

# SISTEMA PARA DETECCIÓN Y RECONOCIMIENTO DE IRREGULARIDADES SOBRE CIRCUITOS VIALES UTILIZANDO EL SENSOR KINECT



Autores: Huincalef Rodrigo A. , Urrutia Guillermo G.

Director de tesina: Ingravallo Gabriel

Tesina presentada a la Facultad de Ingeniería de la Universidad Nacional  
de la Patagonia San Juan Bosco como parte de los requisitos para la  
obtención del título de Licenciado en Sistemas

Facultad de ingeniería - Sede Trelew

UNPSJB - Universidad Nacional de la Patagonia San Juan Bosco

12 De Diciembre Del 2017

---

## **Organización de los contenidos**

---

### **Capítulo 1. Introducción**

En este capítulo se introducen los objetivos de la tesina, el marco teórico y los desarrollos propuestos.

### **Capítulo 2. Antecedentes de software para la gestión de fallas viales**

Este capítulo comienza definiendo el concepto de falla, los tipos de material, tipos de reparación que se pueden efectuar sobre cada uno, y los tipos de falla que se corresponden sobre los distintos tipos de material. Durante este capítulo también se debaten antecedentes históricos, software y proyectos de investigación, relativos al sensado de fallas en distintos contextos.

### **Capítulo 3. Dispositivos hardware y herramientas software para el sensado de fallas**

El objetivo de este capítulo es enumerar los distintos tipos de dispositivos y sensores que pueden utilizarse para realizar la captura de irregularidades viales, sus principales características y funcionamiento. Por otro lado, se expondrán librerías en distintos lenguajes de programación, y métodos de almacenamiento de las fallas empleados por los mismos.

### **Capítulo 4. Técnicas de reconocimiento y procesamiento de fallas**

Durante el desarrollo de este capítulo se definirá el concepto de Machine Learning, sus usos, contextos de aplicación y algoritmos que brinda para el reconocimiento de patrones en conjuntos datos. Posteriormente, se expondrán aquellas técnicas empleadas para realizar la identificación, el saneamiento y la clasificación de los datos de una nube de puntos con datos de una falla.

### **Capítulo 5. Herramientas GPS y Geocoding**

El objetivo de este capítulo consiste en explicar el concepto de GPS, latitud, longitud, funcionamiento, dispositivos GPS disponibles y utilidades software para compartir conexiones GPS entre sistemas operativos Android y Linux. Además, se explicará el concepto de Geocoding y Reverse Geocoding, librerías disponibles en PHP y Python y, servicios web ofrecidos por terceros compatibles con éstas.

---

## **Capítulo 6. Caso de aplicación**

El objetivo de este capítulo es explicar tanto la arquitectura general de la aplicación cliente como de la aplicación web, los objetivos de las mismas, el ciclo de trabajo y las formas de interacción con ambas aplicaciones. Por un lado, se expondrá el funcionamiento de la aplicación cliente, librerías empleadas para su desarrollo, formas de interacción de ésta con el sensor Kinect y diseño. Por otro lado, se presentará la funcionalidad base que la aplicación web ofrecía anteriormente, diseño, la funcionalidad incluida como parte del desarrollo de la presente investigación y librerías relacionadas al funcionamiento agregado a la misma.

## **Capítulo 7. Conclusiones y líneas futuras**

En este capítulo, se explican las conclusiones de la investigación realizada y posibles líneas de investigación futuras con el presente desarrollo.

---

## Índice:

---

<b>Abstract</b>	<b>1</b>
<b>Reconocimientos</b>	<b>2</b>
<b>Resumen</b>	<b>3</b>
1    Resumen . . . . .	3
<b>Introducción</b>	<b>4</b>
1    Objetivo general . . . . .	4
2    Objetivos específicos . . . . .	4
3    Motivación . . . . .	4
4    Marco teórico . . . . .	5
5    Desarrollos propuestos . . . . .	6
6    Resultados esperados . . . . .	6
7    Cronograma de actividades . . . . .	6
<b>Antecedentes de software para la gestión de fallas viales</b>	<b>8</b>
1    Pavimento . . . . .	8
2    Tipos de juntas . . . . .	9
3    Tipos de fallas sobre pavimentos rígidos y flexibles	10
3.1    Fallas sobre pavimentos rígidos . . . . .	10
3.2    Fallas sobre pavimentos flexibles . . . . .	13
4    Proyectos de software anteriores para la detección de fallas sobre el pavimento . . . . .	16
4.1    Proyectos basados en procesamiento de video e imagen . . . . .	16
4.1.1    Conceptos de procesamiento de imágenes . . . . .	16
4.1.2    Estudios relativos a la detección de fallas . . . . .	66
4.2    Proyectos basados en sensores de vibración . . . . .	72
4.3    Proyectos basados en la reconstrucción 3D . . . . .	74
<b>Bibliografía</b>	<b>77</b>

---

## Abstract

---

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

---

## Reconocimientos

---

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus et augue vel nibh pharetra cursus et vel mauris. Integer a ante in leo fermentum tincidunt quis vitae mauris. In vitae bibendum lectus. Curabitur porttitor, sapien et tincidunt suscipit, nisl ligula euismod nisi, ut suscipit dui metus vel eros. Nullam consectetur erat quam, in sollicitudin lacus lobortis eget. Proin felis diam, pellentesque eu finibus dapibus, congue a ligula. Donec gravida quam ante, ac rutrum mauris pharetra ac. Interdum et malesuada fames ac ante ipsum primis in faucibus. Curabitur malesuada metus nec gravida sollicitudin. Morbi a semper ex. Sed sollicitudin lobortis dolor quis mattis. Sed vel feugiat erat.

Quisque dignissim ligula ligula, id fermentum dui imperdiet ac. Quisque pellentesque lectus dolor, id ultrices nisi porttitor sed. Maecenas quam neque, vestibulum in libero nec, vestibulum venenatis mauris. Aliquam nec orci dictum, efficitur tortor id, lacinia arcu. Phasellus tincidunt tristique dui, pretium convallis est luctus nec. Integer molestie euismod dui, vel pharetra purus egestas vitae. Nullam eu leo est. Integer euismod sodales feugiat. Ut ac scelerisque est.

Fusce auctor ex ut tellus sollicitudin, vel placerat purus auctor. Cras ut dolor iaculis, lobortis felis dignissim, vehicula elit. Ut malesuada lectus ligula, quis sagittis erat consectetur vel. Etiam tempor tellus sit amet varius tempor. Mauris lacinia mi et eros bibendum lacinia. Ut blandit porttitor dictum. Sed pellentesque diam ut tortor faucibus, vel vestibulum metus congue. Sed sed purus sem. Sed sagittis maximus enim, id consequat mi convallis ut.

---

# Resumen

---

## 1. Resumen

El objetivo de la presente tesina consiste en incorporar a un sistema de detección de fallas, funcionalidad para el sensado tridimensional de fallas viales, exponiendo software disponible para registro de distintos tipos de fallas sobre senderos viales, dispositivos hardware y herramientas de software asociadas al sensado de las mismas, y técnicas para el reconocimiento y clasificación de las muestras de fallas capturadas.

Con respecto al software para el registro de fallas sobre senderos viales, se expondrá de manera breve la terminología relacionada a los distintos tipos de fallas, y desarrollos software relacionados con el sensado y registro de fallas en sistemas de información.

En cuanto a dispositivos hardware, se estudiarán distintos tipos de sensores para el relevamiento de superficies, características principales y funcionamiento. Asimismo, se investigarán los lenguajes de programación disponibles, librerías, herramientas y disponibilidad en distintos sistemas operativos. Adicionalmente, se incluirán las utilidades de software para obtención de latitud y longitud con un dispositivo compatible con GPS, y las librerías web que permiten obtener datos de la dirección estimada asociada a dichas coordenadas.

Además, se expondrán las técnicas de filtrado y clasificación empleadas para el análisis de distintos tipos de falla, antecedentes históricos, y proyectos de similares características.

Finalmente, se realizarán modificaciones a una aplicación web previamente desarrollada, con el fin de incluir la visualización de distintos tipos de falla en un navegador, el filtrado de distintos tipos de fallas sobre una calle determinada, y la estimación de la dirección aproximada de una falla, basándose en la latitud y longitud. Además, se realizará el desarrollo de una aplicación cliente que permita la captura de distintos tipos de fallas por medio del sensor Kinect. Ésta será capaz de interactuar con la aplicación web, para obtener información respecto de la ubicación de las fallas registradas anteriormente por sus usuarios, lo que le permitirá asociar archivos de capturas a las mismas. También, la aplicación cliente permitirá registrar en la aplicación web aquellas fallas descubiertas durante un recorrido (sin registro previo), obteniendo la latitud y longitud de la falla por medio de un GPS, para luego realizar una estimación de la dirección en la aplicación web. Posteriormente, se podrá realizar una selección de las todas las fallas capturadas y realizar la subida de las mismas con sus archivos de capturas al servidor, desde la aplicación cliente. Por último, se desarrollará un script separado, que utilizará los filtros y mecanismos de machine learning ofrecidos por la librería Point Cloud Library (PCL), para el aislamiento y la clasificación de los tipos de fallas “Bache” y “Grieta”.

---

## Introducción

---

### 1. Objetivo general

Extender un sistema de notificación y administración de información pertinente a irregularidades en superficies viales, por medio del censado visual tridimensional semiautomático de información vial, con el fin de contribuir a la toma de decisiones respecto del estado de los circuitos viales.

### 2. Objetivos específicos

- Desarrollar un servicio web para el almacenamiento de capturas realizadas por el sensor desde el aplicativo de captura.
- Procesar la información tridimensional contenida en un archivo con el fin de identificar las diferentes propiedades y clasificar el tipo de falla capturada.
- Implementar una aplicación que permita la captura tridimensional, en un formato especificado por el sensor Kinect, y su posterior carga por medio del servicio web.
- Integrar el servicio web de almacenamiento de capturas a la aplicación web de gestión de fallas para su visualización en línea.

### 3. Motivación

El sensor Kinect desarrollado por Microsoft, es un sensor de detección de movimiento que permite la interacción usuario-dispositivo sin la necesidad de controles adicionales. Este sensor, ofrece una amplia gama de características entre las que se destacan una cámara RGB de 640x480 px a 30 FPS, un emisor y un sensor infrarrojos, que le permiten obtener información acerca de la profundidad, capacidades de reconocimiento de voz y reconocimiento facial. Esta amplia variedad de características, sumadas al bajo costo de adquisición, y la disponibilidad de paquetes de código abierto disponibles, hacen que este dispositivo sea apropiado para la explotación en innovaciones.

Las nuevas tecnologías web y la amplia variedad de dispositivos y conectividad actual, hacen de Internet una plataforma computacional con una gran incidencia en la operación y acceso a la información

por parte de las personas. Estas características, sumadas al crecimiento en el uso de estas tecnologías en dispositivos móviles actuales, permiten aumentar la facilidad con la que los usuarios pueden acceder e informar de manera intuitiva, acerca de problemas relacionados a irregularidades en las calles de la ciudad.

La librería Point Cloud Library (PCL), es una herramienta libre, potente y con un gran conjunto de algoritmos para el procesamiento de imágenes y nubes de puntos en 2D/3D. Esta librería reúne investigadores, universidades, compañías e individuos de todo el mundo y se está convirtiendo rápidamente en una referencia para cualquier interesado en el procesamiento 3D y robótica.

El núcleo de PCL está estructurado en pequeñas librerías y algoritmos para áreas específicas del procesamiento 3D que pueden ser combinadas para resolver problemas como el reconocimiento de objetos, registro de nubes de puntos, segmentación y reconstrucción de superficies facilitando la reutilización de código.

Las irregularidades son una característica común en las rutas y calles pavimentadas de una ciudad lo que genera problemas en el flujo del tráfico y puede ocasionar situaciones de conducción potencialmente peligrosas. Debido a la constante degradación de los diferentes tipos de fallas, estas deben ser detectadas y reparadas tan pronto como sea posible. Existen diferentes formas para la detección de la ubicación y las propiedades de las irregularidades como forma, tamaño, profundidad y volumen, lo que juega un rol importante con respecto a la reparación de las mismas.

La reciente introducción de programación de aplicaciones web en la carrera, requirió introducir personal especializado de la Universidad Nacional del Sur (UNS), y por medio de las prácticas realizadas en esta asignatura, se inició con el desarrollo de un trabajo final que inicialmente georeferenciaba fallas. Ésto dio pie a un proyecto de investigación dirigido por Gabriel Ingravallo (UNPSJB Trelew), y asesorado por Diego C. Martínez (UNS Bahía Blanca) para la recolección y análisis de fallas viales y que hacía uso del sensor Kinect, del que participaron desde su inicio los presentes tesis. Sin embargo, el proyecto no cuenta con un mecanismo de análisis y clasificación de diferentes tipos de fallas.

La falta de innovación con respecto al registro y análisis del estado de las vías de circulación de la ciudad y la necesidad de un mecanismo de análisis y clasificación de distintos tipos de fallas en el proyecto de investigación, conforman la principal motivación de esta tesis, que busca extender la funcionalidad de éste, agregando la visualización de las dimensiones de la falla, incluyendo el procesamiento de nubes de puntos para las irregularidades, la clasificación y extracción de sus propiedades relevantes.

## 4. Marco teórico

A través del tiempo, la tarea de registrar los defectos y depresiones en el pavimento siempre ha sido crucial para adoptar una estrategia precisa para el mantenimiento y las tareas de reparación de los senderos viales y, debido a que la medición manual es una tarea costosa en tiempo y recursos, se ha experimentado con diversas tecnologías con el fin de automatizar y reducir este costo. En el rango de estas tecnologías, se han considerado tanto sensores de medición de diferentes características inherentes a las irregularidades viales, como así también tecnologías para el geoposicionamiento y el registro y administración fallas por medio de sistemas web.

Entre las aproximaciones de detección y análisis de fallas, se abarcan la investigación y experimentación con imágenes y videos 3D, empleando técnicas de procesamiento de imágenes, con el fin de aislar las fallas basadas únicamente en la diferencia de niveles color en pixeles y/o textura que las fallas suelen poseer, respecto de los pixeles del sendero vial donde se encuentran. Alternativamente, se ha experimentado con sistemas que emplean sensores de vibración, tales como acelerómetros, que ubicados precisamente en uno o varios lugares de un vehículo, intentan detectar las diferencias de vibración abruptas, que se encuentran en un rango propio de las fallas. Otra aproximación, consiste en emplear sensores para medir las características espaciales de las fallas viales, y luego realizar un procesamiento tridimensional de las coordenadas capturadas aplicando diversos algoritmos, de manera que se puedan detectar diferencias de nivel entre puntos del plano y de la falla, y así lograr aislar la falla del plano. Cada una de las aproximaciones descriptas anteriormente cuentan con ventajas como así también con desventajas, siendo cada una sensible a diferentes factores externos en el entorno en donde se realiza la medición.

## 5. Desarrollos propuestos

Se propone el desarrollo de una aplicación que haciendo uso de las tecnologías web actuales, permitirá a los actores involucrados en el proceso de mantenimiento de vías de circulación recolectar, procesar y visualizar los diferentes tipos de fallas sobre las distintas calles de la ciudad. Así como, permitir a los habitantes de la ciudad informar y mantenerse informados respecto del estado de las calles.

Con respecto al proceso de recolección se propone implementar un pequeño aplicativo desarrollado en Python, que junto al sensor Kinect, posibilitará almacenar la nube de puntos censada (que representa una porción de un sendero vial) por el dispositivo. En cuanto al procesamiento, se investigará y experimentará con las técnicas ofrecidas por la librería PCL, que contribuyan a la identificación y delimitación de un tipo de irregularidad dentro de la captura realizada. Una vez identificado el método más conveniente, se desarrollarán varios módulos en C++, interconectados que permitirán la identificación y clasificación del tipo de falla y la extracción de sus propiedades.

Por último, con respecto a los mecanismos de notificación y visualización, se hará uso de distintas tecnologías web del lado del cliente (Javascript, CSS3, HTML5) que darán el formato a las herramientas ofrecidas para visualizar las fallas, y operará en conjunto con framework PHP (CodeIgniter 2) del lado del servidor.

## 6. Resultados esperados

Con el desarrollo de la presente tesisina, se espera obtener una plataforma que permita mantener informados a los habitantes locales con respecto al estado de las calles de la ciudad a través del acceso a una aplicación web que indique la ubicación de las fallas y el estado actual en el que se encuentran las mismas. Esta a su vez, permitirá a los habitantes informar la localización falla sobre una calle determinada.

Por otro lado, se producirá un módulo de software, que mediante el procesamiento de nubes de puntos tridimensionales posibilite la obtención de una medida objetiva que caracterice distintos tipos de falla. Para ello, se investigará acerca de las estructuras ofrecidas con la librería PCL, y se experimentará con el sensor Kinect, para lograr comprender la estructura interna de una captura realizada por el dispositivo y el funcionamiento de los algoritmos que procesan dicha captura (algoritmos que eliminan el ruido y filtran dicha captura).

## 7. Cronograma de actividades

Las actividades que se efectuarán para cumplir con el objetivo de la tesisina se detallan enumeradas junto a su número de tarea:

1. Documentar la información técnica durante el desempeño de las tareas.
2. Investigar acerca del funcionamiento del sensor Kinect y los distintos módulos ofrecidos por su SDK bajo Linux.
3. Investigar la librería de procesamiento PCL.
4. Investigar acerca métodos de detección de fallas sobre vías de circulación.
5. Investigar sobre la utilización del sensor Kinect para detección de objetos.
6. Experimentar con la librería PCL y el sensor Kinect acerca de distintos métodos de procesamiento y análisis de nubes de puntos tridimensionales.
7. Desarrollar un módulo en Python para la captura de nubes de puntos.
8. Documentar los pasos requeridos para sanear y detectar una falla en el archivo de nube de puntos.
9. Documentar las técnicas requeridas para delimitar y clasificar un tipo de falla a partir de una nube de puntos.

10. Desarrollar módulos en C++ para el análisis de fallas (el saneamiento, la detección, delimitación y clasificación de los tipos de fallas).
11. Investigar el renderizado de archivos de nubes de puntos con WebGL.
12. Integrar la funcionalidad de la aplicación web con los módulos de procesamiento de nubes de puntos desarrollados.
13. Defensa de la tesina ante mesa examinadora.

## Antecedentes de software para la gestión de fallas viales

### 1. Pavimento

El pavimento de una calle o autopista, es una estructura compuesta de un conjunto de capas de materiales procesados sobre el suelo, cuya función consiste en distribuir las cargas de los vehículos al sub-suelo y permitir el tránsito de los mismos. La estructura del pavimento debería proveer una superficie de calidad aceptable para la circulación de vehículos, resistencia adecuada al resbalamiento, evitar la contaminación de ruido producto de la circulación de los vehículos, una superficie impermeable, de manera que el suelo que esta debajo de las capas de material este bien protegido, resistencia estructural (con el fin de soportar todo tipo de fuerza aplicada sobre él) y un diseño con un ciclo de vida prolongado y bajo costo de mantenimiento.

Los pavimentos se pueden clasificar en dos tipos diferentes:

- Pavimento de rígido: Es una estructura compuesta por losas de hormigón cuya resistencia a la flexión es relativamente elevada. Los pavimentos rígidos se integran por una losa de concreto de cemento portland, que se encuentra situada por encima de una capa base de grava, y ésta a la vez descansa en una capa de suelo compactado, denominada subrasante.

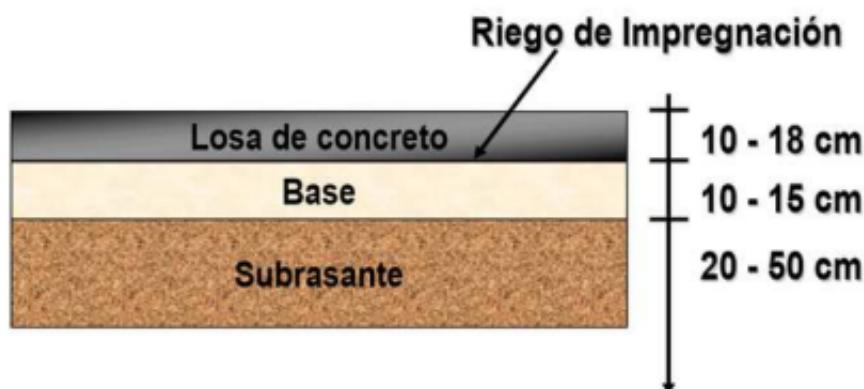


Figura 5.1: Estructura del pavimento rígido

- Pavimento flexible: Es una estructura compuesta por capas donde uno de los materiales presentes es el asfalto, lo que permite la deflexión (la deformación que, material como producto de una fuerza externa) bajo las cargas. Los pavimentos flexibles se componen de una capa de mezcla asfáltica u hormigón asfáltico, que consiste en un agregado de asfalto y materiales minerales (como áridos) compactados y extendidos. Esta se expone a las condiciones más severas debido al clima y tráfico, una capa base que se compone de materiales áridos (conjunto de materiales obtenidos de la fragmentación de rocas y arenas, tales como la grava, la gravilla y la arena), una capa de sub-base con materiales de calidad inferior a los empleados en la capa base.



Figura 5.2: Estructura de pavimento flexible

## 2. Tipos de juntas

Los senderos viales, tales como calles, autopistas, se encuentran sometidos a diversos cambios de temperatura durante el día, a diferencias de temperatura considerables entre las distintas estaciones del año, a cambios de humedad constante, y durante su construcción, a los tiempos de parada debido a las jornadas de trabajo con horas fijas, lo que provoca que el material sufra dilataciones y contracciones y, si éste se encuentra extendido en una porción considerable de terreno, se producirá el agrietamiento aleatorio e irregular del mismo, a corto plazo. Para evitar ésto, se crean juntas en el pavimento, que son mecanismos de control de fisuras, que consisten en cortes realizados con maquinaria especial a lo largo y ancho del material, a una profundidad establecida según las propiedades de éste y de la calle, que lo divide en losas. Estas divisiones, se realizan a una distancia aproximada a la que aparecerían las fisuras, ya que en caso contrario, se fisurarían aquellas losas con dimensiones excesivas.

En base a la posición de las juntas respecto del avance del hormigonado, las juntas en el pavimento se pueden clasificar en juntas longitudinales, que son aquellas paralelas a dicho avance, o juntas transversales, que son aquellas perpendiculares según el avance del mismo. Por otro lado, las juntas se pueden clasificar respecto del tipo de función que cumplen como:

- Juntas de contracción: Pueden ser juntas tanto transversales como longitudinales, y su fin es limitar el tamaño de las losas, disminuyendo hasta valores aceptables, el nivel de tensión producida por fenómenos de retracción como por la variación de temperatura por distancia (gradiente térmico), de manera de evitar roturas en el material.
- Juntas de construcción: Son las juntas que se producen entre bandas del pavimento, o también en una misma banda, entre losas que se encuentran de manera contigua que se construyen con un tiempo de parada prolongado.
- Juntas de dilatación: Son las juntas que se realizan con el fin de absorber las expansiones del material como consecuencia de los aumentos de temperatura, evitando empujes sobre éste que podrían producir la rotura del mismo. En este tipo de junta, se emplea un material compresible

(madera, poliestireno expandido, láminas de poliuretano, etc.) entre las losas con contacto, o entre la losa y otros elementos, como muros y arquetas.

### 3. Tipos de fallas sobre pavimentos rígidos y flexibles

#### 3.1. Fallas sobre pavimentos rígidos

Dentro de los tipos de fallas que pueden ocurrir en pavimentos rígidos se encuentran los siguientes:

- Deficiencia de sellado: Este tipo de falla se ocasiona cuando se deterioran el sellado de las juntas, por ejemplo, cuando existe fluencia fuera de la caja, despegado de ambas paredes, incrustación de materiales ajenos. El método de reparación de este tipo de falla consiste en realizar un sellado de juntas y de las grietas.



Figura 5.3: Deficiencia de sellado

- Losas desniveladas: Ocurre cuando se desintegran las aristas de una junta, ya sea de manera longitudinal o transversal, con pérdida de trozos. Para este tipo de falla el método de reparación consiste en un sellado de juntas y grietas, o reparación de espesor parcial, que consiste en reponer las saltaduras de material superficial en juntas y grietas.



Figura 5.4: Losas desniveladas

- Grietas: Una grieta se define como una abertura larga y estrecha en una losa de material, y ésta dependiendo de su ubicación en la losa, puede ser una grieta de esquina, longitudinal (si se extiende a lo largo de una losa) o transversal (si se extiende de manera perpendicular al volcado del material).

de la losa). El método de reparación para este tipo de falla, consiste en el sellado de juntas y grietas, y la reparación en todo el espesor.



Figura 5.5: Grieta longitudinal

- Desintegración: Esta falla se produce cuando ocurre un desgaste progresivo de la superficie, dejando al material árido expuesto. El método de reparación de este tipo de falla, consiste en realizar una reparación de espesor parcial, en la que se corta la porción de la zona a reparar con una profundidad de corte preestablecida, se limpia la zona y se rellena la zona con un material especial de sellado de juntas.



Figura 5.6: Desintegración

- Baches: Un bache se define como una cavidad, generalmente de forma redondeada producto de la pérdida o hundimiento del pavimento en una parte de la superficie. El método de reparación para este tipo de falla depende del deterioro del mismo, y es especial para cada caso.



Figura 5.7: Bache

- Levantamiento: Es el levantamiento de una porción de la losa, localizado en ambos lados de una junta transversal o grieta. El método de reparación consiste en realizar una reparación en todo el espesor, en la que se remueve y reemplaza una porción de la losa en todo su espesor, con el fin de reparar aquellas partes de la losa con un alto grado de daño.
- Escalonamiento de juntas o grietas: Este tipo de falla ocurre cuando existe un desnivel entre dos superficies del pavimento, separadas por una junta transversal o grieta. El método de reparación



Figura 5.8: Levantamiento de juntas

para este tipo de falla es el fresado de la superficie, donde se separan las partes defectuosas del pavimento, de las que se encuentran en buen estado.

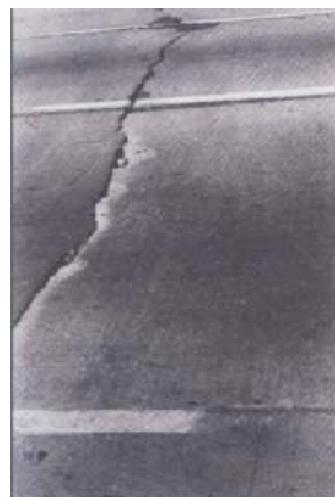


Figura 5.9: Escalonamiento de juntas

- Descenso de banquinas: Es la diferencia de alturas que existe entre el borde del pavimento y la banquina. El método de reparación de este tipo de falla, consiste en realizar el nivelamiento de las banquinas no revestidas.



Figura 5.10: Descenso de banquinas

- Separación banquina-pavimento: Consiste en una rajadura entre el borde del pavimento y la banquina del sendero vial. El método de reparación consiste en realizar un sellado de juntas y grietas.



Figura 5.11: Separación banquina-pavimento

- Parches deteriorados: Este tipo de falla surge cuando una porción de la superficie del asfalto ha sido removido y reemplazado por otro (como hormigón o asfalto), y ésta se ha dañado. El método de reparación de este tipo de falla, varía según el deterioro, y requiere una reparación especial.



Figura 5.12: Parches deteriorados

### 3.2. Fallas sobre pavimentos flexibles

- Exudación: Este tipo de falla se presenta cuando el asfalto pierde sus agregados pétreos en la superficie. El tipo de reparación para esta falla consiste en enarenar y realizar una reparación superficial.

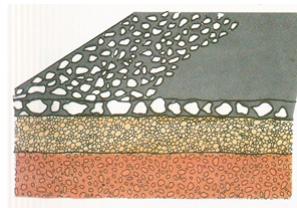


Figura 5.13: Exudación

- Ahuellamiento y depresiones: El ahuellamiento es producido por el hundimiento de manera longitudinal del pavimento, y éste coincide con el área donde la mayor parte de los vehículos circulan. Mientras que depresión, se considera un ahuellamiento de longitud menor al del ahueamiento. Si la profundidad máxima del ahueamiento es inferior a los 20 mm, se realiza perfilado del pavimento, si la profundidad máxima es mayor a 20 mm pero inferior a 40 mm se realiza relleno de la rodadera, y si ésta es superior a 40 mm, se realiza una reparación local del pavimento.

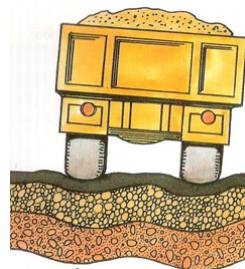


Figura 5.14: Ahueamiento y depresión

- Grietas: Este tipo de falla tiene las mismas características que para pavimento rígido.
- Hundimiento del borde y ahueamiento: Ocurre cuando se hunde el borde del material que limita con el margen. El método de reparación depende de la profundidad máxima del mismo, si ésta es menor a 20 mm se emplea perfilado del pavimento, si es superior a 20 mm pero inferior a 40 mm se emplea la técnica de relleno de rodadura, si es mayor a 40 mm se realiza una reparación local de la estructura del pavimento.

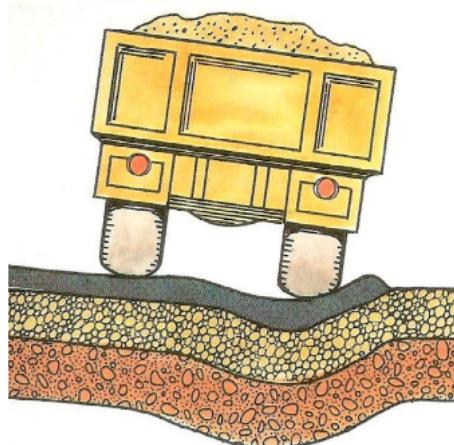


Figura 5.15: Hundimiento del borde y ahueamiento

- Baches: Este tipo de falla produce un hundimiento local del sendero vial, con agrietamiento en malla cerrada y generalmente pérdida parcial de bloques de la capa de rodadura. Las técnicas de reparación de este tipo de falla consisten en restauración local de la estructura del pavimento, o bacheo sobre la base.



Figura 5.16: Baches

- Pérdida local de áridos: Este tipo de falla se presenta cuando ocurre una pérdida de una porción de la capa superficial. El método de reparación para este tipo de falla consiste en reemplazar el material afirmado.

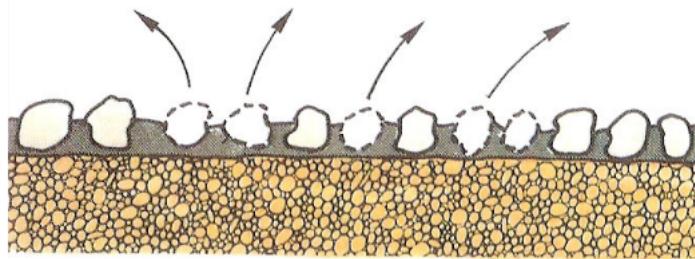


Figura 5.17: Pérdida local de áridos

- Pulimiento o peladuras: Produce desprendimientos de la última capa de tratamientos superficiales. El método de reparación varía según la severidad, siendo un método de tratamiento superficial simple con una profundidad menor a 25 mm y área menor a 0,8 m<sup>2</sup>, y siendo nivelación con sobrecarpa para profundidad o área mayores a 25 mm y 0,8 m<sup>2</sup> respectivamente.

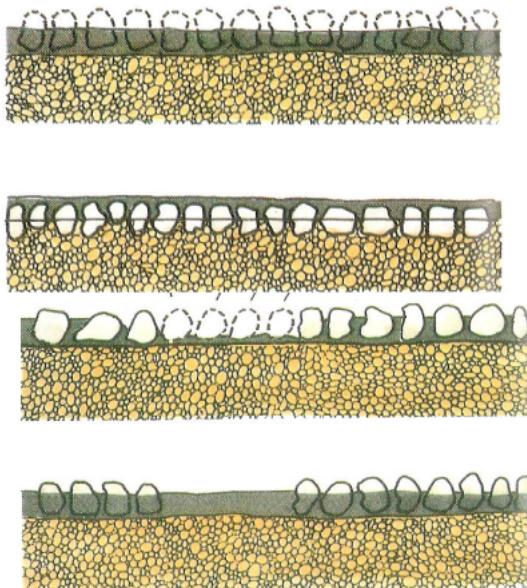


Figura 5.18: Pulimiento o peladuras

- Deformación: Se visualiza en pavimentos donde se produce una desviación longitudinal del material con respecto a su perfil original (asentamientos en el pavimento). El método de reparación con profundidad de la flecha de 13 a 25 mm o entre 25 mm y 50 mm es tratamiento superficial con medida preventiva, mientras que si se superan los 50 mm, se emplea bacheo seguido por aplicación de tratamiento superficial.



Figura 5.19: Deformación

## 4. Proyectos de software anteriores para la detección de fallas sobre el pavimento

Debido a que la recolección manual de fallas es una tarea costosa con respecto al tiempo y al esfuerzo, durante años se han publicado varios papers y tesis con el fin de automatizar esta tarea a través del uso de distintos dispositivos para el sensado, combinados con distintas técnicas de localización, registro de fallas y análisis de la información sensada. Los papers y tesis publicados hasta la fecha se pueden clasificar según el tipo de sensor que emplean, en tres grandes grupos:

- Detección de fallas utilizando técnicas basadas en procesamiento 2D con imágenes o video.
- Detección de fallas por medio de sensores de vibración(acelerómetros).
- Detección de fallas empleando modelos de reconstrucción 3D (sensores para captura de objetos empleando información 3D).

### 4.1. Proyectos basados en procesamiento de video e imagen

#### 4.1.1. Conceptos de procesamiento de imágenes

##### Digitalización de imágenes

El mundo percibido diariamente por las personas se manifiesta en una variedad de formas, colores y texturas que la visión humana puede adquirir, integrar e interpretar con relativa facilidad, como así también, reconocer éstas en sus representaciones asociadas en textos, presentaciones multimedia, imágenes o video digital. No obstante, existe una gran cantidad de radiación que puede ser sensada, que se encuentra delimitada por el **espectro electromagnético**, descubierto por Sir Isaac Newton 1666, cuando un rayo de luz atravesó través de un prisma, y Newton observó que el haz de luz es blanco, sino que se compone de un espectro continuo de colores desde violeta en un extremo (0.43 micrometros) hasta rojo en el otro(0.79 micrometros).

Como se puede observar en la figura anterior, en un extremo del espectro se encuentran las ondas de radio que se caracterizan por poseer longitudes de onda millones de veces mas largas a las de la luz visible, mientras que en el otro extremo se encuentran los rayos gamma con longitudes de onda millones de veces más pequeñas. El espectro electromagnético se puede expresar en función de la energía, la longitud de onda (wavelength,  $\lambda$ ). La longitud de onda del espectro electromagnético representado por  $\lambda$ , se calcula en base a la velocidad de la luz y la frecuencia de la onda, relacionados por la expresión:

$$\lambda = c/v \quad (5.1)$$

donde  $c$  es la velocidad de la luz ( $2.998 \times 10^8$  m/s) y  $v$  es la frecuencia medida en Hertz(Hz), donde 1 Hertz es igual a la cantidad de ciclos de onda senoidal por segundo. Las unidades de las longitudes de onda se

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

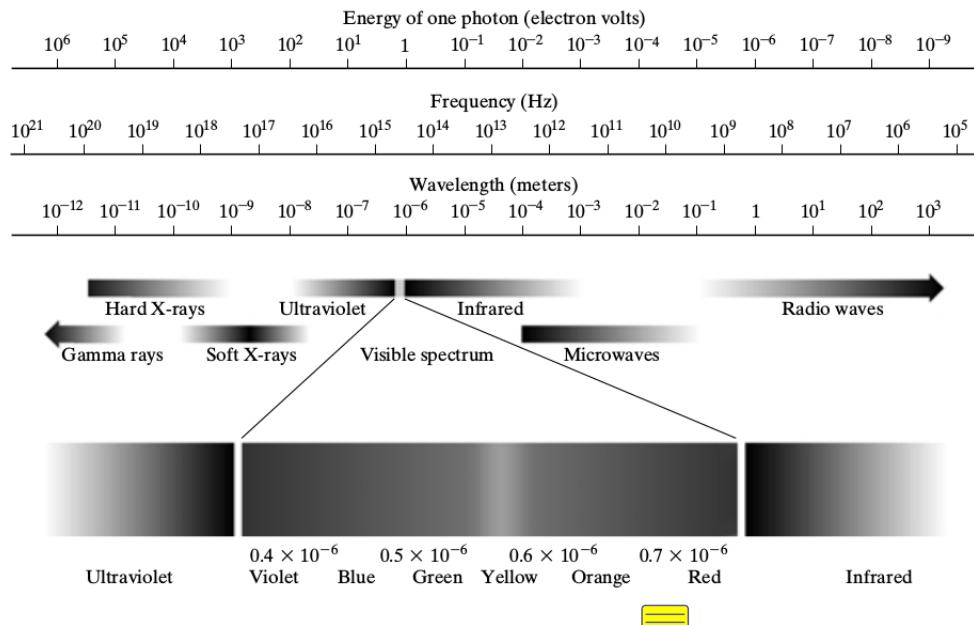


Figura 5.20: División del espectro electromagnético. La zona percibible por las personas del espectro se encuentra ampliado, siendo este una pequeña porción del espectro completo.

miden en metros, empleándose las medidas micrómetros y nanómetros frecuentemente. Alternativamente, el espectro electromagnético se puede expresar en base a la energía asociada con una longitud de onda por medio de la siguiente ecuación:

$$E = h * v \quad (5.2)$$

donde  $h$  es la constante de Planck y  $v$  es la frecuencia de la onda, midiéndose la energía  $E$  a través de la unidad fotón. Por lo tanto, las ondas electromagnéticas pueden ser vistas como ondas senoidales con longitud de onda  $\lambda$ , o pueden ser consideradas como un flujo de partículas sin masa, cada una viajando en un patrón con forma de onda y moviéndose a la velocidad de la luz. Cada partícula sin masa, contiene una cierta cantidad de energía denominada fotón(photon). En la ecuación (5.2), se puede observar que la energía es proporcional a la frecuencia, por lo que cuanto más alta sea la frecuencia el fenómeno electromagnético llevará más energía por fotón. Así las ondas de radio tienen fotones con baja energía, las microondas tienen más energía que las ondas de radio, las infrarrojas aún más, siendo la luz visible, luz ultravioleta, rayos X y finalmente los rayos gamma los que tienen mayor cantidad de energía de todos. Esta es la razón por la cual los rayos gamma son los más dañinos para los organismos vivientes.

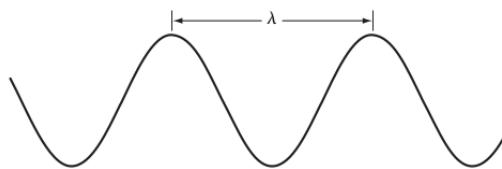


Figura 5.21: Representación gráfica de la longitud de onda (LAMDA)

Sin embargo, el ojo humano sólo puede capturar la luz visible de la radiación electromagnética, que representa una porción mínima de la radiación que puede ser percibida, y aunque esta banda es óptima ya que el volumen de información se encuentra reducido, es altamente confiable y disponible (ya que se encuentra fuertemente proyectada por el Sol y la atmósfera de la tierra es lo suficientemente transparente como para percibirla), la radiación de otras bandas puede ser igualmente útil para ciertas ramas de la ciencia, que graban y hacen uso de casi todo el espectro y emplean esta información con el objetivo de obtener un mejor concepto de la realidad física. Un ejemplo de esto son las ondas de sonido de alta frecuencia o ultrasonido, que son usadas para crear imágenes del cuerpo humano mientras que

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

las imágenes de baja frecuencia son empleadas por compañías, para crear imágenes de la superficie de la tierra. Aunque la captura de imágenes se basa principalmente en la energía generadas por las ondas electromagnéticas, existen otros métodos para la generación de imágenes, tales como capturar el sonido reflejado desde un objeto con el fin de obtener imágenes ultrasónicas, o rayos de electrones como los que emplean los microscopios de electrones para obtener imágenes que permitan recolectar información respecto de especímenes biológicos e inorgánicos, incluyendo microorganismos, muestras de biopsias, metales y cristales.

Las imágenes, que tengan distintas fuentes, comparten el hecho de que existe una radiación que es emitida desde alguna fuente para posteriormente interactuar con algún tipo de material, luego es sensada y trasladada en una señal eléctrica que puede ser digitalizada. Las imágenes se pueden clasificar según la forma en la que la interacción con el dispositivo de sensado ocurre en 3 categorías generales:

- Las imágenes de reflexión son aquellas en que la radiación ha sido reflejada desde la superficie de un objeto. Esta puede ser del ambiente o artificial, y puede provenir desde una fuente localizada o desde fuentes múltiples. Este tipo de imágenes son las que se perciben día a día por las personas por medio de la vista, mientras que algunos ejemplos de imágenes no visibles de este tipo incluyen imágenes por radar, imágenes por sonar y algunos tipos de imágenes por microscopio. El tipo de información que puede ser extraída desde este tipo de imagen es generalmente respecto de la superficie de los objetos, su forma, color, textura y reflectividad.
- Las imágenes de emisión son aquellas cuya radiación es emitida por el objeto que se desea capturar, como las imágenes térmicas o infrarrojas, y que son usadas por áreas como la medicina, pruebas militares, o en objetos luminosos como bombillas de luz, estrellas, imágenes de resonancia magnética (MRI), las cuales obtienen información en base a la capacidad de emisión de las partículas. Cuando se emplea este tipo de imágenes se desea obtener información respecto de la estructura interna del objeto, aunque también pueden ser empleadas para información externa, por ejemplo, una cámara térmica utilizada en situaciones con baja iluminación, con el fin de producir una imagen que capture los objetos que producen calor en una escena.
- Las imágenes de absorción donde la radiación atraviesa el material que compone el objeto y es absorbida o atenuada por éste parcialmente, lo que proporciona información relacionada con la estructura interna del mismo. El grado de absorción determina el nivel de la imagen registrada. Ejemplos de este tipo de imágenes son los rayos X, imágenes de transmisión microscópicas y ciertos tipos de imágenes sónicas.

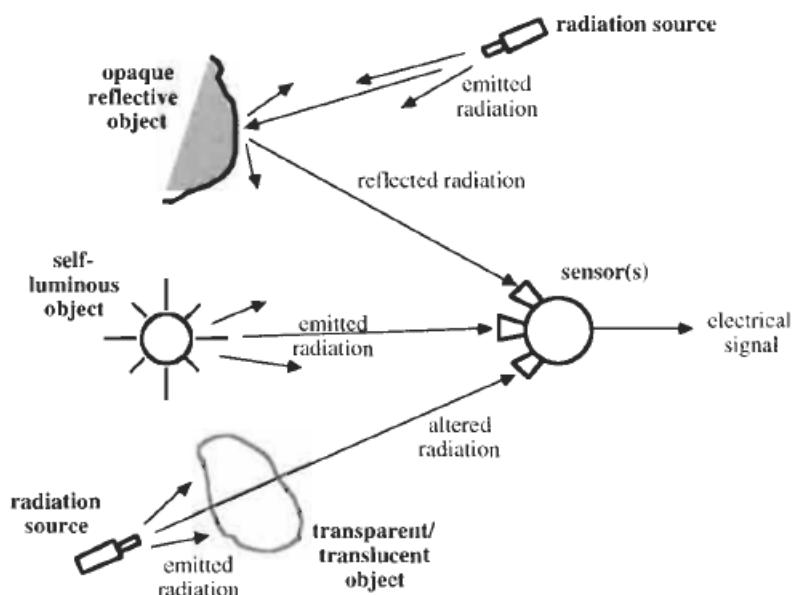


Figura 5.22: Tipos de interacción para el sensado de imágenes

Para que un sensor pueda captar un objeto de determinado tamaño, es necesario que la longitud de onda del sensor sea igual o menor al tamaño del objeto, por lo que este requerimiento junto

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

con el material del sensor, establecen los límites de la capacidad de captura del sensor de imagen y su clasificación en distintos tipos, tales como sensores infrarrojos, de luz visible,etc. Así, con el fin de capturar imágenes digitales en las distintas bandas del espectro electromagnético, es necesario emplear sensores que puedan captar la energía irradiada en cierto rango y produzcan una señal eléctrica de salida (generada por una combinación entre el material sensible a la radiación del sensor y la fuente de alimentación del mismo), que permita la representación de una imagen del mundo tridimensional de interés en formato digital.

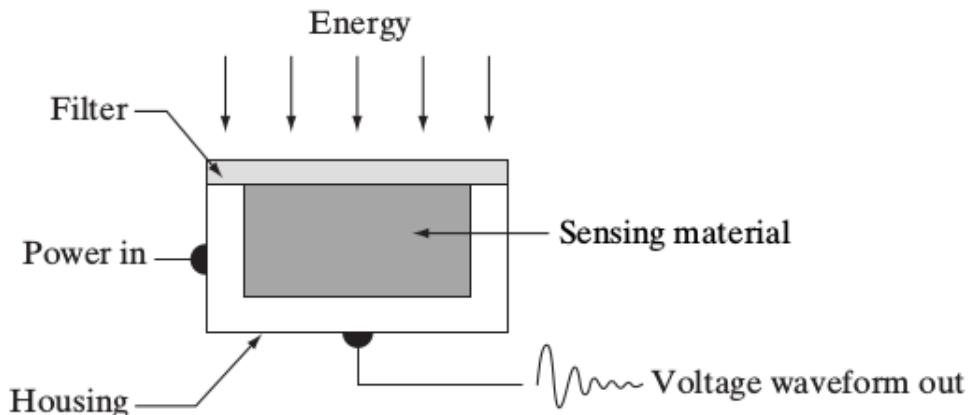


Figura 5.23: Sensor individual de captura

Cuando un fenómeno es captado por un dispositivo con uno o varios sensores, estos en general producen una onda de voltaje continua cuya amplitud y forma esta relacionada a la radiación emitida o reflejada desde el objeto, por lo que para crear una imagen digital, es necesario realizar una conversión de los datos en un formato digital, dando como resultado una imagen digital. Este proceso comienza con la conversión de las coordenadas espaciales de la imagen a una matriz multidimensional que pueda ser indexada por valores numéricos(también llamado proceso de muestreo o sampling), de esta forma la señal puede ser almacenada y procesada como un arreglo de  $M$  filas x  $N$  columnas de valores discretos, donde cada uno de los elementos  $(i,j)$  que pueden ser indexados en la matriz se denomina elemento de imagen(picture element), pel o pixel. Así, si una imagen digital contiene  $M \times N$  pixeles, se representa por una matriz de  $M \times N$  elementos conteniendo desde 0 hasta  $M-1$  índices en las filas y desde 0 hasta  $N-1$  índices en las columnas. Cuando la cantidad de pixeles muestreados no es suficiente(undersampling) como para representar la imagen, se produce un efecto denominado aliasing, que produce que la imagen visual pierda el patrón de la imagen original que intenta representar, produciendo una falso patrón y una imagen distorsionada. Como se observa en la siguiente imagen de una huella digital, a medida que la densidad de pixeles muestreados disminuye, la calidad de la imagen empeora y se produce éste efecto:

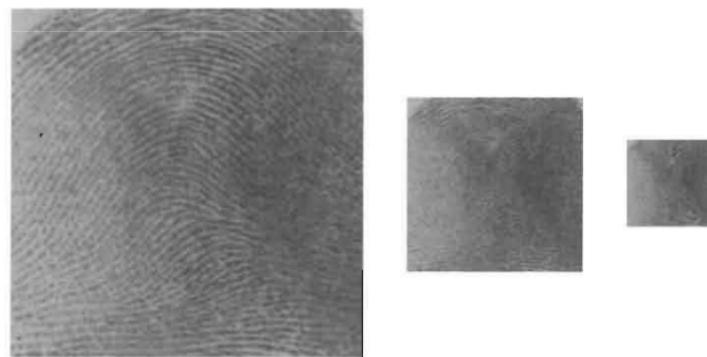


Figura 5.24: Efecto de aliasing. 256x256 ( $2^8 \times 2^8 = 65,536$  muestras). 128x128( $2^7 \times 2^7 = 16,384$  muestras). 64x64( $2^6 \times 2^6 = 4,096$  muestras)

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

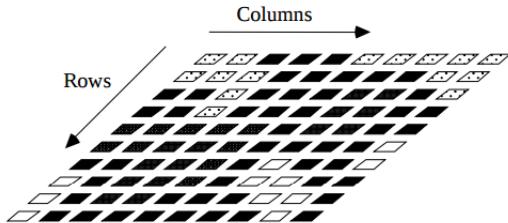


Figura 5.25: Representación de un array de imagen de  $10 \times 10$

El siguiente paso consiste en realizar la cuantificación o quantization, donde se realiza la conversión de las intensidades analógicas captadas por los sensores a valores numéricos discretos, asignando un valor a cada pixel muestreado, de manera que la imagen reconstruida de los valores muestreados sea de una calidad lo más aproximada a la real y el error introducido por la cuantificación sea mínimo. Con el fin de cuantificar, el rango de valores dinámicos que puede adoptar los pixeles de una imagen se divide en un rango finito de intervalos, y a cada intervalo se le asigna un valor. Cuanto mayores sean los intervalos disponibles para cuantificación, la imagen digitalizada se aproximará con más fidelidad a la imagen real. La cuantificación se puede realizar de manera uniforme, cuando los valores de intensidad tienen mayor probabilidad de caer en intervalos regulares y se opta por dividir el rango de niveles en intervalos igualmente espaciados. Por otro lado, cuando la imagen adopta valores en un rango con una frecuencia prolongada y otros valores de manera infrecuente, es preferible emplear la cuantificación no uniforme.

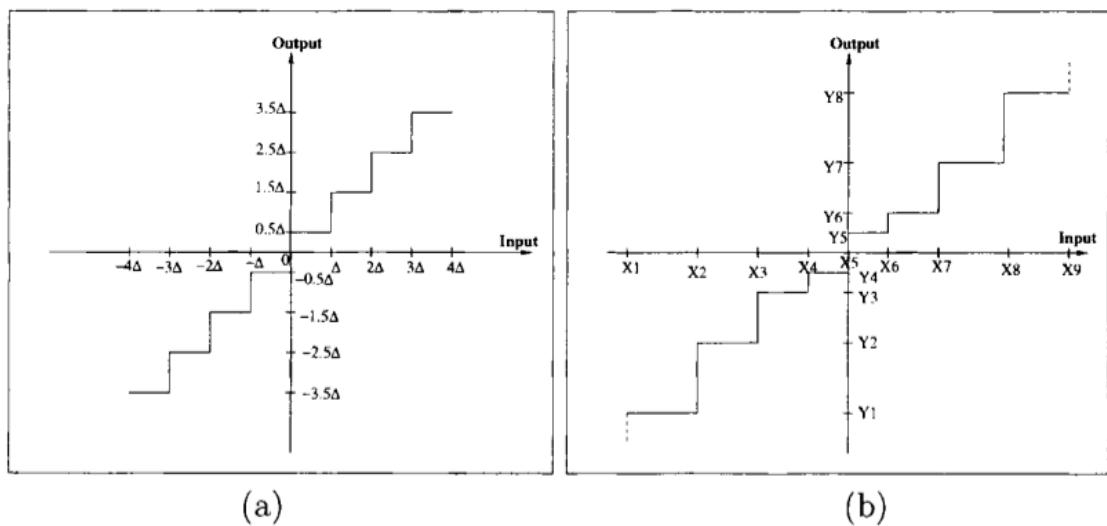


Figura 5.26: Cuantificación de imágenes de 2 dimensiones. Cuantificación uniforme (a). Cuantificación no uniforme (b).

De esta forma, el proceso de digitalización requiere los valores de  $M$ ,  $N$  y la cantidad de niveles de intensidad  $L$  (niveles de gris en el caso de las imágenes con escala de grises o de valores en las bandas roja, verde y azul para las imágenes a color) como valores positivos, permitidos para cada pixel. No obstante, debido a las consideraciones de hardware, procesamiento y almacenamiento, el número de niveles es típicamente una potencia de 2:

$$L = 2^k \quad (5.3)$$

Donde  $k$  es el número de bits empleados para representar el nivel de cada pixel. En general, el número de bits  $k$  se encuentra entre  $1 \leq k \leq 8$ , empleándose  $k = 1$  para imágenes binarias,  $k = 8$  para imágenes por escala de grises (donde cada nivel ocupa un byte) y, para el caso de las imágenes a color, con múltiples valores, cada nivel de color ocupa 8 bits usando los colores rojo, verde y azul (RGB), empleándose 24 bits por pixel con el fin de representar el color de éste. Así, cuando una imagen puede tener  $2^k$  niveles de gris, es una práctica común referirse a la imagen como una “imagen de  $k$ -bits”. Por ejemplo, una imagen con 256 niveles posibles es llamada una imagen de 8 bits. Por lo tanto, la cantidad de bits requeridos para almacenar una imagen será:

$$b = MxN \times k \quad (5.4)$$

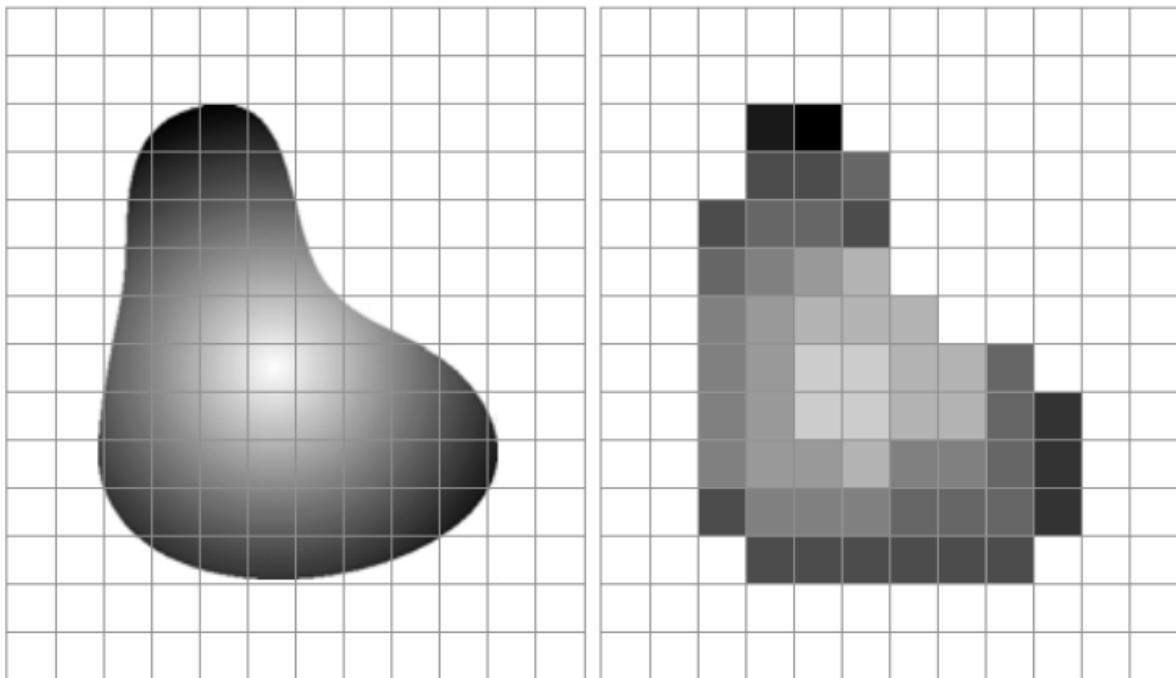


Figura 5.27: Representación del proceso de muestreo y cuantificación. Imagen continua captada por un dispositivo de sensado(izquierda). Imagen muestreada y cuantificada(derecha).

### Relaciones entre pixeles

Los pixeles  $P_k$  en la coordenada  $(i,j)$ , con  $k$  siendo la cantidad total de pixeles con los indices  $i=1,2,\dots,n$  y  $j=1,2,\dots,m$ , que componen una imagen digital cuentan con distintas propiedades entre las que se encuentran las siguientes:

- **Pixeles conectados:** Un pixel en un punto  $P_0$  en  $(i_0,j_0)$  se conecta a otro pixel  $P_n$  en  $(i_n,j_n)$  si y sólo si existe un camino desde  $P_0$  hasta  $P_n$ , que es una secuencia de puntos  $(i_0,j_0), (i_1,j_1), \dots, (i_n,j_n)$  tal que el pixel  $(i_k,j_k)$  es un vecino del pixel en  $(i_{k+1},j_{k+1})$  y  $P_k = P_{k+1}$  para todos los  $k$ ,  $0 < k < n-1$ . La secuencia de pixeles distintos de un pixel a otro también se denomina camino digital (digital path) y, si el primer pixel del camino se encuentra conectado con el primer pixel, se denomina un camino cerrado.
- **4-vecinos(4-connected pixel):** Cuando un pixel  $P$  en la ubicación  $(i,j)$  tiene cuatro vecinos en las coordenadas  $(i+1,j)$ ,  $(i-1,j)$ ,  $(i,j+1)$  e  $(i,j-1)$  se conocen como 4-vecinos. Es decir, que cada pixel está a una unidad de distancia de  $(i,j)$  y algunas de las ubicaciones de  $P$  yacen fuera de la imagen digital en el borde la imagen.

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

- 8-vecinos(8-connected pixel): Se dice que un pixel  $P$  ubicado en  $(i,j)$  tiene una conexión diagonal de 4 pixeles, cuando tiene pixeles en las coordenadas  $(i+1,j+1), (i+1,j-1), (i-1,j+1)$  e  $(i-1,j-1)$ . Si además este pixel tiene 4-vecinos, se dice que estos pixeles son 8-vecinos de  $P$ .

Otra propiedad de los pixeles es la adyacencia que se define en términos de los niveles de intensidad, siendo  $V$  el conjunto de valores de intensidad que un pixel puede adoptar, con  $V = \{1\}$  en imágenes binarias (considerándose adyacentes dos pixeles que tienen intensidad 1) y  $V$  siendo un subconjunto de todos los niveles de la imagen (para el caso de imágenes por escala de grises) y considerándose adyacentes dos pixeles cuyos valores de intensidad están en ese subconjunto, existen 3 tipos de adyacencia:

- 4-adyacentes(4-adjacency). Dos pixeles  $P$  y  $Q$  con valores del conjunto  $V$  son 4-adyacentes si  $Q$  está en el conjunto de los 4-vecinos de  $P$ .
- 8-adyacentes(8-adjacency). Dos pixeles  $P$  y  $Q$  con valores del conjunto  $V$  son 8-adyacentes si  $Q$  está en el conjunto de los 8-vecinos de  $P$ .
- adyacencia mixta(m-adjacency o mixed-adjacency). Dos pixeles  $P_0$  y  $P_1$  con valores del conjunto  $V$  son m-adyacentes si:
  - $P_0$  es un 4-vecino de  $P_1$ , o
  - Si  $P_0$  está en una conexión diagonal de  $P_1$  y el conjunto de 4-vecinos de  $P_0$  y de  $P_1$  no tienen valores en común con el conjunto  $V$  de niveles.
- Componente conectado: Si dado un subconjunto de pixeles  $S$  en una imagen, dos pixeles  $P_0$  y  $P_1$  se dicen conectados si existe un camino digital que se compone de los pixeles en  $S$ . Así, para cualquier pixel  $P$  que esté en  $S$ , el conjunto de pixeles que están conectados a él es llamado un componente conectado de  $S$ . Un conjunto de pixeles conectados (4 u 8 pixeles) forman un componente conectado, que representa un objeto en escena.
- Región. Dado un subconjunto de pixeles  $R$  en una imagen,  $R$  se denomina una región si es un componente conectado, y dos regiones  $R_1$  y  $R_2$  se dicen adyacentes si su unión forma un conjunto conectado, o disjuntas en caso contrario.
- Fondo(background) y Frente(foreground). Si una imagen contiene  $R_k$  regiones con  $k=1,2,\dots,N$ , la unión de todas las regiones se considera el frente, mientras que el resto de los pixeles que no están en ninguna región se considera el complemento.
- Borde o Contorno(boundary,border,contour). El contorno de una región  $R$  es el conjunto de los puntos que son adyacentes a los puntos que no están en  $R$  (complemento), es decir, que éste se compone de aquellos pixeles en la región que tienen al menos un vecino que forma parte del fondo. Si  $R$  es una imagen entera(matriz de pixeles), entonces su contorno se define como el conjunto de pixeles en la primera y última fila y columna de la imagen, ya que una imagen no contiene más vecinos más allá de los bordes.

Una vez que un objeto es identificado algunos de sus atributos se pueden definir de la siguiente manera:

- Área del objeto: El área de un objeto se da como la sumatoria de todos los pixeles  $i,j$  que forman el objeto(pixeles con valor 1).
- Ubicación del objeto: La ubicación del objeto se define como el centro del objeto en  $X$  e  $Y$ , calculados por medio de la sumatoria de las coordenadas del objeto dividido por el área del mismo. En la siguiente ecuación se puede observar la forma de calcular los centroides  $X_c$  e  $Y_c$ :
- Orientación de un objeto: Cuando el objeto tiene una forma alargada, los ejes de la elongación producen la orientación del mismo. El eje de elongación es una línea recta tal que la suma de las distancias al cuadrado, de todos los puntos del objeto desde esta línea es mínimo(distancia perpendicular de un punto del objeto hacia la línea).

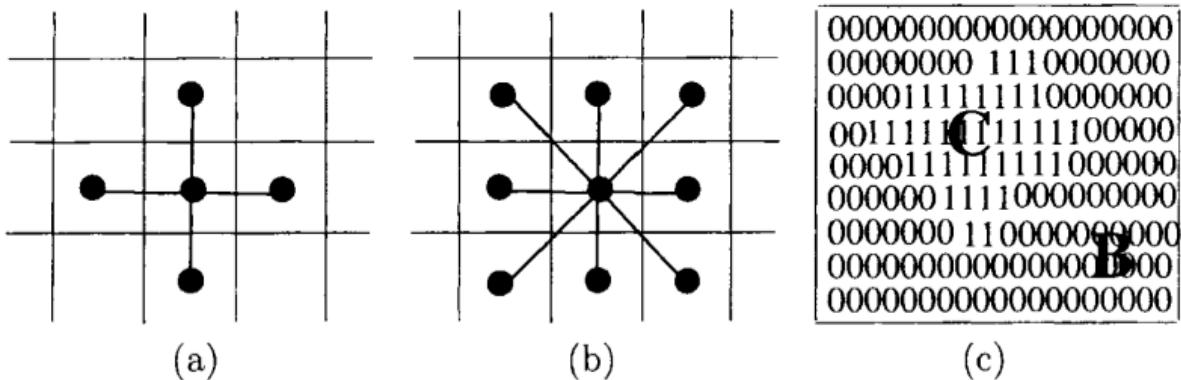


Figura 5.28: Tipos de conexiones entre píxeles. 4-vecinos(a). 8-vecinos(b). Componente conectado y fondo(c).

$$x_c = \frac{\sum_i \sum_j [i O(i, j)]}{A}, \quad y_c = \frac{\sum_i \sum_j [j O(i, j)]}{A}$$

Figura 5.29: Fórmula para el cálculo del objeto

- Perímetro de un objeto: El perímetro de un objeto se obtiene sumando los píxeles que forman parte del límite del objeto y que son parte del área. El límite o contorno de un objeto está formado por aquellos píxeles que tienen uno o más vecinos que no están en el área.

### Herramientas y técnicas sobre imágenes digitales

De forma general, existen dos tipos de aproximaciones que pueden emplearse en una imagen para aplicar técnicas de mejora de imagen y transformación: Emplear técnicas que actúen en el dominio espacial de la imagen, es decir modificando ciertas características sobre los píxeles de la imagen directamente; o emplear técnicas que se ejecutan sobre el dominio de frecuencias de la imagen, que consisten en realizar una conversión de los valores de la imagen para llevarla a otro dominio, ejecutar transformaciones sobre ese dominio y finalmente, realizar la transformación inversa para obtener la imagen de salida.

#### Técnicas que actúan sobre el dominio espacial

Debido a que las imágenes se representan como matrices, es posible tanto aplicar operaciones aritméticas y lógicas entre matrices, como ejecutar operaciones que modifiquen características de los píxeles, con el fin de modificar ciertas características de éstas. Los tipos principales de operaciones que se pueden emplear se pueden clasificar en 3 tipos generales:

- Operaciones de manipulación de intensidad (modificación del nivel de intensidad de uno o varios píxeles).
- Operaciones aritméticas entre matrices de la misma dimensión (estas operaciones incluyen suma, resta, multiplicación y división entre matrices).
- Operaciones geométricas de transformación (interpolación, traslación, rotación, filtrado espacial).

#### Operaciones de manipulación de intensidad

En este tipo de técnicas las relaciones entre píxeles vecinos no se consideran, como así tampoco la localización de los píxeles, sino que se modifican las intensidades de los píxeles. Estas operaciones pueden realizarse sobre píxeles individuales, tomando como entrada el valor de intensidad de un pixel

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

y produciendo el valor de intensidad transformado para ese pixel. Alternativamente, se pueden aplicar sobre un vecindario de pixeles(neighbourhood), donde dada la coordenada de un pixel en la imagen de entrada  $f(x,y)$ , se toman los valores de intensidad de un conjunto de pixeles vecinos y, por medio del procesamiento de estos valores, se obtiene el valor de intensidad para ese pixel en la imagen de salida  $g(x,y)$ .

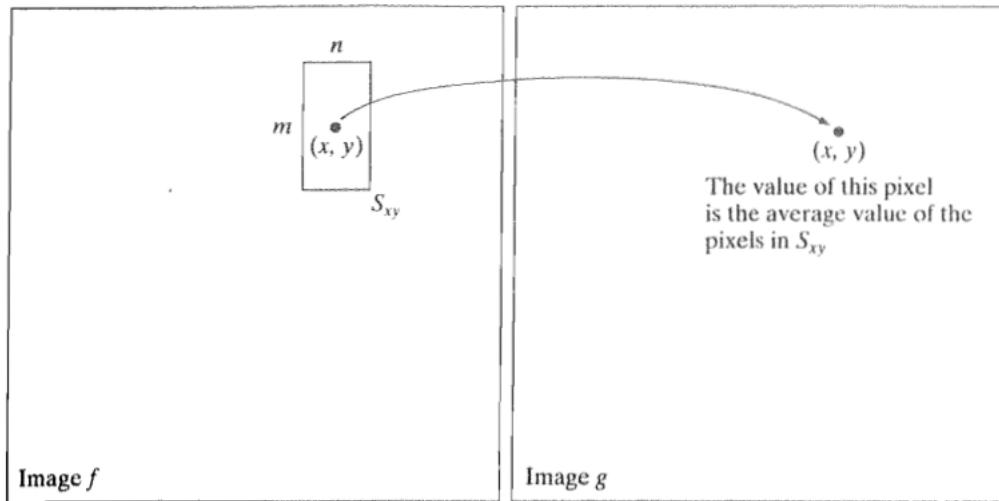


Figura 5.30: Vecindario de pixeles

En la parte restante de la sección, se describen algunas de las herramientas que se emplean para modificar los valores de intensidad de una imagen asociadas al histograma de imagen (operaciones por pixeles individuales) y la operación de filtrado espacial (operación que abarca un conjunto de pixeles) y sus principales utilidades.

#### Histograma de imagen

La herramienta básica para este tipo de operaciones es el histograma de imagen, que es una representación gráfica que agrupa las frecuencias de ocurrencias de cada nivel de intensidad (nivel de gris en imágenes por escala de grises) en los pixeles de la imagen. De esta manera, si la imagen  $f()$  cuenta con  $K$  niveles de intensidad  $\{0, 1, \dots, K-1\}$  y una cantidad  $N \times M$  de pixeles, el histograma  $Hf$  se define matemáticamente de la siguiente manera:

$$Hf(k) = J \quad (5.5)$$

Donde  $f()$  es la función que mapea el nivel de intensidad a cada pixel  $P(x,y)$ , y  $J$  representa la cantidad de ocurrencias de ese nivel en los pixeles, con  $K$  niveles. Aunque este tipo de histograma no contiene información espacial con respecto a la imagen, es una herramienta valiosa que permite visualizar si la distribución de niveles de intensidad en una imagen es correcta, o si la imagen tiene tonalidades más oscuras o más claras. Por ejemplo, en un histograma que corresponde a una imagen con escala de grises, los niveles más oscuros se concentran sobre la parte más baja de la escala del histograma, mientras que los niveles más brillantes están en la parte alta del diagrama. Así, una imagen por escala de grises con bajo contraste, tendrá un histograma cuyos puntos se encuentran centrados en la escala y abarcan pocos valores en el rango, mientras que si ésta tiene un contraste alto, los valores del histograma abarcarán un rango amplio de la escala y, su distribución tenderá a ser uniforme.

En la siguiente figura se puede observar, que la figura de la izquierda presenta niveles de gris más oscuros, mientras que la figura de la derecha presenta niveles de grises con más brillo, lo que indica que han estado expuestas a condiciones de luz excesiva y escasa.

4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

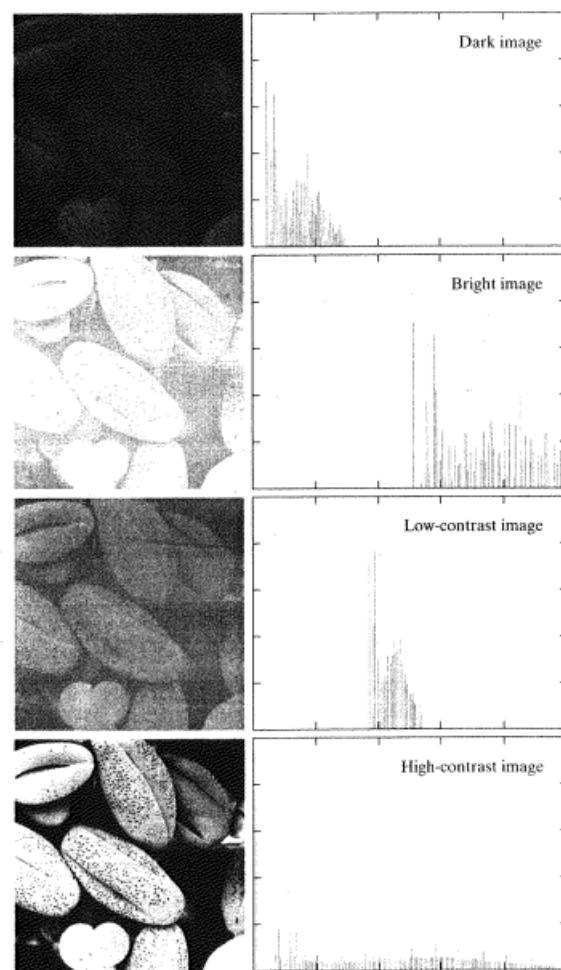


Figura 5.31: Imágenes con distintos niveles de contraste y sus histogramas asociados

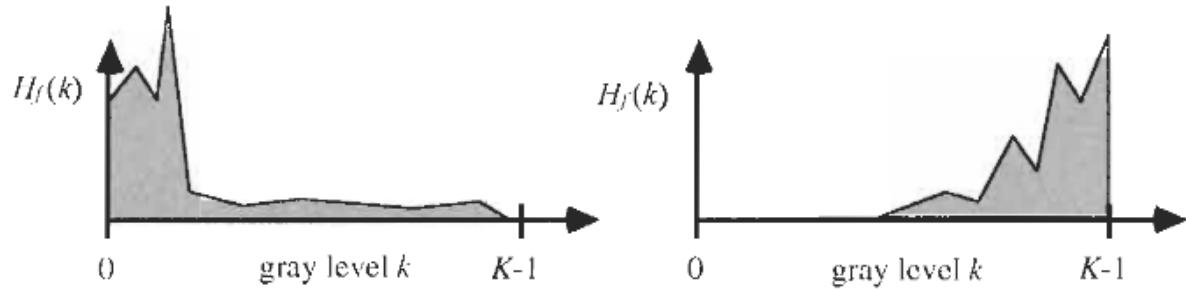


Figura 5.32: Histograma de imagen.

Algunas veces el histograma de imagen se normaliza, dividiendo la cantidad de ocurrencias en cada nivel de intensidad, por el número total de píxeles en la imagen ( $N \times M$ ), de manera que la sumatoria de los componentes de un histograma normalizado sea 1.

El histograma de imagen es una herramienta básica empleada por varias técnicas de procesamiento de imágenes con intensidad como la mejora de imagen, además de proveer información de utilidad para la compresión y la segmentación de imágenes.

### Escalado de imagen

El escalado de imagen consiste modificar el rango de niveles de intensidad que se consideran para representar un histograma. Este procedimiento dada una imagen  $f(n)$  con  $n$  píxeles, implica multiplicar cada uno de estos valores por una constante numérica  $P$  (mayor o menor a 1):

$$g(n) = P * f(n) \quad (5.6)$$

Por ejemplo si se emplea un histograma de una imagen de escala de grises, si el valor de la constante  $P > 1$ , los niveles de gris cubrirán un rango más amplio que aquellos de la función del histograma  $f()$ , mientras que si  $P < 1$  se empleará un rango de niveles de gris más reducido, lo que puede producir pérdida de información en la imagen y disminuir su nitidez. A continuación se pueden observar imágenes originales y los efectos de aplicar el escalado de imagen con dos escalas distintas:

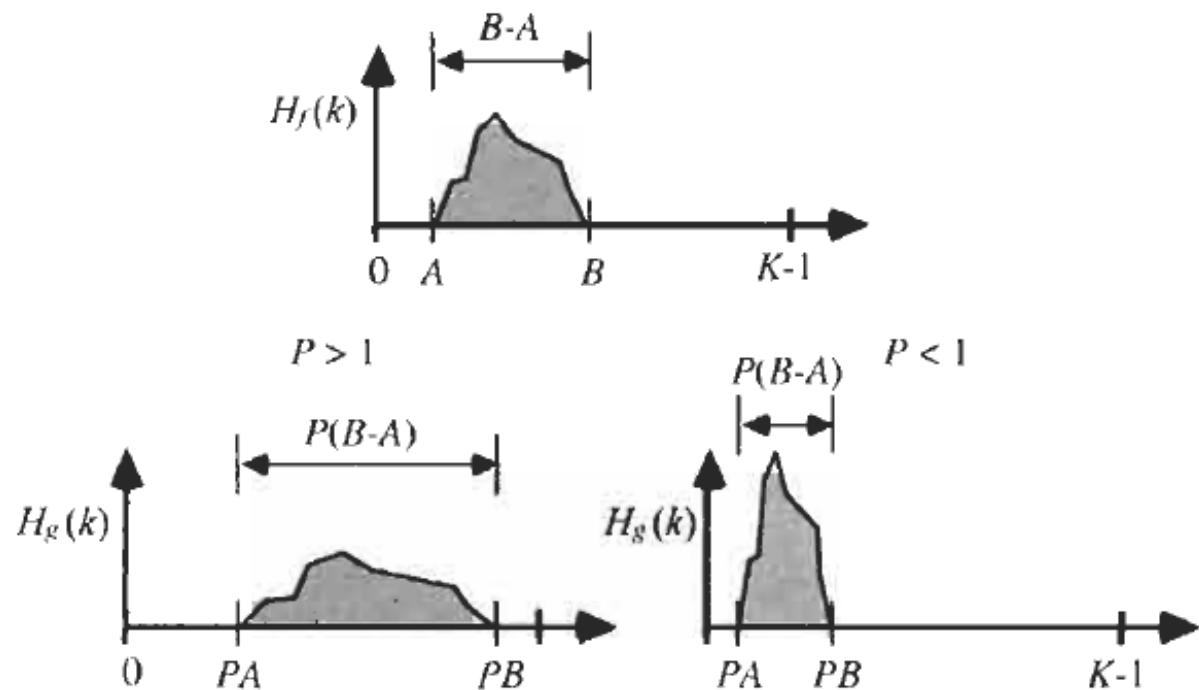


Figura 5.33: Efectos del escalado de histograma entre dos puntos A-B.

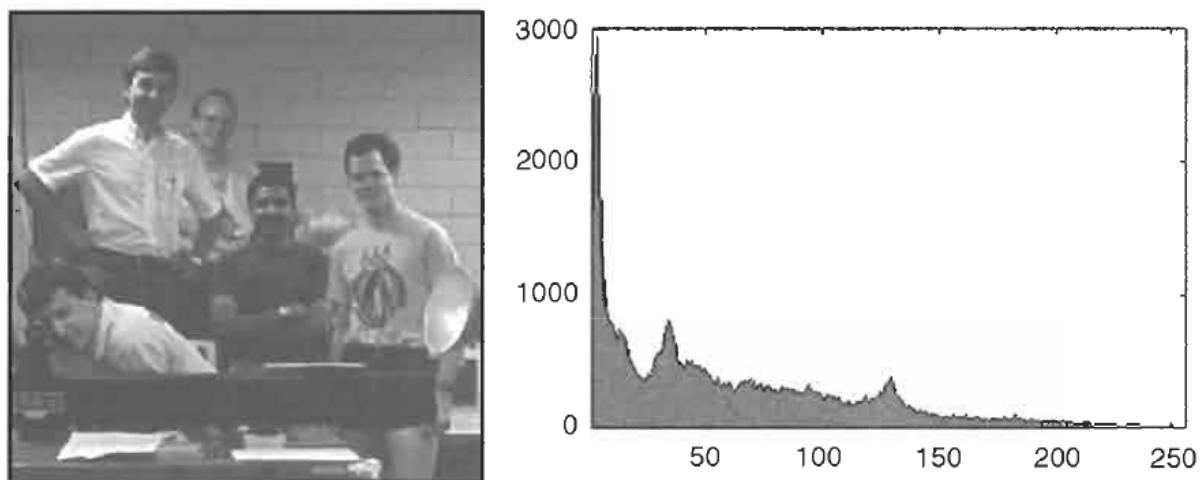


Figura 5.34: Imagen de estudiantes original (izquierda) y su histograma de imagen asociado(derecha).

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

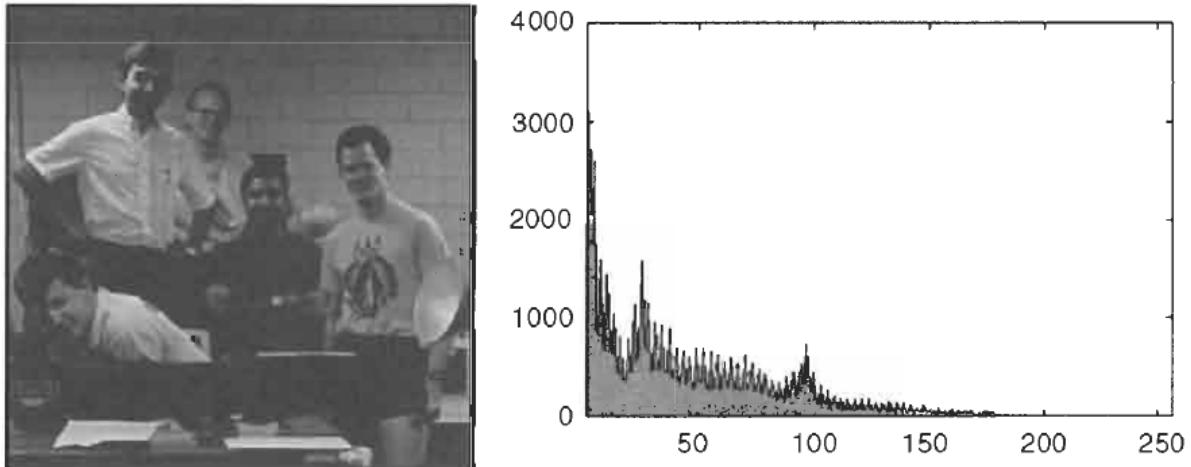


Figura 5.35: Modificación de la escala del histograma con  $P=0,75$ , en este caso los niveles de gris de la imagen tienden a juntarse, provocando que la imagen disminuya su calidad.

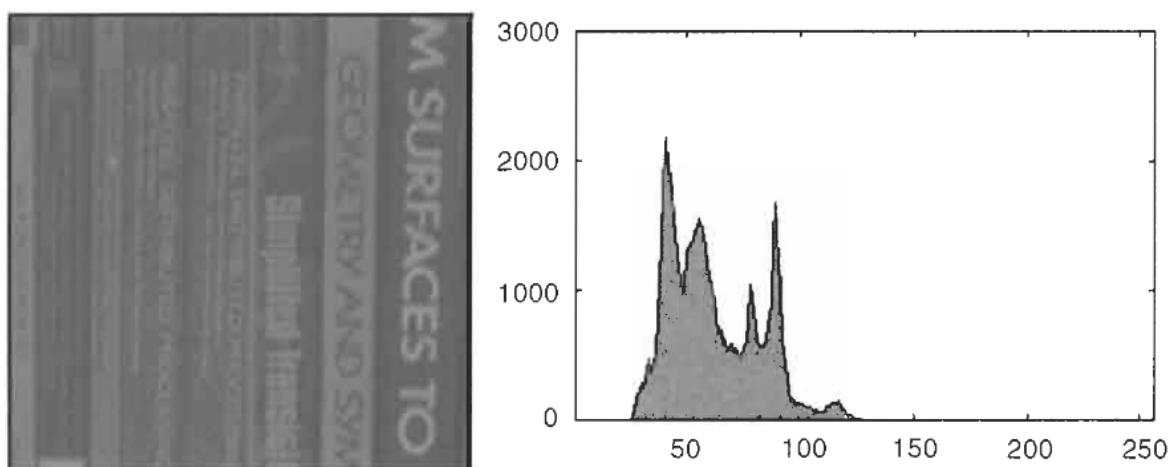


Figura 5.36: Imagen de libros(izquierda) y su histograma(derecha)

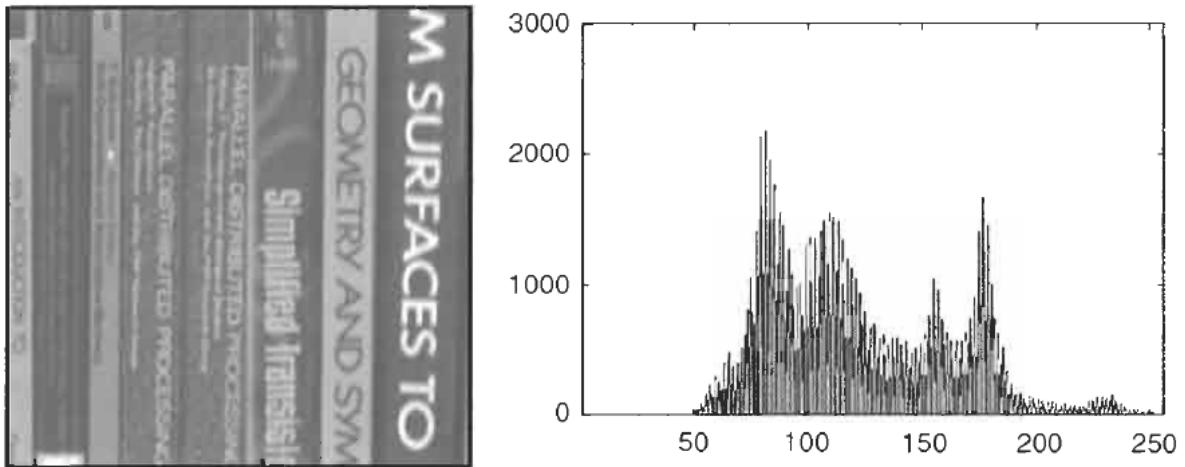


Figura 5.37: Modificación de la escala del histograma con  $P = 2$ . En este caso, la expansión de los valores del histograma de imagen, produce que se haga un mejor uso de los niveles de gris, produciendo una mejora en la nitidez de la misma

### Negativo de imagen

El negativo de una imagen consiste en escalar los valores de la imagen  $f(n)$ , con  $P = -1$  revirtiendo el signo de los valores y sumar un desplazamiento a los valores de intensidad de cada pixel  $K-1$  con el fin de que éstos se encuentren en el rango del histograma:

$$g(n) = -f(n) + (K - 1) \quad (5.7)$$

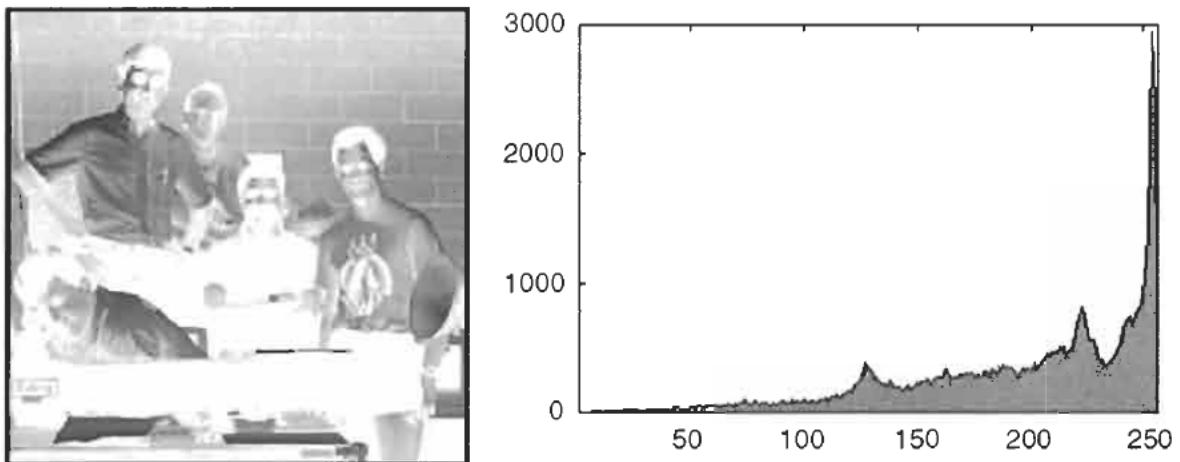


Figura 5.38: Imagen negativa con su histograma modificado

Esta técnica se emplea para mejorar imágenes donde se pierde el nivel de detalle en las regiones con niveles blanco y negro, percibiéndose ésta como demasiado oscura. Un ejemplo de esta operación es la inspección de imágenes telescopicas con campos de estrellas y galaxias, donde con una imagen negativa los objetos brillantes, aparecen con una tonalidad oscura sobre un fondo brillante que es mas sencillo de apreciar.

### Estiramiento de contraste(Contrast Stretching,Histogram Stretching)

Este procedimiento consiste en distribuir las frecuencias de los niveles de intensidad, por medio de una fórmula matemática, en un nuevo histograma donde éstos se encuentren organizados de manera uniforme y abarquen la escala completa de niveles de intensidad. Por ejemplo si se emplea un histograma de una imagen de escala de grises, como el de la figura Fig. 5.33, donde los niveles de intensidad de toda la escala están en el rango  $[0, K-1]$  y los niveles empleados por la figura se encuentran en el rango  $[A, B]$  con  $A$  y  $B$  siendo los valores máximos y mínimos de intensidad respectivamente, se puede emplear la siguiente fórmula matemática que mapee los valores en el nuevo histograma:

$$g(n) = \text{FSHS}[f(n)] = \left( \frac{K-1}{B-A} \right) [f(n) - A].$$

Figura 5.39: Fórmula de estiramiento de contraste

De esta forma, este procedimiento modifica el contraste de la imagen en general si sus niveles de grises no están distribuidos adecuadamente, aunque si ésta abarca varios valores en la escala de grises del histograma, esta técnica puede producir poca o ninguna diferencia con respecto a la imagen original. A continuación se puede observar un ejemplo que contrastado con la figura Fig. 5.36, tiene una mejora en el contraste de la misma:

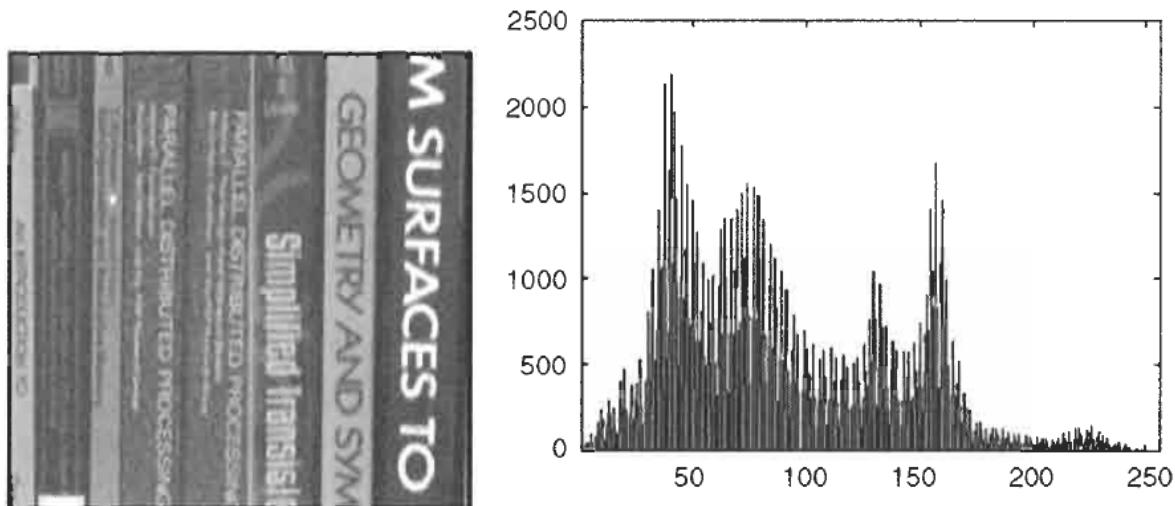


Figura 5.40: Ejemplo de estiramiento de contraste

Este tipo de técnica se emplea en aquellas imágenes con bajo contraste, o con poca iluminación, o con una configuración inapropiada del dispositivo de captura durante la adquisición de la misma, con el fin de lograr una mejor visualización de los detalles en ésta.

### Igualación de histograma(Histogram Equalization)

Este procedimiento consiste en normalizar los niveles de intensidad del histograma de imagen, de manera que éstos sigan una distribución uniforme, y luego realizar un estiramiento de contraste para los niveles abarquen la mayoría de los valores en la escala del histograma. Este procedimiento provoca que el histograma se estire en el eje de las abscisas y tiende a aplandarlo de manera que se adapte a la distribución. Si se considera un histograma de imagen de una imagen con escala de grises, el primer paso consiste en realizar la normalización del histograma, obteniendo la función de densidad de probabilidad (PDF) de los niveles de gris  $p_f(k)$ , para cada uno de los  $K$  niveles de intensidad. La función de densidad

de probabilidad trata los valores de los niveles de intensidad como cantidades aleatorias, y definiéndose la probabilidad  $p_f(k)$  de un nivel  $k$  ocurriendo en una imagen como:

$$p_f(k) = \frac{1}{NM} H_f(k)$$

Figura 5.41: Fórmula de normalización de histograma

Donde la imagen digital tiene  $M \times N$  pixeles,  $H_f(k)$  es el nivel de intensidad para un nivel  $k$  y, con\*  $0, 1, \dots, K-1$ \* niveles. Éstas deben cumplir con la siguiente propiedad de sumatoria:

$$\sum_{k=0}^{K-1} p_f(k) = 1.$$

Figura 5.42: Propiedad de sumatoria de los valores normalizados

En base a esta función, se define la función de distribución acumulada,  $P_f(r)$  para  $r = k$  niveles:

$$P_f(r) = \sum_{k=0}^r p_f(k); \quad r = 0, \dots, K - 1.$$

Figura 5.43: Función de distribución acumulada

Así para obtener un histograma igualado, primero se debe computar la función de distribución acumulada del histograma de imagen  $P_f(k)$  de la imagen digital, para cada uno de los niveles del histograma, lo que provocará que éste tienda a aplanarse gráficamente, y luego aplicar la función de estiramiento de contraste para cada uno de los elementos, con el fin de distribuirlos a lo largo de la escala. Esto provocará que la imagen final sea más impactante y visible que la original, sin embargo, este proceso no eliminará aquellos picos resultantes del proceso de cuantificación. A continuación se observa el proceso de igualación de histograma aplicado a la imagen de libros:

Así, este método se aplica cuando se desea una implementación simple que produzca una mejora automática en la imagen.

### **Limitado de imagen(Image thresholding)**

Esta técnica se emplea principalmente en imágenes con escala de grises, con el fin de abstraer información relevante respecto de los objetos en una imagen y optimizar el procesamiento y análisis subsecuente de la imagen. Este proceso consiste en, dada una imagen con  $K-1$  niveles de gris, definir un límite entero  $T$  dentro del rango de niveles y comparar cada pixel con el límite  $T$ , y si la intensidad del pixel  $p(x,y)$  supera ese límite asignarle la intensidad 0, y en caso contrario asignarle el valor de intensidad 1. De esta forma, modificando el nivel del límite  $T$  se controla la abstracción de información que se generará en la imagen de salida, y dependiendo de las características del histograma de imagen, se abstraerá la cantidad de información relevante de ésta.

Este procedimiento es útil en imágenes que cuentan con histogramas bimodales, es decir, aquellos histogramas donde los promedios de brillo entre el fondo y los objetos de la imagen se encuentran claramente delimitados, como en aquellas imágenes que contienen objetos oscuros con fondo brillante, u objetos brillantes sobre un fondo oscuro. De esta forma, el objeto consiste en separar concisamente los objetos del fondo de la imagen, para luego etiquetarlos.

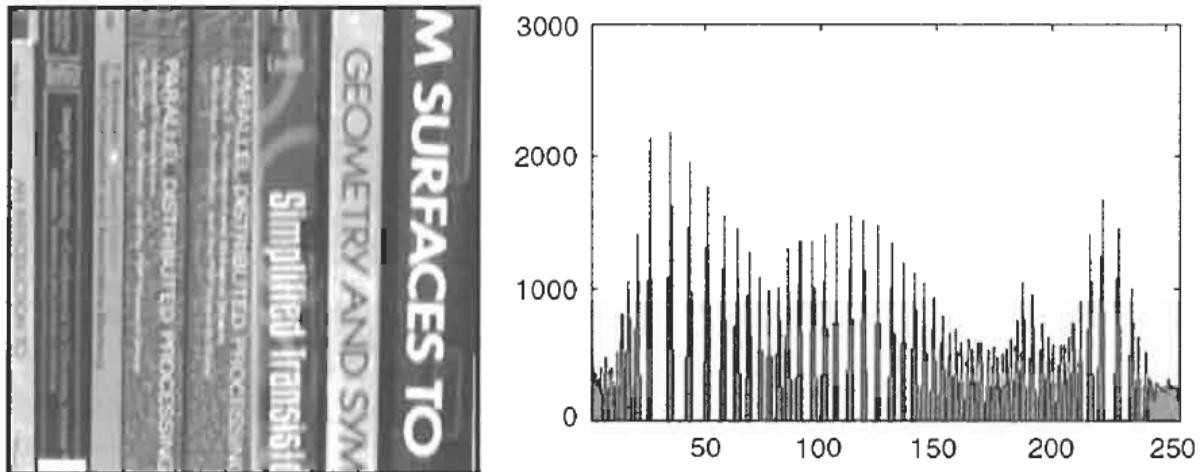


Figura 5.44: Imagen de libros y su histograma luego de aplicar la igualación

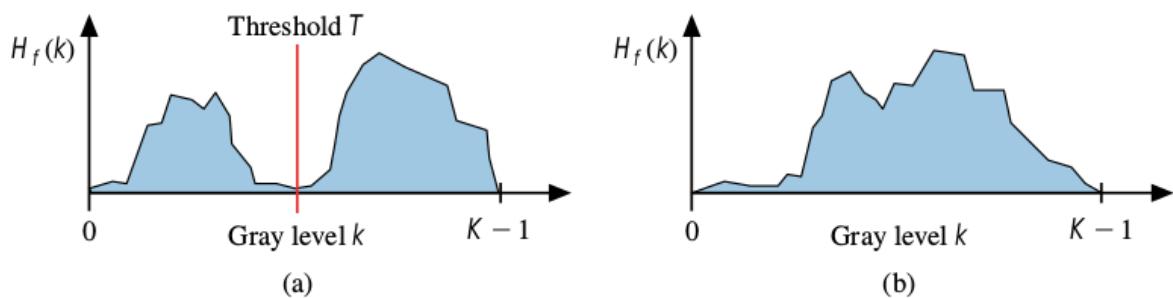


Figura 5.45: Límite de la imagen. En la izquierda se puede apreciar una imagen con niveles de intensidad correctamente delimitados, mientras que en la imagen de la derecha, se puede observar una imagen con un límite poco claro entre objetos y fondo.

Existen varias estrategias para la elección acerca de donde colocar el límite  $T$ : Si el histograma de imagen es bimodal, el límite se tiende a colocar entre medio de los modos de la imagen, como en la figura anterior. Sin embargo, esta aproximación tiene problemas si la imagen contiene múltiples objetos de un brillo promedio diferente en un fondo uniforme(histograma multimodal), excluyéndose algunos objetos. También es difícil asignar un límite si el histograma es plano, contenido imágenes complejas, con variaciones de gris significativas, detalles, iluminación no uniforme, etc.

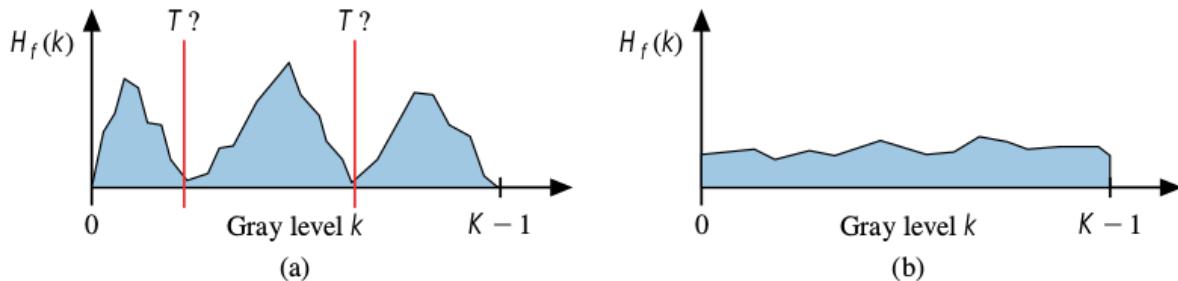


Figura 5.46: (a)Histograma multimodal que señala la dificultad de seleccionar un límite.(b) Histograma plano, para el que la selección de un límite es difícil o imposible.

Por otro lado, también se pueden emplear aproximaciones que usen un modelo estadístico sobre el histograma, con una función de distribución de probabilidad (pdf), donde se plantea la decisión de asignar 0 o 1 a cada pixel de forma estadística. De esta manera, se puede seleccionar la función de distribución que mejor se adapte a las ubicaciones de los modos del histograma(picós de intensidad en éste), el ancho de cada modo y la decisión acerca de donde termina un modo y comienza otro; adiendando aplicarse un modelo probabilístico, dependiendo de la forma de los modos, como por ejemplo, una pdf Gaussiana. Esta alternativa puede producir resultados aceptables con respecto a la colocación de límites, sin embargo cualquier modelo probabilístico simple no tiene en cuenta factores importantes como la continuidad del fondo o de los objetos, apariencia visual, e iluminación no uniforme, por lo que un modelo estadístico no produciría resultados visuales tan eficientes, como los que generaría una persona manualmente.

Un ejemplo de aplicación de esta técnica, son las aplicaciones biomédicas, que permiten la iluminación de los objetos y el fondo, o imágenes microscópicas de una o múltiples células que contienen objetos brillantes sobre un fondo oscuro.

### Especificación(Histogram Matching o Specification)

Este método consiste en realizar un mapeo entre los valores de un histograma de imagen igualado y una función de transformación(con una función de densidad de probabilidad ), de forma que se puedan obtener los valores aproximados de la imagen de entrada, en el dominio de la transformación. El primer paso consiste en aplicar la técnica de igualación de histograma, para obtener valores distribuidos uniformemente, y redondearlos al rango de  $[0, K-1]$ , con  $K$  niveles. A continuación, se debe realizar la computación de la función de transformación  $G()$ , para cada uno de los  $q$ -niveles de intensidad,  $q=0,1,\dots,K-1$ . Los valores resultantes, son escalados y redondeados a sus valores más cercanos en el rango  $[0, K-1]$  y almacenados en una tabla. Esta fórmula define una función de densidad de probabilidad  $p_z(z_i)$ , que es la función de densidad de probabilidad que se desea que la imagen de salida adopte, sobre una variable aleatoria  $z$ :

$$G(z_q) = (L - 1) \sum_{i=0}^q p_z(z_i)$$

Figura 5.47: Fórmula de transformación G

Posteriormente, para cada valor de intensidad en cada pixel del histograma igualado  $sk$ , se emplean los valores almacenados (luego de aplicar  $G()$ ), para encontrar el valor más próximo  $zq$  a  $sk$ ,

dentro del dominio de los valores producidos por  $G()$ , de manera que  $G(zq)$  es el valor más cercano y alinear este mapeo de  $s$  a  $z$ . Si ocurre que más de un valor de  $zq$  satisface la condición de  $sk$  (con un mapeo no único), se elige el valor más pequeño por convención. Finalmente, con estos valores obtenidos se produce el nuevo histograma de imagen con los  $zq$  valores obtenidos, empleando los mapeos almacenados con anterioridad.

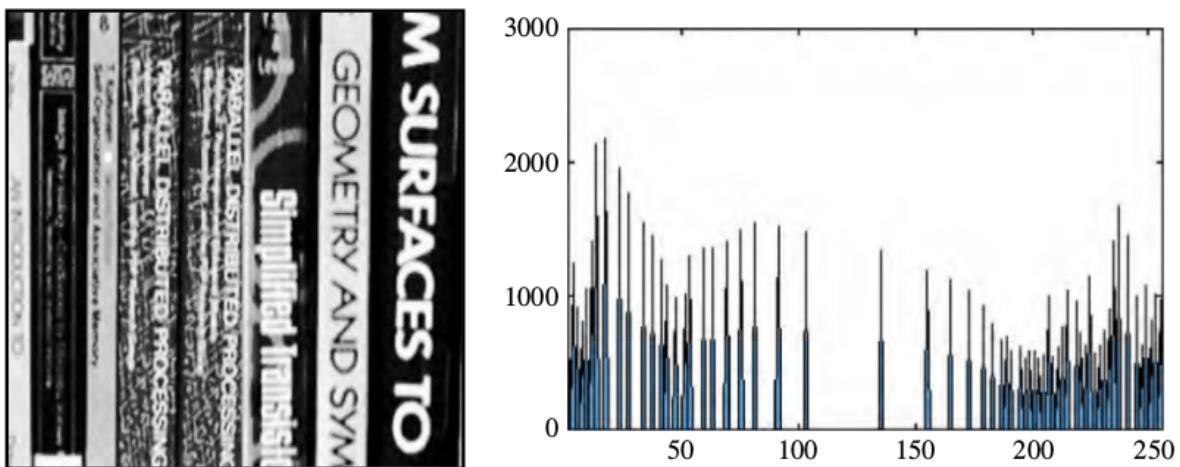


Figura 5.48: Ejemplo de especificación de histograma. Se realiza la especificación de histograma de libros para que se aproxime a una “V” centrada en los niveles de gris y que se extienda a lo largo de la escala de grises, produciendo una imagen con un alto contraste.

### Operaciones aritméticas entre matrices

Debido a que las imágenes se representan como matrices de números, pueden aplicarse operaciones aritméticas entre matrices que operen con los pixeles de éstas, siempre y cuando estas sean de la misma dimensión. Dadas dos imágenes con  $N \times M$  pixeles y representándose los niveles de intensidad de éstas por medio de las funciones  $f(x,y)$  y  $g(x,y)$  con  $x = 0, 1, \dots, M-1$  e  $y = 0, 1, \dots, N-1$  y siendo  $M(x,y)$  la matriz resultante, las operaciones que se pueden aplicar a las matrices se pueden definir de la siguiente manera:

- Suma:  $M(x,y) = f(x,y) + g(x,y)$ . Un ejemplo de aplicación de la operación de suma es la corrección de las imágenes que se encuentran con degradaciones aleatorias o ruido, debido a diversas factores en el ambiente. La técnica más sencilla para eliminarlo, es el modelo de ruido aditivo, donde se considera que una imagen con ruido es la suma de una imagen original y una imagen con ruido, y se supone que el ruido en cada par de coordenadas no está correlacionado y que la media de éste es cero. Así, se puede afirmar que al calcular un promedio de  $R$  imágenes con ruido, tomadas en una rápida sucesión y sin ruido en la escena, la media de este calculo tenderá a cero (una matriz  $N \times M$  con valores cercanos a cero), lo que mejorará el grado de fidelidad con respecto a la imagen original por un factor de  $R$ . Sin embargo, si existen diferencias en la escena o, si existen dependencias entre las imágenes con ruido (en caso de que todas éstas sean casi idénticas), entonces la reducción de ruido será limitada.

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

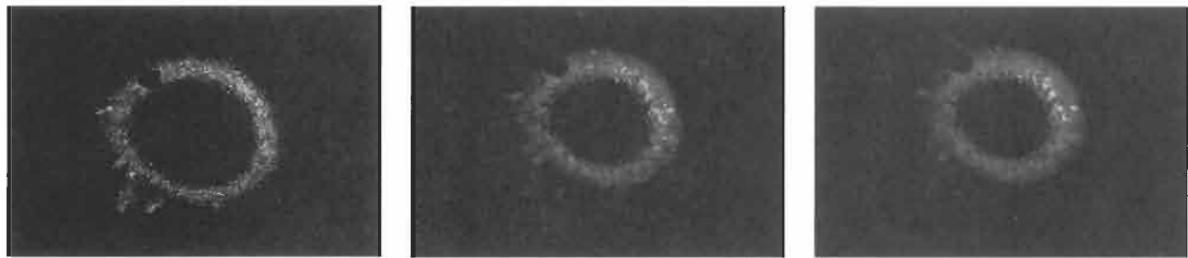


Figura 5.49: Ejemplo de promediado de imágenes. La imagen de la izquierda es una imagen individual con ruido. La imagen del centro es un promedio de 4 imágenes. La imagen de la derecha es un promedio de 16 imágenes.

- Resta:  $M(x,y) = f(x,y) - g(x,y)$ . La diferencia de imágenes es una técnica que se emplea para detectar cambios en imágenes tomadas sobre la misma escena en diferentes momentos, esto permite que se le pueda aplicar para el rastreo de objetos, reconocer el movimiento de objetos, para computar información 3-D del movimiento 2D, en cámaras de vigilancia y, campos de la astronomía donde los bajos niveles de frecuencia introducen ruido en el dispositivo de sensado. De esta forma, para detectar si existe un cambio de imagen significativo se realiza la sustracción de las mismas, y se analiza el histograma de imagen: Si los niveles de intensidad del histograma en el nivel  $n$  no son significativos, significa que no existe una diferencia considerable entre ambas; el contrario, si los valores en un intervalo de niveles  $n$  o en un nivel  $n$  es significativa, se podrá percibir que el histograma tendrá un nivel de intensidad(más brillo) en ese punto. A continuación, se puede observar el histograma luego de realizar la resta entre dos imágenes:

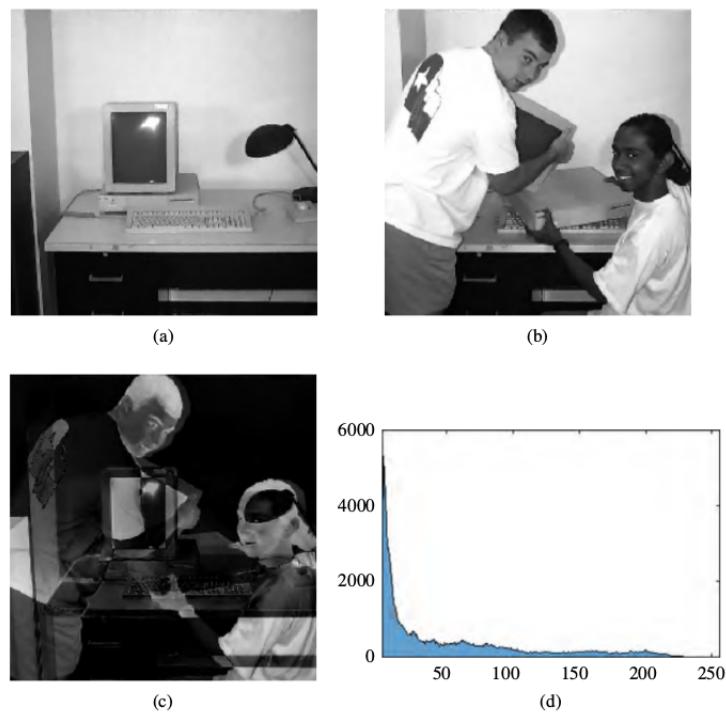


Figura 5.50: Las figuras (a) y (b) son las imágenes originales, la figura (c) es la imagen resultante de la diferencia y la figura (d) es su histograma asociado.

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

- Multiplicación:  $M(x,y) = f(x,y) * g(x,y)$ . Este tipo de operación se emplea con el uso de una máscara para aislar regiones de interés(ROI) en la imagen final. Este proceso consiste en multiplicar una imagen por una máscara de imagen que tiene unos en la región de interés y cero en cualquier otra coordenada. Pueden existir más de una ROI en la máscara de imagen, con una forma arbitraria, aunque las formas rectangulares son usadas frecuentemente por la facilidad de implementación. En la siguiente imagen se puede observar el proceso de enmascaramiento de muelas en una imagen de rayos X:

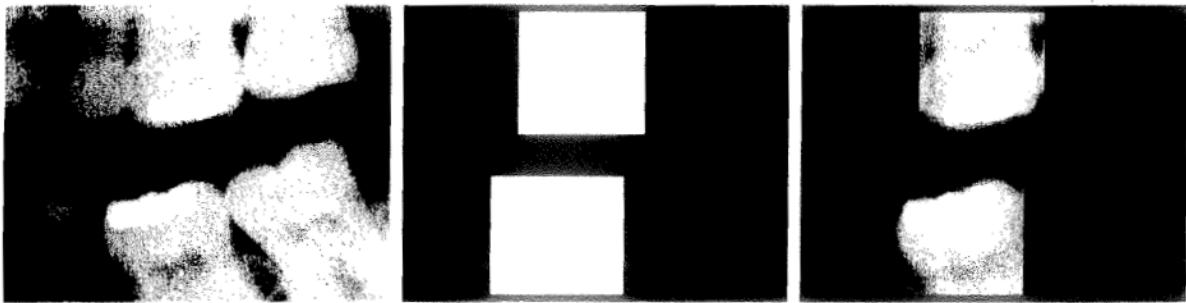


Figura 5.51: Multiplicación de histograma. La figura de la izquierda es la imagen original, la imagen del centro es la máscara de la ROI que aísla muelas (donde blanco corresponde a 1 y negro corresponde a 0), y la figura de la derecha es el producto entre estas dos imágenes.

- División:  $M(x,y) = f(x,y) / g(x,y)$ . Este tipo de operación (en conjunto con la multiplicación) se emplea para la corrección de sombras, ya que si se tiene un sensor que captura una imagen  $g(x,y)$  que puede ser descompuesta en una imagen perfecta  $f(x,y)$  y una función de sombreado  $h(x,y)$  esto es:  $g(x,y) = f(x,y) * h(x,y)$ ; Entonces, se puede obtener la imagen ideal dividiendo:  $g(x,y)/h(x,y)$ .

#### Operaciones geométricas de transformación

Las operaciones geométricas modifican la relación espacial entre píxeles, realizando como primer paso la transformación espacial de las coordenadas de éstos, a nuevas coordenadas en otro sistema de coordenadas, y posteriormente empleando la técnica de interpolación de intensidad, para asignar valores de intensidad a los píxeles transformados espacialmente. El esquema más empleado para definición de los métodos de la transformación de imágenes son las transformaciones afines (affine transformation) que son aquellas transformaciones que conservan la colinearidad entre puntos, líneas rectas y planos, es decir, que todos aquellos puntos que yacen en una línea recta inicialmente aún lo hacen luego de aplicar la transformación al igual que las proporciones en las distancias entre los puntos, lo que significa que si un punto en una línea es el centro en la imagen digital de entrada, lo seguirá siendo en la imagen digital de salida. De esta forma, la combinación de estas transformaciones permite generar operaciones geométricas que actúan sobre cada pixel y producen una nueva imagen de salida.

Dada la coordenada de un pixel en una imagen digital de entrada  $(v,w)$  definida como una coordenada proyectada, es decir, representado como un vector de tres valores  $(v,w,1)$  y siendo la coordenada para un pixel en el nuevo espacio  $(x,y)$ , las transformaciones afines se pueden definir como una matriz de  $3 \times 3$   $T$ , donde dependiendo los valores definidos para los índices  $t_{11}$  y  $t_{32}$ , se puede cambiar el tipo de transformación:

$$[x \ y \ 1] = [v \ w \ 1] \mathbf{T} = [v \ w \ 1] \begin{bmatrix} t_{11} & t_{12} & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & 1 \end{bmatrix}$$

Figura 5.52: Fórmula de la transformación afín  $T$  definida de manera genérica.

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

En las siguientes secciones, se detallan las definiciones y efectos de las transformaciones afines principales.

##### Traslación

La traslación es la operación mas sencilla y consiste en dada una imagen  $g(x,y)$ , desplazar ésta en dirección horizontal y vertical, por medio de la suma de un valor tanto en el eje  $x$  como en el eje  $y$ .

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix} \quad \begin{aligned} x &= v + t_x \\ y &= w + t_y \end{aligned}$$


Figura 5.53: Definición matricial traslación

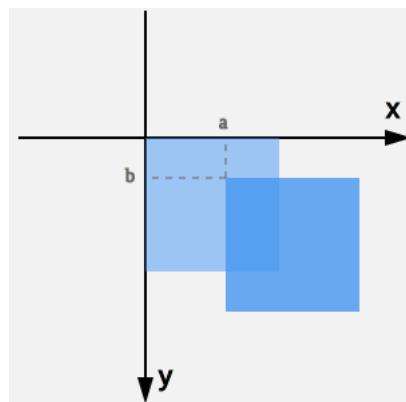


Figura 5.54: Traslación gráficamente

##### Rotación

La rotación consiste en girar una imagen por un ángulo  $\theta$  relativo al eje  $x$ , empleando la siguiente matriz de transformación:

$$\begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{aligned} x &= v \cos \theta - w \sin \theta \\ y &= v \sin \theta + w \cos \theta \end{aligned}$$


Figura 5.55: Definición matricial de la rotación

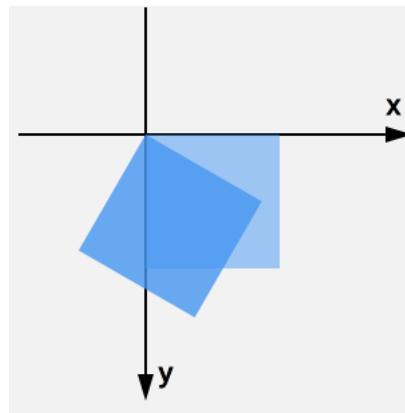


Figura 5.56: Rotación gráficamente

### Escalado

El escalado consiste ampliar o reducir la escala de una imagen, empleando para ello los valores  $cx$  y  $cy$  como factores de escala del eje  $X$  y del eje  $Y$  respectivamente. Si los factores son menores a 1, la imagen se reducirá, mientras que si éstos son mayores a 1 la imagen aumentará su tamaño. Cuando se escala una imagen se modifica tanto la escala como la posición en el plano, por lo que si se desea volverla a posicionar sobre el origen se debe aplicar una traslación.

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

$$\begin{aligned} x &= c_x v \\ y &= c_y w \end{aligned}$$

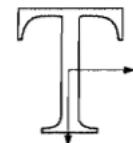


Figura 5.57: Definición matricial del escalado

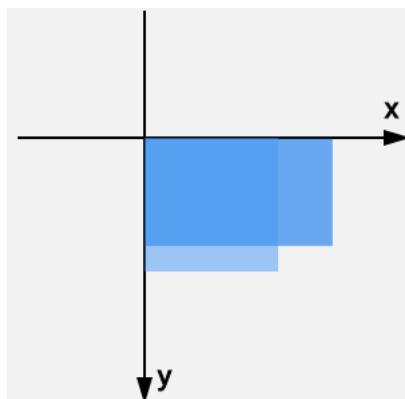


Figura 5.58: Escalado gráficamente. Esta imagen fue escalada por un factor de escala de 1.4 en el eje  $X$  y por un factor de escala de 0.8 en el eje  $Y$ .

Un ejemplo de la aplicación de esta transformación es la técnica de zoom, donde simplemente se escalan cada coordenada de pixel por un valor para el eje  $X$  y otro para el eje  $Y$ , y luego se aplica la interpolación para obtener los niveles de intensidad en la imagen resultante.

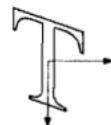
### Inclinación o transvección(Shearing)

La inclinación o transvección, consiste en desplazar los puntos en un eje de manera lineal, por una cantidad proporcional a la coordenada en el eje perpendicular al mismo. Esta transformación puede realizarse de manera horizontal sobre el eje  $X$  (en cuyo caso se desplaza cada punto de este eje por un valor proporcional a su coordenada en  $Y$ , quedando intactos los valores en  $y$  de cada punto), de manera vertical sobre el eje  $Y$  (en este caso las líneas en verticales paralelas al eje  $Y$  se mantienen inalterables, modificándose las líneas paralelas al eje  $X$ ). A continuación se muestra su definición matemática horizontal y verticalmente y un ejemplo de inclinación horizontal:

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

$$x = v + s_v w$$

$$y = w$$



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

$$x = v$$

$$y = s_h v + w$$



Figura 5.59: Definición matricial de la inclinación

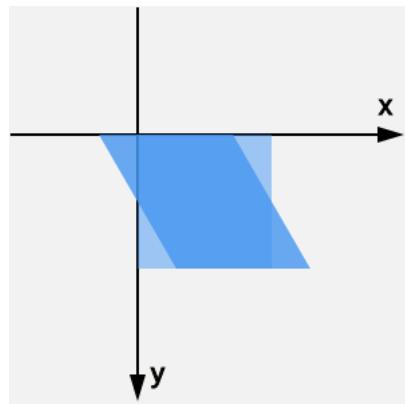


Figura 5.60: Inclinación horizontal gráficamente.

### Interpolación

Una herramienta relacionada con las imágenes digitales es la interpolación, empleada en tareas como hacer zoom, reducción(shrinking), rotación y correcciones geométricas. Esta herramienta consiste en emplear datos conocidos de la imagen de entrada para estimar valores en coordenadas desconocidas. Por ejemplo, si se necesitara convertir una imagen a una escala mayor, la cantidad de pixeles y la correspondencia entre las intensidades diferirían por lo que sería necesario contar con un método que permita la asignación aproximada de intensidades. Un método para realizar ésto es asignar a cada pixel en la imagen mayor, el valor del pixel vecino más cercano si se superpone, esta imagen con la imagen de entrada, este método se conoce como interpolación de vecino más cercano. Existen otros métodos para asignar intensidades que consideran más vecinos y la forma en que consideran éstos sigue alguna fórmula

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

matemática, entre los que se encuentran la interpolación bilineal (donde se emplean los 4 vecinos más cercanos para estimar la intensidad) y la interpolación bicubica (que toma los 16 vecinos más cercanos):

$$v(x, y) = ax + by + cxy + d$$

Figura 5.61: Fórmula de interpolación bilinear

Donde  $v(x,y)$  es la intensidad del pixel en la coordenada  $(x,y)$ , los coeficientes  $a,b,c$  y  $d$  se emplean para determinar los vecinos que de los que se obtendrá la intensidad.

$$v(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j$$

Figura 5.62: Fórmula de interpolación bicubica

#### Técnicas de filtrado espacial

El filtrado espacial consiste en rechazar aquellos elementos en una imagen que no son de interés y conservar aquellos necesarios para el proceso en que se aplica. El filtrado se basa en emplear un filtro (también denominado mascara, kernel, template y ventana o window) que se define como un conjunto de pixeles (o vecindario) sobre un pixel con una coordenada  $(x,y)$  y luego aplicar diferentes operaciones sobre ese conjunto de pixeles (pueden ser operaciones estadísticas para filtros no lineales, como la media, mediana, etc., o operaciones lógicas AND, OR, NOT, NOR), de manera que se obtenga un nuevo valor de intensidad para el pixel del centro en la coordenada  $(x,y)$ . Generalmente, en la práctica el valor del pixel filtrado se asigna a una ubicación en una nueva imagen que se crea para mantener los resultados filtrados durante el proceso, de manera que el contenido de la imagen de entrada no cambie mientras se está realizando el filtrado. La forma en que se organizan los pixeles que forman el filtro dependen del objetivo que se persigue al aplicar éste y se define generalmente por una regla que dependiendo de la cantidad de vecinos que se desea abarcar, aumenta esta cantidad manteniendo la forma original del filtro.

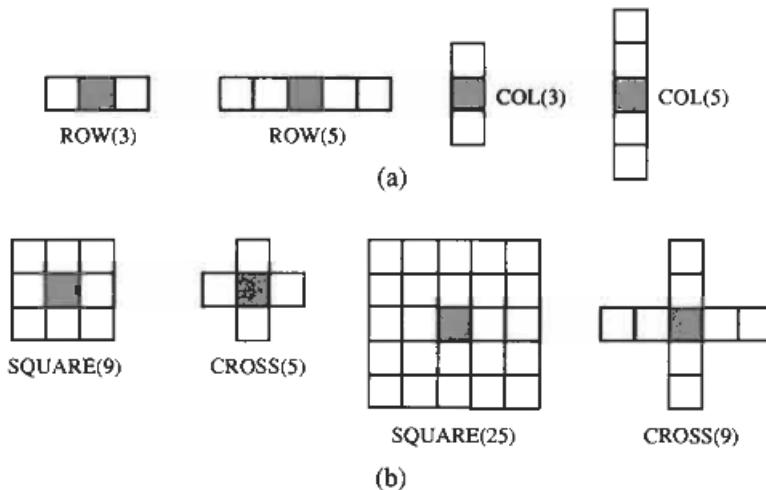


Figura 5.63: Tipos de filtros. (a) Filtros de una sola dimensión con una formula de fila  $ROW=2P + 1$  y de columna  $COL=2P + 1$  para pixeles  $P=1,2$ . (b) Filtros de dos dimensiones con formulas cuadrado  $SQUARE = (2P + 1)^2$  y cruz  $CROSS = 4P + 1$ , para  $P = 1,2$ .

De esta forma, el proceso de filtrado consiste en realizar un desplazamiento de la máscara de filtrado por cada uno de los pixeles de ésta y obtener el valor del pixel central producto de la operación

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

efectuada. Matemáticamente este proceso se puede definir como: Dado un pixel con coordenada  $(x,y)$  en la imagen de entrada de tamaño  $M \times N$ , variando  $x,y$  en el rango de los píxeles de la imagen, una máscara de  $m \times n$ , con una máscara de filtrado rectangular con  $m = 2a + 1$ ,  $n = 2b + 1$  y, siendo los coeficientes de filtrado los valores  $w_{ij}$  que se aplican a las intensidades  $f(x,y)$  de cada pixel de la imagen:

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)$$

Figura 5.64: Definición matemática de la operación de filtrado

La máscara de filtrado requiere que se especifiquen coeficientes de filtrado para cada una de las intensidades del filtrado, lo que puede hacerse especificando valores iguales para todos los píxeles de la máscara, o estos ponderados para asignar más prioridad a ciertos píxeles en la imagen, o cuando se tiene una función estadística obtener la máscara de filtrado en base a ésta, aplicando la formula al vecindario de un pixel y obteniendo como salida, el nivel de intensidad del pixel en la imagen de salida. Con ésta ultima aproximación, solamente se requiere especificar la fórmula matemática y las dimensiones del vecindario para aplicarla. Un ejemplo de esta ultima aproximación, es aplicar la distribución de Gauss en un pixel, donde se considera la varianza de los píxeles y las coordenadas del pixel central  $(x,y)$ , de forma que la aplicación de la máscara de filtrado consiste en tomar muestras y aplicar la formula en distintas ubicaciones:

$$h(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Figura 5.65: Fórmula de Gauss aplicada a un pixel de imagen.

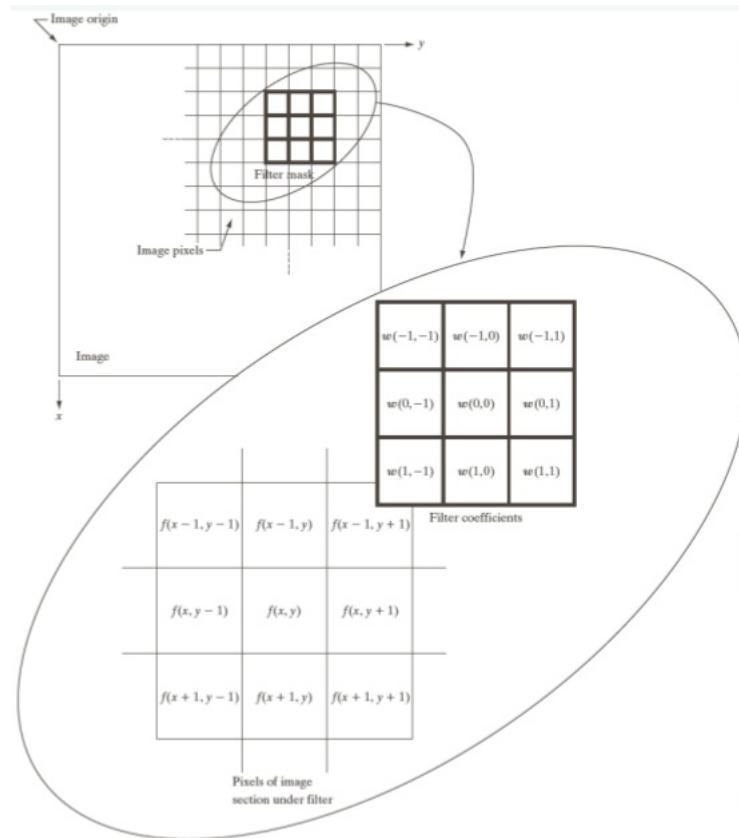


Figura 5.66: Representación gráfica de los elementos considerados para el filtrado espacial

Existen dos tipos de filtros según el tipo de operación que se realiza en ellos: Los filtros lineales que son aquellos donde la operación que se aplica sobre los pixeles de una imagen es una operación lineal, y los filtros no lineales. Una operación lineal es aquella donde se cumplen las propiedades de homogeneidad y adición para una imagen de entrada  $f(x,y)$  y una imagen de salida  $g(x,y)$ :

- La homogeneidad ocurre cuando al multiplicar la salida de un operador aplicado a la imagen produce el mismo resultado, que multiplicar esa constante en la imagen de entrada y luego aplicar el operador.
- La adición que consiste en que la salida de una operación lineal a la suma de dos entradas, es lo mismo que aplicar la operación lineal a las entradas y luego realizar la suma de las mismas.

Los filtros pueden ser aplicados tanto en el dominio espacial o en el dominio de las frecuencias, abarcando un diverso rango de aplicaciones, aunque principalmente se destacan: La mejora de la imagen, donde se intenta mejorar la calidad de imagen en algún aspecto para interpretación artificial o humana por medio del suavizado de imagen;  La restauración de imágenes, donde se intenta recuperar información de una imagen degradada con conocimiento del proceso de degradado, empleando filtros que se pueden aplicar en el dominio espacial y en el dominio de las frecuencias.

El suavizado de imagen, aplicado en técnicas de pre-procesamiento de imagen (como remover detalles antes de la segmentación de objetos), es un ejemplo del uso tanto de filtros lineales como no lineales, ya que durante esta tarea se busca reducir el ruido y en suavizar la imagen (blurring). Un ejemplo de filtro lineal para la reducción de ruido, es aplicar el promediado de los pixeles vecinos definidos por la máscara en una imagen de entrada. Debido a que el ruido en una imagen ocurre cuando existen transiciones abruptas en niveles de intensidad entre pixeles vecinos, el promediado provoca una disminución en la diferencia de niveles, sin embargo, debido a que los contornos de las figuras se caracterizan por ésto, también produce que los bordes se vean poco nítidos. Otras aplicaciones de este tipo de filtro de suavizado, es reducir el detalle irrelevante, o la disminución de contornos falsos producto de niveles de intensidad insuficientes. A continuación, se puede observar el efecto de suavizado empleando el filtro de promedio de vecinos:



Figura 5.67: Efecto de suavizado. La imagen original se sitúa en la parte superior izquierda. La imagen superior derecha tiene un filtro rectangular con una máscara de  $m = 3$ . La imagen inferior izquierda con un filtro rectangular con  $m = 5$ . La imagen inferior derecha tiene un filtro rectangular con  $m = 9$ .

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

Alternativamente, se pueden emplear filtros no lineales estadísticos para el suavizado de imagen, cuya respuesta se basa en ordenar los pixeles contenidos en el área de la imagen abarcada por el filtro, y reemplazar el valor del pixel del centro con el valor determinado por el resultado del ordenamiento. Existen varios tipos de filtros para suavizado que se basan en distintos valores (máximo, mínimo), no obstante el ejemplo más relevante es el filtro que emplea la mediana (valor del conjunto para el cual la mitad de los valores son mayores o iguales a la misma y el resto son menores) de los valores de intensidad en el pixel (incluyendo el valor del pixel en el cálculo), donde se lo obtiene y luego se asigna este valor como el valor de intensidad del pixel de salida. Este tipo de filtro es empleado debido a que produce excelentes resultados para la reducción de ruido aleatorio, con respecto a los lineales del mismo tamaño. A continuación se destaca el efecto del suavizado de imagen empleando un filtro de mediana y un filtro de promediado lineal:

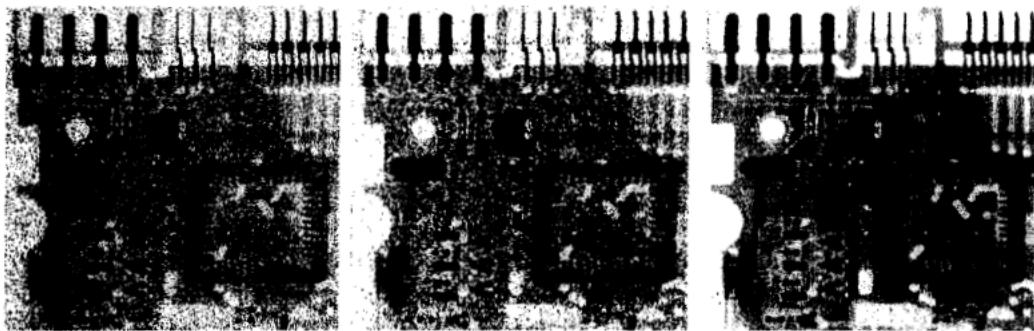


Figura 5.68: Filtro de mediana vs filtro lineal. La imagen izquierda es un circuito electrónico con ruido. La imagen central es el resultado de la aplicación de un filtro lineal de 3x3. La imagen derecha es el resultado de aplicar un filtro no lineal de mediana de 3x3.

#### Técnicas sobre el dominio de las transformaciones

Las operaciones que se ejecutan sobre el dominio de las transformaciones hacen uso de la transformada de Fourier para llevar las funciones al dominio de las frecuencias, efectuar operaciones sobre este dominio y finalmente, obtener la imagen de salida aplicando la función inversa a esta transformación. La transformada de Fourier es una técnica altamente empleada en el procesamiento de imágenes, ya que permite realizar operaciones que de alta complejidad que requerirían un alto tiempo de procesamiento si no se empleara esta técnica, y realizar tareas asociadas al procesamiento de imagen de manera más eficiente que empleando filtros lineales (cuando estos son de grandes dimensiones). Esta técnica se emplea en un amplio espectro de tareas tales como la mejora de video, restauración, compresión, segmentación y métodos que emplean la técnica de wavelets. La transformada de Fourier se desprende de las series de Fourier, propuestas por el matemático Francés Jean Baptiste Joseph Fourier 1822 para ser aplicadas en el campo del flujo del calor, donde se expresa que sin importar lo compleja que sea una función, si ésta es periódica se puede expresar matemáticamente como la suma de senos y cosenos de diferentes frecuencias, multiplicados cada uno por un coeficiente diferente. De esta forma, la transformada de Fourier\*(FT)\* extiende este concepto afirmando que, aquellas funciones cuya área debajo de la curva es finita y no necesariamente son periódicas, puede expresarse para las variables continuas, como la integral de los senos y/o cosenos multiplicados por una función de ponderación; siendo ser ésta, reconstruida o recuperada completamente por el proceso inverso, sin pérdida de información. Esta característica permite trabajar sobre el dominio de Fourier y luego retornar al dominio espacial de una imagen sin perder información.

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

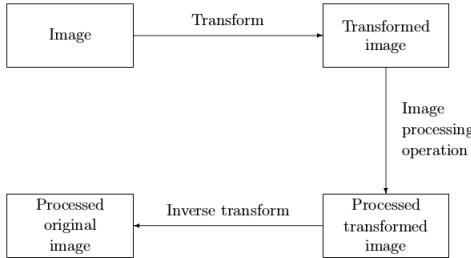


Figura 5.69: Esquema del dominio de las transformaciones

Debido a que para representar una imagen digital se debe trabajar con valores discretos, la transformada de Fourier se debe definir de manera discreta (*DFT* Discrete Fourier Transform), donde ésta se define como una función  $F(u,v)$  sobre los valores  $(u,v)$  que representan las frecuencias con que se representará la imagen, definiendo  $u$  como la frecuencia de oscilación a lo largo del eje  $X$  y  $v$  la frecuencia de oscilación a lo largo del eje  $Y$  (expresadas en ciclos por pixel) que se emplearan para representar la imagen definida en el dominio espacio-temporal sobre éste dominio:

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)}$$

Figura 5.70: Fórmula de la Transformada de Fourier Discreta

Además, esta función toma valores discretos  $x, y$  que son las coordenadas de los pixeles para una imagen digital de  $M \times N$ , con  $u = 0, 1, 2, \dots, M-1$ ,  $v = 0, 1, 2, \dots, N-1$  junto con su inversa que permite obtener la imagen en el dominio espacial partiendo de la matriz de frecuencias definida formalmente como:

$$f(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M + vy/N)}$$

Figura 5.71: Formula de la Transformada de Fourier Inversa

Esta formula cuenta con distintas propiedades entre las que se destacan:

- Magnitud o espectro(Espectro de frecuencias): El espectro consiste en aplicar para cada uno de los elementos de la frecuencia de la matriz la siguiente formula:

$$|F(U, V)| = \sqrt{R^2(U, V) + I^2(U, V)}$$

Figura 5.72: Fórmula para el calculo del espectro 

De manera que se obtenga una matriz de magnitud de frecuencias de la imagen del mismo tamaño de la imagen digital  $M \times N$  (con la amplitud de las mismas), donde los vectores se representan con números complejos  $C = R + jI$ , correspondiendo  $R$  a la parte real del mismo y, siendo  $I$  la parte imaginaria (donde  $j$  es la raíz cuadrada de -1) y donde el conjunto de números reales se representan como números complejos con  $I = 0$ . Los componentes del espectro de la DFT determinan las amplitudes de las sinusoidales que componen la imagen, almacenando información acerca de las intensidades en la imagen, por lo que en cualquier frecuencia dada de una imagen, una gran amplitud implica una mayor relevancia de la onda sinusoidal para esa frecuencia; mientras que si se cuenta con una pequeña amplitud, implica en menor medida la presencia de una onda sinusoidal en esa frecuencia.

- Angulo de fase (Fase). El ángulo de fase para cada elemento de la matriz se obtiene por medio de la siguiente fórmula:

$$\phi(u, v) = \arctan\left[\frac{I(u, v)}{R(u, v)}\right]$$

Figura 5.73: Fórmula para el cálculo del ángulo de fase 

El angulo de fase o fase, es una medida del desplazamiento de varias ondas sinusoidales con respecto al origen, por lo que este arreglo contiene los ángulos que contienen información respecto de donde los objetos se encuentran localizados en la imagen.

- Espectro potencia. El espectro potencia se calcula para cada elemento de la imagen con la siguiente fórmula:

$$\begin{aligned} P(u, v) &= |F(u, v)|^2 \\ &= R^2(u, v) + I^2(u, v) \end{aligned}$$

Figura 5.74: Fórmula para el espectro potencia 

- Simetría. La simetría con respecto a la transformada de Fourier discreta, enuncia que la magnitud del espectro es simétrica par respecto del punto central por lo que se cumple la siguiente igualdad:

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

$$|F(U, V)| = |F(-U, -V)|,$$

Figura 5.75: Formula simétrica par en el espectro de magnitud 

Mientras que el angulo de fase es simétrica impar con respecto al origen, lo que significa que se cumple la siguiente igualdad:

$$\angle F(U, V) = -\angle F(-U, -V),$$

Figura 5.76: Formula asimétrica par en la fase 

- Traslación. La propiedad de traslación implica que al multiplicar la imagen  $f(x,y)$  por la parte exponencial de la transformada de Fourier en coordenadas  $(u_0, v_0)$ , provoca un desplazamiento de la transformada al punto  $(u_0, v_0)$ , y en consecuencia, multiplicar  $F(u, v)$  por el negativo de esa exponencial cambia el origen de  $f(x,y)$  hacia  $(x_0, y_0)$ .

$$f(x, y) e^{j2\pi(u_0x/M + v_0y/N)} \Leftrightarrow F(u - u_0, v - v_0)$$

$$f(x - x_0, y - y_0) \Leftrightarrow F(u, v) e^{-j2\pi(x_0u/M + y_0v/N)}$$

Figura 5.77: Formulas de traslación DFT.

- Rotación. La rotación implica que al multiplicar la imagen en el dominio espacial  $f(x,y)$  por un angulo  $A$ , rota la función en el dominio de las frecuencias  $F(u,v)$  por el mismo ángulo. Del mismo modo, rotar  $F(u,v)$  por  $A$ , rota la imagen en el dominio espacial  $f(x,y)$  por el mismo ángulo.
- Teorema de la convolución. La convolución (representada por  $\star$ ) se emplea en el filtrado de imágenes, y consiste en rotar el filtro 180° y luego aplicarlo pixel a pixel por la imagen digital de entrada. De esta forma, la convolución en el dominio espacial entre una imagen  $f(x,y)$  y una transformación  $g(x,y)$  equivale a realizar la convolución en el dominio de las frecuencias de las funciones  $F(x,y)$  y  $H(x,y)$  respectivamente.

$$f(x, y) \star g(x, y) \Leftrightarrow F(u, v) * H(u, v) \quad (5.8)$$

De la misma forma, el producto en el dominio espacial de una imagen  $f(x,y)$  y una transformación  $g(x,y)$ , es equivalente a aplicar la convolución entre las funciones del dominio de frecuencias  $F(x,y)$  y  $G(x,y)$ .

$$f(x, y) * g(x, y) \Leftrightarrow F(u, v) \star H(u, v) \quad (5.9)$$

Cuando se quiere convertir una imagen digital al dominio de las frecuencias, como primera medida se debe aplicar una función de centrado del espectro, multiplicando cada uno de las coordenadas de la imagen por  $(-1)^{\hat{x}+y}$ , esto permite que el espectro de la imagen este centrado. Adicionalmente, se puede aplicar una transformación logarítmica  $(1 + \log(|F(u,v)|))$  con el fin de aumentar la intensidad de los valores de la matriz de magnitud. Luego, se procede a aplicar la fórmula de Fourier discreta para cada una de las coordenadas en el dominio de la imagen digital con  $M \times N$  pixeles, para obtener las matrices de magnitud y de fase, que tendrán las mismas dimensiones que la imagen de entrada. Aunque estas no contienen fácilmente asociable a las coordenadas de la imagen digital, se pueden visualizar igual que las anteriores:

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

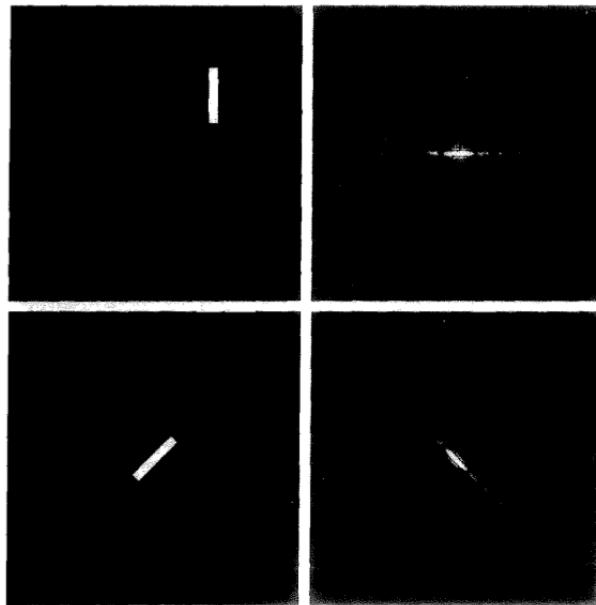


Figura 5.78: Gráficos correspondientes a un rectángulo y su matriz de magnitud. Imagen superior derecha: Imagen original de un rectángulo. Imagen superior izquierda: Magnitud con las frecuencias. Imagen inferior izquierda: Imagen original rotada. Imagen inferior derecha: Efecto de la rotación en el dominio de las frecuencias.

Debido a que los valores de frecuencia  $F(u,v)$  son modificaciones de los valores de intensidad de la imagen con términos exponenciales, la correspondencia entre objetos y sus frecuencias es compleja, sin embargo observando el gráfico anterior se puede observar que la frecuencia está directamente asociada a las tasas de cambios de intensidad en el dominio espacial, por lo que se las puede relacionar con los patrones de intensidad de una imagen. De esta forma, aquellos componentes con una frecuencia que varíen más lentamente son proporcionales al promedio de valores de frecuencia y se encuentran localizados en el centro del gráfico, y a medida que se desplazan los puntos desde este origen, las bajas frecuencias representan componentes cuya variación de intensidad es lenta en la imagen, pudiendo corresponderse éstos, por ejemplo, con las variaciones asociadas a paredes o pisos; mientras que aquellas frecuencias que se encuentran más alejadas del origen, y poseen un valor más alto, se corresponden con cambios de intensidad más altos en la imagen, como los bordes de los objetos.

Un ejemplo de aplicación del proceso de la transformada Fourier es el filtrado de elementos durante el suavizado de imagen, donde dada una imagen  $f(x,y)$  de  $M \times N$  pixeles que necesita ser de un tamaño mayor para ser filtrada, se debe llenar esta imagen con ceros con los pixeles  $P = 2*M$ ,  $Q=2*N$  (por lo que la imagen pasa a tener  $P \times Q$  pixeles). A continuación, se procede a centrar la transformación multiplicando por  $(-1)^{x+y}$  y a computar TFD de esta matriz  $F(u,v)$  y se aplica el filtro  $H(u,v)$  de tamaño  $P \times Q$  a la matriz de intensidad  $G(u,v)=H(u,v)*F(u,v)$ . Finalmente, se procede a obtener la imagen  $g(x,y)$  en el dominio espacial, aplicando la función inversa de la transformada de Fourier y centrándola de nuevo con:  $g(x,y) = [IFTD(G(u,v))](1)^{-x-y}$ , y se recorta esta imagen extrayendo la región  $M \times N$  de la imagen. En la siguiente figura se puede apreciar el proceso de transformación, ubicándose las figuras de manera descendente de izquierda a derecha:

#### Tipos de imágenes digitales

Existen distintos tipos de imágenes digitales según la metodología seleccionada para representar la intensidad, entre los que se destacan los siguientes tipos: Imágenes binarias, imágenes por escalas de grises, imágenes a color e imágenes indexadas.

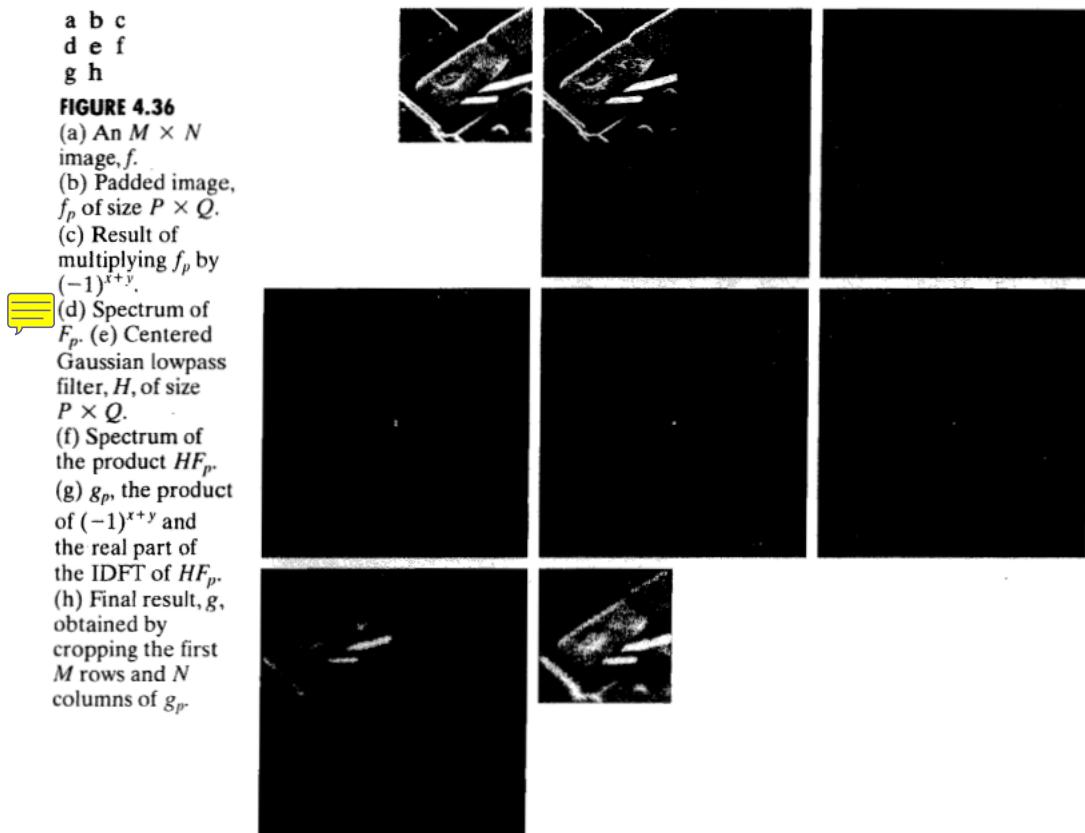


Figura 5.79: Ejemplo de transformada de Fourier aplicada al suavizado de imagen.

### Imágenes por escala de grises

Este tipo de imágenes se representa por medio de un conjunto de valores, que abarcan distintas tonalidades de grises desde blanco hasta negro, representándose cada pixel con 8 bits. |

Existen distintos tipos de operaciones que pueden realizarse sobre imágenes con escalas de grises, aunque se pueden clasificar de manera general en: Operaciones de puntos, operaciones aritméticas y operaciones geométricas. Las operaciones de puntos son aplicadas a los pixeles individuales de una imagen, por lo que las interacciones y las dependencias entre pixeles vecinos no son consideradas, ni las operaciones que toman un conjunto de pixeles, sino que se basan en el procesamiento de las intensidades de los pixeles. Por lo tanto, este tipo de operación no altera la posición de los objetos en la imagen, sino que modifican la apariencia general de la imagen, cambiando la distribución de grises de la misma, obteniendo el negativo o, desplazando los niveles de grises para aclarar la imagen.

Las operaciones aritméticas se realizan entre imágenes de las mismas dimensiones espaciales, este tipo de operaciones es similar a las operaciones por puntos debido a que la información espacial no es considerada, sino que la información se comparte entre imágenes y se ejecutan pixel por pixel. Este tipo de operaciones se emplea para la reducción del ruido en la imagen (distorsiones aleatorias en la imagen producidas por radiación antes de capturar la misma o por fallos eléctricos en el dispositivo de sensado), donde se realiza un promediado de las tonalidades de grises de un conjunto de frames y el resultado final es una imagen cuyo nivel de ruido ha sido reducido considerablemente. Otra área donde se emplean operaciones de este tipo es en la detección de movimiento en sistemas de vigilancia, o en sistemas automatizados de inspección visual, donde se realiza la diferencia entre las matrices que representan las imágenes y luego se computa el histograma de imagen, que mostrará variaciones importantes en la intensidad (valores de brillo mayores) si cambios significativos han ocurrido entre dos frames.

Finalmente, las operaciones geométricas son operaciones complementarias a las operaciones por puntos debido a que no modifican los valores de los niveles de gris, sino que modifican las posiciones

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

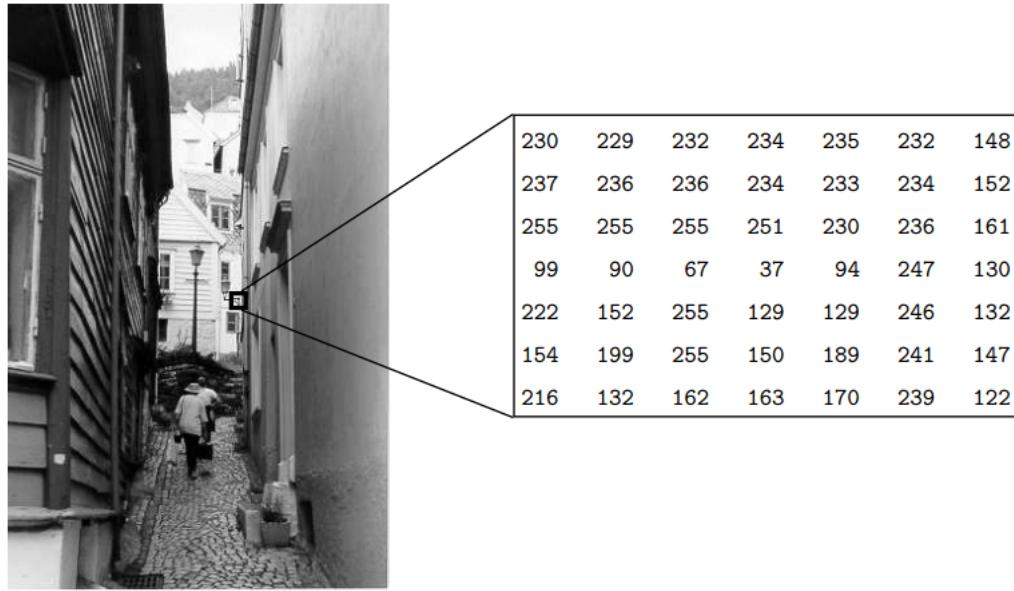


Figura 5.80: Representación de imagen en escala de grises

de los elementos de la imagen. Este tipo de operaciones se emplea para realizar la rotación, traslación o zoom-in o zoom-out en la imagen.

#### Imágenes binarias

En este tipo de imagen digital la intensidad de los pixeles sólo puede asumir dos valores 0 o 1, por lo que sólo se requiere un bit para su representación y los objetos se representan como una secuencia de 1 conectados, siendo éstas las que requieren menos espacio y tiempo de almacenamiento. Estas imágenes contienen suficiente información respecto de los objetos en la imagen y permiten que éstos se reconozcan fácilmente. Este tipo de imágenes se emplean en distintos tipos de aplicaciones de visión por computadora, como el reconocimiento de objetos, el rastreo, etc. aunque su aplicabilidad es limitada debido al contenido limitado de información que brindan. Las imágenes binarias surgen de una variedad de fuentes, generalmente son creadas por medio del procesamiento de imágenes de escala de grises, aunque algunos tipos de sensores entregan una imagen binaria como salida, como los dispositivos que se emplean para obtener dibujos o texto escrito a mano con un pad resistivo, un lápiz de luz. Generalmente estos dispositivos, inicializan todas las coordenadas de la imagen binaria en cero, y al detectar la presión o un cambio de resistencia, o luz sensada en una coordenada, entonces se le asigna a la misma el valor 1. Ejemplos de imágenes binarias, son los dibujos de líneas, texto escrito o impreso, siluetas, huellas digitales, planos empleados por arquitectos.

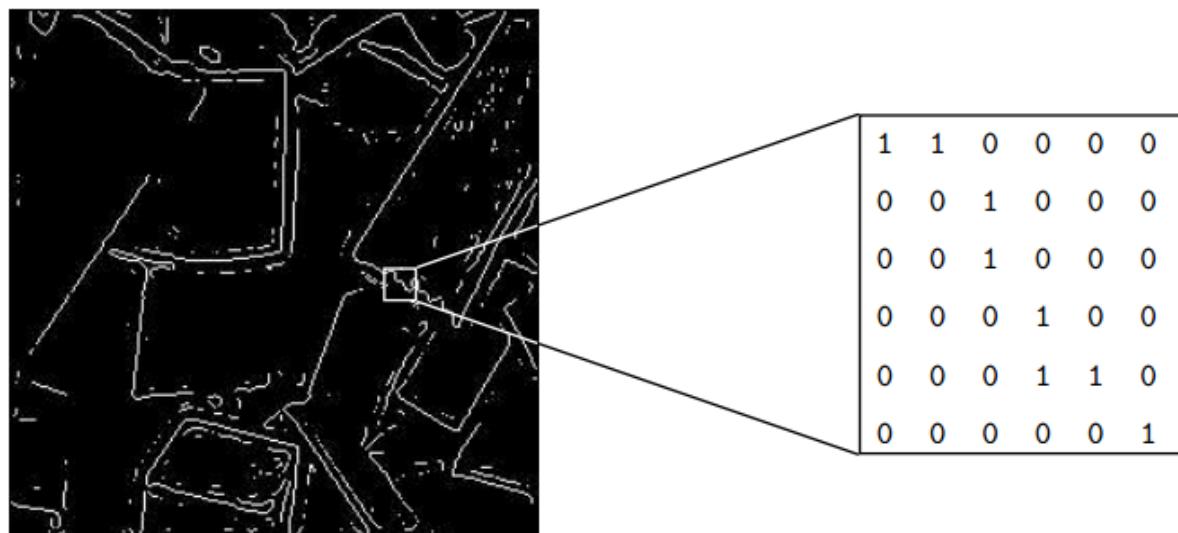


Figura 5.81: Imagen binaria

### Imágenes a color

Con la finalidad de incluir el color en el procesamiento de imágenes, se debe emplear un modelo de color que permita la especificación de las intensidades de los colores en un sistema de coordenadas y un rango de valores dentro de ese sistema de coordenadas, donde cada color sea representado por un único valor. Debido a la variedad de campos de aplicación del color, existen diferentes esquemas de representación según el objetivo, entre los que se encuentran:

- **RGB**(Red, Green, Blue). Emplea diversas combinaciones de colores primarios(normalizados entre [0,1] o alternativamente valores en el rango 0-255) para la representación de colores en imágenes. Este modelo es utilizado principalmente por monitores a color y cámaras de video y para la manipulación y generación de imágenes digitales.
- **CMY**(Cyan, Magenta, Amarillo) y **CMYK**(Cyan, Magenta, Amarillo, Negro). Hace uso de los colores secundarios para representar el color, y es empleado para la impresión de imágenes color, realizándose una conversión interna del esquema RGB a CMY/K.
- **HSI**(Tonalidad, Saturación, Intensidad). La tonalidad es un valor que describe el nivel de pureza de un color (rojo, verde o azul) percibido por un observador, la saturación brinda una medida del grado en que la luz blanca esta mezclada con la tonalidad de un color, y, el brillo es una medida subjetiva que abarca la noción de la intensidad en imágenes sin color. Este modelo hace uso de estas características y además permite desacoplar la información de color y el componente de intensidad.

El esquema empleado para la representación de imágenes digitales comunmente es RGB, donde se emplea un vector para representar la composición de colores, de manera que cada pixel tenga asociadas las combinaciones correspondientes de colores primarios (RGB), utilizando para cada color una representación de 8 bits. De lo tanto, una imagen a color emplea 24 bits por cada pixel, necesitándose un total de  $(2^8)^3 = 16,777,216$  valores posibles de color. De esta forma, para una imagen con  $N \times M$  elementos, existe un vector que contiene la intensidad asociada a cada color primario, que se corresponde con las coordenadas  $(x,y)$  de la siguiente forma:

$$\mathbf{c}(x, y) = \begin{bmatrix} c_R(x, y) \\ c_G(x, y) \\ c_B(x, y) \end{bmatrix} = \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix}$$

Figura 5.82: Vector de color para una coordenada en la imagen

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

Por lo tanto, la representación de una imagen de colores se reduce a realizar combinaciones entre el vector RGB de cada pixel:

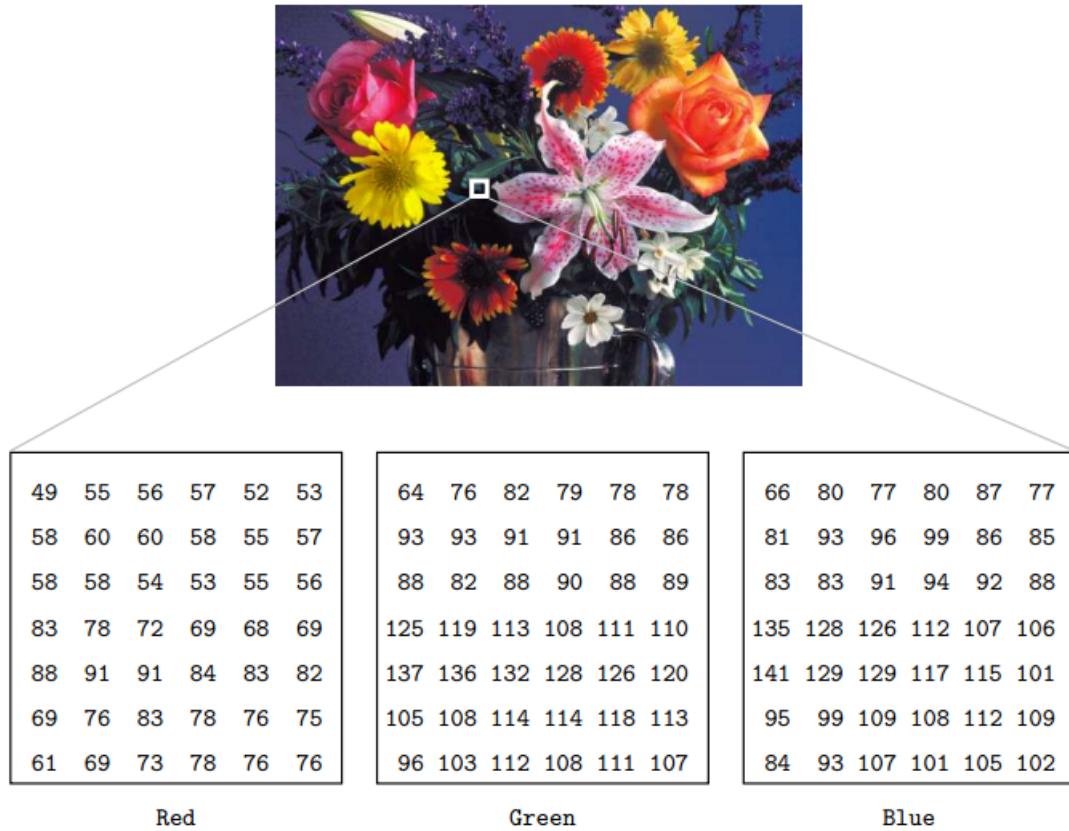


Figura 5.83: Representación de una imagen digital a color

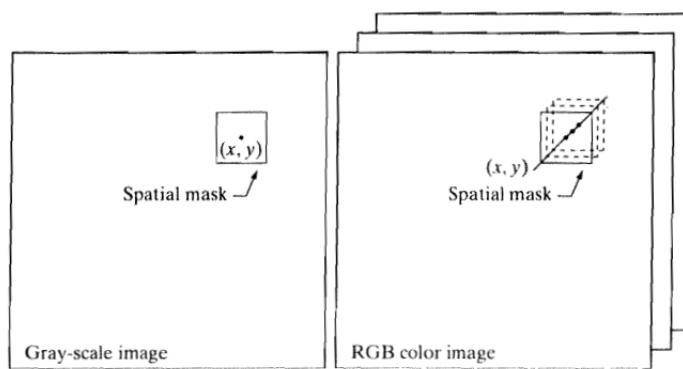


Figura 5.84: Representaciones de imagen en escala de grises vs imagen de color

Existen varias aproximaciones para afrontar el procesamiento de imágenes de color, aunque se pueden clasificar en 2 grupos generales: Aquellas aproximaciones que procesan cada componente de la imagen individualmente, para luego formar la imagen de salida con éstos; aquellas aproximaciones que trabajan con los colores de los píxeles en la imagen directamente. Este tipo de imágenes pueden ser procesadas con algunos tipos de técnicas que se emplean con las imágenes de escala de grises, mientras que algunas tienen que ser modificadas para ser aplicadas sobre las bandas de color individuales.

### Imágenes indexadas

En general las imágenes indexadas solo emplean un subconjunto pequeño de los 16 millones de colores, por lo que para mejorar la eficiencia de almacenamiento, la imagen puede tener asociado un mapa de color o paleta de colores, la cual es solamente un listado de todos los colores en la imagen. Así, cada pixel tiene un valor que no da su color, como en la imagen RGB, sino que es un índice al color en el mapa. Este tipo de imágenes digitales, se emplea en algunos formatos donde la cantidad de colores permitidos para una imagen es de 56 colores o menos, como en el formato GIF.



Figura 5.85: Imagen de color indexada

### Procesamiento de imágenes

Debido a que el procesamiento de imágenes abarca varios tipos de imágenes, comprendidas a lo largo de todo el espectro electromagnético, éste tiende a solaparse con otras áreas como el análisis de imágenes, basado en la extracción de información de utilidad desde la imagen, y la visión artificial, que es un área de la inteligencia artificial cuyo objetivo es lograr que una computadora adquiera conocimiento y pueda efectuar decisiones, basada en imágenes o video de entrada. Así, el procesamiento de imágenes digitales, es un conjunto de técnicas que toman una imagen como entrada y, por medio de una computadora, producen una imagen de salida con el objetivo de extraer información y reconocer objetos en ésta.

Las herramientas y técnicas definidas anteriormente, se emplean en distintos tipos de operaciones que se pueden clasificar según:

- \* El nivel de abstracción
- \* El objetivo perseguido

Dependiendo del nivel de abstracción los procedimientos de procesamiento de imágenes se pueden clasificar en distintos tipos:

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

- Procesos de bajo nivel, que aceptan imagen como entrada, y ejecutan operaciones primitivas sobre ésta como la reducción de ruido, mejora del contraste y aplicación de filtros sobre la imagen para mejorar alguna característica (como aumentar el brillo entre áreas oscuras y con brillo), y producen una imagen modificada como salida.
- Procesos de nivel medio, que abarcan tareas como la segmentación (división de la imagen en partes), descripción de dichos objetos para reducirlos a una forma aceptable para el procesamiento por computadora, y el reconocimiento de objetos individuales(o clasificación de objetos). Estos procesos se caracterizan por el hecho de que las entradas son generalmente imágenes, pero sus atributos son atributos extraídos de una imagen, tales como: bordes, contornos, objetos individuales,
- Procesos de alto nivel, que involucran generar conocimiento a partir de estos objetos ensamblados (empleando mecanismos de machine learning) y efectuar operaciones relacionadas con el análisis de imágenes y, algunas operaciones relacionadas con el campo de visión por computadora.

Alternativamente, los tipos de operaciones en el procesamiento de imagen según el objetivo que se persiga se pueden clasificar en:

- Aplicación de filtros y mejora imagen. Estas técnicas buscan manipular la imagen, de manera que el resultado sea mas adaptable que la original para una aplicación específica. Esto se hace con el fin de recuperar detalles que no se visualizan debido al bajo nivel de brillo, o simplemente subrayar ciertas características de interés en una imagen. Un ejemplo de este tipo de operaciones es aumentar el contraste para mejorar visualmente los objetos que se perciben en la imagen.
- Restauración y reconstrucción de la imagen. Este tipo busca mejorar la apariencia la imagen, sin embargo a diferencia de la mejora de imagen, esta técnica es objetiva debido a que se basan en modelos probabilísticos de degradación de imagen, mientras que la primera se basa en la subjetividad del observador para establecer una mejora adecuada. La restauración intenta recuperar una imagen que ha sido degradada empleando un conocimiento previo del fenómeno de degradación, por lo que estos procesos se encuentran orientados hacia el modelado de la degradación y la aplicación del proceso inverso, para recuperar la imagen original.
- Wavelets. Las Wavelets son un conjunto transformaciones que forman parte del área de procesamiento de señales e imágenes denominado teoría de multiresolución, que abarca un conjunto de técnicas, incluyendo la división de bandas de una señal (subbanding), filtrado de voz digital y representación piramidal de una imagen. Esta rama se relaciona con la representación de imágenes (o señales) en más de una resolución, con el fin de obtener a una resolución específica, características que no se podrían identificar en otra resolución. Este tipo de herramienta es utilizada ampliamente para la compresión y la representación piramidal de una imagen. Esta última fue originalmente aplicada para la visión artificial y la compresión de imágenes, y consiste en subdividir una imagen con el fin de obtener una colección de imágenes de menor resolución organizadas para su posterior procesamiento.

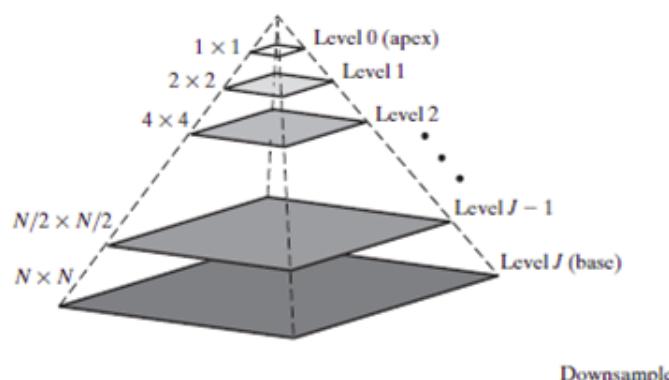


Figura 5.86: Representación piramidal para un arreglo de imagen de 2-Dimensiones de  $N \times N$

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

- Compresión. El objetivo de la compresión (o coding) es representar una imagen digital con la cantidad mínima de bits sin pérdida de información, persiguiendo así, la aceleración de la velocidad de transmisión y reducción del ancho de banda necesario para transmitir las mismas y la minimización del espacio requerido para almacenarlas, manteniendo a la vez el fidelidad de la misma. La compresión es posible porque existe una redundancia presente en las imágenes, que es proporcional a la cantidad de correlación entre las muestras de datos. Por ejemplo, en imágenes estáticas existe un alto grado de correlación en los alrededores de un pixel, mientras que en los videos ésta se encuentra presente entre frames sucesivos del mismo. De esta forma para lograr un grado eficiente de compresión, estos métodos intentan remover los datos redundantes como así también, aquellos que se perciben pero son irrelevantes, produciendo que las imágenes de entrada y salida sean visualmente iguales, y no necesariamente numéricamente iguales. Las técnicas de compresión de las imágenes, se emplean frecuentemente en la extensión de las mismas, que a la vez representan el estándar seguido para la compresión como JPEG (Joint Photographic Experts Group).
- Procesamiento morfológico. La morfología se relaciona con la forma y las relaciones entre las partes de una imagen. El procesamiento morfológico consiste en aquellas herramientas que permiten extraer componentes de la imagen que son útiles en la representación y descripción de la forma. Este utiliza los mecanismos de la teoría de conjuntos y considera cada uno de los elementos que componen una imagen digital como conjuntos. Por lo tanto, si se emplea una imagen binaria cada pixel con coordenada ( $x,y$ ) se define como un conjunto de dos dimensiones que representa el pixel blanco o negro de la misma, mientras que si se emplea una imagen con escala de grises, los dos primeros elementos del conjunto representan las coordenadas y el tercero corresponde al nivel de intensidad, e imágenes con distintos modelos de color se representan con conjuntos de mayores dimensiones.

Así, el procesamiento morfológico consiste en aplicar a las imágenes representadas por conjuntos elementales de estructurado (ESs), que son pequeños conjuntos o sub-imágenes, que pueden adoptar distintas formas (aunque se rellenan con elementos para que adopten formas rectangulares), utilizadas con el fin de probar una imagen bajo estudio para propiedades de interés. A continuación se pueden observar distintos ES, con un origen se encuentra marcado por un punto negro, aunque si éste no se encuentra marcado se considera que es el centro del elemento de estructurado:

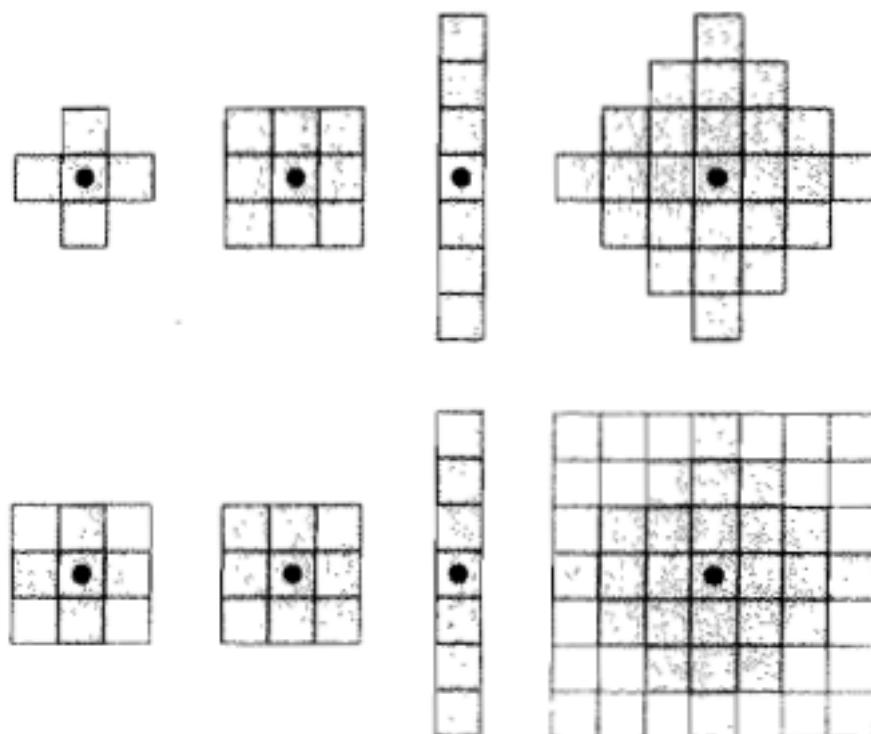


Figura 5.87: Ejemplos de ES de procesamiento morfológico

Por lo tanto el procesamiento morfológico, consiste en crear un nuevo conjunto que representará

la imagen de salida, aplicando el origen de los elementos de estructurado B a cada uno de los pixeles de la imagen A, conservando el pixel que esta en el origen, en aquellos casos en los que todos los elementos del conjunto B pertenezcan a la vez al conjunto A. Esto quiere decir, que al aplicar B sobre A, se conserva el elemento en el origen<sup>1</sup> lo si todos los pixeles de B se encuentran superpuestos con pixeles de A que representan un objeto.<sup>2</sup> En la siguiente figura se puede observar un ejemplo donde para la imagen A con un objeto sombreado, y un ES denominado B, B se aplica<sup>3</sup> a la imagen y se genera una nueva imagen con el objeto original reducido en tamaño:

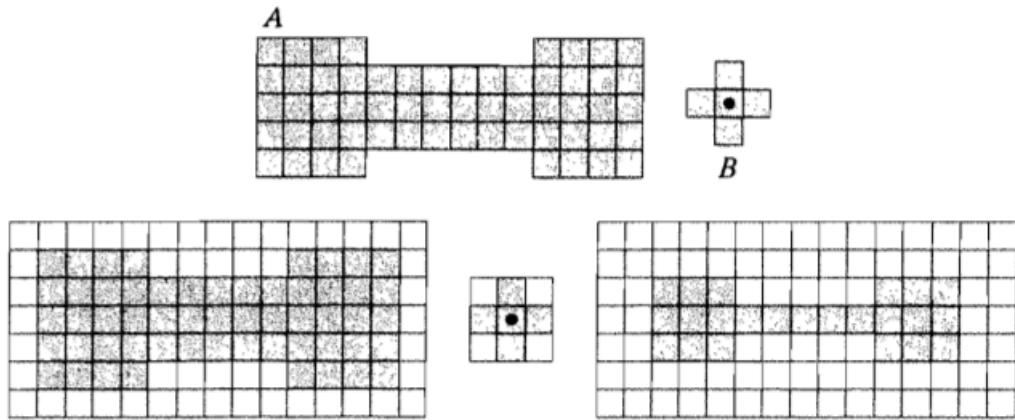


Figura 5.88: Ejemplo de procesamiento morfológico. En la fila superior se observa la imagen y el ES asociado<sup>1</sup>, la parte inferior-izquierda se visualiza la imagen original, en la parte inferior-central se encuentra el ES, ampliado para adoptar una forma rectangular (donde la parte sombreada representa los pixeles de interés que deben tenerse en cuenta con respecto a la pertenencia), y la parte inferior-derecha representa el objeto sombreado luego de haber aplicado el ES.

Las operaciones morfológicas que se aplican en una imagen, emplean los elementos de estructurado en conjunto con las propiedades de reflección y de traslación, y operaciones primitivas de erosión y dilatación. La propiedad de reflección consiste en reemplazar las coordenadas de la imagen por sus opuestas modificando el signo de cada uno de los elementos del conjunto. Mientras que la traslación consiste en desplazar los elementos del conjunto por un valor  $z$ , cuya dimensión dependerá de la cantidad de elementos del conjunto.

$$\hat{B} = \{w | w = -b, \text{ for } b \in B\}$$

Figura 5.89: Fórmula de reflexión para un conjunto B.



$$(B)_z = \{c | c = b + z, \text{ for } b \in B\}$$

Figura 5.90: Fórmula de traslación para un conjunto B para un valor  $z$ .

Por una lado la erosión de una imagen A por B dentro de un espacio  $\mathbb{Z}^2$ , es el conjunto de todos los puntos  $z$  tales que B, trasladado por  $z$  esta contenido en A, es decir,<sup>4</sup> elimina todos aquellos puntos que desplazados por  $z$  estan localizados dentro de A. Ésto<sup>5</sup> una imagen binaria hace mas finos los

objetos de la imagen, removiendo de la misma aquellos elementos que son mas pequeños que el elemento de estructurado.

$$A \ominus B = \{z | (B)_z \cap A^c = \emptyset\}$$

Figura 5.91: Fórmula de erosión 

Por otro lado, la dilatación de una imagen A por B dentro de un espacio  $Z^2$ , consiste en realizar la reflección de la imagen alrededor del origen de ésta y luego aplicar la traslación a la reflección por un valor z. De esta manera, la dilatación consiste en mantener aquellos z para los que la reflexión de B y los valores de A se superponen por al menos un elemento. Esta técnica a diferencia de la erosión, provoca que los objetos se vuelvan más gruesos (crezcan de grosor), dependiendo la medida en que estos se modificarán según la forma del elemento de estructurado aplicado a la imagen.

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

Figura 5.92: Fórmula de dilatación 

Por lo tanto, las operaciones principales asociadas este tipo de procesamiento, se definen como operaciones sobre conjuntos que combinan las propiedades de traslación, la erosión y dilatación, y abarcan tanto procedimientos relativos a la extracción de características de la misma, como aquellos que se emplean antes y después de los mismos. A continuación se detallan las operaciones relacionadas a la extracción de características:

- Extracción de límites(boundary extraction) y componentes conectados. La extracción de límites consiste en extraer el contorno de un objeto representado en una imagen binaria, mientras que la extracción del componente conectado abarca la obtención del contorno y los pixeles que conforman la forma del objeto. Las fórmulas para estas operaciones se definen de la siguiente manera:

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

Figura 5.93: Fórmula para la extracción de borde.

- Esqueletos(Skeletons). Un esqueleto es junto de elementos de una imagen que representan la forma de la misma y se encuentran equidistantes a los límites.Esqueletos en una imagen representan los objetos con el menor número de pixeles, siendo cada uno de éstos parte de la estructura básica y siendo absolutamente necesarios. Por lo tanto, los esqueletos permiten obtener información respecto de la imagen original tales como la posición, la orientación y la longitud de los segmentos que lo conforman. El proceso para la obtención de esqueletos, consiste en aplicar una sucesión de operaciones de erosión a una imagen con un elemento ES, hasta el punto donde si se aplicara la siguiente erosión se eliminarían los objetos que pertenecen a ésta. Luego, se puede aplicar la diferencia entre las múltiples erosiones menos las múltiples erosiones aplicadas a la operación de opening(descrita más adelante), de manera que se consideren aquellos elementos no suavizados como parte del esqueleto. Finalmente, la unión de todos los esqueletos de las figuras en la imagen constituyen la imagen final.

$$X_k = (X_{k-1} \oplus B) \cap A \quad k = 1, 2, 3, \dots$$

Figura 5.94: Fórmula para la extracción de componentes conectados. Los valores  $X_k$  representan los elementos de una nueva imagen (con las mismas dimensiones que la imagen de entrada A) resultante de aplicar B sobre A.



Figura 5.95: Ejemplo de la imagen binaria original

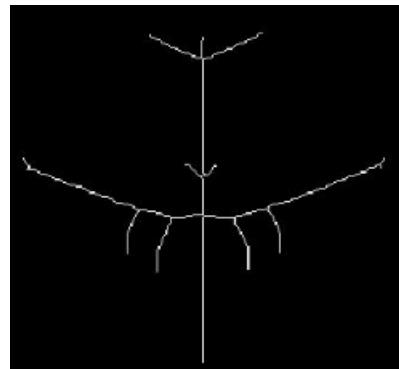


Figura 5.96: Ejemplo de la imagen luego de la obtención del esqueleto

- Convex Hull. Un conjunto de elementos A se dice que es convexo si, contiene al menos dos puntos de un objeto de imagen que pueden ser vinculados por una línea recta dentro de él. Un convex hull H de un conjunto A por lo tanto, es el conjunto convexo más pequeño que contiene a A.

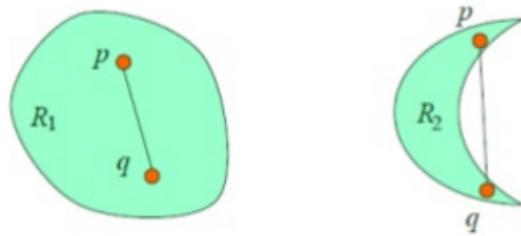


Figura 5.97: Ejemplos de convex hull convexo y no convexo. A la izquierda es convexo debido a que se puede vincular p y q, mientras que en la figura de la derecha no es convexo, sino concavo.

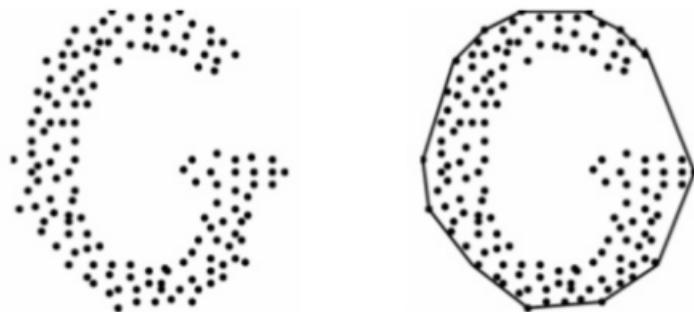


Figura 5.98: Ejemplo de convex hull aplicado a una imagen.

El funcionamiento para la extracción del convex hull, consiste en emplear varios elementos de estructurado e iterativamente, aplicar una operación de detección de forma(shape detection) entre cada uno de los ES de la imagen de entrada. Esta operación consiste en emplear la erosión entre la imagen de entrada y un ES y su complemento y el ES, con el fin de obtener resultados intermedios. De esta forma, el resultado final se obtiene uniendo cada uno de los conjuntos producto de la detección de forma.

Las técnicas que se emplean en conjunto con la extracción de características son las siguientes:

- Opening y Closing. La técnica de opening se emplea con el objetivo de suavizar el contorno de los objetos, eliminando conexiones entre elementos que contengan un número bajo de pixeles, y eliminando protuberancias finas. Por el contrario, closing tiende a suavizar los contornos de las figuras, pero fusiona líneas que no se encuentran completamente unidas en la figura y golpes largos y finos de pixeles, y rellena espacios en el contorno y elimina pequeños hoyos en la imagen.

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

Figura 5.99: Fórmula de opening

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

Figura 5.100: Fórmula de closing

- Thinning. Esta técnica consiste en eliminar elementos de una imagen A por medio de la aplicación de la operación de shape detection, aplicando una secuencia de n elementos de estructurado que se encuentran en diferentes ángulos. De esta forma el resultado final es una imagen donde los elementos que pertenecen a ésta han reducido su grosor y se ha ampliado la cantidad de píxeles que representan el fondo de ésta.
- Region filling. Esta operación consiste en efectuar el relleno con información de los elementos que componen una región.
- Pruning. Este método es empleado para la eliminación de elementos de imagen excedentes producto del empleo de skeletons y thinning.

 Segmentación.  Segmentación de una imagen es el proceso de subdividir los píxeles en una imagen en regiones uniformes y homogéneas, donde cada región es un grupo de píxeles, que representa un objeto o una parte de la escena que se muestra en la imagen. Así, la segmentación permite obtener agrupaciones de píxeles que comparten características similares, interconectadas y no solapadas, donde cada pixel de una región  segmento en la imagen adquiere una etiqueta de región que indica la región a la que pertenece.  Este proceso es uno de los más importantes elementos en análisis de imágenes automatizado, principalmente porque posibilita extraer aquellas entidades de interés en la imagen para aplicar otros métodos de procesamiento, como la descripción y el reconocimiento.

- Procesamiento de color de imagen. Debido a que el color se considera como un descriptor potente para el reconocimiento y extracción de objetos de una escena, se han desarrollado métodos que permiten emplearlo en el procesamiento de imágenes. En general, este tipo de procesamiento consiste en trabajar con los componentes de color (o espacios de color) RGB por separado y asociar las componentes de cada uno de los píxeles de entrada para generar el pixel a color en la imagen de salida. Las tareas de procesamiento de imágenes digitales con color son variadas, aunque se distinguen las siguientes:

- Asignación de colores basado en tonalidades de grises. Este tipo de operación se realiza con el fin de lograr un mejor entendimiento de la imagen por parte del observador. Existen diferentes aproximaciones para realizar esta tarea, destacándose el rebanado por intensidad y la transformación intensidad-color. El rebanado por intensidad consiste en dividir los  $L-1$  valores de la escala de grises en  $P$  planos, de manera que los píxeles cuyas intensidades se encuentren entre dos distintos planos sean representados por diferentes colores y, aquellos elementos que caigan en la división de un plano se le asigne el color del plano.

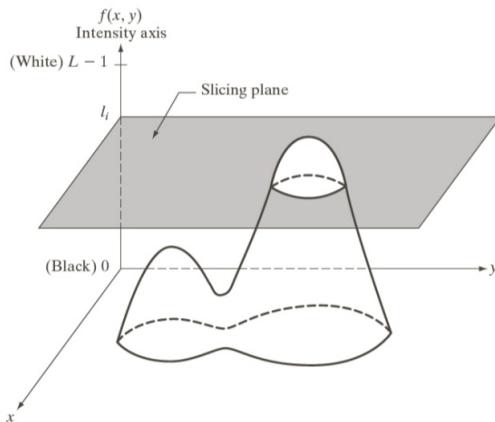


Figura 5.101: Rebanado por intensidad

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

La asignación de colores empleando transformaciones de intensidad a color se basa en aplicar tres transformaciones independientes a las intensidades para producir cada uno de los campos de color de los pixel a color de la imagen de salida, de manera que los colores de la imagen resultante, se asignan según las características del tipo de transformación que se aplique. Las transformaciones aplicables con este método pueden ser transformaciones matemáticas, o basadas en filtros de suavizado y funciones no lineales, lo que brinda flexibilidad. A continuación, se puede apreciar un ejemplo donde se aplica una transformación que produce una onda sinusoidal para cada elemento, asignando colores más intensos para los elementos cuyo valor de onda sinusoidal se encuentre con mayor inclinación:

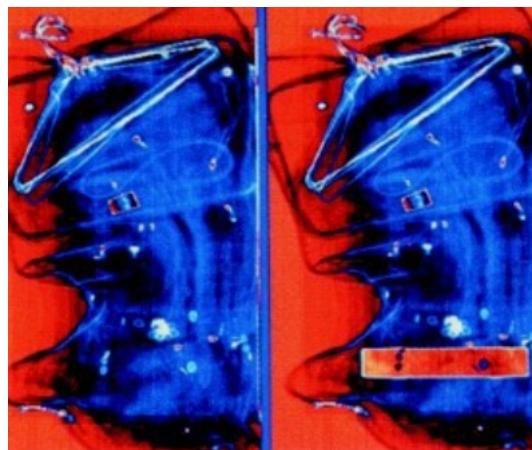


Figura 5.102: Ejemplo de transformación intensidad a color para una imagen. La imagen de la izquierda pertenece a una maleta sin explosivos, mientras que la de la derecha pertenece a una maleta con un explosivo simulado.

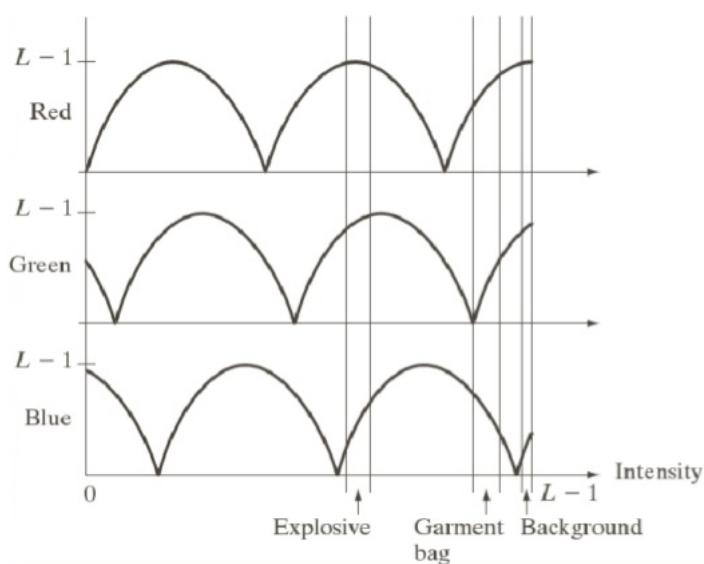


Figura 5.103: Onda sinusoidal para cada componente que representa los valores de los diferentes objetos.

Alternativamente, este procedimiento también se puede emplear aplicando transformaciones sobre varias imágenes monocromáticas de distintas bandas del espectro electromagnético (capturadas por distintos sensores) y posteriormente combinarlas en una sola,

de manera que se puedan visualizar características en imágenes más complejas, que sería imposible visualizar en caso contrario. Ésta variación de la técnica se emplea en el procesamiento de imágenes multiespectrales captadas por satélites. A continuación, se puede visualizar un ejemplo donde las imágenes superior-izquierda, superior-derecha y media-izquierda se tratan como los componentes RGB para formar la imagen media-izquierda. Y la imagen inferior-derecha, se produce como resultado de combinar el campo rojo de la imagen infrarroja inferior-izquierda, sobre la imagen RGB obtenida anteriormente:

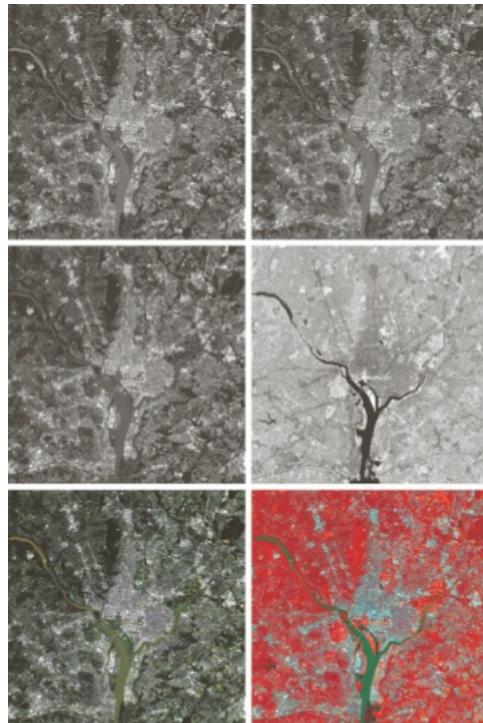


Figura 5.104: Ejemplo de imagen multiespectral

- Transformaciones de intensidad. Las transformaciones de intensidad como el suavizado de imágenes con filtros, se pueden aplicar de igual manera a los componentes de color de una imagen considerando el vector de intensidades como la entrada.
- Procesamiento de Histograma. Esta tarea emplea el mecanismo de histograma de imágenes y sus herramientas asociadas, aunque para procesar imágenes a color considera cada uno de los componentes de color rojo, verde y azul como un histograma separado, y emplea los niveles de intensidad de éstos para generar la imagen de salida.
- Complementos de color. El complemento de color es equivalente al negativo en las imágenes con escala de grises, y para obtenerlo se debe modificar la tonalidad de la imagen cambiando cada uno de los componentes por su opuesto, de manera que el opuesto para un color de entrada se define en función de este por la rueda de colores que se detalló en la siguiente figura:

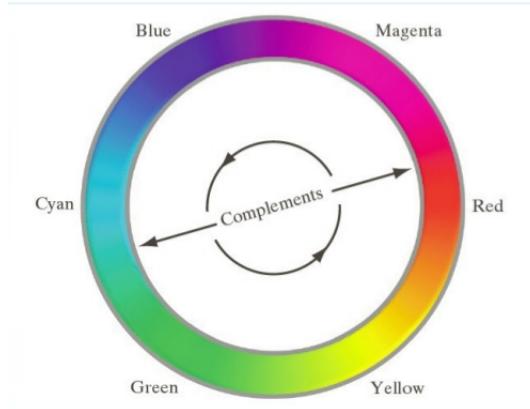


Figura 5.105: Ilustración de los complementos de cada color

- Color Slicing (Rebanado por color). Esta transformación se emplea con el objetivo de separar objetos en la imagen del fondo, para ello se define un vector que representa el color neutral y posteriormente, se verifica para cada pixel si éste se encuentra dentro de un área determinada (centrada en una coordenada de la imagen) aplicando una fórmula matemática (que define la forma de ésta, si es un radio o un rectángulo por ejemplo) y si es así, se conserva el pixel con el color que posee. En caso contrario, se le asigna el valor del color neutral. En la siguiente figura se puede observar el efecto de esta operación sobre una imagen que contiene un bowl con frutillas:

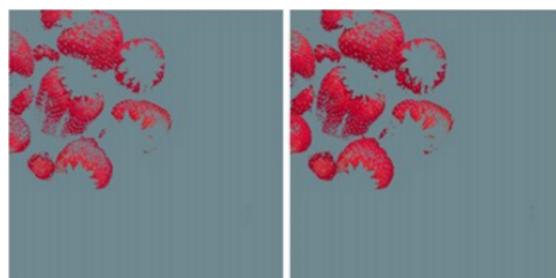


Figura 5.106: Color slicing ejemplos. En el ejemplo de la izquierda se empleo un cubo centrado en el punto  $(0,6863, 0, \dots)$  y en la de la derecha se empleo una esfera centrada en la misma coordenada.

- Suavizado de colores(Color Smoothing). El suavizado de imágenes al igual que en imágenes representadas por escala de grises, consiste en aplicar un filtro a la imagen digital de entrada con el fin producir los píxeles de salida, aunque en este las operaciones se aplican sobre los vectores que representan los componentes de color. De esta forma, la intensidad del pixel de salida se compone de procesar cada uno de los componentes de color por separado.
- Segmentación por colores. Dependiendo de si el modelo de representación de la imagen es HSI o RGB se puede realizar esta operación de dos maneras diferentes: Si la imagen es HSI, la segmentación consiste en crear una máscara binaria generada a partir de aplicar el thresholding para un valor dado al componente de tonalidad o Hue (ya que los colores se representan con mayor detalle en este componente) provocando que los valores que no están sobre ese límite se descarten (seteándose a 0) y conservando el resto (seteado en 1), y luego se aplique esta máscara al componente de tonalidad. La salida de este procedimiento es una imagen donde las regiones con mayor predominancia de color han sido filtradas.

Si la imagen emplea RGB, la segmentación de objetos en rangos de color específicos simplemente consiste en tomar un conjunto de puntos que el color es representativo de los colores de interés, y obtener el promedio de color para cada componente y almacenar esto como un vector. Luego por medio de una fórmula de distancia, se realiza una

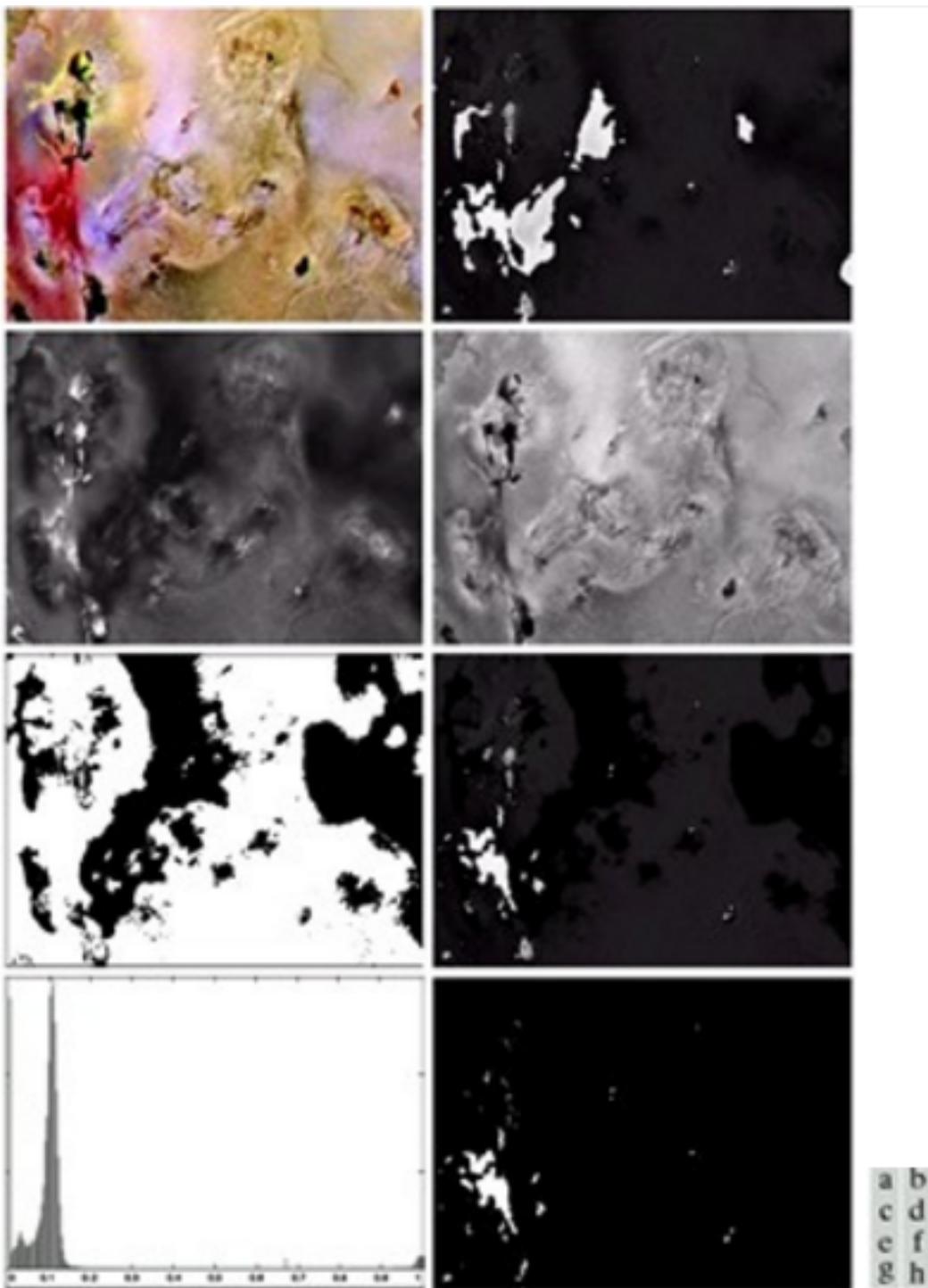


Figura 5.107: Ejemplo de segmentación a color para imagen con HSI. En este ejemplo, se realiza la segmentación por color de la región rojiza en la imagen original (a), siendo (b) el matiz o tonalidad (Hue), (c) la saturación y (d) intensidad. En (e) se define una máscara binaria de saturación que se aplica a (b) y el resultado se muestra en (f). En (g) se muestra el histograma resultante de este procedimiento y en (h) se muestran los pixeles que pertenecen a la región de la imagen que tiene tonalidad roja resaltados en blanco.

comparación de los componentes de color de cada pixel con el vector y si se encuentra dentro de ésta distancia se lo conserva, y en caso contrario se lo descarta. La fórmula de distancia más común para realizar la medida de distancia es la Euclidiana, donde si la distancia entre el vector del pixel  $z$  y el vector promedio  $a$  se encuentra a menos de un límite  $D\theta$ , se considera un color válido.

$$\begin{aligned} D(\mathbf{z}, \mathbf{a}) &= \|\mathbf{z} - \mathbf{a}\| \\ &= [(\mathbf{z} - \mathbf{a})^T(\mathbf{z} - \mathbf{a})]^{\frac{1}{2}} \\ &= [(z_R - a_R)^2 + (z_G - a_G)^2 + (z_B - a_B)^2]^{\frac{1}{2}} \end{aligned}$$

Figura 5.108: Fórmula de distancia Euclidiana 

- Representación y descripción. Este proceso mayormente se emplea a continuación del proceso de segmentación, ya que ésta produce datos relacionados con los pixeles contenidos en el límite o en la región y es preferible emplear esquemas que compacten la información segmentada para mejorar el procesamiento de descriptores. as técnicas, consisten en transformar los pixeles que forman una región en una representación conveniente para su procesamiento. La representación de una región proporciona dos alternativas: Representar la región en términos de sus características externas (su límite o boundary por ejemplo), describiéndose el límite por sus características como su longitud, la orientación de la línea recta que conecta sus puntos extremos, y el número de concavidades en el límite; presentarla según sus características internas, es decir, los pixeles que comprenden la región. En este caso, se emplean características propias de la región tales como el color y la textura. La característica principal de la textura, es que consiste en la repetición de un patrón o conjunto de patrones sobre una región. Éste puede ser repetido exactamente o con pequeñas variaciones de posición y, ciertas características como: forma, tamaño, color y orientación de los elementos que forman el patrón pueden variar sobre la región. mas veces, la diferencia entre texturas se obtiene por medio del grado de variación de los mismos o, en la distribución estadística de los elementos del patrón.

Con respecto a la descripción de una textura, existen 3 aproximaciones para esta tarea:

- La aproximación estadística. La aproximación estadística consiste en emplear medidas cuantitativas estadísticas (media aritmética, varianza, desvío estándar y otras) respecto a la distribución de los niveles de intensidad a una imagen, asociando distintas combinaciones de estos valores a características  como la suavidad, la aspereza, la granularidad, etc.; descriptores que, en base operación que define relaciones entre pixeles y las frecuencias de esas relaciones entre pixeles en la imagen, organizados en forma de matriz de  $N \times N$  siendo  $N$  los niveles de intensidad (matriz de co-ocurrencias), permiten encontrar y describir patrones tales como la uniformidad, homogeneidad, contraste y correlación.

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

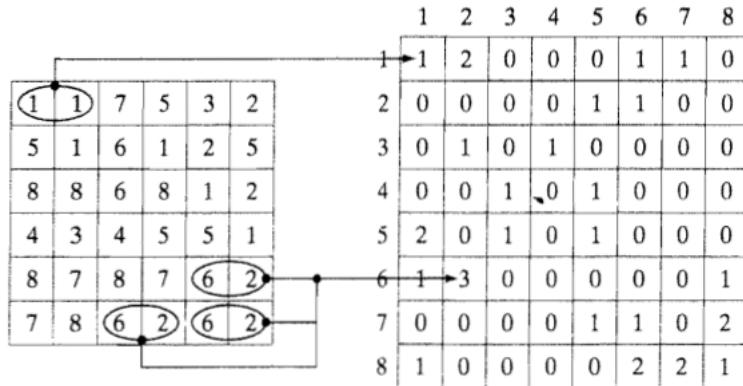


Figura 5.109: Ejemplo de matriz de co-ocurrencias(derecha) de una imagen (izquierda), 8 niveles de intensidad, definiendo la relación entre píxeles como aquel pixel que se encuentra inmediatamente a la derecha.

- La aproximación estructural. Esta técnica consiste en describir una textura por medio de un conjunto de reglas(o primitivas) que representan el patrón de la misma, que se repite de manera constante.



- La aproximación espectral. Esta aproximación se basa en las propiedades de la transformada de Fourier, y son usadas primariamente para detectar periodicidad en la totalidad de la imagen a través de picos en el espectro de las frecuencias de la imagen.

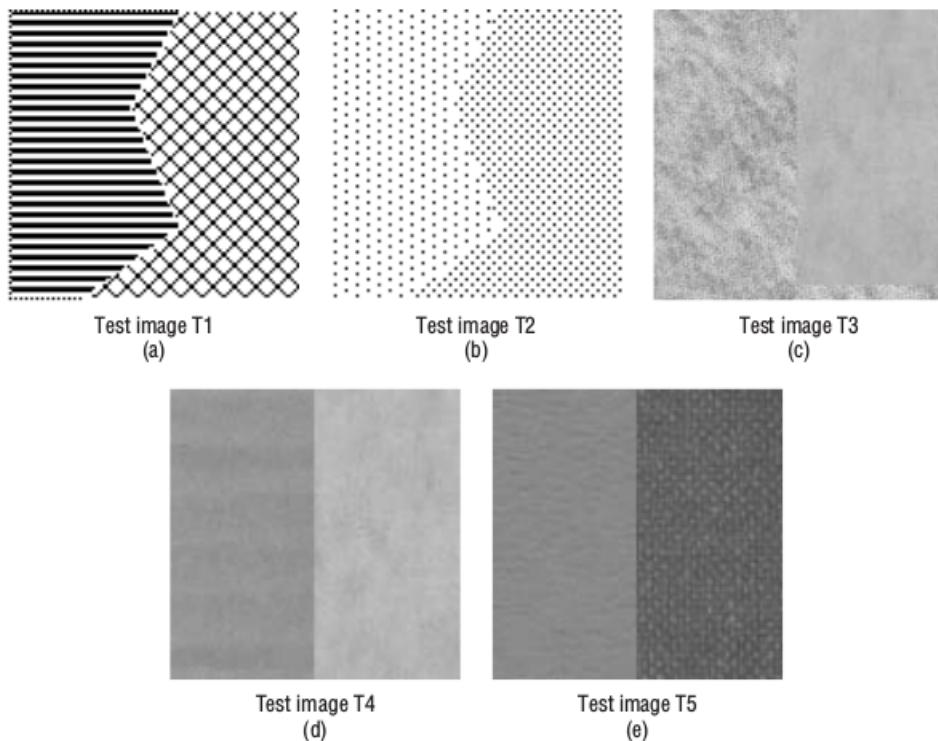


Figura 5.110: Ejemplos de texturas artificiales (a-b) y texturas naturales(c-e)

#### 4.1.2. Estudios relativos a la detección de fallas

En lo que respecta al procesamiento de imágenes, una de las aproximaciones que se han empleado dentro de esta área es la de Koch y Brikalis [13], que se centra en el reconocimiento de hoyos en el pavimento por medio de una variedad de imágenes provenientes de distintas fuentes (tomadas manualmente, Google o capturas de videos con calles), tomadas desde una distancia menor al metro del suelo y en distintos ángulos. En este estudio se emplea un modelo que utiliza las sombras en las imágenes para detectar la ubicación del hoyo dentro del pavimento, y posteriormente aislarlo de éste. Este modelo consiste de 3 fases: Segmentación de la imagen, extracción de forma y comparación de texturas.

Durante la segmentación de la imagen, se realiza la conversión de la imagen desde el modelo de color RGB a una escala de grises, se obtiene el histograma de imagen y se aplica la técnica de limitado de imagen para un valor, y se la convierte a una imagen binaria. Luego, en la etapa de extracción de forma se procede a aplicar thinning hasta obtener el esqueleto de imagen y a conectar las ramificaciones empleando un algoritmo de regresión elíptica, debido a que el ángulo de captura de las imágenes no es perpendicular al suelo. En este estudio, se definieron parámetros para determinar el grado de curvatura de los elementos en la imagen (eccentricity o  $e$ , donde cuando más próximo a 0 es  $e$ , mayor es su curvatura), la posición del centroide (centro de gravedad de la imagen, que representa la posición promedio de todos los puntos en un plano) y el tamaño de la región. De esta manera, por medio de estos parámetros obtenidos experimentalmente se clasificaron las regiones en cada imagen como candidatas a un hoyo o, como una región externa al mismo.

Finalmente, en la fase de comparación de texturas se caracterizan las mismas por medio de la aproximación estadística, y se emplea un vector de características que se genera con una serie de filtros para cada región interna y externa y, finalmente estas se comparan empleando un mecanismo de machine learning de MATLAB, en combinación con el set de herramientas de procesamiento de imágenes integrados.

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

De esta forma, el entrenamiento se realizó con 50 muestras de imágenes seleccionadas aleatoriamente, y la prueba fue llevada a cabo con 70 muestras, que contenían no solo hoyos, sino también elementos típicos que se pueden encontrar en caminos pavimentados, como son grietas, irregularidades y juntas. El resultado de este estudio fue favorable, ya que se lograron valores de precisión cercanos al 80 %, sin embargo, este modelo tiene la desventaja de depender fuertemente de condiciones de luz solar óptimas, la naturaleza de su funcionamiento. Además, si el ángulo del sol con respecto al suelo es perpendicular, la cantidad de sombra generada en la imagen es mínima por lo que algunos baches pueden pasar inadvertidos.

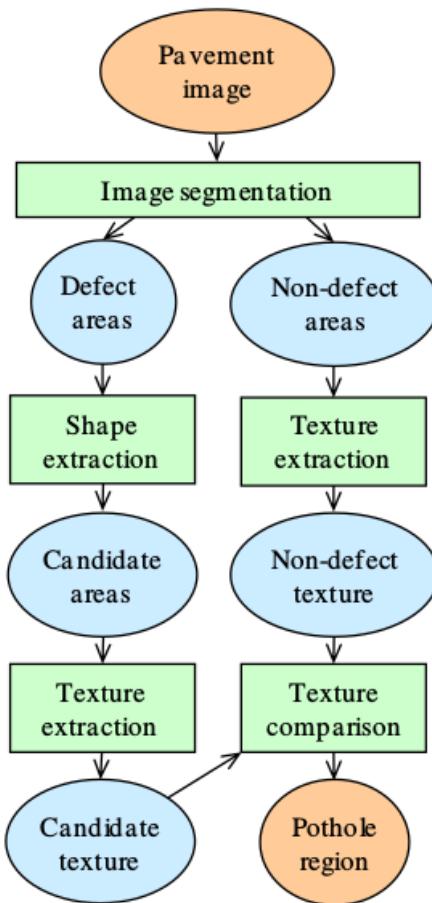


Figura 5.111: Modelo de procesamiento propuesto por Brikalis y Koch. La extaída desde [12].

Posteriormente, la aproximación de Koch y Brikalis definida en [12] se extendió para incluir el procesamiento de frames de video, debido a que en el estudio anterior el procesamiento completo se debe aplicar a cada imagen individual, lo que resulta computacionalmente ineficiente, ya que el bache tiene que ser detectado sucesivamente dentro de cada cuadro de video. Con este método, empleando las características de la textura en un bache, la forma elíptica y la sombra que se forma alrededor de ésta, el procesamiento consiste en subdividir la región de obtención de texturas a un área central con una cierta cantidad de pixeles, y aplicar filtros a cada una de las regiones, con el fin de obtener vectores de características para cada región. Este procedimiento se emplea en el primer frame sobre un área de pavimento no dañada, obteniendo así un valor de vector de características para un sector intacto. Después, a medida que se obtienen los subsecuentes frames se compara el promedio de los vectores de características anterior con el nuevo promedio, si no existe una diferencia significativa se actualiza y en caso contrario se produce la detección de un hoyo. Adicionalmente, este método, una vez que detecta un bache procede a detener el algoritmo de comparación de texturas, e inicia un algoritmo de tracking de objetos basado en valores de texturas para efectuar el seguimiento de éste, hasta que ya no se visualiza en la imagen. En este punto, se procede a reanudar el algoritmo de detección de fallas hasta detectar un nuevo bache en el pavimento. Este experimento, fue testeado capturando el video desde un robot equipado con una cámara, con un total de 39 videos (10180 frames), logrando una precisión del 75 % y un recall 84 % (ver

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

machine learning). Sin embargo, esta aproximación tiene el inconveniente de que únicamente se considera que solamente un único hoyo entrará en el campo de visión a la vez.

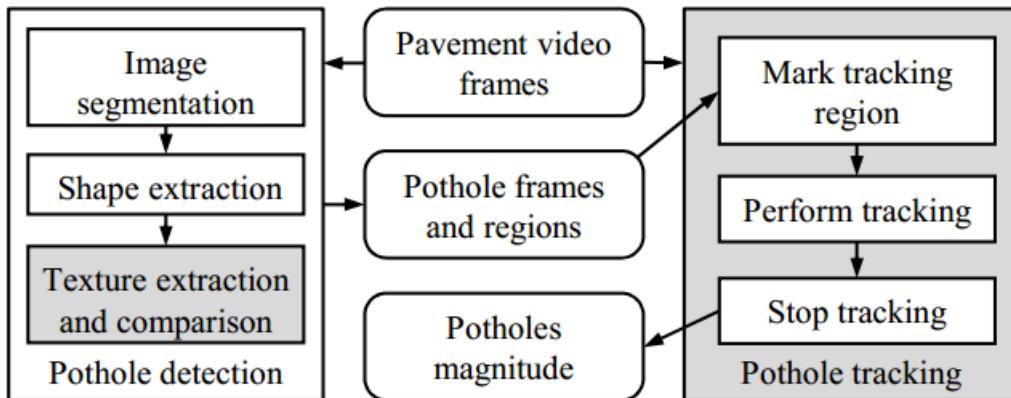


Figura 5.112: Método anterior de Brikalis y Korch, con mejoras subrayadas. Fuente extraída desde [12].

Otra aproximación que se ha empleado para la detección de fallas, es la expuesta por Buza, Omanovic y Huseinovic en [21], que consiste en realizar la obtención de imágenes o frames de video a partir de grabaciones realizadas con cámaras digitales montadas exteriormente a vehículos. El primer paso en esta aproximación consiste en transformar la imagen a color (en RGB) a escala de grises, para luego continuar con la segmentación de la imagen a través del thresholding (o limitado) de los valores en el histograma de imagen. Así, el límite de la imagen se calcula por medio del método de Otsu para el clustering en imágenes, que consiste en aplicar una fórmula matemática (ecuación (2) de [21]) que realiza la división entre los valores de intensidad del histograma de manera que, se realice la separación entre los pixeles del fondo y de los objetos, obteniendo una variación mínima de intensidad dentro de los elementos que componen cada una de estas clases y una variación considerable entre elementos de distintas clases. Luego, se extraen aquellas formas de la imagen que son lineales, por medio del valor de curvatura (eccentricity) y aquellas formas en la imagen que se encuentran conectadas al límite de ésta. Como salida, se obtiene una imagen intermedia con formas lineales, figuras conectadas al borde la imagen y que no satisfacen el límite de Otsu removidas.

Posteriormente, se procede a realizar la extracción de forma aplicando un algoritmo de clustering, donde se realiza la agrupación de los elementos en  $k$  clusters (definidos por el usuario) según su similitud en conjuntos (o clusters) de pixeles. El método seleccionado fue el Spectral Clustering, que se basa en aceptar como entrada una matriz de similitud, donde cada uno de los elementos  $(i,j)$  de ésta representa el grado de similitud de los puntos  $i$  y  $j$ , empleando para medir éste una fórmula de distancia Euclídea normalizada por un factor de escala (ecuación (4) de [21]). El factor de escala se calcula en base a los valores del histograma de imagen (ecuación 5 de [21]).

El próximo paso consiste en seleccionar de la imagen intermedia un conjunto de “semillas” o puntos, tales que cada uno de esos puntos se emplea como una representación de otros 50 puntos para el próximo paso; se realiza con el fin de reducir el tiempo de cómputo. Estas semillas se emplean para la extracción de la región del bache vertical, donde se considera cada una de éstas eligiendo aquellas cuyas coordenadas tengan el valor mayor y menor en Y y por medio de la inspección de la imagen clusterizada, se procede a seleccionar los pixeles cuyos valores de intensidad sean más próximos al valor de la semilla, obteniendo así la región vertical del bache. El procedimiento anterior, se realiza de manera análoga para definir una región horizontal del hoyo, a excepción de que se seleccionan las semillas que tienen una coordenada con valor X mayor y menor (dentro del conjunto de las semillas), seleccionando aquellos pixeles ubicados en los extremos derecho y e izquierdo. Como paso final, se procede a unir los pixeles que componen la región del bache con líneas a fin de identificar su área y posición en la imagen. Este método fue testeado en MATLAB con la suite de imágenes integrada, con 50 imágenes seleccionadas de Google, donde 25 imágenes son hoyos, 10 grietas y el resto imágenes con reparaciones y áreas sin defectos, lográndose una precisión del 80%.

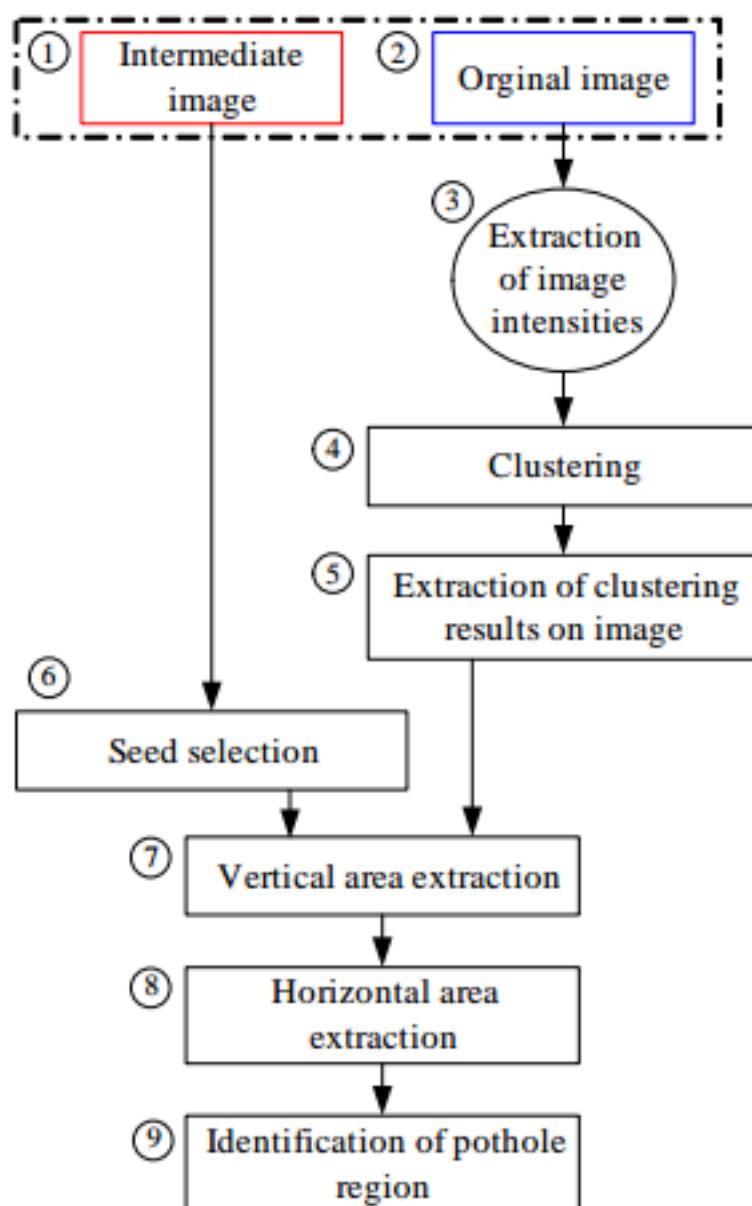


Figura 5.113: Proceso de extracción de potholes. Extraído desde [21].

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

En otros estudios como [22] se especifica un sistema inteligente de transporte(ITS) para la advertencia de fallas, donde se integran métodos deterministas para el reconocimiento y ponderación de la gravedad de los baches, basados en su localización y características. Este sistema consiste en emplear un vehículo con dispositivo de sensado estacionario, que agrupa elementos tales como: una cámara de video digital CCD, GPS para la obtención de la ubicación, espacio de almacenamiento para las capturas, y comunicación Wi-Fi. Este dispositivo permite obtener muestras del pavimento que son enviadas a un algoritmo de detección de baches para análisis, y en caso de detectar una falla, se estima su criticidad por medio de un valor establecido y ésta, junto con su ubicación, puede ser enviada por medio de dispositivos ubicados al costado del camino, a un sistema para la administración de fallas por medio de Comunicación de corto Rango Dedicada(DSRC), WAVE o Wi-Fi. Así, cuando otro vehículo que emplee el mismo sistema circule cerca de la misma zona, el conductor puede ser advertido acerca de una falla.

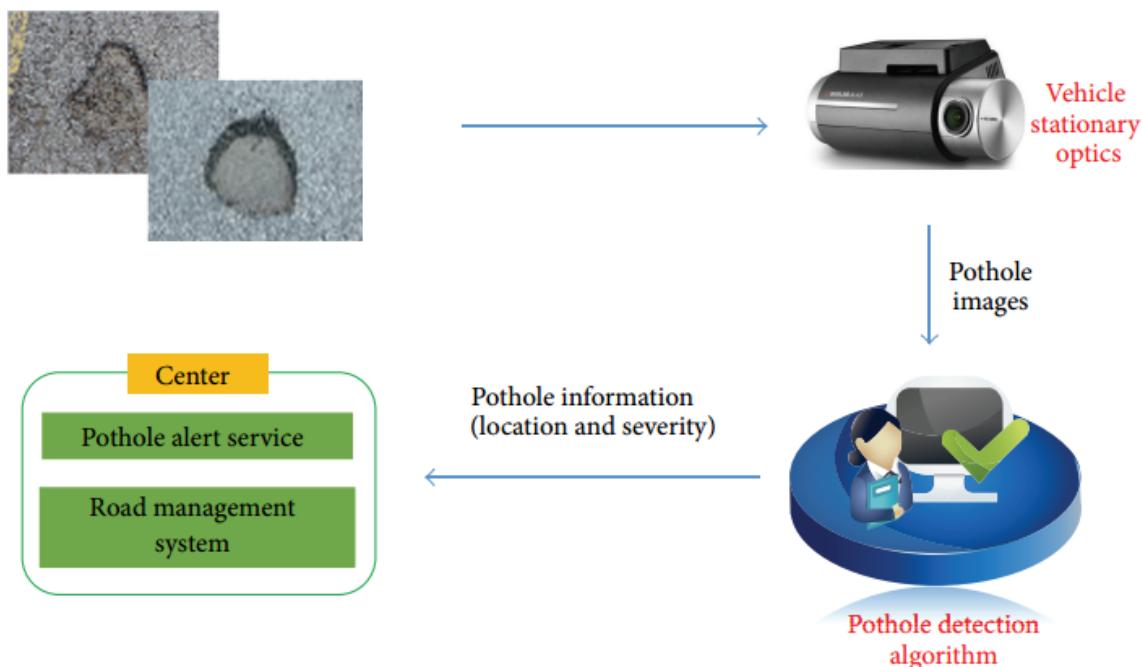


Figura 5.114: Sistema propuesto para la detección de fallas. Extraído desde [22].

Con respecto al método de detección de hoyos en imágenes, el primer paso consiste en entregar la imagen aplicando para ésto el limitado de histograma basado en forma(HST, histogram Shape-based thresholding) lo que produce una imagen binaria con las regiones segmentadas, y luego aplicar un filtro 9x9 a la imagen para reducir el ruido y la operación morfológica de closing, lo que permite eliminar el ruido producido por la segmentación y cerrar las regiones candidatas. A continuación, se computan las características para la selección de regiones candidatas, tales como: la linearidad, el tamaño, el nivel de compactación (definida en base al área y el perímetro de la región), la elipseid, los centroides y el convex hull y se definen límites para éstos de manera de filtrar solo aquellas regiones que cumplen con éstos requisitos. Posteriormente, se aplica la técnica de intersección de histograma ordenado (OHI), cuya ecuación se encuentra definida en (4) de [22], donde se computa un valor producto de la comparación de los mínimos valores de histograma entre el histograma de una región candidata y el fondo de la imagen. Finalmente, si el desvío estándar entre ambos histogramas (region y fondo) es inferior a un valor límite, o si el valor de OHI es cercano a 1 y aplicando la operación de Sobel (operador para la detección de bordes que computa la pendiente de la función de intensidad de la imagen) es cercano a 1, se considera que la región candidata y el fondo son parte de la misma superficie, por lo que la región no es hoyo. Las pruebas para este método se realizaron con la cámara del dispositivo de captura en Korea, implementando el algoritmo de selección de regiones en C++, capturando un total de 90 imágenes en 100 experimentos, logrando valor de accuracy de 73 %, con una precisión del 80 % y 73.3 % de recall.

Otras investigaciones plantean un método mas directo, sin emplear machine learning o clasificación para el procesamiento de baches, como en [24] donde se creó una librería de imágenes empleando

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

un vehículo con una cámara GoPro montada en el parabrisas y extrayendo frames de estas filmaciones. En esta metodología no se consideran otros elementos del camino como vehículos, arbustos o árboles al costado del camino, por lo que con cada imagen se calcula el desvío estandar de los canales de color de la imagen, se filtran aquellos elementos que se encuentran en el rango de valores del color del asfalto (justo en frente del vehículo) y, se obtiene el convex hull del camino, para obtener una forma uniforme. Una vez aislado el camino, se convierte la imagen de color a escala de grises, se remueve el ruido aplicando una técnica de filtrado Gaussiano, y finalmente, se aplica el algoritmo de Canny (empleado para la detección de bordes en imágenes) que produce una imagen binaria, y se refina la imagen con el método de dilatación para eliminar contornos sobrantes producto de éste algoritmo. Este experimento se implementó en Visual C++ con OpenCV, empleando 53 imágenes en total con 97 fallas de la librería bajo distintos escenarios, a una velocidad de 40 km/h, logrando una precisión de 81.8 % y un recall 74.4 %. No obstante, este algoritmo tiene el inconveniente de que los baches que se encuentran en el borde del campo de visión del vehículo son mayormente ignorados por el algoritmo de Canny, y que ciertos tipos de bordes en el campo de visión del vehículo no son captados debido a la rigidez del algoritmo de detección de bordes.

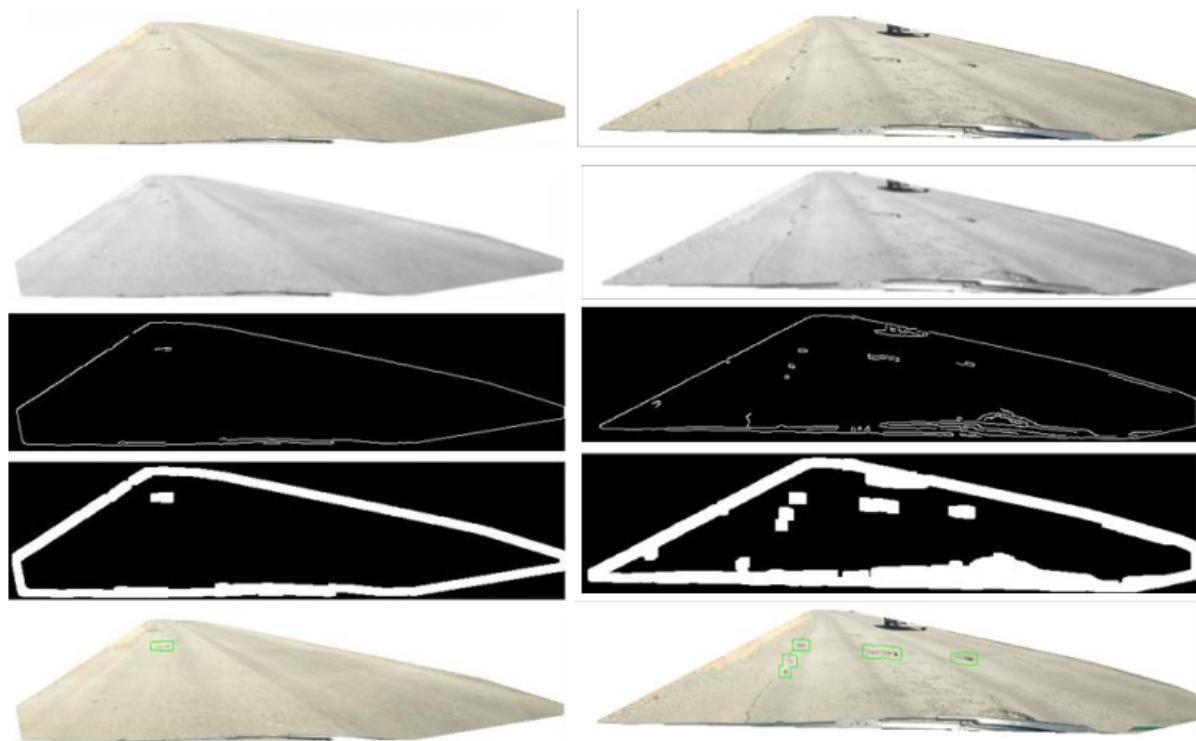


Figura 5.115: Ejemplo de los pasos realizados en el método propuesto. Extraído de [24].

Además de baches, también se han efectuado estudios donde se intentan aislar del pavimento como [48]. La metodología elegida consiste en realizar un pre-procesamiento de la imagen, escalando ésta a una resolución pre establecida y aplicando una normalización de los valores de intensidad de la imagen y realizando una saturación de pixeles (donde se ajustan los valores de pixeles para que no superen un determinado límite). Luego, se subdivide la imagen en bloques de imagen no solapados y se generan las características de cada bloque, considerando solamente el desvío estandar y la media aritmética, y se aplica el algoritmo de clustering K-Means para estas características. Así, la matriz de desvío estandar de cada uno de los bloques de la imagen de entrada, se resta contra una matriz de desvíos estándar perteneciente a una imagen con pavimento sano, y si la diferencia entre ambos es significativa, el bloque se guarda en una imagen binaria nueva con cada uno de sus valores en 1 (blanco), en caso contrario se le asigna 0 (negro). Utilizado el procedimiento anterior, se procede a asignar un nivel de severidad a la grieta calculando thresholding con el método de Otsu a la imagen de entrada, seleccionando como límite el menor valor del límite de Otsu para el asfalto sano y la imagen con la grieta, obteniendo como salida el componente conectado. Luego la forma de la grieta se approxima obteniendo el esqueleto, y eliminando los valores sobrantes con region opening. Por último, se calcula el grosor de la grieta como el número total de pixeles en una grieta dividido por el número de pixeles en el esqueleto de la grieta.

Alternativamente a este estudio, otra aproximación definida en [20] propone emplear los niveles de intensidad de una imagen en escala de grises(256 niveles) para detectar la diferencia entre la grieta y el fondo, teniendo en consideración que la forma de la grieta es fina y que, existe una diferencia considerable de intensidades entre ésta y el resto de la imagen. Esta técnica consiste en aplicar una máscara de filtrado (o ventana) que cuenta con un tamaño inicial y un tamaño final, variable a medida que se aplica ésta sobre la imagen. De esta forma, al realizar el proceso de filtrado se aceptan aquellos píxeles cuyas intensidades sean menores que un valor límite  $T$  establecido, actualizándose el límite para valores que lo superan, y al alcanzar una cantidad máxima de píxeles para la región candidata filtrada, el valor de la máscara se incrementa. Este procedimiento se repite hasta alcanzar el tamaño máximo de la máscara.

Adicionalmente, a medida que la máscara se aplica sobre la imagen, se calcula un valor de circularidad  $F_c$  a través de una fórmula matemática (definido en (2) en [20]) en base a la cantidad de píxeles filtrados en la zona actual de la imagen (que cuanto más próximo a 1 este indicará que la grieta tiende en mayor medida a ser una elipse). Este valor, en combinación con un parámetro de aceleración (definido en ecuación (3) y (1') en [20]), cuyo valor inicial es un parámetro de entrada, se emplea para modificar la rapidez con que se modifica el valor  $T$  de intensidad, de manera que si la grieta es fina y delgada, el valor  $T$  se modifica más lentamente captando con más detalle la grieta, mientras que si por el contrario tiende a otro tipo de forma la velocidad de actualización es constante. También se agrega otra modificación en este estudio para en lugar de emplear un tamaño mínimo y máximo de ventana, se emplee longitud de la grieta mínima y máxima (especificados por el usuario) y se la compara con un valor de circularidad  $C_{max}$  para aumentarlo, solamente si se supera ese valor. Este estudio fue probado con 114 imágenes capturadas con camaras digitales, brindando un recall de 0.753 y precision de 0.788, lo que lo convierte en un método apropiado para la clasificación de grietas.

## 4.2. Proyectos basados en sensores de vibración

Una de las investigaciones que hace empleo de acelerómetros es la de Eriksson, Rod, Al, ton, Ilden y Balakrishnan [9] donde se diseña un sistema para la detección de baches que captura elementos en el camino (pudiendo ser éstos: ejes de peatón, o depresiones en el pavimento), determina la localización del mismo con GPS y envía estos datos a un servidor para clasificación. De esta forma, la arquitectura general del sistema consiste en 3 módulos: El módulo de sensado, el módulo de detección de baches y el módulo de clasificación; El primer módulo, se compone por 3 acelerómetros 380-Hz que sensan en tres ejes X,Y,Z, ubicados 2 en la parte delantera y uno en la parte trasera de la cabina del vehículo, de manera que las mediciones se capturen por éstos 3 y se reduzca la posibilidad de detectar incorrectamente un cierre de puertas, frenada brusca del chofer, el paso por puentes (entre otras) como un hoyo. De ésta manera, cuando en el recorrido por un sendero vial se atraviesa un bache, se capturan los abruptos cambios de energía entre los ejes X y Z, la fecha y la aceleración, que posteriormente son enviados al módulo de sensado. Este módulo, divide cada uno de los streams de información de los 3 ejes con las vibraciones, en 256 muestras o windows, y posteriormente aplica una serie de filtros (definidos en la sección [9]) a cada uno de éstas muestras por medio de la aplicación de una fórmula matemática que hace uso de un valor límite  $t$ , que permite regular la tolerancia de éstos para considerar algo como bache o no. Estos filtros permiten filtrar sólo aquellos frames que cumplan las características de interés tales como: frames donde existan diferencias de energía en los sensores pero el vehículo se encuentre en marcha, o aquellas muestras donde la variación sea mínima (de acuerdo con el límite fijado para el filtro). Para este estudio, estos filtros se calibraron a través de varias pruebas para asignar un valor  $t$  a cada uno de los filtros, de manera que se maximice la precisión en la detección de baches.

Posteriormente, los datos filtrados junto con la fecha y la localización obtenida por GPS, al disponer de un punto de acceso, pueden ser enviados a través de una red Wifi abierta o, de la red de paquetes de datos de una compañía celular, al servidor central. En este servidor se mantiene una base de datos de las detecciones y se realiza un clustering de las muestras enviadas, considerando que solamente K eventos pueden ocurrir en una localización dada mientras el vehículo se mueve en una dirección, de manera que se restringan por esta frecuencia de ocurrencias aquellos eventos que no son fallas (tales como la vibración dentro del vehículo en movimiento).

En este estudio, se ajustaron los filtros por medio de muestras tomadas manualmente (donde se registraba el tipo de evento que produce la variación en el acelerómetro, además de su fecha y posición) y algunos datos semi-automáticos (sólo contienen las variaciones del sensor de vibración, sin especificar

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

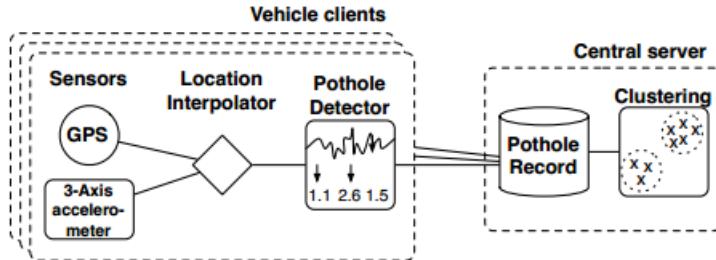


Figura 5.116: Arquitectura propuesta por Eriksson, Grod, Newton, Den y Balakrishnan. Extraído desde [9].

ubicación o fecha) evaluando los valores para  $t$  que brindan el mayor valor en la fórmula, para después validar este algoritmo con otro conjunto de datos recolectados de manera semi-automática. Los resultados obtenidos fueron satisfactorios, aunque este método tiene el defecto de detectar algunos elementos (como bocas de tormenta) que producen variaciones similares a baches como baches, y que para esta técnica funcione las ruedas del vehículo deben circular sobre el bache, por lo que si éste se ubica entre medio de las ruedas, no es sensado.

En otros estudios como [8] se han centrado en clasificar los baches según distintas métricas obtenidas a partir de las mediciones de un sensor acelerómetro, siguiendo algún estándar. En este sistema, se emplea un acelerómetro MEMS (dispositivo microscópico con partes mecánicas desplazables) modelo LIS33DE en combinación con un GPS para capturar la aceleración de los hoyos y, estos valores son limpiados eliminando datos de encendido del sensor, aplicando interpolación para obtener los datos de aceleración faltantes y se remueven aquellos valores extremos que superan un cierto límite. La información saneada de los ejes X y Z, junto con la velocidad, la fecha y la ubicación se suben a un servidor central donde se calculan el Índice de Calidad de Circulación (RQI) (en base al desvío estandar de una métrica para medir el nivel de aspereza del pavimento calculada empleando la Transformada de Fourier) y en base a éste y a la velocidad estimada, se asigna un nivel de aspereza al pavimento según el código técnico de mantenimiento CJJ36 empleado en la República China. Para el experimento se emplearon varios acelerómetros, ubicados de manera fija en la cabina del conductor, colocando el eje X para que concuerde con la dirección en que se maneja, correspondiendo el eje Z a la dirección vertical y el eje Y a la horizontal (respecto al suelo), a lo largo de distintos tipos de caminos, obteniendo medidas sin variaciones de aspereza bruscas en caminos suaves y, fluctuaciones considerables en la circularidad por caminos con fallas viales.

Sin embargo, con el avance de la tecnología móvil se han efectuado estudios como [53] y [54] que hacen uso de un acelerómetro embebido en dispositivos Android, en combinación con otros sensores en éste, para el sensado de hoyos sobre el pavimento. En [53], se emplean no solo los datos del acelerómetro, sino también del giroscopio y del compás digital. En esta aproximación, se realizó un análisis de las características que proporcionarían mayor precisión donde se emplearon dos aplicaciones existentes para los dispositivos Android: "Sensor Logger" que permite obtener los datos de los tres sensores y "Event Timestamper" que permite registrar la fecha y hora de ocurrencia de los eventos, con el fin de añadir un label a cada uno de los eventos capturados. Posteriormente, se les aplicó un filtro a las muestras para acotar el rango de frecuencias y, para reducir el uso de memoria, se realizó la suma de la información vectorial de los sensores para la extracción de features. Luego, durante la extracción de características se empleó el framework para la extracción de features de señales de una dimensión jAudio, se seleccionaron las características con mayor grado de correlación y, se probaron varios modelos de clasificación de Machine Learning de la herramienta WEKA seleccionando una red neuronal Bayesiana. Durante la selección del modelo Machine Learning, se realizaron distintas pruebas y se agruparon los eventos de circulación en distintas clases entre las que se encuentran: Eventos de Caminos, eventos de conducción (aquellos relativos al giro en ambas direcciones o frenadas), movimiento del celular o conducción suave.

Finalmente, durante la etapa de prueba se ejecutó la aplicación de logging de eventos y el algoritmo de clasificación en el dispositivo y, en lugar de emplear un timestamping, se dividió la señal del sensor en segmentos o ventanas de 5 segundos, realizando la clasificación tan pronto como el tiempo de la ventana se complete. De esta forma, se adquirieron 1196 ventanas para testeо, y los resultados se superponieron sobre el video del recorrido captado por una cámara de video, produciendo como resultado

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

final la grabación que marca los instantes en los que se circula por un bache. La precisión obtenida es de 86 % con muestras reales.

Mientras que el sistema RSMS(Roadway State Monitoring System) de información geográfica (GIS) en [54] se empleó una aplicación local en el dispositivo para la captura de datos, programada en Java con Android SDK, que se comunicaba por medio del estándar SOAP (empleando XML) con una aplicación web. En este sistema, el dispositivo de captura se encuentra colocado de forma horizontal sobre el suelo para que el eje Y del acelerómetro coincida con el sentido circulación, siendo el eje Z perpendicular a éste. Los datos sensados durante la captura, son leídos por la aplicación local en Java, la cual aplica un filtro para reducir el ruido en la señal y, junto con la información provista por un GPS(localización, velocidad,itud y longitud) y la información relativa al compás digital del dispositivo (grados azimuth que miden la orientación del dispositivo) se envían al servidor web, por medio de una conexión Wifi si esta disponible o una red GPRS.

El servidor al recibir esta información, realiza el cálculo del valor VIZIR ([46]) que asigna un índice a la muestra, el fin de estimar el daño en la superficie del pavimento y asignarle una categoría a los valores en la escala de éste. Una vez clasificada la muestra, se registra en una base de datos local en MySql, y se emplea la API de Google Maps (servidor LBS ) para indicar la ubicación de las fallas en un mapa y, además se añade funcionalidad para filtrado de fallas por fecha y por estación.

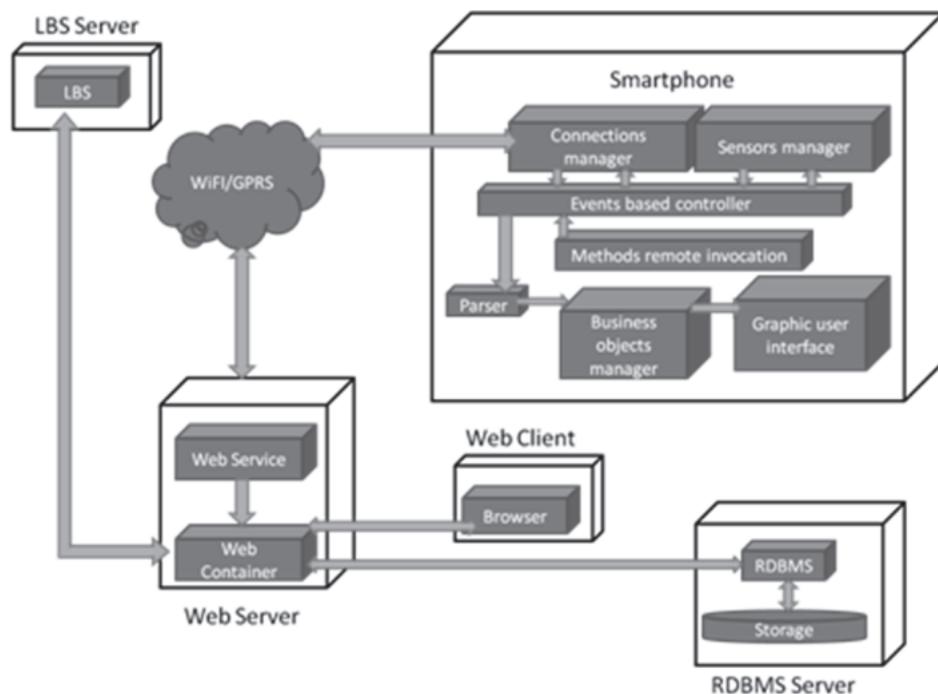


Figura 5.117: Arquitectura del sistema propuesto. Extraído desde [54].

### 4.3. Proyectos basados en la reconstrucción 3D

Con respecto al empleo de modelos 3D para la detección de fallas, entre las aproximaciones que se pueden mencionar, se encuentra la tesis [51] de Sikai Xie, donde se plantea la detección de distintos tipos de grietas y el cálculo de sus propiedades (largo, grosor y profundidad), por medio de la computación de un modelo tridimensional a partir las coordenadas en X, Y, Z recidas en las nubes de puntos del sensor Kinect. Esta metodología, comienza con la conversión de la información espacial de la nube de puntos, a un modelo 3D compuesto de vértices y aristas, por medio de la función de Kinect Fusion ofrecida en la SDK de Microsoft. Este algoritmo consiste en combinar la información de distintas nubes de puntos capturadas desde distintas ubicaciones para obtener datos geométricos 3D de alta resolución, efectuando para ello un seguimiento tanto de la orientación de la cámara como de la posición y, debido a que se conoce la posición de cada sub-captura y como se relaciona con las demás, éstas se combinan en una

única nube de salida. Kinect Fusion emplea internamente ICP (Iterative Closest Point) como herramienta para fusionar varias nubes de puntos en una única;  Funcionamiento del mismo, consiste en mantener una captura fija (vertex cloud) y aplicar diversas transformaciones combinadas de traslación y rotación a la nube fuente (source), necesarias para minimizar el error o diferencia entre ambas nubes de puntos. Como salida a este algoritmo, se obtiene una malla de polígonos desorganizada (polygon soup) que no cuenta con información respecto a vértices vecinos compartidos por distintos triángulos, por lo que ésta es procesada nuevamente para eliminar la redundancia de vértices, conservando aquellos que se encuentran a una distancia determinada.

Posteriormente, se procede a realizar el análisis de la grieta en la malla de triángulos 3D, definiendo para cada vértice  $V$  un conjunto de vértices  $k$ -vecinos, que pueden ser alcanzados desde el mismo recorriendo  $k$  aristas y, un conjunto de triángulos  $k$ -vecinos que contienen al menos un vértice en el camino de los  $k$ -vecinos. Así, para cada uno de los triángulos vecinos del vértice  $V$ , se aplica la ecuación (1.1) en [51] que emplea la normal de los triángulos y el área de cada triángulo, computando la dirección del vector normal promediado para ese conjunto de triángulos. Luego se calcula el  entre éste y cada una de las normales de los triángulos vecinos y, se calcula el desvío estandar, considerándose que ese conjunto de triángulos pertenecen a una grieta si se supera un valor determinado de desvío. Luego, para determinar si los vértices seleccionados son vértices con una diferencia abrupta y por lo tanto pertenecen a una grieta, se emplea el mapeo de Gauss discreto (DGM), donde se mapea el conjunto de vértices por medio de la fórmula (1.3) en [51] a una esfera cuyo centro es el vértice  $V$ , formando clusters de puntos. Finalmente, se refina el agrupamiento de puntos en clusters midiendo la distancia geodésica (unidad de medida empleada en el campo de las superficies esféricas) entre cada par de puntos de los clusters, reagrupandolos según un parámetro de sensibilidad. Finalmente, si la cantidad de clusters se encuentra entre 2 y 4 se considera a  $V$  como un vértice abrupto.

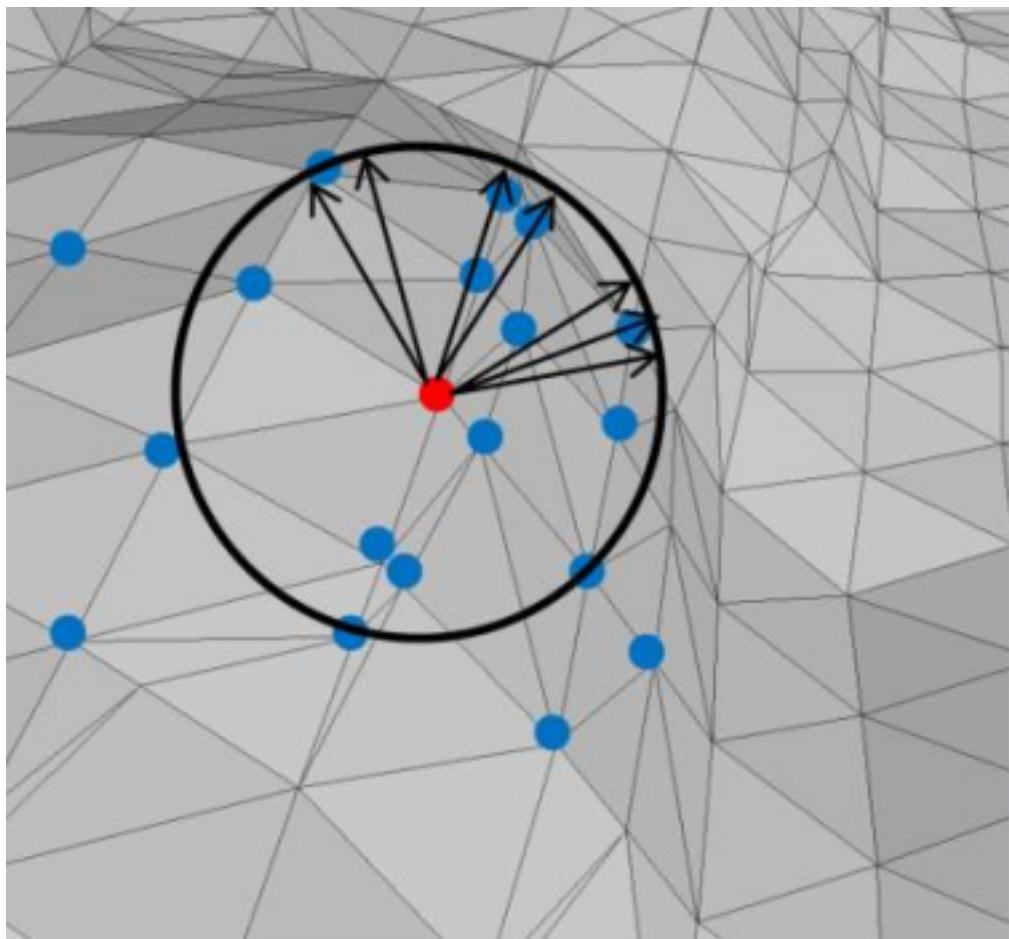


Figura 5.118: DGM aplicado a un vértice  $V$  de la malla. Extraído desde Fig.4 en [51].

Con el fin de separar los vértices aislados de los que se encuentran unidos a una grieta, se

#### 4. PROYECTOS DE SOFTWARE ANTERIORES PARA LA DETECCIÓN DE FALLAS SOBRE EL PAVIMENTO

---

emplea el algoritmo de búsqueda BSF (Breath-first Search) para retornar subconjuntos de vértices conectados en la maya, y se filtran solamente aquellos conjuntos que tienen un tamaño inferior a un valor establecido.

El cálculo de la profundidad se realiza por medio de la computación de aquellos vértices que no son parte del grupo de vértices abruptos, aplicando la ecuación (1.10) en [51] basada en el método de mínimos cuadrados (Least Square Fitting) con la fórmula de la esfera, de manera que la profundidad de un vértice se calcula como la distancia mínima a la superficie de una esfera. Adicionalmente, se subdivide la grieta en una grilla de 10x10 y se calcula el valor de profundidad de aquellas grietas que tienen un nivel mayor de profundidad. Con respecto al grosor se aplicó la sumatoria de los cuadrados que conforman la grieta, mientras que para el cálculo del grosor se aplicó la fórmula (1.11) de [51], que calcula la distancia entre el vértice que se encuentra a mayor profundidad y un vértice que se encuentra más próximo al borde de la grieta (que es parte de la superficie del asfalto). Este procedimiento se realiza 3 veces, con tres puntos del borde de ésta para lograr una mejor aproximación. Finalmente, se realizaron pruebas contando con 290 muestras en promedio por prueba para distintos tipos de grietas, transversales, longitudinales y alligator (varias grietas interconectadas entre si) detectando el 67% de grietas transversales, 77% de grietas longitudinales y solo 54% de las grietas alligator.

---

## Bibliografía

---

- [1] Fundamentals of image processing. URL: <http://www.cs.dartmouth.edu/farid/downloads/tutorials/fip.pdf>.
- [2] Similarity measure. URL: [https://en.wikipedia.org/wiki/Similarity\\_measure](https://en.wikipedia.org/wiki/Similarity_measure).
- [3] Spectral clustering. URL: [https://en.wikipedia.org/wiki/Spectral\\_clustering](https://en.wikipedia.org/wiki/Spectral_clustering).
- [4] -. -. URL: [http://www.lcc.uma.es/~munozp/documentos/procesamiento\\_de\\_imagenes/temas/pi\\_cap6.pdf](http://www.lcc.uma.es/~munozp/documentos/procesamiento_de_imagenes/temas/pi_cap6.pdf).
- [5] -. Color image processing slides. URL: <https://www.slideshare.net/lineking/10-color-image-processing-dip>.
- [6] \_\_\_\_\_. \_\_\_\_\_. URL: <http://civilgeeks.com/2012/03/27/juntas-en-el-concreto/>.
- [7] \_\_\_\_\_. \_\_\_\_\_. URL: [https://www.ieca.es/Uploads/docs/Diseño\\_y\\_ejecución\\_de\\_juntas.pdf](https://www.ieca.es/Uploads/docs/Diseño_y_ejecución_de_juntas.pdf).
- [8] a,b , c, d, e,. Road condition monitoring using on-board three-axis accelerometer. 2011.
- [9] Jakob Eriksson ,Lewis Girod ,BretHull ,Ryan Newton ,Samuel Madden ,Hari Balakrishnan. The pothole patrol: using a mobile sensor network for road surface monitoring. *MobiSys*, 2008.
- [10] AL Bovik. *Handbook of Image and Video Processing*. Academic Press, 2000.
- [11] AL Bovik. *The essential guide to Image Processing*. Academic Press, 2009.
- [12] Christian Korch , Ioannis Brikalis. Improving pothole recognition through vision tracking for automated pavement assessment. *Proceedings of the 2011 EG-ICE International Workshop on Intelligent Computing in Engineering*, 2011.
- [13] Christian Korch , Ioannis Brikalis. Pothole detection in asphalt pavement images. *Advanced Engineering in Informatics*, 2011.
- [14] Harvey Rhody, Chester F. Carlson. Geometric image transformations. URL: <https://www.math.hkbu.edu.hk/~zeng/Teaching/math3615/ls.pdf>.
- [15] Albert Einstein. Zur Elektrodynamik bewegter Körper. (German) [On the electrodynamics of moving bodies]. *Annalen der Physik*, 322(10):891–921, 1905. doi:<http://dx.doi.org/10.1002/andp.19053221004>.
- [16] Research Gate. Voxel grid spanning a volume in a 3d space bounded by (x min,x max)(y min, y max). URL: [https://www.researchgate.net/figure/259053167\\_fig3\\_Figure-3-Voxel-grid-spanning-a-volume-in-a-3D-space-bounded-by-x-min-x-max-y-min](https://www.researchgate.net/figure/259053167_fig3_Figure-3-Voxel-grid-spanning-a-volume-in-a-3D-space-bounded-by-x-min-x-max-y-min).
- [17] PCL Github. Overview and comparison of features (algoritmos para obtencion de descriptores). URL: <https://github.com/PointCloudLibrary/pcl/wiki/Overview-and-Comparison-of-Features>.
- [18] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The \LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.
- [19] Yao Wang, Zhu Liu, Onur Gulerzyuz. Image processing. geometric transformations,warping,registration,morphing. URL: [http://eeweb.poly.edu/~yao/EL5123/lecture12\\_ImageWarping.pdf](http://eeweb.poly.edu/~yao/EL5123/lecture12_ImageWarping.pdf).

- [20] Tomoyuki Yamaguchi ,Shingo Nakamura ,Ryo Saegusa , Shuji Hashimoto. Image-based crack detection for real concrete surfaces. *Wiley InterScience*, 2008.
- [21] Emir Buza , Samir Omanovic , Alvin Huseinovic. Pothole detection with image processing and spectral clustering. *Recent Advances in Computer Science and Networking*, 2013.
- [22] Seing-Ku Ryu , Taehyong Kim , Young-Ro Kim. Image-based pothole detection system for its service and road management system. *Matematical problems in Engineering*, 2015.
- [23] Donald Knuth. Knuth: computers and typesetting. URL: <http://www-cs-faculty.stanford.edu/~uno/abcde.html>.
- [24] S Nieber , M Brooysen , R Kroon. Detection potholes using simple processing techniques and real-world footage. *Proceedings of the 34th Southern African Transport Conference*, 2015.
- [25] Microsoft. Kinect fusion. URL: <https://msdn.microsoft.com/en-us/library/dn188670.aspx>.
- [26] Alasdair McAndrew. Victoria University of Technology. An introduction to digital image processing with matlab. URL: <https://www.math.hkbu.edu.hk/~zeng/Teaching/math3615/ls.pdf>.
- [27] J. R. Parker. *Algorithms for Image Processing and Computer Vision*. Wiley Publishing , Inc., 2011.
- [28] PCL. Downsampling a pointcloud using voxelgrid filter. URL: [http://pointclouds.org/documentation/tutorials/voxel\\_grid.php](http://pointclouds.org/documentation/tutorials/voxel_grid.php).
- [29] PCL. Interest point detection. URL: [https://en.wikipedia.org/wiki/Interest\\_point\\_detection](https://en.wikipedia.org/wiki/Interest_point_detection).
- [30] PCL. Keypoints and features. URL: [http://www.pointclouds.org/assets/uploads/cglibs13\\_features.pdf](http://www.pointclouds.org/assets/uploads/cglibs13_features.pdf).
- [31] PCL. Model fitting(ransac). URL: [http://robotica.unileon.es/index.php/PCL/OpenNI\\_tutorial\\_3:\\_Cloud\\_processing\\_\(advanced\)#Model\\_fitting\\_.28RANSAC.29](http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_3:_Cloud_processing_(advanced)#Model_fitting_.28RANSAC.29).
- [32] PCL. Module keypoints. URL: [http://docs.pointclouds.org/trunk/group\\_\\_keypoints.html](http://docs.pointclouds.org/trunk/group__keypoints.html).
- [33] PCL. Pcl segmentation. URL: <http://www.pointclouds.org/assets/iros2011/segmentation.pdf>.
- [34] PCL. Plane model segmentation. URL: [http://pointclouds.org/documentation/tutorials/planar\\_segmentation.php](http://pointclouds.org/documentation/tutorials/planar_segmentation.php).
- [35] PCL. Removing outliers using a statisticaloutlierremoval filter. URL: [http://pointclouds.org/documentation/tutorials/statistical\\_outlier.php](http://pointclouds.org/documentation/tutorials/statistical_outlier.php).
- [36] ROS PCL. URL: <http://wiki.ros.org/pcl>.
- [37] PCL Pointcloud. Documentacion de pcl. URL: <http://www.pointclouds.org/documentation/>.
- [38] PCL Pointcloud. Tutoriales de pcl. URL: <http://www.pointclouds.org/documentation/tutorials/>.
- [39] Nibcode Solutions. El algebra lineal y el procesamiento digital de imágenes. URL: <http://www.nibcode.com/es/blog/14/linear-algebra-and-digital-image-processing-part-III-affine-transformations>.
- [40] Robotica Unileon. Pcl/openni tutorial 0: the very basics. URL: [http://robotica.unileon.es/index.php/PCL/OpenNI\\_tutorial\\_0:\\_The\\_very\\_basics](http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_0:_The_very_basics).
- [41] Robotica Unileon. Pcl/openni tutorial 1: installing and testing. URL: [http://robotica.unileon.es/index.php/PCL/OpenNI\\_tutorial\\_1:\\_Installing\\_and\\_testing](http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_1:_Installing_and_testing).
- [42] Robotica Unileon. Pcl/openni tutorial 2: cloud processing (basic). URL: [http://robotica.unileon.es/index.php/PCL/OpenNI\\_tutorial\\_2:\\_Cloud\\_processing\\_\(basic\)](http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_2:_Cloud_processing_(basic)).
- [43] Robotica Unileon. Pcl/openni tutorial 3: cloud processing (advanced). URL: [http://robotica.unileon.es/index.php/PCL/OpenNI\\_tutorial\\_3:\\_Cloud\\_processing\\_\(advanced\)](http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_3:_Cloud_processing_(advanced)).
- [44] Robotica Unileon. Pcl/openni tutorial 4: 3d object recognition (descriptros). URL: [http://robotica.unileon.es/index.php/PCL/OpenNI\\_tutorial\\_4:\\_3D\\_object\\_recognition\\_\(descriptors\)](http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_4:_3D_object_recognition_(descriptors)).
- [45] Robotica Unileon. Pcl/openni tutorial 5: 3d object recognition (pipeline). URL: [http://robotica.unileon.es/index.php/PCL/OpenNI\\_tutorial\\_5:\\_3D\\_object\\_recognition\\_\(pipeline\)](http://robotica.unileon.es/index.php/PCL/OpenNI_tutorial_5:_3D_object_recognition_(pipeline)).

- [46] Cerón ,Bermudez V.G. Evaluación y comparación de metodologías vizir y pci sobre el tramo de vía en pavimento flexible y rígido de la vía: museo quimbaya-crq armenia quindío. 2006.
- [47] Ian T. Young , Jan J. Gerbrans , Lucas J. van Vliet. Fundamentals of image processing. URL: [http://www.tnw.tudelft.nl/fileadmin/Faculteit/TNW/Over\\_de\\_faculteit/Afdelingen/Imaging\\_Science\\_and\\_Technology/Research/Research\\_Groups/Quantitative\\_Imaging/Education/doc/FIP2\\_3.pdf](http://www.tnw.tudelft.nl/fileadmin/Faculteit/TNW/Over_de_faculteit/Afdelingen/Imaging_Science_and_Technology/Research/Research_Groups/Quantitative_Imaging/Education/doc/FIP2_3.pdf).
- [48] T.PRATHIBA , THAMARAISELVI.M , MOHANASUNDARI.M , VEERELAKSHMI.R. Pothole detection in road using image processing. *Interational Journal of Managment,Information*, 2015.
- [49] Wikipedia. Otsu's method. URL: [https://en.wikipedia.org/wiki/Otsu's\\_method](https://en.wikipedia.org/wiki/Otsu's_method).
- [50] Rafael C. Gonzales , Richard E. Woods. *Digital image processing 3rd Edition*. Pretince Hall, 2002.
- [51] Sikai Xie. 3d pavement surface reconstruction and cracking recognition using kinect-based solution. 2015.
- [52] Richie Zirbes. Scientific visualization: volume surface rendering. URL: <http://johnrichie.com/V2/ricchie/isosurface/volume.html>.
- [53] , , , , . Assessment of vehicular transportation quality via smartphones. 2013.
- [54] , , , , . Detection and localization of potholes in roadways using smartphones. 2014.

First document. This is a simple example, with no extra parameters or packages included.