libre de defectos”, que es comparada con la imagen original del objeto que está bajo inspección, para determinar si existe una diferencia significativa entre las dos imágenes y así clasificar el objeto como “defectuoso” o “no defectuoso”.

En el trabajo [4], “Development of A Computerized Method to Inspect Empty Glass Bottle”, desarrollado por Huan-jun Liu y Yao-nan Wang, se propone un método para detectar defectos estructurales en las paredes y corona de botellas de vidrio. Para eso adecuan el lugar en donde se tomarán las imágenes fotográficas e instalan lámparas de tipo led en la parte superior de la botella de vidrio, con la finalidad, de evitar reflejos y sombras producidas por otras fuentes de iluminación de tipo incandescente y fluorescente, enseguida emplean filtros para detectar bordes de la imagen, y observan que en la imagen resultante aparecen algunos bordes que no representan los defectos de la botella, por lo que hacen uso de las operaciones morfológicas cerradura y apertura, para eliminar y/o rellenar algunos otros (defectos de la botella) de la imagen, después calculan el promedio de los niveles de gris de todo el cuerpo de la botella, y de las regiones detectadas a simple vista, que representan algún defecto en la botella y hacen una comparación entre el nivel promedio obtenido del cuerpo de la botella y de las regiones con algún defecto aparente, para determinar si la región realmente puede ser un defecto.

Finalmente, los resultados obtenidos de esas comparaciones se utilizan para definir un vector característico de la imagen, el cual es utilizado por una maquina de soporte vectorial difusa con red neuronal para clasificar la imagen como buena o con defecto.

Otro trabajo [5] reportado en la literatura en el año 2009 es el desarrollado por los alumnos: Ruiz Flores Braulio Andrés, Ullauri Ulloa Freddy Daniel y Chávez Patricia, de la Facultad de Ingeniería Eléctrica y Computación de la Escuela Superior Politécnica del Litoral, en Guayaquil Ecuador, que tiene el título: “Detección de fallas en envases de vidrio no cilíndricos utilizando localización de bordes mediante la herramienta Matlab”, en el cual proponen el análisis de envases de vidrio no cilíndricos, a través de la detección de bordes, empleando los operadores: Sobel, Prewitt y Roberts. Para desarrollar dicho trabajo realizan la siguiente metodología:



Proponen la creación de un ambiente de trabajo bajo el cual, puedan adquirir imágenes de los envases con el menor ruido posible provocado por las sombras y reflejos de luz. Para tal fin construyeron una caja con fondo de cartulina blanca, techo y paredes cubiertas de papel calco, además el piso y el fondo de la caja los unieron con una especie de curva para evitar la sombra provocada por esa unión.

Después adquieren y reducen a la mitad el tamaño de las imágenes capturadas por la cámara de tipo CCD, debido a que éstas tienen un tamaño demasiado grande, enseguida filtran las imágenes para eliminar el ruido conocido como sal y pimienta y emplean alguno de los operadores: Sobel, Prewitt o Roberts, para detectar los bordes de la imagen y realizan la convolución entre el gradiente de la imagen y una matriz definida experimentalmente, con el fin de suavizar la imagen y eliminan objetos cuyo tamaño es de 60 píxeles.

El tamaño de los objetos que eliminaron lo definieron después de realizar varias pruebas experimentales.

Después segmentan la imagen, ya que solo la zona central de ésta es de su interés, debido a que en esa zona se encuentra el envase de vidrio y finalmente detectan y etiquetan los objetos presentes en la imagen y con base en esto determinan si la botella tiene o no algún defecto físico.

En el trabajo [6] desarrollado por Genaro Gutiérrez Becerril, en el Centro de Investigación en Computación del Instituto Politécnico Nacional, en año 2008, que tiene el título: “Análisis y detección de fallas estructurales en envases de vidrio utilizando procesamiento digital de imágenes”, se propone una metodología para detectar defectos estructurales en envases de vidrio, que tienen una figura grabada en el cuerpo de la botella y para ello:

Se capturan imágenes en niveles de gris de envases de vidrio con y sin defectos, después se emplea el detector de esquinas de Harris para localizar un punto en común dentro de todas esas imágenes, enseguida se utiliza como referencia la posición de ese punto, para ubicar el envase dentro de la imagen y extraer una zona de la imagen de 200x 200 píxeles en donde es posible encontrar los píxeles que representan las fallas estructurales de las botellas, después se emplea la dilatación morfológica para tratar de eliminar o disminuir los píxeles que están dentro la zona extraída y representan la figura grabada en el cuerpo del envase.

Después se binariza la imagen (zona extraída), empleando el método de Otsu y se entrena una red neuronal de función de base radial con diez imágenes de las cuales cinco son de envases con defecto y el resto de envases sin defecto. Finalmente, se prueba el funcionamiento de la metodología y se obtiene un 100% de efectividad al clasificar cuarenta envases de vidrio con defecto y diez sin defecto.

2.2 Resultados reportados en trabajos relacionados

Los resultados obtenidos y reportados en los trabajos, previamente descritos al inicio de este capítulo son los siguientes:

En el trabajo [4] desarrollado por Huan-jun Liu y Yao-nan Wang con el título, “Development of A Computerized Method to Inspect Empty Glass Bottle”, para demostrar la efectividad del método propuesto, hacen una comparación entre los resultados obtenidos al clasificar botellas de vidrio con: una máquina de soporte vectorial, una red neuronal y su máquina de soporte vectorial difusa con red neuronal. Tales resultados se muestran la Tabla 1 y Tabla 2, en donde la Tabla 1, corresponde a los resultados obtenidos en la clasificación de botellas de vidrio con y sin defectos en el cuerpo, y la Tabla 2, corresponde a los resultados obtenidos en la clasificación de las botellas con y sin defectos en la corona.

Tabla 1. Resultados reportados en el artículo, “Development of A Computerized Method to Inspect Empty Glass Bottle”, en la clasificación de botellas de vidrio, con y sin defectos en el cuerpo.			
	Máquina de Soporte Vectorial	Red Neuronal	Máquina de Soporte Vectorial Difusa con Red Neuronal
Botellas sin defecto	95%	94.8%	97%
Botellas con defecto	96.8%	95%	97.7%

Tabla 2. Resultados reportados en el artículo, “Development of A Computerized Method to Inspect Empty Glass Bottle”, en la clasificación de botellas de vidrio, con y sin defectos en la corona.			
	Maquina de Soporte Vectorial	Red Neuronal	Maquina de Soporte Vectorial Difusa con Red Neuronal
Botellas sin defecto	96.5%	96.3%	97.7%
Botellas con defecto	97.4%	97.3%	98.5%

De acuerdo a los resultados mostrados en las Tablas anteriores se puede observar, que con el método propuesto por Huan-jun Liu y Yao-nan Wang, se obtuvo un porcentaje de efectividad por arriba del 97%, por lo que este método resultó más efectivo que los otros dos que se emplearon para hacer la clasificación.

Los resultados obtenidos y reportados en el trabajo [1] “Fault detection by a gaussian neural network with reject options in glass bottle production” se muestran en la Tabla 3, en donde se puede observar que los mejores resultados al clasificar las botellas de vidrio, se obtuvieron al entrenar la red neuronal propuesta en ese trabajo.

Tabla 3. Resultados reportados en el artículo “Fault detection by a gaussian neural network with reject options in glass bottle production”.		
	Red Neuronal sin opción de rechazo	Red Neuronal con opción de rechazo en la clasificación por distancia (r.d.) y ambigüedad (r.a.) de los datos.
416 botes sin defecto para la fase de entrenamiento de la RNA	Error = 1	Error = 0 r.d.= 1 r.a. = 1
60 botes con defecto para la fase de entrenamiento de la RNA	Error = 0	Error = 0 r.d.= 0 r.a. = 0
624 botes sin defecto para la fase de prueba de la RNA	Error = 2	Error = 0 r.d.= 5 r.a. = 1
60 botes con defecto para la fase de prueba de la RNA	Error = 1	Error = 0 r.d.= 3 r.a. = 0

Los resultados reportados por el Dr. Domingo Mery y Olaya Medina en el artículo [3], “Automated Visual Inspection of Glass Bottles Using Adapted Median Filtering”, se muestran en la Tabla 4, en donde se puede observar que el porcentaje de efectividad del método propuesto es del 85%, el cual se obtuvo al analizar cincuenta y seis imágenes, que corresponden a siete botellas de vidrio con y sin defecto, vistas desde ocho ángulos diferentes.

Tabla 4. Resultados reportados en el artículo, “Automated Visual Inspection of Glass Bottles Using Adapted Median Filtering ”.	
Imágenes analizadas (con defecto y sin defecto)	56
Imágenes sin defecto utilizadas para el entrenamiento	20
Imágenes utilizadas para etapa de inspección	36
Porcentaje de efectividad en la clasificación de imágenes reales	85%
Porcentaje de error en la clasificación de imágenes reales	4%

En el trabajo [5] “Detección de fallas en envases de vidrio no cilíndricos utilizando localización de bordes mediante la herramienta Matlab”, se utilizaron treinta y ocho imágenes de botellas de vidrio (con y sin defectos físicos), para realizar las pruebas de los operadores de detección de bordes, de los que concluyeron que los tres tienen el mismo desempeño ya que detectaron el mismo número de envases con y sin defecto, obteniendo así un porcentaje de efectividad en la clasificación del 89%. En la Tabla 5 se muestra la cantidad total de envases de vidrio, analizados y clasificados correcta e incorrectamente, en ese trabajo.

Tabla 5. Resultados reportados en el artículo “Detección de fallas en envases de vidrio no cilíndricos utilizando localización de bordes mediante la herramienta Matlab”.

Total de envases analizados	38
Envases con fallas analizados	24
Envases sin fallas analizados	14
Verdaderos positivos	21
Verdaderos negativos	13
Falsos positivos	1
Falsos negativos	3

En la Tabla 6, se muestran los resultados que se reportan en el artículo, “A glass bottle defect detection system without touching”, en donde se puede observar, que la cantidad total de botellas analizadas y clasificadas es de doscientas dos, de las cuales cien no tenían ningún defecto y las restantes, estaban afectadas por defectos físicos en el cuello y corona.

El porcentaje de efectividad en la clasificación de botellas de vidrio del sistema propuesto en este trabajo es del 98%.

Tabla 6. Resultados reportados en el artículo “A glass bottle defect detection system without touching”.

	Numero de Muestras	Aciertos en la clasificación	Porcentaje de error de la clasificación
Fracturas transversales en el cuello	62	61	1.6 %
Fracturas inclinadas en el cuello	20	19	5 %
Defectos en la corona	20	20	0 %
Botellas con defecto	102	100	1.96 %
Botellas sin defecto	100	98	2 %

En la Tabla 7, se muestran los resultados que se reportan en el trabajo, “Análisis y detección de fallas estructurales en envases de vidrio utilizando procesamiento digital de imágenes”, y se puede observar la cantidad de neuronas de la capa oculta de la red neuronal de función de base radial, utilizadas en cada una de las pruebas realizadas durante la fase de entrenamiento y prueba de dicha red neuronal, para clasificar cuarenta imágenes de botellas con defecto y diez sin defecto.

El porcentaje de efectividad en la clasificación de imágenes de botellas de vidrio del sistema propuesto en este trabajo es del 100%, tomando en consideración que la posición de los envases dentro de las imágenes debe ser la misma y además no debe haber variaciones en la escala las imágenes.

Tabla 7. Resultados reportados en el trabajo: “Análisis y detección de fallas estructurales en envases de vidrio utilizando procesamiento digital de imágenes”.

No. Prueba	Aciertos en		No. de neuronas en la capa oculta	No. de iteraciones de entrenamiento	Porcentaje de efectividad en la clasificación
	Envases sin defecto	Envases con defecto			
1	9	40	10	500	97.5 %
2	9	40	20	500	97.5 %
3	9	40	30	500	97.5 %
4	9	40	30	750	97.5 %
5	10	40	40	500	100 %

2.3 Uso de los trabajos desarrollados

Aunque las metodologías, métodos y/o técnicas desarrolladas en los trabajos consultados están de cierta forma relacionadas con el problema que se pretende solucionar en este trabajo, no es posible emplearlas directamente debido a que en esos trabajos analizan botellas de lisas y de mayor tamaño, por lo que no les provoca demasiado ruido la sombra de la botella y además no hay bordes creados por la propia estructura de la botella, que estén cercanos a la zona donde se ubican los defectos que desean detectar.

De acuerdo a lo anterior sólo se tomarán en consideración algunos de los métodos y/o técnicas empleadas en algunos de ellos para probar, analizar y comprobar si con ellos es posible dar solución total o parcial al problema planteado en este trabajo.

A continuación se describen los métodos, metodologías y técnicas de los trabajos consultados, que se tomarán en consideración, para tratar de dar solución al problema planteado en este trabajo.

Del artículo [2] se considerará la forma en la que segmentan la imagen y extraen la zona en que se encuentran (en caso de existir) los bordes que representan los defectos de las botellas.

Del artículo [4] se tomará en consideración el uso de las operaciones morfológicas: erosión, apertura y cerradura, debido a que con éstas logran cerrar los bordes que representan el defecto de la botella y eliminan algunos bordes que no representan ningún defecto y pueden causar problemas al clasificar la imagen y además gradiente morfológico para resaltar los bordes y defectos de las botellas dentro de las imágenes.

Del trabajo [5] se considera el uso de los operadores de Roberts, Prewitt y Sobel, para resaltar los bordes de las imágenes.

Del trabajo [6] se tomará en cuenta el uso de los detectores de esquinas: Harris y SUSAN, ya que con ellos obtienen un punto en común dentro de todas las imágenes, y lo utilizan como referencia para desplazarse dentro de la imagen y extraer la zona de la imagen en donde es posible encontrar los bordes que representan los defectos de las botellas.

Capítulo 3

En este capítulo se exponen los conceptos, definiciones, métodos, algoritmos y técnicas de: procesamiento de imágenes, de visión computacional y de reconocimiento de patrones en las que se fundamenta este trabajo.

Capítulo 3. Marco teórico

3.1 Etapas fundamentales del procesamiento digital de imágenes

El procesamiento digital de imágenes, tiene como objetivo mejorar la calidad de las imágenes, para su posterior utilización e interpretación. Por su parte, la visión por computadora tiene como objetivo reconocer y localizar objetos en un entorno dado, lo cual permite entender, estudiar y construir máquinas con capacidades similares a las humanas.

Las etapas fundamentales o básicas de un sistema para el análisis de imágenes son:

- Captura de imágenes.
- Preprocesamiento.
- Segmentación.
- Representación y descripción.
- Reconocimiento e interpretación.

3.1.1 Captura de imágenes

Para realizar la captura de las imágenes, es necesario contar con un dispositivo físico sensible a una determinada banda del espectro electromagnético. Actualmente existen en el mercado diferentes tipos de dispositivos que pueden ser utilizados para este fin, desde los más básicos, como por ejemplo los que responden únicamente a la presencia o ausencia de luz, hasta las cámaras CCD que son utilizadas frecuentemente en los sistemas de procesamiento de imágenes, debido a que brindan mayor resolución que las cámaras normales. La señal de salida de estos dispositivos puede ser digital o analógica. Para efectos del procesamiento digital, debe usarse un convertidor analógico/digital para las señales que no se encuentren en formato digital.

Definición: Una función de imagen o imagen es cualquier función real $i(x, y)$, con integral finita y soporte compacto S tal que, para todo punto $p \in S, x > 0, y > 0$.[7]

3.1.2. Preprocesamiento

El objetivo de la etapa de preprocesamiento de una imagen, es realzar características como los bordes o reducir el ruido presente en la imagen. Para obtener estas mejoras se utilizan filtros cuya respuesta se ajuste a las necesidades del proceso. Esta etapa es usada también para reducir la influencia de transformaciones geométricas sobre la imagen, como por ejemplo, las causadas por la óptica del sistema de captación.

En algunos de los casos, cuando se sabe bajo qué condiciones se capturan las imágenes, la etapa de preprocesamiento puede ser eliminada del lazo de reconocimiento del sistema. Si se sabe, por ejemplo, que las imágenes que se van a obtener son de buena calidad y ya, presentan algún tipo de segmentación, dicha etapa puede ser eliminada.

3.1.3 Segmentación

Cuando el ojo humano observa una escena, el cerebro humano divide esa escena en regiones. Una imagen es dividida en partes o regiones, donde los píxeles de estas regiones compartirán algunas características comunes como intensidad, color o textura. En un sistema de visión por computadora, la segmentación es una etapa determinante en el análisis de imágenes, cuyo objetivo es aislar a los objetos de interés, para después realizar la extracción y el análisis de sus características.

Existen una gran cantidad de algoritmos para realizar la segmentación de objetos en una imagen, pero su utilización depende de la aplicación específica.

En el caso más simple, cuando se sabe que los objetos se encuentran aislados y contrastan con el fondo de la imagen, el proceso de segmentación es relativamente sencillo de realizar.

Un umbralado seguido de un etiquetado de componentes conectadas, en muchos casos, puede dar los resultados esperados. Para el caso de objetos complejos o solapándose entre ellos, dicho proceso de segmentación es, sin duda, más difícil de realizar. La cantidad de objetos resultantes de la segmentación dependerá del nivel de abstracción o detalle que se desee tener de la escena observada.

La segmentación de una imagen puede definirse como el proceso que consiste en subdividir o particionar una imagen en sus regiones, componentes, partes u objetos. La segmentación puede también ser definida como: encontrar por medio de un algoritmo numérico, sus regiones homogéneas y sus bordes [7].

3.1.3.1 Segmentación basada en el umbralado

Las diferentes técnicas para segmentar mediante el uso de un umbral permiten separar un objeto dentro de la imagen del fondo que lo circunda, la técnica se basa en comparar alguna propiedad de la imagen con un umbral fijo o variable, si el valor de la propiedad de un píxel supera el valor de umbral, entonces el píxel pertenece al objeto, en caso contrario, el píxel pertenece al fondo [8].

Cuando la segmentación se realiza basada en el nivel de gris de la imagen, el valor del nivel de gris de cada píxel debe ser comparado con el umbral, para decidir si tal píxel pertenece al objeto o al fondo. La imagen de salida, es una imagen binaria en la cual aquellos píxeles cuyo valor es uno, pertenecen al objeto y los píxeles cuyo valor es cero pertenecen al fondo.

La selección del valor de umbral, se realiza generalmente a partir del histograma de la imagen. Si una imagen está compuesta de un objeto que aparece en la escena sobre un fondo, entonces es de esperar que el histograma sea bimodal, es decir, si por ejemplo el objeto es más claro que el fondo, en el histograma aparecerán dos picos, uno ubicado en los valores de gris más elevados correspondientes al objeto y el otro pico, para los niveles de gris más bajos, correspondientes al fondo.

En la Fig. 1, se muestra un histograma bimodal, en el cual el umbral se ubica entre los dos picos del histograma.

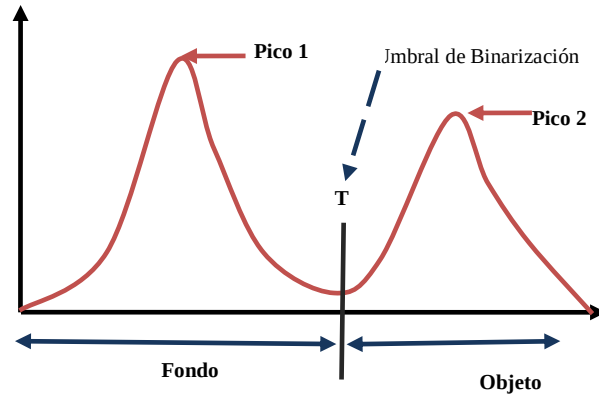


Figura 1. Histograma bimodal.

La selección automática del umbral, es un problema difícil debido a que el histograma no siempre es bimodal, en cuyo caso resulta necesario combinar la información espacial presente en la imagen, con la información referente al nivel de gris.

3.1.3.1.1 Método de Otsu

El método de umbralado de Otsu [9], proporciona un umbral óptimo para la segmentación. Este método está basado en el análisis discriminante. El umbral obtenido divide a la imagen en dos clases, C_0 y C_1 (objetos de interés y fondo) a partir del nivel de gris g . Es decir $C_0 = \{0, 1, 2, \dots, g\}$ y $C_1 = \{g + 1, g + 2, \dots, L - 1\}$, donde $L - 1$ es el nivel de gris máximo posible en la imagen. Sea σ_d^2 , σ_e^2 y σ_T^2 , la varianza dentro de la clase, la varianza entre las clases y la varianza total de la imagen, respectivamente.

El umbral óptimo se obtiene al maximizar alguna de las siguientes funciones (fórmula 1) con respecto a g .

$$\lambda = \frac{\sigma_e^2}{\sigma_d^2}, \quad \eta = \frac{\sigma_e^2}{\sigma_T^2}, \quad \kappa = \frac{\sigma_T^2}{\sigma_d^2} \quad (1)$$

El umbral es seleccionado maximizando la separación entre las clases a partir del histograma resultante de la composición de los histogramas de las clases presentes, es decir se maximiza la razón entre la varianza entre clases (σ_e^2), con respecto a la varianza dentro de las clases σ_d^2 . Mientras mayor sea la varianza entre clases σ_e^2 y menor la varianza dentro de la clase σ_d^2 , se discriminan mejor los grupos presentes teniendo en cuenta que entre ellos existen diferencias notables. Por otro lado, si dentro de los grupos la varianza es pequeña, es que existe similitud dentro de ellos, lo cual es lo deseado.

3.1.3.2 Etiquetado de regiones o componentes conectadas

Una vez que la imagen ha sido umbralada a través de alguno de los métodos, sus componentes conectadas pueden ser etiquetadas. Los píxeles en una región conectada forman una región que puede representar un objeto dado a reconocer. El etiquetado de componentes conectadas puede ser visto como una segunda etapa en el proceso de segmentado de una imagen. Un algoritmo de etiquetado de componentes conectadas (CC) es secuencial por naturaleza, debido a que la operación de encontrar componentes conectadas es una operación de tipo global.

Si se sabe que en la imagen hay un solo objeto, entonces el algoritmo de etiquetado puede ser innecesario.

Si por el contrario en una imagen hay varios objetos, y si su número, posición y orientación requieren ser obtenidos, entonces un algoritmo de etiquetado de CC es necesario.

Un algoritmo para encontrar CC en una imagen encuentra todas las CC en la imagen y origina una etiqueta para todos los píxeles en una misma componente. En la Fig. N se muestra una imagen sencilla y sus correspondientes CC etiquetadas.

Hay varios tipos de algoritmos para el etiquetado de CC entre ellos el recursivo [10], el iterativo [10] y el algoritmo de dos pasos que usa una tabla de equivalencias [11].

3.1.3.2.1 Algoritmo iterativo

Este algoritmo [10], no utiliza almacenamiento auxiliar para producir una imagen etiquetada a partir de una imagen binaria. Consta de tres pasos básicos, uno de etiquetado inicial, uno de propagación de arriba hacia abajo de etiquetas y finalmente, uno de propagado de etiquetas de abajo hacia arriba. En más detalle, dada una imagen binaria $b(x, y)$:

- 1) Barrer $b(x, y)$ hasta encontrar el píxel de tipo objeto (con valor 1 o L-1) aun no etiquetado y asignarle una nueva etiqueta E . Esto da como resultado la imagen $e_1(x, y)$.
- 2) Barrer $e_1(x, y)$ de arriba hacia abajo hasta encontrar un píxel etiquetado y propagar su etiqueta a sus vecinos, según la métrica elegida. Esto da como resultado la imagen $e_2(x, y)$.
- 3) Barrer $e_2(x, y)$ de abajo hacia arriba hasta encontrar un píxel etiquetado y propagar su etiqueta a sus vecinos, según la métrica elegida. Esto da como resultado la imagen $e_f(x, y)$.

En la Fig. 2(a) se muestra una imagen binaria y en las Figs. 2(b), 2(c) y 2(d) se muestra la secuencia de etiquetado hasta obtener la imagen final.

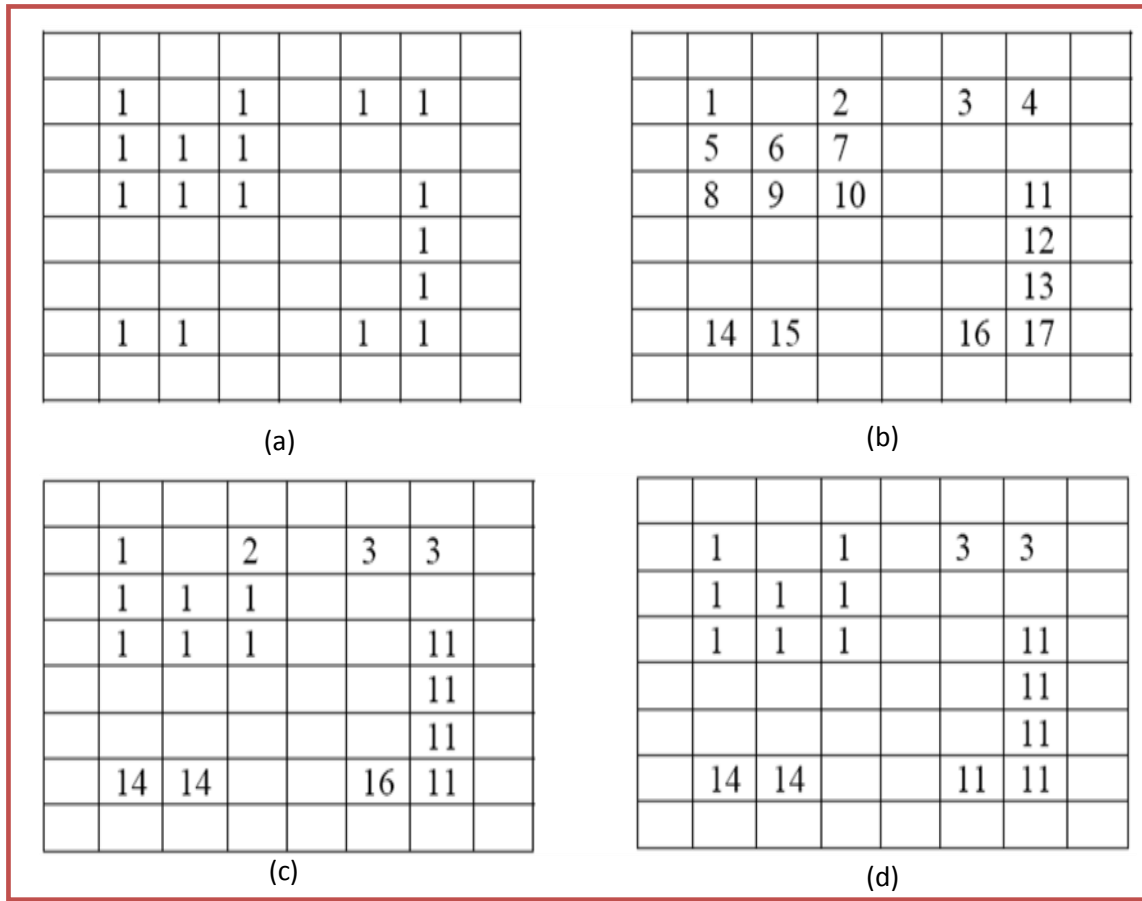


Figura 2. (a) Imagen binaria, (b), (c) y (d) Secuencia de etiquetado.

3.1.4 Representación y descripción

En esta etapa del análisis se aplican operadores sobre una imagen segmentada y no segmentada, lo que permite identificar posiciones de rasgos que ayudarán a la formación de hipótesis sobre la presencia de un objeto dado en la escena correspondiente. Los rasgos usados por el sistema, dependen del tipo de objetos a ser reconocidos, así como de la estructura del banco de modelos utilizado.

3.1.5 Reconocimiento e interpretación

En esta etapa se utilizan los rasgos extraídos de una imagen para asignar certidumbres a los objetos presentes en la escena correspondiente. Esta etapa es normalmente usada para reducir el espacio de búsqueda del reconocedor usando dichos rasgos. El banco de modelos es organizado empleando algún tipo de mecanismo de indexado que facilite la eliminación de candidatos no deseados. Finalmente, se usan los modelos de los objetos contenidos en el

banco de modelos para verificar las hipótesis generadas por la etapa generadora de hipótesis, que busca refinar las certidumbres de los objetos. EL sistema entonces selecciona el o los objetos con más altas certidumbres, basándose en la evidencia, para marcarlos como los correctos.

3.1.5.1 Métodos basados en matching

Los métodos basados en matching involucran la comparación de regiones de imágenes locales “muestras” (o características locales intrínsecas), para determinar cuáles regiones son más similares de acuerdo con el juicio desde algún criterio de matching. Los métodos de matching se clasifican como: locales, globales, basados en rasgos y correlación.

1. Los métodos locales toman punto a punto y lo conectan en la imagen independientemente de los demás.
2. Los métodos globales consideran los puntos como una sola unidad.
3. Los métodos basados en rasgos, por lo general producen una estimación de la profundidad con un carácter esparcido. Usualmente estos métodos son lentos y dependen de las condiciones externas, como el tamaño de los objetos y la iluminación, por lo tanto, el sistema de control debe preservar el esquema libre de cualquier tipo de ruido que sea generado durante el proceso de captura lo que se podría corregir con la implementación de filtros.
4. Contrario al anterior, los métodos basados en correlación producen una estimación de la profundidad en forma más densa, e implícitamente asumen que la disparidad es similar para cada píxel en la nueva ventana que se tomara en consideración.

3.1.5.1.1 Correlación

La correlación entre dos funciones es una medida de similitud o parecido que existe entre ellas [11]. La extensión a dos dimensiones es inmediata y en imágenes se pueden encontrar aplicaciones en el campo de formatos o ajuste de prototipos, donde el problema consiste en hallar el mayor parecido entre una imagen desconocida y un conjunto de imágenes conocidas. La imagen más parecida será la que obtenga correlación máxima.

El cálculo de la correlación (fórmula 2) se puede ver, como un problema de búsqueda de replicas, en una muestra $w(n,m)$ de tamaño $J \times K$ dentro de una imagen $f(n,m)$ de dimensión $P \times Q$ donde $J \leq P$ y $K \leq Q$.

$$f(s,t) \cdot w(s,t) = \sum_{t=-M}^M \sum_{s=-N}^N f(i,j)w(i+s,j+t) \quad (2)$$

Los límites de la sumatoria son tomados del tamaño de las imágenes f y w , donde $s = 0,1,2,\dots,N-1$ y $t = 0,1,2,\dots,M-1$, donde M es el ancho y N la altura en píxeles.

El proceso consiste en desplazar la imagen muestra sobre la imagen bajo análisis y sumar los productos de los valores de cada imagen. Esta función de correlación tiene la desventaja de ser sensible a los cambios de amplitud de $w(n, m)$ y $f(n, m)$. Una solución a este problema es la utilización del coeficiente de correlación cruzada normalizada definida por la fórmula 3.

$$\gamma(s, t) = \frac{\sum_n \sum_m [f(n, m) - f'(n, m)][w(n-s, m-t) - w']}{\left\{ \sum_n \sum_m [f(n, m) - f'(n, m)]^2 \sum_n \sum_m [w(n-s, m-t) - w']^2 \right\}^{1/2}} \quad (3)$$

Donde s y t toman los valores en el mismo rango expresado en la fórmula 2, f' y w' son el valor medio de los píxeles; con $w(n, m)$ y $f(n, m)$ en el valor medio de $f(n, m)$ se obtiene la ubicación actual de w .

El coeficiente de correlación está normalizado en el rango de -1 a 1, y es independiente de los cambios de escala aplicados a la amplitud de $f(n, m)$ y $w(n, m)$.

3.2 Imágenes a color y en niveles de gris

Una imagen RGB es una imagen multicanal, cada una de las cuales está, asociada a los colores rojo (R), verde (G) y azul (B), cada una de las bandas ocupa un plano homónimo independiente. Si la intensidad de las componentes rojo, verde y azul de cada píxel en una imagen son iguales, se forma una imagen acromática en tonos, o en niveles de gris.

Una imagen en tonos o niveles de gris es una imagen bidimensional en la que cada píxel solo presenta un valor de intensidad acotado entre 0 y 2^n , donde n es la cantidad de bits, utilizados para representar cada uno de los valores de intensidad. Por conveniencia, los valores extremos de este rango representan el negro (cero) y el blanco (255), respectivamente.

Para convertir una imagen en color (RGB) a niveles de gris se puede emplear la fórmula 4 que proviene del estándar NTSC (National Television Standard Code).

$$\text{Nivel de gris}_{p_{ij}} = 0.299R_{p_{ij}} + 0.587G_{p_{ij}} + 0.11B_{p_{ij}} \quad (4)$$

Otra conversión común se obtiene en forma aproximada, promediando los valores de las tres componentes R, G y B de un píxel y se obtiene con la fórmula 5.

$$\text{Nivel de gris}_{p_{ij}} = (R_{p_{ij}} + G_{p_{ij}} + B_{p_{ij}})/3, \quad (5)$$

donde p_{ij} , representa el píxel en cuestión.

Un caso especial de una imagen en niveles de gris, lo constituye una imagen binaria, en la cual, los píxeles solo pueden adquirir dos valores de intensidad en forma normalizada: 0 para el negro y 1 para el blanco.

3.3 Índices visuales

Un índice visual es una parte local de un objeto, con significado y que puede ser detectado. Con significado, implica que el índice visual se encuentre asociado con algún elemento interesante de la escena al momento de obtener la imagen del objeto. Ejemplos típicos de índices visuales son las variaciones de intensidad prominentes creadas por los contornos de los objetos de la escena. Que se pueda detectar, significa que debe existir al menos un algoritmo para su localización, de otra forma los índices visuales no son útiles [7].

3.3.1 Bordes y su detección

Un borde en una imagen es un cambio local significativo, normalmente asociado con una discontinuidad, ya sea en la imagen de intensidad o la primera derivada de dicha imagen. Discontinuidades de este tipo en una imagen pueden ser:

De tipo escalón. En éste caso la intensidad en la imagen cambia de manera abrupta de un valor en un lado de la discontinuidad, a otro valor muy diferente en el lado opuesto de la discontinuidad.

Lineales. En éste caso la intensidad en la imagen cambia de nuevo abruptamente de valor pero regresa al valor inicial en una distancia corta.

Debido a las componentes de baja frecuencia en la señal o el alisamiento introducido por la mayoría de los dispositivos de captación de imágenes, rara vez se presentan discontinuidades agudas en una imagen real. Los bordes de tipo paso aparecen como borde de tipo rampa, mientras que los bordes de tipo línea aparecen como bordes de tipo techo [12]. Los cambios de intensidad no se dan de manera instantánea sino a lo largo de distancias finitas.

El ruido en una imagen provoca también variaciones de intensidad, lo cual resulta en bordes espurios. Un buen detector de bordes debería ser capaz de suprimir la mayoría de estos bordes.

El proceso de detección de bordes en análisis de imágenes incorpora usualmente las siguientes tres etapas:

1. Alisado de ruido. El objetivo de esta etapa es suprimir tanto ruido como sea posible de la imagen, sin destruir bordes verdaderos.
2. Mejoramiento de bordes. La idea es convolucionar la imagen con un filtro sensible a cambios locales significativos. Se busca que su salida sea grande en las posiciones de los bordes y pequeña fuera de ellas, de manera que un borde pueda ser detectado como un máximo local a la salida del filtro.

3. Localización del borde. El objetivo en este caso es decidir cuáles de los máximos producidos por el filtro deben ser considerados como verdaderos bordes y cuales son causados por el ruido. Este paso involucra a su vez:

- Un esqueletizado de los bordes de forma que su grosor sea de un píxel (supresión de no máximos).
- El establecimiento de un valor mínimo que permita declarar un máximo local como borde (umbralado).

El alisamiento del ruido es necesario para mejorar el desempeño del detector de bordes. Para conseguir esto se puede usar cualquier filtro: lineal o no lineal.

El mejoramiento de bordes se requiere para enfatizar las posiciones en la imagen donde ocurren cambios significativos locales en los valores de intensidad. La mayoría de los enfatizadores de bordes fundamentan su funcionamiento en el cálculo del gradiente de la imagen.

La localización es necesaria para separar los máximos locales verdaderos (bordes verdaderos) de falsos máximos (bordes provocados por ruido).

3.3.1.1 Detectores de bordes basados en la primera derivada

Estos enfatizadores, basan su operación en el cómputo de la primera derivada de la señal. El gradiente como se sabe, es el equivalente de la primera derivada y se define como el vector

$$G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (6)$$

Su magnitud viene dada por

$$G[f(x, y)] = \sqrt{G_x^2 + G_y^2} \quad (7)$$

Que como se sabe, iguala la tasa máxima de crecimiento de $G[f(x, y)]$ por unidad de distancia en la dirección G .

En lugar de la formulación exacta dada por la ecuación (7), se pueden utilizar otras similares, pero con tiempos de cómputo obviamente más reducidos:

$$G[f(x, y)] = |G_x| + |G_y| \quad (8)$$

y

$$G[f(x, y)] = \max(|G_x|, |G_y|) \quad (9)$$

Otra cantidad importante usada para estimar la dirección del gradiente está dada como:

$$\theta(x, y) = \arctan \frac{G_y}{G_x}, \quad (10)$$

donde θ se mide con respecto al eje x .

3.3.1.1.1 El operador de Roberts

El operador cruzado de Roberts proporciona una aproximación sencilla para el cálculo de la magnitud del gradiente dado por:

$$G[f(x, y)] = f(x, y) - f(x + 1, y + 1) + f(x + 1, y) - f(x, y + 1) \quad (11)$$

Al usar máscaras de convolución, éste operador se transforma en:

$$G[f(x, y)] = G_x + G_y \quad (12)$$

Donde G_x y G_y son calculados usando las máscaras mostradas en la Tablas 8 y 9.

Tabla 1. Máscara G_y .	
0	1
-1	0

Tabla 2. Máscara G_x .	
1	0
0	-1

Nótese que en este caso las diferencias son computadas en la posición interpolada: $(x + 1/2, y + 1/2)$ y no en la posición esperada (x, y) .

3.3.1.1.2 El operador o enfatizador de Sobel

Una manera de evitar el cálculo del gradiente en una posición interpolada entre píxeles consiste en usar una vecindad de 8 píxeles, como la que se muestra en la Tabla 10.

Tabla 3. Vecindad 8 entre píxeles.		
a_0	a_1	a_2
a_3	(x, y)	a_4
a_5	a_6	a_7

En este caso, la magnitud del gradiente de Sobel puede obtenerse como sigue:

$$M = \sqrt{S_x^2 + S_y^2} \quad (13)$$

En este caso las derivadas parciales son calculadas con la siguiente ecuación:

$$S_x = (a_6 + ca_5 + a_4) - (a_0 + ca_1 + a_2) \quad (14)$$

Y

$$S_y = (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6) \quad (15)$$

La constante c toma el valor de 2.

Al igual que el caso del gradiente de Roberts, los gradientes de Sobel, S_x y S_y , pueden ser instrumentados por medio de las máscaras de convolución que se muestran en las Tablas 11 y 12.

Tabla 4. Máscara no. 2 de Sobel.		
-1	0	1
-2	0	2
-1	0	1

Tabla 5. Máscara no. 1 de Sobel.		
-1	-2	-1
0	0	0
1	2	1

Por sus características, el operador de Sobel es uno de los detectores de bordes más usados por la comunidad de tratamiento de imágenes [7].

3.3.1.1 .3 El operador de Prewitt

El operador de Prewitt utiliza las mismas ecuaciones que el operador de Sobel, excepto que la constante $c=1$. El par de máscaras a través de las cuales puede ser utilizado éste operador se muestran en las Tablas 13 y 14:

Tabla 6. Máscara no.2 de Prewitt.		
-1	0	1
-1	0	1
-1	0	1

Tabla 7. Máscara no. 1 de Prewitt.		
-1	-1	-1
0	0	0
1	1	1

Una propiedad interesante tanto del operador de Sobel como el de Prewitt es que ambos incorporan una etapa de alisado de ruido [7], cosa que no pasa, por ejemplo, con el gradiente de Roberts.

3.3.2 Rasgos puntuales

Al igual que los bordes, los vértices y las esquinas son también usados en la detección y reconocimiento de objetos. La experimentación ha demostrado que estos rasgos son bastante estables (detectables) a lo largo de secuencias de imágenes; esto los hace muy útiles, por ejemplo, en el seguimiento de objetos.

En lo que sigue se debe distinguir entre dos tipos de rasgos puntuales, los vértices y las esquinas. Los vértices son puntos de gran curvatura a lo largo del contorno (frontera) de la forma de un objeto, mientras que las esquinas son producidas por la intersección de tres o más superficies.

3.3.2.1 Detección de esquinas

Muchos métodos han sido desarrollados a lo largo de los últimos años para la detección de esquinas como por ejemplo [13], [14],[15], [16], [17], [18], [19] y [20].

Como en el caso de los detectores de bordes, el gradiente de una imagen es casi siempre la base de funcionamiento de la mayoría de los detectores de esquinas.

Para detectar una esquina en una imagen consideremos las dos imágenes de gradiente G_x y G_y , obtenidas a partir de una imagen. Tomemos cualquier punto ρ en la imagen, una imagen, una vecindad V alrededor de ρ y la siguiente matriz C , y veamos cómo C caracteriza la estructura de los niveles de gris de los píxeles dentro de V :

$$C(x, y) = \begin{bmatrix} \sum_V G_x^2 & \sum_V G_x G_y \\ \sum_V G_x G_y & \sum_V G_y^2 \end{bmatrix} \quad (16)$$

La clave está en los valores propios de C y su interpretación geométrica. Primero que todo, nótese que C es simétrica, pudiendo ser diagonalizada por una simple rotación de los ejes coordenados, luego entonces, sin pérdida de generalidad, se puede pensar que C toma la siguiente forma:

$$C = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (17)$$

Los dos valores propios λ_1 y λ_2 son ambos no negativos. Se tienen tres casos:

1. V es perfectamente uniforme (contiene niveles de gris iguales o muy parecidos), esto es: $\lambda_1 = \lambda_2$, ya que el gradiente es cero dentro de V .
2. V contiene un borde de tipo paso que cambia de negro a blanco, en este caso tenemos que $\lambda_2 = 0$ y $\lambda_1 > 0$; el vector propio asociado a λ_1 es paralelo al gradiente de la imagen.

3. V contiene la esquina de un cuadrado negro con un fondo blanco; como hay dos direcciones principales en V , se esperaría que $\lambda_1 \geq \lambda_2 > 0$; entre más grandes son los valores propios más pronunciada será la esquina. Una esquina es así identificada por dos bordes fuertes como $\lambda_1 \geq \lambda_2$, una esquina es la posición en donde el valor propio más pequeño λ_2 es lo suficientemente grande.

3.3.2.1.1 Detector de esquinas de Harris

El detector de esquinas de Harris [21] es, probablemente, el más comúnmente usado, debido a su elevada invariancia ante la escala, rotación, cambios de iluminación y ruido en la imagen. Este detector está basado en la siguiente matriz $C(x, y)$ que se calcula sobre una subventana $p \times p$ para cada punto de interés en la posición (x, y) .

$$C(x, y) = \begin{pmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{pmatrix} \quad (18)$$

donde I_x , I_y son los gradientes de la imagen en dirección horizontal y vertical respectivamente. Sean λ_1 y λ_2 los valores propios de la matriz $C(x, y)$, la función de autocorrelación R tendrá un pico si ambos valores propios son altos. Esto significa que desplazamientos en cualquier dirección producirán un incremento significativo, indicando que se trata de una esquina.

$$R = \lambda_1 \lambda_2 - \kappa(\lambda_1 + \lambda_2) \quad (19)$$

Es decir que el detector de esquinas de Harris es capaz de detectar esquinas o puntos aislados, en la imagen, existe una variación de intensidad en todas las direcciones esto permite medir los cambios cuando deslizamos una ventana horizontal y una vertical alrededor del punto de interés.

3.3.2.2 Detección de líneas

Las líneas son otros índices visuales bastante usados en análisis de imágenes para seguir objetos a lo largo de una secuencia de imágenes, así como en el modelado automatizado de dichos objetos, para su posterior detección o reconocimiento.

Segmentos de línea en una imagen son en forma normal obtenidos a partir de los bordes previamente detectados en dicha imagen. Para poder determinar la presencia de segmentos de línea recta en una imagen se puede usar algún algoritmo de umbralado por histéresis como [7].

Uno de los problemas de la mayoría de los algoritmos para detectar líneas es la presencia indeseable, en muchas ocasiones, de rompimientos. Esto puede deberse, a la mala selección

del umbral para la detección final de los bordes, al mal contraste en las imágenes, al ruido, incluso a las sombras y la dirección de la iluminación.

Una técnica desarrollada ya hace algunos años que ha demostrado su efectividad en la detección de instancias de curvas complejas en una imagen es la transformada de Hough [11], [22], [23], [24]. La idea consiste en mapear puntos de tipo imagen (normalmente bordes), en un espacio de parámetros (el espacio de Hough). El principio de funcionamiento es muy sencillo y puede ser descrito como sigue:

Supóngase que $b(x, y)$ es la imagen de bordes adelgazados a la salida de un método de supresión de no máximos y además que para el caso de interés de detección de líneas rectas estas pueden expresarse con la siguiente ecuación:

$$x \cos \theta + y \sin \theta = \rho \quad (20)$$

Si además, $\theta \in [0, \pi]$, los parámetros de la recta son únicos. Luego entonces, a cada recta en el plano $x - y$ le corresponde un punto en el plano $\rho - \theta$. Como se muestra en la Fig. 3, la transformada de Hough nos permite mapear rectas sobre puntos.

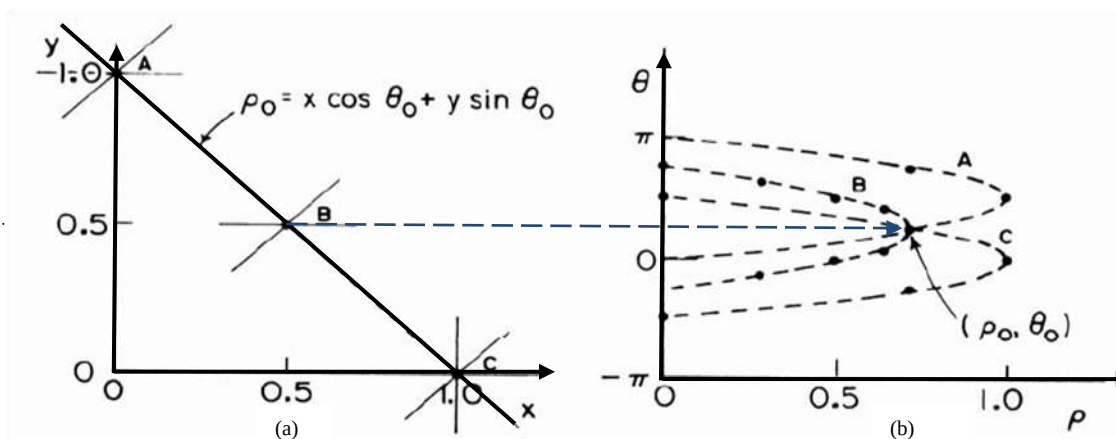


Figura 3. (a) Puntos colineales (A, B y C) y (b) Transformada de Hough de (a).

Considérese un solo punto de la imagen $b(x, y)$. De acuerdo con la ecuación, éste punto se caracteriza por su posición en la imagen x_i, y_j y por su orientación θ_i . Luego entonces se puede definir una recta que pase por ese punto en la dirección del contorno como:

$$x \cos \theta_i + y \sin \theta_i = \rho_i \quad (21)$$

Supóngase ahora que la imagen contiene n puntos colineales. De lo visto se tiene que $\theta_1 = \theta_2 = \dots = \theta_n$ y $\rho_1 = \rho_2 = \dots = \rho_n$. Estos puntos se corresponderán con la familia de senoides en espacio de parámetros $\rho - \theta$. Todos estos senoides pasarán por el mismo punto (θ, ρ) en el espacio de parámetros $\rho - \theta$. Este punto corresponde, a su vez, a los parámetros de la línea original en $b(x, y)$.

Para detectar segmentos de línea recta sobre una imagen $b(x, y)$ se puede aplicar el siguiente procedimiento:

- Primeramente se calcula la transformada de Hough a partir de la imagen $b(x, y)$.
- Enseguida, en la imagen de Hough se detectan los picos de mayor intensidad (arriba de un umbral predeterminado). Estos picos corresponden a las posiciones en la imagen de Hough con más votos.
- Finalmente esos puntos son transformados en líneas rectas y enviados sobre la imagen original.

En la práctica, el procedimiento descrito puede aplicarse como sigue.

Dada una imagen binaria $b(x, y)$ de puntos borde, con $b(x, y) = 1$ para un borde y $b(x, y) = 0$ en caso contrario:

- Primeramente definir dos arreglos: ρ_d y ϕ_d , discretizados a intervalos de ρ y ϕ , $\rho \in [-\sqrt{M^2 + N^2}, \sqrt{M^2 + N^2}]$ $\phi \in [-\pi, \pi]$, R y P sus números de elementos.
- Enseguida, discretizar el espacio de parámetros ρ y ϕ a pasos $\delta\rho$ y $\delta\phi$. En conjunto estos dos parámetros determinaran la precisión deseada.
- Después, suponer que cada celda en el espacio de parámetros discretizado $\rho - \phi$ es un contador. Igualar el valor de cada celda a cero, $A(x, y) = 0, i = 1, \dots, R, j = 1, \dots, P$.
- Luego para cada punto en $b(x, y)$, con coordenadas (x, y) , tal que $b(x, y) = 1$ y para $j = 1, 2, \dots, P$:
 - a) Sea $\rho = x \cos \phi_d(j) + y \sin \phi_d(j)$;
 - b) Encontrar el índice i del ρ_d más cercano a ρ , e incrementar en uno el contador $A(i, j)$ correspondiente.
- Enseguida, de entre los máximos locales en el arreglo $A(i, j)$, seleccionar aquellos para los cuales $A(i, j) > u$, (u umbral seleccionado). Los índices correspondientes serán los valores de los parámetros ρ y ϕ de cada línea detectada en $b(x, y)$.
- Finalmente determinar si hay segmentos de línea asociados con estos máximos locales, asimismo, especificar dónde comienzan y donde terminan. Para realizar lo anterior en cada máximo local hay que efectuar lo siguiente:
 - a) Localizar en $b(x, y)$, todos los píxeles diferentes de cero que contribuyan a ese máximo.
 - b) Encontrar y llenar los espacios entre segmentos de recta.
 - c) Dibujar los segmentos de recta sobre la imagen original.

3.4 Morfología matemática

La descripción básica de la morfología matemática descansa en la “teoría de conjuntos” cuyos primeros trabajos se deben a Minkowsky y Hadwiger [25]. La continuación de estos trabajos de investigación, bajo la impulsión y reformulación de Matheron y Serra, se darían posteriormente a conocer bajo la denominación de Morfología Matemática como una técnica no lineal de tratamiento de señales.

Actualmente el ámbito y alcance de los procesamiento morfológicos es tan amplio como el propio procesamiento de imágenes. Se pueden encontrar aplicaciones tales como la segmentación, restauración, detección de bordes, aumento de contraste, análisis de texturas, compresión, entre otros. Las obras Matheron [26] y Serra [27], contienen en detalle la definición de los conceptos básicos de la morfología matemática.

3.4.1 Transformaciones morfológicas elementales

La transformación morfológica es la extracción de estructuras geométricas en los conjuntos sobre los que se opera, mediante la utilización de otro conjunto de forma conocida denominado elemento de estructura. El tamaño y la forma de este elemento se selecciona a priori de acuerdo a la morfología del conjunto sobre el que va a interaccionar y de acuerdo a las formas que se desean extraer. Ejemplos básicos de elementos de estructura se muestran en la Fig. 4.



Figura 4. Formas básicas de elementos de estructura planos.

La idea básica es probar la imagen con un elemento de estructura y cuantificar el modo en que éste, está contenido dentro de la imagen.

En una determinada ubicación dentro de la imagen pueden pasar dos cosas: que el elemento de estructura esté contenido o que no lo esté. Marcando las ubicaciones en que está contenido se obtiene información estructural de la imagen. Esta depende de la forma y del tamaño del elemento de estructura. La característica de estar contenido depende de la relación de subconjunto.

En adelante, A es la imagen de entrada y B el elemento de estructura.

3.4.2 Dilatación

La dilatación es una transformación morfológica que combina dos conjuntos usando la suma vectorial de elementos de un conjunto [28].

Definición:

$$A \oplus B = \{a + b : a \in A \text{ y } b \in B\} \quad (22)$$

o bien,

$$x \in A \oplus B \Leftrightarrow x = a + b, a \in A, b \in B \quad (23)$$

Es decir, el conjunto de todos los posibles vectores suma de pares de elementos, uno perteneciente a A y el otro a B .

En la Fig. 5(c), se muestra el resultado obtenido con la operación morfológica: dilatación del conjunto de puntos $A = \{(0,1), (1,1), (2,1), (2,2), (3,0)\}$, con el elemento de estructura $B = \{(0,0), (0,1)\}$.

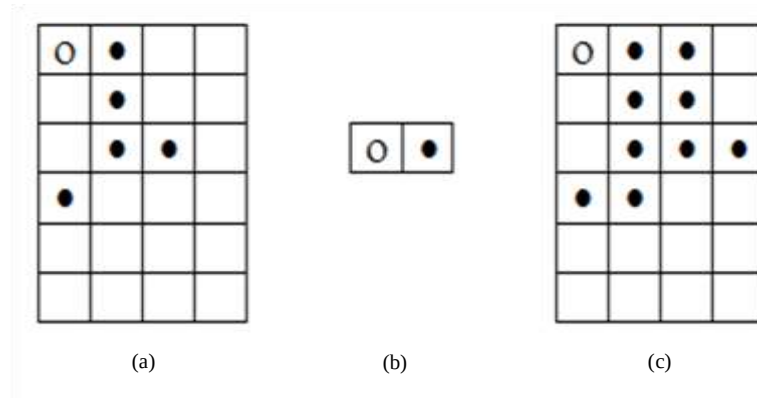


Figura 5. (a) Conjunto A, (b) Elemento de estructura y (c) Dilatación de (A) con (B).

En la Fig. 6(c), se muestra el resultado de la operación morfológica: dilatación de la Fig. 6(a), con un elemento de estructura circular de 3 píxeles de radio (Fig. 6(b)).

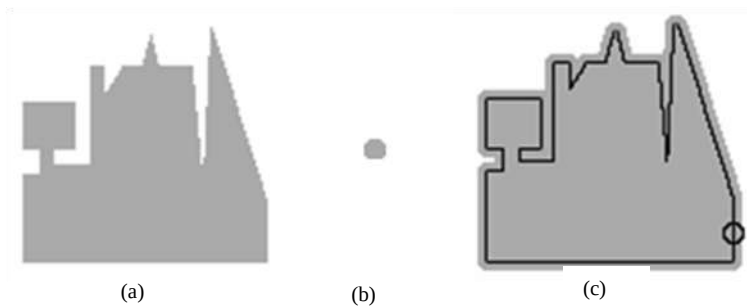


Figura 6. (a) Imagen binaria, (b) Elemento de estructura y (c) Dilatación de (a) con (b).

3.4.3 Erosión.

Es la transformación que combina dos conjuntos usando la resta vectorial de elementos de conjuntos [28]. Si A y B son conjuntos en el espacio euclidiano de dimensión N , entonces la erosión de A por B es el conjunto de todos los elementos x para los cuales $x + b \in A$, y para todo $b \in B$.

Definición:

$$A \ominus B = \{x | (B)x \subseteq A\} \quad (24)$$

es decir, está formada por el conjunto de puntos x que hacen que B , trasladado según el vector x , esté completamente contenido dentro del conjunto A .

En la Fig. 7 (c), se muestra el resultado obtenido con la operación morfológica: erosión del conjunto de puntos $A = \{(1,0), (1,1), (1,2), (1,3), (1,4), (1,5), (2,1), (3,1), (4,1), (5,1)\}$ con el elemento de estructura $B = \{(0,0), (0,1)\}$.

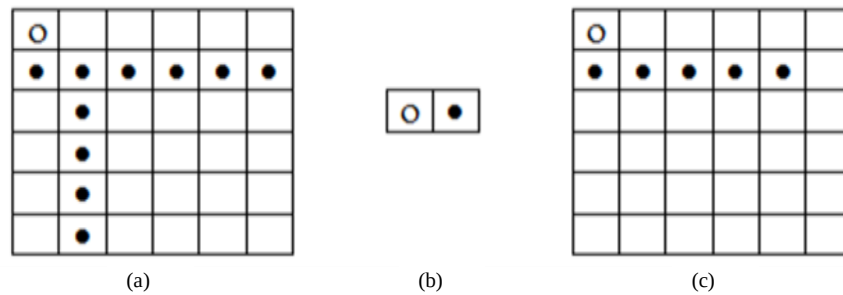


Figura 7. (a) Conjunto A, (b) Elemento de estructura y (c) Erosión de (A) con (B).

$$A \ominus B = \{(1,0), (1,1), (1,2), (1,3), (1,4)\}$$

En la Fig. 6(c), se muestra el resultado de la operación morfológica: erosión de la Fig. 6(a), con un elemento de estructura circular de 3 píxeles de radio (Fig. 6(b)).

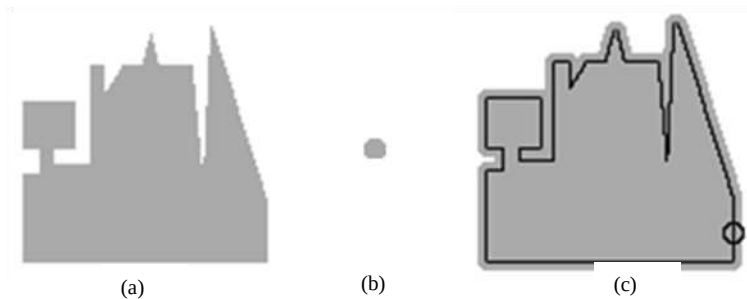


Figura 8. (a) Imagen binaria, (b) Elemento de Estructura y (c) Erosión de (a) con (b).

Como se vio en los párrafos anteriores, la dilatación y la erosión están muy relacionadas con la forma; la primera operación expande la imagen mientras que la segunda la contrae. La dilatación y la erosión usualmente se utilizan en pares, es decir la dilatación seguida de

la erosión o viceversa. En cualquier caso, el resultado de la aplicación sucesiva de erosiones y dilataciones es una eliminación de detalles menores que no distorsiona la forma global del objeto.

3.4.4 Apertura

La apertura de un conjunto A por el elemento de estructura B , se define como:

$$A \circ B = (A \ominus B) \oplus B \quad (25)$$

Es decir, la apertura de A por B es la erosión de A por B seguida por la dilatación del resultado con el elemento de estructura B . [28]

En la Fig. 9(c), se muestra el resultado obtenido con la operación morfológica: apertura del conjunto A (Fig. 9(a)), con el elemento de estructura B (Fig. 9(b)).

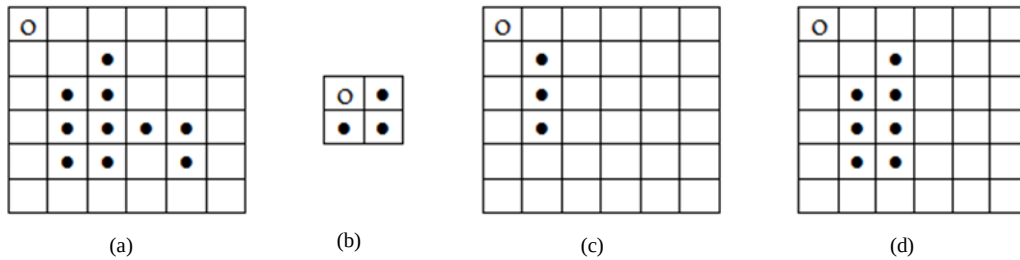


Figura 9. (a) Conjunto A, (b) Elemento de estructura, (c) Erosión de (a) con (b) y (d)

Apertura de (a) con (b).

3.4.5 Clausura

La clausura de un conjunto A por el elemento de estructural B , se define como:

$$A \cdot B = (A \oplus B) \ominus B \quad (26)$$

En otras palabras la clausura es la dilatación de A por B seguida por la erosión del resultado con el elemento de estructura B .

En la Fig. 10(c), se muestra el resultado obtenido con la operación morfológica: clausura del conjunto A (Fig. 10(a)), con el elemento de estructura B (Fig. 10(b)).

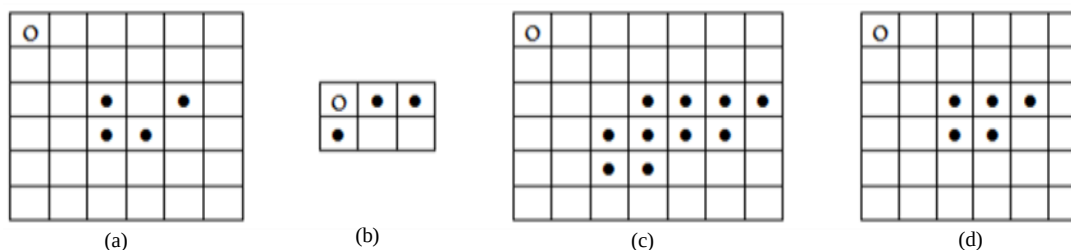


Figura 10. (a) Conjunto A, (b) Elemento de estructura, (c) Dilatación de (a) con (b) y (d) Clausura de (a) con (b).

La apertura generalmente suaviza los contornos de un objeto y elimina protuberancias finas. La clausura también suaviza los contornos pero, contrariamente a la apertura, generalmente fusiona las hendiduras finas y largas presentes en los objetos, elimina agujeros pequeños y rellena brechas en el contorno.

3.5 Redes neuronales artificiales

Desde tiempos remotos, los seres humanos han tenido la ilusión de construir máquinas inteligentes, que se comporten como el cerebro humano, que lo emulen en las actividades que cotidianamente puede realizar una persona normal. Las leyendas sobre autómatas, robots y hombres mecánicos han pululado por miles de años, y continúan en nuestros días. Una preocupación real de quienes han estado involucrados en los intentos de crear modelos de máquinas inteligentes, es tratar de entender cómo funcionan las neuronas del cerebro humano y sus mecanismos de operación e interacción.

El año de 1943 ha sido marcado por un hecho de gran trascendencia. Dos científicos estadounidenses de la University of Illinois, el sicólogo Warren Sturgis McCulloch y un joven matemático autodidacta de 17 años llamado Walter Pitts (de tortuosa, breve y misteriosa vida), publicaron el primer modelo matemático serio de neurona artificial de que se tenga noticia: la neurona de McCulloch-Pitts [29].

Dadas las condiciones de guerra que imperaban en el mundo en esa época, el artículo pasó casi desapercibido; no obstante, catorce años después, la neurona de McCulloch-Pitts fue la base fundamental sobre la que Frank Rosenblatt, psicopatólogo experimental del Cornell Aeronautical Laboratory, en 1957 inventó el perceptrón [30].

La relevancia del perceptrón reside en que fue la primera máquina que aprendía artificialmente: ha pasado a la historia como la primera red neuronal (basada en las neuronas de McCulloch-Pitts) que clasificaba patrones de manera automática. El perceptrón de Rosenblatt despertó gran entusiasmo en la comunidad científica mundial, y en todo el orbe los científicos y tecnólogos iniciaron una frenética búsqueda de reglas de aprendizaje para mejorar la eficacia y el rendimiento de los perceptrones.

Así, en 1960, Bernard Widrow y Marcian Hoff de Stanford, desarrollaron el ADALINE (Adaptive Linear Element), aplicando la regla delta de aprendizaje [31]. Fue tal el éxito de

ADALINE cuando Widrow y Hoff aplicaron su red neuronal a la eliminación de ecos en las líneas telefónicas, que poco después apareció una extensión llamada MADALINE (Multiple ADALINES) [32].

El entusiasmo duraría sólo nueve años más, período en el cual muchos científicos fracasaron al intentar aplicar sus modelos de reglas de aprendizaje, ideados al calor de los éxitos de las redes de neuronas de McCulloch-Pitts, cristalizadas en los perceptrones y las redes de Widrow y Hoff.

En 1969 se publicó un libro: su título era *Perceptrons* y sus autores Marvin Minsky y Seymour Papert, quienes eran dos preclaros científicos del afamado MIT (Massachusetts Institute of Technology) [33]. Según el criterio de muchos autores serios en el área de las redes neuronales artificiales, la aparición de este libro marcó el nacimiento de una era de oscurantismo en el área, en virtud de que Minsky y Papert plantearon de manera muy pesimista las limitaciones de los perceptrones, al grado de afirmar que era estéril buscar el desarrollo de perceptrones más eficaces que los que existían en esa época.

Habían de pasar trece años para que esa época de pesimismo terminara y se iniciase una nueva era para las redes neuronales artificiales [34]. Al igual como lo hizo con las memorias asociativas, el responsable del resurgimiento de las redes neuronales en 1982 fue el físico estadounidense John Hopfield.

Si 1972 fue el año de los pioneros en el área de las memorias asociativas, 1982 fue el año de John J. Hopfield. Su artículo de ese año [35], publicado por la prestigiosa y respetada National Academy of Sciences (en sus *Proceedings*), impactó positivamente y trajo a la palestra internacional su memoria asociativa, que al mismo tiempo es una red neuronal. Dos años después, publicaría su segundo artículo, donde presentaba una extensión de su modelo original [36].

Las voces autorizadas de los editores del compendio *Neurocomputing* [34] aseguran que la era moderna de las redes neuronales (y de las memorias asociativas) nace a raíz de la publicación del artículo de Hopfield; afirman que el éxito de este artículo se debe en gran parte a que, además de tener un estilo claro, coherente y sofisticado, fue escrito por el distinguido y reconocido físico John J. Hopfield, en cuyas manos la teoría se convierte en algo legítimo y respetable.

La formación como físico del autor queda de manifiesto cuando declara que en los sistemas físicos constituidos por un gran número de elementos simples, las interacciones entre estos elementos dan lugar a fenómenos colectivos (las orientaciones de los dominios en sistemas magnéticos y los patrones de vórtices en sistemas de fluidos ejemplifican esta afirmación).

A partir de estas consideraciones, Hopfield se pregunta si la interacción de elementos simples de procesamiento similares a las neuronas, cuyo modelo simplificado se conocía desde hacía cuatro décadas [29], da lugar a la aparición de propiedades computacionales colectivas, tales como la estabilidad de memorias; acto seguido, el autor afirma que en efecto, su artículo de 1982 demuestra que este tipo de propiedades computacionales aparecen espontáneamente.

El trabajo de Hopfield ha tenido un gran impacto en las áreas de memorias asociativas y redes neuronales. Si se realiza una búsqueda en Google con la palabra clave "Hopfield"; es sorprendente la cantidad de trabajos científicos, ingenieriles y de aplicación a que ha dado lugar el conjunto de ideas plasmadas por John Hopfield en sus trabajos científicos.

Entre los científicos que incursionaron de nuevo en redes neuronales motivados por el trabajo de Hopfield, algunos se dedicaron a tratar de darle un nuevo impulso al *longevo perceptrón*: tres científicos estadounidenses (Rumelhart, Hinton & Williams) lo lograron al inventar en 1986 el algoritmo *backpropagation* para entrenar *perceptrones multicapa* [37].

No obstante los éxitos que ha tenido la aplicación de *backpropagation*, un nuevo impulso a las redes neuronales apareció en 1989 cuando Gerhard X. Ritter y sus colaboradores crearon y desarrollaron las redes neuronales morfológicas [38] [39]. Estas redes neuronales usan máximos o mínimos de sumas, a diferencia de las redes neuronales anteriores, las cuales basan su funcionamiento en sumas de productos. Con este nuevo esquema y con la incorporación de las nuevas ideas sobre redes neuronales dendríticas, las redes neuronales morfológicas han igualado y en ocasiones superado al rendimiento de las redes neuronales *backpropagation*.

Del mismo modo como las operaciones binarias Alfa y Beta presentadas por primera vez en [40] dieron lugar al surgimiento de un nuevo tipo de memorias asociativas que igualaron y mejoraron a las memorias asociativas morfológicas, estas mismas operaciones binarias Alfa y Beta han sido la base teórica que sustentar el surgimiento, desarrollo y aplicaciones de las redes neuronales Alfa-Beta [41]. Los creadores de este nuevo modelo de redes neuronales tiene como una de sus principales motivaciones el generar un producto de investigación autóctono, original, que eventualmente contribuya con su granito de arena a avanzar en el afán de lograr ese noble propósito de independencia científica y tecnológica para nuestro país[42].

3.5.1 Red Neuronal de función de base radial (RBF)

Broomhead y Lowe, 1988, introducen un método alternativo al *perceptrón multicapa (MLP)* (ej.: *backpropagation*) para hacer ajuste a funciones no lineales. Esto es clasificación no lineal: las redes RBF.

A diferencia de la disposición que se tiene en la funciones de activación que permite construir modelos de entrenamiento mediante *backpropagation*, estas nuevas redes basadas en RBF construyen sus modelos con funciones de activación que son diferente tanto en la capa oculta como la de salida. Esto es, una red RBF está diseñada con neuronas en la capa oculta activadas mediante funciones radiales de carácter no lineal con sus centros gravitacionales propios y en la capa de salida mediante funciones lineales.

A diferencia de las MLP, el modelo clásico de las redes RBF está construido con una arquitectura rígida de tres capas: la de entrada, la oculta y la de salida.

En general, una red RBF tiene un mejor desempeño con un mayor volumen de datos de entrenamiento.

La construcción de una red RBF requiere de una mayor cantidad de neuronas en los nodos ocultos que en las redes que usan backpropagation.

Aunque las redes RBF no son comúnmente utilizadas en aplicaciones que impliquen un alto volumen de patrones de entrenamiento, se le reconoce como una red con una alta eficiencia en la fase de entrenamiento.

Método alternativo para aproximar funciones y clasificar patrones.

3.5.1.1 Como funciona una RBF

Tal como ya se dijo anteriormente, una red RBF, a diferencia de una MLP, está conformada de tres capas.

1. La capa de entrada que sirve para los ejemplos o patrones de entrenamiento y prueba.
2. La capa oculta completamente interconectada entre todos sus nodos con la capa de entrada y activada a través de la función radial (gaussiana).
3. La capa de salida, también completamente interconectada a la capa oculta y activada a través de una función lineal continua.

El entrenamiento, a diferencia de la red usando backpropagation, es solamente hacia delante. De este modo, la salida z de una red RBF, en general, está influenciada por una transformación no lineal originada en la capa oculta a través de la función radial y una lineal en la capa de salida a través de la función lineal continua.

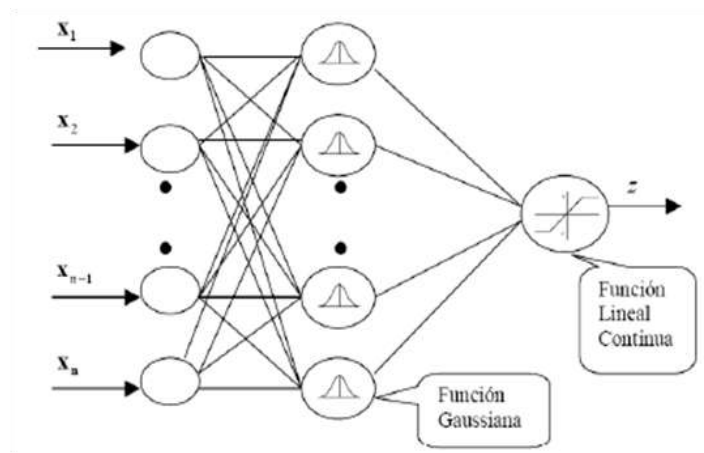


Figura 11. Estructura de una red de base radial.

3.5.1.2 Topología de la RBF

- Los nodos ocultos contienen una función base radial, la cual tiene como parámetros a centro y ancho.
- Existe un centro para cada función radial involucrada en la capa oculta.
- Regularmente, definen un vector de la misma dimensión del vector de entrada y hay normalmente un centro diferente por cada nodo de la capa oculta.
- Por otro lado, el ancho es el término empleado para identificar a la amplitud de la campana de gauss originada por la función radial. Es decir, la desviación estándar de la función radial. Algunos autores consideran a este ancho como un valor constante para cada una de las funciones radiales consideradas en la capa oculta y de este modo, así contribuiría a simplificar los pasos de construcción del modelo de entrenamiento de la red.

El primer cálculo efectuado en la capa oculta es hallar en un nodo de la capa oculta la distancia radial (distancia euclidiana) d entre el vector de entrada x , con n observaciones, a ese nodo en particular y el centro de gravedad c de ese mismo nodo. Es decir:

$$d = |x - c| = \sqrt{(x_1 - c_1)^2 + (x_2 - c_2)^2 + \dots + (x_n - c_n)^2} \quad (27)$$

Este valor d es un componente de la entrada para activar la función radial $G(\bullet)$. Este valor establece la principal diferencia con las redes MLP, entre ellas la backpropagation, quienes incluyen el producto interno en sus capas ocultas de las entradas por sus respectivos pesos. En cuanto a la función radial $G(\bullet)$, siendo una de las más comunes $\exp(-r^2)$, siendo r el contenido evaluado en cada nodo de la capa oculta. En este caso particular, el contenido evaluado en cada nodo es la distancia euclidiana d . De ahí la expresión, entonces sería $\exp(-d^2)$. Una de las derivaciones del modelo RBF es emplear el ancho (desviación estándar) para activar la función $G(\bullet)$. En este caso se estaría trabajando con algo como $\exp(-d^2/a)$, donde a es el ancho para ese nodo oculto.

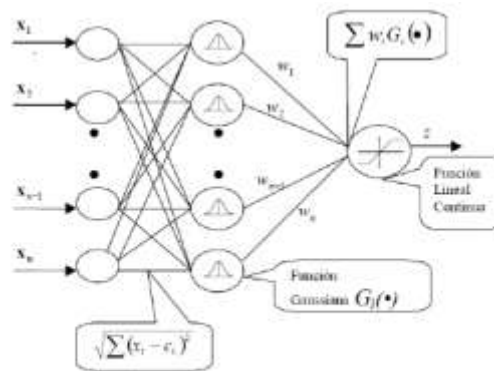


Figura 12. Representación de función de base radial.

Entre la capa oculta y la capa de salida se derivan un conjunto de pesos w que se verían afectados de acuerdo al algoritmo de aprendizaje. En este caso particular, sería la combinación lineal entre los pesos y la resultante de cada función radial para determinar la salida z .

Tal como hemos visto con anterioridad, sería,

$$z = \sum w_i G(\bullet), \quad (28)$$

donde $G(\bullet)$ es la salida de la capa oculta y se corresponde con la función radial aplicada a la distancia euclidiana en cada una de las unidades ocultas.

Del resultado de este tipo de entrenamiento podemos observar que:

1. Los valores de entrada se recomiendan que previamente sean de algún modo transformados a una escala.
2. En la capa oculta, en la medida que los valores de entrada se parezcan más a un centro su distancia tenderá a *cero* y de este modo la función gaussiana se dispararía a las vecindades de *uno*. Por otro lado, en la medida que los valores de entrada no se parezcan a su centro la distancia será mayor y la función radial parecería tender a *cero*. Este proceso es una clasificación no lineal de las entradas.
3. En la capa de salida del modelo RBF, los valores obtenidos en las salidas de la capa oculta serían transformados por la función lineal que permite aproximar los valores z a los valores deseados, mediante la combinación lineal que se sucede en esta capa entre sus pesos y el resultado de aplicar la función radial. Es decir,

$$z = \sum w_i G(\bullet), \quad (29)$$

4. El tiempo de entrenamiento es substancialmente inferior al requerido por otros algoritmos. Es una pasada hacia adelante en la mayoría de los casos. La diferencia la establece si se incorpora en la salida del modelo de entrenamiento, una supervisión a través del control del error que se produce entre los valores calculados y los observados, conduciendo a una retropropagación del error.
5. Alrededor del algoritmo clásico iniciado por Broomhead y Lowe, se han implementado algoritmos que contribuyan a la mejor selección de los centros y anchos de las funciones radiales.
6. Nuevos cambios incorporados a las funciones de activación de salida originan nuevos modelos de entrenamiento. Tal es el caso de las redes neuronales GRNN

(Generalized Regression Neural Network) y PNN (Probabilistic Neural Network), PCANN (Principal Component Analysis Neural Network).

3.5.1.3 Entrenamiento de una RBF

Diferente a las redes supervisadas vistas anteriormente, en este caso, suponiendo un hiperplano definiendo un espacio N -dimensional, lo que pretende una red RBF es ejecutar una correspondencia no lineal entre los patrones de entrenamiento que definen el espacio de entrada al espacio oculto definido por la capa oculta y una correspondencia lineal desde este espacio al espacio de salida. Es decir definir a la salida una superficie que describa las entradas.

En vista de que esta superficie es desconocida, se acude un proceso de entrenamiento usando ejemplos representativos tanto para la entrada como para la salida.

De acuerdo a ello, han surgido variantes como producto fundamentalmente de las siguientes desventajas:

- De no conocer los centros (a veces el ancho) para cada función radial.
- De situaciones de singularidad presentes en la implementación del algoritmo con problema de dimensionalidad.
- De un gran volumen de entradas haciendo inmanejable la aplicación del algoritmo. Se presentan problemas de regularización [43]. De acuerdo a Broomhead y Lowe el proceso de aprendizaje de la red RBF puede ser visto en dos fases:

Fase de entrenamiento: Constituye la optimización de un procedimiento de ajuste de una superficie que se define como producto de los ejemplos de entrada-salida presentados a la red.

Fase de generalización: Una interpolación entre los datos o interpolación realizada a lo largo de la superficie generada por un procedimiento de ajuste de la aproximación óptima de la superficie real.

De este modo en el sentido estricto de interpolación, podemos decir que existe una función que satisfaga la condición de interpolación $F(x_i) = d_i$, donde x_i son los puntos que describen la superficie de un espacio N dimensional y d_i representa su respuesta. Tal como lo describe esta función, la interpolación estricta se refiere a que la función está restringida a pasar por todos los puntos.

Es decir, es la aproximación óptima de la superficie real.

Clásico. Bajo esta premisa, tenemos que la función que puede describir dicha interpolación, de acuerdo a Powell es la siguiente.

$$F(x) = \sum_{i=1}^N w_i G(\|x - x_i\|), \quad (30)$$

donde la función $F(x)$ está involucrada con la función lineal $G(\bullet)$ y la combinación lineal con los pesos. En forma matricial, sería

$$Gw = z \quad (31)$$

Cada elemento:

$$g_{j,i} = g(|x_j - x_i|) \quad j, i = 1 \dots N \quad (31)$$

$$z = [z_1, z_2, z_3, \dots, z_N]^T \quad (32)$$

$$w = [w_1, w_2, w_3, \dots, w_N]^T \quad (33)$$

Provistos que todas las observaciones son distintas, entonces G se podría decir que es positiva definida y por lo tanto los pesos podrían ser calculados mediante la inversa de G . Es decir

$$w = G^{-1}z \quad (34)$$

Sin embargo se puede correr el riesgo de que la inversa de la matriz de interpolación G está próxima a ser singular. En este caso se procedería mediante la teoría de la regularización para perturbar la matriz mediante $G = G + \lambda I$. [30]

De esta manera sería un aprendizaje directo, provocando cambio a los pesos que están ubicados entre la capa oculta y la capa de salida.

3.6 La cámara CCD

La cámara CCD (*charge-coupled-device*) es un sensor de imagen que utiliza elementos semiconductores fotosensibles en forma de arreglos matriciales. Los receptores activos de este arreglo son distribuidos en píxeles individuales. En un sensor CCD se transforma la energía de la luz incidente en energía eléctrica. La carga eléctrica almacenada en la celda es posteriormente transportada utilizando un registro de desplazamiento (*shift register*) para conformar una señal de video. Cabe señalar que en las cámaras CCD se discretiza la imagen en píxeles, sin embargo el valor de la carga eléctrica almacenado en cada celda no se digitaliza en el arreglo CCD sino en una conversión posterior realizada por un conversor analógico/digital. Las cámaras CCD son consideradas como dispositivos que poseen una muy baja deformación geométrica de la imagen, una muy buena sensibilidad a la luz y una muy buena resolución llegando típicamente a los 400.000 píxeles. El tiempo de captura de la imagen está típicamente en el rango de 1/60s y 1/10000s.

La formación geométrica de la imagen se considera como una transformación de las coordenadas x, y , en un nuevo sistema de coordenadas u, v . Para esta transformación es necesario considerar los siguientes cuatro aspectos:

Cambio de escala: Las coordenadas de una imagen están comúnmente expresadas en diferentes unidades métricas a las empleadas para medir el espacio 3D. Por lo general la imagen se expresa en píxeles y el espacio 3D en milímetros. Por esta razón, en la transformación de coordenadas $(x, y) \rightarrow (u, v)$ necesario considerar un factor de escala.

Adicionalmente es necesario tomar en cuenta que debido a que los píxeles no son cuadrados, sino rectangulares, el factor de escala es distinto en cada eje de la imagen. Los factores de escala utilizados son: α_x y α_y , expresados en [píxel/mm].

Traslación del origen: Se introducen las variables (u_0, v_0) , para denotar el punto principal de la imagen en el nuevo sistema de coordenadas, es decir $u = u_0, v = v_0$, corresponden al punto $x = 0, y = 0$.

Rotación de los ejes: Los ejes x, y y los ejes u, v no tienen la misma orientación. En la modelación de la cámara existen dos métodos: uno que considera un ángulo θ de rotación, y otro que asume que este ángulo es cero y que el ajuste debe hacerse en los ejes X, Y, Z del sistema de coordenadas del espacio 3D. En este segundo método el eje Z sigue siendo el eje óptico de la proyección y los ejes x, y siguen siendo paralelos a los ejes X, Y , sin embargo se hace coincidir, mediante una rotación del eje Z , la orientación de los ejes x, y con los ejes u, v .

Factor de torcimiento: Muchas veces los ejes u, v no son ortogonales debido a que los píxeles en los arreglos CCD no son rectangulares. En este caso es necesario introducirle al modelo de la cámara un factor de torcimiento (*skew factor*) s . En la mayoría de cámaras s es cero.

3.7 Módulo NI Vision Development

El Módulo Vision Development de National Instruments es una colección de funciones para procesamiento digital de imágenes y visión artificial, con las que se puede: leer códigos de barras, clasificar objetos, analizar partículas, ubicar características, filtrar imágenes, medir componentes, etc., además incluye los paquetes: NI Vision Assistant y NI Vision Acquisition y miles de drivers para distintas cámaras USB, IEEE 1394 y GigE Vision.

Capítulo 4

En el presente capítulo se propone la metodología con la que se pretende dar solución al problema planteado en el capítulo 1, y se describen los experimentos realizados en cada una de las etapas que conforman dicha metodología.

Capítulo 4. Metodología propuesta y experimentación

4.1 Metodología propuesta

Las etapas de la metodología que se propone para resolver el problema planteado en el capítulo 1, se muestran en el diagrama a bloques de la Fig. 13.

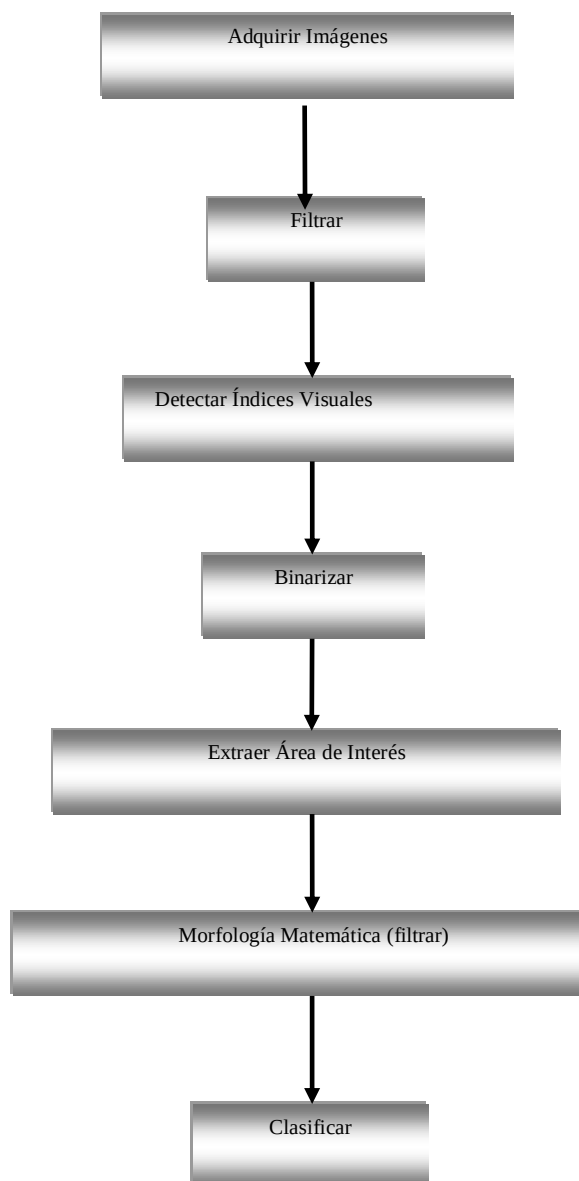


Figura 1. Diagrama a bloques de la metodología de solución propuesta.

4.2 Descripción general del las etapas de la metodología de solución propuesta

Adquirir imágenes

Se capturarán, transmitirán y almacenarán imágenes fotográficas en escala de grises de botellas de vidrio con y sin defectos, empleado una cámara CCD de la marca Basler modelo Scout sca640-70fm.

Filtrar

Se filtrarán las imágenes adquiridas, para eliminar algunas impurezas que puedan causar problemas en las etapas posteriores y para ello se utilizaran filtros lineales y no lineales.

Detectar índices visuales

Se emplearán los operadores de Roberts, Prewitt y Sobel para obtener el gradiente de la imagen y este pueda utilizarse en conjunto con el alguno de los detectores de Harris y SUSAN, o con la transformada de Hough, para encontrar índices visuales como: bordes, esquinas, vértices o líneas, y con ellos pueda ubicarse el objeto de interés dentro de la imagen y además para resaltar los bordes que representan el defecto físico de la botella ya que este produce cambios de intensidad en los niveles de gris de los píxeles que lo representan dentro de la imagen.

Extraer zona de interés

El tamaño y posición del objeto de interés (cuello de la botella de vidrio), puede variar dentro de la imagen, es decir este puede estar rotado y/o desplazado horizontalmente, por lo que se utilizará como referencia alguno de los índices visuales encontrados en la etapa anterior para ubicar el objeto de interés dentro de la imagen, y extraer de forma automática el área en donde es posible encontrar los bordes que representan el defecto físico de la botella (si este existe), y desechar el resto ya que la información contenido fuera de esa área no será utilizada en la etapa siguiente.

Binarizar

La posición de ésta etapa dentro la metodología que se está proponiendo, puede variar ya que ésta depende de los resultados obtenidos al emplear los detectores de esquinas ya que si con éstos se encuentran puntos en común dentro de las todas imágenes disponibles, ésta etapa se realizará después de extraer el área de interés, pero si los resultados no son satisfactorios, entonces ésta etapa se coloca antes de detectar los índices visuales, debido a que la transformada de Hough requiere de una imagen binaria para que pueda desarrollarse.

Para binarizar la imagen se establecerá una condición y un valor de umbral con el cual serán comparados los píxeles de la imagen obtenida en la etapa anterior, y se asignarán valores binarios “1” y “0”, a los píxeles que cumplan o no con la condición establecida. Esto con el fin de mantener sólo la información útil, en la siguiente etapa.

Morfología matemática (filtrar)

Se emplearán algunas operaciones morfológicas: como erosión, apertura, cerradura, y dilatación, para: 1) eliminar algunos bordes no deseados que hayan permanecido después de binarizar la imagen, 2) unir bordes separados y 3) aumentar el tamaño de los bordes que representan el defecto de la botella.

Clasificar

Para dar un resultado final del análisis de la imagen de la botella, se empleará una red neuronal de base radial para clasificar dicha imagen en alguna de las siguientes clases: “botella defectuosa” o “botella sin defecto”, de acuerdo a los bordes (defectos) detectados o no, en el área de interés.

4.3 Desarrollo de las etapas de la metodología de solución propuesta

Matlab® 2008a y LabVIEW® 2009, es el software que se utilizará para desarrollar los experimentos en cada una de las etapas de la metodología propuesta para resolver el problema planteado en el capítulo 1.

A continuación se describen los experimentos realizados, en cada una de las etapas de la metodología propuesta, para clasificar las botellas de vidrio.

4.3.1 Adquirir imágenes

Para capturar y obtener imágenes de las botellas de vidrio, se acondicionó el lugar en donde se obtuvieron tales imágenes y para ello se consideró: el tipo de cámara fotográfica, la iluminación, el tamaño de la botella de vidrio y la distancia de ubicación entre cada uno de esos dispositivos. En la Fig. 14, se muestra el esquema general de la ubicación de cada uno de los dispositivos utilizados.

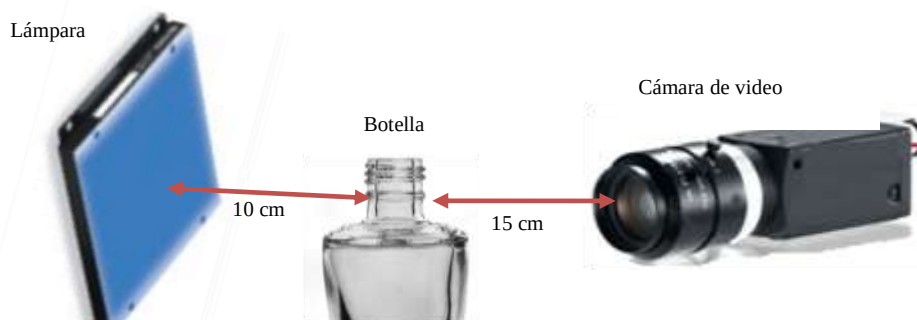


Figura 2. Ubicación espacial de los dispositivos utilizados para adquirir imágenes.

La cámara digital fotográfica que se utilizó es monocromática de tipo CCD (siglas en inglés de *Charge-Coupled Device*: ‘dispositivo de carga acoplada’), de la marca Basler modelo Scout sca640-70fm.

El tipo de iluminación que se seleccionó es posterior difuso, debido a que con éste se obtiene mayor contraste en el contorno y bordes de materiales transparentes. Por tal motivo se instaló una lámpara marca KEYENCE modelo CA-DB de color blanco.

La distancia de ubicación entre la lámpara y la botella de vidrio es de 7 cm, y de la botella a la cámara fotográfica es de 15cm, el ángulo de la cámara con respecto a la corona de la botella es de aproximadamente 10 grados.

Después de acondicionar el lugar e instalar el equipo mencionado anteriormente, se desarrolló un software sobre la plataforma LabVIEW® 2009, para activar la cámara fotográfica e iniciar la captura y almacenamiento de imágenes en la computadora. Las características generales de las imágenes adquiridas son las siguientes:

El tamaño de cada imagen es de 480 píxeles de alto por 640 píxeles de ancho, con una resolución de niveles de gris 8 bits con un formato de almacenamiento PNG, ocupando aproximadamente 150 KBytes de espacio en la memoria física de la computadora.

4.3.2 Filtrar

Debido a que en la etapa posterior se emplearán algunos operadores para obtener el gradiente y realzar los bordes de las imágenes, se emplearán filtros mediana y media aritmética para tratar de eliminar o disminuir el ruido presente en las imágenes y evitar que este provoque bordes falsos dentro de las imágenes.

El tamaño de la máscara del filtro se determinará experimentalmente, ya que además de disminuir o eliminar el ruido presente en la imagen, se debe tener cuidado en no eliminar los píxeles que representan el defecto físico de la botella.

Después de observar cada una de las imágenes disponibles, se notó que éstas contenían puntos negros en algunas partes del cuello, y aunque al principio se supuso que esto se

debía al polvo adherido a las botellas de vidrio durante la manipulación humana, se limpiaron y posteriormente se capturaron otras fotografías, pero se observó que esos puntos seguían presentes en las imágenes, entonces se examinó de manera más detallada cada una de las botellas y se notó que éstas, no son completamente lisas, es decir, tienen textura en algunas partes y los puntos que se observan en algunas zonas del cuello de la botella, representan dicha textura y estos dependen del ángulo de vista de la cámara fotográfica.

En la Fig. 15(a) y Fig.15 (b), se pueden observar los puntos y/o pequeñas manchas visibles en las imágenes, provocadas por la textura del vidrio.

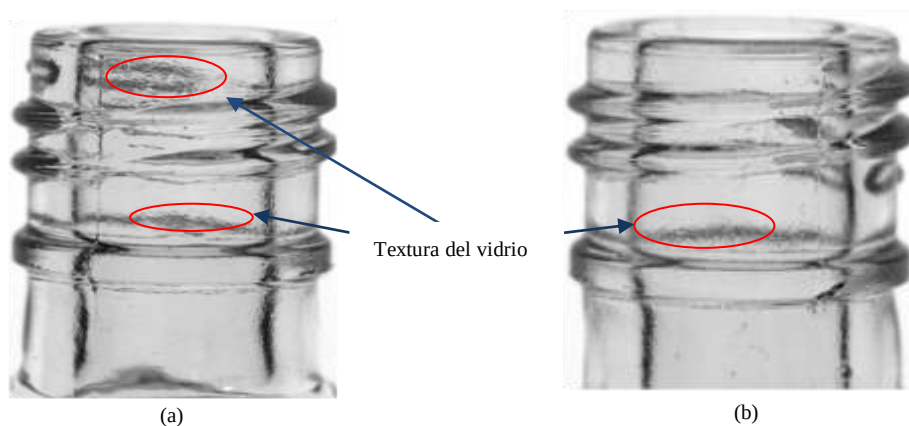


Figura 3. Textura y ruido presente en las imágenes adquiridas.

Después de emplear filtros de mediana y media aritmética de tamaño 3x3, 3x5, 3x7, 5x5, 5x7, 5x9, 7x7, 7x9, 9x9, 9x11, 11x11 y 15x15, el que entrego resultados más cercanos a lo deseado es el filtro mediana con un tamaño de 5x5 píxeles, debido a que con este se logra disminuir y/o eliminar el ruido de las imágenes sin afectar demasiado los píxeles que representan el defecto de la botella.

En la Fig. 16: (a) y (b), se observa el resultado obtenido al aplicar un filtro mediana de tamaño 5x5, a las imágenes mostradas en la Fig. 15.

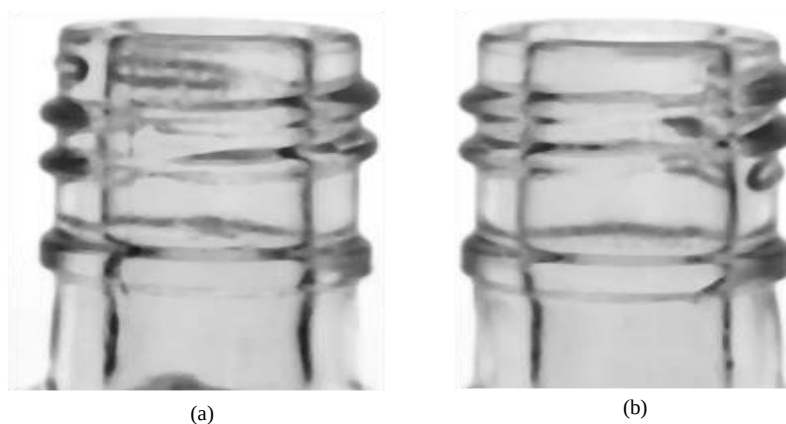


Figura 4. Imágenes obtenidas después de emplear el filtro mediana de 5x5 píxeles

4.3.3 Detectar índices visuales

Se empleará morfología matemática y los operadores de Roberts, Prewitt y Sobel para obtener el gradiente de la imagen y este pueda utilizarse en conjunto con el alguno de los detectores de Harris, SUSAN, o con la transformada de Hough, para encontrar índices visuales como: bordes, esquinas, vértices o líneas, y con ellos pueda ubicarse el objeto de interés dentro de la imagen, y además para resaltar los bordes de las imágenes que representan el defecto físico de la botella.

Después de emplear cada uno de los operadores mencionados en el párrafo anterior se observarán las imágenes resultantes y al final se seleccionará el operador que mejor resalte los píxeles que representan el defecto físico de la botella y además permita encontrar por lo menos algún índice visual en común dentro de la todas las imágenes disponibles.

Para realizar una comparación entre los distintos operadores que se emplearán, se utilizarán las mismas imágenes para convolucionar las distintas mascararas de cada operador.

El primer operador que se empleará es el de Roberts, con las máscaras que se muestra en las Tablas 15y 16.

Tabla 1. Máscara 2 (Gy).	
0	1
-1	0

Tabla 2. Máscara 1 (Gx).	
1	0
0	0

Después de convolucionar las máscaras anteriores con tres imágenes distintas, se eligieron los valores de umbral: $u = 35$ y $u = 180$ para que las imágenes resultantes fueran binarizadas, ya que no se apreciaban muy bien los bordes de la imagen en niveles de gris.

En las Figs. 17 (a), 17(b) y 17(c) se muestran los resultados obtenidos al realizar lo descrito anteriormente.

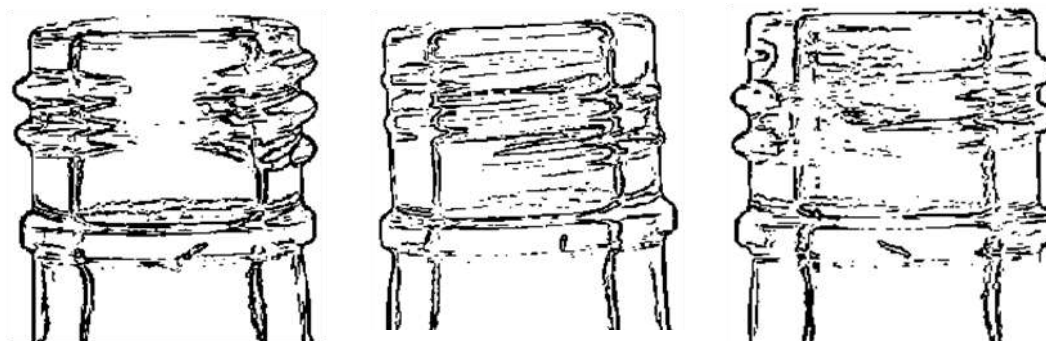


Figura 5. Imágenes binarias obtenidas después de detectar bordes con el operador de Roberts.

Como se puede observar en la Fig. 17, los bordes que representan los defectos físicos de las botellas se aprecian muy bien pero también hay algunos otros que no deberían estar y aunque éstos pueden eliminarse variando los valores de umbral como se muestra en las imágenes de la Fig. 18, se continuarán probando los demás operadores.

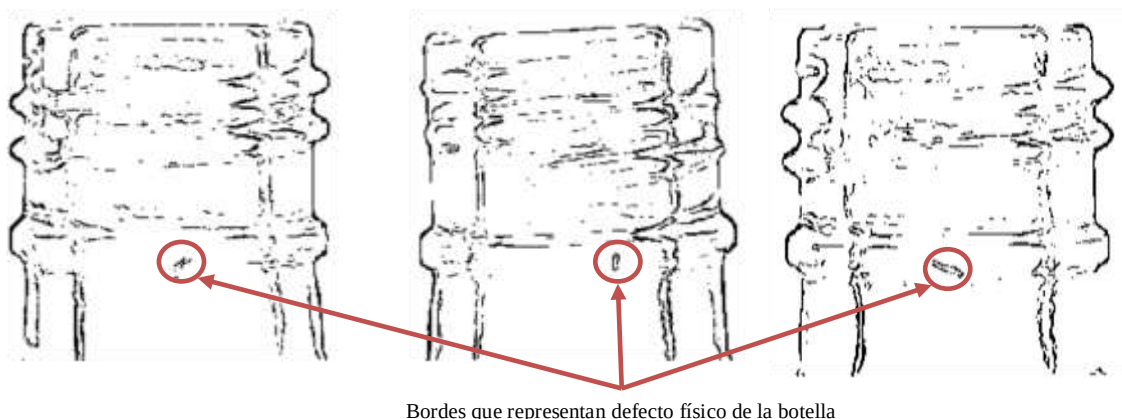


Figura 6. Imágenes binarias obtenidas después de realzar bordes con operador de Roberts con valores de umbral $u = 20$ y $u = 180$.

El siguiente operador que se empleará es el de Sobel, con las máscaras que se muestran en las Tablas 17 y 18.

Tabla 3. Máscara 2 (Gy).		
-2	-1	0
-1	0	1
0	1	2

Tabla 4. Máscara 1 (Gx).		
0	1	2
-1	0	1
-2	-1	0

En las Figs. 19 y 20, se muestran los resultados obtenidos al convolucionar las máscaras de Sobel de las Tablas 17 y 18, con tres imágenes distintas.

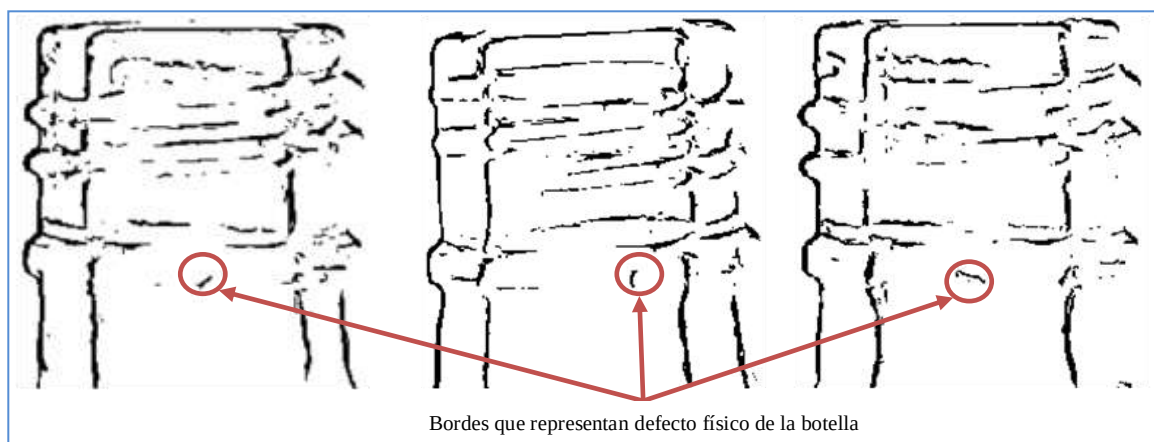


Figura 7. Imágenes binarias obtenidas después de detectar bordes con la máscara 1 de Sobel.

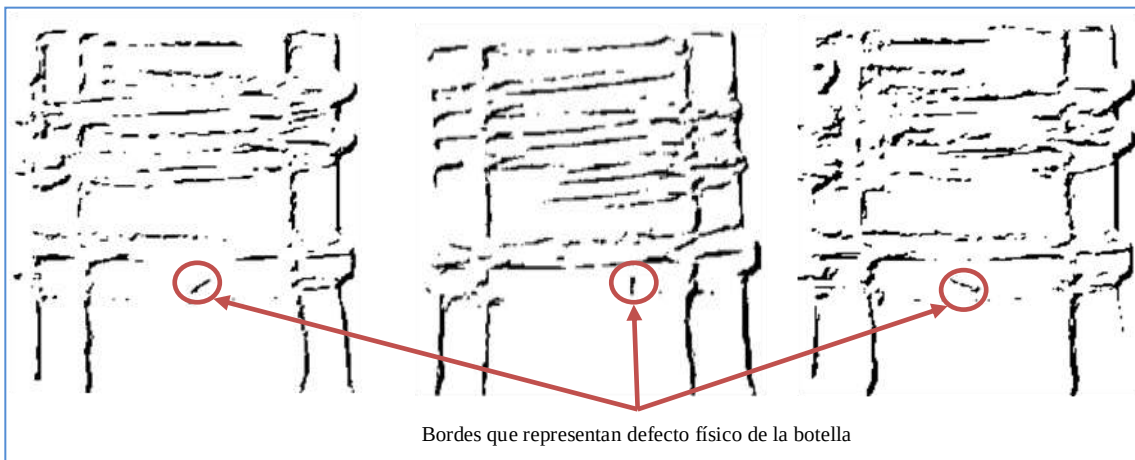


Figura 8. Imágenes binarias obtenidas después de detectar bordes con la máscara 2 de Sobel.

Como se puede observar en las tres imágenes de la Fig. 19 y Fig. 20, los bordes que representan el defecto físico de la botella se pueden apreciar muy bien en todas ellas, por lo que cualquiera de las dos máscaras empleadas para realzar los bordes, puede ser utilizada más adelante, aunque tal vez la que resulte más útil es la máscara 1, ya que con ésta se obtiene un número menor de bordes falsos provocados por la sombra de la botella.

El siguiente operador que se empleará es el gradiente morfológico con elementos de estructura de tipo: columna de dimensiones 1x3, 1x5, 1x7 y 1x9 píxeles, fila de 3x1, 5x1, 7x1, 9x1, cuadrados de dimensiones 3x3, 5x5, 7x7, 9x9 y 11x11, rectangulares de 3x5, 3x7, 5x7, 5x9, 5x11, 7x9, 7x11 y de tipo disco de diámetro de 3,5,7,9 píxeles.

En las Figs. 21, 22 y 23, se muestran las imágenes obtenidas con el gradiente morfológico utilizando algunos de los elementos de estructura descritos anteriormente.



Figura 9. Realce de bordes con elemento de estructura Fila de 1x3 píxeles.



Figura 10. Realce de bordes con elemento de estructura cuadrado de 5x5 píxeles.



Figura 11. Realce de bordes con elemento de estructura circular de 7x7 píxeles.

De acuerdo a los resultados que se obtuvieron con el gradiente morfológico y con los operadores para realzar bordes, se puede decir que cualquiera de ellos es útil, para emplearlo en éste trabajo, debido a que con todos ellos es posible visualizar los bordes que representan el defecto de la botella, por lo que en este momento no es posible tomar la decisión de cuál de ellos emplear, sino hasta después de buscar y encontrar (si es que existe) por lo menos un índice visual en común, dentro de todas las imágenes disponibles para esta investigación.

4.3.4 Binarizar

Ésta etapa se realizó junto con la anterior ya que para poder observar los resultados del gradiente morfológico y de los operadores utilizados para resaltar los bordes de las imágenes, fue necesario definir un valor de umbral y una regla para binarizar la imagen.

Para ello se considero, que debido a que las condiciones de iluminación bajo las cuales se adquieren las imágenes se tienen de cierto modo controladas, es posible seleccionar un valor de umbral de forma manual, ya que los niveles de gris de los píxeles, no cambian drásticamente entre una y otra imagen [7].

De acuerdo a lo anterior el valor del umbral se estableció experimentalmente, con la siguiente regla:

$$g(x,y) = \left\{ \begin{array}{ll} 1 & \text{si } u_1 \leq f(x,y) \\ 0 & \text{en caso contrario} \end{array} \right\} \quad (1)$$

Donde:

$f(x,y)$ = Valor del píxel de la imagen en niveles de gris, en la posición (x, y).

$g(x,y)$ = Valor binario asignado al píxel de la imagen, en la posición (x, y).

u_1 = Valor de umbral 1.

Los valores de umbral que se utilizaron para binarizar las imágenes son: 50, 60, 70, 80, 90, 100, 110, 120,..., 210 y el que mejores resultados mostro es el de 100, ya que los píxeles de menor intensidad que representan el defecto de la botella están por debajo de este valor.

En las Figs 24: (a), (b) y (c), se muestran las imágenes obtenidas después de ser binarizadas con valores de umbral de 70, 100 y 130 respectivamente.

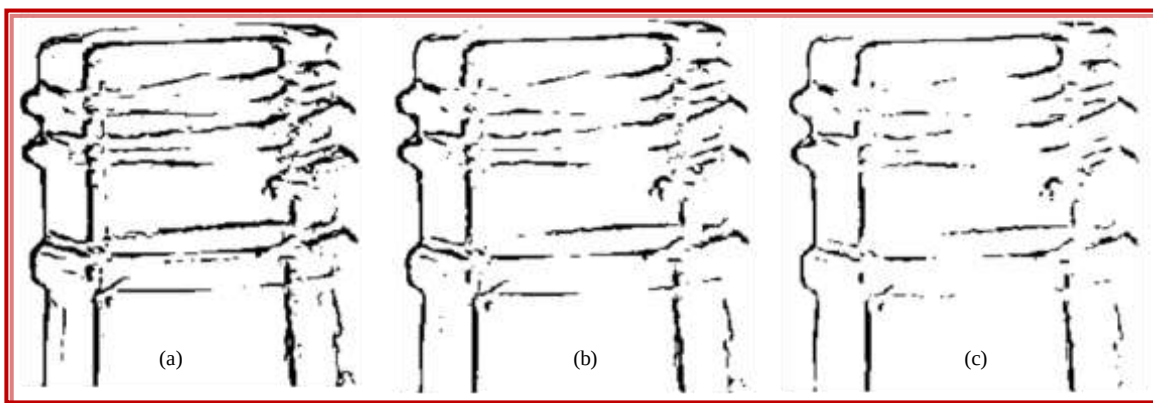


Figura 12. (a), (b) y (c) Imágenes binarias obtenidas con valores de umbral de 70,100 y 130.

4.3.5 Extraer zona de interés

Debido a que las imágenes adquiridas muestran el cuello completo de la botella de vidrio y a que el defecto que se pretende detectar en esta investigación, sólo se encuentra dentro de una zona específica, es posible y necesario extraer de forma automática la zona de la imagen en donde es posible encontrar (en caso de existir) el defecto, y desechar el resto ya que la información contenida fuera de esta zona, no será utilizada en la etapa de clasificación.

Para extraer de forma automática el área de interés de la imagen al principio se propuso realizar algo similar al trabajo [2], es decir, realizar un barrido horizontal y vertical de los píxeles de la imagen, para detectar el primer borde vertical y horizontal del cuello de la botella, obtener su posición y tomarla como punto de referencia para definir y extraer el

área de la imagen en donde es posible encontrar el borde que representa el defecto de la botella.

Al realizar dicho experimento con distintas imágenes y comparar los resultados se observaron variaciones de más de 10 píxeles en la posición del primer borde, ya que en ocasiones este se encontraba desplazado algunos píxeles a la derecha o izquierda, arriba o abajo, debido a dos razones: 1) no se había considerado que el cuello de la botella puede tener cierta inclinación, provocando que en la imagen se pueda observar la parte trasera de la corona superior de la botella y por lo tanto los primeros píxeles detectados correspondan a esa parte de la botella, 2) el detector de bordes de Sobel se basa en la primera derivada por lo que es posible obtener bordes discontinuos, lo cual implica que en ocasiones algunas partes del borde superior sean resaltadas y otras no aparezcan.

En la Fig. 25(a) se muestra una de las imágenes en las que se observo inclinación en el cuello de la botella y en la Fig. 25(b), se observa una imagen de una botella sin inclinación en el cuello.



Figura 13. (a) Botella con inclinación en el cuello y (b) Botella sin inclinación en el cuello.

En la Fig. 26, se muestran tres imágenes donde es posible ver la variación de la posición del primer borde encontrado en cada una de las imágenes, empleando la técnica descrita anteriormente.

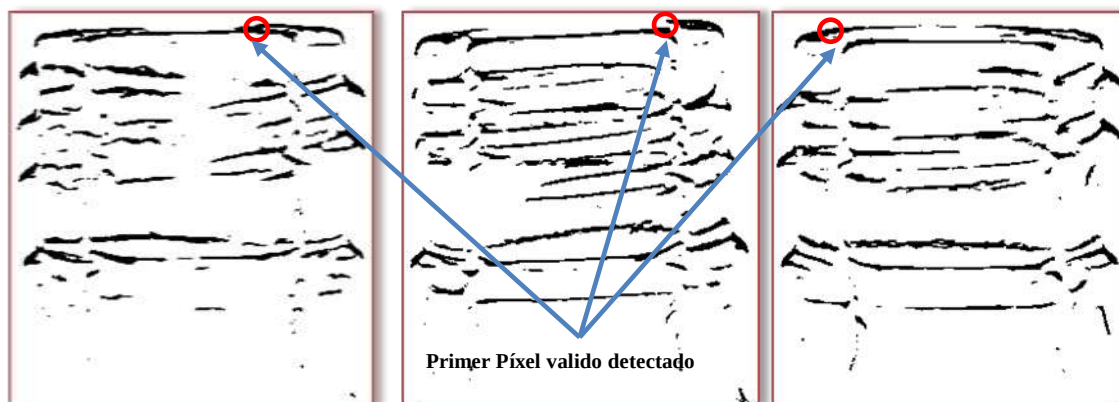


Figura 14. Ubicación del primer píxel valido detectado.

Después de observar, comparar y analizar los resultados obtenidos al emplear la técnica descrita anteriormente, no se descartó al 100% utilizarla más adelante, pero se decidió buscar otro método y/o técnica con la que se pueda mejorar o encontrar algún otro punto de referencia, debido a que con el actual, no es posible obtener un punto de referencia constante o con poca variación en su posición, lo cual provoca que al extraer el área de interés de la imagen, esta contenga en algunas ocasiones píxeles que están a la izquierda o derecha, arriba o abajo de la región deseada, y dependiendo de las dimensiones de la zona de interés, ésta podría contener o no el borde que representa el defecto (si este existe).

Entre los otros métodos y técnicas que se analizaron para tratar de mejorar el resultado obtenido anteriormente, está el detector de Harris [21], ya que este permite detectar esquinas en una imagen, además es invariante ante la translación y rotación de los objetos dentro de la imagen. Pero no fue posible emplearlo en esta investigación, ya que no es posible encontrar algún punto en común dentro de todas las imágenes en niveles de gris de las botellas, debido a que estas no presentan la misma información, porque al captar las imágenes, la botella se rotó angularmente por lo que en ocasiones es posible observar la parte frontal de la botella y en otras la trasera.

En la Fig. 27, se presentan algunas imágenes donde se observa una botella con diferente ángulo de rotación.

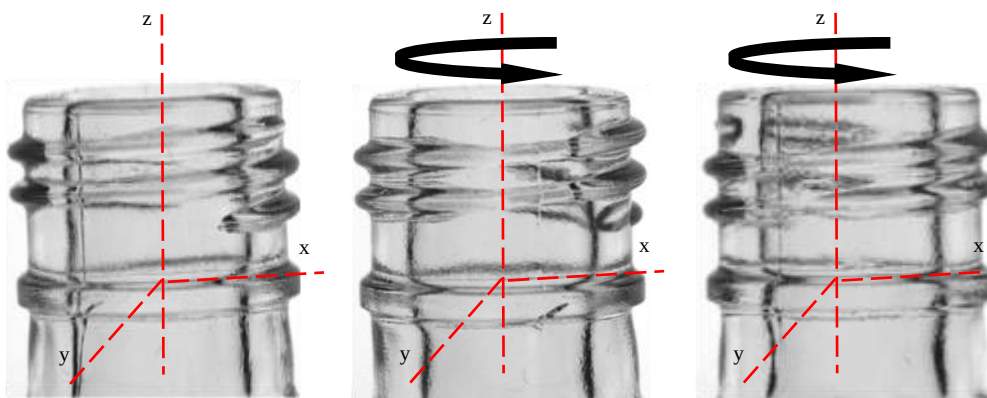


Figura 15. Imágenes de la botella vista desde diferentes ángulos.

Otro método analizado fue el detector de esquinas SUSAN [44], pero se tiene el mismo problema, que con el de Harris. Finalmente, se analizó la posibilidad emplear la transformada de Hough [22] para detectar líneas, ya que al observar algunas imágenes obtenidas después de ser binarizadas, se notó una aparente línea recta en la parte superior de la imagen, la cual representa el borde superior de la corona de la botella y que puede servir como el punto de referencia que se está buscando para extraer de forma automática la zona de interés de la imagen. En la Fig. 28 se muestran dos imágenes binarias de botellas, en donde se puede observar una aparente línea recta en la parte superior de la imagen.

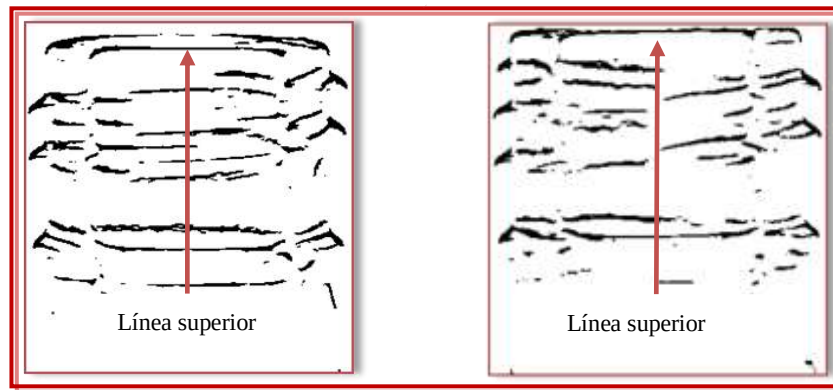


Figura 16. Imágenes binarias de botellas de vidrio con línea superior.

Después de revisar visualmente y encontrar una aparente línea recta en la parte superior de todas las imágenes disponibles para esta investigación, se utilizó la transformada de Hough para detectar dicha línea y algunos de los resultados obtenidos se muestran en la Fig. 29, en donde se dibujan las primeras cinco líneas de mayor tamaño detectadas, en seis diferentes imágenes.

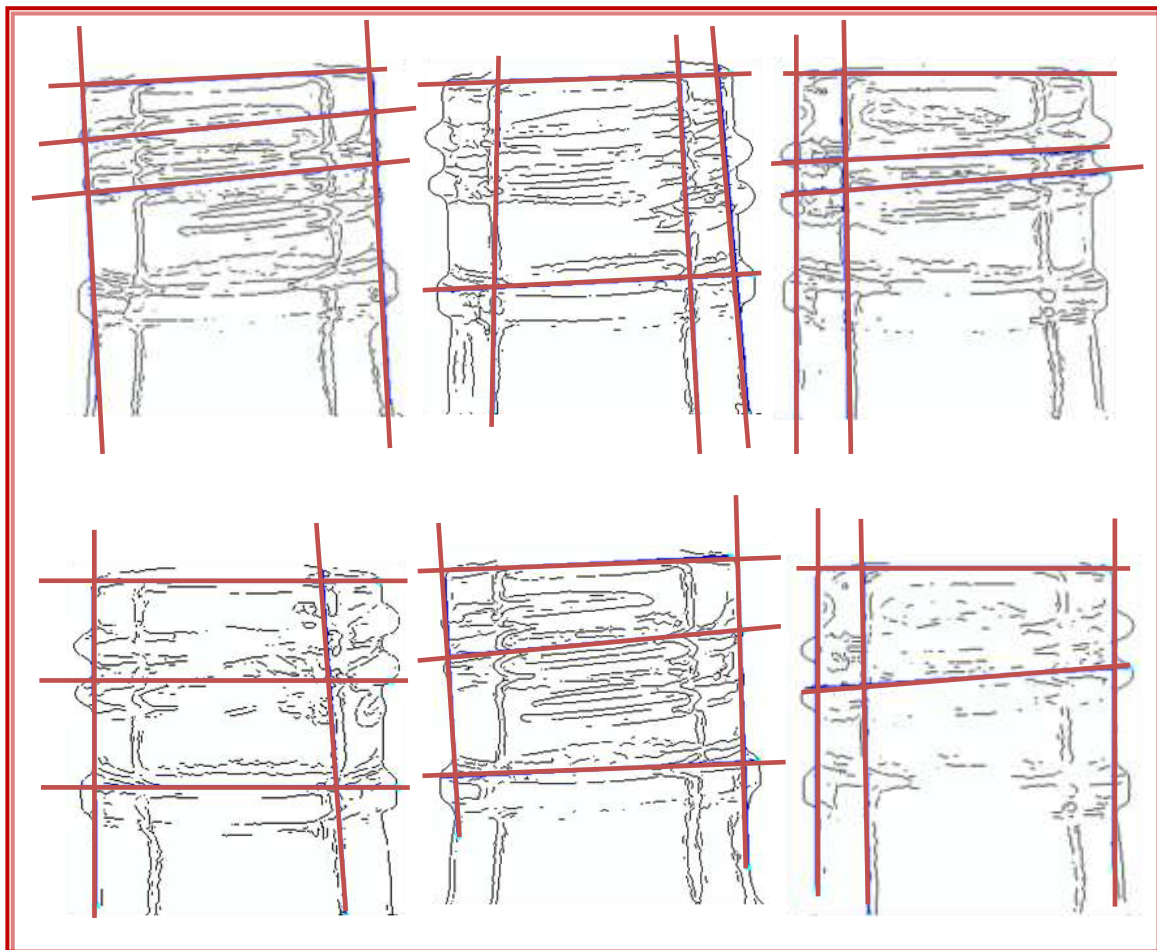


Figura 17. Imágenes binarias de botellas de vidrio, con las cinco líneas de mayor tamaño detectadas con la transformada de Hough.

Como se puede observar en las imágenes de la Fig. 29, la única línea que está presente en todas ellas, es la línea horizontal de la parte superior de cada imagen, ya que los niveles de gris en ésta parte de la imagen están por encima del valor del umbral seleccionado y por lo tanto estos se preservan, y aunque en algunas imágenes hay algunas discontinuidades en ese borde aun así sigue siendo uno de los de mayor tamaño.

Después de observar las cinco líneas detectadas y comprobar que la línea que se quiere utilizar como referencia, es una de las de mayor tamaño en todas las imágenes disponibles, se decidió programar y acotar la transformada de Hough, para que solo detecte líneas rectas con ángulos de inclinación dentro del intervalo de $[-3\ 3]$ grados, debido a que la línea que se quiere detectar solo está dentro de ese intervalo, ya que el cuello de la botella puede o no, tener inclinación como se mostró en las imágenes de la Fig. 25.

Al finalizar el programa se realizaron pruebas con todas las imágenes disponibles, para obtener la línea de mayor tamaño y solo en doce del total de las imágenes la línea ubicada en la parte superior de la imagen no fue la de mayor tamaño, debido que en esas imágenes la línea presentaba discontinuidades y además había otras más grandes provocadas por los bordes que representan la rosca presente en el cuello de la botella.

En la Fig. 30 se presentan dos de las doce imágenes en las que la línea horizontal de mayor tamaño detectada, no fue la ubicada en la parte superior de la imagen y en la Fig. 31, se presentan dos imágenes en las que dicha línea si lo fue.

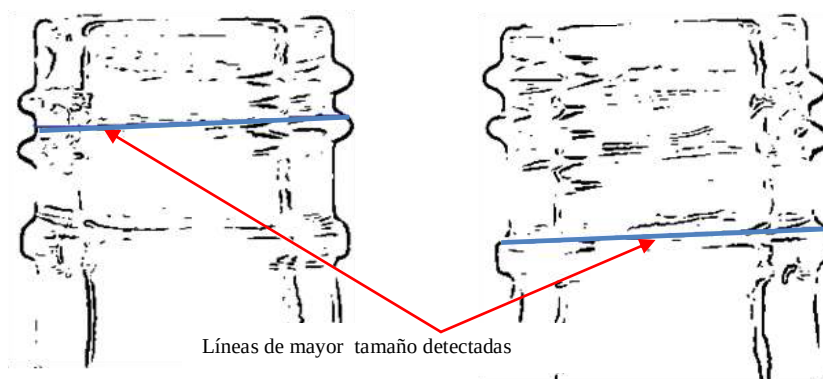


Figura 18. Líneas de mayor tamaño detectadas con la transformada de Hough.

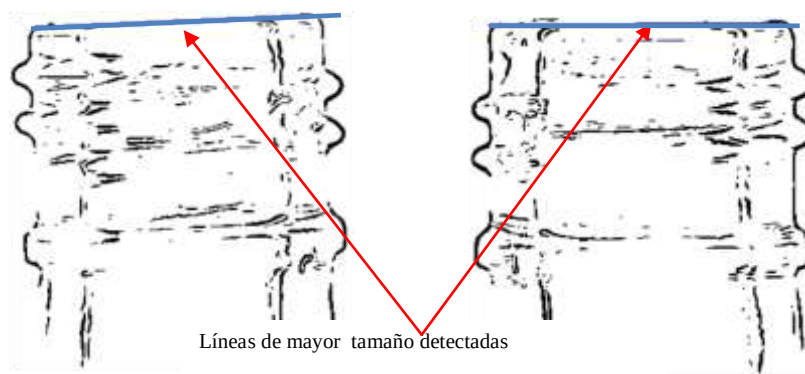


Figura 19. Líneas de mayor tamaño detectadas con la transformada de Hough.

Como se puede observar en las imágenes de las Figs. 30 y 31, la línea horizontal de mayor tamaño detectada con la transformada de Hough [22], no es la misma en todas ellas y por lo tanto es necesario buscar alguna otra característica con la que se pueda discriminar de manera correcta a las otras líneas detectadas.

Para solucionar el problema anterior y conocer la posición del punto inicial de la línea horizontal superior, se obtuvieron los parámetros de las tres líneas horizontales de mayor tamaño y con ellos se calculó la posición dentro del eje “Y” de cada una de ellas y se compararon para determinar cuál de ellas es la más cercana al origen dicho eje, y finalmente se almacenaron los parámetros de dicha línea, ya que estos pertenecen a la línea deseada.

Después de obtener los parámetros de la línea horizontal más cercana al origen del eje “Y”, solo hacía falta encontrar la posición de inicio dentro del eje “X” de dicha línea, ya que la transformada de Hough [22], no almacena las coordenadas de los puntos que pertenecen a cada una de las líneas, por lo que se realizó un barrido horizontal de los píxeles sobre este eje, tomando como punto inicial en el eje “Y” el punto calculado anteriormente y en el eje “X” el origen. Al encontrar el primer píxel con valor de “1” se abrió una ventana de 3x3 píxeles y se verificó que el número total de píxeles con valor de “1” existentes dentro de esta fuera mayor que dos, para comprobar que no se trataba de un punto aislado.

Para extraer de forma automática la zona de la imagen en donde es posible encontrar el defecto de la botella, se utilizó como referencia la posición del borde superior de la botella detectado anteriormente, para realizar un desplazamiento vertical hacia la parte inferior de la imagen y a partir de ese punto extraer todos los píxeles dentro de un área de 180 píxeles de largo por 50 píxeles ancho.

En la Fig. 32 se muestra de forma gráfica lo descrito anteriormente.

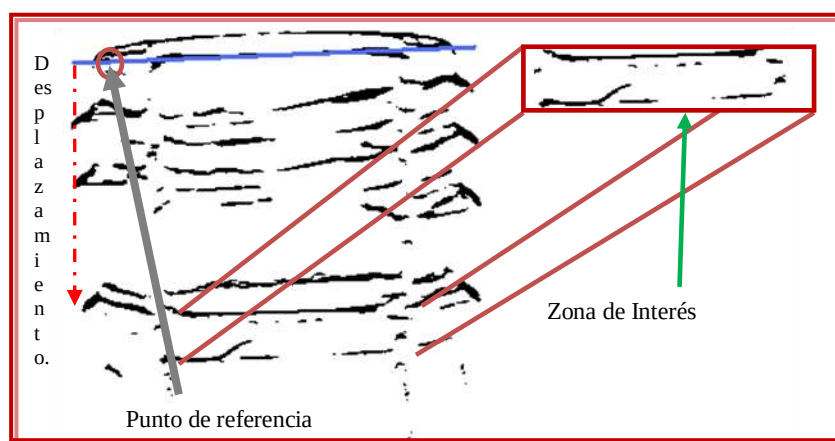


Figura 20. Extracción de la zona de interés.

4.3.6 Morfología matemática

Al observar las imágenes obtenidas en la etapa anterior, se notó que los bordes que representan el defecto de la botella, se habían reducido en tamaño, adelgazado y/o separado (observar Figs. 33 y 35), y además había algunos otros de menor tamaño (observar Fig. 34) que no representan ningún defecto y que podrían causar problemas en la etapa de clasificación, entonces se tomó la decisión de que emplear algunas operaciones morfológicas como: dilatación, erosión, cerradura y apertura, para tratar de unir y/o aumentar el tamaño de los bordes que representan el defecto de la botella y para eliminar algunos otros (no deseados).

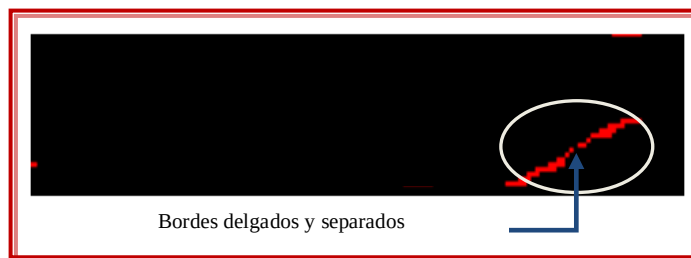


Figura 21. Adelgazamiento de bordes que representan el defecto de la botella de vidrio.

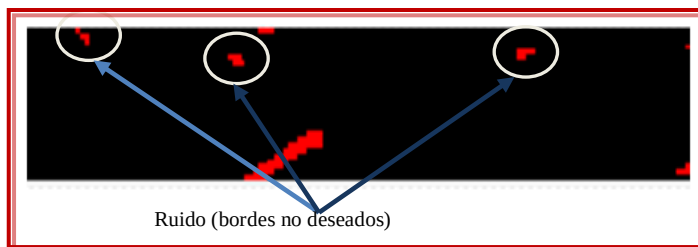


Figura 22. Bordes no deseados presentes en el área de interés.



Figura 23. Separación de bordes que representan el defecto de la botella de vidrio.

Para eliminar bordes no deseados como los que se muestran en la Fig. 33, se utilizó la operación de erosión con un elemento de estructura cuadrado como el que se muestra en la Tabla 19.

Tabla 5. Elemento de estructura para eliminar bordes no deseados	
1	1
1	1

Además de los pequeños bordes no deseados presentes en la zona de interés, se observó que también puede aparecer un borde que es provocado por el molde con el que se fabrican las botellas y cruza verticalmente la zona de interés, y que si no es procesado puede ser considerado por el clasificador como un defecto de la botella.

En la Figs. 36(a) y 36(b), se muestran dos imágenes en niveles de gris, en donde se presenta el borde creado por el molde, dentro y fuera de la zona de interés, y en la Fig. 37 se muestra la zona de interés extraída de la imagen 36(a), en donde también se observa el borde creado por dicho molde.

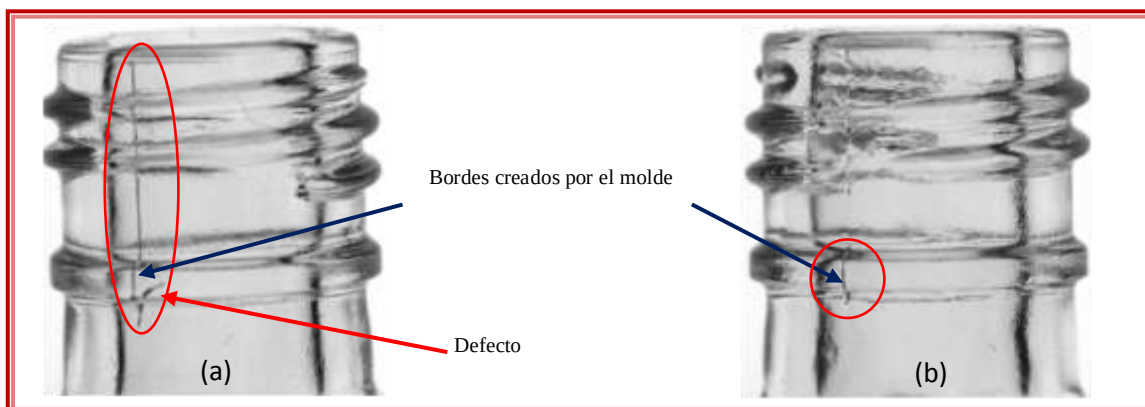


Figura 24. Bordes creados por molde de fabricación de las botellas.

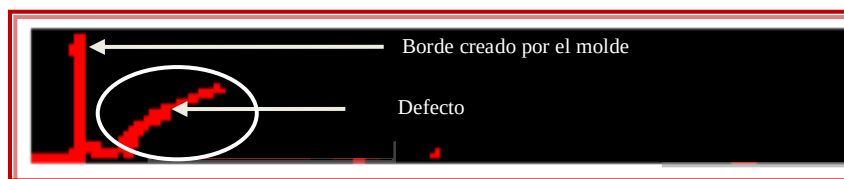


Figura 25. Imagen binaria con borde creado por el molde.

Antes de aplicar alguna operación morfológica para tratar de eliminar el borde creado por el molde, se intentó hacerlo, cambiando el valor del umbral para binarizar la imagen, pero no se obtuvieron resultados favorables ya que los niveles de gris de los pixeles que conforman dicho borde son muy similares a los que tienen los pixeles que representan el defecto de la botella, y por lo tanto, al variar el valor de umbral para binarizar la imagen y eliminar dicho borde, también se elimina parte del defecto.

Para solucionar el problema anterior se empleó la operación morfológica: apertura con el elemento de estructura mostrado en la Tabla 20.

Tabla 6. Elemento de estructura para eliminar línea vertical		
0	0	0
1	1	1
0	0	0

En la Fig. 38, se muestra la imagen binaria obtenida al emplear la operación morfológica apertura, para eliminar el borde credo por el molde con el que se fabrican las botellas y está presente en la figura 37.

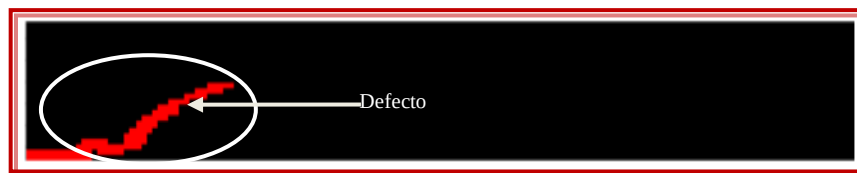


Figura 26. Imagen binaria después de emplear operación de apertura morfológica.

4.3.7 Clasificar

Finalmente, para clasificar las imágenes obtenidas de la etapa anterior, en alguna de las clases: “botella defectuosa” o “botella sin defecto”, se empleará una red neuronal de función de base radial con la siguiente estructura:

- Nueve mil neuronas en la capa de entrada, debido a que ese es el número total de píxeles dentro del área de interés de la imagen.
- Dos neuronas en la capa de salida. Una para clasificar las imágenes de las botellas con defecto y otra para clasificar las botellas sin defecto.
- Treinta y cinco neuronas en la capa oculta.

El número de neuronas en la capa oculta se determinó experimentalmente y para ello se entrenó y probó la RBF con 10, 20, 25, 30, 35 y 40 neuronas, durante 100, 200, 500 y 1000 iteraciones.

La cantidad de imágenes utilizadas en la fase de entrenamiento de la RBF fue de quince de las cuales, cinco imágenes eran de botellas con defecto y diez de botellas sin defecto.

Tras haber entrenado y probado la RBF se obtuvieron los resultados deseados, ya que el porcentaje de efectividad de la metodología al clasificar treinta y seis imágenes, fue superior al noventa por ciento. Pero aun con esos buenos resultados, se decidió hacer algunas modificaciones, ya que la metodología desarrollada hasta este punto no es invariante ante el posible cambio de escala de las imágenes.

Para eliminar la limitación descrita en el párrafo anterior, se modificará la etapa de extracción de la zona de interés, y se realizarán algunas mediciones en la estructura de las botellas de vidrio y se obtendrán los siguientes datos:

1. El diámetro de la corona superior.
2. La distancia entre la corona superior y la inferior.
3. La distancia entre las paredes internas del cuello de la botella.
4. El espesor del vidrio en el cuello de la botella.

Después de obtener los cuatro datos anteriores se programó una función, para: 1) normalizar el tamaño de la botella dentro la imagen, 2) calcular el número de píxeles que se deben recorrer horizontal y verticalmente a partir del punto de referencia encontrado anteriormente con la transformada de Hough, para llegar al inicio de la zona de interés de la imagen y 3) el número de píxeles a lo largo y ancho de la zona de interés.

Las formulas que se emplearon para calcular los datos descritos en el párrafo anterior, son las siguientes:

$$d_{vert} = \frac{T_{LHough} * d_{CI}}{d_C} \quad (2)$$

$$d_{horz} = \frac{T_{LHough} * d_{CI}}{d_{HORZ}} \quad (3)$$

$$T_{corInf} = \frac{T_{LHough} * d_{CI}}{T_{CORINF}} \quad (4)$$

$$E_{Par} = \frac{T_{LHough} * d_{CI}}{E_{PAR}} \quad (5)$$

Donde:

d_{vert} = Distancia (en píxeles) entre la corona superior y la inferior de la botella en la imagen.

T_{LHough} = Tamaño (en píxeles) de la línea detectada con la transformada de Hough.

d_C = El diámetro (en centímetros) de la corona superior de botella de vidrio.

d_{CI} = Distancia (en centímetros) entre la corona superior y la inferior de la botella de vidrio.

d_{horz} = Distancia (en píxeles) entre las paredes internas del cuello de la botella en la imagen.

d_{HORZ} = Distancia (en centímetros) entre las paredes internas del cuello de la botella de vidrio.

T_{corInf} = Tamaño (en píxeles) de la corona inferior de la botella de vidrio.

T_{CORINF} = Tamaño (en centímetros) de la corona inferior de la botella de vidrio.

E_{PAR} = Espesor (en centímetros) del vidrio en el cuello de la botella de vidrio.

E_{Par} = Espesor (en píxeles) del vidrio en el cuello de la botella en la imagen.

Finalmente, para extraer la zona de interés de la imagen, se realizó el mismo procedimiento descrito al final del apartado 4.3.5, excepto que ahora el desplazamiento vertical y horizontal, largo y ancho de la zona de interés se calcula en tiempo de ejecución del programa, logrando así, que siempre sea posible extraer la zona de interés de la imagen independientemente de las variaciones de escala de las imágenes.

Después de hacer las modificaciones anteriores surgió otro problema, ya que al variar la escala de las imágenes, las dimensiones de la zona de interés cambian, provocando que el número de datos (píxeles) de entrada de la RBF no sea el mismo y por lo tanto no sea posible entrenar a la RBF con lo obtenido hasta este punto.

Para solucionar el problema descrito anteriormente se plantean tres posibles soluciones:

1. Estandarizar el número de datos de entrada de la RBF, es decir, definir el tamaño máximo de la zona de interés que puede haber dentro de todas las imágenes disponibles y cuando se detecte una imagen con una zona de interés de menor tamaño que el máximo, se deberán completar las entradas faltantes de la RBF con ceros.
2. Determinar y obtener vectores característicos de la zona de interés de imágenes de distinta escala de botellas con y sin defecto, y con ellos entrenar una RBF.
3. De acuerdo a que la zona de interés obtenida hasta este punto, solo contiene los bordes que representan el defecto de la botella y tal vez algunos otros que permanecieron después de emplear las operaciones morfológicas descritas anteriormente, debido a la forma u estructura deforme que esos bordes tienen, entonces es posible emplear alguna metodología y/o técnica con la cual sea posible:
 1. Etiquetar objetos (píxeles binarios conectados) dentro de la zona de interés.
 2. Calcular el área de cada uno de los objetos.
 3. Obtener el porcentaje que ocupa el objeto (defecto) dentro de la imagen (zona de interés), esto para que dicha característica se invariante ante el cambio de escala de la imagen.
 4. Eliminar los objetos cuyo valor de área esté por debajo del valor promedio, logrando así que la zona de interés solo contenga (en caso de existir) el objeto que representa el defecto de la botella, y con ello, se clasifique a la botella como defectuosa o no defectuosa en caso contrario.

De las posibles soluciones planteadas anteriormente, se decidió desarrollar y emplear la tercera y para ello, se programó una librería de enlace dinámico en lenguaje C y se instanció dentro del software desarrollado hasta este momento en LabVIEW®.

Después se seleccionaron veinte imágenes diferentes y se calculó: 1) el área que ocupa el defecto en cada imagen, 2) el porcentaje que dicha área ocupa dentro de la zona de interés y 3) el promedio de los veinte porcentajes anteriores. Este último se calculó para comparar y eliminar los objetos, cuyo porcentaje del área del defecto de la botella bajo análisis está por debajo de este promedio y con ello lograr que dentro de la zona de interés sólo esté el objeto que representa el defecto (si este existe) de la botella.

Finalmente, para clasificar la imagen solo basta con identificar si dentro de la zona de interés hay algún objeto, y si es así la imagen se clasificara dentro de la clase “botella con defecto” de lo contrario en la clase “botella sin defecto”

Después de realizar las modificaciones anteriores, finalmente, se realizaron algunas pruebas en el software desarrollado en la plataforma LabVIEW® 2009 (en la sección de anexos se muestra el diagrama a bloques del software desarrollado en dicha plataforma).

En la Figs.39, 41, 43 y 45, se muestra la interface grafica mostrada al ejecutar el software desarrollado y además se observa el resultado obtenido por dicho software al procesar imágenes de botellas con defecto y en la Figs. 40, 42 y 44, se muestra el resultado obtenido al procesar imágenes de botellas sin defecto.

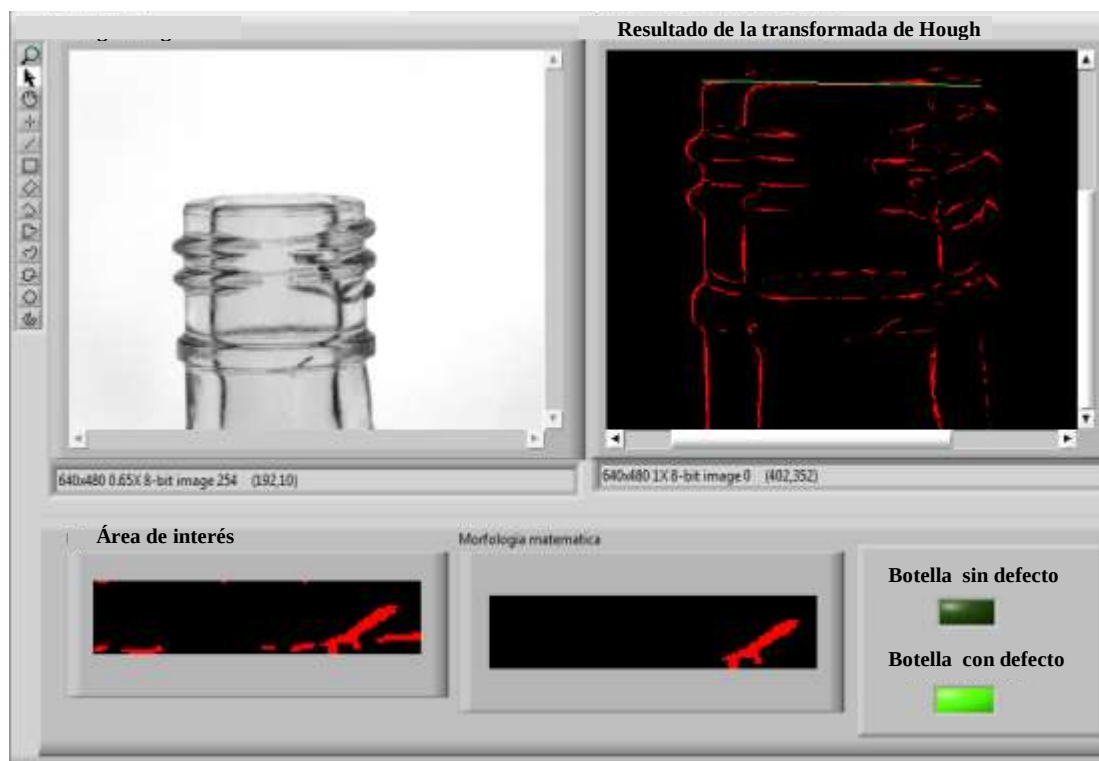


Figura 27. Resultado obtenido al procesar una imagen de una botella con defecto, con la metodología propuesta

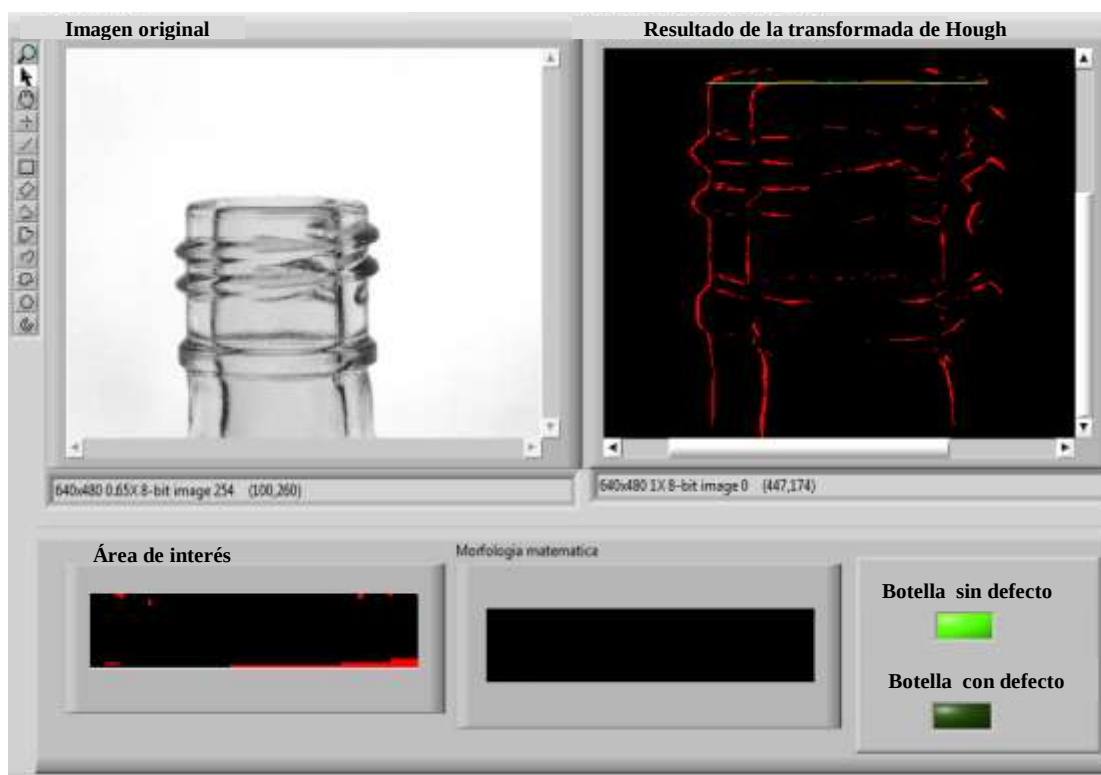


Figura 28. Resultado obtenido al procesar una imagen de una botella sin defecto, con la metodología propuesta

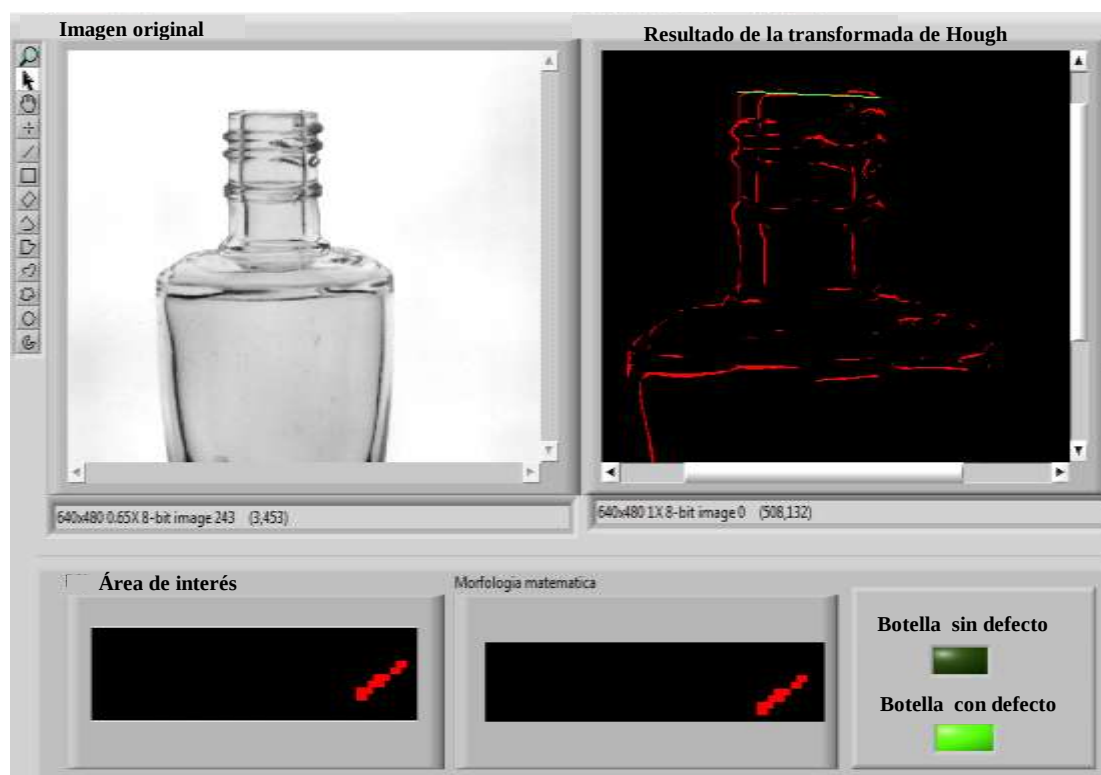


Figura 29. Resultado obtenido al procesar una imagen de una botella con defecto, con la metodología propuesta

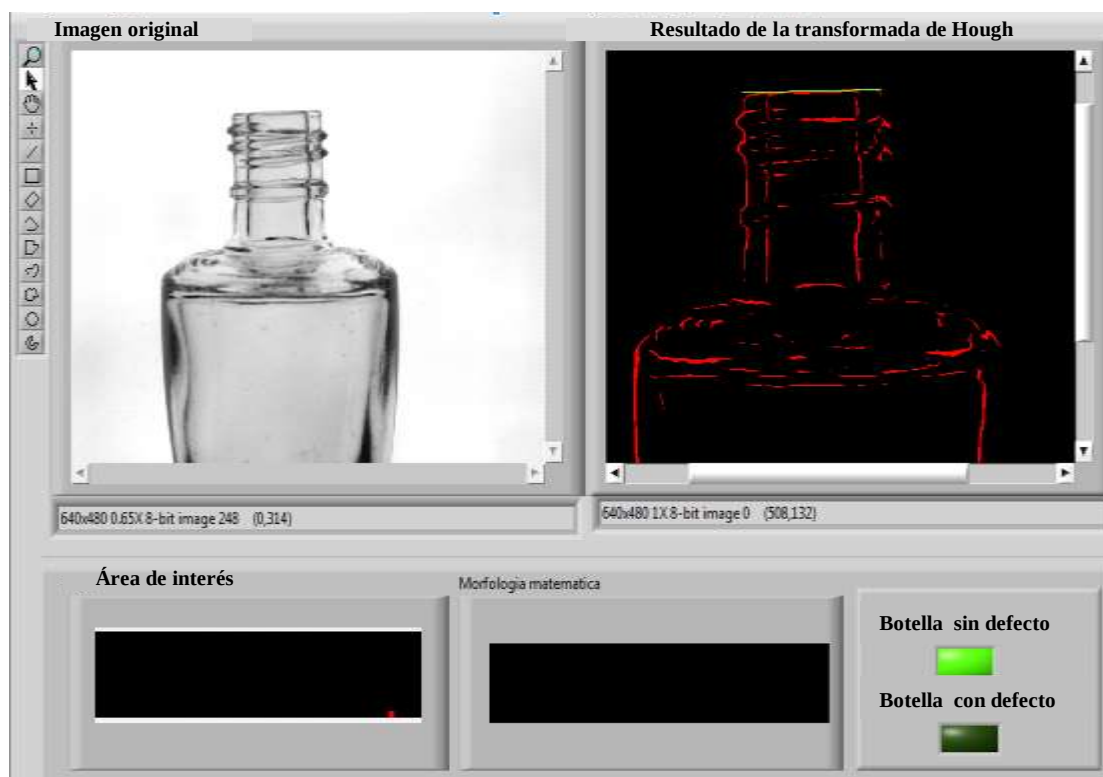


Figura 30. Resultado obtenido al procesar una imagen de una botella sin defecto, con la metodología propuesta

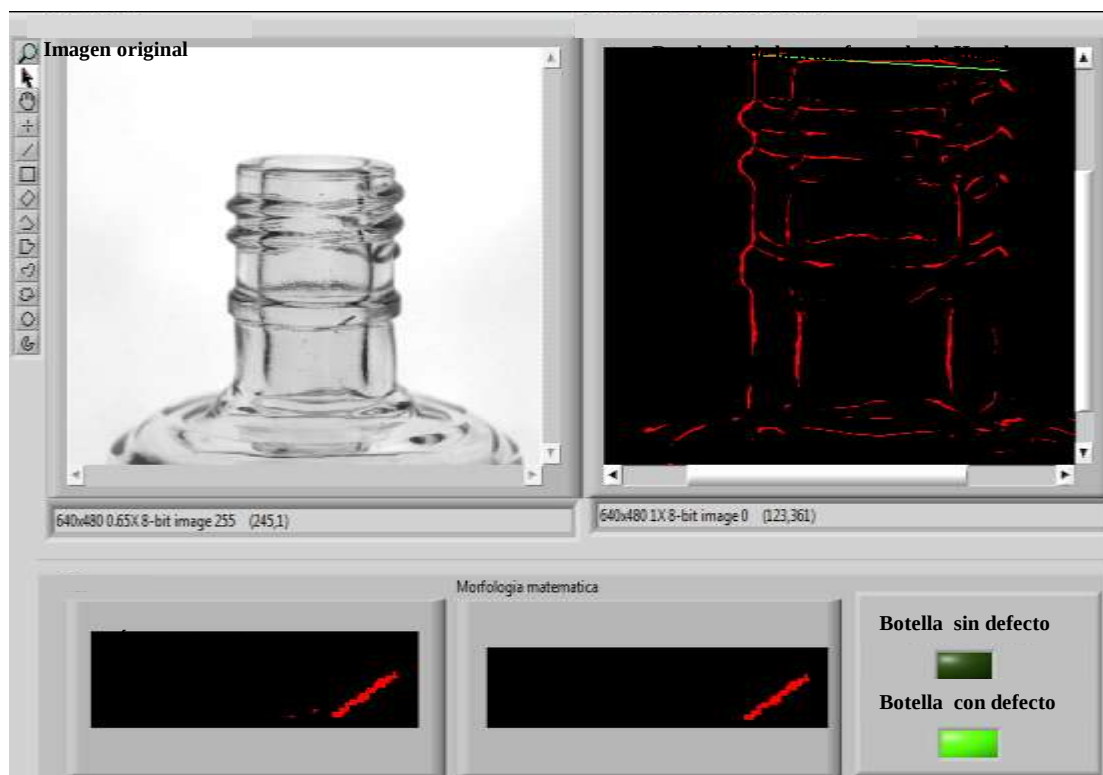


Figura 31. Resultado obtenido al procesar una imagen de una botella con defecto, con la metodología propuesta

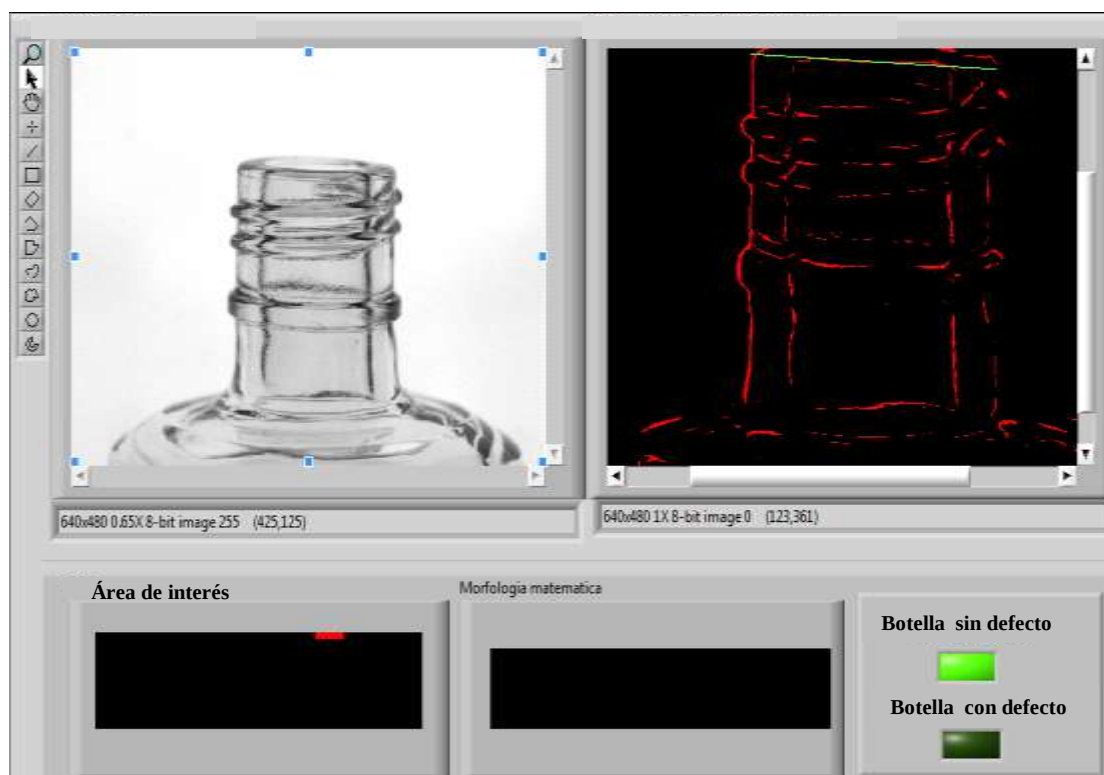


Figura 32. Resultado obtenido al procesar una imagen de una botella sin defecto, con la metodología propuesta

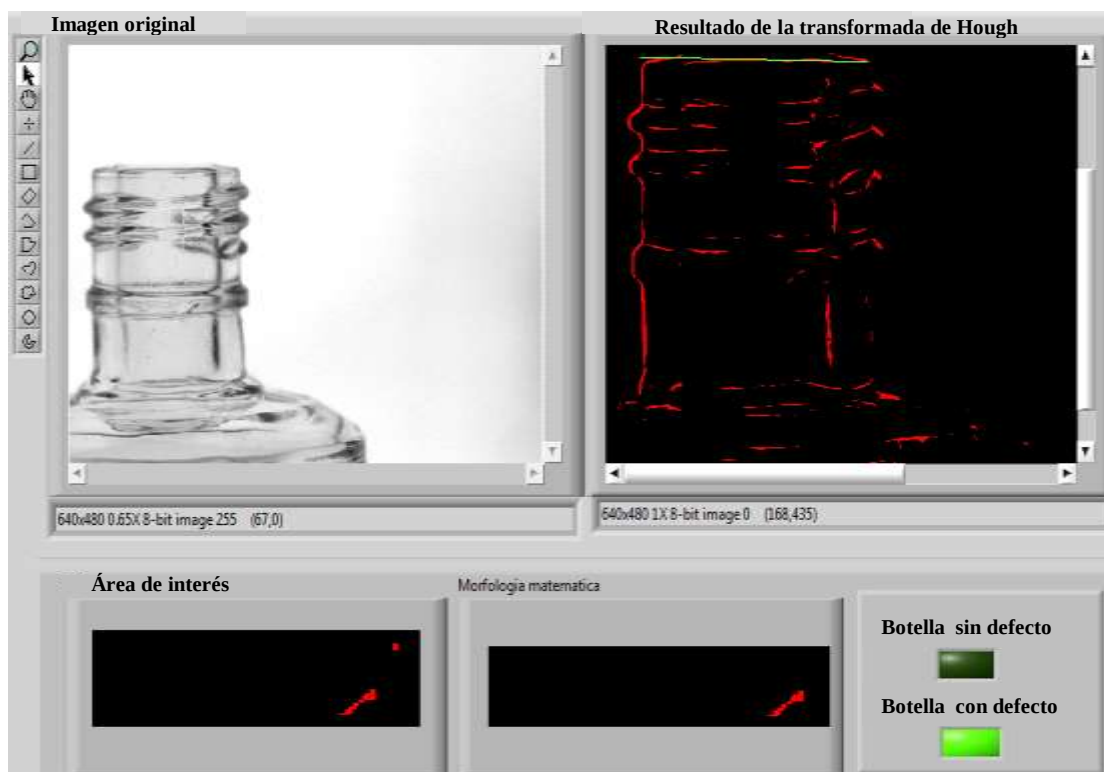


Figura 33. Resultado obtenido al procesar una imagen de una botella con defecto, con la metodología propuesta

Capítulo 5

En este capítulo se describen los resultados de las pruebas realizadas en las etapas de la metodología de solución propuesta.

Capítulo 5. Resultados

5.1. Resultados en pruebas de detectores de esquinas: Harris y SUSAN

Al emplear los detectores de esquinas de Harris y SUSAN y comparar los resultados de cada uno ellos, se observó (ver Tabla 21) que con ninguno de ellos es posible encontrar algún punto en común dentro de todas las imágenes disponibles, por lo que ninguno de ellos se empleará en este trabajo.

Tabla 1. Detectores de esquinas de Harris y SUSAN	
Tipo de Detector	Puntos en común en todas las imágenes disponibles
SUSAN	0
Harris	0

5.2. Resultados en pruebas realizadas con la transformada de Hough

Los resultados obtenidos después de haber detectado y comparado las cinco líneas de mayor tamaño detectadas por la transformada de Hough en cada una de las imágenes, se presentan en la Tabla 22, en donde se puede observar que la única línea que se mantiene en todas las imágenes disponibles, es la ubicada en la parte superior la imagen, la cual representa la corona superior de la botella.

Tabla 2. Resultados de prueba con la transformada de Hough.	
	Considera dentro de las cinco de mayor tamaño en todas las imágenes
Línea lateral derecha	No
Línea lateral izquierda	No
Línea superior	SI

Debido a que la línea que representa el borde de la corona superior de la botella, es considerada dentro de las cinco de mayor tamaño en cada una de las imágenes, entonces es posible utilizarla como referencia para poder segmentar y extraer la zona de la imagen en donde se encuentran (en caso de existir) los bordes de la imagen que representan el defecto de la botella.

Después de acotar la transformada de Hough para que solo detecte líneas rectas con ángulos de inclinación de -3, -2, -1, 0, 1, 2 y 3 grados, nuevamente se realizaron pruebas con todas las imágenes, pero ahora solo para obtener la línea de mayor tamaño dentro de cada imagen y se observó que solo en doce del total de las imágenes, la línea superior no fue la más grande (ver Tabla 23).

Tabla 3. Prueba 2 con transformada de Hough.	
	Número de ocasiones considerada como la de mayor tamaño
Línea superior	148
Otra línea	12

5.3. Resultados en pruebas de clasificación de botellas con la red neuronal de función de base radial

Tras sólo haber considerado imágenes de botellas a la misma escala, no fue posible entrenar y probar la RBF con patrones de imágenes con diferentes escalas, ya que al variar la escala de la imagen varía el número de píxeles dentro de la zona de interés, lo cual provoca que haya una variación en el número de datos de entrada de la RBF.

Debido a lo descrito en el párrafo anterior, solo se seleccionaron imágenes con las mismas características para probar entrenar y probar la RBF, pero se buscará otro método, metodología y/o técnica con la que se pueda clasificar todas las imágenes de las botellas sin importar la escala.

En la Tabla 24 se muestran los resultados obtenidos en la clasificación de botellas tras haber entrenado, validado y probado la RBF, variando el número de neuronas de la capa oculta y el número de iteraciones de entrenamiento. Las imágenes utilizadas para el entrenamiento fueron quince, de las cuales siete eran de botellas con defecto y las restantes de botellas sin defecto.

Tabla 4. Resultados de prueba de clasificación con red neuronal de base radial.					
No. de prueba	No. de neuronas en la capa oculta	No. de iteraciones	Aciertos en		Porcentaje de Efectividad (%)
			Envases sin defecto	Envases con defecto	
1	10	500	21	3	66
2	20	500	24	3	75
3	25	1000	26	4	83
4	30	500	27	6	91
5	35	1000	28	8	100
6	40	1000	28	8	100

Como se puede observar en la Tabla 24, después de entrenar una RBF con nueve mil neuronas de entrada, treinta y cinco neuronas en la capa oculta y dos neuronas de salida, durante quinientas iteraciones de entrenamiento se alcanzó un porcentaje de efectividad del 100% al clasificar treinta y seis imágenes de botellas, de las cuales veintiocho no presentaban ningún defecto y las restantes sí.

5.4 Resultados de pruebas de clasificación de imágenes de botellas de diferente escala por medio del cálculo del momento de orden cero (área).

Finalmente para verificar y comprobar que las modificaciones hechas en la etapa de extracción de la zona de interés, para lograr que la metodología procese de manera correcta y entregue un resultado satisfactorio al clasificar imágenes de botellas a diferente escala, se realizaron pruebas con imágenes de botellas de diferente escala y los resultados se muestran en la Tabla 25.

Tabla 5. Resultado de prueba de clasificación por medio del cálculo del área ocupada por el defecto.	
Total de envases analizados	160
Envases con fallas analizados	55
Envases sin fallas analizados	105
Envases con fallas detectadas	54
Envases sin fallas detectadas	103
Falsos positivos	1
Falsos negativos	2

De acuerdo a los resultados de la Tabla 25, el porcentaje de efectividad obtenido al clasificar las 160 imágenes disponibles en esta investigación fue del 98.12%.

El motivo por el cual no se obtiene un porcentaje de efectividad mayor al obtenido (98.12%), se debe a que las dos imágenes clasificadas erróneamente dentro de la clase “botella con defecto”, tenían una mancha demasiado grande dentro de la zona de interés, provocada por la textura del vidrio y por la ubicación de la cámara fotográfica. La imagen de la botella mal clasificada en la clase “botella sin defecto”, se debe a que el defecto se encuentra muy pegado a los bordes provocados por la sombra de las paredes internas del cuello de la botella y no se podía apreciar completamente, provocando que la cantidad de píxeles que representan el defecto, estuviera por debajo del promedio.

El error descrito anteriormente es posible solucionarlo, reduciendo la zona de interés de la imagen y colocando una cámara fotográfica más.

Aun cuando los errores descritos anteriormente son posibles solucionarlos, se tomaron en cuenta para obtener el porcentaje de efectividad de la metodología propuesta en este trabajo y aun así, resultó conveniente emplear el cálculo del área ocupada por el objeto (unión de bordes que representan el defecto), cómo una característica útil para clasificar la imagen, ya que dicho porcentaje se encuentra dentro del margen establecido en los trabajos consultados y comentados en capítulo 2. En la Tabla 26, se muestra en porcentaje de efectividad obtenido en cada uno de esos trabajos.

Tabla 26. Porcentaje de efectividad de trabajos desarrollados para detectar defectos en botellas de vidrio	
Título del trabajo	Porcentaje de efectividad
Fault detection by a gaussian neural network with reject options in glass bottle production [1]	97%
A glass bottle defect detection system without touching [2]	98%
Automated Visual Inspection of Glass Bottles Using Adapted Median Filtering [3]	85%
Development of a Computerized Method to Inspect Empty Glass Bottle [4]	97 %
Detección de fallas en envases de vidrio no cilíndricos utilizando localización de bordes mediante la herramienta Matlab [5]	89%
Análisis y detección de fallas estructurales en envases de vidrio utilizando procesamiento digital de imágenes [6]	100 %
Vision artificial para detección automática de fallas estructurales en botellas de vidrio (metodología propuesta en este trabajo).	98%

Capítulo 6

En este capítulo se exponen las conclusiones del trabajo y se describen las modificaciones que se pretender realizar en el futuro.

Capítulo 6. Conclusiones y trabajos futuros

6.1 Conclusiones

- De los métodos, metodologías y/o técnicas empleadas en otros trabajos relacionados con la detección y clasificación de defectos en botellas de vidrio, que se emplearon y probaron, para tratar de resolver parcialmente el problema planteado en este trabajo, sólo las operaciones morfológicas: erosión, apertura y clausura, y el operador de Sobel, fueron los que debido a sus resultados se pudieron emplear en este trabajo para solucionar parte del problema.
- De los detectores de SUSAN y Harris no se obtuvieron los resultados deseados ya que con ninguno de ellos fue posible localizar alguna esquina en común dentro de todas las imágenes disponibles en este trabajo, debido a que las imágenes no presentan la misma información ya que al capturar las imágenes, las botellas se rotaron (sobre su eje, no dentro de la imagen).
- Con la metodología propuesta para resolver el problema planteado en el capítulo 1 de esta investigación si fue posible, detectar y clasificar imágenes de botellas de vidrio con y sin defectos físicos en la corona inferior, obteniendo un porcentaje de efectividad por encima del 98%.
- Para obtener un porcentaje de efectividad mayor al obtenido (98.12 %) con la metodología propuesta en este trabajo, es necesario colocar tres cámaras fotográficas alrededor de la botella de vidrio, ya que con esto es posible reducir la zona de interés de la imagen para evitar que la sombra provocada por las paredes internas del cuello de las botellas, disminuya la visibilidad del defecto (si es que existe) de la botella.
- Aunque la metodología propuesta en este trabajo, solo está enfocada a detectar defectos físicos en la corona inferior de botellas de vidrio, es posible, adaptarlo para detectar defectos en otras partes de la botella siempre y cuando se defina la zona de interés de la imagen a procesar.
- El desarrollo de la metodología propuesta, se realizó sobre la plataforma LabVIEW® 2009, y aunque se utilizaron algunas herramientas para procesamiento de imágenes de ésta plataforma, se tuvieron que programar y crear librerías de enlace dinámico en lenguaje C de la transformada de Hough, y de la red neuronal de

función de base radial para obtener un menor tiempo de respuesta del software y con ello sea posible migrar la aplicación directamente a un Compaq Rio ®.

- Aunque la red neuronal de función de base radial propuesta para clasificar las imágenes de las botellas utilizadas en este trabajo, al final se descartó, no implica que con ésta no se hayan podido clasificar las imágenes de manera correcta, sino todo lo contrario, ya que el porcentaje de efectividad obtenido al clasificar treinta y seis imágenes fue del 100%.

El motivo por el cual se descartó emplear dicha red neuronal, fue porque debido a que durante el transcurso de la investigación se decidió agregar a la metodología la característica de ser invariante ante el posible cambio de escala de las imágenes, lo cual implicó que la zona donde se encuentra el defecto variara de tamaño y por lo tanto el número de datos (píxeles) también, provocando que no se pueda diseñar una RBF ya que el número de neuronas de entrada varia también tendría que variar conforme cambie la escala de la imagen y aunque eso se pudo solucionar entrenando la red neuronal con vectores característicos utilizando los momentos de Hu, no se realizó.

6.2 Trabajos futuros

- Ampliar el set de muestras a por lo menos 50 envases, de los cuales 15 fueran sin defecto y los restantes fueran con diferentes defectos en la corona inferior de la botella, para entrenar a la red neuronal de función de base radial, con un número mayor de patrones y se obtengan mejores resultados al clasificarlos y además se pueda clasificar botellas con distinto tipo de defecto y escala.
- Realizar una variante de la Transformada de Hough, para que el tiempo y número de operaciones computacionales, se reduzca y se obtenga el resultado de la clasificación en el menor tiempo posible.
- Emplear otras metodologías o métodos de reconocimiento de patrones, como una máquina de soporte vectorial o una red neuronal de segundo orden ya que estas son invariantes a la translación y rotación de los patrones de entrada, para comparar los resultados y comprobar cuál de ellos se adapta mejor a este trabajo.

Referencias bibliográficas

- [1] C. Firmin , D. Hamad, J.G. Postaire, y R.D. Zhang, Fault detection by a gaussian neural network with reject options in glass bottle production”, Proc. SPIE Vol. 2785, p. 152-162, Vision Systems: New Image Processing Techniques, Philippe Refregier, September 1996.
- [2] Yun-Yan Wang, Guang-Da Su y Hui-Min Ma, A GLASS BOTTLE DEFECT DETECTION SYSTEM WITHOUT TOUCHING. Proceedings of the First International Conference on Machine Learning and Cybernetics, Beijing, 4-5 November 2002.
- [3] D. Mery, O. Medina, Automated visual inspection of glass bottles using adapted median filtering. Proceedings of the International Conference on Image Analysis and Recognition 2004 (ICIAR 2004), Porto, Sept. 29 - Oct 01, 2004.
- [4] Huan-jun Liu; Yao-nan Wang, Development of A Computerized Method to Inspect Empty Glass Bottle, International Symposium on Computer Science and Computational Technology, 2008.
- [5] Braulio Andrés Ruiz Flores, Freddy Daniel Ullauri Ulloa y Patricia Chavez: Detección de fallas en envases de vidrio no cilíndricos utilizando localización de bordes mediante la herramienta Matlab, Artículos de Tesis de Grado – FIEC, Mar 10, 2009.
- [6] Genaro Gutiérrez Becerril, Análisis y detección de fallas estructurales en envases de vidrio utilizando procesamiento digital de imágenes, Tesis de Grado, I.P.N., 2008.
- [7] J.H. Sossa, Rasgos Descriptores para el Reconocimiento de Objetos, I.P.N., 2006.
- [8] Pasarrello Gianfranco, Imágenes médicas, Adquisición, Análisis, Equinoccio, 1999.
- [9] N. Otsu, “A threshold Selection Method from Gray-Level Histogramas”, IEEE Transaction on Systems, Man and Cybernetics, 9(1):62-66.1979.
- [10] R. M. Haralick y L.G. Shapiro, Computer and Robot Vision, Addison-Weslwy, 1991.
- [11] R.C. Gonzales y R. E. Woods, Digital Image Proccesing, Prentice Hall, New Jersey, 2002.
- [12] R. Jain, R. Kasturi y B.G. Schunck, Machine Vision, 1a. ed., McGraw-Hill Science/Engineering/Math., 1995.
- [13] H.P. Movarec, Towards Automatic Visual Obstacle Avoidance, Fifth International Join Conference on Artificial Intelligence, pp. 548, 1977.
- [14] L. Kitchen y A. Rosenfeld, “Gray Level Corner Detection, Pattern Recognition Letters, 1:95-102, 1982.
- [15] W. Forstner, “A Faecture Based Correspondence Algorithm for Image Matching”, International Arch. Photogrametry, Remote Sensing, 26:150-166, 1986.
- [16] C. Tomasi y T. Kanade. Detection and Tracking of Point Features Technical Report CMU-CS-91-132, Carnegie Melon University, Pittsburgh, PA, abril de 1991.
- [17] R. Deriche y G. Giraudon, “A Computational Approach for Corner and Vertex Detection”, International Journal on Computer Vision, 10(2):101-124, 1993.
- [18] J. Shi y C. Tomasi , Good Features to TRack Proceedings of the International Conference on Computer Vision, 37(2):151-172, 2000.
- [19] B. Zitova, J. Kautsky, G. Peters y J. Flusser, “Robust Detection of significant Points in Multi Frame Images”, Pattern recognition Letters, 20:199-206, 1999.
- [20] C. Schmit, R. Mohr y Ch. Bauckhage, “Evalation of Interest Point Detectors”, International CONference on Computer Vision and Pattern Recongition, pp. 593-600.
- [21] C.G., Mohr, R., Bauckhage, C., A combined corner and edge detector. Alvey Vision Conference, (1998).
- [22] J. Song y Michael R. Lyu, A Hough transform based line recognition method utilizing both parameter space and image space, Pattern Recognition 38 (2005) 539-552, Elsevier, 2004.
- [23] R. O. Duda y P.E. Hart, Use of Hough transform to detect lines and curves in pictures, Commun. ACM 15(1) 11-15, 1972.

- [24] J. Illingworth y J. Kittler, A survey of the Hough transform, *Computer Vision Graphics Image Process*, pp. 87-116, 1998.
- [25] Minkowski, H. Allgemeine Lehrsätze über die konvexe Polyeder, *Nachr. Ges. Wiss., Göttingen*, 198-219, 1897.
- [26] G. Matheron, *Eléments pour une Théorie des Milieux Poreux*, Masson, Paris, (1967).
- [27] J. Serra, *Image Analysis and Mathematical Morphology*, Theoretical Advances, Academic Press, (1988).
- [28] Al Bovik, *The Essential guide to Image Processing*, Academic Press Elsevier, pp.79-86 ,2009.
- [29] McCulloch, W. & Pitts, W., A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics*, 115-133, 1943.
- [30] Rosenblatt, F. (1962). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Spartan Books, Washington DC.
- [31] Widrow, Bernard, & Hoff, Marcian, E. (1960) *Adaptive Switching Circuits*, 1960 IRE WESCON Convention Record, New York: IRE pp. 96-104.
- [32] Widrow, Bernard. (1962). Generalization and information Storage in Networks of Adaline Neurons. In M.C. Yovits, G.T. Jacobi, & G.D. Goldstein (Eds.), *Self-Organizing Systems*. Washington D.C.: Spartan Books.
- [33] Minsky, M. & Papert, S. (1988). *Perceptrons*, Cambridge: MIT Press.
- [34] Anderson, J. A. & Rosenfeld, E. (Eds.) (1990). *Neurocomputing: Foundations of Research*, Cambridge: MIT Press.
- [35] Hopfield, J.J. (1982). Neural networks and physical systems with emergent collective computational abilities, *Proceedings of the National Academy of Sciences*, 79, 2554-2558.
- [36] Hopfield, J.J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons, *Proceedings of the National Academy of Sciences*, 81, 3088.
- [37] Rumelhart, D.E., Hinton, G.E. & Williams, R.J. (1986). Learning internal representations by error propagation, in D. Rumelhart and J. McClelland, editors. *Parallel Data Processing*, Vol.1, Chap. 8, the M.I.T. Press, Cambridge, MA, pp. 318-362.
- [38] G. X. Ritter and J. N. Wilson. "Image Algebra and its Relationship to Neural Network", *SPIE Vol. 1098 Conf. Proc. Aerospace Pattern Recognition*, Orlando, 1989, p.p. 90-101.
- [39] Raducanu, B., Grana, M. & Sussner, P. (2001). Advances in Mobile Robot Self-Localization using Morphological Neural Networks", in *Proceedings of IFAC 2001*, pp. 146-151, Cheju Island, South Korea.
- [40] Yáñez-Márquez, C., *Memorias Asociativas basadas en Relaciones de Orden y Operadores Binarios*. Tesis doctoral, CIC-IPN, México, 2002.
- [41] Yáñez Márquez, C. & Díaz-de-León Santiago, J.L. (2003). *Introducción a las memorias asociativas*, Serie Research on Computing Science, Vol. 6, IPN, México. ISBN: 970-360116-2.
- [42] Instituto Politécnico Nacional. (Página consultada el 28 de Septiembre de 2008), [On-line]. Dirección URL: <http://www.aldape.com.mx/sp/rnab.htm>
- [43] Broomhead and Lowe, Multivariable functional interpolation and adaptative networks. *Complex System*, 2:321-355, 1988.
- [44] S.M. Smith and J.M. Brady, SUSAN - a new approach to low level image processing. *Int. Journal of Computer Vision*, 23(1):45--78, May 1997.

Anexos

Anexo A

**Diagrama a bloques del software desarrollado en la plataforma
LabVIEW® 2009**



Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:55 p.m.

Connector Pane

Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi



Dibuja la linea encontrada por la transformada de Hough

Cord X = Vector con las coordenadas X de la linea. a partir del punto inicial

Cord y= Vector con las coordenadas Y de la linea. a partir del punto inicial.

Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

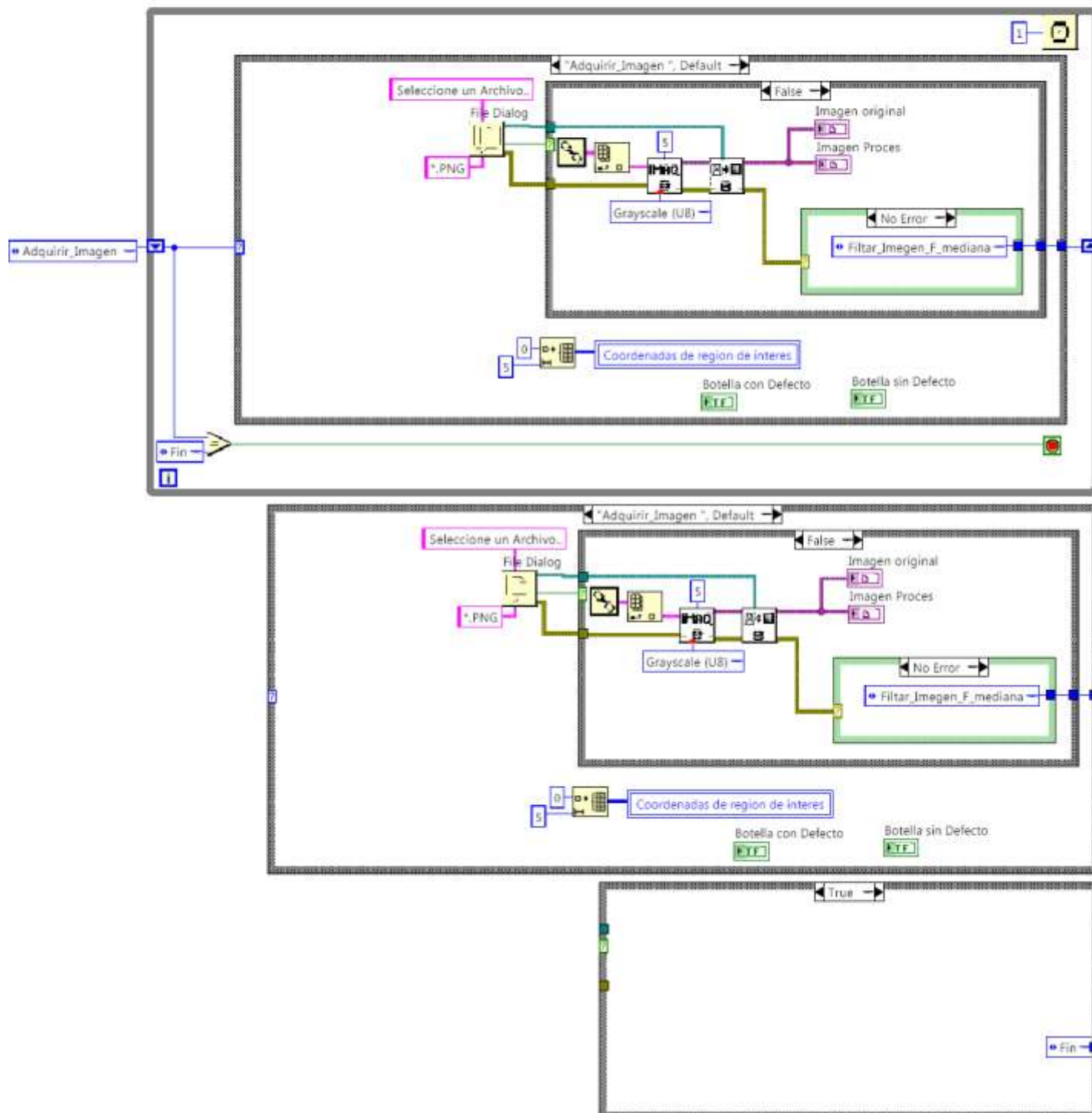
C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\

MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:55 p.m.

Block Diagram

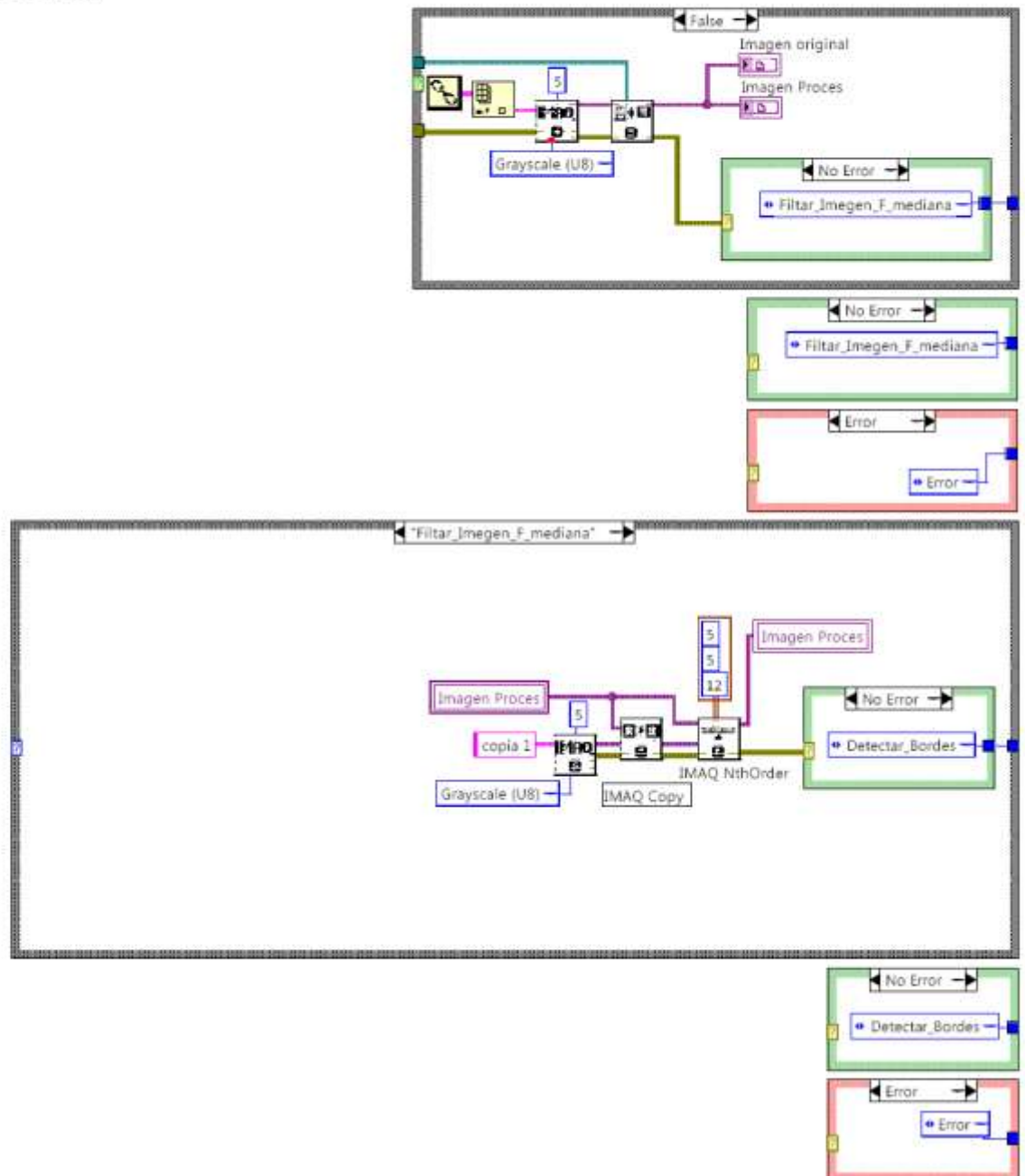


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:55 p.m.

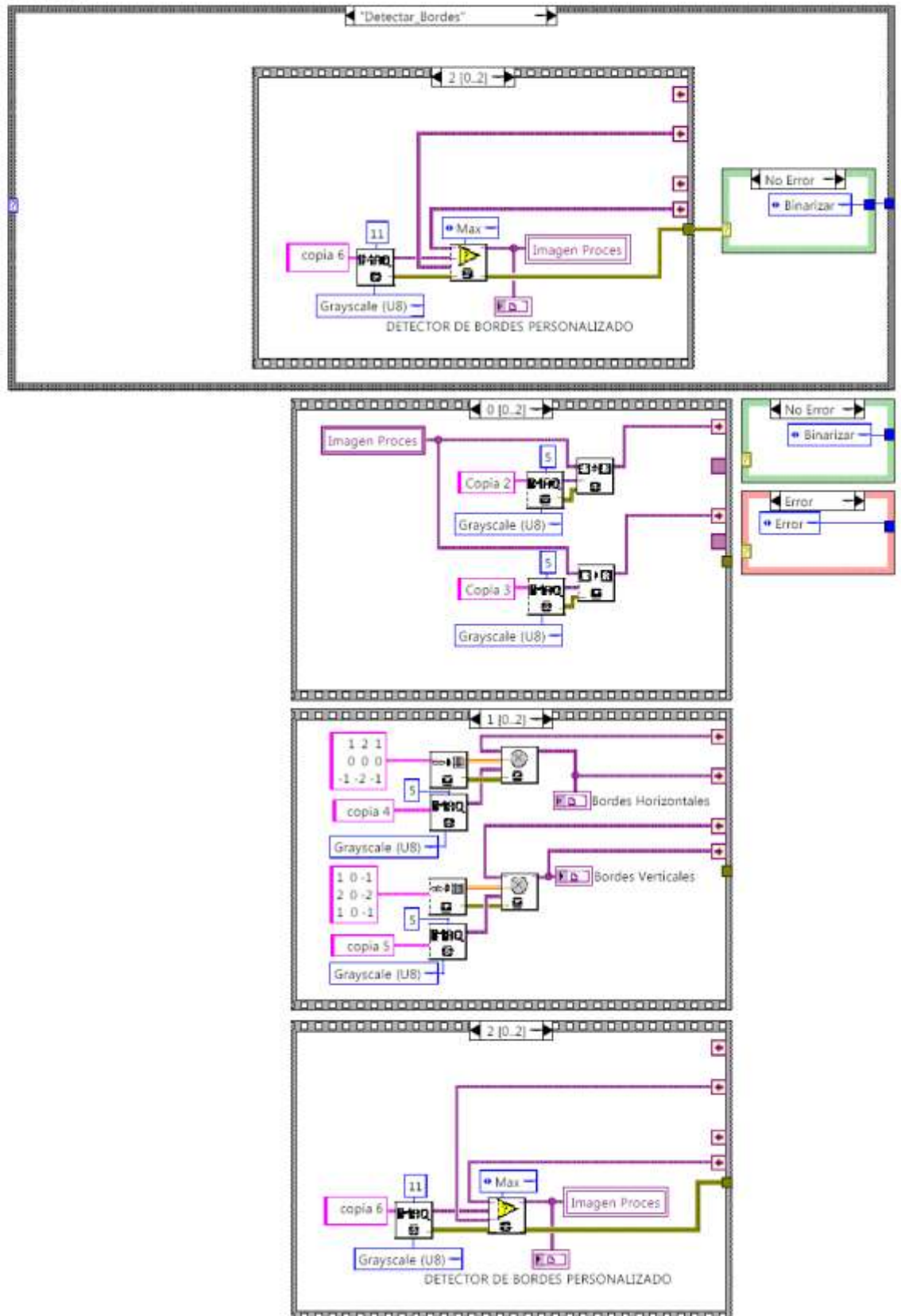


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:55 p.m.



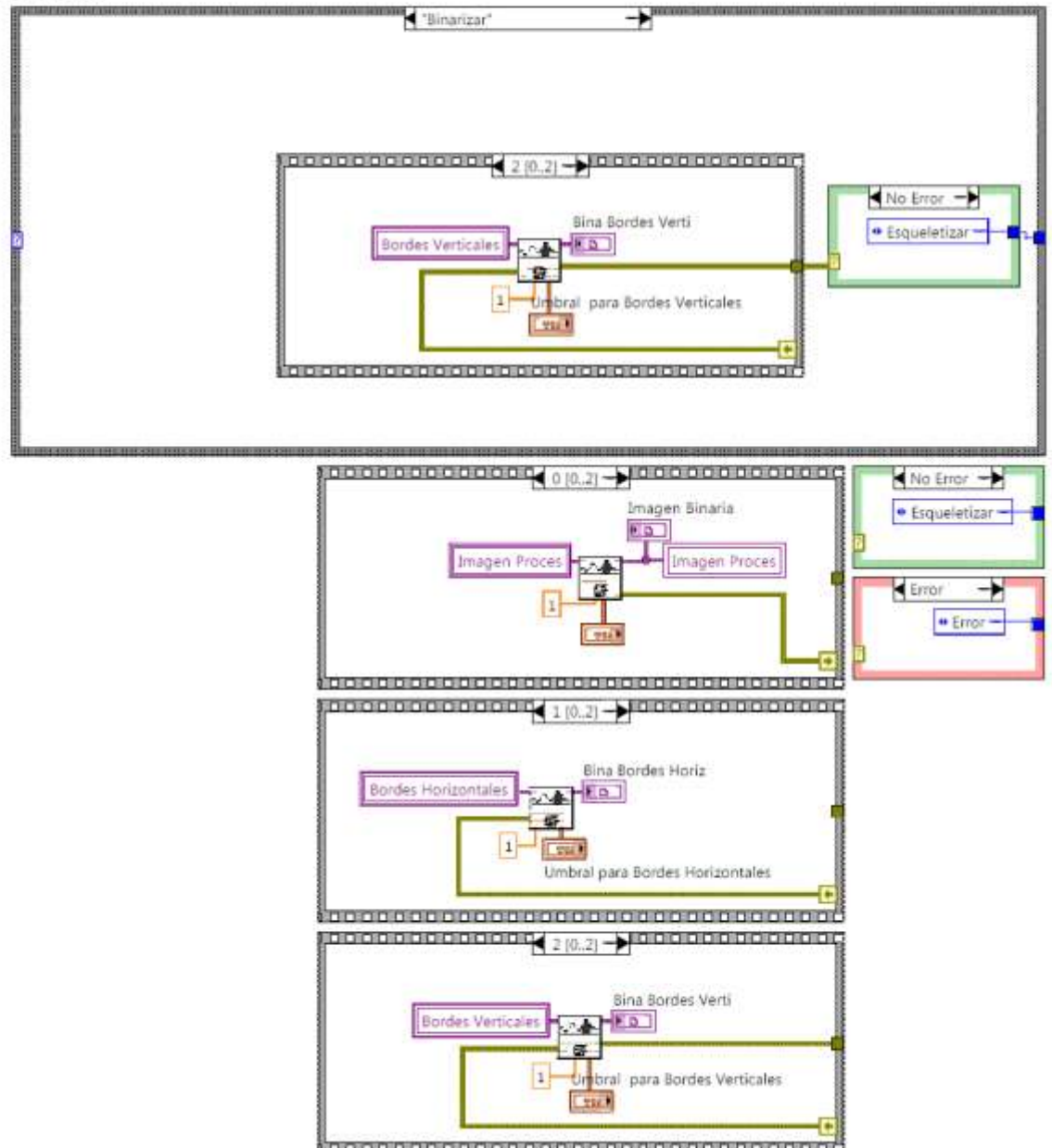


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:55 p.m.



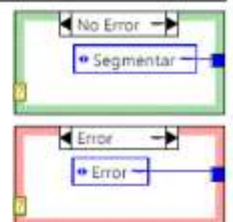
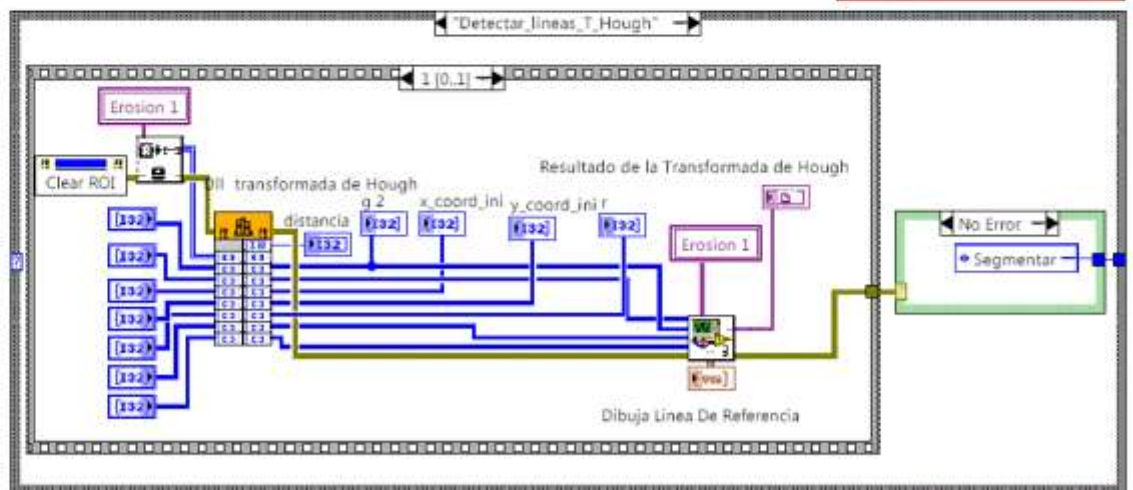
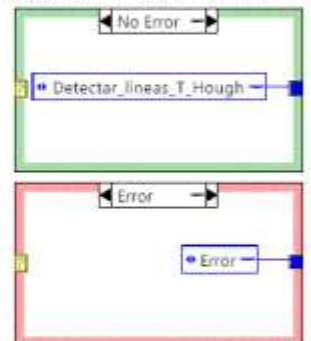
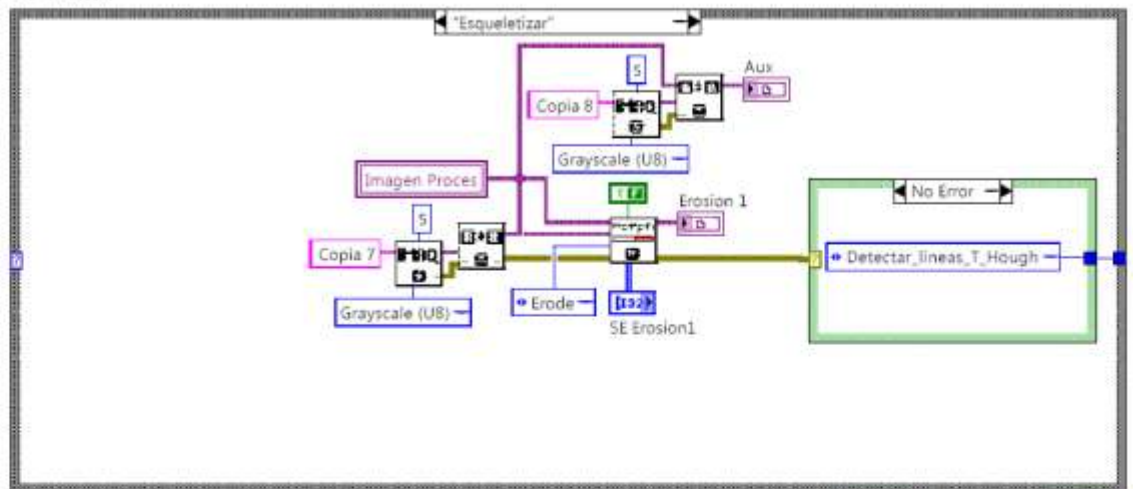


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:55 p.m.

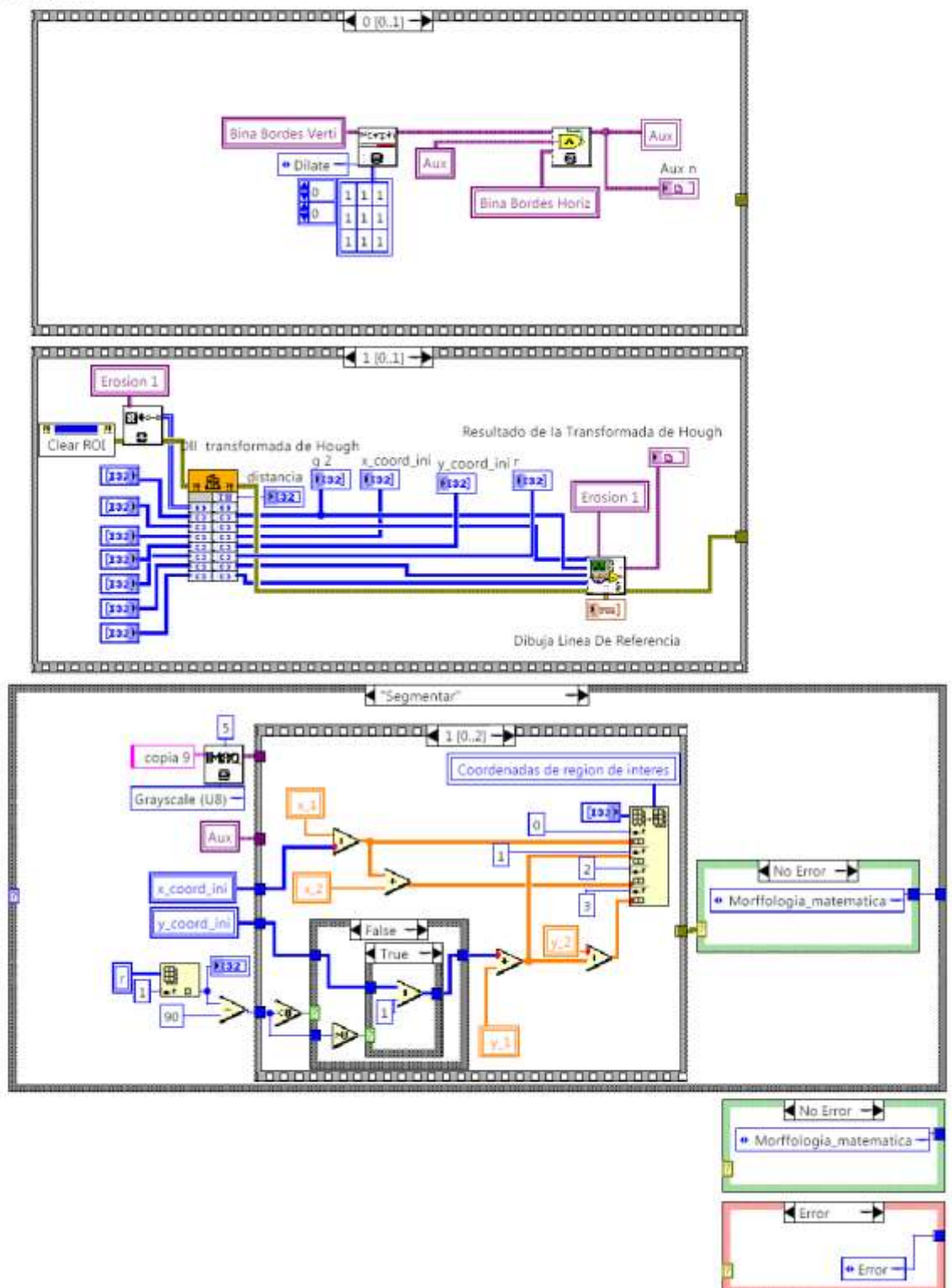


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:56 p.m.



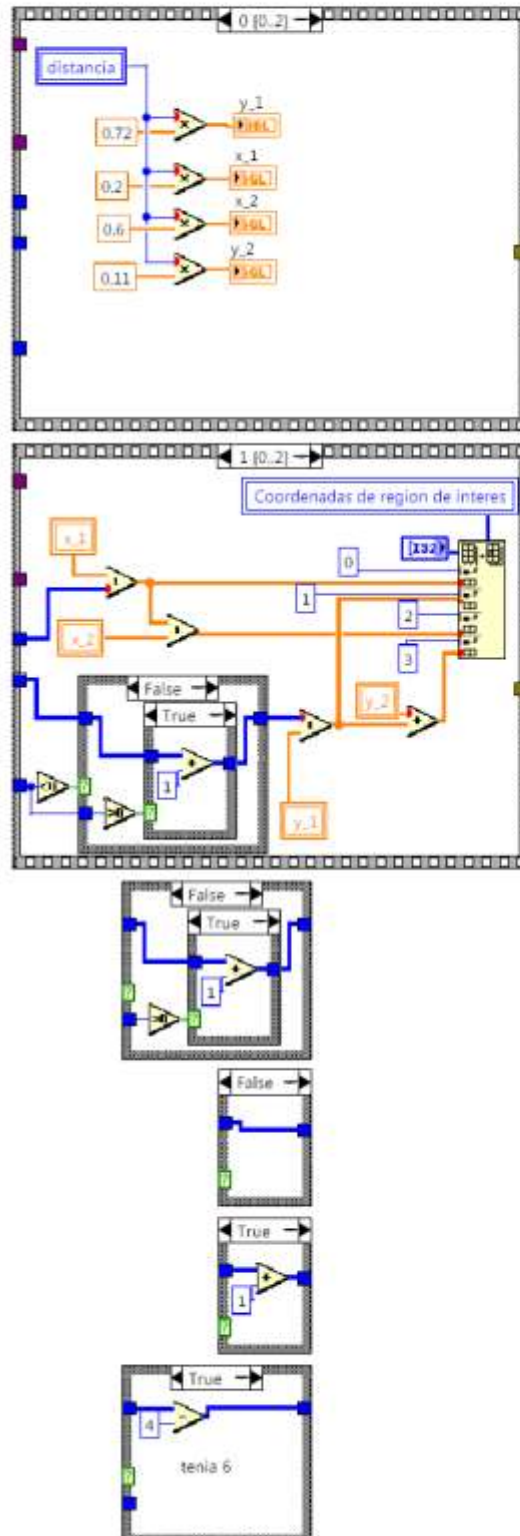


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:56 p.m.

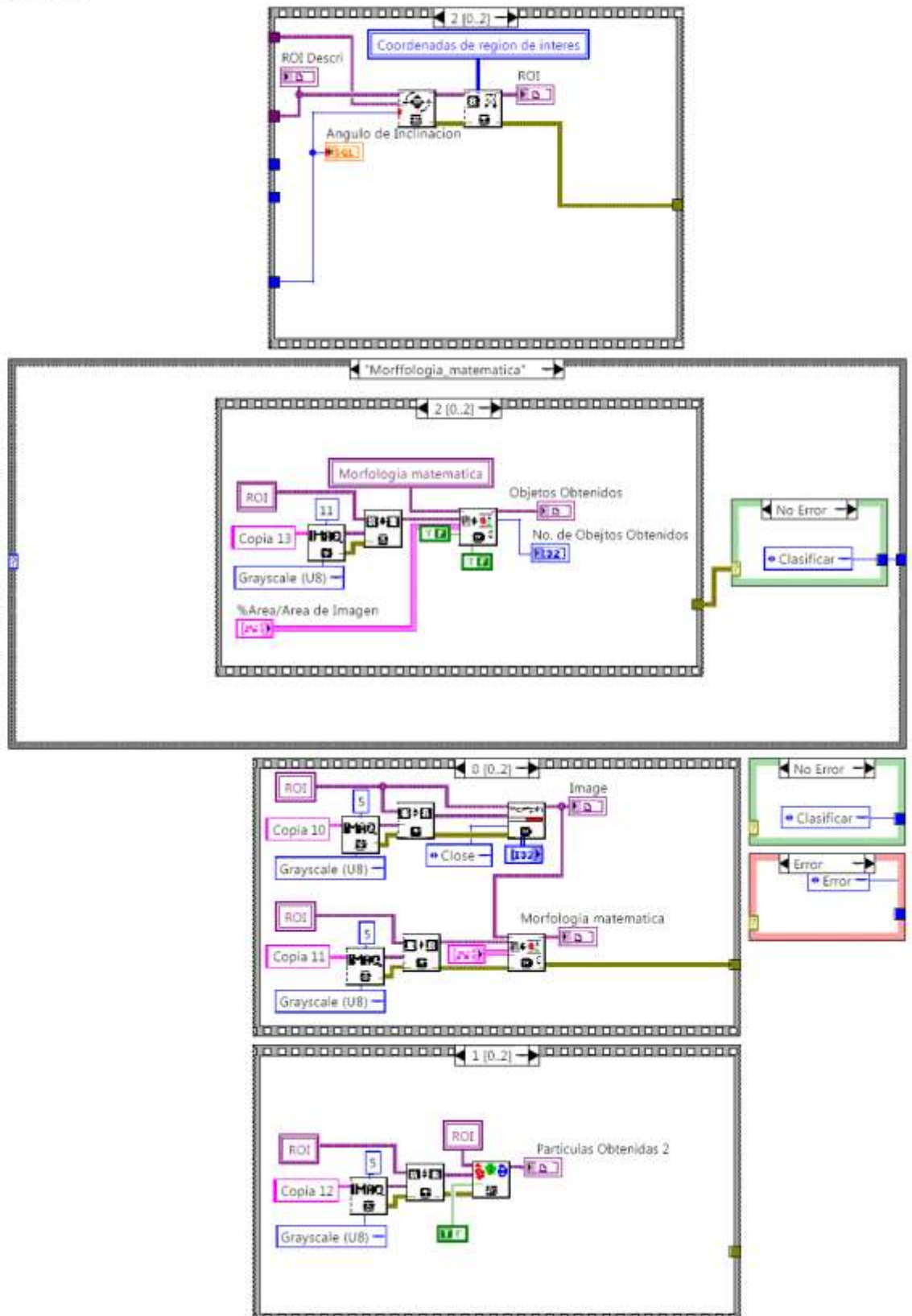


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:56 p.m.



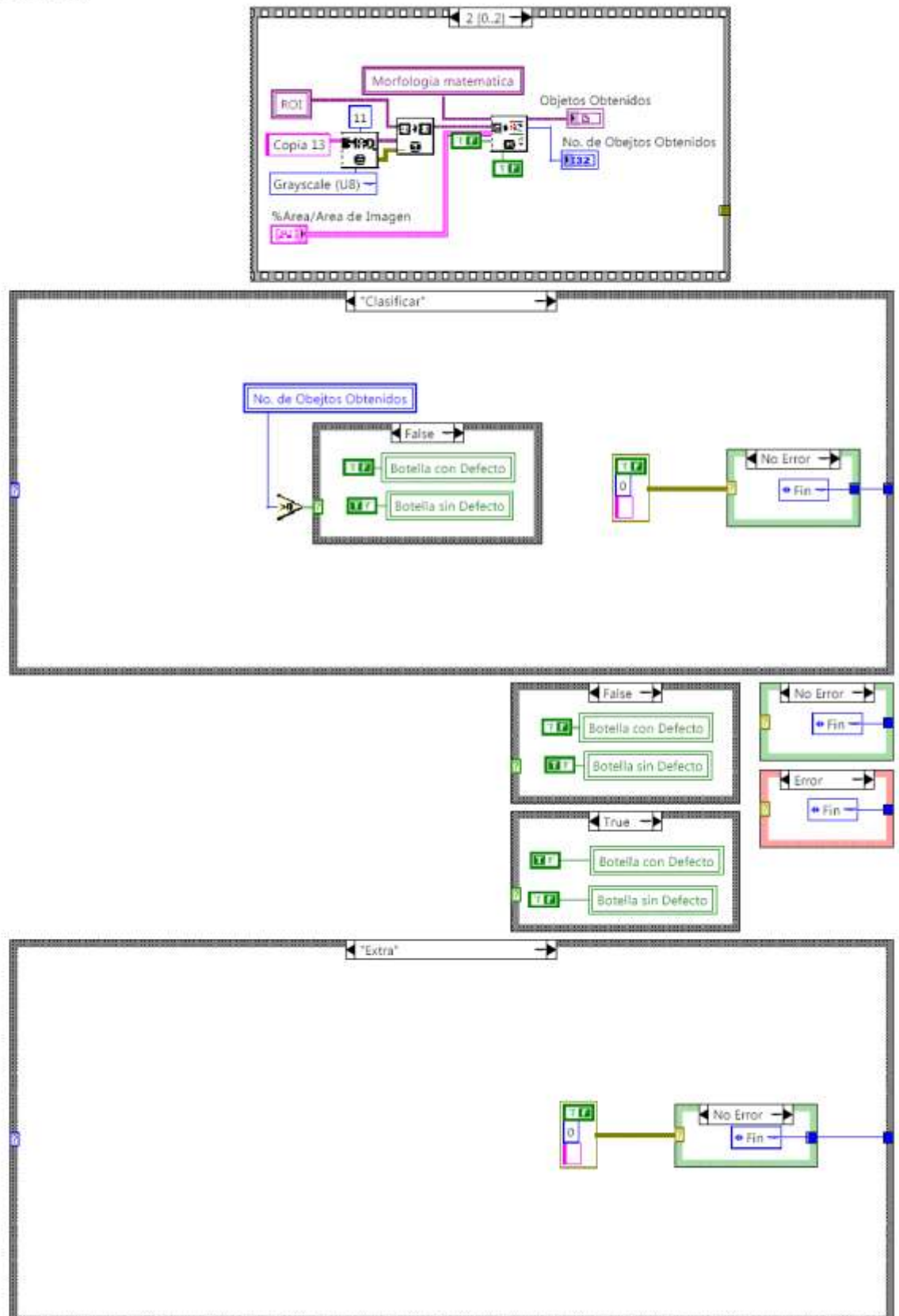


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:56 p.m.



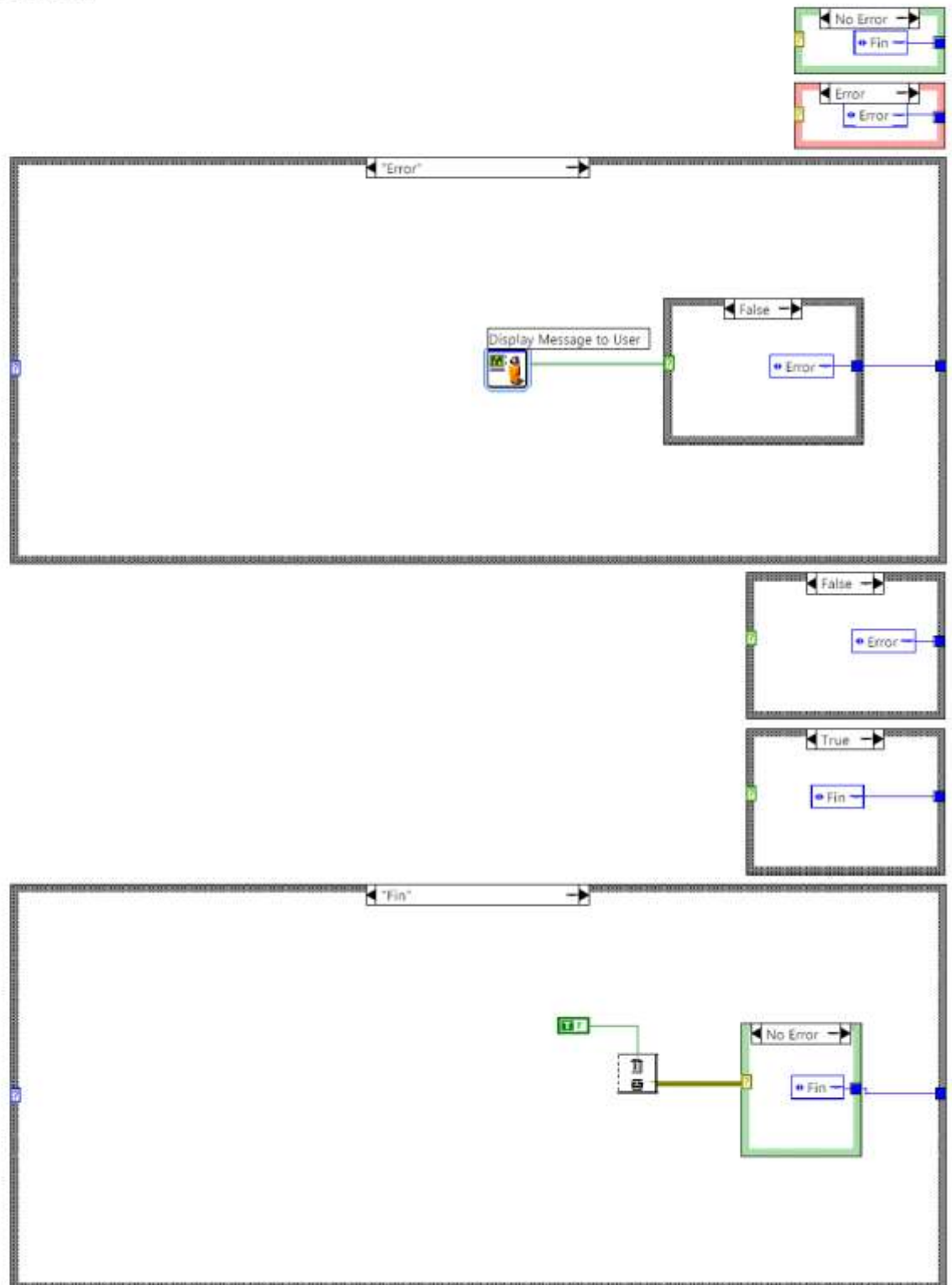


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:56 p.m.



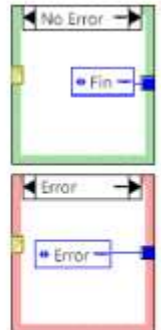


Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

C:\Users\Fernando\Desktop\RESPAL~1\DETECC~1\Users\Fernando\DOCUME~1\LABVIE~1\MAQUIN~1\Metodologia_para_deteccion_de_fallas_estructurales_en_botellas.vi

Last modified on 18/01/2011 at 10:00 a.m.

Printed on 20/01/2011 at 02:56 p.m.



Anexo B

**Código fuente de la librería de enlace dinámico de la
transformada de Hough, desarrollada en NI
LabWindows CVI ® 2009**

```

//=====
//
/// Title: Transformada_de_Hough.h
/// Purpose: Detecta líneas con ángulos de inclinación de -3 -2 -1 0, 1 2 3
/// grados dentro de una imagen
//
// Created on: 09/10/2010 at 11:33:15 a.m. by Fernando Ramirez Neyra.
// Copyright: All Rights Reserved.
//
//=====

#ifndef __Transformada_de_Hough_H__
#define __Transformada_de_Hough_H__

#ifdef __cplusplus
    extern "C" {
#endif

//=====
// Include files

#include "cvideo.h"

//=====
// Constants
#define pi 3.141592

//=====
// Types
    int p;
    int theta;

//=====
// External variables

//=====
// Global functions

    int Detectar_Lineas(char imagen[][640], int *g, int *t, int *x_coord_ini, int *y_coord_ini, int *r, int l[620],int q[620]);

#ifdef __cplusplus
    }
#endif

#endif /* ndef __Transformada_de_Hough_H__ */

```

```

//=====
//
// Title:   Transformada_de_Hough.c
// Purpose: Detecta lineas con angulos de inclinacion de -3 -2 -1 0, 1 2 3
//          grados dentro de una imagen.
//
// Created on: 09/10/2010 at 11:33:15 a.m. by Fernando Ramirez Neyra.
// Copyright: All Rights Reserved.
//=====
// Include files
#include <ansi_c.h>
#include <utility.h>
#include <math.h>
#include <utility.h>
#include "Transformada_de_Hough.h"
#define puntos_max 7
#define num_angulos 7;

//=====
// Global variables

    int rows=200;
    int columns=640;
    int rmax=800;
    int max1[puntos_max];
    int ps[puntos_max];
    int thetas[puntos_max];
    int angulos[7]={87,88,89,90,91,92,93};
    int acc[810][181];

//=====
// Global functions

int Detectar_Lineas(char imagen[][640], int *g, int *t, int *x_coord_ini, int *y_coord_ini, int *r, int l[620], int q[620])
{
    int res;
    for (int x=0;x<810;x++)
        for (int y=0;y<181;y++)
            acc[x][y]=0;
    for (int x=0;x<columns;x++)
    {
        for (int y=0;y<rows;y++)
        {
            if(imagen[y][x]==1)
                for(int z=0;z<7;z++)
                {
                    res=RoundRealToNearestInteger(x*cos(angulos[z]*pi/180)+y*sin(angulos[z]*pi/180));
                    if(res<max && res>0)
                        acc[res][angulos[z]]=acc[res][angulos[z]]+1;
                }
        }
    }

    float ys[puntos_max];
    for (int i=0;i<puntos_max;i++)
    {
        max1[i]=acc[0][0];
        for (int x=0;x<rmax;x++)
        {
            for (int y=0;y<181;y++)
            {
                if(acc[x][y]>=max1[i])
                {
                    max1[i]=acc[x][y];
                    ps[i]=x;
                    thetas[i]=y;
                }
            }
        }
        acc[ps[i]][thetas[i]]=0;
    }
}

```

```

for (int x=0;x<puntos_max;x++)
{
    ys[x]=(-x+1/tan(theta[s[x]*pi/180])+(ps[x]/sin(theta[s[x]*pi/180))); //Coordenada y
    lx[x]=x; //Coordenada x
}
int min=ys[0];
for (int i=0;i<puntos_max;i++)
{
    if(ys[i]<=min)
    {
        min=ys[i];
        p=ps[i];
        theta=theta[i];
    }
}

//=====Puntos de linea completa=====

for (int x=0;x<620;x++)
{
    q[x]=RoundRealToNearest Integer(-(x+1)/tan(theta*pi/180)+(p/sin(theta*pi/180))); //Coordenada y
    lx[x]=x; //Coordenada x
}

//=====Busca el primer borde de la botella=====
int pixeles_promediados=620;
int coory=0;
float suma;
int aux_1[620];

for (int h=0;h<620;h++)
    aux_1[h]=0;

for (int i=7;i<612;i++)
{
    coory=q[i];
    suma=0;
    for(int j=9;j<=9;j++)
        for(int k=-1;k<=1;k++)
            suma=suma+imagen[coory+j][lx[i+k]];
    if(suma>0)
        aux_1[i]=1;
}
int i=0;
while(aux_1[i]==0)
    i=i+1;

y_coord_ini[0]=q[i]; //coordenada y del primer borde de la botella
x_coord_ini[0]=lx[i]; //coordenada x del primer borde de la botella

//=====Busca el ultimo borde de la botella=====
int i2=611;
int yfinal;
int xfinal;
while(aux_1[i2]==0)
    i2=i2-1;

yfinal=aux_1[i2];
xfinal=i2;
for (int x=0;x<(i2-i);x++)
{
    g[x]=i+x; //Almacena los puntos de la recta a partir del primer punto o borde
    t[x]=q[i+x]; //de la botella de vidrio
}
r[0]=p;
r[1]=theta;
return i2-i;
}

```

```

//=====
// DLL main entry-point functions

int __stdcall DllMain (HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpvReserved)
{
    switch (fdwReason) {
        case DLL_PROCESS_ATTACH:
            if (InitCVIRTE (hinstDLL, 0, 0) == 0)
                return 0; /* out of memory */
            break;
        case DLL_PROCESS_DETACH:
            CloseCVIRTE ();
            break;
    }

    return 1;
}

int __stdcall DllEntryPoint (HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpvReserved)
{
    /* Included for compatibility with Borland */
    return DllMain (hinstDLL, fdwReason, lpvReserved);
}

```


Anexo C

**Código fuente de la librería de enlace dinámico de la red
neuronal de función de base radial, desarrollada en NI
LabWindows CVI ® 2009**



```
//=====
//
// Title:   Dll_RNA_de_funcion_de_base_radial.h
// Purpose: Clasifica una imagen de entrada en alguna de las clases
//          "botella con defecto" o "botella sin defecto"
//
// Created on: 14/10/2010 at 03:58:18 a.m. by Fernando Ramirez Neyra
// Copyright: All Rights Reserved.
//
//=====

#ifndef __Dll_RNA_de_Base_Radial_H__
#define __Dll_RNA_de_Base_Radial_H__

#ifdef __cplusplus
extern "C" {
#endif

//=====
// Include files

#include "cvideo.h"

//=====
// Constants

//=====
// Types

//=====CNeurona
typedef struct
{
    int i,j;
    double *w;
    double umbral;
    double salida;
    double delta;
    double *centroide;
    double distancia;
    double phi;
}CNeurona;

//=====CCapa

typedef struct
{
    int No_Neurons; //Almecenar el numero de neuronas de la capa
    CNeurona *ONeurona; //Vector de la clase Neurona
}CCapa;

//=====
//=====
// Global functions
int Your_Functions_Here (int x);
int Clasificar(int a, char Imagen[88][25],int tam_x, int tam_y, double *resultado,char Imagen_regreso[]);

#ifdef __cplusplus
}
#endif

#endif /* ndef __Dll_RNA_de_Base_Radial_H__ */
```



```
//=====
//
// Title:   DLL_RNA_de_funcion_de_base_radial.c
// Purpose: Clasifica una imagen de entrada en alguna de las clases
//          "botella con defecto" o "botella sin defecto"
// Created on: 09/10/2010 at 11:33:15 a.m. by Fernando Ramirez Neyra.
// Copyright: All Rights Reserved.
//=====
//=====
// Include files
#include <ansi_c.h>
#include <utility.h>
#include <math.h>
#include "DLL_RNA_de_funcion_de_base_radial.h"

//=====
// Global variables
int No_Capas=3;
int m_No_Neu_Entrada=2200;
int m_No_Neu_Ocultas=35;
int m_No_Neu_Salida=2;
int m_No_Patrones=2200;
int num_centroides=0;
char *Patron;
CCapa *ocapa;

//=====
// Global functions
int Clasificar(int a, char Imagen[88][25],int tam_x, int tam_y, double *resultado,char Imagen_regreso[])
{
    Patron=(char*)malloc((m_No_Neu_Entrada+2)*sizeof(char));

//=====Lee la imagen=====
    int aux=0;
    for(int y=0;y<tam_y;y++)
    {
        for(int x=0;x<tam_x;x++)
        {
            Patron[aux]=Imagen[y][x];
            Imagen_regreso[aux]=Imagen[y][x];
            aux=aux+1;
        }
    }
    num_centroides=m_No_Neu_Ocultas;
    ocapa=(CCapa*)malloc((No_Capas)*sizeof(CCapa));
    ocapa[0].No_Neurons=m_No_Neu_Entrada;
    ocapa[1].No_Neurons=m_No_Neu_Ocultas;
    ocapa[2].No_Neurons=m_No_Neu_Salida;

//=====Crea las neuronas de cada capa=====
    for(int i=0;i<No_Capas;i++)
    {
        ocapa[i].ONeurona=(CNeurona*)malloc(ocapa[i].No_Neurons*sizeof(CNeurona));
    }

//=====Crear_Vectores_de_Pesos y Centroides=====

    for(int j=0;j<ocapa[1].No_Neurons;j++)
    {
        ocapa[1].ONeurona[j].centroide=(double*)malloc(m_No_Neu_Entrada*sizeof(double));
    }
    for(int j=0;j<ocapa[2].No_Neurons;j++)
    {
        ocapa[2].ONeurona[j].w=(double*)malloc((ocapa[1].No_Neurons+2)*sizeof(double));
    }
}
```



```
    }
//=====asignación de los valores de los centroides, distancias, pesos y umbrales=====
//=====Pesos y Umbrales/=====

//Colocar todos los valores de los pesos y umbrales obtenidos en la fase de entrenamiento de la RNA
ocapa[2].ONeurona[0].umbral=-0.535356;
ocapa[2].ONeurona[0].w[0]=-0.0275177;
ocapa[2].ONeurona[0].w[1]=0.0179938;
//
//
//
//
// Faltan colocar los pesos y umbrales de toda la RNA, pero no se colocaran ya que ocupan demasiado espacio
// como para ser mostrados en el archivo impreso.

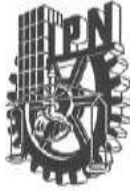
//=====calcula PHi=====
    for(int i=0;i<m_No_Neu_Ocultas;i++) //calcula el valor de phi de todas las neuronas
    {
        ocapa[1].ONeurona[i].phi=0.0;
        for(int j=0;j<m_No_Neu_Entrada;j++)
        {
            ocapa[1].ONeurona[i].phi+=pow(Patron[j]-ocapa[1].ONeurona[i].centroide[j],2);
        }
        ocapa[1].ONeurona[i].phi=exp(-
(ocapa[1].ONeurona[i].phi/(2*pow(ocapa[1].ONeurona[i].distancia,2))));
    }
//=====Calcula la salida de la RNA=====
    for(int j=0;j<ocapa[2].No_Neurons;j++)
    {
        ocapa[2].ONeurona[j].salida=0.0;
        for(int k=0;k<ocapa[1].No_Neurons;k++)
        {
            ocapa[2].ONeurona[j].salida+=(ocapa[1].ONeurona[k].phi*ocapa[2].ONeurona[j].w[k]);
        }
        ocapa[2].ONeurona[j].salida+=ocapa[2].ONeurona[j].umbral;
    }
    resultado[0]=ocapa[2].ONeurona[0].salida;
    resultado[1]=ocapa[2].ONeurona[1].salida;

//=====Libera Memoria=====
    for(int k=0;k<ocapa[1].No_Neurons;k++)
    {
        free(ocapa[1].ONeurona[k].centroide);
    }
    for(int j=0;j<ocapa[2].No_Neurons;j++)
    {
        free(ocapa[2].ONeurona[j].w);
    }
    for(int i=0;i<No_Capas;i++)
    {
        free(ocapa[i].ONeurona);
    }
    free(ocapa);
    free(Patron);
    return 1;
}
//=====

// DLL main entry-point functions
int __stdcall DllMain (HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpvReserved)
{
    switch (fdwReason) {
        case DLL_PROCESS_ATTACH:
            if (InitCVIRTE (hinstDLL, 0, 0) == 0)
                return 0; /* out of memory */
    }
}
```



```
        break;
    case DLL_PROCESS_DETACH:
        CloseCVIRTE ();
        break;
    }
    return 1;
}
int __stdcall DllEntryPoint (HINSTANCE hinstDLL, DWORD fdwReason, LPVOID lpvReserved)
{
    /* Included for compatibility with Borland */
    return DllMain (hinstDLL, fdwReason, lpvReserved);
}
```



INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

SIP-14

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D. F. siendo las 14:00 horas del día 13 del mes de Diciembre de 2010 se reunieron los miembros de la Comisión Revisora de Tesis designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis de grado titulada:

**"VISIÓN ARTIFICIAL PARA DETECCIÓN AUTOMÁTICA
DE FALLAS ESTRUCTURALES EN BOTELLAS DE VIDRIO "**

Presentada por el alumno:

RAMÍREZ

Apellido paterno

NEYRA

Materno

FERNANDO GABINO

nombre(s)

Con registro:

B	0	8	1	3	0	4
---	---	---	---	---	---	---

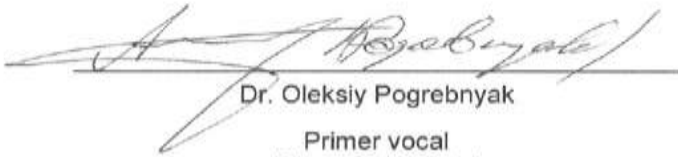
aspirante al grado de: ***MAESTRO EN CIENCIAS EN INGENIERÍA DE CÓMPUTO CON OPCIÓN
EN SISTEMAS DIGITALES***

Después de intercambiar opiniones los miembros de la Comisión manifestaron ***SU APROBACIÓN DE LA TESIS***, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Presidente

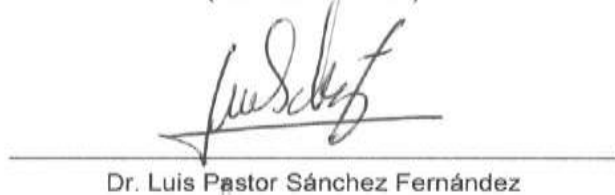
Secretario


Dr. Oleksiy Pogrebnyak

Primer vocal
(Director de Tesis)


Dr. José de Jesús Medel Juárez

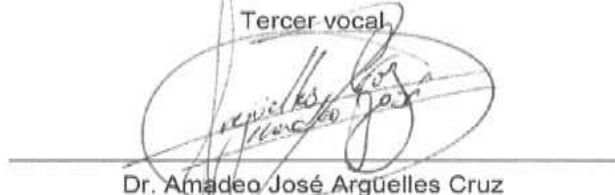
Segundo vocal
(Director de Tesis)


Dr. Luis Pastor Sánchez Fernández

Tercer vocal


Dr. Norberto Flores Guzmán

Suplente


Dr. Amadeo José Argüelles Cruz


M. en C. Pablo Manrique Ramírez

EL PRESIDENTE DEL COLEGIO


Dr. Luis Alfonso Villa Vargas

INSTITUTO POLITÉCNICO NACIONAL
CENTRO DE INVESTIGACIÓN
EN COMPUTACIÓN
DIRECCIÓN



INSTITUTO POLITÉCNICO NACIONAL

SECRETARIA DE INVESTIGACIÓN Y POSGRADO

CARTA DE CESIÓN DE DERECHOS

En la ciudad de México Distrito Federal el día 13 del mes 12 del año 2010, el que suscribe Fernando Gabino Ramírez Neyra alumno del Programa de Maestría en Ciencias en Ingeniería de Cómputo con Opción en Sistemas Digitales con número de registro B081304, adscrito al Centro de Investigación en Computación, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del Dr. Norberto Flores Guzmán y el Dr. Luis Pastor Sánchez Fernández, cede los derechos del trabajo intitulado “Vision artificial para detección automática de fallas estructurales en botellas de vidrio”, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección framirez0300@ipn.mx. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Fernando Gabino Ramírez Neyra