



INSTITUTO POLITÉCNICO NACIONAL



**ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y
ELÉCTRICA**

UNIDAD AZCAPOTZALCO

SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

**"VERIFICACIÓN DIMENSIONAL DE PIEZAS METÁLICAS DE
ACERO, MEDIANTE LA COMPARACIÓN CON UNA PIEZA
PATRÓN, APLICANDO UN SISTEMA DE VISIÓN ARTIFICIAL
PARA VALIDAR SU MANUFACTURA"**

**TESIS QUE PARA OBTENER EL GRADO DE:
MAESTRO EN INGENIERÍA DE MANUFACTURA**

PRESENTA:

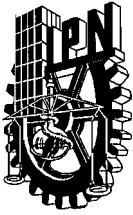
ING. ENRIQUE ADAN NUÑEZ PACHECO

DIRECTORES DE TESIS:

DR. JAIME PACHECO MARTÍNEZ

DR. JOSÉ DE JESÚS RUBIO ÁVILA

FECHA: DICIEMBRE/2009



INSTITUTO POLITÉCNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

SIP-14

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México D.F. siendo las 12 horas del día 27 del mes de Noviembre del 2009 se reunieron los miembros de la Comisión Revisora de Tesis designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de ESIME-UA para examinar la tesis de titulada:

"Verificación dimensional de piezas metálicas de acero, mediante la comparación con una pieza patrón, aplicando un sistema de visión artificial para validar su manufactura"

Presentada por el alumno:

Núñez

Apellido paterno

Pacheco

Apellido materno

Enrique Adan

Nombre(s)

Con registro:

B	0	7	1	3	7	5
---	---	---	---	---	---	---

aspirante al grado de:

Maestro en Ingeniería de Manufactura

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

Director de Tesis

Dr. Jaime Pacheco Martínez

Director de Tesis

Dr. José de Jesús Rubio Ávila

M. en C. Raúl Rivera Blas

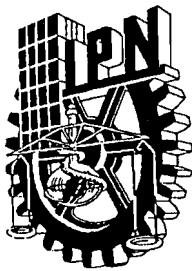
Dr. Bernardino Benito Salmerón Quiroz

Dr. Salvador Antonio Rodríguez Paredes

EL PRESIDENTE DEL COLEGIO

Dr. Jaime Pacheco Martínez





INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México, D. F. el día 02 del mes de Diciembre del año 2009, el que suscribe Enrique Adan Nuñez Pacheco alumno del Programa Maestría en Ingeniería de Manufactura con número de registro B071375, adscrito a SEPI-ESIME-UA, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del Dr. Jaime Pacheco Martínez y el Dr. José de Jesús Rubio Ávila y cede los derechos del trabajo intitulado "Verificación dimensional de piezas metálicas de acero, mediante la comparación con una pieza patrón, aplicando un sistema de visión artificial para validar su manufactura", 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 jpachecoma@ipn.mx; o jrubioa@ipn.mx; ing.enrique.nunez@gmail.com; Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Enrique Adan Nuñez Pacheco

Índice general

Dedicatoria	7
Agradecimientos	8
Resumen	9
Abstract	10
1. Introducción	11
1.1. Justificación	13
1.2. Objetivos	15
1.3. Planteamiento del problema y descripción del sistema	16
2. Estado del arte	18
3. Fundamentos	21
3.1. Técnicas de iluminación	21
3.2. Sensores de video	23
3.2.1. Diferencia entre sensores CCD y CMOS	24

3.2.2.	Cámaras de estado sólido	25
3.2.3.	Cámaras matriciales	28
3.3.	Tarjetas digitalizadoras	29
3.4.	Definiciones de interés	30
3.5.	Modelo de la cámara	32
3.5.1.	Lente simple	33
3.5.2.	Formatos gráficos.	35
3.6.	Imagen binaria	36
3.6.1.	Distancias entre pixeles	39
3.7.	Procesamiento de la imagen	41
3.7.1.	Filtrado o pre-procesamiento	41
3.7.2.	Segmentación o aislamiento de los objetos de interés	42
3.7.3.	Extracción y selección de características	42
3.7.4.	Filtrado espacial	43
3.7.5.	Filtrado e histograma de una imagen	53
3.7.6.	Segmentación de imagenes	58
3.7.7.	Operaciones morfológicas en imagenes binarias	67
3.8.	Protocolo de comunicación	81
4.	Propuesta de solución	84
4.1.	National instruments	84
4.2.	Matlab	86

4.2.1. Selección de la aplicación	87
4.3. Algoritmo a desarrollar	88
4.3.1. Adquisición de la imagen	88
4.3.2. Conversión de la imagen en formato RGB a escala de grises	88
4.3.3. Incremento de la calidad de la imagen adquirida	89
4.3.4. Conversión a imagen binaria.	89
4.3.5. Conversión de segmentos en líneas, áreas, y posterior análisis	91
5. Pruebas y análisis	100
5.1. Conclusiones	106
5.2. Trabajo Futuro	109

Índice de figuras

1.1. Ilusiones ópticas	14
1.2. Vista superior de la pieza	16
1.3. Elementos de un sistema de visión artificial	17
3.1. Tipos de iluminación	22
3.2. Sensor CCD	24
3.3. Sensor CMOS	25
3.4. Funcionamiento de la cámara lineal	26
3.5. Funcionamiento de la cámara TDI	27
3.6. Tarjeta de captura de video	29
3.7. Espacio de colores RGB	30
3.8. Elemento estructural circular de radio 3	31
3.9. Funcionamiento de la cámara oscura	32
3.10. La separación aumenta o disminuye la escala	32
3.11. Lente convexa de radio de curvatura R	33
3.12. Modelo de lente convergente delgada	34
3.13. Escenario de captura de la escena	37

3.14. La misma escena capturada con diferentes resoluciones	38
3.15. Descomposición de una imagen en sus componentes RGB	40
3.16. Funcionamiento de la máscara	44
3.17. Ejemplo de la convolución y la correlación en una imagen	47
3.18. Ejemplo de aplicación de un filtro pasa bajos	48
3.19. Funcionamiento del filtro de mediana	49
3.20. Ejemplo de aplicación de un filtro para detectar líneas horizontales	52
3.21. Imagen de 10 monedas y su histograma correspondiente	55
3.22. Modificación del histograma mediante una LUT	59
3.23. Representación normal de la recta	61
3.24. Imágenes binarias y su representación paramétrica de Hough	62
3.25. Imagen binaria	68
3.26. Imagen binaria con dos conjuntos	69
3.27. Ejemplos de morfología de imágenes	72
3.28. Separación de objetos aplicando dilataciones y erosiones	74
3.29. Apertura y cierre de una imagen	76
3.30. Ejemplos de aplicaciones en morfología	78
3.31. Extracción de la frontera de la imagen	79
3.32. Proceso de relleno de una imagen	80
3.33. Interfaz gráfica de comunicación con la cámara DVT530	83
4.1. Sistema NI compact vision	85

4.2. NI Smart camera	85
4.3. Imagen RGB adquirida por la cámara web	89
4.4. Preprocesado de la imagen	90
4.5. Obtención de líneas y centroide de la pieza	91
4.6. Segmentación usando el método de <i>Otsu</i>	92
4.7. Separación de las áreas de interés	93
4.8. Selección de áreas que se requiere procesar	94
4.9. Graficado de los radios localizados	95
4.10. Superposición de las líneas encontradas en la figura original para analizarlas	96
4.11. Los pixeles mínimo y máximo varían en cada rotación	97
4.12. Áreas que definen los huecos de los dientes	98
4.13. Procesado de la figura para ubicar el último radio	99
5.1. Análisis dimensional en Matlab	101
5.2. Valores de alto y ancho de diente	102
5.3. Alto y ancho de dientes encontrados por la cámara IP	105
5.4. Gráfica de errores con la cámara web	106
5.5. Gráfica de errores con la cámara IP	107

Dedicatoria

A mis padres Enrique y Guadalupe, así como mi hermano Cesar, por forjar un ser de provecho en la incertidumbre de la adolescencia y rebeldía de la juventud, con amor, buenos ejemplos y apoyo en toda circunstancia, quien principia a enfrentarse al mundo, hoy y siempre les da las gracias.

A mi esposa e hijo, por ser ellos el principal motor que motiva mi esfuerzo diario.

A todos mis amigos, por apoyarme, aconsejarme y animarme para terminar el presente trabajo, gracias por contribuir en mi desarrollo personal y profesional.

Agradecimientos

Al Instituto Politécnico Nacional por brindarme las herramientas para forjar mi carrera profesional

Al CONACYT por apoyarme en mi desarrollo, durante mis estudios de posgrado en la Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Azcapotzalco

A la Secretaría de Investigación y Posgrado del IPN por apoyarme como becario en el Programa Institucional de Formación de Investigadores a través del proyecto SIP 20080514 "Modelado de sistemas robóticos con un enfoque de estructura variable".

A los miembros de la comisión revisora: Dr. Jaime Pacheco Martínez, Dr. José de Jesús Rubio Ávila, M. en C. Raúl Rivera Blas, Dr. Bernardino Benito Salmerón Quiroz, Dr. Salvador Antonio Rodríguez Paredes. Por sus comentarios, aportaciones y tiempo invertido en la revisión del trabajo de tesis

A todos aquellos profesores que colaboraron de manera positiva en mi enseñanza y desarrollo de este trabajo

Resumen

La visión artificial comprende la descripción y propiedades del mundo real, a partir de una o varias imágenes bidimensionales captadas.

Un proyecto de visión artificial suele ser de tipo multidisciplinario. La ejecución de las etapas que conforman al proyecto como son la adquisición, preprocesamiento, segmentación, descripción e interpretación, requiere el conocimiento y uso de diversas técnicas entre las cuales se tienen las siguientes:

- Fotografía y óptica
- Procesamiento digital de imágenes
- Reconocimiento de patrones
- Computación gráfica

El presente trabajo muestra una aplicación de la visión artificial a través de la inspección visual de piezas, utilizando software como Matlab para desarrollar la interfaz gráfica, el procesamiento y postprocesamiento de la imagen.

La adquisición de imágenes se realiza mediante dos tipos de cámaras, con el objetivo de realizar una comparación y analizar ventajas y desventajas de la utilización de las mismas en la aplicación desarrollada.

En este proyecto se analizan varios elementos para utilizar en el procesamiento de imágenes, como son los elementos estructurales, los filtros de detección de líneas, operaciones morfológicas, entre otros, para llegar a la correcta segmentación e interpretación de datos, que definen las fronteras de la pieza a analizar.

Abstract

Artificial vision deals with the real world description and properties, from one or multiple bidimensional images acquired.

An Artificial vision project is usually multidisciplinary. Project task execution involves acquisition, preprocessing, segmentation, description and interpretation, and requires the knowledge and use of various techniques, some of them are:

- Picture and optics
- Digital image processing
- Pattern recognition
- Graphic computing

This work shows an application of artificial vision through parts visual inspection, using Matlab software to develop graphic interface, image preprocessing and postprocessing.

Image acquisition is performed through two camera models, which main objective is to compare and analyze pro and cons of using each model in the application developed.

In this project , many characteristics are analyzed to be used in image processing, such as structure elements, line detection filters, morphological operators, etc., to obtain a correct segmentation and data interpretation that allow part boundaries definition.

Capítulo 1

Introducción

Uno de los sentidos más importantes de los seres humanos es la visión, pues es este sentido el que se encarga de describir el entorno físico. De hecho, se calcula que más de 70% de las tareas del cerebro son empleadas en el análisis de la información visual. Casi todas las disciplinas científicas emplean elementos gráficos para transmitir conocimiento. Por ejemplo, en Ingeniería Electrónica se emplean esquemas de circuitos para describirlos, los cuales podrían hacerse mediante texto pero, para los humanos resulta mucho más eficiente procesar imágenes que procesar texto. La visión humana es el sentido más desarrollado y el que menos se conoce debido a su gran complejidad. Es una actividad inconsciente y difícil de saber cómo se produce. De hecho, hoy en día, se carece de una teoría que explique cómo los humanos perciben el exterior a través de la vista.

Desde que se inventó la fotografía se ha intentado extraer características físicas de las imágenes. La Astronomía avanzó enormemente con el análisis de imágenes recibidas por los telescopios. El análisis de radiografías transformó la Medicina. Se podrían citar muchos más ejemplos que durante décadas han transformado la percepción de la Ciencia con el procesamiento de las imágenes, algunas veces por separado y otras de forma multidisciplinaria. Históricamente, la primera vez que se aplicaron las técnicas de procesamiento de imágenes fue en los años veinte, cuando se transmitieron imágenes de fotografías periodísticas a través de cable submarino entre Londres y Nueva York, los primeros sistemas eran capaces de codificar cinco niveles de gris distintos. En 1929 ya se utilizaban quince.

Existen versiones que opinan que el procesamiento de imágenes como tal comenzó en

los años 50-60 debido a la combinación de dos hechos, por una parte la aparición de las computadoras digitales y por otra los programas espaciales, fundamentalmente de los Estados Unidos, y otras que comienza en la década de los 80 con la revolución de la Electrónica, con las cámaras de video CCD y los microprocesadores, junto con la evolución de las Ciencias de la Computación que hace factible la visión artificial.

El primer trabajo cuyo objetivo era analizar directamente imágenes capturadas, fue realizado por el "Jet Propulsion Laboratory" de la NASA en California, como apoyo al programa Apollo y Survellor, estudiando las características de la superficie lunar. Se estableció el programa Ranger cuyo principal objetivo era tomar fotografías de la superficie lunar para su posterior análisis. Estas fotografías, digitalizadas, fueron procesadas mediante computadoras.

Por tanto, la visión artificial o también llamada Visión por Computadora, pretende capturar la información visual del entorno físico para extraer características relevantes visuales, utilizando procedimientos automáticos. Según [Marr82], "Visión es un proceso que produce a partir de imágenes del mundo exterior, una descripción útil para el observador y no tiene información irrelevante"

1.1. Justificación

Para asegurar la calidad en los procesos productivos, uno de los factores más importantes es el reconocimiento de los defectos generados durante la manufactura, por lo tanto es indispensable realizar inspecciones para descubrir los defectos de producción.

Su implementación en una empresa genera un aumento en el nivel de producción y una reducción en los costos de fabricación, elevando los niveles de competitividad en el mercado nacional e internacional.

Para la industria, la visión artificial incrementa la calidad y la seguridad de los procesos de fabricación. Esta técnica tiene muchas aplicaciones. Se utiliza especialmente en el control de calidad, la manipulación de materiales, la ordenación por calidades (grading), la prueba y la calibración de aparatos y la monitorización de procesos.

La visión por computador permite inspeccionar el proceso de producción sin fatigas ni distracciones, facilitando la cuantificación de las variables de calidad traduciéndose en un mejoramiento continuo.

Entre algunas ventajas de la visión artificial se pueden ejemplificar las siguientes:

- Facilidad en mediciones geométricas. Las figuras del inciso A), B), y C) de la Figura 1.1 son el claro ejemplo de éste defecto.
- No crea contornos ilusorios. En el inciso D) de la Figura 1.1 vemos inconscientemente un triángulo en el centro de los círculos, y en la figura del inciso G) en la Figura 1.1 es difícil reconocer los círculos blancos, pues parecen cambiar de color cada que fijamos la atención en uno diferente.
- No influye el entorno. Los cilindros del inciso E) en la Figura 1.1 son del mismo tamaño, pero el techo y demás proyecciones aplicadas en el entorno, engañan a nuestra percepción.
- Dificultad para medir magnitudes de intensidad. Por ejemplo el inciso F) de la Figura 1.1 ilustra éste punto.

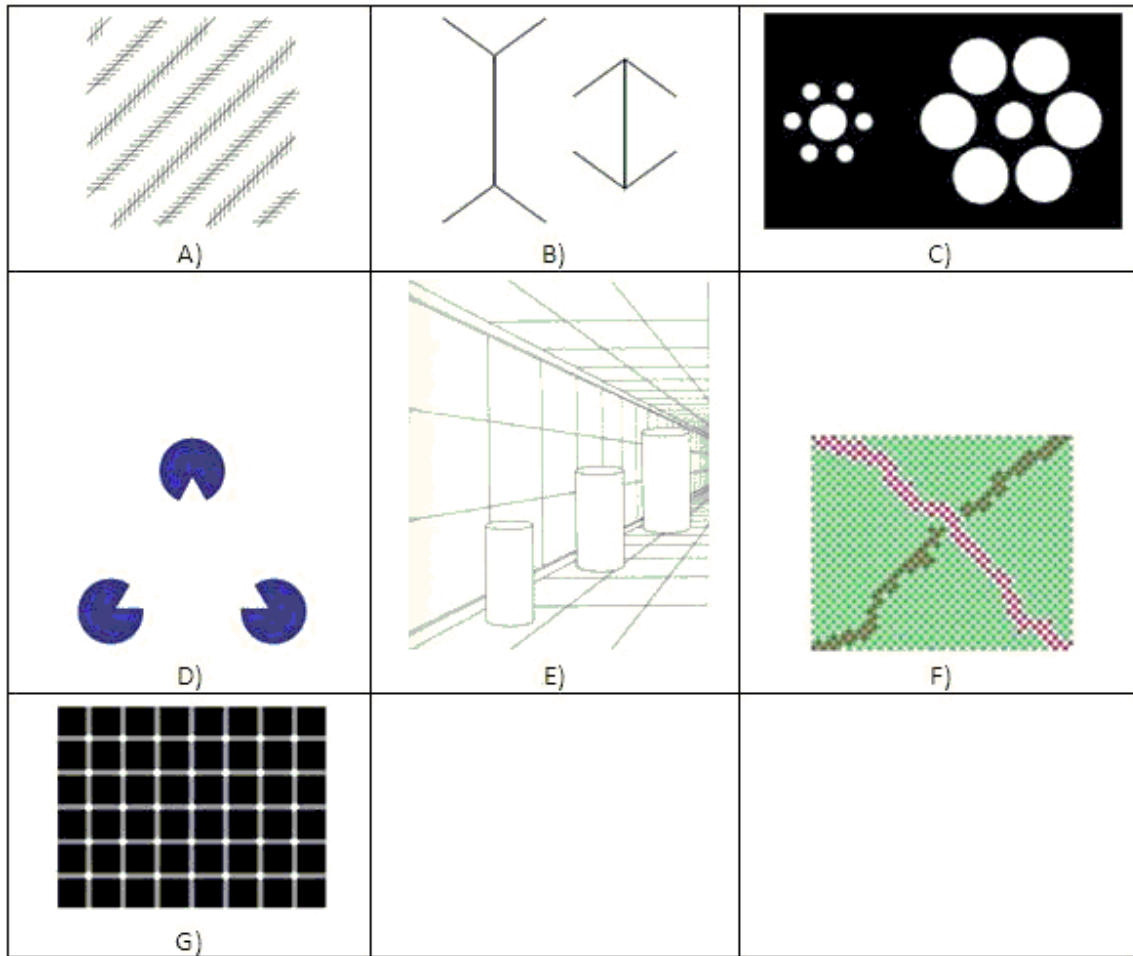


Figura 1.1: Ilusiones ópticas

1.2. Objetivos

Los objetivos del presente trabajo son los siguientes:

- Obtener una fotografía digital del objeto a medir
- Mejorar la calidad de la imagen adquirida, con el fin de realizar una adecuada segmentación del objeto de interés
- Identificar los errores de alto y ancho de diente, diámetros interiores y exterior, en las piezas metálicas de acero, aplicando algoritmos de análisis de la imagen a través de una pieza patrón
- Realizar una comparación usando una cámara de conexión ethernet y una digital tipo webcam para validar la eficacia de la cámara profesional

1.3. Planteamiento del problema y descripción del sistema

El proyecto consiste en realizar mediciones dimensionales para piezas metálicas como la que se ilustra en la Figura y 1.2, sin la participación de un individuo de manera directa, evitando los errores accidentales de la medición, los cuales son causados por fallas del observador. Este tipo de fallas son difíciles de eliminar pues los errores accidentales son producidos por causas fortuitas, y se presentan por ejemplo debido a la coincidencia entre índice y escala, a descuidos por parte del observador, etc.

Como solución a éste problema se propone utilizar un sistema de visión artificial para medir las piezas producidas.

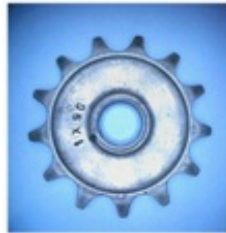


Figura 1.2: Vista superior de la pieza

Un sistema de visión artificial se conforma de los siguientes elementos:

- Fuente de luz: Es un aspecto clave ya que debe proporcionar condiciones de luz uniformes e independientes del entorno, facilitando rasgos de interés para una determinada aplicación.
- Sensor de imagen: Se encarga de recoger las características del objeto a estudiar.
- Tarjeta de captura o adquisición de imágenes: es la interfaz entre el sensor y la computadora o módulo de proceso que permite al mismo disponer de la información capturada por el sensor de imagen.

- Algoritmos de análisis de imagen: Es la parte inteligente del sistema. Su misión consiste en aplicar las transformaciones necesarias para extraer información de interés en el proceso.
- Computadora o módulo de proceso: Es el sistema que analiza las imágenes recibidas por el sensor, para extraer la información de interés en cada uno de los casos, implementando y ejecutando los algoritmos diseñados para la obtención de los objetivos
- Sistema de respuesta: Con la información obtenida, los sistemas de visión artificial pueden ayudar a tomar decisiones que afecten al sistema productivo con el fin de mejorar la calidad global de producción.

Un diagrama que ejemplifica los elementos previamente mencionados se muestra en la Figura 1.3

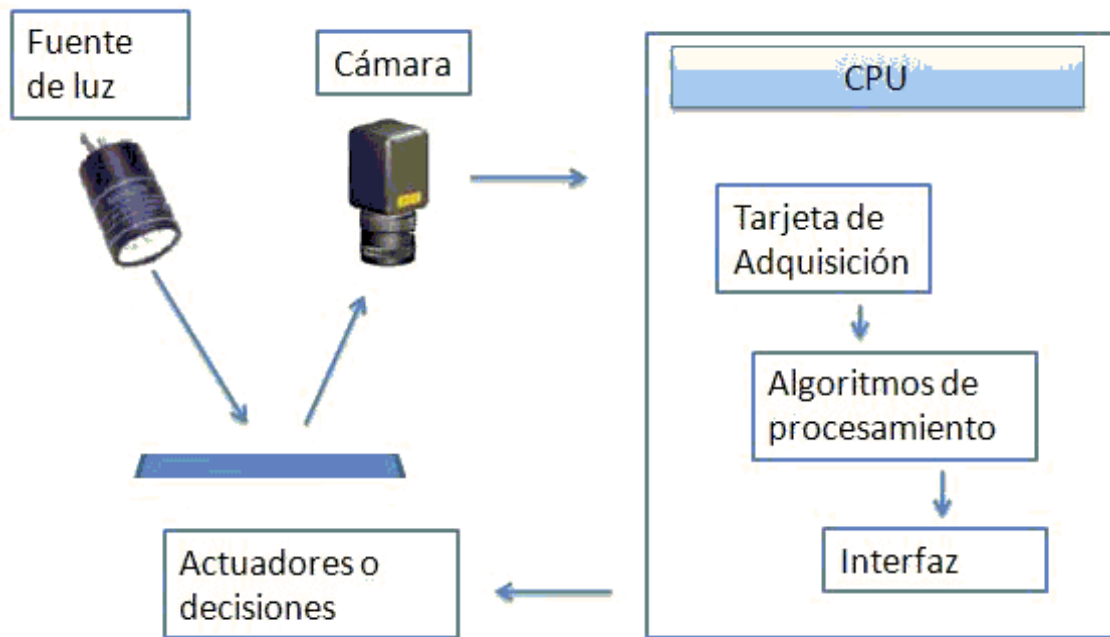


Figura 1.3: Elementos de un sistema de visión artificial

Capítulo 2

Estado del arte

A diferencia del estudio de los mecanismos de la visión humana, el procesamiento y análisis de imágenes digitales nace en el momento en que se dispone de recursos tecnológicos para captar y manipular grandes cantidades de información en forma de matrices. Esta distinción sitúa al procesamiento y análisis de imágenes digitales como una tecnología asociada a las ciencias de la computación y por tanto cabe pensar de ella como una proyección del término visión artificial dentro del ámbito de la Inteligencia Artificial.

Históricamente la primera vez que se hizo uso de las técnicas de imágenes digitales fue entre 1920-1930 en la transmisión de imágenes de fotografías periodísticas a través de cable submarino entre Londres y Nueva York. Esto supuso que el tiempo entre emisión y recepción de las imágenes pasase de una semana (por barco) a tres horas (por cable). A partir de esto y durante algún tiempo, se desarrollaron las técnicas de codificación y reproducción de las imágenes logrando pasar en la representación de imágenes de cinco a quince tonos de gris.

Pero estos hechos pueden considerarse aislados y no es hasta el siglo XX, durante los años 50-60 en que aparecen los primeros computadores digitales, y la necesidad de disponer de técnicas para la transmisión, y procesamiento de imágenes desde satélite, cuando estas técnicas empiezan a ser desarrolladas de forma sistemática.

Durante el auge de la carrera espacial, se dio un impulso considerable a las técnicas de transmisión, realce y restauración de imágenes por el Jet Propulsion Laboratory (EEUU), encargado del procesamiento de las imágenes mandadas por los primeros satélites. Como

consecuencia de los resultados obtenidos sobre las imágenes de satélite distintos organismos realizaron durante finales de los años sesenta y principios de los setenta grandes proyectos de investigación para el desarrollo de técnicas relativas tanto al estudio de los mecanismos de la visión como al tratamiento y análisis de imágenes digitales. Por su parte el National Institute of Health (EEUU) dedicó considerable esfuerzo al estudio de imágenes de rayos X, microscopía óptica y electrónica.

El MIT junto con la Universidad de Stanford trabajó durante los setenta en temas de visión aplicada a robótica bajo el proyecto Hand-Eye. Otros proyectos que también han influido en el desarrollo de estas técnicas son el PIPS (Pattern-Information Processing System) Japonés y el Norteamericano IUS (Image Understanding System). Este último proyecto marca el comienzo del uso de estas técnicas en aplicaciones de tipo militar lo que ha supuesto, desde entonces, una fuente importante de inversiones para su desarrollo.

Dentro del contexto europeo tan solo cabe destacar la investigación desarrollada, durante los años setenta, por los profesores G. Matheron y J. Serra de la École National Supérieure des Mines de Paris (Francia) con base en resultados de la geometría integral y el cálculo de probabilidades para la caracterización y medición de formas. Esta metodología se denomina Morfología Matemática y ha tenido sus principales áreas de aplicación en los campos de la geología, la biología y la medicina. Referencias obligadas del nivel alcanzado a comienzos de los ochenta por las técnicas de procesamiento y análisis de imágenes digitales son las monografías descritas en las siguientes citas [Castleman79], [Duda73], [Fu82], [Gonzales79], [Herman82], [Marr82], [Pratt78], [Rosenfeld82] y [Serra82].

Es a partir de este momento cuando se puede observar el comienzo de cierta especialización y catalogación de las técnicas existentes. Así, aparecen los conceptos de técnicas para el procesamiento de imágenes digitales como el conjunto de todas aquellas técnicas asociadas a la captura, codificación y representación de las imágenes que no introducen sobre las mismas ningún tipo de interpretación, y técnicas para el análisis de imágenes digitales, técnicas de visión por computador o visión mediante robot como acepciones que se refieren a aquellas técnicas que tratan de extraer la información presente en la imagen con el fin último de hacer una interpretación de la escena representada por dicha imagen.

Durante los años ochenta las técnicas de análisis de imágenes se desarrollan de forma vertiginosa como consecuencia de la gran cantidad de aplicaciones que aparecen y la madurez

alcanzada en el diseño de arquitecturas de computadores. Los desarrollos más teóricos han seguido en gran medida las pautas marcadas por Marr en [Marr82], habiendo sido la línea marcada por el MIT la que más influencia ha tenido. Las mayores contribuciones se han centrado en el desarrollo de algoritmos para la detección de características (bordes, líneas, texturas) que ayudan a definir lo que Marr llamo el esbozo primitivo, así como en el desarrollo de técnicas globales de segmentación de una imagen en regiones. A este último respecto cabe destacar aquellas aproximaciones que introdujeron la información de contexto en los procesos de clasificación y segmentación [Besag86], [Cross83],[Rosenfeld82].

De especial relevancia han sido también los numerosos trabajos que han usado técnicas de representación del conocimiento para los problemas de interpretación de imágenes, en relación con aplicaciones de ambiente industrial, iniciando una fuerte hibridación entre las técnicas de la Inteligencia Artificial para la representación del conocimiento y las técnicas de interpretación de escenas a partir de imágenes digitales.

A partir de la segunda mitad de la década de los 80 se inicia un creciente interés hacia el desarrollo de nuevas teorías y algoritmos para la interpretación de la imagen 2D como proyecciones de escenas 3D. En particular cabe mencionar los problemas asociados a la reconstrucción de escenas a partir de distintas proyecciones, el estudio del flujo óptico y sus múltiples aplicaciones a la caracterización de superficies 3D, la caracterización de superficies a partir del estudio del movimiento, el estudio de las formas a partir de las sombras, las técnicas de representación y búsqueda de objetos, el estudio de la orientación a partir de texturas.

Este interés por el estudio de problemas de mayor complejidad ha ido paralelo al desarrollo de nuevas generaciones de ordenadores de precio accesible, que permiten la experimentación con imágenes cada vez más complejas en su interpretación.

De forma paralela, en el tiempo, al desarrollo de las técnicas y métodos matemáticos hasta ahora expuestas, se han venido desarrollando diferentes arquitecturas de computadores específicas para el procesamiento de datos de imágenes digitales.

Capítulo 3

Fundamentos

Los sistemas de visión artificial trabajan, normalmente, con iluminaciones estructuradas. El espacio captado por las cámaras está aislado del resto. No hay contaminación lumínica, no hay sombras inesperadas, no existen objetos que interseccionen a los escenarios que se desean capturar. La iluminación se diseña para que sea capaz de resaltar los objetos de interés. La iluminación depende básicamente de la naturaleza de las superficies donde se va a reflejar y del tipo de fuente luminosa.

3.1. Técnicas de iluminación

Existen muchas técnicas de iluminación y resulta difícil de generalizar sobre los distintos tipos de iluminación, ya que suelen depender de cada caso concreto, sin embargo algunos tipos comunes de iluminación pueden ser:

- **Direccional.** Consiste en dar una iluminación orientada al objeto empleando un haz altamente direccional. Se aplica en la localización y reconocimiento de piezas, inspección de superficies, seguimiento de puntos para soldadura o pegamento, etc. en la Figura 3.1 inciso A) ilustra éste tipo de iluminación.
- **Difusa.** Tiene como propósito que los haces luminosos incidan sobre el objeto como en la Figura 3.1 inciso B), desde todas las direcciones y no sólo desde la fuente luminosa.

Proporciona el mínimo contraste del objeto y elimina las sombras. Se utiliza cuando los objetos a analizar presentan superficies suaves.

- A contraluz. Consiste en iluminar el objeto por detrás de forma que la fuente luminosa, el objeto y la cámara formen una línea recta como en la Figura 3.1 inciso C). Se emplea preferentemente para la localización de piezas y análisis dimensional. El principal inconveniente es la pérdida de los detalles de los cuerpos.
- Estructurada. Se sirve de la proyección de puntos, franjas o rejillas sobre la superficie de trabajo, la técnica de iluminación estructurada se muestra en la Figura 3.1 inciso D). En función de cómo se deforme este patrón de luz sobre la superficie se puede detectar las singularidades de la pieza objeto de análisis.

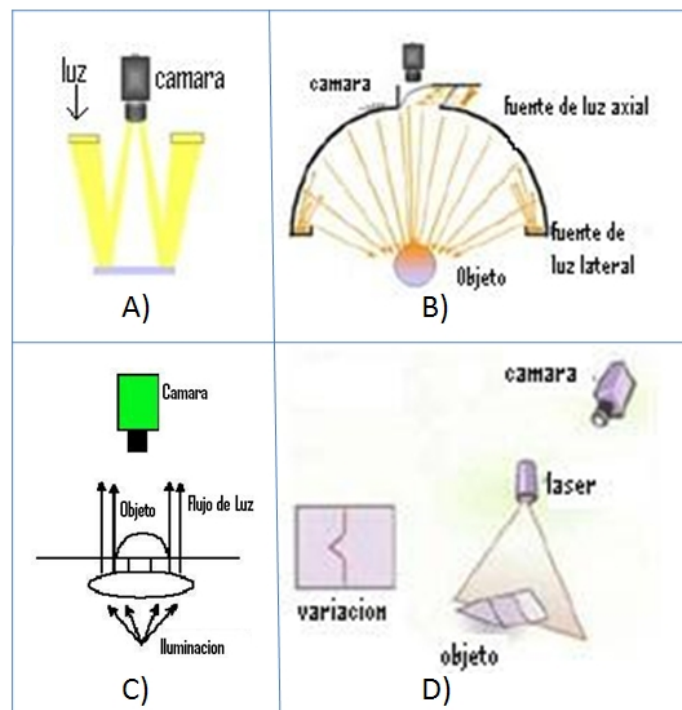


Figura 3.1: Tipos de iluminación

Entre los elementos comerciales que se pueden utilizar para una correcta iluminación de un sistema de visión se pueden mencionar:

- Luces incandescentes

- Focos halógenos
- Focos fluorescentes
- Láseres
- Leds

Las luces incandescentes tienen costo bajo, son fuentes no direccionales, tienen la posibilidad de ajustar la intensidad luminosa. Su desventaja es la cantidad de calor que desprenden.

El caso de los focos halógenos presenta la ventaja de poder funcionar con corriente continua, pues en ocasiones el trabajar con corriente alterna implica variaciones en la alimentación, lo cual deriva en variación de intensidad luminosa, también trabajan a menor temperatura.

El láser es empleado en iluminación estructurada como fuente monocromática que puede presentarse en diferentes patrones de luz, por ejemplo: línea, punto, rejillas, cruz, etc. Su desventaja es que requiere medidas especiales para evitar dañar a los operarios que trabajen cerca de éstos dispositivos.

3.2. Sensores de video

En 1970 se inventó el primer sensor de estado sólido. La imagen era formada por la conversión de los fotones a cargas eléctricas. Esta transformación se realiza por un elemento semiconductor llamado pixel que se disponía en forma matricial. Las cargas eléctricas eran leídas posteriormente por filas y por columnas, transformando la señal de video en información eléctrica. Se inició entonces una batalla por la dominación del mercado entre los CCD (Charge Coupled Deviced), y las cámaras de tubo de vacío. Ésta fue ganada por el CCD antes sus innumerables ventajas, como espacio, inmunidad a vibraciones, costo de fabricación, etc. Algo similar se libra hoy en día entre las clásicas cámaras fotográficas analógicas y las digitales con CCD o CMOS

3.2.1. Diferencia entre sensores CCD y CMOS

Las cámaras CCD presentan una mejor calidad de las imágenes y flexibilidad a expensas de un mayor tamaño. Sus aplicaciones son fotografía digital, aplicaciones industriales de altas prestaciones, aplicaciones científicas y médicas. Los sensores CCD, mostrados en la Figura 3.2 llevan utilizándose en las cámaras desde hace más de 20 años y presentan muchas ventajas de calidad, entre las cuales cabe destacar una mejor sensibilidad a la luz que los sensores CMOS. Esta mayor sensibilidad a la luz se traduce en mejores imágenes en situaciones de luz escasa. Sin embargo, los sensores CCD son caros ya que están fabricados siguiendo un proceso no estandarizado y más complejo para ser incorporados a una cámara. Además, cuando existe un objeto muy luminoso en la escena (como, por ejemplo, una lámpara o la luz solar directa), el CCD puede tener pérdidas, provocando rayas verticales por encima y por debajo del objeto. Este fenómeno se llama "smear" (mancha).



Figura 3.2: Sensor CCD

Los recientes avances en los sensores CMOS como se muestran en 3.3, los acercan a sus homólogos CCD en términos de calidad de la imagen, pero los sensores CMOS siguen siendo inadecuados para cámaras donde se exige la máxima calidad de imagen posible. Los sensores CMOS proporcionan soluciones de cámaras más económicas ya que contienen todas las funciones lógicas necesarias para fabricar cámaras a su alrededor. Hacen posible la producción de cámaras de un tamaño menor. Los sensores de tamaño mayor ofrecen una resolución megapixel para una variedad de cámaras de red. Una de las limitaciones actuales de los sensores CMOS es su menor sensibilidad a la luz. En condiciones de luz normales esto no supone ningún problema, mientras que en situaciones de escasa luz se vuelve manifiesto. El resultado es una imagen muy oscura o una imagen con apariencia granular.

Las cámaras CMOS tienen una integración mayor, menor disipación y tamaño más re-

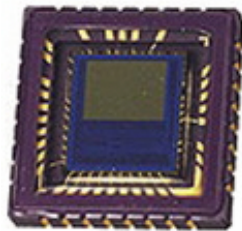


Figura 3.3: Sensor CMOS

ducido, a expensas de una menor calidad de imagen y menor flexibilidad. Sus principales aplicaciones son webcam, código de barras, scanners, juguetes, teléfonos móviles, cámaras de seguridad, entre otras.

Actualmente, el nivel que ofrecen las cámaras con CCD es mejor que el de las equipadas con CMOS. Sin embargo, la calidad de estas últimas aumenta progresivamente. Las cámaras sencillas, incluso las Webcams, están ofreciendo prestaciones sorprendentes para su precio, y esa tendencia va en aumento. El CMOS es una tecnología que tiene las mejores perspectivas para el futuro

3.2.2. Cámaras de estado sólido

La mayoría de las cámaras de estado sólido están basadas en dispositivos de cargas acopladas o CCD, si bien existen otras tecnologías menos comunes y más caras, como son la de inyección de carga o CID (Charge Injection Deviced) o las basadas en integración o TDI (Time Delay Integration). Aun así, el funcionamiento básico en las distintas tecnologías son similares y están basadas en el efecto fotoeléctrico. Los fotones de la luz pasan por una estructura cristalina de silicio y son absorbidos creando pares electrón-hueco. La cantidad de pares electrón-hueco será proporcional a la intensidad de luz que ha incidido sobre el elemento sensor.

Las cámaras se pueden clasificar según la disposición de los pixeles en:

- Unidimensionales. Son cámaras lineales, asociadas a un vector de sensores
- Bidimensionales Cámaras matriciales, asociadas a una matriz de sensores

Las cámaras lineales utilizan sensores lineales que acostumbran a tener entre los 512 y 8192 elementos (píxeles), con una longitud lo más corta posible, y con una gran calidad con el fin de obtener la mejor sensibilidad y prestaciones.

En una cámara lineal convencional, el máximo tiempo de exposición está limitado a la velocidad de la escena, y hace prácticamente imposibles las aplicaciones de alta velocidad y baja iluminación.

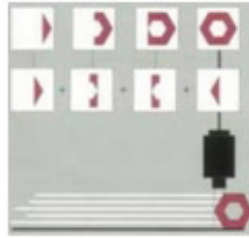


Figura 3.4: Funcionamiento de la cámara lineal

La tecnología TDI utiliza el movimiento sincronizado para capturar múltiples tomas de la misma línea y sumarlas, obteniendo una línea con sensibilidad amplificada. TDI es el nombre dado al método de combinar muchas líneas individuales a la vez, con el fin de incrementar la sensibilidad de la cámara. La ventaja de este método reside en que mejora la relación señal - ruido.

Para entender el modo de operar de una cámara TDI, se sigue la Figura 3.5. En el tiempo T1, la imagen se sitúa en el primer detector y crea una carga. Llegado el tiempo T2, la imagen se ha desplazado al segundo detector. Simultáneamente, el reloj del pixel ha movido la carga del primero al segundo.

En este punto, se crea una imagen formada por la carga anterior añadiéndose la nueva creada en el segundo detector, y así sucesivamente. La carga (señal) se incrementa linealmente con el número de detectores, también lo hace el ruido pero no proporcionalmente sino con la raíz cuadrada del número de elementos TDI, N_{TDI} . El resultado es una mejora de la relación señal-ruido en $\sqrt[2]{N_{TDI}}$.

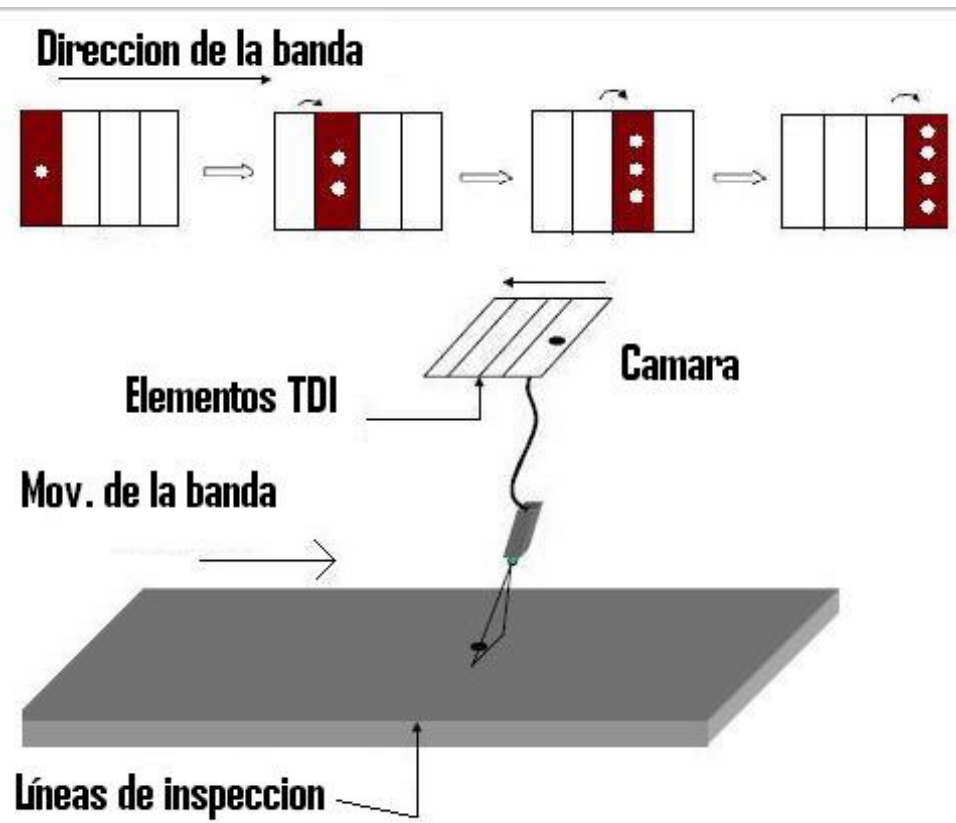


Figura 3.5: Funcionamiento de la cámara TDI

3.2.3. Cámaras matriciales

Los sensores de cámaras matriciales modernas son mayoritariamente de tipo CCD ó CMOS. Miles de diodos sensibles se posicionan de forma muy precisa en una matriz, y los registros de desplazamiento transfieren la carga de cada pixel para formar la señal de video. Los tamaños de las cámaras matriciales están definidos en pulgadas. Los formatos más comunes actualmente son de $\frac{1}{3}$ ", $\frac{1}{2}$ ", y $\frac{2}{3}$ ".

El problema que se presenta cuando se dispone de un conjunto de elementos fotosensores para captar una imagen, es la transmisión de las señales recogidas. Dependiendo de la tecnología de la cámara, existen diferentes estructuras para la adquisición y transmisión de la información.

Cualquier cámara, en principio, requiere un área fotosensora que sea capaz de captar toda una imagen enfocada sobre ella, mediante un sistema óptico. Esta matriz de elementos fotosensores es la encargada de recoger la información, que presenta la escena que se desee captar. Tras un cierto tiempo (denominado tiempo de integración), la información luminosa de la escena se encuentra recogida en esta matriz de elementos. Para extraer esta información de los elementos, se utiliza la técnica de registro de desplazamiento analógico. De esta forma, la carga generada en cada elemento sensor, se desplaza hasta la salida del dispositivo en su conjunto.

Existen distintas configuraciones posibles que darán lugar a diferentes posibilidades, tanto en la extracción como en la transmisión de la información adquirida.

Las transferencias más empleadas son:

- Transferencia de cuadro (Full frame). Son los CCD que emplean un registro paralelo simple para exposición de los fotones, integración de la carga y transporte de la carga. Se utiliza un obturador mecánico para controlar la exposición
- Transferencia interlínea. Son los más comunes y utilizan registros de desplazamiento que se encuentran entre las líneas de los pixeles y que se encargan de almacenar y transferir los datos de la imagen. La principal ventaja de este tipo de CCD es la alta velocidad de obturación, pero carecen de alta sensibilidad y precisión.

- Transferencia de cuadro interlínea. Esta estructura ofrece las mejores prestaciones de los CCDs actuales. Sin embargo, su estructura es compleja y requieren un área total mayor al tener la zona de almacenamiento separada. En concreto el área activa es alrededor del 22 % del total de la zona sensora, lo cual ofrece una idea de la relación entre la zona activa y la zona sensora presentada en el proceso de captación de imágenes.

3.3. Tarjetas digitalizadoras

Una tarjeta digitalizadora o frame grabber tiene como propósito principal muestrear, en tiempo real, la salida estándar de una cámara de video, pasando la información al computadora. La información es almacenada temporalmente y una vez allí, la información es procesada siguiendo las directrices de un programa. Dependiendo de la aplicación, la tarjeta puede ser programada para capturar y almacenar cuadros individuales, recoger una cierta secuencia de cuadros, o quizás leer de manera continua la imagen de video y realizar una tarea determinada en tiempo real. Algunas tarjetas tienen características avanzadas como puede ser captura síncrona de las imágenes.

Las digitalizadoras con capacidades para el procesamiento en tiempo real derivan sus prestaciones de los procesadores digitales de señal (DSP). Éstos pueden realizar procesamiento básicos sobre la señal de una manera encadenada y en tiempo real.



Figura 3.6: Tarjeta de captura de video

3.4. Definiciones de interés

A continuación se describen algunos conceptos de interés

Pixel. Elemento básico de una imagen (picture element).

Imagen.- Arreglo bidimensional de pixeles con diferente intensidad luminosa.

Si la intensidad luminosa de cada pixel se representa por n bits, entonces existirán 2^n escalas de gris diferentes.

Matemáticamente, una imagen se representa por $r = f(x, y)$, donde r es una función que define la intensidad luminosa del pixel cuyas coordenadas son (x, y) . Matemáticamente, un sistema para procesar imágenes se representa como $g(x, y) = T[f(x, y)]$, donde g representa la imagen procesada y T representa la transformación que será aplicada a la imagen

Espacio RGB.- se basa en la combinación de tres señales de luminancia cromática distinta: rojo, verde, azul (Red, Green, Blue). La forma más sencilla de obtener un color específico es determinar la cantidad de color rojo, verde y azul que se requiere combinar para obtener el color deseado, ver la Figura 3.7; para lo cual se realiza la suma aritmética de las componentes: $X = R + G + B$, gráficamente representada por un cubo.

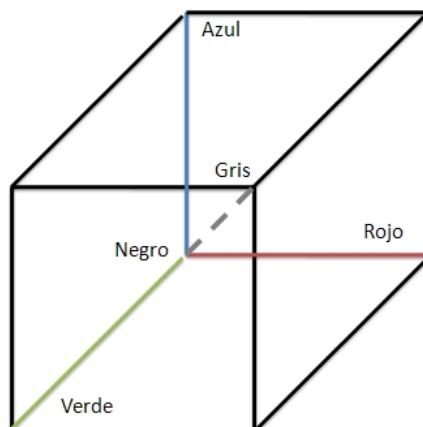


Figura 3.7: Espacio de colores RGB

En la recta que une el origen con el valor máximo se encuentran ubicados los grises (escala

0	1	0
1	1	1
0	1	0

Tabla 3.1: Elementos de conectividad tipo 4

de gris) debido a que sus tres componentes son iguales.

Histograma de una imagen. El histograma de una imagen es una representación del número de pixeles de cierto nivel de gris en función de los niveles de gris.

Conectividad. La conectividad es un concepto importante utilizado para establecer los límites de objetos en regiones dentro de una imagen. Para determinar si dos pixeles están conectados, se determina si son adyacentes en algún sentido ($N_8(p)$, $N_4(p)$ por ejemplo), y si sus niveles de gris satisfacen un criterio de similaridad (por ejemplo si son iguales). Por ejemplo, en una imagen binaria con valores de 1 y 0, dos pixeles pueden ser vecinos $N_4(p)$, pero se dice que están conectados solo cuando tienen el mismo valor, la Tabla 3.1, muestra un ejemplo de conectividad 4.

Elemento estructural. Son matrices binarias, las cuales generan figuras geométricas para realizar operaciones morfológicas en imágenes binarias principalmente, un ejemplo es la Figura 3.8 donde se muestra un círculo de radio $r = 3$.

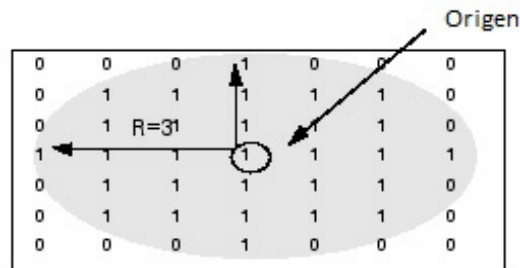


Figura 3.8: Elemento estructural circular de radio 3

Operación Morfológica. Se dedica a la identificación y descomposición de objetos, la extracción de rasgos, la localización de defectos, etc. y está sumamente relacionadas con las formas geométricas, algunos ejemplos son la dilatación, erosión, complemento, etc.

3.5. Modelo de la cámara

Desde hace mucho tiempo es conocida la experiencia de la cámara oscura. Cardán, en 1550, empleó la cámara oscura con una lente. Se trata de realizar en un habitáculo cerrado, en una de sus paredes, un pequeño orificio. La luz, tras entrar en la cámara, se proyecta sobre la pared opuesta a la que tiene el orificio. Por cada rayo procedente del exterior se obtiene un punto luminoso en el plano proyectivo. Extendiendo el razonamiento a los infinitos rayos que atraviesan el agujero se obtiene una imagen invertida y proyectiva, del mundo exterior al plano sensor. la Figura 3.9 ilustra su funcionamiento



Figura 3.9: Funcionamiento de la cámara oscura

También se observa que a medida de que la separación entre las paredes de la cámara aumente, la imagen proyectiva aumenta de tamaño. A este modelo de cámara oscura se la denomina pin-hole y un ejemplo se muestra en la Figura 3.10.

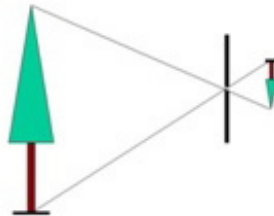


Figura 3.10: La separación aumenta o disminuye la escala

Sin embargo, en la cámara oscura, la energía incidente que atraviese no es lo suficientemente elevada como para excitar al plano sensor. Por lo que se resuelve aumentando el

tamaño del agujero. El efecto provocado es una degradación de la imagen, ya que son varios los rayos los que inciden, pero no necesariamente del mismo objeto, produciéndose una imagen borrosa. Estos inconvenientes son subsanados con el uso de lentes.

3.5.1. Lente simple

Con el objeto de aumentar la luminosidad en el interior de la cámara, sin que la imagen se haga borrosa al aumentar el tamaño del agujero, se utiliza una lente. Una lente es una sustancia transparente y refringente, limitada por dos caras, una de las cuales es curva y la otra puede ser plana o curva, pero con la condición de que sus centros de curvatura están en el mismo eje. El efecto deseado es la convergencia o divergencia del frente de ondas.

La Figura 3.11 representa una lente convexa de radio de curvatura R . El centro de curvatura de la lente se define por C . Sea un punto P exterior, situado a una distancia Z . Los infinitos rayos procedentes del emisor que chocan sobre la lente producirán una dirección especular, respecto a la normal de la superficie de la lente. La refracción del frente de ondas, gracias a la curvatura constante, se concentra en el punto P' , separado a una distancia z , respecto a la superficie. La relación entre estas distancias con el radio de curvatura viene definido por la ecuación de Gauss:

$$\frac{1}{Z} + \frac{1}{z} = \frac{2}{R} \quad (3.1)$$

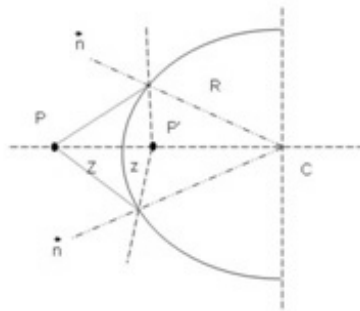


Figura 3.11: Lente convexa de radio de curvatura R

En la formación de imágenes se emplean tanto lentes convergentes o positivas como divergentes o negativas. Las lentes positivas tienen la propiedad de hacer converger en un

punto, los distintos rayos divergentes que proceden de un mismo lugar del objeto exterior. Las lentes divergentes hacen el trabajo al contrario, no focalizan la luz, sino que las hacen divergir.

Al emplear lentes convergentes, todos los puntos del plano imagen tienen, por tanto, la característica de que todos los rayos incidentes en cada uno de dichos puntos, provienen de un mismo punto del objeto, obteniéndose una imagen nítida con una mayor energía incidente, como en la Figura 3.12.

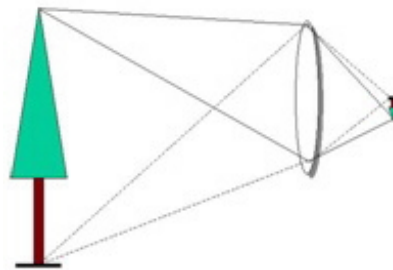


Figura 3.12: Modelo de lente convergente delgada

La zona sensible de la cámara deberá estar situada en el plano de convergencia de los rayos, en caso contrario, los rayos incidentes sobre el sensor provendrán de distintos objetos, obteniéndose una imagen borrosa o desenfocada.

Se observa que para un mismo objeto exterior, a medida de que éste se encuentre más alejado respecto a la cámara, el punto de convergencia de los rayos, al otro lado de la lente, disminuye, ya que el grado de convergencia de la lente permanece constante y la radiación incidente tienen menor divergencia. Al procedimiento de ajustar la distancia entre la lente y el plano sensor, con el objeto de tener una imagen nítida, se le denomina enfoque.

Dependiendo de la curvatura de las lentes, éstas poseen distintos grados de convergencia de los rayos provenientes del objeto, y por tanto será distinta la distancia entre el plano de la imagen y la lente. Con el fin de medir el grado de convergencia de una lente, se utiliza dicha distancia lente-plano de imagen. Pero debido a que esta distancia es variable según la distancia a la que está enfocado el objeto, su medición se realiza cuando el objeto se encuentra enfocado a una distancia infinito delante de la cámara; en cuyo caso, los rayos

procedentes del mismo punto del objeto serán paralelos. La distancia focal, f , se define como la distancia entre la lente y el plano sensor o focal cuando se encuentra enfocado hacia el infinito. Esta definición equivale a decir que la distancia focal es la distancia entre la lente y el punto de convergencia de todos los rayos paralelos que inciden perpendicularmente a ella. A este punto de convergencia se le denomina foco de la lente. Si el objeto se halla a una distancia finita, la separación entre la lente y el plano focal será mayor que la distancia focal.

3.5.2. Formatos gráficos.

Las imágenes digitales se pueden almacenar en distintos soportes físicos tales como: discos duros, CDs, DVDs o memoria flash. Para este objetivo se han inventado cientos de formatos con la función principal de comprimir la enorme cantidad de información que contiene una imagen digital.

Una primera clasificación podría ser:

- **Formatos gráficos vectoriales** Se almacenan las coordenadas de las figuras geométricas simples como segmentos, triángulos, rectángulos, circunferencias, etc., así como las órdenes necesarias para rellenar con los colores adecuados. Para dibujos simples el fichero resultante ocupará poco tamaño comparando con lo que ocuparía si estuviera en formato de mapa de bits. Tal es el caso de ficheros generados por programas de generación de imágenes 3D como AutoCAD, librerías de computación gráfica como VTK o programas de dibujos en general
- **Formatos gráficos en mapa de bits.** Este formato es el más intuitivo, ya que lo único que hace es almacenar secuencialmente los píxeles que componen la imagen en un fichero. El orden de almacenamiento se puede entender considerando la imagen como una gran matriz de píxeles. Así partiendo de la esquina superior izquierda, avanzando por las columnas de izquierda a derecha y por filas de arriba abajo. Este último puede dividirse además en formatos con pérdida o no de la información.
- Si el fichero se almacena sin previo procesamiento, se dice que está en formato crudo (*.raw), será necesario para su lectura saber cuántas filas y columnas tiene la imagen.

El inconveniente del formato en crudo es la gran cantidad de espacio que se necesita.

La mayoría de los formatos actuales llevan consigo una reducción sustancial del volumen informativo. La compresión de las imágenes se hace siguiendo algunas de estas tres estrategias:

- Compresión en la codificación: en vez de emplear un número fijo de bits para cada nivel de gris o color, se emplea un código más corto a los niveles más probables y más largo a los menos probables.
- Compresión en la representación: La similitud de intensidad en los vecinos hace emplear las rachas, las cuales indican el número de vecinos que se repiten. Se almacenan en vez de los valores de intensidad, el número de vecinos con la misma intensidad.
- Compresión por redundancia visual: Las imágenes vistas por el ojo humano no es capaz de percibir el contraste de tonos muy próximos entre sí, por dicha razón en vez de almacenar los mapas de bits se aplican transformaciones matemáticas sobre la imagen, con el objeto de guardar los principales coeficientes de la conversión. Evidentemente, la imagen resultante es de peor calidad que la entrante. Hay pérdidas de detalle. El ejemplo más característico es el algoritmo de compresión JPEG basadas en las transformadas discretas del coseno. También son ejemplos de compresión basada en redundancia visual los estándares de video MPEG y MPEG2.

3.6. Imagen binaria

Una vez preparada la escena para ser capturada por las cámaras de visión artificial como se ve en la Figura 3.13, se deberá de tratar el aspecto tanto de cuantificación como de codificación que sufre la señal de video al ser introducida en la computadora. Depositada la información, como una matriz o conjunto de matrices de valores discretos se procederá, tal cual se vio en las etapas de visión del capítulo 1, a un procesamiento de la imagen. Esta etapa, de bajo nivel, trata de obtener una nueva imagen que o bien mejore su calidad o bien destaque algún atributo primario de los objetos capturados. En el primer aspecto del procesamiento de bajo nivel, se tratará de corregir las posibles faltas de iluminación

uniforme, la eliminación del ruido o de aumentar el contraste en la imagen. Mientras en el segundo, el objetivo es realzar los bordes de los objetos, regularizar sus colores, destacar sus texturas, etc.

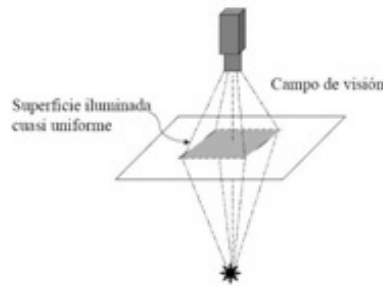


Figura 3.13: Escenario de captura de la escena

Las imágenes digitales son obtenidas por el proceso de muestreo y cuantificación de las señales analógicas adquirida a través de los sensores de visión. El muestreo consiste en la medición a intervalos de tiempo de la señal de video analógica. La imagen es convertida en una matriz discreta de $M \times N$ píxeles. El muestreo está relacionado con la resolución espacial de la imagen. Ésta se define como el espacio muestreado sobre un objeto, a una distancia l respecto a la cámara, entre la distancia entre dos píxeles adyacentes.

Para determinar la resolución requerida, hay que determinar tres factores:

1. Determinar la característica más pequeña a observar
2. Determinar el campo de visión deseado a adquirir.
3. Determinar el factor de magnificación y el número de píxeles que debe de tener la cámara.

Como regla general, se puede emplear que la característica más pequeña sea representada por, al menos, un cuadrado de 2×2 píxeles, la Figura 3.14 muestra un ejemplo de variación en la resolución de la misma imagen.

Por otro lado, el campo de visión suele venir dado por la capacidad del sistema de iluminación que por lo general, se pretenderá conseguir una intensidad luminosa uniforme



Figura 3.14: La misma escena capturada con diferentes resoluciones

en toda la superficie a inspeccionar por la cámara. Por tanto, se plantea el problema en los siguientes términos: defecto más pequeño a visualizar y el campo de visión a capturar por la cámara.

En el caso de requerir una resolución elevada, se puede optar por dos soluciones:

1. Si es posible parar la escena, puede resultar ser más ventajoso utilizar varias cámaras de baja resolución en lugar de una de alta resolución, pues se consigue disminuir anchos de banda puntuales y resulta ser una solución más económica.
2. Si la escena no se puede detener y el número de detectores a utilizar es elevado, porque la arquitectura de iluminación sólo permite iluminar uniformemente una estrecha banda, o bien porque se quiere cubrir el ancho de la banda del material a inspeccionar, las cámaras lineales y TDI son las más apropiadas. Estas cámaras tienen la ventaja de poner decenas de miles de píxeles en línea, cubriendo toda la cubierta de la superficie y con una resolución muy alta. La adquisición de una sola línea permite un ancho de banda razonable, aun con decenas de miles de detectores. Por otro lado, las altas velocidades hacen que los tiempos de exposición sean muy pequeños, obligando a aumentar la intensidad luminosa. Esta consecuencia, puede ser disminuida con cámaras TDI, siendo éstas las más utilizadas en inspección de superficies continuas en línea.

3.6.1. Distancias entre pixeles

Antes de abordar el tema de procesado, se definen ciertas relaciones que se establecen entre los pixeles de una imagen y que serán aplicadas en las técnicas de procesamiento de las imágenes. En primer lugar, se mencionarán los diferentes conceptos de distancias.

La distancia más usual entre dos pixeles es la distancia geométrica o distancia euclídea. La distancia euclídea entre el pixel p de coordenadas (x, y) y el pixel q de posición (s, t) será como en la ecuación 3.2:

$$D_e(p, q) = \sqrt{(x - s)^2 + (y - t)^2} \quad (3.2)$$

Que se puede escribir de forma matricial como 3.3:

$$\begin{array}{ccccc} \sqrt{8} & \sqrt{5} & 2 & \sqrt{5} & \sqrt{8} \\ \sqrt{5} & \sqrt{2} & 1 & \sqrt{2} & \sqrt{5} \\ 2 & 1 & 0 & 1 & 2 \\ \sqrt{5} & \sqrt{2} & 1 & \sqrt{2} & \sqrt{5} \\ \sqrt{8} & \sqrt{5} & 2 & \sqrt{5} & \sqrt{8} \end{array} \quad (3.3)$$

El problema de esta métrica está en el tiempo de cálculo que requiere. Por dicha razón también se suele emplearse, en imágenes digitales, la distancia rectangular o Manhattan, definida como en la ecuación 3.4:

$$D_m(p, q) = (|x - s| + |y - t|) \quad (3.4)$$

Cuya forma matricial se muestra en: 3.7

$$\begin{array}{ccccc} 4 & 3 & 2 & 3 & 4 \\ 3 & 2 & 1 & 2 & 3 \\ 2 & 1 & 0 & 1 & 2 \\ 3 & 2 & 1 & 2 & 3 \\ 4 & 3 & 2 & 3 & 4 \end{array} \quad (3.5)$$

Y la distancia de Tchebychev, tomada según la ecuación 3.6:

$$D_t(p, q) = \max(|x - s|, |y - t|) \quad (3.6)$$

Que tiene por representación matricial la forma 3.7

$$\begin{bmatrix} 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{bmatrix} \quad (3.7)$$

Una imagen puede ser definida como una función de dos dimensiones, $f(x, y)$, donde x y y son coordenadas del plano espacial, y la amplitud f en cualquier par de coordenadas es llamada intensidad de imagen en ese punto.

Una representación para una función de imagen digitalizada podría ser como la función matricial 3.8:

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \cdots & f(0, N - 1) \\ f(1, 0) & f(1, 1) & \cdots & f(1, N - 1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M - 1, 0) & f(M - 1, 1) & \cdots & f(M - 1, N - 1) \end{bmatrix} \quad (3.8)$$

Las imágenes de color están formadas por una combinación de imágenes 2-D como lo muestra la Figura **3.15** ,. Las imágenes RGB consisten en la combinación de tres componentes individuales de cada componente de color (rojo, verde, azul), para a través de sus diversos valores de intensidad, cambiar la tonalidad de la imagen.

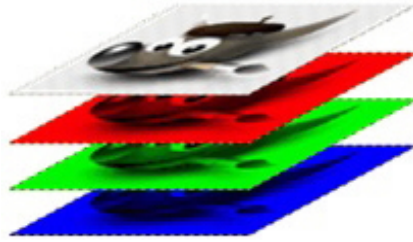


Figura 3.15: Descomposición de una imagen en sus componentes RGB

3.7. Procesamiento de la imagen

3.7.1. Filtrado o pre-procesamiento

Cuando se adquiere una imagen mediante algún sistema de captura, por lo general esta no es directamente utilizable por el sistema de visión. La aparición de variaciones en intensidad debidas al ruido, por deficiencias en la iluminación, o la obtención de imágenes de bajo contraste, hace necesario un preprocesamiento de la imagen con el objetivo fundamental de corregir estos problemas, además de aplicar aquellas transformaciones a la imagen que acentúen las características que se deseen extraer de las mismas, de manera que se facilite las operaciones de las etapas posteriores. A este campo pertenecen las técnicas de transformaciones geométricas, las basadas en el histograma, el filtrado espacial.

Aunque existen líneas de investigación abiertas en el filtrado de las imágenes, no es el campo más activo en ese sentido, y las técnicas existentes son bastante útiles para la mayoría de las aplicaciones.

Algunas de las técnicas más habituales son:

- Conversión de los niveles de gris: su función es la mejora de la calidad de la imagen.
- Transformaciones geométricas: los objetivos de ésta son la corrección de la perspectiva y la reconstrucción tridimensional de los objetos de la escena.
- Transformación del histograma: las transformaciones del histograma pueden facilitar la segmentación de objetos de la imagen, aunque habitualmente sólo sirve para modificar el contraste de la imagen y el rango dinámico de los niveles de gris.
- Filtrado espacial y frecuencial: dado que pueden suponer una importante modificación de la imagen original, generalmente sólo se consideran filtrados aquellas operaciones realizadas en el dominio espacial o frecuencial que suponen transformaciones pequeñas de la imagen original.

3.7.2. Segmentación o aislamiento de los objetos de interés

Cuando ya se dispone de la imagen capturada y filtrada, es necesario aislar o separar los objetos de interés de la escena. Se pretende por tanto dividir una imagen en diferentes regiones, o dicho de otra forma, detectar automáticamente los bordes entre los elementos o regiones. Las operaciones de segmentación de una escena dependen de la propia escena y de la información que se busque dentro de la imagen. Por ejemplo, en fotografía aérea se utiliza para separar regiones pobladas, agrícolas, bosques, aguas, etc. En imagen médica se usa para aislar zonas enfermas de los órganos, o para detectar y contar microorganismos en microscopía. Industrialmente, se usa como primer paso para la medición de piezas, para determinar la correcta disposición del etiquetado de los envases (aislar el envase, aislar la etiqueta, determinar si la relación entre estos dos objetos es correcta), para la medición de irregularidades en superficies homogéneas, para la detección de cambios de texturas en las que el sistema ha de determinar si es un defecto o no lo es, etc.

Las técnicas básicas de segmentación se pueden dividir en tres grupos:

- Aplicación de umbrales de niveles de gris
- Agrupación por rasgos comunes.
- Extracción de bordes.

La segmentación de imagen es una temática actual de investigación y se han definido numerosos métodos diseñados para aplicaciones particulares. No obstante, no hay métodos que sean útiles en una amplia variedad de situaciones, por lo que cada problema necesita la adaptación o desarrollo de nuevas técnicas.

3.7.3. Extracción y selección de características

Con los objetos a estudiar ya definidos, es necesario extraer una serie de medidas que los caractericen adecuadamente, en forma de vector de características.

Como norma general, para la elección de características se intenta que:

- Sean discriminantes, es decir, tengan valores numéricos diferentes para clases diferentes.
- Sean fiables, es decir, tengan valores parecidos dentro de una misma clase.
- Estén correlacionadas, es decir, que se pueda obtener la misma información con el mínimo número de características.
- Se calculen en un tiempo aceptable, de manera que sea utilizable en problemas de tiempo real.

Dado que las características relevantes son a menudo desconocidas a priori, habitualmente se introducen muchas características candidatas para intentar representar mejor las clases, aún cuando muchas de ellas puedan ser parcialmente o completamente redundantes o irrelevantes. No obstante, existen unas técnicas para mejorar las prestaciones del conjunto de características escogido inicialmente. Éstas están íntimamente ligadas al reconocimiento de formas en su etapa de diseño del clasificador.

3.7.4. Filtrado espacial

El filtrado espacial es una operación que sirve para resaltar detalles espaciales, para mejorar la interpretación visual o facilitar un procesamiento posterior. Algunos ejemplos incluyen aplicar filtros para mejorar los detalles de bordes en imágenes, o para reducir o eliminar patrones de ruido.

Un filtro espacial modifica el valor de cada pixel de acuerdo con los valores de los pixeles que lo rodean, esto es, asemejarlos a los pixeles cercanos. Su mecánica consiste en mover el centro de la máscara o filtro w en cada punto de la imagen como lo demuestra la Figura 3.16

Frecuencia Espacial

La frecuencia espacial define la magnitud de cambios en el nivel de gris por unidad de distancia en una determinada zona de la imagen. Las áreas de la imagen con transiciones graduales son de bajas frecuencias, las áreas de cambios rápidos se llaman áreas de altas frecuencias. De esta forma, los filtros espaciales se clasifican en:

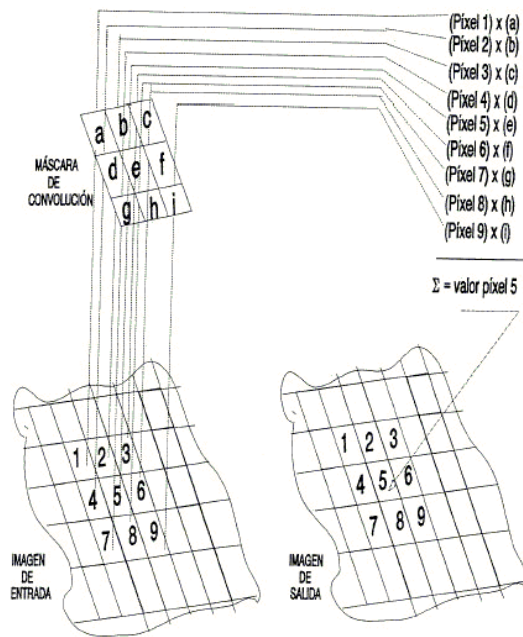


Figura 3.16: Funcionamiento de la máscara

- Filtros pasa bajos: Suavizan las imágenes y atenúan ruidos, su objetivo es homogeneizar el nivel de gris de cada pixel al nivel de gris de los pixeles vecinos. Ello produce un borronado de los bordes, perdiéndose nitidez visual de la imagen, aunque se gane homogeneidad
- Filtros pasa altos: Enfatizan las altas frecuencias, para mejorar las características lineales como los límites en general
- Filtros detectores de bordes: realizan las operaciones necesarias para enfatizar los bordes que rodean a un objeto en una imagen para analizarlo más fácilmente. Típicamente crean una imagen con fondo gris y líneas blancas y negras, rodeando los bordes de los objetos y características de la imagen

Existen dos conceptos relacionados entre sí que deben ser aclarados antes de realizar un filtrado espacial.

- Correlación
- Convolución

Convolución espacial

El filtrado espacial se realiza trasladando una matriz rectangular de dos dimensiones (también conocida como máscara o kernel), la cual contiene *pesos* o ponderaciones sobre la imagen en cada localización de pixel. Se procede a colocar la matriz sobre el pixel a evaluar y su valor es calculado de acuerdo con los pixeles de alrededor y sus valores de ponderación. Cuando se calcula un nuevo valor, se desplaza la ventana al siguiente pixel realizando la misma operación. Este proceso de evaluar la vecindad ponderada del pixel se denomina *convolución bidimensional*.

En general, la convolución de una imagen f de $M \times N$ con una máscara h de $m \times n$ está dada por la ecuación 3.9:

$$g(x, y) = \sum_{i=-a}^a \sum_{j=-b}^b f(x+i, y+j) \cdot h(i, j) \quad (3.9)$$
$$a = \frac{m-1}{2} \quad b = \frac{n-1}{2}$$

Por ejemplo, consideremos la máscara de convolución h y una imagen genérica representada por f .

$$h = \begin{bmatrix} h(-1, -1) & h(-1, 0) & h(-1, 1) \\ h(0, -1) & h(0, 0) & h(0, 1) \\ h(1, -1) & h(1, 0) & h(1, 1) \end{bmatrix} \quad (3.10)$$

Dado que $m = 3$ y $n = 3$, a y b son iguales a 1. Teniendo en cuenta todo lo anterior, la respuesta a la máscara de convolución dada, o sea, $g(x, y)$, está dada por la expresión anterior. Evaluando la respuesta para un punto cualquiera:

$$g(5, 4) = \sum_{i=-1}^1 \sum_{j=-1}^1 f(5+i, 4+j) \cdot h(i, j)$$

$$g(5, 4) = f(4, 3) \cdot h(-1, -1) + f(4, 4) \cdot h(-1, 0) + f(4, 5) \cdot h(-1, 1) + f(5, 3) \cdot h(0, -1) + f(5, 4) \cdot h(0, 0) + f(5, 5) \cdot h(0, 1) + f(6, 3) \cdot h(1, -1) + f(6, 4) \cdot h(1, 0) + f(6, 5) \cdot h(1, 1)$$

Correlación espacial

Se aplica de la misma forma que la convolución, sin embargo, la máscara de convolución h debe ser rotada 180° previamente para posteriormente realizar el mismo procedimiento que indica la ecuación 3.9

Por ejemplo si tenemos la imagen f y aplicamos una máscara h el resultado de su correlación será a

$$f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad h = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

$$a = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 & 0 \\ 0 & 4 & 5 & 6 & 0 \\ 0 & 7 & 8 & 9 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Para aplicar la correlación con la misma máscara h a la imagen a , tendrá por resultado la imagen b

$$h = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad \text{donde se usa la rotación de } h, h' = \begin{bmatrix} 3 & 2 & 1 \\ 6 & 5 & 4 \\ 9 & 8 & 7 \end{bmatrix},$$

$$b = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 2 & 1 & 0 \\ 0 & 6 & 5 & 4 & 0 \\ 0 & 9 & 8 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

De forma gráfica la diferencia entre operaciones se puede observar en la Figura 3.17

Filtros Pasabajos

El resultado de aplicar un filtro pasabajos a una imagen es simplemente el promedio de los pixeles contenidos en el vecindario de la máscara utilizada.

Generalmente se utilizan para atenuar los detalles irrelevantes en una imagen. Otra de las utilidades del filtro pasabajos, aparte de las más obvia que es la atenuación del ruido, es el

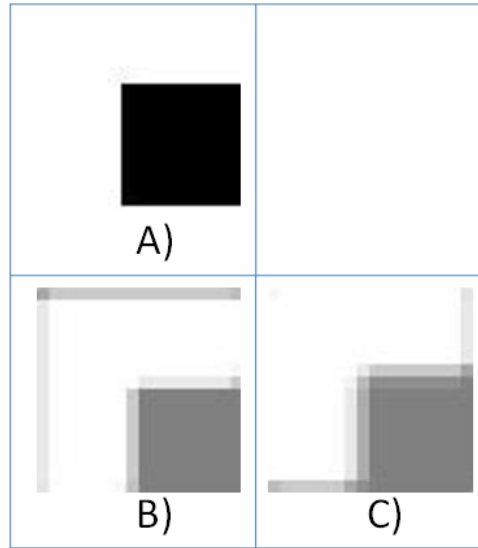


Figura 3.17: Ejemplo de la convolución y la correlación en una imagen

suavizado de los falsos contornos producidos por una cuantización con un número insuficiente de niveles de gris.

El procedimiento básico del filtro pasabajos es reemplazar el valor de cada pixel en una imagen por el promedio de los niveles de gris del vecindario definido por la máscara, un ejemplo se muestra en la Figura 3.18.

Algunos ejemplos de éstas máscaras o filtros pueden ser:

$$\frac{1}{9} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\frac{1}{16} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Las máscaras de los ejemplos anteriores se encuentran expresadas en números enteros pues resulta más eficiente computacionalmente calcular enteros que números en punto flotante, posteriormente de aplicar la máscara, la imagen completa se divide por la fracción previa a la máscara (la cual es la suma de todos los coeficientes de la máscara).

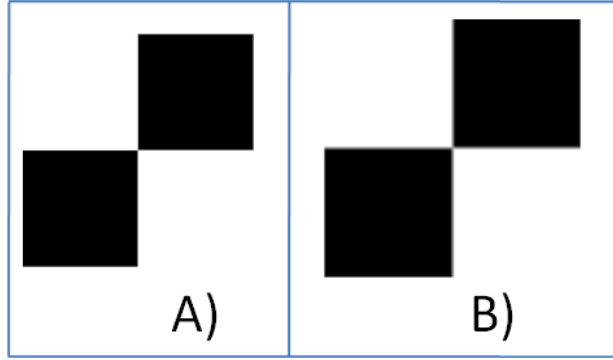


Figura 3.18: Ejemplo de aplicación de un filtro pasa bajos

El valor interno de los pixeles nos permite incrementar o restar la importancia de los pixeles vecinos de la imagen que queramos analizar.

La implementación general para filtrar una imagen de $m \times n$ con una máscara de promedio ponderado de $m \times n$ (m y n impares), es aplicando la fórmula 3.11:

$$g(x, y) = \frac{\sum_{i=-a}^a \sum_{j=-b}^b f(x+i, y+j) \cdot h(i, j)}{\sum_{i=-a}^a \sum_{j=-b}^b h(i, j)} \quad (3.11)$$

Filtros estadísticos: filtro de mediana

Los filtros estadísticos son filtros espaciales no lineales cuya respuesta está basada en ordenar los pixeles abarcados por una máscara y luego reemplazar el valor del pixel central con el valor determinado por el resultado del ordenamiento.

El más conocido de éstos es el filtro de mediana, el cual reemplaza el valor del pixel central por la mediana de los niveles de gris del vecindario de ese pixel. Los filtros de mediana son usados para eliminar o reducir ciertos tipos de ruidos aleatorios, y generan un borronado menor que los filtros lineales de suavizado del mismo tamaño. Los filtros de mediana son particularmente efectivos cuando el ruido es del tipo impulso (también llamado ruido sal y pimienta) debido a que aparece como puntos negros o blancos sobrepuestos en la imagen.

Esta técnica fue desarrollada por Tukey (1977) y Pratt (1978) a finales de los años

setenta, y partió de la idea de conseguir un suavizado de imágenes y eliminación de ruido aplicando una técnica no lineal, pero de simple implementación. Al ser un filtrado no lineal, hay que empezar destacando las diferencias entre un filtrado lineal y otro que no lo sea. En ambos casos, se ha de recorrer la imagen pixel a pixel, y operar con los pixeles cercanos para obtener un suavizado de la imagen, que reduzca el ruido. En el caso del filtrado por mediana, la operación que se implementa es la convolución, en este caso bidimensional, de una matriz de pixeles con otra matriz de coeficientes, que define el comportamiento del filtro, como lo ilustra 3.19:

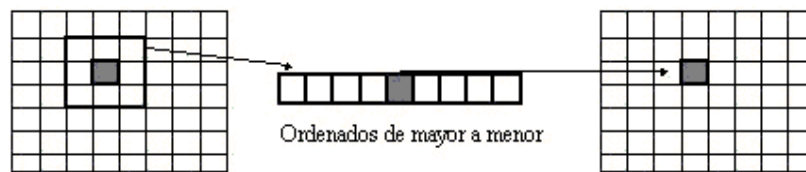


Figura 3.19: Funcionamiento del filtro de mediana

En el filtrado de mediana, el nivel de gris de cada pixel se reemplaza por la mediana de los niveles de gris en un entorno de este pixel, en lugar de por la media. Recordar que la mediana M de un conjunto de valores es tal que la mitad de los valores del conjunto son menores que M y la mitad de los valores mayores que M , es decir en un conjunto ordenado de mayor a menor o viceversa, sería el valor de la posición central.

El filtro de la mediana no puede ser calculado con una máscara de convolución, ya que es un filtro no lineal. Podemos ver como este tipo de filtro elimina totalmente el punto que tenía un valor muy diferente al resto de sus vecinos. Como se selecciona el valor de centro, el filtrado de mediana consiste en forzar que puntos con intensidades muy distintas se asemejen más a sus vecinos, por lo que observamos que el filtro de mediana es muy efectivo para eliminar pixeles cuyo valor es muy diferente del resto de sus vecinos, como por ejemplo eliminando ruido de la imagen.

Filtros Pasa-altos

El objetivo principal de estos filtros es resaltar los detalles delicados de una imagen (por ejemplo, los bordes) o realzar los detalles de una imagen borrosa.

$\frac{df}{dx} = f(x+1, y) - f(x, y)$	$\frac{df}{dx} = f(x+1, y) + f(x-1, y) - 2f(x, y)$
$\frac{df}{dy} = f(x, y+1) - f(x, y)$	$\frac{df}{dy} = f(x, y+1) + f(x, y-1) - 2f(x, y)$

Tabla 3.2: Derivada de una funcion bidimensional

El suavizado de una imagen, en el dominio espacial, se logra promediando el vecindario de cada pixel, dado que el promediado es análogo a la integración, es lógico pensar que el filtrado con un pasaaltos se realice mediante operaciones de diferenciación espacial. La diferenciación de imagen realza o enfatiza los bordes y otras discontinuidades tales como el ruido y atenúa aquellas áreas cuyos niveles de gris varían lentamente.

Las derivadas de una función discreta se definen en términos de diferencias. Existen muchas maneras de definir las, sin embargo, algunos requerimientos mínimos que debe cumplir cualquier definición de aquéllas.

La primera derivada debe ser:

- 0 en las zonas que tienen un nivel de gris constante
- $\neq 0$ ante un cambio del tipo escalón o rampa en el nivel de gris
- $\neq 0$ en toda la zona que tiene un cambio de tipo rampa

Cualquier definición de segunda derivada debe ser:

- 0 en zonas con nivel de gris constante
- $\neq 0$ al principio o al final de un cambio del tipo escalón o rampa
- 0 en toda la zona que tiene un cambio del tipo rampa con pendiente constante

Las definiciones básicas para la primer y segunda derivada de una función bidimensional discreta con respecto a x y a y son las que se muestran en la Tabla 3.2

Los bordes en una imagen se relacionan con los cambios bruscos en el nivel de gris, y para analizarlos de una manera eficiente, se han desarrollado diferentes métodos para detectar líneas y bordes en general:

- Métodos asociados con la primera derivada
- El gradiente
- Métodos asociados con la segunda derivada (Laplaciano)

Detección de líneas

Si aplicamos ciertas máscaras en el dominio espacial podemos obtener respuestas favorables en el análisis de ciertas líneas que nos interesa analizar, por ejemplo si queremos observar únicamente las líneas horizontales en la imagen deseada podemos aplicar una máscara con la forma:

$$\begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix} \quad (3.12)$$

La cual obtendrá una respuesta máxima cuando la línea pase a lo largo de la segunda fila de la máscara.

Para una máscara que nos responda mejor para líneas orientadas a 45° podemos aplicar la siguiente matriz:

$$\begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} \quad (3.13)$$

De ésta forma se pueden aplicar filtros sencillos para realizar un análisis de las líneas que nos interesan, como en la Figura 3.20.

Detección de bordes

Operador Gradiente. El gradiente de una imagen $f(x, y)$ en la posición (x, y) viene dado por el vector ∇f

$$\nabla f = \begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

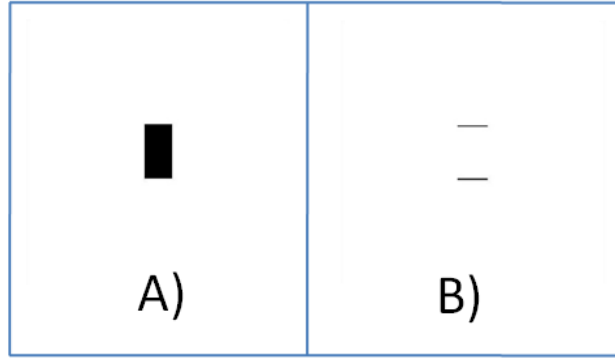


Figura 3.20: Ejemplo de aplicación de un filtro para detectar líneas horizontales

El vector gradiente siempre apunta en la dirección de la máxima variación de la imagen f en el punto (x, y) . En la detección de bordes es muy importante la magnitud de este vector, dado por $\|\nabla f\|$

$$\|\nabla f\| = \sqrt{f_x^2 + f_y^2}$$

Esta cantidad representa la variación de la imagen $f(x, y)$ por unidad de distancia en la dirección del vector ∇f . En general, el gradiente se suele aproximar mediante la expresión 3.14 que es mucho más simple de implementar en la práctica.

$$\nabla f \approx |f_x| + |f_y| \quad (3.14)$$

La dirección del vector gradiente también es una cantidad importante.

Sea $\alpha(x, y)$ el ángulo del vector ∇f en el punto (x, y) , se tiene la ecuación 3.15, donde los ángulos se miden respecto al eje de abscisas.

$$\alpha(x, y) = \text{Atan2}(f_y, f_x) \quad (3.15)$$

El cálculo del gradiente se basa en obtener las derivadas parciales para cada pixel.

Las derivadas se pueden implementar digitalmente de varias formas. De forma matricial se pueden representar los operadores de Roberts 3.16, Prewitt 3.17, Sobel 3.18 y Frei-Chen

3.19 para determinar las derivadas parciales. Sin embargo, los operadores de Sobel y de Frei-Chen tienen la ventaja de que proporcionan un suavizado además del efecto de derivación, el filtro sharpen 3.20 mejora los bordes, pero tiene la desventaja que puede incrementar el ruido en las imágenes.

A continuación se muestran algunos operadores de tipo derivación:

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (3.16)$$

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.17)$$

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (3.18)$$

$$\begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix} \quad (3.19)$$

$$\begin{bmatrix} -0,125 & -0,125 & -0,125 \\ -0,125 & 2,000 & -0,125 \\ -0,125 & -0,125 & -0,125 \end{bmatrix} \quad (3.20)$$

3.7.5. Filtrado e histograma de una imagen

El histograma de una imagen contiene la información de la probabilidad de aparición de las distintas tonalidades de color que se pueden dar en cada caso, ya que podemos trabajar en distintos tipos de colores o en escala de grises. En el caso de una imagen en color, no podemos hablar de un único histograma que caracteriza a la imagen, sino de tres histogramas, uno para

cada color (RGB, por ejemplo). Si bien el uso del histograma y sus posteriores modificaciones son más aplicables a imágenes en escala de grises.

La creación de un histograma, en el caso de una imagen en escala de 256 tonalidades del blanco al negro (escala de grises), se realiza fácilmente mediante medios informáticos, de manera que, en primer lugar crearemos un vector (array) que contenga 256 posiciones, una por cada nivel de gris, el algoritmo recorrerá cada uno de los píxeles de la imagen, aumentando en una unidad el valor guardado en la posición del array correspondiente al tono del píxel en cuestión, un ejemplo se muestra en la Figura 3.21.

El histograma proporciona una descripción de la apariencia global de una imagen. De forma que si los niveles de gris están concentrados hacia el extremo oscuro del rango de la escala de gris, la apariencia global de la imagen será oscura; mientras que si sucede justo lo contrario, la imagen correspondiente será brillante. Por su parte, un histograma que presente un perfil estrecho corresponderá a una imagen de bajo contraste y un histograma con una dispersión considerable a una imagen de alto contraste.

Durante la creación del histograma se considera la aplicación de una determinada función (T) a los niveles de gris de la imagen original (r) para obtener los niveles de gris de la imagen transformada (s) como la ecuación 3.21

$$s = T(r) \quad (3.21)$$

El histograma de una imagen corresponde a la función densidad de probabilidad de los niveles de gris que componen una imagen. Visualizando de esta forma el histograma y teniendo en cuenta el teorema de cambio de variable, se tiene que la función densidad de probabilidad de los niveles de gris de la imagen transformada se escribe como la ecuación 3.22:

$$P_s(s) = \left[P_r(r) \frac{dr}{ds} \right]_{r=T^{-1}(s)} \quad (3.22)$$

En el caso de la imagen en color, aparecen tres histogramas, de forma que el tratamiento de imágenes en color se complica por la aparición de nuevos componentes. Ahora el histograma no contiene los tonos del negro al blanco, sino del negro al color correspondiente

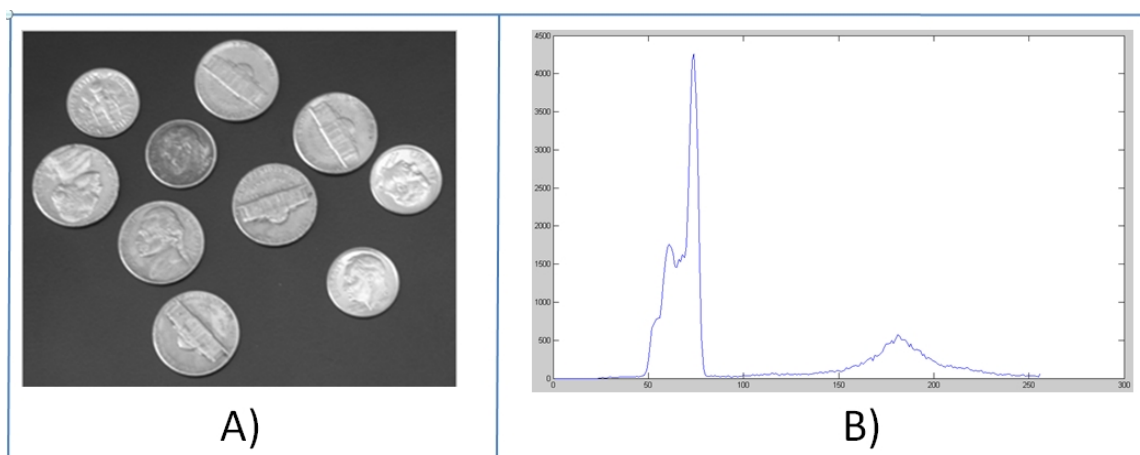


Figura 3.21: Imagen de 10 monedas y su histograma correspondiente

(rojo, verde ó azul -caso RGB-). Al elaborar el histograma, el algoritmo debe separar el color correspondiente a cada pixel en sus componentes RGB (rojo, verde y azul). El tema en el que nos centraremos más tarde, es decir, la ecualización del histograma, es poco viable a la hora de trabajar con imágenes en color, ya que la ecualización se realiza por separado, y al recomponer la imagen los resultados no son muy satisfactorios.

Ecualizacion del histograma

La ecualización global del histograma de una imagen, al igual que otros muchos tratamientos que se basan en la manipulación del histograma, son mucho más usados para imágenes en escala de grises, por lo que será este caso en cuestión el que se analice más profundamente. El tratamiento que vamos a estudiar en este caso es el de ecualización del mismo.

La ecualización del histograma consiste, muy básicamente y a grandes rasgos, en una expansión del histograma de la imagen, dotando al mismo de mayor linealidad y haciendo que éste ocupe el ancho del espectro de tonalidades grises por completo, ello implica unas mejoras en la imagen que serán expuestas a continuación:

- Una mayor utilización de los recursos disponibles: al ecualizar el histograma, vemos como los tonos que antes estaban más agrupados, ahora se han separado, ocupando

todo el rango de grises, por lo que la imagen se está enriqueciendo al tener niveles de gris más distintos entre sí, mejorando, por tanto, la apariencia visual de la imagen.

- Un aumento del contraste: esta ventaja es consecuencia del punto anterior, ya que si hacemos que el histograma de la imagen ocupe todo el rango de grises, estamos aumentando la distancia entre el tono más claro y el más oscuro, convirtiendo a estos, en blanco y negro y consecuentemente aumentando el contraste de la imagen.
- Constituye una regulación óptima y automática del contraste de la imagen. Evitando los ajustes manuales con los que no se consigue un equilibrio óptimo entre el blanco y el negro.

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

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

Para el caso de que el formato se encuentre en escala de grises de 8 bits, es decir 256 tonos distintos de gris se pueden aplicar los pasos que se comentan a continuación.

Se parte del histograma original, el primer paso para trabajar con el mismo es el de normalización, para ello hay que normalizar tanto el eje horizontal (niveles de gris) como el vertical (aparición del nivel de gris en cuestión), de manera que ambas magnitudes queden comprendidas entre 0 y 1. Para normalizar el eje horizontal, es decir, el que indica el nivel de gris, dividimos cada magnitud entre 255, con lo que queda: $0, 1/255, 2/255 \dots 255/255=1$, obteniendo un rango de niveles de gris comprendido entre 0 y 1. Para la normalización del eje vertical, debemos dividir cada componente entre el número total de píxeles, así en el caso más extremo, es decir en el que toda la imagen fuese de un mismo color, la representación sería una única barra de altura igual a 1.

El siguiente paso consiste en la acumulación del histograma, esto consiste en sustituir cada magnitud, por el valor de ella misma más el total de la suma de las anteriores, con ello conseguimos que el histograma (que reflejaba la función "densidad de probabilidad" para cada color) se transforme en una representación de la función de distribución. Quedando una función creciente, cuyo valor máximo será siempre 1. La acumulación del histograma, sería equivalente a la expresión 3.23:

$$s = T(r) = \int_0^r p_r(w)dw \quad (3.23)$$

para $0 \leq r \leq 1$

Siempre que los niveles de gris fuesen cantidades continuas (infinitos niveles de gris), con lo que conseguiríamos la expresión 3.24:

$$P_s(s) = \left[p_r(r) \frac{d_r}{d_s} \right]_{r=T^{-1}(s)} = \left[p_r(r) \frac{1}{p_r(r)} \right]_{r=T^{-1}(s)} = 1_{r=T^{-1}(s)} = 1 \quad (3.24)$$

para $0 \leq s \leq 1$

Es decir, la función densidad de probabilidad de los niveles de gris de la imagen resultante sería uniforme (como era nuestro objetivo).

Aplicando la expresión 3.25, obtendremos los nuevos niveles de gris (haciendo que el histograma se extienda hasta el negro), además conseguiremos que estos vuelvan a encontrarse entre 0 y 255:

$$s^* = Int \left[\frac{s - s_{\min}}{1 - s_{\min}} 255 + 0,5 \right] \quad (3.25)$$

Donde "Int" representa la operación de tomar el entero más cercano por defecto.

s_{\min} es el menor valor de s distinto de cero.

En nuestro caso, 255 representa el número de niveles de gris menos uno.

Veamos un ejemplo que posiblemente pueda clarificar todo lo expuesto anteriormente. Supongamos que disponemos de una imagen de 9 pixeles cuyos niveles de gris son:

253	253	255
253	255	254
253	254	254

El histograma normalizado de dicha imagen sería el siguiente:

$$H(i) = 0 \leq i \leq 252/255$$

$$H(253/255) = 4/9 \quad H(254/255) = 3/9 \quad H(255/255) = H(1) = 2/9$$

Mientras que el histograma normalizado acumulado quedaría:

$$HA(i) = 0 \leq i \leq 252/255$$

$$HA(253/255) = 4/9 \quad HA(254/255) = 7/9 \quad HA(255/255) = HA(1) = 1$$

Los niveles de gris de la imagen original (253, 254, 255) se transformarían en:

$$\begin{aligned} Nivel253 &\rightarrow Int \left[\frac{\frac{4}{9}-\frac{4}{9}}{1-\frac{4}{9}} \cdot 255 + 0,5 \right] \rightarrow 0 & Nivel254 &\rightarrow \left[\frac{\frac{7}{9}-\frac{4}{9}}{1-\frac{4}{9}} \cdot 255 + 0,5 \right] \rightarrow 153 \\ Nivel255 &\rightarrow \left[\frac{\frac{9}{9}-\frac{4}{9}}{1-\frac{4}{9}} \cdot 255 + 0,5 \right] \rightarrow 255 \end{aligned}$$

Con lo que la imagen resultante sería la siguiente:

$$\begin{array}{ccc} 0 & 0 & 255 \\ 0 & 255 & 153 \\ 0 & 153 & 153 \end{array}$$

La igualación del histograma, al igual que cualquier transformación de intensidades, puede implementarse también mediante una LUT (Look-UpTable) como lo muestra la Figura 3.22

3.7.6. Segmentación de imagenes

En las técnicas de procesamiento de las imágenes cada pixel era transformado de un valor a otro. Estos cambios, mayoritariamente, se realizan para facilitar la partición de la imagen en áreas de pixeles con significado. En la segmentación de imágenes, se trata de agrupar los pixeles, por algún criterio de homogeneidad, para particionar la escena en regiones de interés. Por tanto, la segmentación de una imagen es un proceso de extracción de objetos de interés insertados en la escena capturada. La agrupación de los pixeles se tiene a razón de que sus vecinos sean similares en criterios como de luminancia, color, bordes, texturas, movimientos, etc.

Una vez que la imagen ha sido particionada, la unidad dejara de ser el pixel para ser la agrupación de pixeles que constituye el objeto. La imagen estará definida por un conjunto

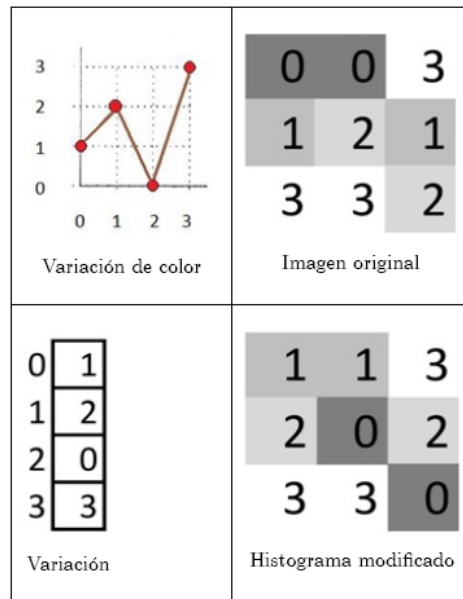


Figura 3.22: Modificación del histograma mediante una LUT

de objetos, habiendo pasado de un nivel bajo a otro más elaborado o nivel medio visual. La información estará preparada para el reconocimiento e interpretación de la imagen.

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

1. Similitud: los pixeles agrupados del objeto deben ser similares respecto algún criterio (nivel de gris, color, borde, textura,...).
2. Conectividad: los objetos corresponden a áreas de pixeles con conectividad. Las particiones corresponden con regiones continuas de pixeles.
3. Discontinuidad: los objetos tienen formas geométricas que definen unos contornos. Estos bordes delimitan unos objetos de otros.

Técnicas basadas en los bordes

Para que un pixel etiquetado como borde se defina como pixel frontera de un objeto se necesitará que otros pixeles bordes tengan similar dirección y módulo del gradiente. Dos

pixeles serán considerados pertenecientes a una misma frontera se presentan alguna condición de conectividad y las diferencias entre sus gradientes no superan un determinado umbral:

$$||G_1| - |G_2|| < T_M$$

$$|\vartheta_1 - \vartheta_2| < T_A$$

Donde $|G_i|$ y $|\vartheta_i|$ se ha denotado como el módulo y argumento del gradiente del pixel i , respectivamente.

Un planteamiento funcional es la agrupación de los pixeles que cumplen un cierto criterio en una primitiva más elaborada, como puede ser un pequeño segmento con orientación. Posteriormente, se procede a encadenar estas primitivas dando paso a la delimitación de los objetos.

Muchas aplicaciones se basan en las localizaciones de líneas rectas o de otras formas básicas como círculos o elipses, ya que se consideran elementos fundamentales en la percepción de entornos estructurados o semi-estructurados. Se basa en el hecho de que las escenas artificiales, construidas por el hombre, están plagadas de contornos rectos o de objetos con formas circulares o elípticas. En este sentido, se aplican las transformadas de Hough para la localización de estas fronteras.

Transformada de Hough

Este algoritmo trata de detectar formas geométricas sencillas en una imagen. En su procesamiento emplea el conjunto total de la imagen, haciéndolo robusto ante la presencia del ruido o discontinuidades de las etapas previas. De hecho, para su ejecución requiere de una imagen binarizada en la que se han seleccionado previamente los bordes. Por tanto, la entrada al algoritmo de Hough es una imagen en la que se ha hecho la detección de los bordes. El mayor inconveniente es su alto costo computacional.

Considerando un pixel de coordenadas (x_i, y_i) , seleccionado como elemento que representa un borde; sobre éste pasarán infinitas rectas de la forma 3.26:

$$y_i = ax_i + b \tag{3.26}$$

El modelo de todas estas posibilidades quedan definidas por los infinitos valores de a y b . Haciendo la transformación al espacio paramétrico, los valores de a y b quedarán definidos como la ecuación 3.27:

$$b = y_i - ax_i \quad (3.27)$$

Al variar el parámetro a desde $-\infty$ a $+\infty$ se obtendrá los infinitos valores de b . La representación geométrica de $b = y_i - ax_i$, en el espacio paramétrico, será una recta. La característica interesante de esta presentación consiste que si dos pixeles que pertenezcan a una misma línea son representados en el espacio paramétrico, el modelo de la recta está definido en la intersección de las dos rectas del espacio paramétrico.

La transformada de Hough (1962) aplica este concepto para la localización de líneas rectas en la imagen. Discretiza el espacio paramétrico en intervalos de $[a_{min}, a_{max}]$ y $[b_{min}, b_{max}]$, creando una rejilla de celdas de acumulación. Por cada pixel, considerado como borde, se hace recorrer el rango dinámico de a , obteniendo los valores de b . Por cada valor de a y b se incrementa el valor en la celda correspondiente. Esta operación se hace con todos los pixeles etiquetados como bordes. Al finalizar, aquellas celdas con mayor valor, indicarán la presencia de rectas en la imagen, cuyos modelos corresponderán con las coordenadas de la celda.

La representación de una línea en forma normal se muestra en la Figura 3.23 inciso A) la cual responde a la ecuación 3.28

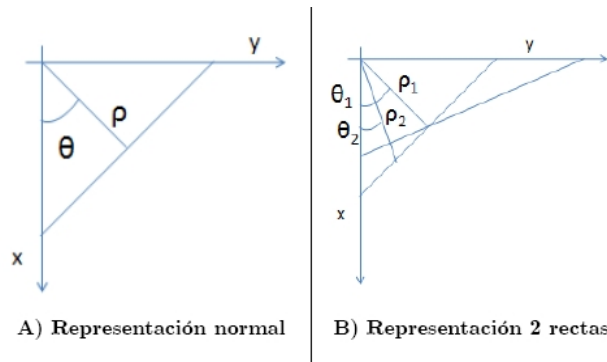


Figura 3.23: Representación normal de la recta

$$x \cos \theta + y \cos(90^\circ - \theta) = x \cos \theta + y \sin \theta = \rho \quad (3.28)$$

Por un punto pueden pasar, en teoría, infinitas líneas. La Figura 3.23 inciso B) muestra 2 rectas con diferentes valores de ρ y θ que pasan por un punto.

Aplicando la transformada de Hough se convierte la representación $[x, y]$ a $[\rho, \theta]$, como se muestra en la representación de la Figura 3.24

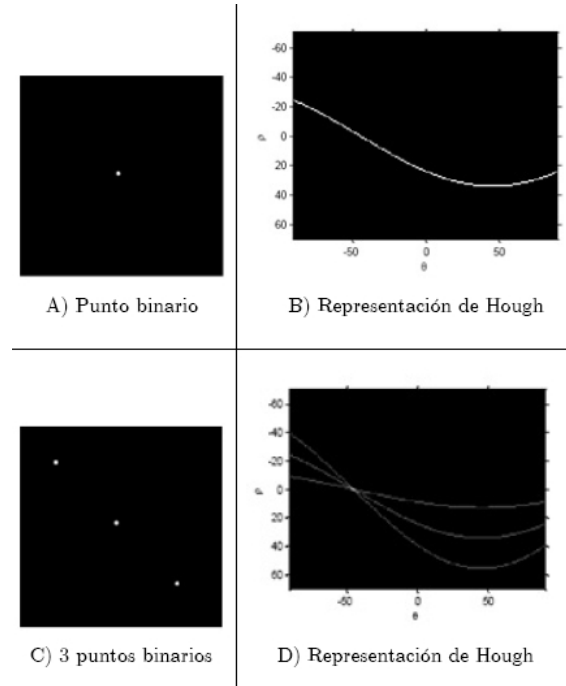


Figura 3.24: Imágenes binarias y su representación paramétrica de Hough

En el espacio de Hough habrá tantas curvas sinusoidales como puntos hayan, como se observa en la Figura 3.24

Existe un punto donde se interceptan las tres curvas sinusoidales, lo cual indica la posibilidad de existencia de una línea. Para determinar los valores de ρ y θ que definen esa línea, se determina donde se interceptan las curvas sinusoidales.

Por el inciso D) de la Figura 3.24, podemos observar que los valores que buscamos son $\rho = 0$ y $\theta = -45$. Si sustituimos los valores encontrados en la ecuación 3.28 obtendremos la ecuación 3.29:

$$0,7x - 0,7y = 0 \quad (3.29)$$

Lo cual describe la línea que se compone de los 3 puntos del inciso C) de la Figura 3.24.

Umbralización

La umbralización es una técnica de segmentación ampliamente utilizada en las aplicaciones industriales. Se emplea cuando hay una clara diferencia entre los objetos a extraer respecto del fondo de la escena. Los principios que rigen son la similitud entre los píxeles pertenecientes a un objeto y sus diferencias respecto al resto. Por tanto, la escena debe caracterizarse por un fondo uniforme y por objetos parecidos.

Al aplicar un umbral, la imagen de niveles de grises quedará binarizada; etiquetando con ‘1’ los píxeles correspondientes al objeto y con ‘0’ aquellos que son del fondo. Por ejemplo, si los objetos son claros respecto del fondo, se aplicará:

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

$f(x, y)$ es la función que retorna el nivel de gris del píxel (x, y) , $g(x, y)$ será la función de la imagen binarizada y T es el umbral. En el caso de que los objetos sean oscuros respecto del fondo, la asignación sería a la inversa

Segmentación por histograma La segmentación por histograma es una técnica global que se basa en considerar una definición clara de los objetos respecto del fondo, con una iluminación relativamente uniforme en toda la escena. Se supone que el histograma está constituido por dos picos y un valle. Al segmentar empleando el histograma, se denota que este proceder no considera las relaciones de vecindad de los píxeles. El umbral estará definido justamente en el valle.

Esta técnica es extensible a un histograma con N picos y $N - 1$ valles, presentado varios umbrales. Sin embargo, a mayor número de umbrales en la imagen, menos fiables serán los resultados. La umbralización por histograma presenta ciertas limitaciones:

- Es difícil identificar correctamente los mínimos del histograma

- Se tienen problemas cuando las regiones varían suavemente su nivel (por ejemplo, el efecto de las sombras)
- Se aplica sólo cuando hay pocas regiones
- No se pueden distinguir regiones separadas de niveles similares de gris
- No considera la conectividad de los píxeles. Debido a este inconveniente, se suele emplear a continuación técnicas de post-procesado

Existen diversas estrategias para la elección de un umbral óptimo, las cuales están basadas en:

- Ajuste gaussiano
- Minimización de la varianza interclase
- Entropía del sistema
- Métodos basados en momentos.

Un resumen detallado de los distintos algoritmos se puede encontrar en revistas especializadas como [Sahoo88]. De entre todas las propuestas destaca el umbral de Otsu, por su alta utilización en las aplicaciones de visión artificial.

Método de Otsu Una imagen es una función bidimensional de la intensidad del nivel de gris, y contiene N píxeles cuyos niveles de gris se encuentran entre 1 y L . El número de píxeles con nivel de gris i se denota como f_i , y la probabilidad de ocurrencia del nivel de gris i en la imagen está dada por

$$P_i = \frac{f_i}{N}$$

En el caso de la umbralización en dos niveles de una imagen (a veces llamada binarización), los píxeles son divididos en dos clases:

- C_1 , con niveles de gris $[1, \dots, t]$
- C_2 , con niveles de gris $[t + 1, \dots, L]$.

Entonces, la distribución de probabilidad de los niveles de gris para las dos clases son:

$$C_1 = \frac{P_1}{w_1(t)}, \dots, \frac{P_t}{w_1(t)}$$

$$C_2 = \frac{P_{t+1}}{w_2(t)}, \frac{P_{t+2}}{w_2(t)} \dots, \frac{P_L}{w_2(t)}$$

La media para la clase C_1 y la clase C_2 es:

$$\mu_1(t) = \sum_{i=1}^t \frac{iP_i}{w_1(t)}$$

$$\mu_2(t) = \sum_{i=t+1}^L \frac{iP_i}{w_2(t)}$$

donde

$$w_1(t) = \sum_{i=1}^t P_i$$

$$w_2(t) = \sum_{i=t+1}^L P_i$$

Sea μ_T la intensidad media de toda la imagen. Es fácil demostrar que

$$w_1 + w_2 = 1$$

Otsu definió la varianza entre clases de una imagen binarizada como:

$$\sigma_B^2(t) = w_1(\mu_1 - \mu_T)^2 + w_2(\mu_2 - \mu_T)^2$$

Para una umbralización de 2 niveles, Otsu verificó que el umbral óptimo t^* se elige de manera que σ_B^2 sea máxima; esto es:

$$t^* = \max_t \{ \sigma_B^2(t) \} \quad (3.30)$$

Por ejemplo, analicemos la imagen f_1 para obtener el umbral óptimo para la misma

$$f_1 = \begin{bmatrix} 1 & 1 & 2 & 2 & 2 \\ 1 & 8 & 9 & 8 & 1 \\ 2 & 7 & 8 & 9 & 2 \\ 2 & 8 & 9 & 7 & 2 \\ 4 & 4 & 3 & 3 & 3 \end{bmatrix}$$

$$N = 25$$

$$L = 9$$

$$f_1 = 4 \quad f_2 = 7 \quad f_3 = 3 \quad f_4 = 2 \quad f_5 = 0$$

$$f_6 = 0 \quad f_7 = 1 \quad f_8 = 4 \quad f_9 = 3$$

$$P_1 = 0,16 \quad P_2 = 0,28 \quad P_3 = 0,12 \quad P_4 = 0,08 \quad P_5 = 0$$

$$P_6 = 0 \quad P_7 = 0,08 \quad P_8 = 0,16 \quad P_9 = 0,12$$

Para $t = 1$ tenemos que:

$$w_1(t) = \sum_{i=1}^1 P_i = 0,16$$

$$w_2(t) = \sum_{i=2}^9 P_i = 0,28 + 0,12 + 0,08 + 0,08 + 0,16 + 0,12 = 0,84$$

$$w_1(t) + w_2(t) = 1$$

$$\mu_1(t) = \sum_{i=1}^1 \frac{iP_i}{w_i(t)} = \frac{1*0,16}{0,16} = 1$$

$$\mu_2(t) = \sum_{i=2}^9 \frac{iP_i}{w_2(t)} = \frac{(2*0,28)+(3*0,12)+(4*0,08)+(7*0,08)+(8*0,16)+(9*0,12)}{0,84} = 4,9524$$

$$\mu_T(t) = w_1(t)\mu_1(t) + w_2(t)\mu_2(t) = 0,16 + (0,84 * 4,9) = 4,318$$

$$\sigma_B^2(t) = w_1(\mu_1 - \mu_T)^2 + w_2(\mu_2 - \mu_T)^2 = 0,16(1 - 4,318)^2 + 0,84(4,9524 - 4 - 318)^2 = 3,215$$

Siguiendo la metodologia para los valores de t hasta 9 obtenemos la Tabla 3.3

Y según 3.30 observamos el valor máximo de $\sigma_B^2(t)$ el cual es 35,1561, como se repite 3 veces, se puede tomar los valores de 4, 5 ó 6 para una correcta segmentación de la imagen

t	$w_1(t)$	$w_1(t)$	$\mu_1(t)$	$\mu_2(t)$	$\sigma_B^2(t)$
1	0,16	0,84	1	4,95	3,215
2	0,44	0,56	1,64	6,43	16,210
3	0,56	0,44	1,93	7,36	26,77
4	0,64	0,36	2,19	8,11	35,1561
5	0,64	0,36	2,19	8,11	35,1561
6	0,64	0,36	2,19	8,11	35,1561
7	0,72	0,28	2,72	8,428	25,748
8	0,88	0,12	3,68	9	7,046
9	1	0	4,32	0	0

Tabla 3.3: Resultado de tabular los valores del método de Otsu

3.7.7. Operaciones morfológicas en imágenes binarias

La morfología matemática es una herramienta muy utilizada en el procesamiento de imágenes.

A través de las operaciones morfológicas es posible simplificar la información que nos muestra una imagen, preservar las características esenciales y eliminar aspectos irrelevantes.

Teniendo en cuenta que la identificación y descomposición de objetos, la extracción de rasgos, la localización de defectos e incluso los defectos en líneas de ensamblaje están sumamente relacionados con las formas, es obvio el papel tan importante que juega la morfología matemática en el procesamiento de imágenes.

La morfología matemática se puede usar, entre otros, con los siguientes objetivos:

- Preprocesamiento de imágenes (supresión de ruido, simplificación de formas).
- Destacar la estructura de objetos (extraer el esqueleto, ampliación, reducción).
- Descripción cualitativa de objetos (área, perímetro, diámetro, etc).

Una imagen binaria puede definirse como una función de dos variables discretas $a[m, n]$ que puede tomar dos valores, 0 o 1 , dependiendo del nivel de gris de la imagen (una imagen binaria tiene dos niveles: blanco y negro).

También es posible proponer una definición alternativa si consideramos que una imagen consiste en un conjunto de coordenadas discretas (también pueden ser reales). En este sentido, el conjunto corresponde a todos aquellos puntos o píxeles que pertenecen a la imagen. Por lo tanto, se puede decir que en morfología matemática los conjuntos representan objetos en una imagen. Por ejemplo, el conjunto de todos los píxeles negros en una imagen binaria constituye una descripción completa de la misma.



Figura 3.25: Imagen binaria

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Representación de la imagen 3.25

En las imágenes binarias, los conjuntos en cuestión pertenecen al espacio Z^2 , el cual está comprendido en el espacio bidimensional, formado por las coordenadas $[m, n]$ que definen al conjunto, donde cada elemento del conjunto es una 2-upla (vector 2-D) cuyas coordenadas son las coordenadas $[m, n]$ de un píxel blanco (o negro, según la convención) de la imagen.

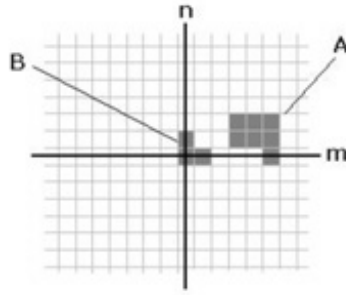


Figura 3.26: Imagen binaria con dos conjuntos

En la Figura 3.26 se pueden ver dos conjuntos, A y B . Observemos que se ha colocado un sistema de coordenadas. El conjunto A consiste en los puntos $\{[2, 3]; [2, 4]; [2, 5]; [1, 3]; [1, 4]; [1, 5]; [0, 5]\}$ mientras que el B contiene los puntos $\{[0, 0]; [0, 1]; [1, 0]\}$.

Cabe recordar que en la mayoría de los lenguajes de programación los arreglos de elementos que serán los encargados de contener la imagen no admiten índices negativos y en general menores a 1. Por lo tanto, será necesario realizar una pequeña modificación al sistema de coordenadas que consiste en un simple desplazamiento para poder operar sobre una imagen (contenida en un arreglo).

Conceptos Básicos de la Teoría de Conjuntos Sea A un conjunto en Z^2 . Si $a = (a_1, a_2)$ es un elemento de A , entonces podemos afirmar que

$$a \in A$$

Similarmente, si a no es un elemento de A , decimos que

$$a \notin A$$

El conjunto que no tiene elementos se llama conjunto vacío y se denota por el símbolo \emptyset .

Un conjunto se especifica por el contenido de dos llaves: $\{\}$. Los elementos de un conjunto son las coordenadas de pixeles que representan objetos u otros rasgos de interés en la imagen.

Por ejemplo, cuando escribimos la expresión $C = \{w | w = -d, d \in D\}$ queremos decir que el conjunto C es el conjunto de elementos, w , tal que w sea el resultado de multiplicar por -1 cada una de las dos coordenadas de todos los elementos de D .

Si cada elemento del conjunto A es también un elemento de otro conjunto B , se dice que A es un subconjunto de B , y se expresa como

$$A \subseteq B$$

La unión de dos conjuntos A y B , denotada por

$$C = A \cup B$$

es el conjunto de todos los elementos pertenecientes a A o a B o a ambos.

La intersección de conjuntos A y B , denotada por

$$D = A \cap B$$

es el conjunto de todos aquellos elementos que pertenecen tanto a A como a B .

Dos conjuntos A y B se dicen mutuamente excluyentes o disjuntos si no tienen elementos en común. En este caso,

$$A \cap B = \emptyset$$

El complemento de un conjunto A es el conjunto de elementos que no pertenecen a A :

$$A^c = \{w \mid w \notin A\}$$

La diferencia de dos conjuntos A y B , denotada por $A-B$, se define como

$$A - B = \{w \mid w \in A, w \notin B\} = A \cap B^c$$

Además de estas operaciones básicas, vamos a necesitar de otras adicionales que son usadas ampliamente en morfología pero que generalmente no están en los textos básicos de la teoría de conjunto. La reflexión del conjunto B , que se escribe como

$$\hat{B} = \{w \mid w = -b, b \in B\}$$

La traslación de un conjunto A por el punto (vector) $z = (z_1, z_2)$, denotada $(A)_z$, se define como

$$(A)_z = \{c \mid c = a + z, a \in A\}$$

Las imágenes de la Figura 3.27 ejemplificarán las operaciones más comunes de la teoría de conjuntos aplicadas a imágenes binarias.

Dilatación y Erosión

Estas operaciones son fundamentales en el procesamiento morfológico. De hecho, la mayoría de los algoritmos morfológicos están basados en estas dos operaciones.

Elemento Estructural: Si bien los conjuntos A y B pueden ser considerados como una imagen u objeto, generalmente se considera en operaciones morfológicas que A es la imagen y B es el elemento estructural que va a operar sobre el conjunto A .

El elemento estructural es en morfología matemática lo que la máscara (o núcleo) de convolución es en los filtros lineales.

Los elementos estructurales más comunes son los conjuntos que están 4-conectados, N_4 , y 8-conectados, N_8 , ilustrados en las siguientes matrices

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ Elemento estructural } N_4$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \text{ Elemento estructural } N_8$$

Dilatación Sean A y B conjuntos en Z^2 . La dilatación de A por B , expresada por $A \oplus B$, se define como

$$A \oplus B = \left\{ z \mid (\hat{B})_z \cap A \neq \emptyset \right\}$$

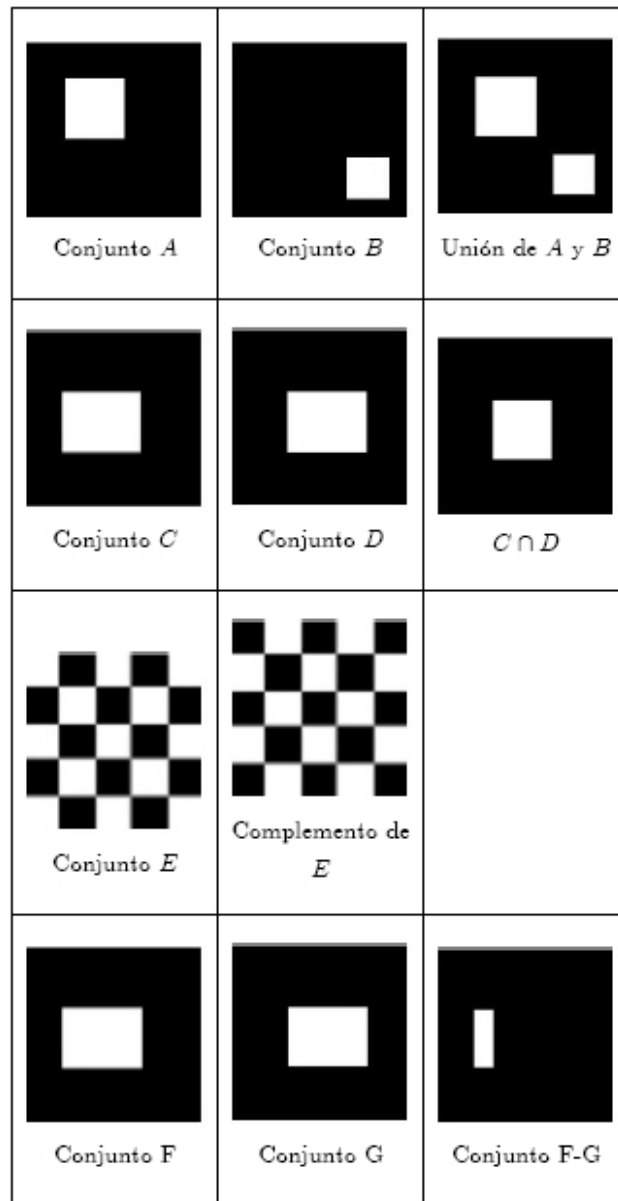


Figura 3.27: Ejemplos de morfología de imágenes

Esta ecuación consiste en obtener la reflexión de B sobre su origen y trasladar este reflexión por z . La dilatación de A por B es entonces el conjunto de todos los desplazamientos, z , tal que la reflexión de B y A se solapan por al menos un elemento. Teniendo en cuenta lo anterior, la dilatación de A por B también se puede expresar como

$$A \oplus B = \left\{ z \mid (\hat{B})_z \cap A \subseteq A \right\}$$

En general, la dilatación aumenta el tamaño de un objeto. La cantidad y la forma en que aumenta el tamaño depende de la elección del elemento estructural.

Erosión Sean A y B conjuntos en Z^2 . La erosión de A por B , que se expresa como $A \ominus B$, se define como

$$A \ominus B = \{ z \mid (B)_z \subseteq A \}$$

Esta ecuación indica que la erosión de A por B es el conjunto de todos los puntos z tales que B , trasladado por z , está contenido en A .

Generalmente, la erosión disminuye el tamaño de los objetos. Como pasaba en la dilatación, la cantidad y la forma en que se produce esta disminución depende del elemento estructural elegido.

Uno de los usos más simples de la erosión es para la eliminación de detalles irrelevantes (en términos de tamaño) de una imagen binaria.

El inciso A) de la Figura 3.28 muestra una imagen compuesta por cuadrados cuyos lados tienen 1, 3, 5, 9, y 15 píxeles. Supongamos que queremos eliminar todos los cuadrados excepto los más grandes. Esto los podemos hacer erosionando la imagen con un elemento estructural cuyo tamaño sea un poco menor que el de los cuadrados que deseamos conservar. Por ejemplo, si elegimos un elemento estructural de 13x13, podemos obtener la imagen del inciso B) de la Figura 3.28. Como se observa, sólo se han mantenido las porciones de los cuadrados más grandes. Luego, podemos restituir el tamaño de estos 3 cuadrados a su tamaño original de 15x15 dilatando la imagen erosionada (inciso B) de la Figura 3.28 con el mismo elemento estructural utilizado para la erosión. El resultado puede verse en el inciso C) de la Figura 3.28.

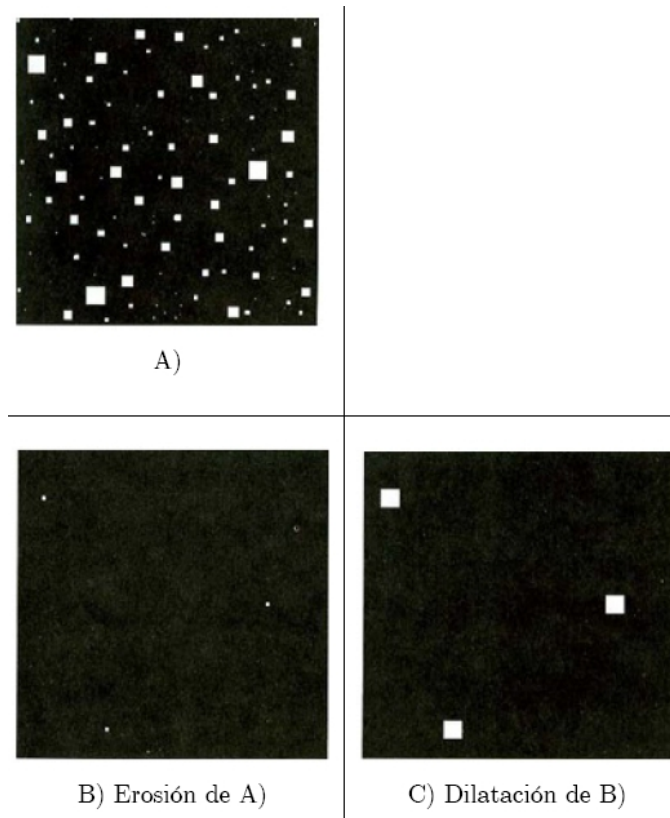


Figura 3.28: Separación de objetos aplicando dilataciones y erosiones

Apertura y Cierre (Opening and Closing) 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 usan de forma complementaria, bien la dilatación seguida de la erosión o la erosión seguida de la dilatación.

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

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

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

Es decir, la apertura de A por B es la erosión de A por B seguida por la dilatación del resultado por B .

De forma similar, la cierre de un conjunto A por el elemento estructural B , se define como

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

A grandes rasgos, es la dilatación de A por B seguida por la erosión del resultado por B .

La apertura generalmente suaviza los contornos de un objeto y elimina protuberancias finas.

El cierre 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.

La figura del inciso A) en la Figura 3.29 muestra un conjunto, el inciso B) de la Figura 3.29 muestra varias posiciones de un elemento estructural con forma de disco durante el proceso de erosión. El resultado de la erosión se puede ver en el inciso C) de la Figura 3.29. Se debe notar la eliminación de las uniones entre las dos secciones principales. El ancho de estas uniones era menor al diámetro del elemento estructural. Los elementos salientes donde el elemento estructural (con forma de disco) no cabía, fueron eliminados. La figura del inciso D) de la Figura 3.29 muestra la dilatación del conjunto erosionado, y la figura del inciso E) de la Figura 3.29 el resultado final de la apertura. Nótese que las esquinas exteriores han sido redondeadas mientras que las esquinas interiores no fueron afectadas.

De manera similar, las figuras a partir del inciso F) de la Figura 3.29 al inciso I) de la Figura 3.29 muestran los resultados del cierre morfológico del inciso A) con el mismo elemento estructural. Notamos que las esquinas interiores fueron redondeadas mientras que las esquinas exteriores permanecieron sin cambios. También se ve el suavizado que resultó en las partes del objeto luego de la apertura o del cierre del conjunto original con el elemento estructural circular.

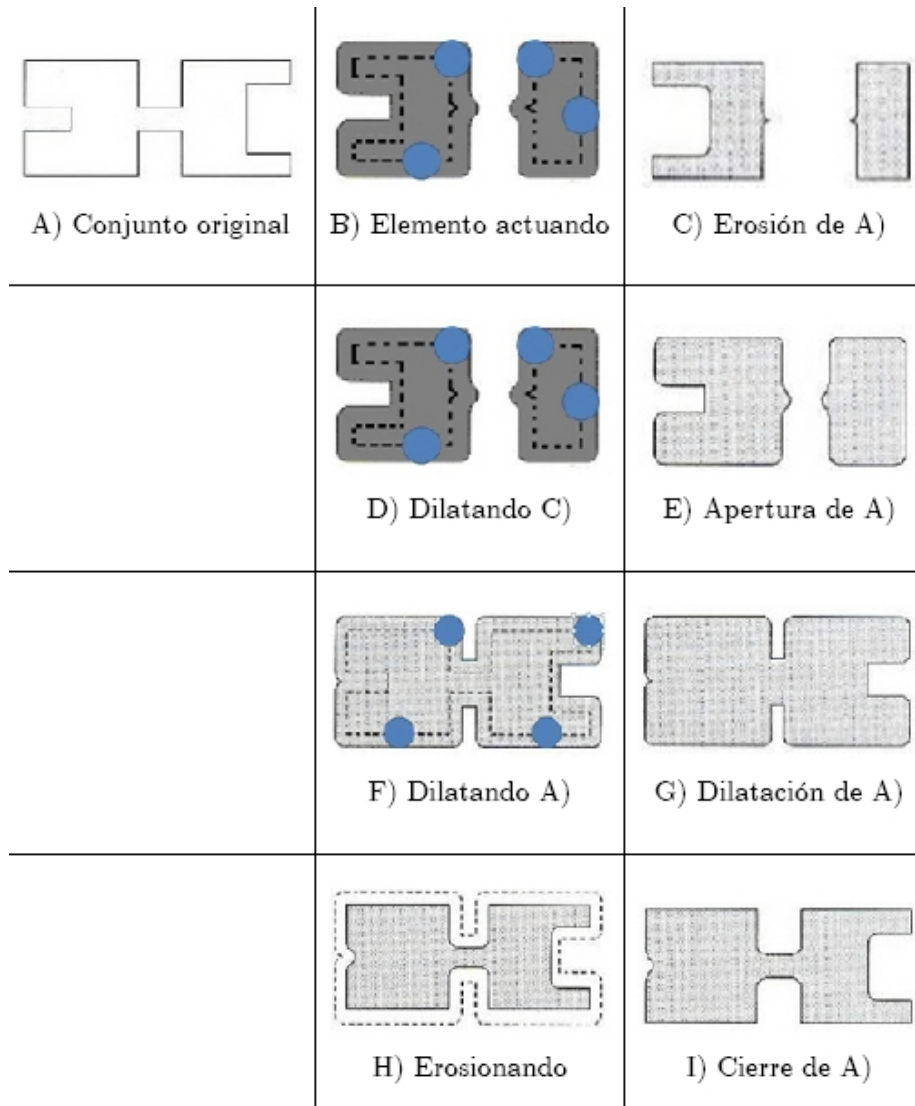


Figura 3.29: Apertura y cierre de una imagen

Como sucedía con la dilatación y la erosión, la apertura y el cierre son duales con respecto

a la complementación y a la reflexión. Como lo expresa la ecuación 3.31

$$(A \bullet B)^C = (A^C - \hat{B}) \quad (3.31)$$

Las operaciones morfológicas pueden ser usadas para construir filtros conceptualmente similares a los filtros espaciales, algunos ejemplos se muestran en la Figura 3.30.

Algunos ejemplos en que se pueden aplicar la apertura, cierre, dilatación y erosión son:

- Conexión de objetos
- Separación de objetos
- Extracción de contornos
- Rellenado de regiones

Extracción de Frontera (Boundary Extraction) La frontera de un conjunto A , escrita como $\beta(A)$, se puede obtener erosionando A por B y luego calcular la diferencia entre A y su erosión. Se expresa como 3.32 siendo β el elemento estructural adecuado

$$\beta(A) = A - (A \ominus B) \quad (3.32)$$

Un ejemplo de la operación de frontera se muestra en la Figura 3.31, cuyo inciso A) muestra una imagen binaria, y el resultado correspondiente en el inciso B), utilizando la ecuación anterior con un elemento estructural de 3×3 estándar N_8

También existen otras elecciones más complejas, por ejemplo, de tamaño 5×5 el cual ampliaría el grosor de la frontera a dos o tres píxeles.

Relleno de Región (Region Filling) En ciertas aplicaciones, es conveniente poder reconstruir una imagen que ha sufrido varias erosiones o poder llenar un objeto que está definido por una frontera. Para ello, se puede utilizar una operación llamada relleno de región cuyos algoritmos están basados en dilataciones, complementaciones e intersecciones.

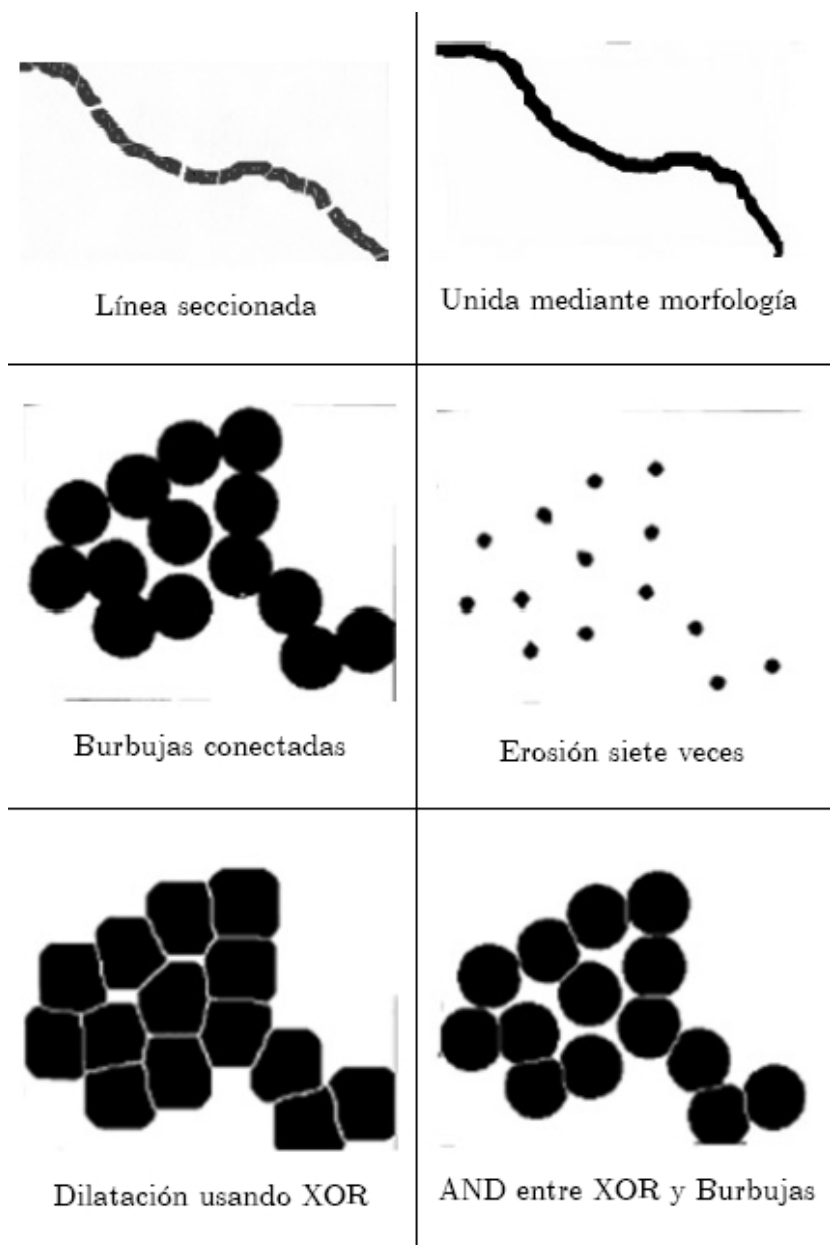


Figura 3.30: Ejemplos de aplicaciones en morfología

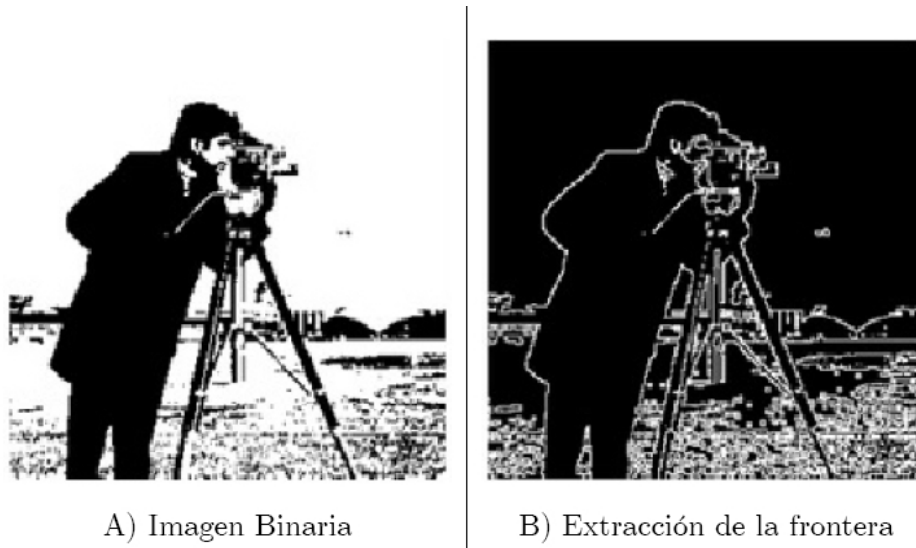


Figura 3.31: Extracción de la frontera de la imagen

La Figura 3.32 inciso A) muestra un conjunto que contiene un subconjunto cuyos elementos están 8-conectados y constituyen la frontera de una región. Comenzando por un punto p (conocido como ‘semilla’ o ‘seed’) que se encuentre dentro de la frontera, el objetivo es rellenar completamente la región con 1’s.

Consideremos que todos aquellos puntos que no pertenecen a la frontera son cero (el fondo o background) y asignemos al punto p (la semilla) el valor uno. El siguiente procedimiento, entonces, rellena la región determinada por el punto p con 1’s:

$$X = (X_{k-1} \oplus B) \cap A^c$$

$$k = 1, 2, 3, \dots$$

donde $X_0 = p$, y B es el elemento estructural de la Figura 3.32 inciso C). El algoritmo converge cuando $X_k = X_{k-1}$. El conjunto resultante de la unión de X_k y A contiene a la región rellena y su frontera.

Es importante notar que el proceso de dilatación rellenaría todo el conjunto. Sin embargo, la intersección en cada paso con A^c limita el proceso a la región de interés (este tipo de dilatación recibe el nombre de dilatación condicionada). Aunque en este ejemplo sólo rellenos una región o subconjunto, el concepto puede aplicarse a un número finito de regiones, siempre que conozcamos puntos dentro de éstas (las semillas).

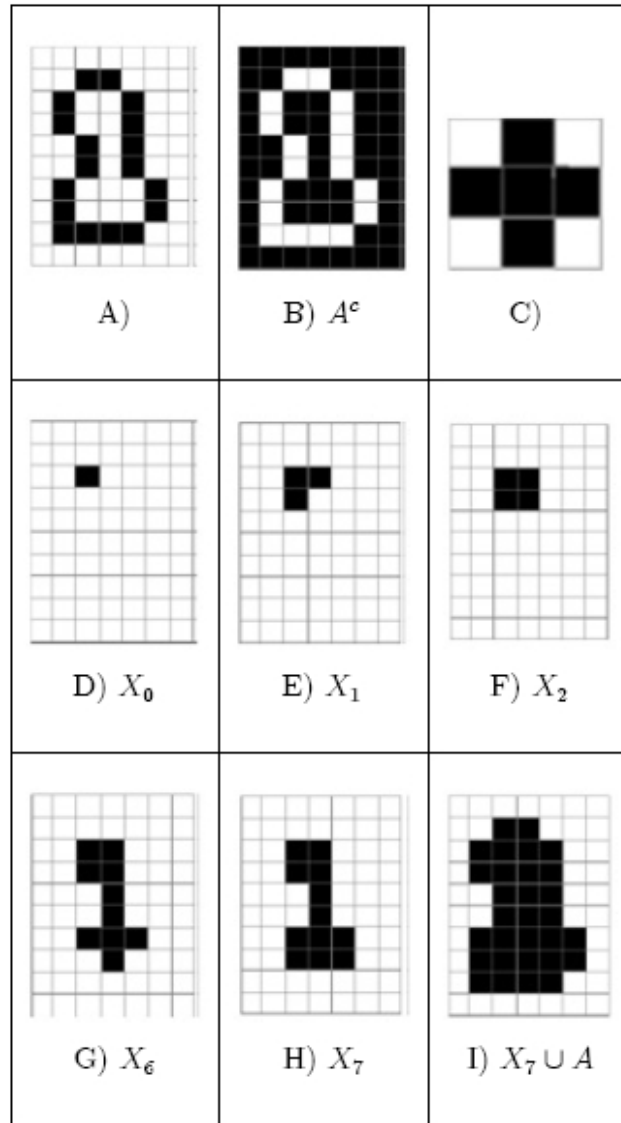


Figura 3.32: Proceso de relleno de una imagen

3.8. Protocolo de comunicación

En el caso de la cámara profesional DVT530 desarrollada por Cognex, debemos utilizar los siguientes controles Active X:

- DVTSID. Proporciona un vínculo entre las cámaras *DVT smart cameras* y la aplicación en la PC. Está diseñado para mostrar imágenes muestreadas desde la cámara para monitorear el proceso de inspección. El control también puede enviar un limitado paquete de comandos para monitorear el proceso de inspección, los cuales son transferidos en una conexión Ethernet a través del protocolo TCP/IP.
- DVT DL. Permite una apropiada comunicación, en plataformas windows que carecen del componente Winsock Control

También es necesario deshabilitar o agregar una excepción para usar el DVTDL y DVT-SID, en el firewall de windows a partir del sistema operativo windows xp service pack 2

Para que la comunicación sea adecuada entre los equipos, se debe configurar la dirección IP de el servidor y de la cámara

El código generado desde Visual Basic 6.0 para conexión con la cámara DVT se muestra a continuación:

```
Private Sub cmd_Conectar_Click()  
    DLControl1.RemoteHost = Texto_IP.Text  
    DVTSID1.RemoteHost = Texto_IP.Text  
  
    'DVTSID1.RemoteHost = DLControl1.Text  
    'DVTSID1.RemoteHost = "192.168.85.125"  
  
    DVTSID1.Connect  
    cmd_Play.Enabled = True  
End Sub\medskip
```

```

Public Sub cmd_Play_Click()
    DVTSID1.PlayImages (All)
    'Do Events
    cmd_Conectar.Enabled = False
    cmd_Guardar_Colgar.Enabled = True
End Sub\medskip

Private Sub cmd_Guardar_Colgar_Click()
    DVTSID1.StopImages
    DVTSID1.Disconnect
    cmd_Conectar.Enabled = True
    cmd_Play.Enabled = False
End Sub\medskip

Public Sub DVTSID1_OnImage()
    DVTSID1.SaveImage ("C:\failedImage.bmp")
End Sub

Private Sub Form_Load()
    cmd_Conectar.Enabled = True
    cmd_Play.Enabled = False
    cmd_Guardar_Colgar.Enabled = False
End Sub\medskip

Private Sub Texto_IP_Click()
    Texto_IP.Text = " "
End Sub\medskip

```

La Figura 3.33 representa la aplicación que corresponde al código descrito anteriormente, la cual es una interfaz muy sencilla pues sólo captura la imagen en el directorio "*C:\failedImage.bmp*" del disco duro.

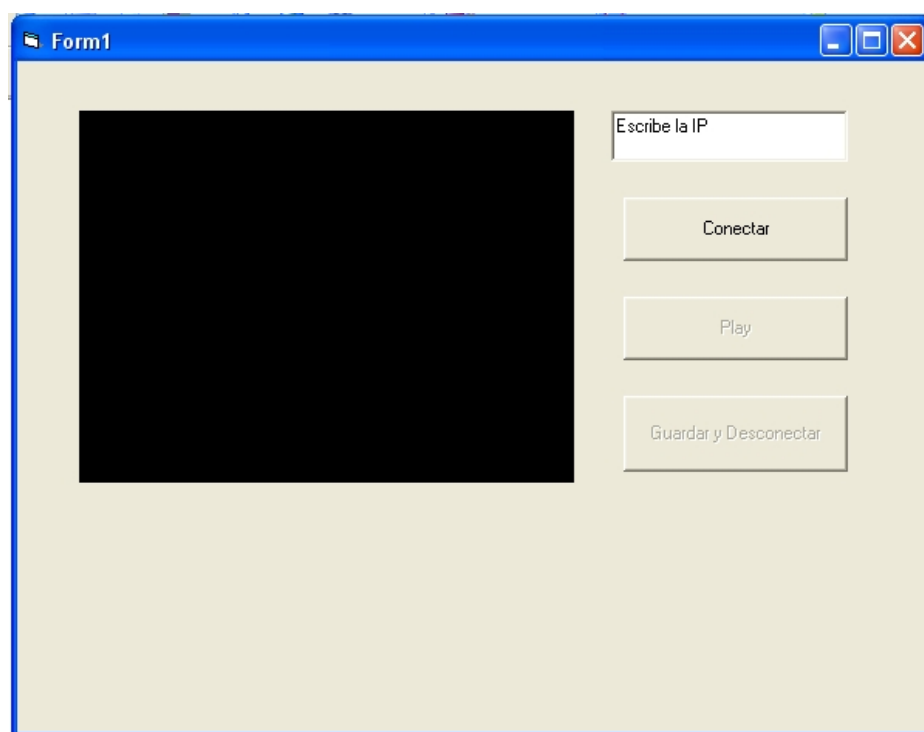


Figura 3.33: Interfaz gráfica de comunicación con la cámara DVT530

Capítulo 4

Propuesta de solución

Entre las propuestas que el mercado ofrece para el procesamiento de imágenes podemos encontrar entre otros:

- National Instruments
- Matlab

Ahora analizaremos las propuestas de cada compañía basados en ventajas y desventajas

4.1. National instruments

National Instruments (NI) es un proveedor líder de herramientas de hardware y software de visión artificial e imagen científica. Desde inspección de partes automotrices hasta investigación de medicina avanzada; ingenieros e investigadores usan software y hardware de visión de NI para resolver más rápido y a un menor costo una variedad de retos en aplicaciones.

NI ofrece un conjunto completo de hardware para imágenes científicas y visión artificial incluyendo tarjetas insertables para sistemas basados en PC y PXI, sistemas de visión artificial embebida y cámaras inteligentes. Un solo conjunto de software controlador, el software NI Vision Acquisition, es compatible con todos los estándares de las cámaras Camera Link,

GigE Vision, IEEE 1394, analógicas y paralelas digitales. Este software ofrece un API para comunicarse con todo el hardware de adquisición de imagen de NI, y es compatible con NI LabVIEW, C, Visual Basic y Microsoft .NET.

Las tarjetas de adquisición de imágenes de NI funcionan de forma transparente e integrada con hardware de NI de adquisición de datos, control de movimiento y CAN. Los dispositivos de adquisición de imagen de NI para PCI, PCI Express y PXI Express comparten señales de temporización para sincronizar y correlacionar imágenes con otras medidas.



Figura 4.1: Sistema NI compact vision



Figura 4.2: NI Smart camera

La familia de NI Smart Cameras consiste en modelos con resoluciones VGA (640 x 480 píxeles) y SXGA (1280 x 1024 píxeles), así como modelos que contienen un coprocesador DSP que proporciona rendimiento óptico para algoritmos comunes incluyendo reconocimiento óptico de caracteres y lectura de código. Todas estas cámaras incluyen dos puertos gigabit Ethernet para comunicarse con E/S de expansión y con redes industriales. Las NI Smart Cameras funcionan con toda la gama de algoritmos de visión disponible en la plataforma de software de visión de NI.

Para procesar imágenes, el software NI Vision está incluido en dos diferentes paquetes: el Módulo NI Vision Development y NI Vision Builder for Automated Inspection (AI). El Módu-

lo Vision Development es una colección de cientos de funciones de visión para programadores que utilizan LabVIEW, NI LabWindowsTM/CVI1 , C/C++ o Visual Basic. Esta biblioteca también incluye una herramienta, llamada NI Vision Assistant, para rápida generación de prototipos para visión. Esta herramienta ofrece ventajas que incrementan la probabilidad de éxito en su aplicación y genera ambos códigos, de LabVIEW y C. Vision Builder AI es un entorno interactivo de software para configurar, evaluar y desplegar aplicaciones de visión artificial sin programar.

4.2. Matlab

Matlab es un lenguaje de programación de alto rendimiento, el cual integra procesamiento, visualización y programación en un entorno de fácil manejo donde los problemas y soluciones se expresan en notación matemática.

Algunos usos comunes son:

- Desarrollo de algoritmos
- Adquisición de datos
- Modelado, simulación y creación de prototipos
- Análisis, visualización y exploración de datos
- Desarrollo de aplicaciones, incluyendo la construcción de una interfaz gráfica de usuario.

Matlab es un sistema interactivo cuyo elemento básico de datos es un arreglo, el cual no requiere ser dimensionado. Esto permite resolver muchos problemas técnicos al programar, especialmente aquellos relacionados con matrices y vectores, de una forma mucho más eficiente y rápida de lo que sería, si se utilizara un programa que maneja valores escalares como C o Fortran por ejemplo.

Para desarrollar aplicaciones Matlab cuenta con una herramienta gráfica llamada GUIDE, la cual es un entorno de programación visual para realizar y ejecutar programas que necesiten

ingreso continuo de datos. Tiene las características básicas de todos los programas visuales como Visual Basic o Visual C++.

Como software de análisis para la propuesta de solución se utilizará el entorno de programación Matlab

4.2.1. Selección de la aplicación

El paquete que se optó adecuado para realizar el desarrollo de la aplicación es Matlab por las siguientes razones:

- El precio total de adquirir el software

Matlab	
MATLAB [®] , Simulink [®] , Symbolic Math Toolbox [™]	\$ 1,022.00
Image Processing Toolbox [™]	\$ 140.00
Image Acquisition Toolbox [™]	\$ 140.00
	\$ 1,302.00

National instruments	
NI Vision Development Module	\$ 3,960.00
NI Vision Development Module Runtime License	\$ 460.00
Sistema de Desarrollo Profesional de LabVIEW (Windows)	\$ 4,730.00
	\$ 9,150.00

En precio, matlab tendría un costo menor, siendo el desarrollador el único que requiere comprar el paquete.

- Matlab es un software que permite visualizar en forma matricial, paso a paso el desarrollo de la aplicación mediante un lenguaje sencillo y de fácil aprendizaje mediante el archivo de ayuda que incluye el programa
- Matlab tiene una mayor facilidad de instalación y mediante el modo de depuración se resuelven con mayor facilidad los problemas

4.3. Algoritmo a desarrollar

El algoritmo que resolverá el problema de la medición de piezas contemplará las siguientes etapas:

1. Adquisición de la imagen
2. Conversión de la imagen en formato RGB a escala de grises
3. Procesamiento de la imagen en escala de grises para acentuar los rasgos de interés
4. Convertir la imagen mejorada a una imagen binaria mediante una apropiada segmentación de la misma.
5. Convertir los segmentos en líneas y áreas, para analizarlas con el fin de conocer los diámetros de interés
6. Análisis de la imagen binaria para evaluación de la pieza

Se propone la utilización de una interfaz gráfica de usuario, debido a que la adquisición de los datos es dinámica y el usuario debe tener control del análisis, siendo deseable que la participación sea limitada.

4.3.1. Adquisición de la imagen

La adquisición de la imagen se puede realizar de dos formas:

1. Abriendo la imagen desde una ubicación en el disco duro
2. Adquiriendo la imagen desde una cámara

4.3.2. Conversión de la imagen en formato RGB a escala de grises

La conversión a escala de grises es necesaria debido a que, disminuye el tiempo de procesamiento y no se pierden los rasgos que definen a la misma.

4.3.3. Incremento de la calidad de la imagen adquirida

Para el ajuste del contraste, de tipo logarítmico, se debe escalar la imagen al rango 0-1, siendo 255 el máximo valor y que corresponde a 1 en la nueva imagen.

El primer paso para realizar el análisis que se propone, es capturar la imagen y guardarla en memoria, para posteriormente trabajar con la misma, éste procedimiento se ilustra en la Figura 4.3



Figura 4.3: Imagen RGB adquirida por la cámara web

Posteriormente se decide recortar el área a analizar, con el fin de acelerar la velocidad del procesamiento de los datos, también es conveniente trabajar con la imagen en escala de grises pues de ésta forma, la matriz que define la imagen es de $m \times n \times 1$

El siguiente paso es aplicar una mejora en la calidad de la imagen, apoyándonos en los filtros de estadísticos o espaciales, que más convengan a la aplicación, para éste caso se opta por filtros de mediana y de mejora tipo "*sharpen*" (una clase especial de filtro pasa altos ver por ejemplo [Gonzales-Woods]), y también una mejora en el contraste de la imagen para obtener la imagen preprocesada de la Figura 4.4

4.3.4. Conversión a imagen binaria.

Posteriormente se trabaja con las líneas que define el filtro tipo *canny*, para ubicar cuál es la que define el perímetro general de la pieza.

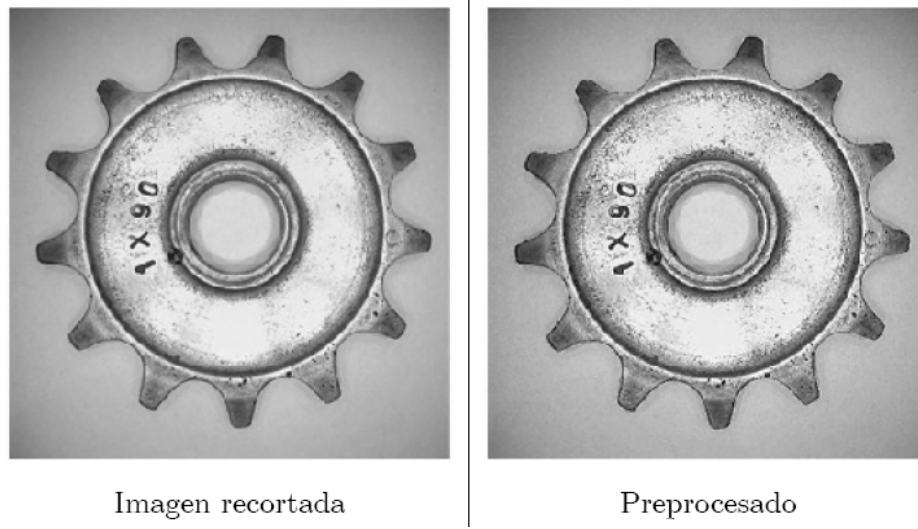


Figura 4.4: Preprocesado de la imagen

El procedimiento es básicamente:

- Definir las líneas que conforman la pieza.
- Se estima un tamaño de línea de 500 unidades como líneas significativas, y eliminan a las que tienen menor tamaño.
- Se cuenta la cantidad de pixeles que están conectados para cada línea y se localiza al de mayor tamaño.
- Se eliminan todas las demás líneas conservando el perímetro general de la pieza

Con el perímetro de la pieza se procede a formar el área y ubicar el centroide de la misma como se muestra en la Figura 4.5

Analizando la pieza se tienen básicamente 3 partes que la constituyen:

1. El diámetro donde ingresa la flecha
2. El área que en la Figura 4.6 tiene escrito 1x90
3. La parte que contiene los dientes

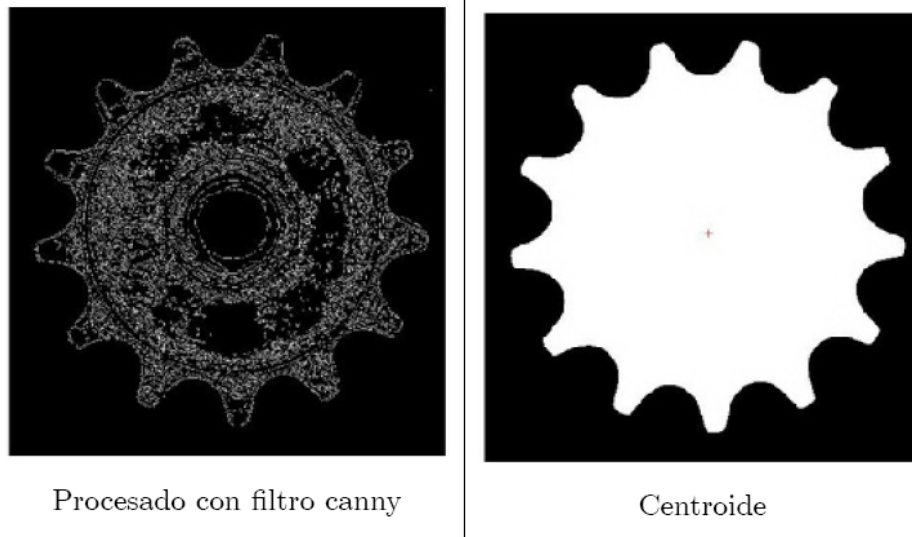


Figura 4.5: Obtención de líneas y centroide de la pieza

Utilizando el método de Otsu para localizar objetos y distinguirlos del fondo, obtenemos la figura 4.6

4.3.5. Conversión de segmentos en líneas, áreas, y posterior análisis

El siguiente paso es procesar nuevamente la imagen, para hacer más evidentes las partes que conforman la pieza, en este caso se aplica un elemento estructural tipo disco de 10 elementos de radio, denotado por SE

Posteriormente verificamos que las áreas queden definidas por diferentes colores, como se muestra en la Figura 4.7, lo cual es signo de que son piezas que no tienen conectividad la una con la otra

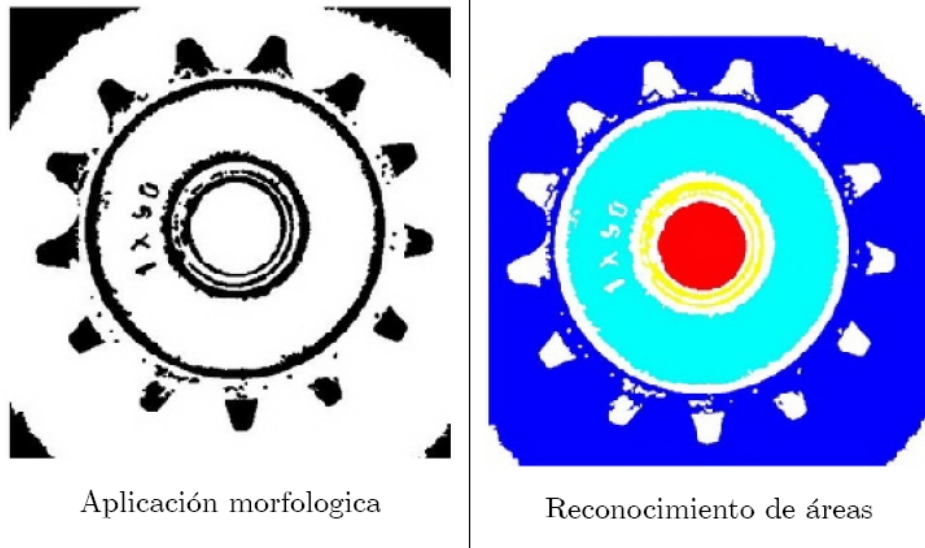


Figura 4.7: Separación de las áreas de interés

Se selecciona área por área, aplicamos un cierre morfológico a la imagen con el elemento estructural SE previamente definido, lo cual permite eliminar los huecos dentro de los límites de cada pieza, si el área final consiste en 2 líneas es un área de interés como se muestra en la Figura 4.8

El siguiente procedimiento consiste en analizar las líneas que se obtienen de las áreas segmentadas, y se describe a continuación

1. Localizar un punto de cada línea, y guardar las coordenadas de n puntos que están 8 conectados con el punto localizado
2. Reescribir la ecuación del círculo de la siguiente forma:

$(x - x_c)^2 + (y - y_c)^2 = r^2$ donde las literales con subíndice c representan las coordenadas del centro de la pieza.

Reacomodando el sistema de ecuaciones se tiene:

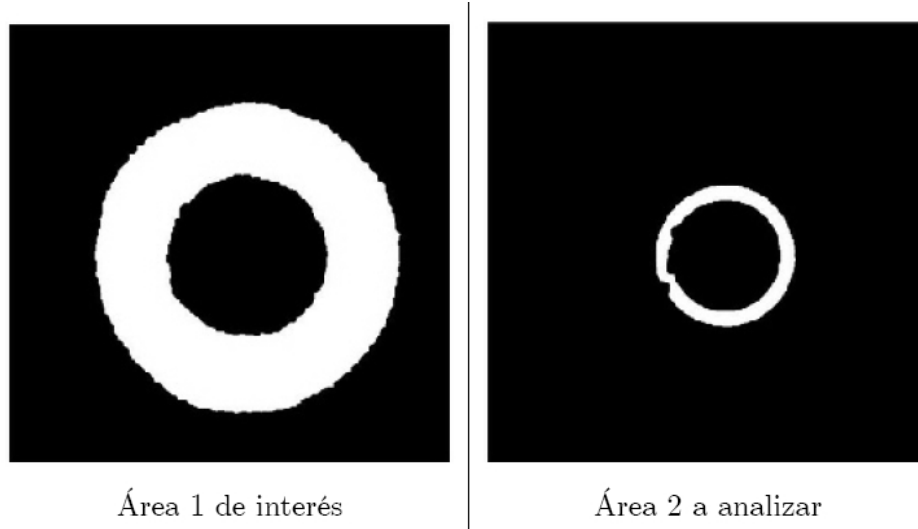


Figura 4.8: Selección de áreas que se requiere procesar

$$x^2 - 2xx_c + x_c^2 + y^2 - 2yy_c + y_c^2 - r^2 = 0$$

Que se puede ver como:

$$x^2 + y^2 + ax + by + c = 0$$

siendo

$$a = -2x_c$$

$$b = -2y_c$$

$$c = x_c^2 + y_c^2 - r^2$$

El sistema de ecuaciones final a resolver se puede reacomodar como:

$$\begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = -(x^2 + y^2) \text{ o bien en matlab se puede desarrollar como}$$

$$\begin{bmatrix} a & b & c \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} \setminus -(x^2 + y^2)$$

Y teniendo las coordenadas x y y guardadas de los n puntos se calculan los términos a, b y c

Finalmente se puede despejar los valores del centro del círculo y el radio como:

$$x_c = -a/2$$

$$y_c = -b/2$$

$$r = \sqrt[2]{(x_c^2 + y_c^2 - c)}$$

Se aplica el mismo procedimiento para ubicar los radios que conforman a la segunda área de interés, lo cual nos resulta en la Figura 4.9

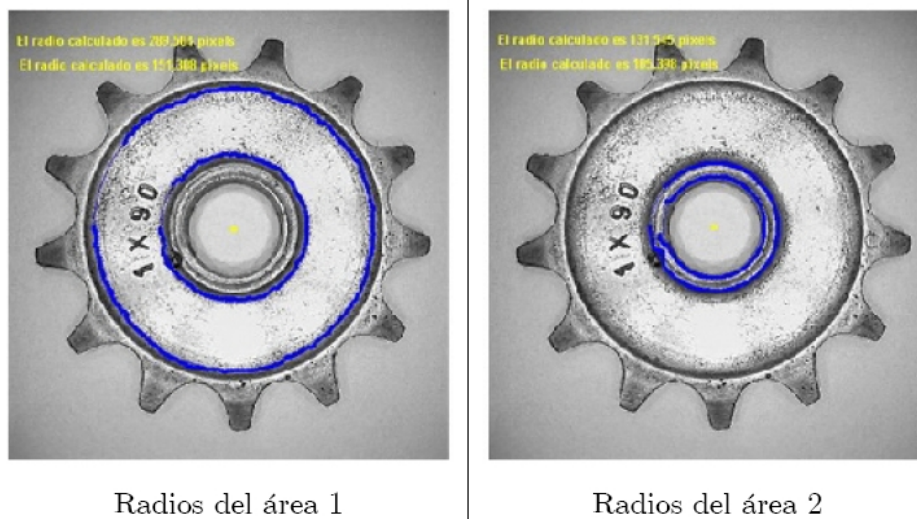


Figura 4.9: Graficado de los radios localizados

Se comprueban los radios sobreponiéndolos en la figura RGB para saber si son coherentes

Como siguiente paso se propone encontrar el diámetro exterior de la pieza, con el fin de analizar los dientes. Un inconveniente de éste tipo de piezas es, que los parámetros de límites cambian conforme se va rotando la pieza, por lo tanto se requieren diversas mediciones para saber cuál es el diámetro real, el proceso se muestra en la Figura 4.11

Después de saber cuáles son los límites reales que pueden definir al círculo exterior, se crea una imagen con un círculo de diámetro igual al calculado, y se aplica una operación lógica y morfológica que defina la negación de la unión de ambas piezas, la cual entrega la Figura 4.12 y a través de la cual se pueden analizar las áreas formadas por los huecos de

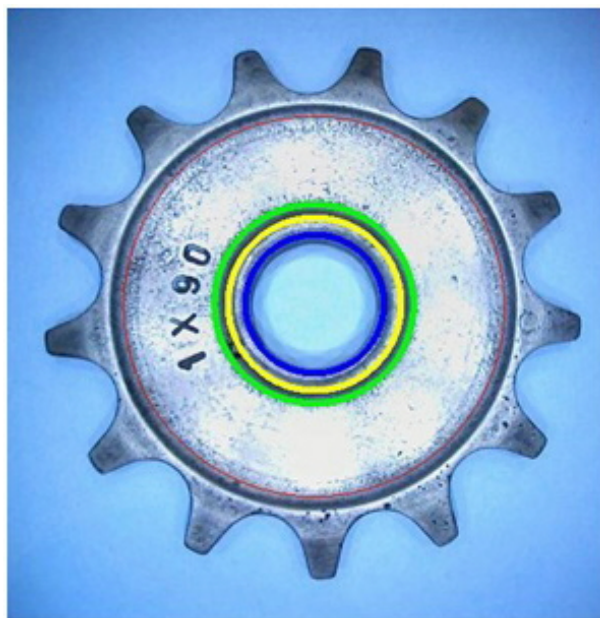


Figura 4.10: Superposición de las líneas encontradas en la figura original para analizarlas

cada diente, se pueden medir tanto áreas como alturas y ancho de diente, el alto y ancho son principalmente los valores importantes, pues ayudan a realizar correctamente el movimiento entre piezas.

Finalmente se realiza la última medición que se decidió como crítica para la pieza, el diámetro último que delimita los dientes de la pieza y la parte circular que nos interesa. Se definen las líneas principales, aplicando un filtro tipo canny y un umbral menor (más próximo a cero) que el aplicado previamente, para obtener las líneas más contrastantes por el método de Otsu.

Se encuentra la línea que define al diámetro con el siguiente procedimiento

1. Obtener la propiedad BoundingBox de todas las líneas, la cual es una herramienta que calcula 4 valores, 2 coordenadas que definen la esquina superior izquierda y 2 que definen el ancho y alto de la mínima caja que podría contener a la figura

Se ordenan los valores en forma creciente y se localizan posibles círculos, basados en que un círculo puede ser contenido en un cuadrado, que tiene por tamaño de lados el diámetro











		
Sólido rotado 3°	Sólido rotado 6°	Sólido rotado 9°
		
Sólido rotado 12°	Sólido rotado 15°	Sólido rotado 18°
		
Sólido rotado 21°	Sólido rotado 24°	Sólido rotado 27°
		
	Sólido rotado 30°	

Figura 4.11: Los pixeles mínimo y máximo varían en cada rotación

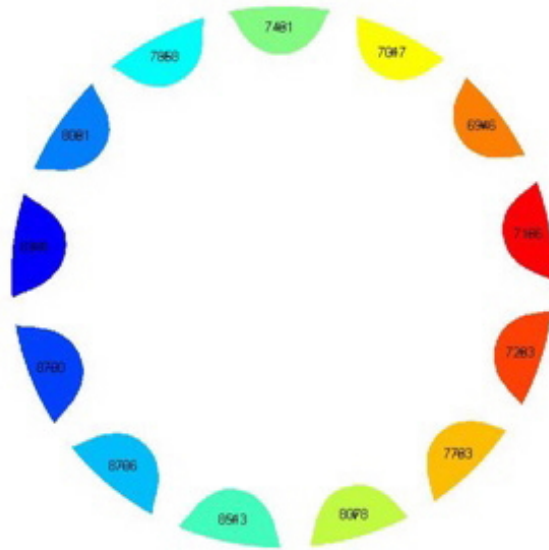


Figura 4.12: Áreas que definen los huecos de los dientes

del mismo, por lo tanto si el tamaño del alto y ancho del boundingbox es similar, es probable que la línea sea un círculo

2. La curva que nos interesa analizar es la segunda en tamaño, ése también es un parámetro que nos ayuda a no asegurar que la línea localizada es un círculo, y la que debemos analizar
3. Utilizamos las ecuaciones con las que analizamos los radios internos, los resultados se muestran en la Figura 4.13

Finalmente se grafican los resultados y se establece la tolerancia que requiere la pieza para ser aprobada o rechazada

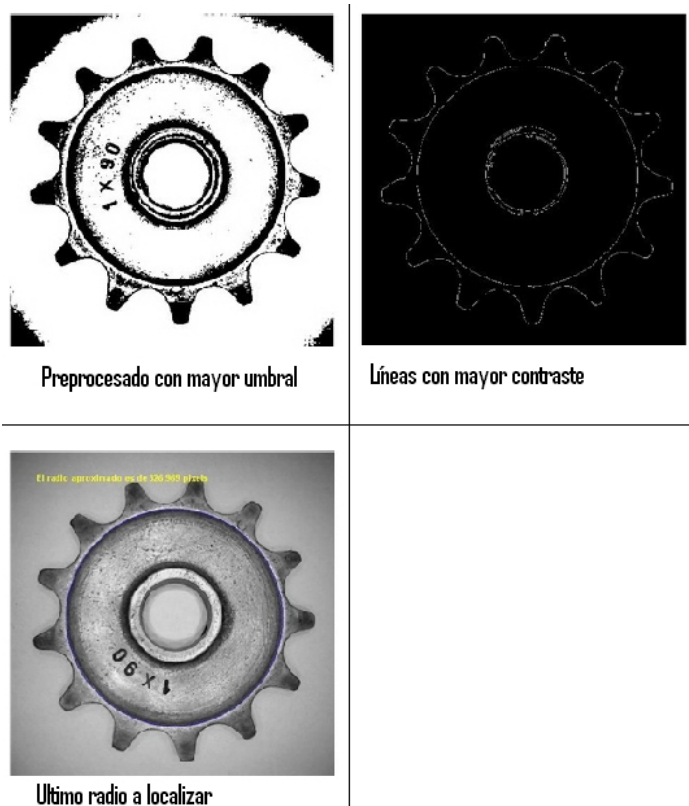


Figura 4.13: Procesado de la figura para ubicar el último radio

Capítulo 5

Pruebas y análisis

Las pruebas se realizaron utilizando una cámara web marca Boyaq modelo IRIS que tiene las siguientes características:

- Webcam USB con microfono integrado
- Resolución máxima 1.3 Megapixeles
- Compatibilidad Plug & Play en sistemas operativos superiores al Windows XP SP2 y Mac OS 10.4.9
- Equilibrio de blanco y gestion de color automático
- 8 LEDS que iluminan en 2 diferentes intensidades
- Dirección y altura ajustables
- Micrófono integrado con Tecnología de Cancelación de Eco

El precio al momento de la compra fue de \$500.00

Utilizando el código desarrollado en Matlab a través de la interfaz gráfica de usuario, se realizaron las mediciones propuestas en la vista superior de la pieza, comenzando con una pieza patrón, cuyos valores fueron guardados en el disco duro de la computadora (C:\datos_engrane_web.mat), los cuales al abrir el programa son cargados automáticamente

para posterior comparación con la pieza a analizar, una imagen que ilustra el programa desarrollado es la figura 5.1

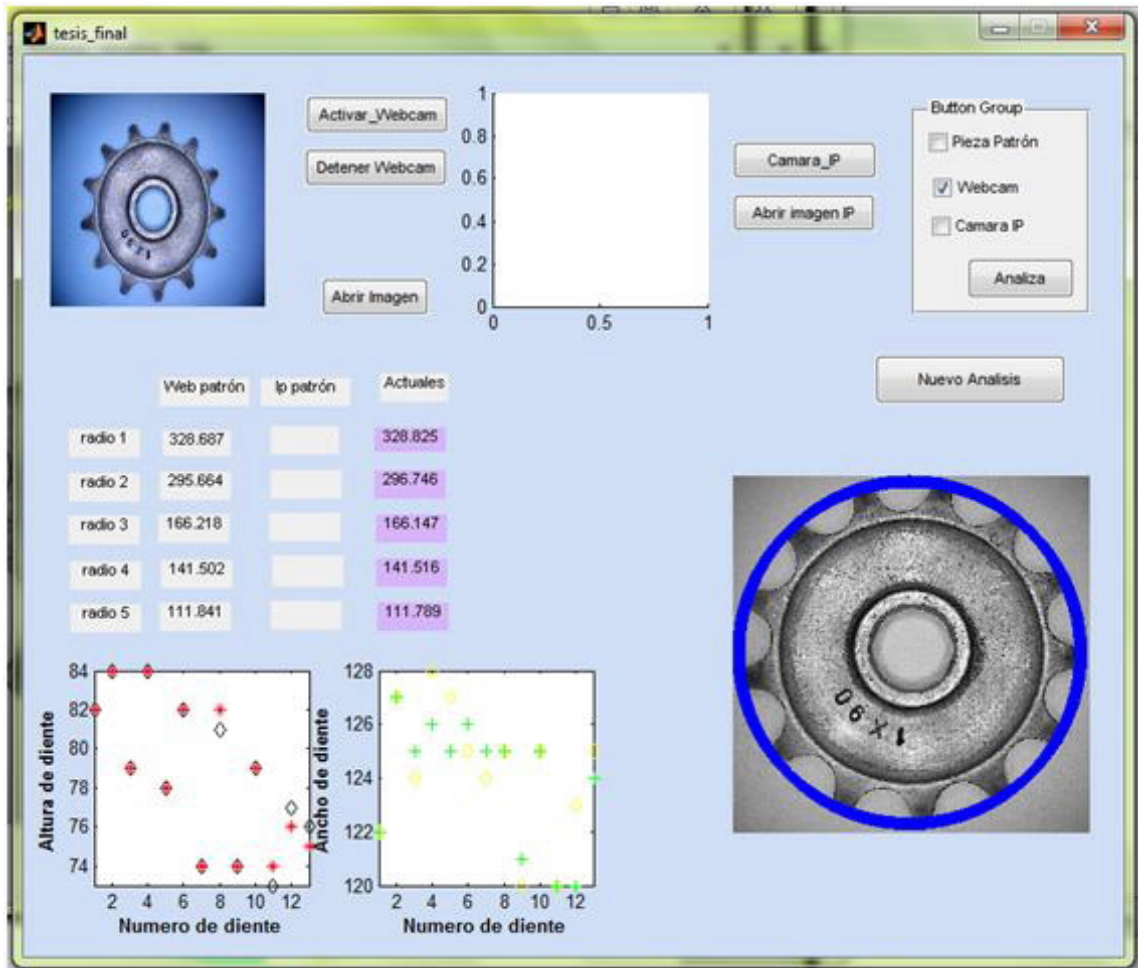


Figura 5.1: Análisis dimensional en Matlab

La Figura 5.2 muestra el resultado de aplicar el software de análisis a las piezas, y los radios que se obtienen se muestran en la Tabla 5.1. Los valores resaltados no cumplen con la tolerancia establecida previamente como ± 2 píxeles, a excepción del radio 2, el cual es un chaflán que por la naturaleza del material, sumado a las condiciones de iluminación es difícil de segmentar apropiadamente.

Los valores del alto y ancho de diente también tienen la tolerancia establecida de ± 2 píxeles, la cual satisface todas las pruebas realizadas, ver la Figura 5.2.

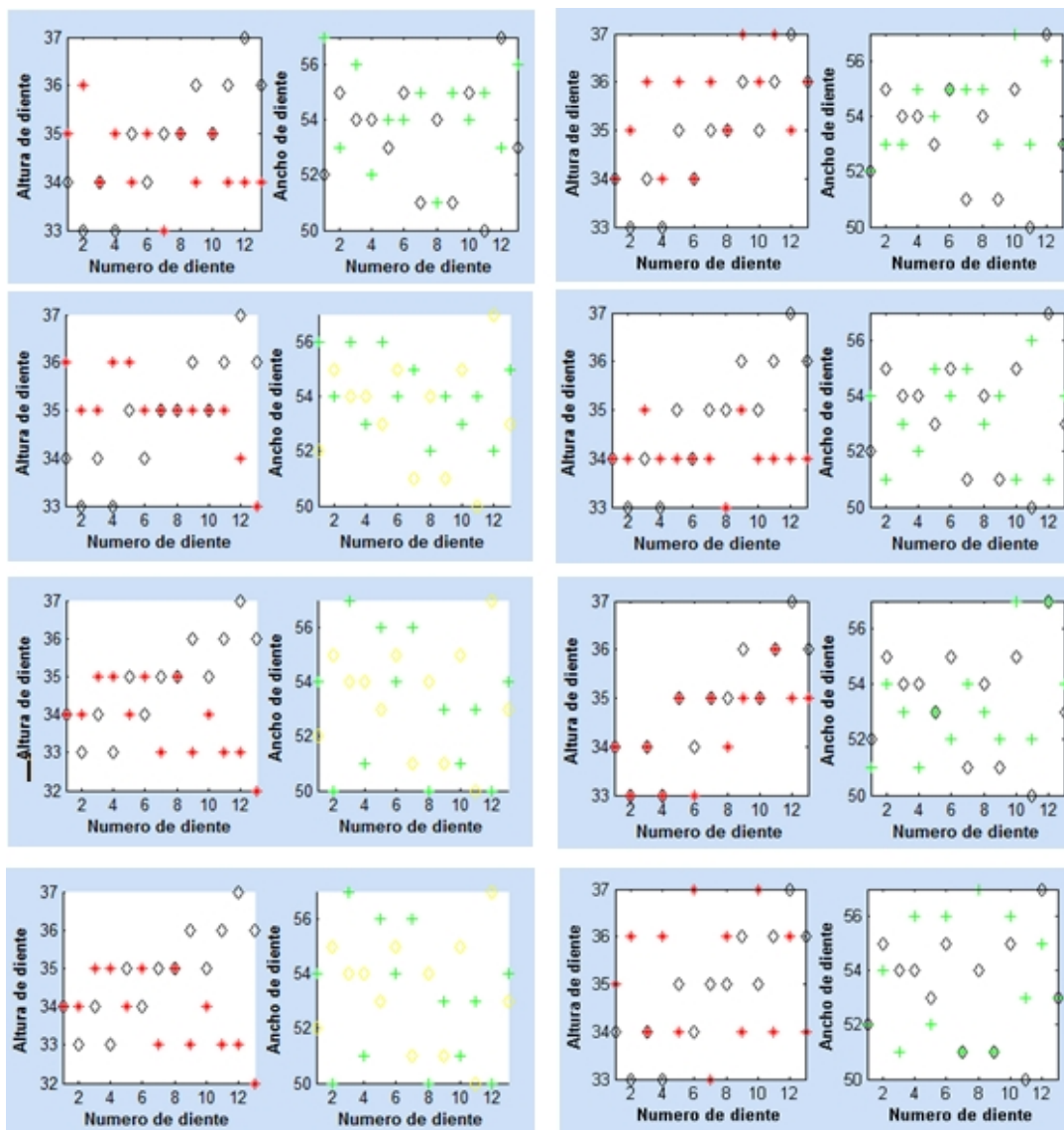


Figura 5.2: Valores de alto y ancho de diente

	radio 1	radio 2	radio 3	radio 4	radio 5
Patrón	328,687	297,664	166,218	141,502	111,841
Pieza1	329.671	299.508	166.258	141.247	111.752
Pieza2	329.968	307,28	161,88	140.6555	116,513
Pieza3	330.197	302,067	161,928	140.198	116,234
Pieza4	328.951	299.58	166.969	141.366	111.895
Pieza5	326.824	299.15	166.928	141.433	111.915
Pieza6	324,934	297.386	161,014	140.425	111.315
Pieza7	325,25	298.78	162,011	141.264	117.835
Pieza8	328.825	296.746	166.147	141.516	111.789

Tabla 5.1: Resultados de los radios calculados por la camara web

	radio 1	radio 2	radio 3	radio 4	radio 5
valor (mm)	29	27	16	12.5	9.5
pieza patrón (pixeles)	328,687	295,664	166,218	141,502	111,841
resolución (mm/pixel)	0.0882	0.0913	0.09625	0.0883	0.0849

Tabla 5.2: Resolución de la camara web

Los valores de la pieza patrón analizada con la webcam y la resolución por pixeles de las imágenes adquiridas son los mostrados en la Tabla 5.2:

Para el caso de la medición con la cámara IP marca Cognex modelo DVT530 cabe mencionar que:

- Las imágenes adquiridas son en escala de grises
- Tienen como resolución máxima un tamaño de 800×600 pixeles
- El lente es más avanzado que el de la cámara web
- Mediante la aplicación que está integrada con su driver (software intellect) se puede realizar una autocalibración del lente
- Posee iluminación propia mediante leds color rojo

	radio 1	radio 2	radio 3	radio 4	radio 5
Patrón	146.443	132.62	75.4259	63.0856	50.9489
Pieza1	146.416	132.545	75.41	62.5064	50.8969
Pieza2	148.248	130.518	77.404	63.1355	49.876
Pieza3	148.226	131.823	77.2724	63.1854	49.9212
Pieza4	146.454	132.459	75.3844	63.1163	50.8425
Pieza5	148.626	131.514	77.4188	63.2652	50.0002
Pieza6	146.466	131.462	75.4758	62.0705	51.2353
Pieza7	146.437	130.018	75.6902	62.7009	49.8659
Pieza8	146.466	131.697	75.3641	62.1406	50.6443

Tabla 5.3: Resultados de los radios calculados por la camara IP

La Figura 5.3 muestra las mediciones en las 8 piezas que se analizaron, y los resultados de las mediciones de los radios se presentan en la Tabla 5.3. Los valores resaltados en la Tabla 5.3 no cumplen con la tolerancia establecida previamente como ± 2 pixeles, a excepción del radio 2, el cual es un chaffán que por la naturaleza del material, sumado a las condiciones de iluminación es difícil de segmentar apropiadamente.

Los valores del alto y ancho de diente también tienen la tolerancia establecida de ± 2 pixeles, la cual satisface todas las pruebas realizadas.

Los valores resaltados no cumplen con la tolerancia establecida previamente como $\pm 1,25$ pixeles, ésto debido a que la resolución que muestra la Tabla 5.4 es menor al caso de la cámara web, el cual se muestra en la Tabla 5.2.

Los valores del alto y ancho de diente tienen la tolerancia establecida de ± 1 pixeles, la cual cumple todas las pruebas realizadas, ver Figura (5.3)

Los valores de la pieza patrón tomada con la cámara IP y la resolución por pixeles de las imágenes adquiridas se agrupan en la Tabla 5.4:

Analizando las Figuras 5.4 y 5.5, podemos destacar lo siguiente:

- En el análisis con la cámara web, cumplen con las tolerancias prácticamente cuatro piezas:

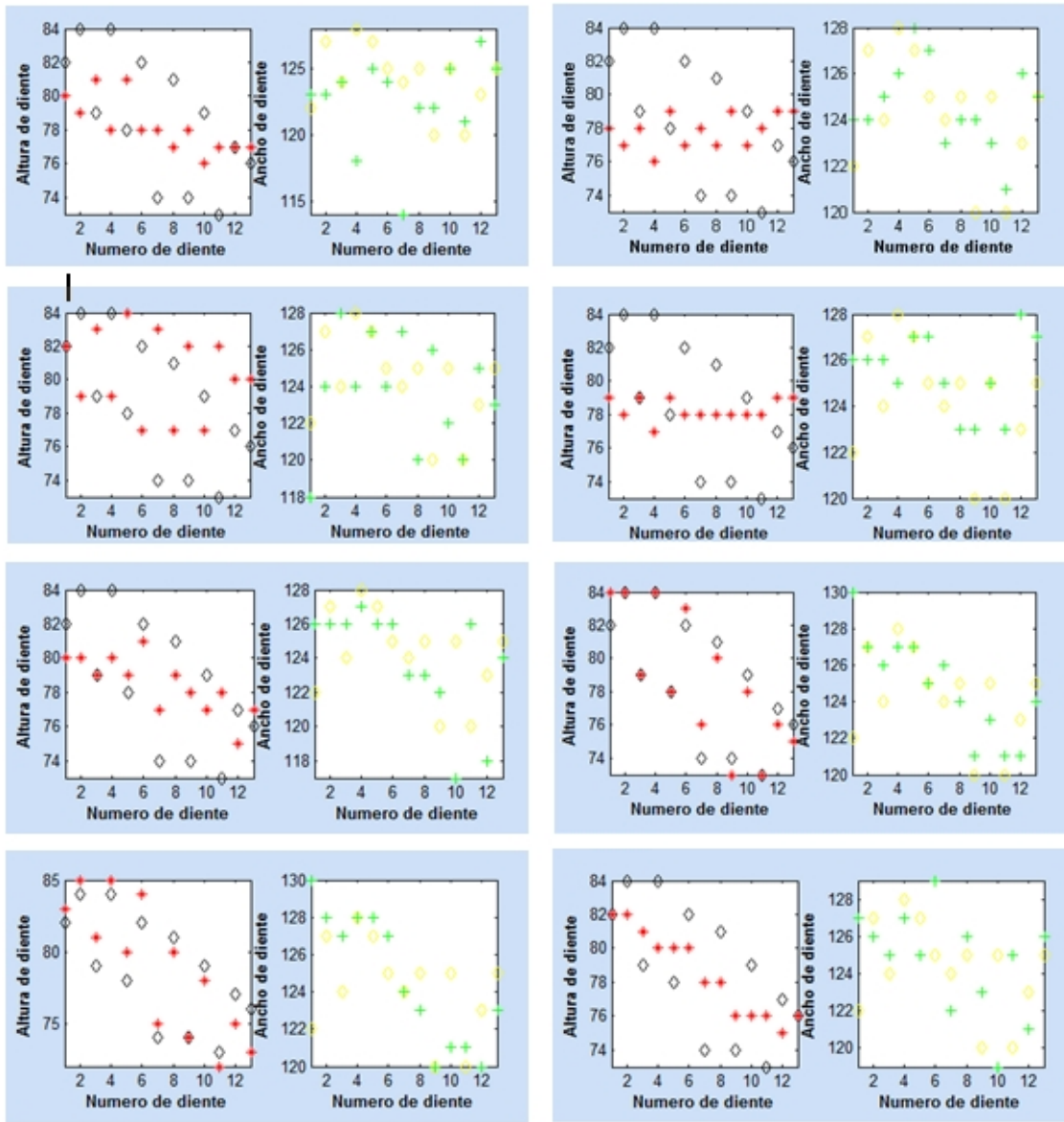


Figura 5.3: Alto y ancho de dientes encontrados por la cámara IP

	radio 1	radio 2	radio 3	radio 4	radio 5
valor (mm)	29	27	16	12.5	9.5
pieza patrón (pixeles)	146.443	132.62	75.4259	63.0856	50.9489
resolución (mm/pixel)	0.198	0.20355	0.2121	0.1981	0.18645

Tabla 5.4: Resolución de la cámara Cognex

- En el análisis con la cámara IP, cumplen las tolerancias cuatro de las ocho piezas:
- La variación máxima en los análisis con cámara web fue de cerca de 10 píxeles, los cuales equivalen a $0,93mm$, mientras que con la cámara IP fué de $-2,5$ píxeles, que equivalen a $0,4071mm$

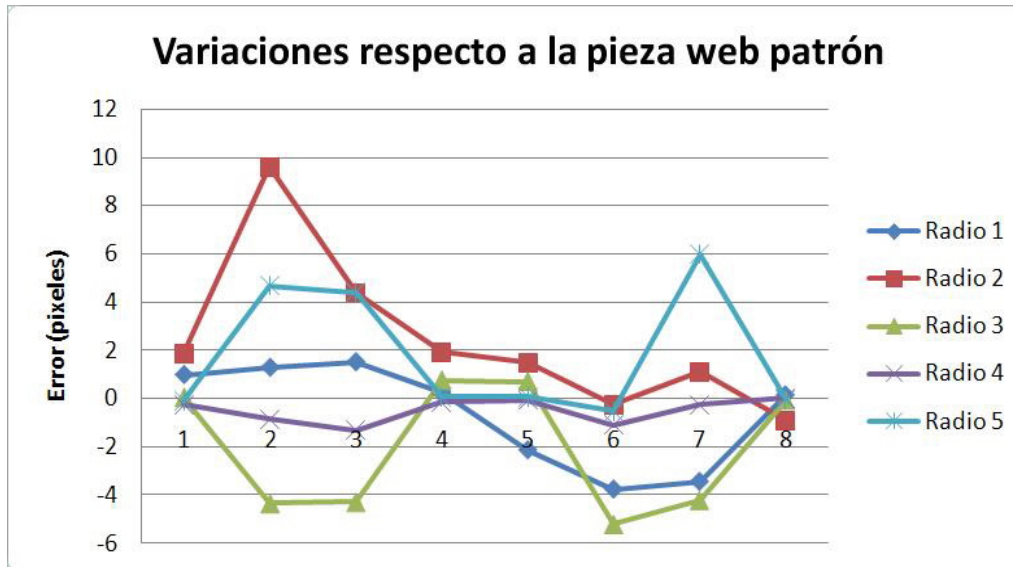


Figura 5.4: Gráfica de errores con la cámara web

5.1. Conclusiones

Durante el desarrollo de la aplicación de la verificación dimensional, se aplicaron algoritmos de mejora de la calidad de la imagen adquirida. Debido a que las pruebas se realizaron en dos cámaras diferentes, se adaptaron algunos comandos al código según la cámara utilizada, principalmente una transformación logarítmica de contraste para la cámara IP, debido a que la cámara web al manejar escala de colores RGB, facilitaba la segmentación de las piezas a analizar.

La adquisición de la imagen a través de la cámara IP, presenta un mayor grado de dificultad pues se requiere noción básica de configuración redes de área local, así como manejar algún programa que permita utilizar controles ActiveX. Matlab maneja controles

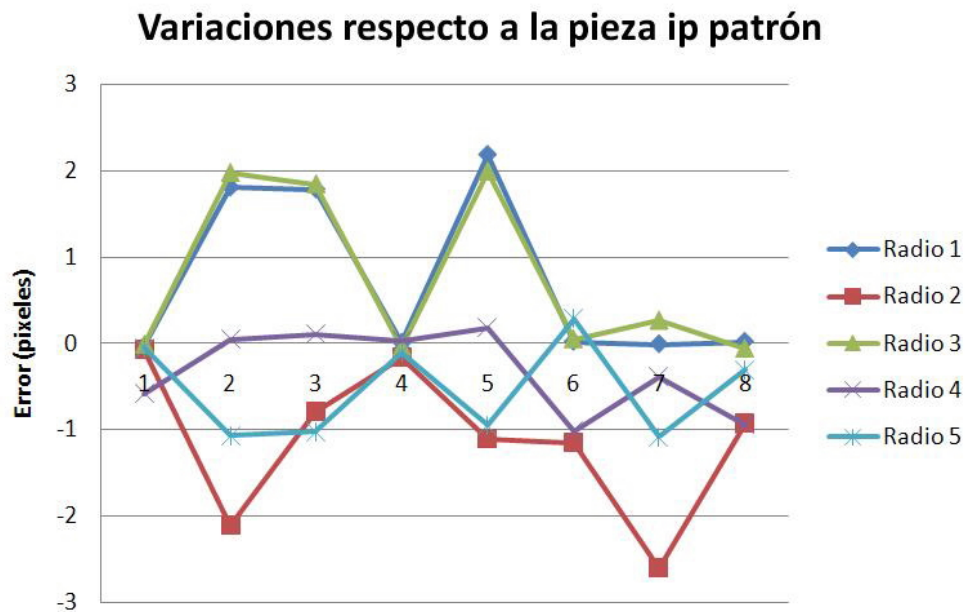


Figura 5.5: Gráfica de errores con la cámara IP

ActiveX, sin embargo tiene algunas carencias en el desarrollo de éste tipo de aplicaciones, el control ActiveX proporcionado por Cognex, está desarrollado para ser usado con software de entorno Visual Basic, el cual maneja básicamente tres tipos de instrucciones, eventos, métodos, y propiedades, y en lo personal sólo encontré la forma de manejar métodos y propiedades dentro del entorno de Matlab, es por ésta razón que la adquisición de la imagen con la cámara IP, se realiza mediante un archivo ejecutable desarrollado en Visual Basic 6, el cual guarda la imagen en una dirección específica de la computadora, para ser procesada posteriormente en Matlab.

En cuanto a la velocidad de procesamiento de la imagen en comparativa con ambas cámaras, se puede asegurar que no existe diferencia destacable debido a que, lo que determina la velocidad del procesamiento de la imagen es su tamaño, la cámara IP tuvo menor resolución que la cámara web, por lo tanto fué más rápido el procesamiento de la cámara IP, pero tuvo como consecuencia un menor tamaño en la captura de la imagen de la pieza a analizar.

La cantidad de piezas que no cumplen las tolerancias establecidas, es la misma en ambas cámaras, por lo que se puede concluir que ambas se pueden utilizar para realizar análisis dimensional, con un grado de confiabilidad que dependerá de las habilidades del programador,

y el diseño del espacio de trabajo e iluminación adecuados para la aplicación.

Matlab presenta ventajas en el procesamiento de imágenes, pues es un software que maneja vectores como unidad básica de información, las funciones que contiene en su toolbox de procesamiento digital de imágenes es interesante, sin embargo está un poco limitado respecto al hardware con el cual se puede comunicar, aún en la versión 7.8.0.347 (R2009a) sólo puede adquirir imágenes o video desde una sola cámara a la vez, lo cual puede ser una desventaja si se deseara realizar una aplicación para reconstrucción 3D de imágenes.

Aunque la verificación dimensional fue satisfactoria, las mediciones en tiempo real son aplicables sólo para imágenes con una resolución menor a 640×480 pixeles, y con necesidades de información que no sean muy complejas, ésto es presencia o ausencia, completo o incompleto, etc. características de tipo binario principalmente.

La tecnología de las cámaras web está avanzando de una forma muy rápida, lo cual permite ampliar la gama de aplicaciones que pueden realizarse con ellas, desde una adquisición de una fotografía personal, pasando por la video conferencia a través de internet, hasta su uso para seguridad.

Como se estableció en los objetivos, se demuestra que se puede desarrollar una aplicación que puede realizar verificación dimensional con sólo una cámara web, alguna fuente de iluminación que mantenga adecuada la luminancia del lugar y el objeto a analizar, y una computadora personal, el software Matlab es sólo necesario para el desarrollador debido a que se puede empaquetar el programa, de tal forma que el usuario final sólo ejecute un fichero y se utilice la aplicación desarrollada.

5.2. Trabajo Futuro

Las aplicaciones de la visión artificial son muy diversas y tan complejas como se deseen. En el presente trabajo se realizó la verificación dimensional de una pieza utilizando un patrón como base del análisis, un tema interesante de abordar podría ser que la medición se realizara sin la necesidad de una pieza patrón, analizando dimensiones en pixeles y transformándolas a mediciones lineales del mundo real.

Dentro del mismo campo de aplicación de la Metrología dimensional es interesante trabajar en una reconstrucción tridimensional de piezas a través de 3 o más imágenes, éste tema puede ser desarrollado utilizando una cámara móvil o tres cámaras que integren algoritmos de identificación de esquinas y una posterior relación de esos pixeles para la construcción en tercera dimensión de la pieza a analizar

En el campo de la robótica industrial, el presente trabajo puede ser de utilidad cuando se investigue sobre generación de trayectorias, las cuales pueden ser de utilidad al definir un espacio para aplicar soldadura o pintura a una pieza determinada. También relativo al tema de la robótica se puede desarrollar una aplicación que, a partir de la imagen de un objeto, se pueda posicionar y orientar el efector final de un manipulador, con el fin de que se reduzcan los tiempos de programación del mismo.

Otro tema de investigación aplicable al impacto ambiental que genera la basura podría ser la identificación de botellas de plástico, latas de aluminio o algún producto de interés para ser recolectado de una línea de desperdicios en algún centro de reciclaje.

Una aplicación interesante que puede ayudar al desarrollo de personas con capacidades diferentes es, el desarrollo de algún prototipo que pueda identificar caracteres escritos en un libro, para posteriormente convertirlos en audio y que de ésta forma las personas invidentes, cuenten con una variedad literaria más amplia que la que se les ofrece en lenguaje braille.

Como se mencionó previamente, la visión es el sentido más complejo que poseemos, prácticamente todas las actividades que realizamos en la vida cotidiana están muy relacionadas con la perspectiva que tenemos del mundo, es por ello que la cantidad de investigaciones que pueden ser desarrolladas, es tan diverso como nuestro propio pensamiento.

Bibliografía

- [Sahoo88] Sahoo, P.K., A survey of Thresholding Techniques, Computer Vision, Graphics and Image processing, 41, pp. 233-260, 1988.
- [Gonzales-Woods] Digital Image Processing Second Edition, Rafael C. Gonzalez-Richard E. Woods
- [Revista DYNA] Iñaki Goiricelaya Ordorika, Introducción a la visión artificial Volumen 61, pp 29-33, Diciembre 1986
- [Castleman79] Castleman K.R., Digital Image Processing Prentice-Hall, Inc., 1979
- [Duda73] Duda R. and Hart P., Pattern Classification and Scene Analysis Wiley-Interscience, 1973
- [Fu82] Fu K.S., Syntactic Pattern Recognition and Applications. Prentice-Hall, Inc., 1982.
- [Gonzales79] Gonzalez R.C., Wintz P., Digital Image Processing. Addison-Wesley, 1979.
- [Herman82] Herman G., Image Reconstruction from Projections. Academic Press, 1980.
- [Marr82] Marr D., Vision - A Computational Investigation into the Human Representation and Processing of Visual Information. Freeman, 1982.
- [Pratt78] Pratt W., Digital Image Processing. Wiley-Interscience, 1978.
- [Rosenfeld82] Rosenfeld A., Kak A.C., Digital Picture Processing (V.I y II). Academic Press, 1982.

- [Serra82] Serra J., Image Analysis and Mathematical Morphology. Academic Press, Inc., 1982.
- [Besag86] Besag J., Dirty Pictures. Royal Statistical Society B, 1986
- [Cross83] Cross G. and Jain A.K., Markov Random Fields texture Models. IEEE Trans. on PAMI, Enero-83, pp 25-39.