

**ALGORITMO PARA RECONOCIMIENTO DE FALLAS TIPO FISURA EN  
PAVIMENTO RÍGIDO MEDIANTE TRATAMIENTO DE IMÁGENES DIGITALES**

**MIGUEL ANDRÉS CASTILLO RANGEL  
SULAY TATIANA MESA RIBERO**

**FACULTAD DE CIENCIAS FÍSICO-MECÁNICAS  
ESCUELA DE INGENIERÍA CIVIL  
BUCARAMANGA, SANTANDER  
2011**

**ALGORITMO PARA RECONOCIMIENTO DE FALLAS TIPO FISURA EN  
PAVIMENTO RÍGIDO MEDIANTE TRATAMIENTO DE IMÁGENES DIGITALES**

**Miguel Andrés Castillo Rangel  
Sulay Tatiana Mesa Ribero**

**Trabajo de grado para optar el título de Ingeniero Civil**

**Director: PhD. Hernán Porras Díaz  
Co-director: Ingeniero Duvan Yahir Sanabria Echeverry**

**Facultad de Ciencias Físico-Mecánicas  
Escuela de Ingeniería Civil  
Bucaramanga, Santander  
2011**

## **DEDICATORIA**

A Dios, a mi familia y a mi novia Natalia.

*Miguel Castillo*

Este proyecto está dedicado primero a Dios, quien me da todo en la vida y segundo a todas las personas que de alguna manera estuvieron aportando para hacerlo posible.

*Sulay Mesa*

## AGRADECIMIENTOS

A Dios, a quien todo le debo; a mi familia, y en especial a mi mamá, por su amor y compañía en este proceso, y por haber forjado en mí la persona que hoy alcanza este éxito; a Natalia, quien con su amor y amistad ha creído en mí más que yo mismo, y con quien quiero compartir todos los logros de mi vida; a la Universidad Industrial de Santander; a la Escuela de Ingeniería Civil; al grupo de investigación Geomática, y en particular al Doctor Hernán Porras y a los ingenieros Duvan Sanabria y Yerly Martínez, por haberme brindado la posibilidad de enseñar, investigar y encontrar mi vocación profesional; y finalmente, a mis amigos, con quienes aprendí a apreciar el arte y el oficio de la Ingeniería Civil.

*Miguel Castillo*

Proverbios 22:4

Para mí todo lo que en la vida pueda llegar a ser o a realizar, sólo podrá ser obra de ese ser supremo que es Dios, es a quién le agradezco todo lo que me ha dado y lo que me seguirá dando, porque sin Él nada sería posible, es la fortaleza que necesito para levantarme siempre y seguir adelante.

Gracias a mi familia, padres y hermanos, quienes estuvieron pendientes de mí en este tiempo, que fueron de gran apoyo y que seguirán siendo esas personas que a pesar de los conflictos siempre intentarán estar juntos.

Siempre he pensado que ninguna persona pasa sin ningún motivo por nuestras vidas, en cambio cada uno tiene un propósito, de todos se aprende, por eso no podría dejar de agradecerle a todos mis compañeros y profesores de carrera que me ayudaron a crecer profesional y personalmente.

Por último, quiero agradecerle a esos amigos que durante estos cinco años me vieron llorar, reír, molestar, que me soportaron, que me explicaron y que en todo momento estuvieron apoyándome y dándome el ánimo que necesitaba en cada situación.

Hace cinco años pensaba que la mejor etapa de mi vida había sido el colegio, hoy puedo decir que la universidad es un tesoro que al explorar correctamente se pueden llevar grandes enseñanzas y que cada etapa de nuestra vida es única e irrepetible.

*Sulay Mesa*

# CONTENIDO

Pág.

Introducción.....	19
Objetivos .....	21
Objetivo General.....	21
Objetivos Específicos .....	21
Alcance.....	22
Justificación.....	23
1. Estado del arte .....	24
1.1 Tipos de fisuras en el pavimento rígido .....	24
1.1.1 Fisuras de borde.....	25
1.1.2 Fisura longitudinal.....	25
1.1.3 Fisura transversal .....	25
1.1.4 Fisuras en bloque .....	26
1.2 Técnicas para el tratamiento de imágenes digitales en el reconocimiento de fisuras en el pavimento.....	26
1.2.1 Recolección de las imágenes digitales.....	27
1.2.2 Pre-procesamiento de la imagen digital.....	30
1.2.3 Segmentación.....	35
1.2.4 Operaciones morfológicas .....	38
1.2.5 Caracterización de las fisuras en el pavimento .....	40
1.2.6 Algoritmos de clasificación.....	45
1.3 Algoritmos para el reconocimiento de fisuras en el pavimento mediante imágenes digitales .....	50
2. Recolección de las imágenes digitales .....	53
2.1 Metodología para la recolección de las imágenes digitales.....	53
2.2 propuesta de un sistema semi-automático para la recolección de las imágenes digitales.....	54

2.2.1	Esquema del sistema semi-automático .....	55
2.2.2	Especificaciones del sistema semi-automático.....	55
2.2.3	Características del sistema semi-automático .....	56
3.	Pre-procesamiento .....	57
3.1	Filtrado .....	57
3.2	Homogenización de la imagen.....	58
4.	Segmentación .....	60
4.1	Crecimiento por regiones.....	61
4.2	Algoritmo para eliminar el ruido y las juntas de las losas .....	63
4.3	Algoritmo para unir las regiones .....	64
5.	Caracterización de las fisuras en el pavimento rígido.....	67
5.1	Clases de fisuras en el pavimento rígido .....	67
5.2	Descriptores de las fisuras en el pavimento rígido .....	67
6.	Clasificación de fisuras en el pavimento rígido .....	69
6.1	Arquitectura de la red neuronal.....	69
6.2	Entrenamiento de las redes neuronales .....	70
6.3	Resultados del entrenamiento de las redes neuronales .....	70
7.	Validación del algoritmo propuesto .....	73
7.1	Validación dentro de la metodología propuesta.....	73
7.2	Validación del algoritmo en tres escenarios de captura.....	74
8.	Algoritmo para el reconocimiento de fisuras en pavimentos de losa de concreto .....	76
8.1	Diagrama de flujo del algoritmo propuesto .....	76
8.2	Tiempo de ejecución del algoritmo .....	77
8.3	Interfaz gráfica del usuario ALFIL .....	77

9. Conclusiones.....	80
10. Recomendaciones.....	82
Bibliografía .....	83
Anexos.....	86

## LISTA DE FIGURAS

	Pág.
Figura 1. Ancho de las fisuras .....	25
Figura 2. Fisura de borde y longitudinal .....	26
Figura 3. Fisura transversal y figura de bloque .....	27
Figura 4. Esquema general de un sistema de recolección de imágenes digitales para reconocer fisuras en el pavimento .....	28
Figura 5. Sistemas de recolección .....	29
Figura 6. Filtro lineal y no-lineal en una imagen digital .....	31
Figura 7. Técnica de ecualización del histograma .....	32
Figura 8. Uso de operadores de detección de bordes .....	33
Figura 9. Técnicas de homogenización del fondo de la imagen .....	34
Figura 10. Métodos de segmentación por umbral.....	36
Figura 11. Otras técnicas de segmentación.....	38
Figura 12. Operación de erosión.....	39
Figura 13. Operación de dilatación.....	39
Figura 14. Códigos empleados en la representación por el código de la cadena.....	41
Figura 15. Representación por el código de la cadena.....	41
Figura 16. Representación del esqueleto de una fisura en las imágenes digitales.....	42
Figura 17. Elipse con igual segundo momento central que la región.....	44
Figura 18. Principales descriptores de textura .....	45
Figura 19. Estructura de una red neuronal artificial.....	47
Figura 20. Funciones de activación logsigmoide, tansigmoide y lineal en las redes neuronales.....	47
Figura 21. Clasificador por máquinas de soporte vectorial en clases separables linealmente .....	49
Figura 22. Técnica de cruce en dos puntos en un algoritmo genético .....	50
Figura 23. Muestra de las imágenes digitales recolectadas.....	54
Figura 24. Comparación del filtro mediano con un filtro lineal.....	57
Figura 25. Máscaras empleadas para el filtro mediano.....	58

Figura 26. Homogenización de la imagen en escala de grises.....	59
Figura 27. Resultados de las distintas técnicas de segmentación.....	60
Figura 28. Comparación de las técnicas de segmentación de recorte sucesivo y crecimiento por regiones. ....	61
Figura 29. Histogramas empleados para definir los parámetros del algoritmo de crecimiento por regiones. ....	62
Figura 30. Representación del funcionamiento del algoritmo de crecimiento por regiones. ....	63
Figura 31. Influencia del pre-procesamiento de la imagen en la segmentación. ....	63
Figura 32. Rectángulos de búsqueda empleados en el algoritmo para unir regiones según la orientación de la región base. ....	65
Figura 33. Implementación del algoritmo para eliminar juntas de losas y otras regiones. ....	66
Figura 34. Implementación del algoritmo para unir regiones. ....	66
Figura 35. Gráfica de confusión de la red neuronal [3 6 10 5] .....	72
Figura 36. Imagen del pavimento sin fisuras.....	75
Figura 37. Imagen de resolución de 480x640 .....	75
Figura 38. Imagen del pavimento en condiciones húmedas .....	75
Figura 39. Diagrama de flujo del algoritmo ALFIL.....	76
Figura 40. Interfaz gráfica del usuario para el algoritmo ALFIL.....	78
Figura 41. Fisura longitudinal y de borde. ....	78
Figura 42. Fisura de bloque y transversal. ....	79
Figura 43. Fisura de bloque y longitudinal.....	79
Figura 44. Fisura longitudinal y transversal.....	79

## LISTA DE TABLAS

Pág.

Tabla 1. Caracterización morfológica de las fisuras, tramos de juntas y otras regiones....	64
Tabla 2. Comportamiento de los descriptores en cada clase.....	68
Tabla 3. Configuraciones de las redes neuronales evaluadas.....	69
Tabla 4. Parámetros del progreso de la red neuronal durante el entrenamiento.....	71
Tabla 5. Resultados globales del entrenamiento de las redes neuronales.....	71
Tabla 6. Resultados de la validación de la RNA dentro de la metodología propuesta.....	73
Tabla 7. Resultados de la validación en distintos escenarios de captura.....	74
Tabla 8. Tiempos de ejecución de los algoritmos en Matlab.....	77

## LISTA DE ANEXOS

	Pág.
Anexo A. Esquema gráfico del sistema semi-automático propuesto para la recolección de las imágenes digitales .....	86
Anexo B. Códigos en Matlab de los algoritmos propuestos .....	87
Algoritmo para el reconocimiento de fisuras en pavimentos de losas de concreto (ALFIL) .....	87
Algoritmo para homogeneizar la imagen (imhomogenea) .....	87
Algoritmo de crecimiento por regiones (cregiones) .....	88
Algoritmo para eliminar juntas y otras regiones (eregiones).....	89
Algoritmo para unir regiones (uregiones).....	89
Anexo C. Manual de la interfaz gráfica del usuario ALFIL .....	92

## GLOSARIO

**ALGORITMO:** Es la organización de un número finito de operaciones que permite solucionar un problema.

**CÁMARA DIGITAL:** Es una cámara fotográfica que captura la imagen en un sensor electrónico y la almacena en una memoria digital.

**CÁMARAS DE ESCANEADO DE ÁREA:** Son cámaras digitales cuyo sensor cubre un área del objeto, resultando en arreglos de píxeles matriciales.

**CÁMARAS DE ESCANEADO LINEAL:** Son cámaras digitales cuyo sensor es lineal, por lo que construye la imagen línea a línea; son ideales para capturar objetos en movimiento o cuando la cámara se mueva con respecto al objeto.

**DIAGRAMA DE FLUJO:** Es la representación gráfica del esquema de un algoritmo para que su interpretación sea más sencilla.

**FISURA:** Es una hendidura o discontinuidad presente en la superficie del pavimento.

**HISTOGRAMA:** Es la representación gráfica de las intensidades de grises que tiene cada imagen.

**LUZESTROBOSCÓPICA:** Fuente luminosa que emite una serie de destellos muy breves y sucesivos, empleado para producir exposiciones de un objeto en movimiento.

**LUZ HALÓGENA:** Fuente luminosa que emite un espectro continuo de luz y que funcionan a baja tensión, lo que permite que sean más pequeñas y potentes que las luces incandescentes.

**IMAGEN DIGITAL:** Es una imagen que se captura a través de un medio electrónico y se representa en un archivo de información leído por una serie de pulsos electrónicos.

**PAVIMENTO:** Es una estructura conformada por un conjunto de capas de material cuya función es recibir las cargas del tránsito y transmitir las al suelo, de manera que la carga se vaya disipando, además de garantizar un rodamiento cómodo para los vehículos.

**PIXEL:** Es la unidad mínima para almacenar información en una imagen digital.

**TRATAMIENTO DIGITAL DE IMÁGENES:** Son las técnicas que se le aplican a las imágenes digitales para mejorar la calidad y extraer información específica.

**UMBRAL:** Es el valor por el cual se empieza a ser perceptible cierto efecto físico que se quiera ver.

## RESUMEN

**TÍTULO:** ALGORITMO PARA RECONOCIMIENTO DE FALLAS TIPO FISURA EN PAVIMENTO RÍGIDO MEDIANTE TRATAMIENTO DE IMÁGENES DIGITALES\*

**AUTORES:** MIGUEL ANDRÉS CASTILLO RANGEL Y SULAY TATIANA MESA RIBERO\*\*

**PALABRAS CLAVES:** Algoritmo, Pavimento rígido, Fisuras, Procesamiento de imágenes digitales.

La necesidad de avanzar hacia planes de gestión de pavimentos más eficaces, eficientes y económicos ha impulsado el uso de tecnologías para la evaluación del estado del pavimento, destacándose el reconocimiento de fisuras mediante el procesamiento de imágenes digitales. En este contexto, y como parte del desarrollo de un sistema automatizado para la clasificación y evaluación del estado del pavimento en vías urbanas para Bucaramanga y su área metropolitana, en el presente trabajo se propuso un algoritmo capaz de identificar, clasificar y medir las fisuras registradas en imágenes digitales del pavimento rígido.

Para realizar el algoritmo, se partió de establecer un procedimiento de recolección de las imágenes digitales, las cuales fueron pre-procesadas con un filtro mediano y una homogenización del fondo. Para segmentar las fisuras se empleó la técnica de crecimiento por regiones, seguida de dos algoritmos: el primero para eliminar los objetos que por su forma discreparan de una fisura, y el segundo para unir los tramos de grietas segmentados. Posteriormente, se establecieron cinco clases de fisuras: borde, longitudinal, transversal, de bloque u otro objeto, las cuales fueron caracterizadas con tres descriptores de forma (extensión, elongación y orientación) y clasificadas con una red neuronal perceptron multicapa. Finalmente, el algoritmo desarrollado se validó para imágenes capturadas dentro y fuera del procedimiento de recolección, obteniéndose, respectivamente, porcentajes de acierto en el clasificador de 84 y 74%.

En definitiva, el algoritmo propuesto presenta una gran capacidad de generalización y es considerado novedoso en su tipo por representar, caracterizar y clasificar las fisuras como regiones dentro de las imágenes digitales. Asimismo, los tiempos de ejecución medidos tras implementar el algoritmo en el *software* Matlab, permiten que éste sea empleado en cualquier sistema semi-automático para la evaluación del estado del pavimento.

---

\*Proyecto de grado.

\*\*Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería Civil. Director: PhD. Hernán Porras Díaz. Co-director: Ing. DuvanYahir Sanabria Echeverría.

## ABSTRACT

**TITLE:** CONCRETE PAVEMENT CRACKS RECOGNITION ALGORITHM BASED ON DIGITAL IMAGE PROCESSING\*

**AUTHORS:** MIGUEL ANDRES CASTILLO RANGEL Y SULAY TATIANA MESA RIBERO\*\*

**KEY WORDS:** Algorithm, Concrete pavement, Cracks, Digital image processing.

The necessity to improve the pavement management plans in terms of efficacy, efficiency and economy have increased the use of technologies for the pavement evaluation, being remarkable the role played by the pavement cracks recognition based on digital image processing. In this context, and as a part of the development of an automated pavement classification and evaluation system, an algorithm capable of identify, classify and measure the distress presents in concrete pavement digital images was proposed.

To realize the algorithm, a proceeding for the digital images recollection was established at first, and then a median filter and a background homogenization technique were used to pre-process the images collected. For the cracks segmentation, the region grow technique was used, followed with two algorithms focused on: eliminate the pavement joints or another object which had been segmented but that does not seems like a pavement distress, and join the cracks pieces previously segmented. Latterly, five concrete pavement cracks classes were established: edge, longitudinal, transversal, block and another object, which each of one were characterized using three shape descriptors (extension, elongation and orientation), and then were classified through a perceptron multilayer artificial neuronal network. Finally, the developed algorithm was validated for images that were captured either following or not the established pavement crack recollection proceeding, obtaining, respectively, an accuracy percentage of 84 y 74%.

In conclusion, the proposed algorithm presents a great generalization capacity and could be considered as novel in its kind because it represents, characterizes and classifies the concrete pavement distress as regions inside the digital images. Also, the running times registered once the algorithm was implemented in the software Matlab, could asseverate that it may be uses on any semi-automatic pavement evaluation system.

---

\* Work of degree.

\*\*Phishyc- Mechanic Engineering Faculty.Civil Engineering school.Director: PhD. Hernán Porras Díaz. Co-director: Ing. DuvanYahir Sanabria Echeverría.

## INTRODUCCIÓN

A nivel mundial el desarrollo económico ha estado ligado a la promoción de la construcción y conservación de la infraestructura vial, y esta última, a su vez, ha repercutido en un incremento del desarrollo económico<sup>1</sup>. Esta relación no es ajena a Colombia, donde el retraso en materia de infraestructura vial ha obstaculizado el progreso económico, y viceversa. En particular, a nivel urbano, las vías terrestres de nuestro país han sufrido un abandono en cuanto al mantenimiento de sus estructuras de pavimento, lo que ha repercutido en un detrimento del servicio que dichas vías prestan a la movilidad de las personas y al desarrollo de las actividades económicas.

La deficiencia que han presentado los planes de mantenimiento vial en Colombia se hace manifiesta en casos como el de Bucaramanga, donde actualmente el 31% de las vías presentan un grado de deterioro severo en su capa de pavimento<sup>2</sup>, lo que ha ocasionado conflictos de tráfico, pérdidas económicas e inconformismo en la ciudadanía. La premisa anterior plantea un desafío para los planes de mantenimiento vial, los cuales deben contar con mejores y más eficaces sistemas de evaluación y rehabilitación de las capas de rodadura de la malla vial.

En este sentido, la Universidad Industrial de Santander ha presentado a Colciencias el proyecto denominado “Sistema automatizado para clasificación y evaluación de daños de pavimentos de vías urbanas”, en el cual se encuentra inmerso el presente trabajo con la propuesta “Algoritmo para reconocimiento de fallas tipo fisura en pavimento rígido mediante tratamiento de imágenes digitales”. En particular, éste último tiene por objeto principal proponer un algoritmo que sea capaz de detectar, clasificar y medir de las fisuras presentes en las vías urbanas construidas en pavimento rígido.

Para plantear el algoritmo, en primera instancia, se identifican las distintas metodologías de recolección de imágenes digitales de fisuras en el pavimento presentes en la literatura, y a raíz de ello, se propone un procedimiento de captura para el presente trabajo. Posteriormente, se implementa un algoritmo para segmentar las grietas en las imágenes digitales, se definen las clases de fisuras a contemplar, se caracterizan las mismas y se desarrolla un algoritmo de clasificación para reconocer las fisuras en el pavimento rígido. Finalmente, se valida el algoritmo propuesto para imágenes que se tomaron con la metodología propuesta y para imágenes que se tomaron por fuera de ésta.

---

<sup>1</sup> OMC. Informe sobre el comercio mundial 2004. II Coherencia. Capítulo B: La infraestructura en el comercio y el desarrollo económico. Berna: OMC, 2004. p. 124 – 161.

<sup>2</sup> Alcaldía de Bucaramanga – Universidad Industrial de Santander. Modelamiento para la gestión del mantenimiento, rehabilitación y mejoramiento de pavimentos urbanos. Bucaramanga: La Alcaldía - UIS, 2011. p.14.

De acuerdo a la metodología señalada, en el Capítulo 1 se presentan el marco teórico y referencial sobre la recolección y el reconocimiento de fisuras en el pavimento rígido mediante imágenes digitales. En el Capítulo 2 se establece la metodología seguida para recolectar las imágenes digitales de fisuras en el pavimento rígido. Por otro lado, en el Capítulo 3 se presentan las técnicas de tratamiento de imágenes digitales utilizadas para el pre-procesamiento de las imágenes recolectadas, y en el Capítulo 4 se indica el algoritmo de segmentación implementado para separar las fisuras del fondo de la imagen. En los Capítulos 5 y 6 se presentan, respectivamente, la caracterización de las fisuras y el clasificador desarrollado para el reconocimiento de las mismas en las imágenes digitales. En el Capítulo 7 se señalan los resultados de la validación para distintos escenarios de captura (dentro y fuera de la metodología de recolección propuesta), y por último, en el Capítulo 8 se presenta la implementación del algoritmo denominado como Algoritmo Para El Reconocimiento De Fisura En Pavimentos De Losas De Concreto (ALFIL) en el *software* Matlab.

# **OBJETIVOS**

## **OBJETIVO GENERAL**

Proponer un algoritmo para reconocer fallas tipo fisura en pavimentos rígidos tomadas a partir de imágenes digitales.

## **OBJETIVOS ESPECÍFICOS**

- Identificar las distintas metodologías para la recolección de las imágenes digitales que registren fisuras en el pavimento.
- Implementar un algoritmo de segmentación de imágenes empleando las técnicas de tratamiento de imágenes digitales.
- Caracterizar las fisuras del pavimento rígido en las imágenes digitales.
- Desarrollar un algoritmo para clasificar los distintos tipos de fisuras a partir de las imágenes segmentadas.
- Validar el algoritmo desarrollado en diferentes escenarios de captura.

## ALCANCE

El presente trabajo propone un algoritmo para el reconocimiento de fallas tipo fisura en pavimentos rígidos mediante el tratamiento de imágenes digitales, que se constituya en un aporte para el desarrollo de un sistema automatizado para la clasificación y evaluación de fallas en las estructuras de pavimento de vías urbanas para Bucaramanga y su área metropolitana, y en general, para cualquier ciudad de Colombia. El algoritmo, además de identificar las fisuras en el pavimento rígido y clasificarlas entre cinco clases (borde, longitudinal, bloque, transversal y otras) según su morfología, permite cuantificar el área mínima a restaurar de la losa de concreto.

El algoritmo propuesto se enfoca a estudiar las imágenes recolectadas con la metodología de captura planteada, por ende, se establecen como limitaciones del algoritmo todas aquellas imágenes que sean capturadas por una metodología distinta a la empleada excepto que conduzca a resultados semejantes. Por ende, para futuros estudios a realizarse en el desarrollo de este algoritmo (u otros semejantes) se debe fortalecer, en primera instancia, la metodología de captura, y así dar paso a una herramienta con mayor campo de acción. En este sentido, el presente proyecto presenta, adicionalmente, una propuesta para elaborar un equipo de recolección semi-automático de imágenes digitales de fisuras en el pavimento, coherente con las capacidades tecnológicas, económicas y sociales de nuestro país.

Finalmente, es importante mencionar que los códigos desarrollados en el *software* Matlab (los cuales fueron realizados con las debidas licencias del programa) y la interfaz gráfica de usuario (UI) denominada como ALFIL, constituyen un aporte valioso para futuros trabajos en el reconocimiento de fisuras en el pavimento rígido mediante imágenes digitales. Sin embargo, dado que estos códigos se realizaron buscando optimizar el tiempo de ejecución del algoritmo, algunos algoritmos pueden resultar complejos para quienes estén comenzando a programar en Matlab, en cuyo caso se recomienda recurrir a algún manual básico de Matlab o a la ayuda del programa para una mejor comprensión de las sintaxis y la estructura de los códigos.

## JUSTIFICACIÓN

El hecho de que 31% de las vías pavimentadas en Bucaramanga presenten un grado de deterioro avanzado en su estructura de pavimento<sup>3</sup> evidencian la deficiencia de los actuales planes de gestión de pavimentos para atender por completo y oportunamente el mantenimiento de la malla vial urbana. Esta problemática ha obedecido, en parte, a que estos planes no han contado con la tecnología necesaria para evaluar el estado de las vías de forma rápida, económica y confiable, la cual a su vez, permita mejorar la planeación e inversión para la rehabilitación de la malla vial.

En este contexto, se ha propuesto el desarrollo de un sistema de clasificación y evaluación de fallas en las estructuras de pavimento de vías urbanas dentro del cual encontramos el algoritmo de reconocimiento de fallas tipo fisura en pavimento rígido mediante el tratamiento de imágenes digitales. En particular, este último se justifica a raíz de que el pavimento rígido constituye el segundo tipo de pavimento presente en las vías urbanas de Bucaramanga y las fisuras la primera causa de fallas en éstas vías<sup>4</sup>; adicionalmente, el algoritmo ofrece, principalmente, cinco beneficios con respecto al procedimiento manual que se emplea actualmente: optimización del tiempo empleado para el reconocimiento de fallas, ahorro económico a largo plazo, confiabilidad en la información recopilada, y una mayor precisión y exactitud en la identificación y análisis de las fallas tipo fisura en las estructuras de pavimento rígido<sup>5</sup>.

---

<sup>3</sup> IBID. p. 14

<sup>4</sup> IBID. p. 12-22

<sup>5</sup>Chang, Y. et al. Critical assessment of pavement distress segmentation method. En: Journal of transportation engineering. Vol. 136, no. 11; 2010.p. 1.

# 1. ESTADO DEL ARTE

El pavimento es una estructura compuesta por capas de diferentes materiales cuya función es soportar cargas de tránsito y prestar un mejor servicio para el rodamiento de los vehículos sobre las vías. Estas estructuras son susceptibles al deterioro de su resistencia y servicio por efectos del exceso de cargas tránsito, cambios abruptos de temperatura o carencias de soporte del suelo, manifestándose en forma de fallas tales como fisuras, deformaciones o desintegración de los materiales. En general, en los pavimentos cuya capa de rodadura es de concreto hidráulico – pavimentos rígidos- el principal tipo de falla que se presenta son las fisuras, entre las cuales se pueden mencionar las fisuras de borde, longitudinales, transversales, entre otras. La presencia de estas fisuras en el detrimento de las vías ha centrado la atención de los sistemas de visión por computador (emulaciones computacionales de la visión humana) en el desarrollo de algoritmos capaces de reconocer y cuantificar este tipo de fallas, empleando, principalmente, imágenes digitales y técnicas para el tratamiento de imágenes digitales. A continuación, se presenta la recopilación de los marcos teóricos y referenciales correspondientes a los tipos de fisuras, las técnicas de tratamiento de imágenes digitales y los algoritmos desarrollados para el reconocimiento, clasificación y cuantificación de fisuras en las estructuras de pavimento.

## 1.1 TIPOS DE FISURAS EN EL PAVIMENTO RÍGIDO

Como se mencionó anteriormente, el pavimento rígido es una estructura diseñada para soportar cargas y mejorar el servicio de tránsito vehicular, la cual se encuentra soportada sobre el suelo, compuesta por distintas capas de materiales y cuya capa superior es una losa de hormigón. Por otro lado, las fisuras en el pavimento rígido son grietas que se en forma de discontinuidades en la superficie del pavimento con dimensiones mínimas de 1 mm de ancho y 25 mm de largo, y que se pueden extender en profundidad por las diferentes capas del pavimento<sup>6</sup>. Estas grietas obedecen, por lo general, a excesos en las cargas de tránsito, deficiencias de soporte o cambios extremos de temperatura en el pavimento, y pueden clasificarse de acuerdo a su origen, morfología y/o grado de afectación en el pavimento, conocido como nivel de severidad<sup>7</sup>.

Es común que existan diversos tipos de clasificación de acuerdo a la agencia de infraestructura vial que se considere, sin embargo, pueden determinarse ciertos tipos de fisuras consideradas universalmente, entre las cuales se destacan:

---

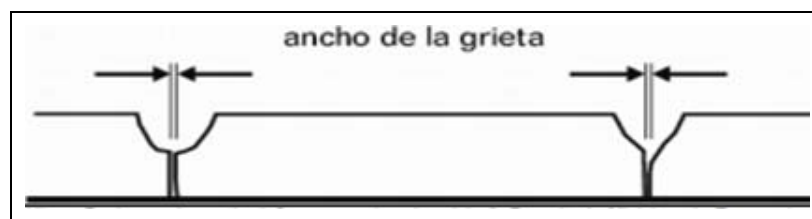
<sup>6</sup>National Cooperative Highway Research Program. Automated pavement distress collection techniques. Synthesis 334. Washington: NCHRP, 2004. p.10.

<sup>7</sup>Consejo de directores de carreteras de Iberia e Iberoamérica. Catálogo de deterioros de pavimentos rígidos. DIRCAIBEA. 2002. p. 1-2, 12-15.

### 1.1.1 Fisuras de borde

También conocidas como grietas de esquina, se definen como fisuras que “generalmente se intercepta con las juntas transversales y longitudinales, describiendo, aproximadamente, un ángulo de 45° con la dirección del tránsito”<sup>8</sup> y que generan un bloque de forma triangular en una esquina del pavimento de lados que van desde 30 cm hasta la mitad del ancho de la losa de concreto. Estas fisuras pueden ocasionarse por la falta de soporte del suelo, exceso de cargas de tránsito en las esquinas de la losa o una mala transición de cargas entre las juntas del pavimento rígido. Su nivel de severidad varía entre bajo a alto en función del ancho de la grieta – entendido como lo ilustra la Figura 1- y el agrietamiento del bloque formado.

Figura 1. Ancho de las fisuras



Fuente: Consejo de directores de carreteras de Iberia e Iberoamérica. Catálogo de deterioros de pavimentos rígidos. p. 2.

### 1.1.2 Fisura longitudinal

Las grietas longitudinales son “predominantemente paralelas al eje de la calzada o que se extienden desde una junta transversal hasta el borde de la losa, pero la intersección se produce a una distancia (L) mucho mayor que la mitad del ancho de la losa”<sup>9</sup>. Por lo general, suelen originarse por la carencia de juntas longitudinales, un ancho excesivo de la losa, asentamientos en la fundación o un mal posicionamiento de las dovelas de transferencia de cargas. Su nivel de severidad puede ser bajo (ancho de grieta menor a 3 mm) hasta alto (ancho superior a los 10 mm).

### 1.1.3 Fisura transversal

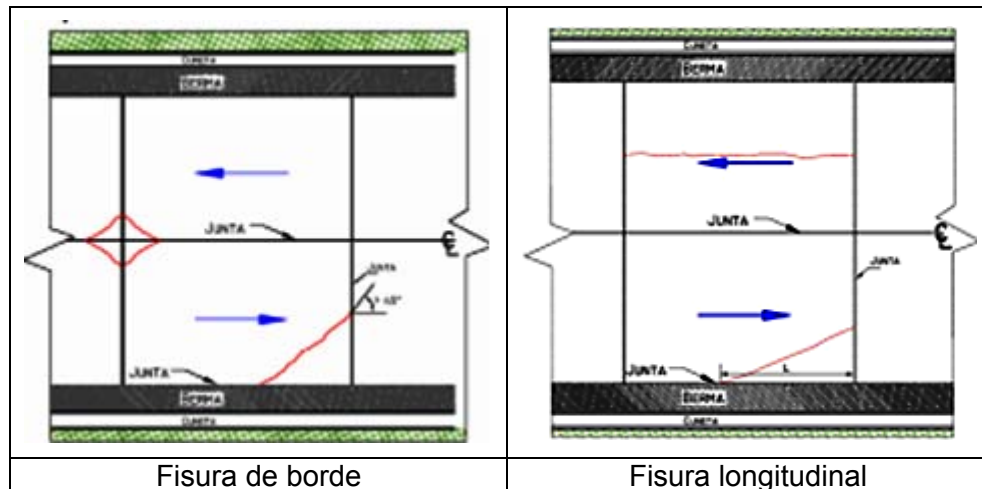
Son grietas perpendiculares al eje de la calzada que, al igual que con las fisuras transversales, pueden extenderse desde una junta transversal hasta el borde de la losa, siempre y cuando la intersección con la losa sea a una distancia mayor a la mitad del ancho de la losa. Dentro de sus causas se destacan longitudes excesivas de la losa, espesor de losa insuficiente para soportar las cargas de tránsito o alabeos en la losa. Al

<sup>8</sup>U.S. Department of Transportation, Federal Highway Administration. Distress identification manual for the long-term pavement performance program. Washington: FHWA, 2003. P. 36

<sup>9</sup>INVIAS – Universidad Nacional de Colombia. Manual de inspección visual para pavimentos rígidos. Estudio e investigación del estado actual de las obras de la red nacional de carreteras [convenio interadministrativo 0587-03], p.7.

igual que las fisuras longitudinales los niveles de severidad varían entre bajos (menores a 3 mm) y altos (mayores a 6 mm). (Ver figura 2).

Figura 2. Fisura de borde y longitudinal



Fuente: INVIAS – UNIVERSIDAD NACIONAL DE COLOMBIA. Manual de inspección visual para pavimentos rígidos. Estudio e investigación del estado actual de las obras de la red nacional de carreteras [convenio interadministrativo 0587-03], Octubre 2006. p. 8.

#### 1.1.4 Fisuras en bloque

También conocidas como fisuras de fracturación múltiple, son la combinación de fisuras longitudinales y transversales, presentando orientaciones variables, que dividen a la losa en bloques, considerándose la evolución de configuraciones de fisuras más simples. Su origen puede deberse a fatigas en el concreto o a carencias de soporte del suelo y a diferencia de las demás fisuras mencionadas anteriormente su severidad se determina según el porcentaje de losa afectado.

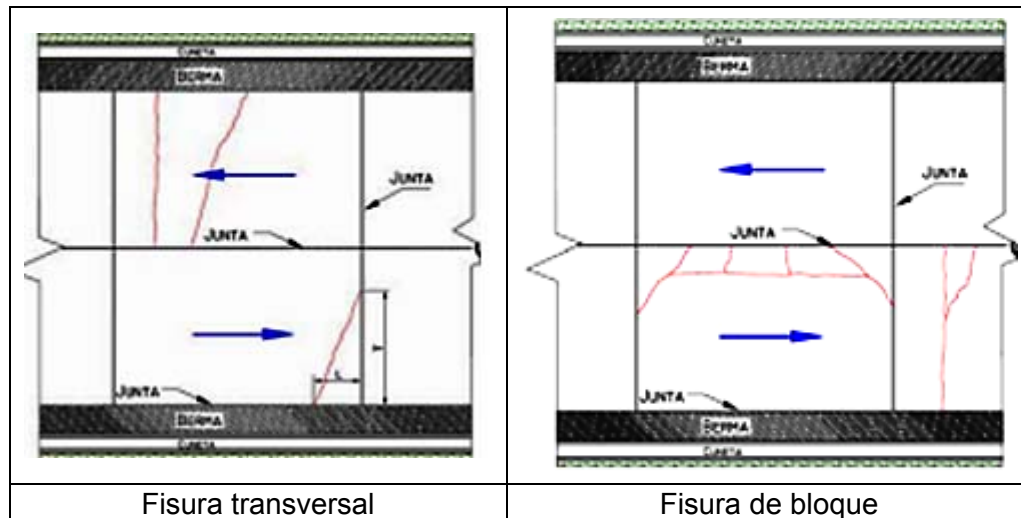
Finalmente, cabe resaltar que existen otras clasificaciones de fisuras en pavimento rígido (grietas en pozos de inspección, grietas en los extremos de los pasadores o grietas de durabilidad, entre otras) que no se consideran en este trabajo por su poca ocurrencia o falta de común acuerdo entre las distintas agencias de infraestructura vial para reconocerlas y describirlas.

## 1.2 TÉCNICAS PARA EL TRATAMIENTO DE IMÁGENES DIGITALES EN EL RECONOCIMIENTO DE FISURAS EN EL PAVIMENTO

Una imagen digital es una representación bidimensional de un objeto obtenida mediante una cámara digital, ésta última almacena en un arreglo matricial las intensidades emitidas por el objeto empleando para ello un sensor electrónico. A la unidad elemental de los

sensores electrónicos se les denomina pixel, y su cantidad en el sensor determinan el tamaño de la imagen y en gran medida la resolución o calidad de ésta. (Ver figura 3)

Figura 3. Fisura transversal y figura de bloque



Fuente: INVIAS – UNIVERSIDAD NACIONAL DE COLOMBIA. Manual de inspección visual para pavimentos rígidos. Estudio e investigación del estado actual de las obras de la red nacional de carreteras [convenio interadministrativo 0587-03], Octubre 2006. p. 12.

Por otro lado, el tratamiento de las imágenes digitales es el conjunto de procedimientos que se le aplican a una imagen digital con el ánimo de obtener cierta información del objeto capturado. En adelante, en este proyecto se entenderá por tratamiento de imágenes digitales como el conjunto de técnicas que permiten identificar, clasificar y cuantificar las fisuras en el pavimento rígido, incluyendo la captura de la imagen.

A continuación, se presenta una descripción básica, pero fundamental, para comprender los conceptos y las técnicas en el tratamiento de imágenes digitales empleados a lo largo de este trabajo.

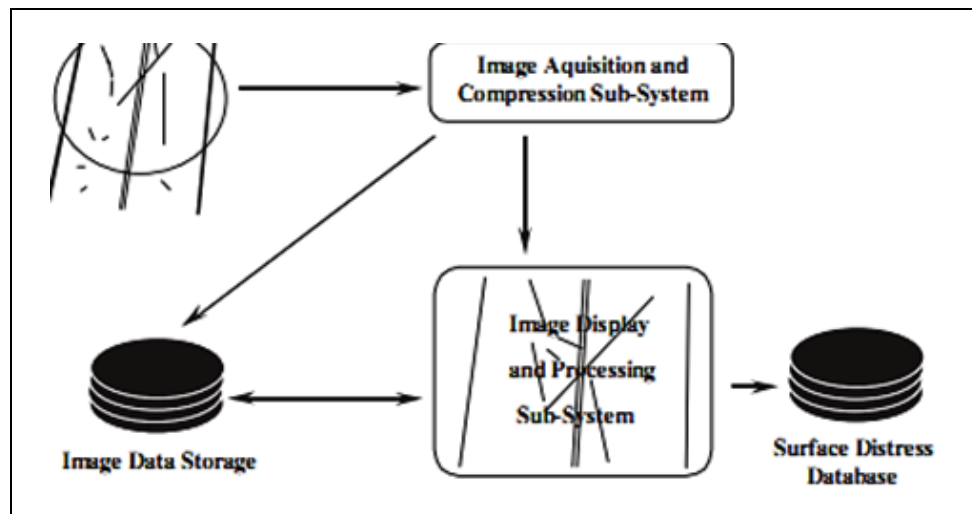
### 1.2.1 Recolección de las imágenes digitales

Históricamente, la inspección del estado del pavimento se ha realizado mediante técnicas manuales realizadas in-situ por un grupo de expertos. Sin embargo, la necesidad de recopilar más información y de forma rápida, segura y a menores costos, ha propiciado el uso de tecnologías para realizar las inspecciones, tales como las imágenes análogas y digitales, los videos y los sensores láser. En particular, para el estudio de las fisuras en la superficie del pavimento, las imágenes digitales han cobrado un auge por su simpleza, economía y efectividad, llegando a ser la en el 2004 la técnica empleada por 30% de las

agencias estatales o federales en Estados Unidos dedicadas a inspeccionar fisuras en el pavimento, contra 42% que emplean técnicas manuales<sup>10</sup>.

Las técnicas de inspección mediante imágenes digitales se han basado en equipos conformados por un vehículo que cuenta con una o dos cámaras, un GPS, un odómetro, un sistema de iluminación artificial, un dispositivo de almacenamiento y un computador para procesar las imágenes<sup>11</sup>; de acuerdo a si el procesamiento de las imágenes se realiza a medida que se van capturando las imágenes, o se hace posteriormente, en un equipo de oficina, se le denomina al sistema como automático o semi-automático, respectivamente. Adicionalmente, al sistema se le puede clasificar como de escaneo por área o línea de acuerdo a si la cámara digital empleada construye la imagen del objeto mediante barridos por área o por línea. A continuación, (ver figura 4) se presenta el esquema general de un sistema de recolección de imágenes digitales para reconocer fisuras en el pavimento, independientemente de su clasificación:

Figura 4. Esquema general de un sistema de recolección de imágenes digitales para reconocer fisuras en el pavimento



Fuente: NATIONAL. COOPERATIVE. HIGHWAY. RESEARCH.PROGRAM. Automated pavement distress collection techniques.Synthesis 334.Washington: NCHRP, 2004. p.12.

Entre los sistemas semi-automáticos desarrollados hasta la fecha se destaca el propuesto por Chamorro, A. et al (2009) en Chile, el cual consta de un vehículo equipado con una cámara digital de escaneo de área con una resolución de 1380x1030 pixeles, un sistema de iluminación híbrido entre luces halógenas y de haluros metálicos, y un instrumento de medición de distancias (DMI por su abreviatura en inglés), tal y como se ilustra en la Figura 5 para los sistemas semi-automáticos. Adicionalmente, sistemas como el *Pavement Condition Evaluation Services* (1990), el *Automated Distress Analysis for*

<sup>10</sup>NATIONAL.COOPERATIVE. HIGHWAY. RESEARCH.PROGRAM. Automated pavement distress collection techniques. Synthesis 334.Washington: NCHRP, 2004. p.11-12.

<sup>11</sup>Ibíd.Pág. 8 – 20.

*Pavement* (1995), el *D-Rate Digital Distress Rating System* (2002), el *Roadview GDPlot* (2002) y el propuesto por Miller, R. et al (2003) consisten en herramientas semi-automáticas para la inspección del pavimento, caracterizadas por emplear cámaras con resoluciones de entre 512 y 2048 pixeles para las cámaras de escaneo lineal, y de 2048x2048 pixeles para las de escaneo por área. Adicionalmente, la velocidad a la cual se puede desplazar el vehículo para los distintos sistemas varía entre los 10 y los 96 km/h, alcanzándose a reconocer fisuras de hasta 3 y 1 mm.

Por otro lado, entre los sistemas totalmente automáticos se destaca el desarrollado por la Universidad de Arkansas, denominado como *Digital Highway Data Collection* (DHDV), cuya última versión cuenta con dos cámaras de escaneo lineal, un GPS, un DMI e iluminación laser, permitiendo capturar imágenes de hasta 4.5 metros de ancho, a una velocidad de 100 km/h y con una precisión de 1 mm<sup>12</sup>. Otros sistemas que cuentan con este tipo de tecnología son: *WiseCrax* - descrito por Wang and Elliot (1999)- y *GIE GIECRK*, las cuales emplean 2 y 4 cuatro cámaras digitales, respectivamente, permitiendo capturar fisuras de hasta 3 mm, anchos de imágenes de 4 m y con una resolución de 640x480 pixeles, empleando iluminación estroboscópica.

Figura 5. Sistemas de recolección



Fuente: Wang, Kelvin, C.P. y otros. Automated imaging technique for runway condition survey. FAA Worldwide Airport Technology. p. 23

Finalmente, existen ciertos criterios externos a la tecnología empleada que se deben considerar para recolectar las imágenes digitales de la superficie del pavimento con el ánimo de obtener una imagen de buena calidad. Estas condiciones fueron estandarizadas por *Austrroads*<sup>13</sup>, quien en 2010 publicó una recopilación de las técnicas empleadas en Australia a la hora de fotografiar las fisuras en el pavimento, destacándose los siguientes aspectos:

<sup>12</sup>Wang, Kelvin, C.P. y otros. Automated imaging technique for runway condition survey.(FAA Worldwide Airport Technology. 2007. p. 21.

<sup>13</sup>Austrroads. Review of specification for the collection of Digital Road Imaging. En: Austrroads Technical Report no. AP-T169-10. Sydney: Austrroads, 2010. p.34.

- La imagen debe ser tomada perpendicular a la superficie del pavimento.
- Si no se emplea iluminación artificial, la hora de captura debe ser tal que el ángulo de elevación del sol sea inferior a los 40° con respecto al horizonte, con el fin de reducir el efecto denominado *sun glare*.
- La superficie del pavimento debe estar preferiblemente seca para evitar reflejos de luz producidos cuando el pavimento se encuentra húmedo.

## 1.2.2 Pre-procesamiento de la imagen digital

El objeto principal del procesamiento de las imágenes digitales de fisuras en el pavimento es poder distinguir dichas fisuras y separarlas del resto de los elementos presentes en la imagen, para posteriormente caracterizarlas, clasificarlas y analizarlas. Sin embargo, en ocasiones, no es posible realizar este procedimiento de forma sencilla por deficiencias en la iluminación o por la simple presencia de ruido en la imagen, por ende, se recurre a ciertas técnicas denominadas de pre-procesamiento, cuyo fin es mejorar el contraste de la imagen y reducir el ruido presente en la misma, y así facilitar el posterior procesamiento de las fisuras. En seguida, se presentan los tipos de ruidos presentes en las imágenes digitales de fisuras en el pavimento, las técnicas de filtrado que permiten atenuar dicho ruido, así como las técnicas para incrementar el contraste en la imagen.

### 1.2.2.1 Tipos de ruidos

El ruido en una imagen digital es una variación en la intensidad o el color de las imágenes digitales generado durante la captura del objeto. Existen tres tipos de ruidos a saber: el ruido impulsional o de “sal y pimienta”, el ruido gaussiano y el ruido de disparo, entre otros. De los distintos tipos de ruido existentes el más común en las imágenes digitales de fisuras en el pavimento es el ruido de “sal y pimienta”, el cual se define como una variación en la intensidad en ciertos píxeles, presentando valores más claros u oscuros que su entorno, considerado como un ruido de índole aleatoria, originado, principalmente, por variaciones en la sensibilidad de detección del sensor electrónico.

### 1.2.2.2 Técnicas de filtrado espacial

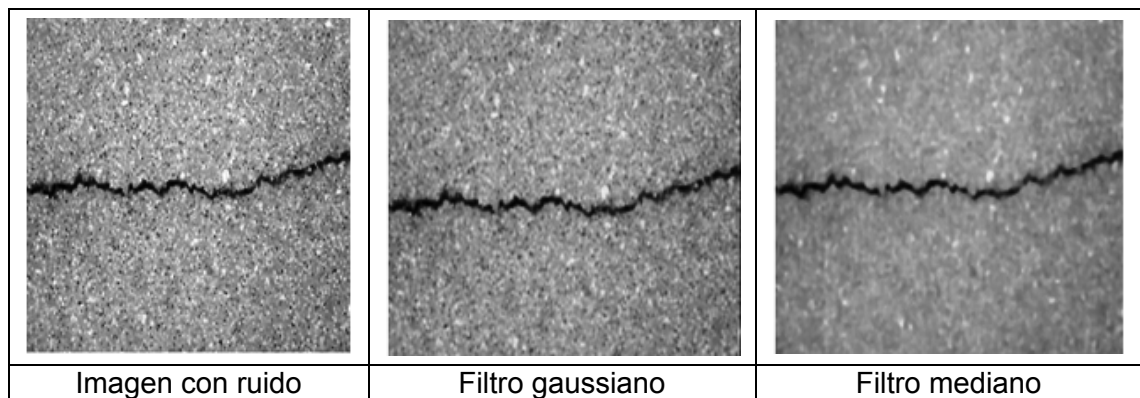
El filtrado es el procedimiento en el cual se reduce la cantidad de ruido presente en la imagen y se suavizan los contornos de los objetos. Este procedimiento puede realizarse en dos dominios de la imagen: el frecuencial o el espacial; en este trabajo solo se estudiara este último caso, adicionalmente, solo se consideraran las técnicas para imágenes en escala de grises en un rango de 0 a 255 dado que la mayoría de sensores electrónicos emplean 8 bits para el almacenamiento de las intensidades en cada píxel.

El filtrado espacial en una imagen digital en escala de grises consiste en reemplazar la intensidad registrada en cada píxel por el resultado de una operación estadística sobre sí mismo y un conjunto de píxeles vecinos - denominados como vecindad o máscara-, las operaciones a realizar pueden catalogarse como de promedio o de rango, mientras que las máscaras suelen ser cuadradas y de dimensiones impares (3x3, 5x5, 9x9, etc.).

Dentro de los filtros que emplean operadores de promedio se encuentran los filtros medios (se promedian las intensidades en la máscara y se asigna éste valor al pixel central) y los filtros con operadores de “kernel”, los cuales son una generalización del filtro medio al reemplazar la intensidad de cada pixel por la obtenida al promediar las intensidades dentro de la vecindad, considerando distintos “pesos” a cada pixel vecino. Las distintas configuraciones de estos pesos permiten suavizar las imágenes, eliminar los efectos conocidos como *blurring* reducir el ruido en la imagen, sin embargo, también ocasiona reducciones en el contraste de la imagen y pérdida de detalle en los bordes de los objetos.

Por otro lado, los filtros que emplean operadores de rango reemplazan la intensidad de cada pixel por el resultado de una función no lineal aplicada en la vecindad del pixel. Dentro de estos filtros el más común es el filtro mediano, el cual ordena las intensidades registradas en la vecindad de cada pixel y toma el valor mediano para asociarlo al pixel central de la máscara. Este filtro permite eliminar el ruido aleatorio en las imágenes y conserva la nitidez en los bordes de los objetos. Otros operadores existentes son el filtro de moda y el filtro mediano híbrido<sup>14</sup>. La Figura 6 presenta el uso de un filtro gaussiano (un filtro promedio donde los pesos de la vecindad siguen una distribución normal) y un filtro mediano para una imagen digital de una fisura en el pavimento.

Figura 6. Filtro lineal y no-lineal en una imagen digital



Fuente: Ouyang, Aiguo. Luo, Chagen, Zhou, Chao. Surface distress detection of pavement based on digital image processing. En: Computer and computing technologies in agriculture IV. Nanchang: IFIP, 2010. p. 368-374.

### 1.2.2.3 Mejoramiento del contraste en la imagen

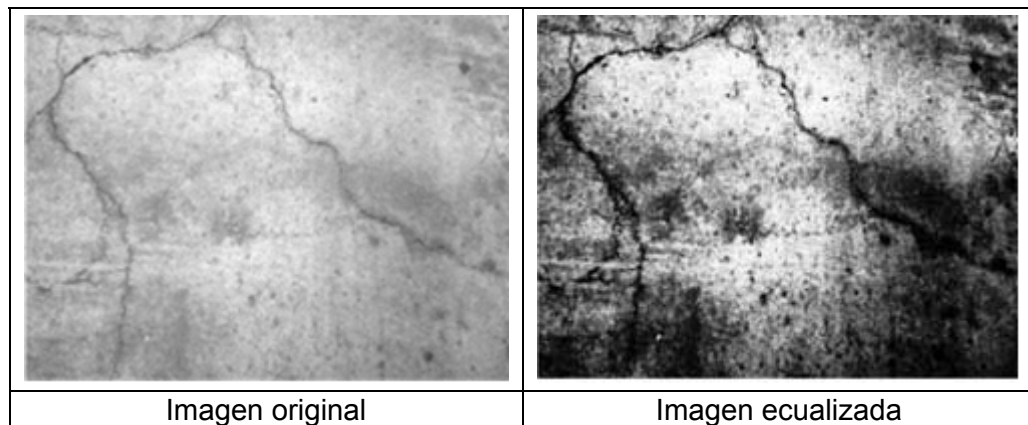
Después de remover el ruido y suavizar los contornos de los objetos, en ocasiones, cuando hay poca homogeneidad en la iluminación, es necesario emplear técnicas que permitan realzar el contraste de la imagen digital y así facilitar la extracción de los objetos en la segmentación.

<sup>14</sup>Russ, J. The image processing handbook. 5ª Edición. Broken Sound Parkway: CRC, 2008. 832p.

La primera familia de técnicas para mejorar el contraste en la imagen es la manipulación del histograma de intensidades de la imagen, para ello, se le asigna a cada pixel una nueva intensidad, producto de aplicar una función matemática que transforma el histograma de la imagen. (Ver figura 7) La principal técnica dentro de esta familia de técnicas es la ecualización del histograma, la cual modifica el histograma de una imagen empleando como función de transformación la propia función acumulada del histograma, de forma tal que en la imagen obtenida todos los posibles valores en la escala de grises (0-255) estén distribuidos de forma uniforme. La ecualización del histograma, pese a incrementar notablemente el contraste de la imagen y resaltar el borde de los objetos, en ocasiones conlleva a pérdida de información en la imagen que podría resultar valiosa en el proceso de segmentación. (Ver figura 7).

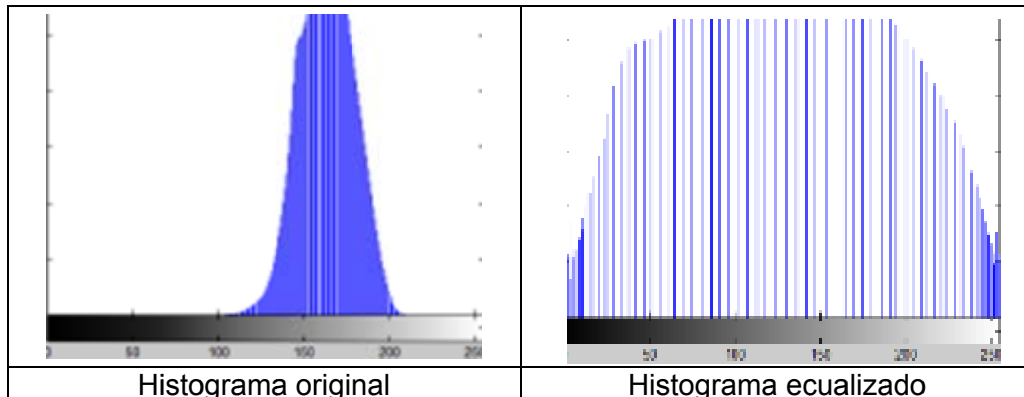
Aparte de las técnicas para la manipulación del histograma, existen técnicas enfocadas exclusivamente a resaltar los bordes de los objetos en una imagen digital empleando operadores de “kernel” cuya suma de “pesos” sea cero, para posteriormente, aproximar la derivada de la intensidad en las distintas direcciones de la máscara (horizontal, vertical, diagonales) y asignarle dichas derivadas al pixel central, sin embargo, estas técnicas también resaltan el ruido en la imagen. (Ver figura 8). Los distintos operadores de “kernel” empleados, así como la función que se emplee para aproximar la derivada de la intensidad, definen los diferentes operadores, entre los cuales cabe resaltar: el operador de Sobel, el de Frei y Chen, el de varianza, el de Prewitt y el de Canny<sup>15</sup>.

Figura 7. Técnica de ecualización del histograma



---

<sup>15</sup>IBID. p.281-336.



Fuente: Los autores

Figura 8. Uso de operadores de detección de bordes



Fuente: Ouyang, Aiguo. Luo, Chagen, Zhou, Chao. Surface distress detection of pavement based on digital image processing. En: Computer and computing technologies in agriculture IV. Nanchang: IFIP, 2010. p. 368-374.

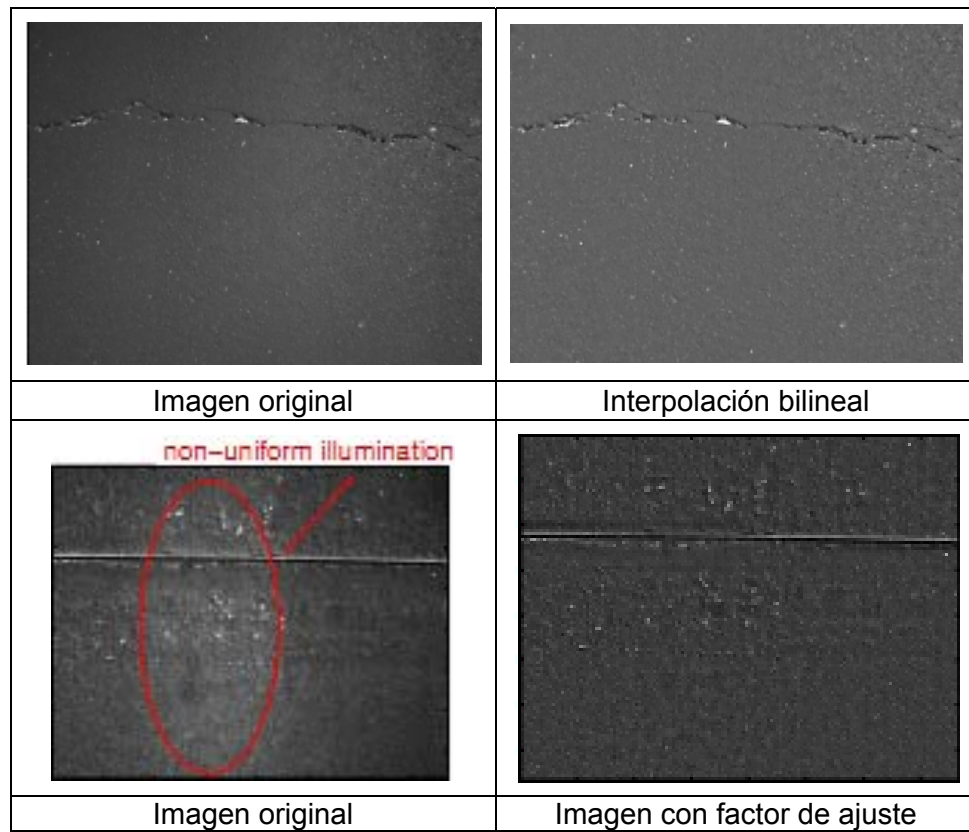
Por último, existe otra familia de técnicas de mejoramiento del contraste que se basan en la premisa de que toda imagen digital se componen de tres partes: el ruido, el fondo y los objetos a analizar, por ende, basta con homogeneizar el fondo para facilitar la segmentación de los objetos, los cuales comúnmente tienen intensidades muy inferiores o muy superiores a la intensidad promedio del fondo. La homogenización del fondo consiste en llevar a un valor único (o a un pequeño rango en la escala de grises) las intensidades de todos los pixeles pertenecientes al fondo de la imagen. (Ver figura 9).

El procedimiento consiste en dividir la imagen en recuadros de menor tamaño, de forma tal que en cada uno de ellos sea posible estimar una intensidad homogénea del fondo como el promedio de las intensidades en el recuadro. Sin embargo, dado que en algunos recuadros persistirá el ruido y objetos distintos al fondo, es necesario establecer un rango en la escala de grises para cada recuadro, en el cual se crea que tan solo están los pixeles del fondo, y dentro de éste rango estimar la intensidad homogénea del fondo del recuadro.

Hasta este punto las principales técnicas pertenecientes a esta familia de técnicas convergen en el procedimiento a seguir, sin embargo, las diferencias se presentan de allí en adelante. Por un lado, existen técnicas que emplean interpolaciones (lineales o

bilineales) para que, con base en los promedios de cada recuadro, se determine una función que le asigne a cada pixel una nueva intensidad congruente con la de una imagen homogénea.<sup>16</sup> Por otro lado, otras técnicas simplemente definen para cada recuadro un factor de ajuste definido como la relación entre el promedio del fondo del recuadro y la intensidad a la cual se quiera homogeneizar el fondo, para posteriormente, amplificar cada pixel de la imagen por el factor de su recuadro correspondiente<sup>17</sup>.

Figura 9. Técnicas de homogenización del fondo de la imagen



Fuente: Zhou, Huilin, Yang, Zhu y ZhuJian. Illumination invariant enhancement and threshold segmentation algorithm for asphalt pavement crack image. En: IEEE (2010); p.3.

<sup>16</sup>Jing, Lou y Aiqin, Zang. Pavement crack distress detection based on image analysis (2010: International Conference on Machine Vision and Human Machine Interface); p. 576-577.

<sup>17</sup>Zhou, Huilin, Yang, Zhu y ZhuJian. Illumination invariant enhancement and threshold segmentation algorithm for asphalt pavement crack image. En: IEEE (2010); p 1-2.

### 1.2.3 Segmentación

La segmentación “es el procedimiento fundamental en el tratamiento de las imágenes digitales de fisuras en el pavimento”<sup>18</sup>, consistiendo en la separación de las fisuras del fondo (pavimento en asfalto o concreto) y dando como resultado una imagen en blanco y negro con valor de uno para los píxeles pertenecientes a la fisura y de cero para los correspondientes al fondo. Existen diversas técnicas y familias de técnicas empleadas para cumplir con este objetivo, sin embargo, a continuación se describen las que más se encuentran en la literatura del procesamiento de imágenes digitales de fisuras en el pavimento.

#### 1.2.3.1 Segmentación por umbral

Las técnicas de segmentación por umbral simplemente definen un valor en la escala de grises que representa el límite entre lo que se considera la fisura y el fondo del pavimento, conocido como umbral. En una segmentación por umbral todos los píxeles con intensidades en la escala de grises inferiores al umbral harán parte del fondo en la imagen binaria, mientras que los píxeles con intensidades superiores al umbral conformarán las fisuras.

Existen distintos procedimientos para determinar el umbral óptimo que permita separar los objetos del fondo, pudiendo mencionarse el método por inspección o de prueba y error, los métodos de regresión, el método iterativo, el de máxima entropía y el método de Otsu.

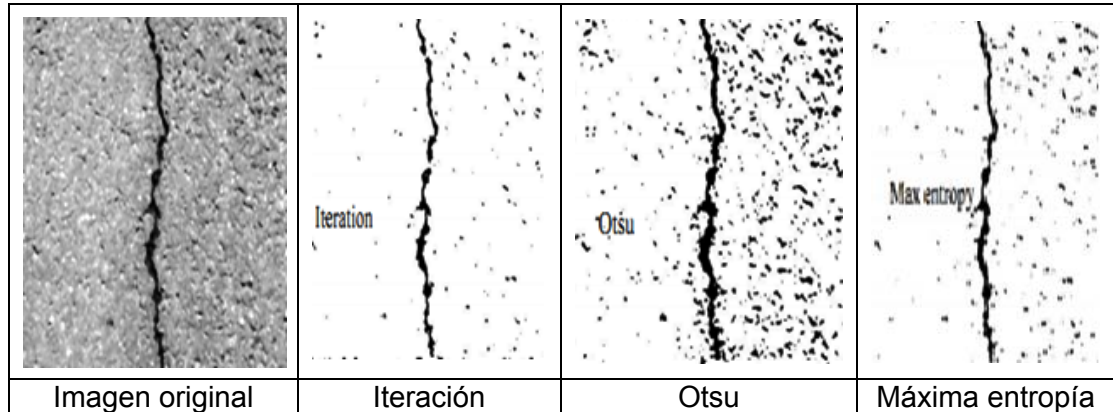
El método de regresión básicamente establece el valor del umbral con base a una ecuación expresada en función del promedio y la desviación de las intensidades en la imagen. Por su parte, el método iterativo define, en primera instancia, como el umbral al valor promedio del histograma de la imagen, generando así dos regiones: una de intensidades superiores y otra de intensidades inferiores. Posteriormente, para cada región se determina el nivel de gris promedio y se define al promedio de estos dos valores como el nuevo valor del umbral, repitiendo el proceso hasta que el cambio en el umbral en dos iteraciones sucesivas sea inferior a una tolerancia establecida.

De forma similar operan las técnicas de máxima entropía y Otsu, sin embargo, en la primera se itera hasta que el umbral encontrado maximice la entropía en la imagen, mientras que en la última la iteración finaliza cuando se estima un umbral que minimice la varianza entre las dos clases generadas en la imagen binaria (las fisuras y el fondo). Pese a la sencillez de estos algoritmos, su comportamiento no es el mejor en imágenes que presenten un histograma unimodal, como es el caso de las imágenes de fisuras en el pavimento (ver figura 10).

---

<sup>18</sup>Tsai, Yi Chang, Kaul, Vivek y Mersereau, Russell M. Critical assessment of pavement distress segmentation methods. En: Journal of Transportation Engineering ASCE. No. 1, 2010.p. 1.

Figura 10. Métodos de segmentación por umbral



Fuente: Wang, Kelvin, C.P. y otros. Automated imaging technique for runway condition survey.FAA WorldwideAirportTechnology.p. 2.

### 1.2.3.2 Método de relajación o segmentación estadística

El método de relajación o método estadístico le asigna a cada pixel de la imagen una probabilidad de pertenecer o no al fondo, y calcula para cada pixel la probabilidad promedio en su vecindario de pertenecer o no al fondo. Con base en estas probabilidades se calcula un indicador de la compatibilidad entre la posibilidad de que los pixeles y sus vecinos pertenezcan o no al fondo. Posteriormente, se re-calculan las probabilidades y el proceso se repite de forma heurística hasta que se maximice la compatibilidad<sup>19</sup>.

### 1.2.3.3 Métodos de detección de bordes

Al igual que como se describió en las técnicas de pre-procesamiento, los operadores de Canny, Sobel, Prewitt entre otros, pueden ser empleados para segmentar las fisuras en una imagen digital siguiendo el mismo concepto de calcular la varianza o derivada en la vecindad de cada pixel y para múltiples direcciones. Sin embargo, la segmentación mediante estas técnicas adiciona un umbral para las variaciones obtenidas, determinando como fisura a todos aquellos pixeles que tengan una variación superior al valor establecido y como fondo a aquellos con valores inferiores. Entre los distintos operadores disponibles, el que presenta mejores resultados para fisuras en el pavimento ha sido el de Canny, toda vez que permite definir un umbral para las fisuras fuertes y otro para las fisuras débiles (de menor ancho)<sup>20</sup>.

<sup>19</sup>Koutsopoulos, Haris N., El Sanhoury, Ibrahim y Downey, Allen B. Analysis of segmentation algorithms for pavement distress images. ASCE, 2011. p. 872-877.

<sup>20</sup>Tsai, Yi Chang, Kaul, Vivek y Mersereau, Russell M. Critical assessment of pavement distress segmentation methods. En: Journal of Transportation Engineering ASCE. No. 1, 2010. p. 13.

#### 1.2.3.4 Método de recorte sucesivo (*iterativeclipping*)

A diferencia de las técnicas descritas hasta ahora, este método primero divide la imagen en recuadros de menor tamaño y para cada uno de ellos calcula el valor promedio. Posteriormente, para cada recuadro se recorta el histograma asignándole a todos los píxeles con intensidades superiores al valor promedio dicho valor. Con los nuevos valores se determina una nueva media ( $\mu_i$ ) y una desviación ( $\sigma_i$ ) y se define un nuevo valor de recorte ( $C_{i+1}$ ). El proceso se repite hasta que la diferencia entre el valor de recorte ( $C_{i+1}$ ) y el promedio del recuadro tras el recorte ( $\mu_i$ ) sea inferior a una tolerancia, que por lo general suele tomarse de 0.3. Una vez se cumpla la condición anteriormente enunciada, la imagen se puede segmentar fácilmente seleccionando un umbral.

$$C_i = \mu_{i-1} - 1.2 * \sigma_{i-1}$$

La principal ventaja de este algoritmo para la segmentación de fisuras en imágenes digitales es que permite subdividir el procedimiento de segmentación en imágenes más pequeñas, segmentando fisuras más finas con respecto a otros métodos. Sin embargo, introduce mucho ruido a la imagen en recuadros donde no hay fisuras.

#### 1.2.3.5 Métodos de segmentación por regiones

Otra familia de técnicas disponibles permite segmentar las fisuras en las imágenes digitales mediante la construcción de regiones, entendiéndose por regiones como aquel conjunto de píxeles que se pueden conectar sin interrupción. Las conexiones que permiten crear regiones pueden ser de dos tipos: de cuatro conexiones (solo se conectan píxeles vecinos en las direcciones verticales y horizontales) o de ocho conexiones (también se permite conectar píxeles vecinos en la dirección de las diagonales).

La técnica más empleada dentro de esta familia de métodos es la denominada como crecimiento por regiones. Este procedimiento parte de establecer unos píxeles semillas que corresponden a píxeles que por su intensidad seguramente hacen parte del objeto a segmentar, posteriormente, a partir de cada una de estas semillas se construyen las regiones conectando tan solo aquellos píxeles que se encuentren dentro de un rango de proximidad al valor semilla, conocido como la tolerancia del método<sup>21</sup>.

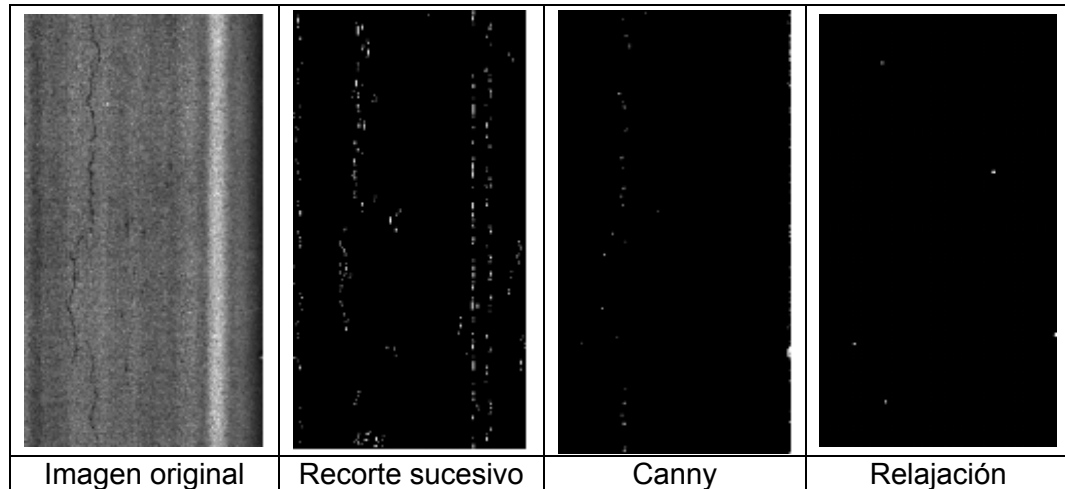
Por otro lado, la técnica de división y unión de regiones parte de la premisa de haber creado las regiones por un método como el de crecimiento por regiones para posteriormente dividir cada región en cuatro regiones, a cada división se le realiza una prueba lógica para determinar si en esa subregión se cumple o no con una condición estipulada de acuerdo a la naturaleza del problema, si dicha condición se cumple la subregión no se divide más, de lo contrario se divide en cuatro nuevas regiones hasta que

---

<sup>21</sup>Gonzalez, Rafael C., Woods, Richard E. y Eddins, Steven L. Digital Image Processing using Matlab. 2 ed. New York: Addison – Wiley, 2009. p.407-412.

se cumpla el criterio. Una vez divididas todas las regiones se procede a unir aquellas que tengan características semejantes (ver figura 11).

Figura 11. Otras técnicas de segmentación



Fuente: Chang, Y. et al. Critical assessment of pavement distress segmentation method. En: Journal of transportation engineering. Vol. 136, no. 11; 2010. p.4

Finalmente, cabe mencionar que existe otro sin fin de técnicas para la segmentación de fisuras en imágenes digitales en el pavimento que no se contemplaron tales como la segmentación por transformación del histograma, segmentación por texturas, segmentación por lógica difusa o redes neuronales, segmentación por optimización dinámica y segmentación en el dominio de la frecuencia de la imagen.

#### 1.2.4 Operaciones morfológicas

Las operaciones morfológicas son técnicas que modifican la forma de los objetos en la imagen a partir de cambios en las intensidades de los píxeles en una vecindad. Estas técnicas pueden ser empleadas tanto en imágenes en escala de grises como en blanco y negro y para el pre-procesamiento, la segmentación o un pos-procesamiento tras segmentar la imagen. En este proyecto se presentarán las operaciones morfológicas como técnicas para el pos-procesamiento en las imágenes en blanco y negro.

##### 1.2.4.1 Operación erosión y dilatación

La erosión y la dilatación son las técnicas fundamentales dentro de las operaciones morfológicas, ambas emplean una vecindad – o estructura- cuyo centro recorre toda la imagen digital.

En la dilatación, cuando en la vecindad se encuentra al menos un píxel con intensidad 1 (blanco) todos los demás píxeles dentro de la estructura toman un valor de 1. Por otro lado, la erosión consiste en que si todos los píxeles dentro de la estructura tienen



#### 1.2.4.3 Operaciones morfológicas *tophat* y *bothat*

Las operaciones de *tophat* y *bothat* consisten en restar a la imagen original el resultado de aplicarle, respectivamente, una operación morfológica de apertura y de cierre.

### 1.2.5 Caracterización de las fisuras en el pavimento

El resultado de la segmentación, independientemente de la técnica empleada, es un conjunto de objetos separados del fondo de la imagen. Posterior a ello, generalmente, se representan dichos objetos y se establecen patrones que permitan. Este reconocimiento de patrones o caracterización es el paso fundamental para la clasificación de los objetos y se espera que los patrones o descriptores encontrados sean simples, precisos, con gran poder de discriminación, y en el caso de las fisuras, invariantes a la posición y al escalado de ésta. A continuación, se presentan las formas de representar y caracterizar los objetos segmentados.

#### 1.2.5.1 Representación de las fisuras en las imágenes digitales

Un objeto segmentado puede representarse de dos formas: por regiones o por el borde, o frontera, de una región. Una región se define como un conjunto de píxeles conectados (ya sea en cuatro u ocho direcciones), mientras que el borde de una región es un conjunto de píxeles que pertenecen a una región pero en cuyas vecindades existe al menos un píxel que pertenece al fondo.

Para representar una frontera de una región existen diversas técnicas que eliminan los puntos internos de la región (les asigna un valor nulo), conservando únicamente aquellos pertenecientes al borde; el código de la cadena, la aproximación por polígonos y el esqueleto son tres de las técnicas más comunes.

El código de la cadena reduce al objeto a una secuencia de líneas conectadas que representan la frontera del objeto, con base en la conectividad en cuatro u ocho direcciones. La frontera estimada se convierte en un código al asignarle a cada segmento de línea un número correspondiente a la dirección del segmento, tal y como se establece en la Figura 14.

Dado que el código de la cadena es función del píxel de inicio y el sentido en el que se recorre la frontera, es necesario normalizar la representación independientemente del punto de inicio y el sentido en que se recorra (ver figura 15). Para ello, se calcula el código de la cadena a partir de todos los puntos de la frontera y se selecciona aquel cuya suma sea mínima<sup>22</sup>.

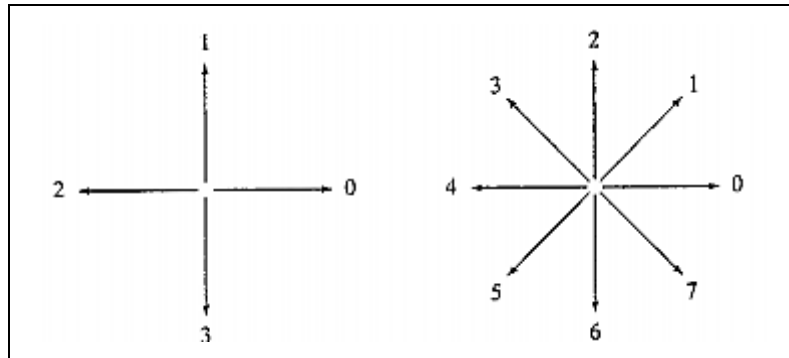
Por otro lado, la aproximación por polígonos determina el polígono de menores vértices que permita representar fielmente la frontera del objeto, conocido como mínimo polígono

---

<sup>22</sup> Gonzalez. Op cit. p. 436-439.

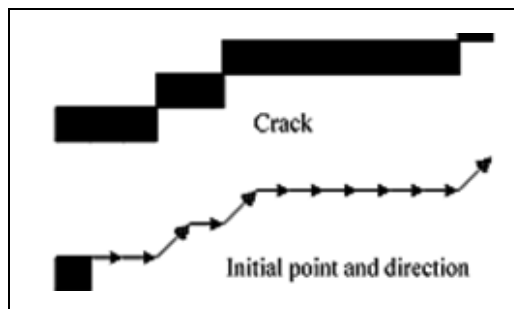
perimetral, el cual se obtiene a partir de unir los puntos que presenten mayor curvatura en el objeto. Asimismo, la aproximación poligonal puede obtenerse mediante división recursiva, la cual consiste en unir con segmentos los puntos más alejados del polígono al borde del objeto, creando nuevos segmentos hasta que la diferencia se minimice.

Figura 14. Códigos empleados en la representación por el código de la cadena



Fuente: Gonzalez, Rafael C., Woods, Richard E. y Eddins, Steven L. Digital Image Processing using Matlab. 2 ed. New York: Addison – Wiley, 2009. p. 345.

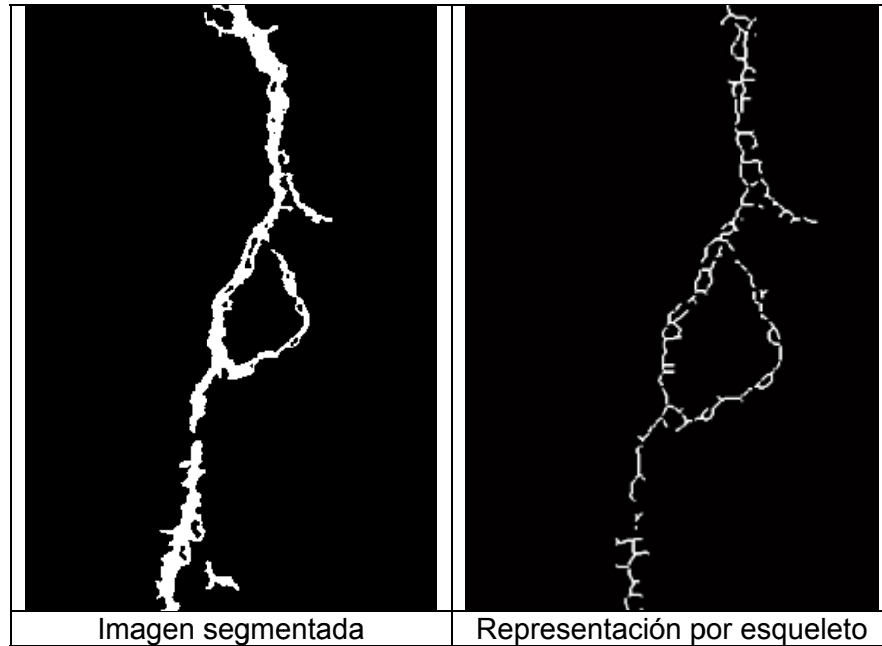
Figura 15. Representación por el código de la cadena



Fuente: Wang, Kelvin, C.P. y otros. Automated imaging technique for runway condition survey.(FAA Worldwide Airport Technology Transfer Conference: Apr. 2007: Atlantic City, NJ). p.596

Finalmente, la técnica del esqueleto reduce un objeto a la mínima cantidad de píxeles que represente su estructura reduciendo el grosor del objeto hasta encontrar su “esqueleto”; este proceso se logra mediante la transformación del eje medio, la cual consiste en determinar los píxeles que se encuentran equidistantes a dos puntos del borde de la región. Esta transformada permite tener una gran representación de las fisuras, sin embargo, computacionalmente es una técnica que demanda muchos recursos y tiempo de procesamiento (ver figura 16).

Figura 16. Representación del esqueleto de una fisura en las imágenes digitales



Fuente: Los autores

### 1.2.5.2 Descriptores de Borde

Una vez se ha representado un objeto como una frontera de una región es posible extraer ciertas características básicas de la forma de dicha frontera, independientemente del tipo de representación empleado, tales como:

- Longitud: número de conexiones entre píxeles en la frontera, considerado que las conexiones verticales y horizontales aportan una unidad de longitud mientras que las diagonales  $\sqrt{2}$  unidades de longitud.
- Diámetro: es denominado como la distancia euclidiana entre el par de píxeles más alejados entre sí.
- Eje mayor: Es el segmento de línea que une el par de píxeles más lejanos de una frontera. Su dirección define la orientación de la frontera.
- Eje menor: Es el segmento de línea perpendicular al eje mayor y de longitud igual a la máxima separación entre dos píxeles en la orientación del eje menor.
- Rectángulo base: Es el menor rectángulo que contiene toda la frontera, teniendo la misma orientación que la del eje mayor.
- Área: es el número total de píxeles en la región que están dentro de la frontera.
- Redondez: relación entre el área de la región y el cuadrado de su perímetro:  $\frac{A}{P^2}$ .
- Elongación: Relación entre el eje mayor (M) y el eje menor (N):  $\frac{M}{N}$
- Curvatura: curvatura media y varianza de curvatura entre un conjunto de puntos de la frontera.

Adicional a estos descriptores básicos de la forma, existe otro descriptor conocido como el número de forma. El número de forma se obtiene de la representación del menor código de la cadena, estableciendo una secuencia de números iguales al cambio de dirección entre cada dos segmentos sucesivos, en conexiones de ocho direcciones cada dígito entre dos segmentos sería: uno (1) para un cambio de 0° a 45°, dos (2) para un cambio de 0° a 90° y así sucesivamente<sup>23</sup>.

#### 1.2.5.3 Representación de fisuras por regiones

Como se mencionó anteriormente, este tipo de representación considera a cada objeto como un conjunto de píxeles conectados siguiendo conexiones en cuatro u ocho direcciones.

#### 1.2.5.4 Descriptores de regiones

Cuando un objeto se ha representado mediante regiones existen diversos descriptores de la región que permiten caracterizarla, tales como de la forma, de la textura y del color. Cabe anotar que el descriptor de textura trabaja sobre la imagen de intensidades, mientras que los de forma lo hacen para la imagen binaria; en este trabajo no se estudiarán los descriptores de colores.

##### 1.2.5.4.1 Descriptores de forma

Dentro de los descriptores de forma de la región se encuentran:

- Área: número total de píxeles en la región.
- Centroides: coordenadas de los píxeles que representan el centro de masa de la región.
- Rectángulo delimitador: mínimo rectángulo que encierra a toda la región en la dirección de los ejes cartesianos (Caja de Fermat) o en la dirección principal de la región (mínimo rectángulo envolvente).
- El número de Euler: diferencia entre el número de regiones conectadas menos el número de huecos.
- Extensión: proporción entre el número de píxeles pertenecientes a la región y al número total de píxeles contenidos en el rectángulo delimitador.
- Solidez: proporción de píxeles que pertenecen a la región y que pertenecen a la menor figura convexa que contiene a la región.

Existen otros descriptores de forma que requieren definir previamente el siguiente concepto: se define como la elipse con igual segundo momento angular que la región a aquella elipse que representa el contorno de la función de densidad de distribución normal multivariable que se ajusta la distribución de los píxeles en la región. La dirección y la longitud de los ejes de dicha elipse se obtienen de los vectores propios de la matriz de

---

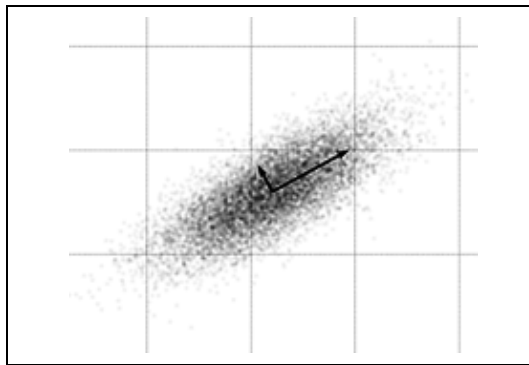
<sup>23</sup> Gonzalez. Op cit. p. 455-463.

covarianza de la región. A partir de este concepto se definen los siguientes descriptores de forma de la región:

- Longitud del eje mayor: es la distancia del eje mayor de la elipse con igual matriz de covarianza que la región.
- Longitud del eje menor: distancia del eje menor de la elipse con la misma matriz de covarianza que la región.
- Orientación: dirección del eje mayor de la elipse con la misma matriz de covarianza que la región, con respecto a la horizontal.
- Excentricidad: Excentricidad de la elipse con igual matriz de covarianza que la región, valores cercanos a uno (1) representan una elipse próxima a una línea. (ver figura 17)

Finalmente, al igual que en los descriptores de frontera es posible definir para las representaciones de regiones los descriptores de redondez y elongación.

Figura 17. Elipse con igual segundo momento central que la región



Fuente: Gaussianscatter. [en línea] disponible en <http://en.wikipedia.org/wiki/File:GaussianScatterPCA.png>. Consultado el 23 de agosto de 2011.

#### 1.2.5.4.2 Descriptores de textura

La textura de una imagen se define como la distribución espacial de los tonos de grises en la imagen<sup>24</sup>, de allí que los descriptores de textura midan propiedades como la regularidad, homogeneidad, suavidad y varianza de las intensidades de grises en el dominio espacial de la imagen. Los descriptores de textura se basan en la construcción de la matriz de co-ocurrencia de niveles de grises (GLCM por su abreviatura en inglés), la cual es una matriz de probabilidades de que co-ocurrán dos intensidades en un par de píxeles vecinos para una imagen dada, Haralick et al (1973) propuso que dicha matriz de co-ocurrencia debía construirse estudiando pares de píxeles en distintas direcciones y a

---

<sup>24</sup>Haralick, Robert M., Shanmugam, K. y Disnstein, ItsHak. Textural features for image classification. En: IEEE Transactions on Systems Man and Cybernetics. Nov. 1973. SMC 3, no. 6; p. 610-613.

diferentes distancias, por lo que comúnmente se definen cuatro matrices de co-ocurrencia para los vecinos más próximos (distancia uno): una matriz para vecinos en dirección horizontal (0°), otra para vecinos en dirección vertical (90°) y otras dos para vecinos en diagonal (45° y 135°) (ver figura 18).

Para cada una de las GLCM calculadas es posible establecer catorce (14) descriptores de textura, que en el caso de considerarse las cuatro matrices se podrían convertir en veintiocho (28) descriptores si se consideran el rango y el valor promedio de los valores obtenidos en cada dirección para cada descriptor<sup>25</sup>. Dentro de los trece descriptores propuestos por Haralick et al (1973), los que se emplean comúnmente son la varianza, la correlación, el contraste y el segundo momento angular (ver figura 18).

Figura 18. Principales descriptores de textura

<p>1) Segundo momento angular:</p> $f_1 = \sum_i \sum_j \{p(i, j)\}^2$ <p>2) Contraste</p> $f_2 = \sum_{n=0}^{N_g-1} n^2 \cdot \left\{ \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i, j) \right\}_{ i-j =n}$	<p>3) Correlación:</p> $f_3 = \frac{\sum_i \sum_j (i \cdot j) \cdot p(i, j) - \mu_x \cdot \mu_y}{\sigma_x \cdot \sigma_y}$ <p>4) Varianza</p> $f_4 = \sum_i \sum_j (i - \mu)^2 \cdot p(i, j)$
--	---

Fuente: Haralick, Robert M., Shanmugam, K. y Disnstein, ItsHak. Textural features for image classification. En: IEEE Transactions on Systems Man and Cybernetics. Nov. 1973. SMC 3, no. 6; p. 618-619.

### 1.2.6 Algoritmos de clasificación

La clasificación de los objetos con base en los descriptores seleccionados para caracterizar las representaciones de los objetos, es el último paso a realizar en el tratamiento de las imágenes digitales para el reconocimiento de fisuras. El objetivo de los algoritmos de clasificaciones poder reconocer patrones que permitan clasificar dentro de una clase o categoría a los objetos segmentados (en las fisuras se podría pensar en categorías como las de borde, longitudinal, transversal, etc.), de allí, que a éstos algoritmos se les denomine como algoritmos para el reconocimiento de patrones. En esencia, todos los algoritmos de reconocimiento de patrones definen una línea (o un plano o un hiperplano) de decisión que permite clasificar un patrón en una clase u otra<sup>26</sup>. Adicionalmente, estos algoritmos pueden clasificarse como supervisados, no

<sup>25</sup> Ibíd. Pág. 612-618.

<sup>26</sup>Theodoridis, Sergios y Koutroumbas, Konstantinos. Pattern Recognition. 4 ed. London: Academic Press, 2009. p.4-9.

supervisados o semi-supervisados, según sea que se tenga o no, entero conocimiento de cuáles son las clases en las que se deben clasificar cada patrón y si se sabe a qué clase corresponde un cierto número de patrones, denominados patrones de entrenamiento. En el presente trabajo solo se estudiarán aquellos algoritmos que sean de tipo supervisados.

#### 1.2.6.1 Redes neuronales

Las redes neuronales artificiales (RNA) son estructuras de procesamiento que emulan la forma de adquirir el conocimiento del cerebro humano a través del aprendizaje, en el caso particular del reconocimiento de fisuras, las RNA establecen una función que genera el hiperplano de decisión a partir de un proceso de aprendizaje supervisado.

Las RNA se componen por un conjunto de capas de neuronas que se clasifican como capas de entrada, capas ocultas y capas de salida, el número de capas y el número de neuronas por capa definen la arquitectura de la red. Las capas de entrada son las encargadas de recibir los descriptores y las capas de salida de asignar la clase a la cual corresponde el patrón en estudio, mientras que las capas ocultas son las encargadas de establecer el hiperplano de decisión. En una RNA se distinguen dos elementos fundamentales: las neuronas y las conexiones entre las neuronas. Las conexiones son las encargadas de llevar la información entre neuronas de acuerdo a un peso que se le asigna a la conexión. En cambio, la neurona es la encargada de asimilar, transformar y emitir la información. (Ver figura 19)

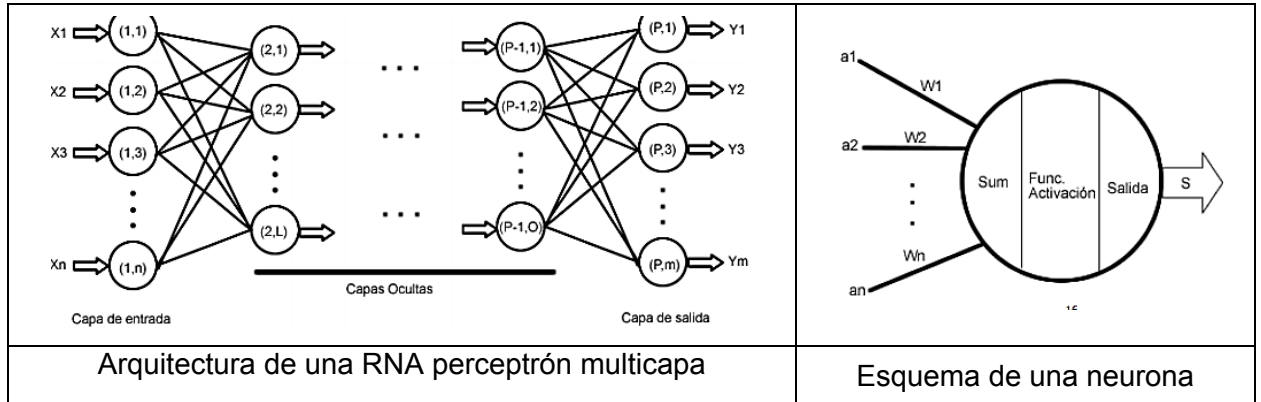
En el reconocimiento de patrones es común que la información dentro de la RNA sea conducida desde la capa de entrada hacia la de salida en una única dirección, por lo que se les denomina redes de alimentación hacia adelante o *perceptrón*. En este tipo de redes, cada neurona genera una salida (S) a partir de transformar mediante una función de activación (G) la sumatoria de las entradas (a) amplificadas por unos pesos (w), para la neurona i-ésima en la capa oculta k-ésima de la red neuronal esto es:

$$S_{i,k} = G_k \left( \sum a_{i,k} \cdot w_{i,k} \right)$$

Las funciones de activación que emplea cada neurona depende de la capa a la cual pertenezca y su función es llevar a un rango específico la sumatoria obtenida en las entradas. Por lo general, las funciones de activación en una RNA perceptrón son *logsigmoide* o *tansigmoide* en las capas ocultas y lineales en las capas de salida, donde la salida es un valor entre cero y uno.

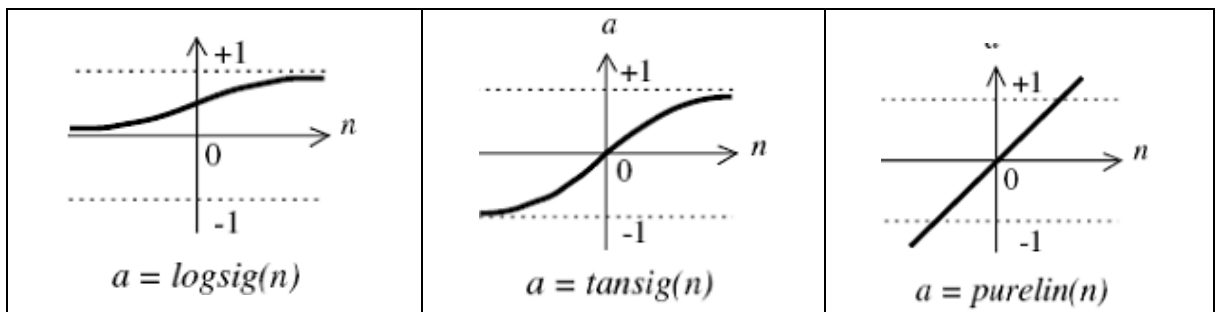
Por otro lado, los pesos de cada conexión en una RNA perceptrón son establecidos inicialmente de forma aleatoria, y se van re-calculando para modificar la función de decisión, hasta que esta permite clasificar con un mínimo error los patrones. El proceso de calibración de los pesos se denomina entrenamiento de la RNA, y es allí donde la red aprende a reconocer los patrones, almacenándose este conocimiento únicamente en las neuronas y las conexiones de las capas ocultas. (Ver figura 20).

Figura 19. Estructura de una red neuronal artificial



Fuente: Méndez, E. Mariño, J. Sistema automático de entrenamiento de redes neuronales artificiales basado en el ajuste genético de parámetros y variación de arquitectura. [trabajo de grado] Bucaramanga: UIS, 2003. p. 30,32.

Figura 20. Funciones de activación logsigmoide, tansigmoide y lineal en las redes neuronales



Fuente: Méndez, E. Mariño, J. Sistema automático de entrenamiento de redes neuronales artificiales basado en el ajuste genético de parámetros y variación de arquitectura. [trabajo de grado] Bucaramanga: UIS, 2003. p. 33.

La familia de algoritmos que comúnmente se emplean para el entrenamiento de la red son los de retro-propagación (*backpropagation*), los cuales permiten reasignarle el peso a cada conexión partiendo desde la capa de salida hasta las capa de entrada, reduciendo en cada época del entrenamiento el error medio cuadrado (MSE) entre la clasificación esperada y la obtenida. La fórmula para ajustar los pesos de las conexiones en cada época depende de la tasa de aprendizaje de la RNA (variación de los pesos entre dos épocas) y del algoritmo de *backpropagation* empleado, el cual puede ser el de: gradiente descendiente con tasa de aprendizaje fija o con tasa de aprendizaje variable, quasi-Newton o Levenberg-Marquardt<sup>27</sup>.

Las redes neuronales perceptrón de una o varias capas ocultas, han demostrado ser algoritmos muy eficaces para el reconocimiento de patrones dada su condición de poder aproximar cualquier problema mediante funciones complejas de decisión. Sin embargo,

<sup>27</sup>Méndez, E. Mariño, J. Sistema automático de entrenamiento de redes neuronales artificiales basado en el ajuste genético de parámetros y variación de arquitectura. [trabajo de grado] Bucaramanga: UIS, 2003. p. 24-44.

los resultados obtenidos por una red neuronal están altamente ligados a la arquitectura que se emplee, y adicionalmente, la variación en los resultados en función de la arquitectura no sigue un comportamiento lineal, por ende, el principal problema de las RNA es encontrar la mejor arquitectura, lo cual suele hacerse por prueba y error o por otros algoritmos<sup>28</sup>.

#### 1.2.6.2 *Máquina de soporte vectorial*

Las máquinas de soporte vectorial (MSV) son técnicas de clasificación fundamentadas en la teoría de aprendizaje estadístico, que buscan determinar un hiperplano de clasificación minimizando el error mediante la estructura de minimización del riesgo (SRM). Las soluciones obtenidas con las MSV se caracterizan por ser única y representar la solución óptima global.

Al igual que en las RNA las MSV se entrenan para aprender a reconocer los patrones, este entrenamiento equivale a resolver un problema de programación cuadrática con restricciones lineales, ideal en caso de contar con pocos datos que permitan realizar la clasificación supervisada y las clases presenten relaciones altamente no lineales.

Las funciones que emplean las MSV para separar las clases se conocen como funciones de kernel, las cuales suelen ser de tres tipos: la función polinomial de kernel, la función de criterio radial y la función sigmoidea. En la Figura 21 se presenta el esquema de la solución presentada por las MSV en el reconocimiento de patrones con clases separables linealmente.

#### 1.2.6.3 *Algoritmos genéticos*

Los algoritmos genéticos (AG) son sistemas computacionales que emplean mecanismos evolutivos para dar solución a un problema, siguiendo la teoría de la evolución de la especie de Charles Darwin. Para cualquier problema dado, los AG comienzan creando un conjunto de posibles soluciones al problema (generación), posteriormente, las soluciones (cromosomas) evolucionan mediante técnicas como la selección, la reproducción y la mutación, dando lugar a nuevas generaciones de cromosomas que brindan cada vez mejores resultados, hasta encontrar el cromosoma más apto para el problema. Históricamente, los algoritmos genéticos han sido empleados de dos formas en el reconocimiento de patrones (ver figura 21): como clasificadores directos<sup>29</sup> o como herramientas de optimización para establecer los parámetros de otros clasificadores<sup>30</sup>.

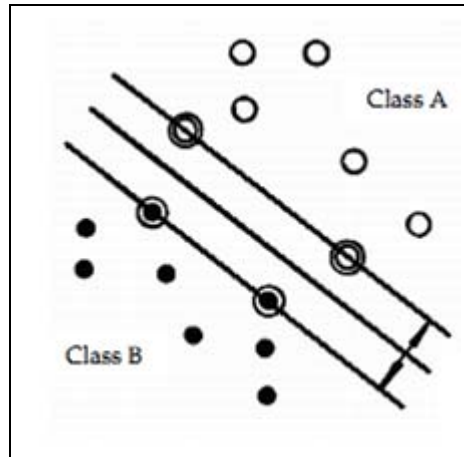
---

<sup>28</sup>Theodoridis, S. Op cit. p. 162-181.

<sup>29</sup>Rababaah, Haroun. Asphalt pavement crack classification: a comparative study of three AI approaches: multilayer perceptron genetic algorithms and self-organizing maps [maestría en ciencias]. Indiana : Indiana University South Bend, 2005. p. 39-47.

<sup>30</sup> Méndez, E. Mariño, J. Op. Cit.p. 6-23.

Figura 21. Clasificador por máquinas de soporte vectorial en clases separables linealmente



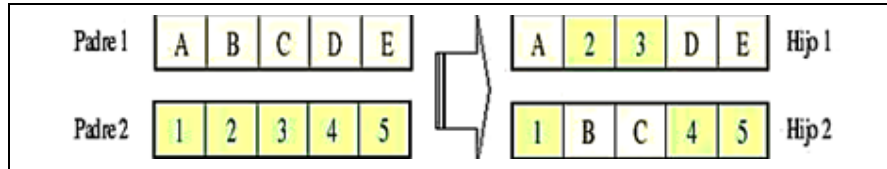
Fuente: Hu, Yong, Zhao, Chun-xia y Wang, Hong-nan. Automatic pavement crack detection using texture and shape descriptors. En: IETE Technical Review. (Sep – Oct. 2010); Vol. 27, no. 5; p. 400. [En línea] disponible en: [www.tr.letejournals.org](http://www.tr.letejournals.org) (2010: oct. 27).

En los AG, los cromosomas son un conjunto de bits o a un arreglo de caracteres, donde, por lo general, cada casilla (gen) tiene valores lógicos. Como se mencionó anteriormente, los cromosomas avanzan de generación en generación mediante tres técnicas evolutivas: la selección, el cruce y la mutación, las cuales representan cambios en los valores contenidos en las casillas del arreglo.

La selección consiste en determinar qué cromosomas de la generación se ajustan mejor para clasificar los patrones (mediante una clasificación supervisada), para determinar los cromosomas más aptos que darán origen a las nuevas generaciones, la selección suele llevarse a cabo por técnicas de selección por ruleta o por torneo. Por otro lado, la reproducción consiste en cruzar dos cromosomas para dar resultado a dos nuevos cromosomas que contenga genes procedentes de ambos padres, este cruce suele realizarse estableciendo puntos de corte en el código genético de los padres, que indican que lado del cromosoma cede cada padre a cada nuevo cromosoma. Finalmente, la mutación es una variación en los genes sin necesidad de la reproducción, lo que le permite al algoritmo buscar nuevas soluciones que no se pudiesen obtener únicamente del cruce, pero que podrían constituirse en una mejor solución al problema.

Los parámetros que configuran un AG son el tamaño máximo de la población de cromosomas, el número de generaciones a evaluar, la probabilidad de que un cromosoma se reproduzca y la probabilidad de que un cromosoma mute. Pese a que estos valores varían de acuerdo al problema, es común considerar un tamaño de población de 20, 25 generaciones, una probabilidad de cruce entre 0.8 y 0.95, y una probabilidad de mutación entre 0.01 y 0.05 (ver figura 22).

Figura 22. Técnica de cruce en dos puntos en un algoritmo genético



Fuente: Méndez, E. Mariño, J. Sistema automático de entrenamiento de redes neuronales artificiales basado en el ajuste genético de parámetros y variación de arquitectura. [trabajo de grado] Bucaramanga: UIS, 2008. P. 13.

### 1.3 ALGORITMOS PARA EL RECONOCIMIENTO DE FISURAS EN EL PAVIMENTO MEDIANTE IMÁGENES DIGITALES

El uso de algoritmos para el reconocimiento de fisuras en el pavimento mediante imágenes digitales han sido ampliamente tratados en las últimas décadas, planteándose distintas metodologías en busca de obtener mejores identificaciones, clasificaciones y mediciones. A continuación, se presentan las principales propuestas para el reconocimiento de fisuras tanto en pavimento en concreto hidráulico como en asfalto disponibles en la literatura, así como una breve descripción de las técnicas que cada algoritmo emplea y los resultados obtenidos.

En la revisión de la literatura, el primer algoritmo encontrado con resultados significativos fue planteado en 1993 por Kaseko y Ritchie (1993), el cual propuso una segmentación en tres etapas: primero se homogenizó la imagen de intensidades empleando un factor de estandarización para cada columna, segundo, se segmentó la imagen usando un umbral estimado por una RNA, y tercero, se unieron las fisuras con la técnica de suavizado modificado y se eliminaron las regiones más pequeñas. La extracción de características constó de cinco descriptores de orientación extraídos para recuadros de 32x29 píxeles con base en la técnica de proyección del histograma. La clasificación se realizó primero para cada recuadro y posteriormente para toda la imagen, usando una RNA perceptrón multicapa con una arquitectura 5-5-5. El procedimiento se planteó para fisuras en el pavimento asfáltico, distinguiendo cinco tipos de fisura: longitudinales, transversales, de piel de cocodrilo, de bloque o no son fisuras. Los resultados obtenidos en la clasificación fueron del 98,5 % en las fisuras transversales y longitudinales y del 85,7% en las de bloque y piel de cocodrilo. Adicionalmente, se calculó la longitud de las fisuras, su área de daño, el ancho promedio y el grado de severidad de las fisuras.

Posterior a ello, en los años noventa y a inicios del siglo XX surgieron una variedad de algoritmos enfocados al reconocimiento de fisuras en el pavimento, dentro de los cuales cabe destacar los plantados por Chou et al (1994) empleando descriptores de momento espacial y redes neuronales, Hsu et al (2001) quien también usó descriptores de momentos y redes neuronales y obtuvo eficacias del 85%, y Lee y Lee (2003) que optó de nuevo una por una RNA, pero empleando descriptores estadísticos en recuadros de la imagen que arrojaron niveles de acierto del 95%.

Ya en el 2008, Yao, X. et al presentó un algoritmo que segmenta las fisuras de área en el pavimento rígido con el uso de un umbral local adaptativo basado en el segundo momento angular. Las imágenes segmentadas se representaron con regiones, y los descriptores seleccionados fueron: el área, la solidez, la orientación y los ejes, así como un nuevo criterio denominado criterio de región cerrada. La clasificación se realizó para cuatro tipos distintos de fisura de área, obteniéndose errores de entre 3,5% y 21,8%.

En ese mismo año, Xu, G. et al planteó un algoritmo que segmentó las fisuras en dos etapas: primero se homogenizó la imagen de intensidades con la técnica de ecualización del histograma por recuadros, y posteriormente, se segmentó la imagen mediante filtros espaciales. La clasificación se realizó, en primera instancia, dividiendo la imagen en recuadros de 25x20 píxeles, los cuales se clasificaron como fisuras o no fisuras, en función de cuatro descriptores alusivos a la distribución de los píxeles en blanco y negro en cada recuadro; el algoritmo de clasificación empleado fue una RNA perceptrón multicapa con una arquitectura de 4-20-1. Finalmente, la clasificación realizada se revisó de acuerdo a la conexión que seguían los recuadros con fisuras. Los resultados obtenidos muestran un error de clasificación de fisura o no fisura entre 2,8% y 1%.

Posteriormente, en el 2009 Zhaoyun, S. et al desarrolló un algoritmo de reconocimiento que segmentó las imágenes empleando la técnica de umbral condicional, seguida de una erosión y una dilatación de la imagen binaria para eliminar el ruido. Las fisuras se representaron en función de su esqueleto y luego se unieron utilizando la operación de relleno de espacios para diferencia de ángulos de hasta 20°. La clasificación de las fisuras se realizó, en primera instancia, con base en el número de huecos presentes en el esqueleto, distinguiéndose dos clases: fisuras de línea y de piel de cocodrilo. Acto seguido, las líneas de fisuras se clasificaron como transversales o longitudinales en función de la cantidad de cerros presentes en el código de la cadena. Finalmente, se calculó la longitud y el área fisurada y se estableció el grado de severidad conforme al índice de fisura del pavimento (*cracking index*).

En el 2010, Jing, L. y Aiquin, Z presentó un algoritmo para identificar únicamente las fisuras longitudinales en el pavimento asfáltico, para ello, primero se homogenizó las imágenes con la técnica de interpolación bilineal, posteriormente, se segmentaron las fisuras utilizando un umbral obtenido del histograma de intensidades, denominado como segmentación bivaluada; finalmente, el ruido generado de la segmentación se eliminó mediante la técnica de expansión de puntos, la cual es semejante a la operación morfológica de apertura. El algoritmo propuesto permitió identificar fisuras longitudinales de ancho superior a los 2 mm e incluso fisuras de piel de cocodrilo.

También en el 2010, Wang, C. et al propuso un algoritmo para la clasificación de las fisuras empleando el código de la cadena. En el pre procesamiento de la imagen se eliminó el ruido mediante un filtro mediano y se redujo la intensidad de las señales horizontales del pavimento. La segmentación de las fisuras se realizó con el método del umbral por máxima entropía, seguido de la aplicación de las técnicas morfológicas de apertura y cierre. Posteriormente, las fisuras se representaron por su esqueleto y se empleó el código de la cadena para su clasificación, considerando como descriptores el porcentaje de cada dirección presente en el código de la cadena normalizado, y

estableciendo como clases de fisuras la longitudinal, la transversal y la de piel de cocodrilo. Los resultados obtenidos concluyen un porcentaje de acierto superior al 85%.

Por su parte, HU, Y. et al (2010) planteo un algoritmo para determinar si en una imagen habían o no fisuras con base en descriptores de forma y de textura. El algoritmo dividió la imagen en recuadros de 16x16 pixeles, en cada uno de los cuales se extrajeron los siguientes descriptores: energía, contraste, correlación, homogeneidad, entropía, la dirección de los ejes principales y la redondez. Posteriormente, en cada recuadro se determinó si había o no una fisura empleando una máquina de soporte vectorial (SVM). Finalmente, tras haber clasificado cada recuadro se segmentaron los que correspondían a fisuras. Los resultados obtenidos muestran una efectividad del algoritmo entre 83% y 93%, dejando de clasificar como fisura tan solo el 1.3% de los recuadros que contenían fisuras.

En el 2011, Ouyang, A. et al desarrolló un algoritmo para el reconocimiento de fisuras en la superficie de pavimentos asfálticos, el cual, en primera instancia, eliminó el ruido de la imagen haciendo uso del filtro mediano, posteriormente, segmentó las fisuras mediante la técnica de recorte sucesivo usando un umbral de 0.6, y finalmente, clasificó las fisuras siguiendo el protocolo del índice universal de fisura del Banco Mundial a partir de medir la longitud, el ancho promedio y el área fisurada para cada fisura.

Finalmente, en el presente año Nejad, F. y Zakeri, H. planteó el uso de la ecualización del histograma y la transformada de Fourier de la imagen digital en recuadros de 32x32 pixeles como técnicas para el pre-procesamiento de la imagen. Posteriormente, se empleó una red neuronal, la transformada de Wavelet y la transformada de Radon para mejorar la calidad de las imágenes digitales. Finalmente, el algoritmo propuso el uso de 18 descriptores de forma (área, perímetro, elongación, número de Euler y momentos espaciales, entre otros) y el uso de dos nuevas RNA perceptrón multicapa: una empleando un algoritmo estadístico backpropagation y otro un algoritmo dinámico backpropagation. Los resultados obtenidos presentan una efectividad promedio del 87% para las redes neuronales dinámicas y del 93% para las redes neuronales estadísticas.

## 2. RECOLECCIÓN DE LAS IMÁGENES DIGITALES

En la revisión de las técnicas existentes para la recolección de imágenes digitales de fisuras en el pavimento, se encontró que no existe un criterio unificado sobre como deba llevarse a cabo este procedimiento, razón por lo cual los sistemas propuestos varían en la metodología seguida, la tecnología empleada y los resultados obtenidos. Sin embargo, la variedad de sistemas consultados sirvieron de guía, primero, para establecer una metodología de recolección manual de las imágenes, y luego, para plantear un sistema semi-automático de captura de imágenes digitales de fallas en pavimentos de vías urbanas.

### 2.1 METODOLOGÍA PARA LA RECOLECCIÓN DE LAS IMÁGENES DIGITALES

La recolección de las imágenes digitales se realizó de forma manual, por los siguientes argumentos: primero, no existe actualmente un equipo automatizado para la recolección de imágenes digitales en la Universidad Industrial de Santander, segundo por los alcances para este proyecto, y tercero, porque ya se han realizado algoritmos para el reconocimiento de fisuras a partir de imágenes tomadas de forma manual<sup>31</sup>, obteniéndose buenos resultados.

La metodología de recolección se ciñó a las recomendaciones dadas por Austroads<sup>32</sup> y se resumió de la siguiente forma:

1. Localizar las vías que en Bucaramanga y su área metropolitana estuvieran pavimentadas en concreto hidráulico y que presentaran fisuras en su superficie - tomando como referencia la información suministrada en el Modelamiento para la gestión del mantenimiento, rehabilitación y mejoramiento de pavimentos urbanos, y seleccionar una muestra de tramos viales con bajos volúmenes de tránsito.
2. Recorrer en horas de la mañana, y en días despejados, los tramos viales seleccionados en el sentido del flujo vehicular, y fotografiar las losas fisuradas, empleando una cámara de 10 megapíxeles de resolución.
3. Para realizar la captura de las imágenes, se ubica manualmente la cámara perpendicular a la superficie y a una altura promedio de 1.8 metros del pavimento.
4. En caso de no contarse con suficientes fisuras de cada tipo de falla (según el Manual de inspección visual de pavimentos rígidos del Invías) seleccionar y recorrer nuevos tramos viales.
5. Eliminar aquellas imágenes donde: se presenten cambios de pavimento rígido a pavimento flexible, el pavimento rígido este húmedo, se observen objetos extraños tales como partes de vehículos y la superficie del pavimento este desgastada.

---

<sup>31</sup>Rababaah Op. cit. p. 17-18.

<sup>32</sup>Austroads.Op cit. p. 2-7.

A partir de la metodología establecidas e identificaron las siguientes características para las imágenes digitales capturas:

- Las imágenes tienen un tamaño de 3240x4320 pixeles (la primera dimensión es en el sentido del flujo vehicular).
- Las imágenes son a color (RGB).
- Las imágenes representan un área de losa de 1,6 metros de largo por 2 metros de ancho, lo que resulta en una relación milímetros por pixel (mm/pixel) de aproximadamente 0,5.
- En total se seleccionaron 175 imágenes o muestras para la caracterización y el desarrollo del algoritmo (ver figura 23).

Figura 23. Muestra de las imágenes digitales recolectadas



Fuente: Los autores.

Finalmente, la metodología propuesta permite establecer los escenarios en los cuales el algoritmo propuesto no será evaluado:

- Imágenes donde el pavimento de losas de concreto esté en condiciones húmedas.
- Imágenes donde se presenten cambios del tipo de pavimento.
- Imágenes donde el pavimento sea asfáltico.
- Imágenes donde la superficie forme un ángulo oblicuo con la cámara.
- Imágenes en presencia de sombras.

## **2.2 PROPUESTA DE UN SISTEMA SEMI-AUTOMÁTICO PARA LA RECOLECCIÓN DE LAS IMÁGENES DIGITALES**

El planteamiento de un sistema semi-automático para la recolección de las imágenes digitales surge de la importancia de contar con un equipo que incremente la rapidez en la recolección de la información, reduzca la labor y los riesgos del trabajo en campo, y permita registrar toda la malla vial de Bucaramanga y su área metropolitana a costos

inferiores a los requeridos con las técnicas de inspección manual; además de presentar un aporte para el desarrollo del proyecto “Sistema automatizado para clasificación y evaluación de daños de pavimentos de vías urbanas”. Es importante señalar que la propuesta elaborada es coherente con la literatura y las tecnologías y los recursos disponibles por la Universidad Industrial de Santander.

### **2.2.1 Esquema del sistema semi-automático**

El sistema de recolección planteado tiene por objeto capturar las imágenes digitales en campo, comprimirlas y almacenarlas en una base de datos para ser almacenadas (también se busca crear una copia temporal en un dispositivo de almacenamiento externo), desde la cual las imágenes serán analizadas y posteriormente ofrecidas como un servicio web en el observatorio SIG del área metropolitana de Bucaramanga.

En particular, para la recolección de las imágenes el esquema general del sistema se compone de un vehículo, una cámara digital, una estructura de soporte para la cámara digital, un equipo de posicionamiento global (GPS), un odómetro para indicar la abscisa a la cual fue tomada la imagen digital sobre el corredor vial, un sistema de iluminación artificial y un computador o un disco externo para almacenar temporalmente las imágenes.

### **2.2.2 Especificaciones del sistema semi-automático**

Para comenzar, el sistema está planteado sobre un vehículo Toyota Hilux 2009 doble cabina, por ser el vehículo con el cual cuenta actualmente la Universidad Industrial de Santander. Adicionalmente, se propone una estructura de soporte para la cámara digital en tubo hueco de aluminio de 2” de diámetro, con conexiones al vehículo en dos puntos: uno sobre la carrocería - en el lugar donde se ubica la defensa del vehículo-empleando dos pernos de  $\frac{3}{4}$ ” a cada lado del vehículo, y dos, sobre el sistema antivuelco del vehículo mediante dos abrazaderas giratorias, como se muestra en el Anexo A.

Por otro lado, se propone el uso de una cámara digital ubicada a 2,5 metros de la superficie del pavimento, con tecnología de escaneo lineal y una resolución superior a los 4096 pixeles para garantizar un grado de detalle de 1 milímetro por pixel, así como una frecuencia de disparo superior a los 7 kHz (una cámara disponible en el mercado y que cumple con estas especificaciones es la Basler L401K). Adicionalmente, se debe contar con un sistema de sincronización entre la captura de las imágenes y su georreferenciación con un dispositivo GPS y un odómetro; en cuanto al GPS y al odómetro a emplear, no se presentan mayores requerimientos excepto que su sensibilidad sea inferior a 1 metro. Mientras tanto, el disco de almacenamiento debe tener una capacidad de almacenamiento superior a los 500 gigabytes, puesto que se espera que cada segundo se requieran 13 megabytes de memoria si las imágenes se almacenan en formato JPG.

Por último, para asegurar una iluminación homogénea en las imágenes capturadas se plantea el uso de siete bombillas halógenas. Se plantó el uso de luces halógenas porque

brindan una iluminación focalizada, proporcionan más nitidez y economía que las luces estroboscópicas, y pueden conectarse a la batería del vehículo mediante un transformador de corriente. Para la distribución de las bombillas se propone que cinco de ellas estén ubicadas a 1 metro de la superficie y las otras dos a 1,8 metros, tal y como se observa en el Anexo A; todas las bombillas se plantean con una potencia de 50 watts y una luminosidad de 1000 lux.

### **2.2.3 Características del sistema semi-automático**

El sistema propuesto tiene las siguientes características:

- Velocidad de operación de hasta 40 km/horas.
- Ancho de la imagen de aproximadamente 3,5 metros.
- Iluminación homogénea en todo el ancho de la imagen.
- Grado de detalle de aproximadamente 1 mm/pixel.
- Georreferenciación inmediata de las imágenes capturadas.
- Capacidad de ser desmontado del vehículo cuando no se le requiera.

### 3. PRE-PROCESAMIENTO

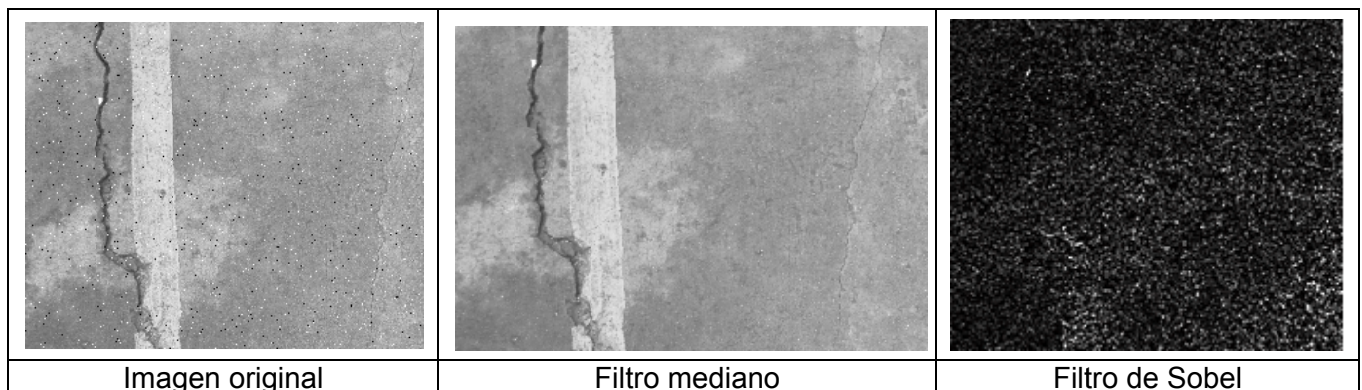
Tras pasar a escala de grises las imágenes capturadas, se pudo evidenciar la presencia de ruido aleatorio - o de *“salt and pepper”*-, y la carencia de homogeneidad en la iluminación, lo que disminuía el contraste de la imagen y dificultaba la segmentación de las fisuras. Por ende, fue necesario implementar técnicas de pre-procesamiento, enfocadas a eliminar el ruido, suavizar los bordes y homogeneizar el fondo de la imagen para realzar el contraste fisura-superficie del pavimento rígido.

#### 3.1 FILTRADO

Para determinar el filtro a utilizar, se compararon los resultados obtenidos por un filtro no-lineal (filtro mediano) y por dos filtros lineales (filtro de Sobel y filtro Laplaciano), encontrándose que el filtro que reduce mejor el ruido es el filtro mediano, tal y como se encuentra en la revisión de la literatura<sup>33</sup>.

Después de haber definido el tipo de filtro a emplear, fue necesario determinar la máscara a utilizar en la técnica de filtro mediano, dicha selección se realizó comparando el comportamiento de una máscara de tamaño 3x3, 5x5, 7x7 y 9x9, encontrando que la máscara de 3x3 no reduce significativamente el ruido en la imagen, mientras que las máscaras superiores a 7x7 generan pérdidas en el detalle del contorno de las fisuras, por lo que en definitiva se optó por un filtro mediano con una máscara de 5x5.

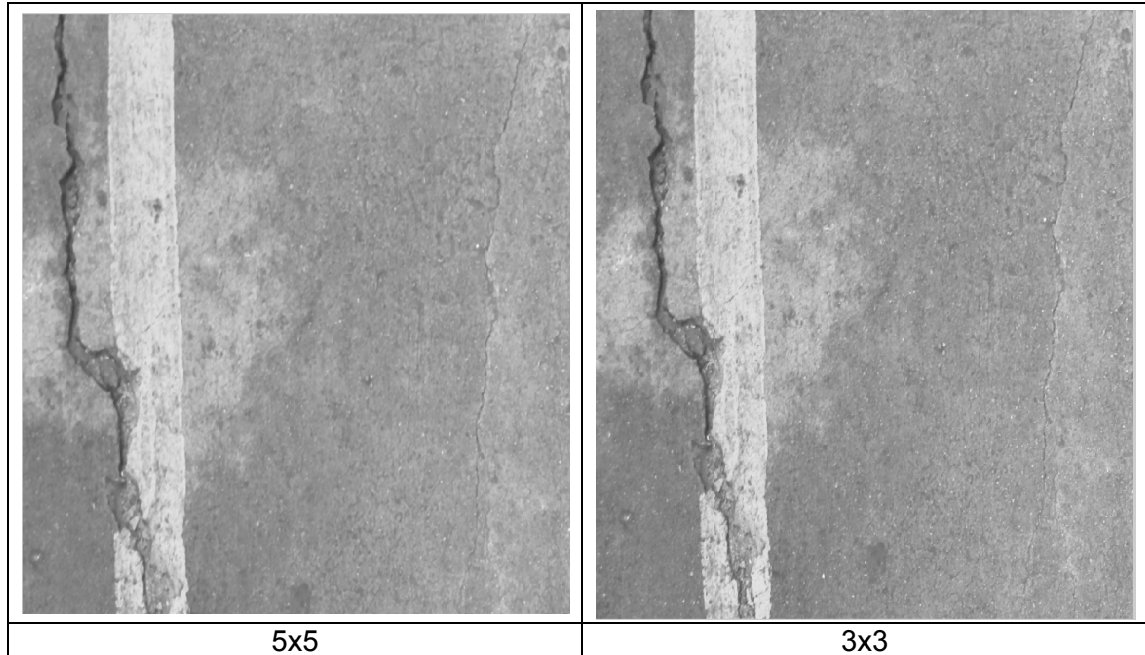
Figura 24. Comparación del filtro mediano con un filtro lineal.



Fuente: Los autores.

<sup>33</sup>Ouyang, Aiguo. Luo, Chagen, Zhou, Chao. Surface distress detection of pavement based on digital image processing. En: Computer and computing technologies in agriculture IV. Nanchang: IFIP, 2010. p. 370-371.

Figura 25. Máscaras empleadas para el filtro mediano.



Fuente: Los autores.

### 3.2 HOMOGENIZACIÓN DE LA IMAGEN

Tras haber eliminado la mayoría del ruido presente en la imagen se procedió a implementar una técnica que permitiera mejorar el contraste entre las fisuras y la superficie del pavimento rígido.

Para seleccionar qué algoritmo implementar, primero se descartó la técnica de la ecualización del histograma dado que según la literatura no presenta buenos resultados en las imágenes de fisuras en el pavimento, luego se analizó el desempeño de las técnicas que, a cambio de mejorar el contraste modificando el histograma de toda la imagen, dividen la imagen en recuadros y uniformizan la intensidad de la superficie en cada recuadro, encontrándose resultados satisfactorios, en particular, se optó por la técnica de homogenización empleando un factor de ajuste. El algoritmo implementado se presenta a continuación, como una adaptación del presentado por Ying, L. (2009).

En primer lugar, se definió un valor de intensidad  $B$  -al cual se quería homogeneizarla superficie del pavimento- como el promedio de las intensidades de la imagen original (además de la media se intentó hacer uso de la mediana, sin embargo, los resultados obtenidos no fueron satisfactorios). Luego, la imagen se dividió en aproximadamente 36 recuadros, cada uno de  $540 \times 720$  píxeles, y en cada uno de ellos se calculó la media ( $G_{m,i}$ ), el valor máximo ( $G_{máx,i}$ ) y el valor mínimo ( $G_{mín,i}$ ) de las intensidades. El número

de recuadros en los que se dividió la imagen se obtuvo mediante prueba y error, de forma tal que no se perdiera el detalle de los bordes de las fisuras.

Con el fin de calcular en cada recuadro la intensidad promedio de la superficie del pavimento se estableció un límite superior y un límite inferior para cada recuadro, dentro de los cuales se calculó la media modificada del recuadro ( $G'_{m,i}$ ); para el recuadro  $i$ -ésimo los límites se definieron como:

$$L_{sup,i} = G_{m,i} + k \cdot (G_{máx,i} - G_{máx,i})$$

$$L_{inf,i} = G_{m,i} - k \cdot (G_{m,i} - G_{mín,i})$$

Donde  $k$  es una constante que suele considerarse como 0.6 según la literatura. Posteriormente, se calculó el factor de ajuste para cada recuadro, entendido como el cociente entre la intensidad objetivo ( $B$ ) y la media modificada de cada recuadro ( $G'_{m,i}$ ):

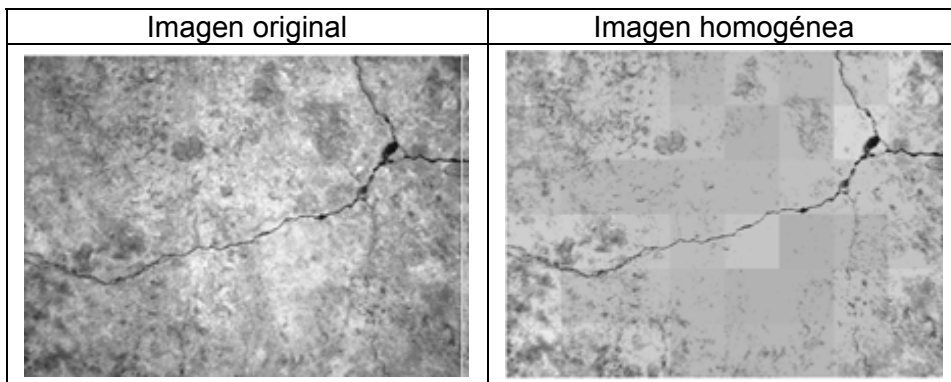
$$f_i = \frac{A}{G'_{m,i}}$$

Finalmente, todos los pixeles de la imagen original se amplificaron por el factor  $f_i$  del recuadro correspondiente, exceptuando aquellos pixeles cuya intensidad fuese mayor a  $B$ , los cuales fueron igualados a la media de la imagen original. Esto es:

$$P'(x,y) = \begin{cases} P(x,y) \cdot f_i & \leftrightarrow P(x,y) < B \\ B & \leftrightarrow P(x,y) \geq B \end{cases}$$

Los resultados del algoritmo se presentan en la Figura 26, donde se aprecia que además de obtener una imagen homogénea, adicionalmente, se uniformizaron las señales horizontales y se preservó el detalle en los bordes de las fisuras cuyo ancho promedio fuese superior a los 3 mm. El código desarrollado en Matlab para este algoritmo se encuentra en el Anexo B titulado **imhomogenea**.

Figura 26. Homogenización de la imagen en escala de grises.



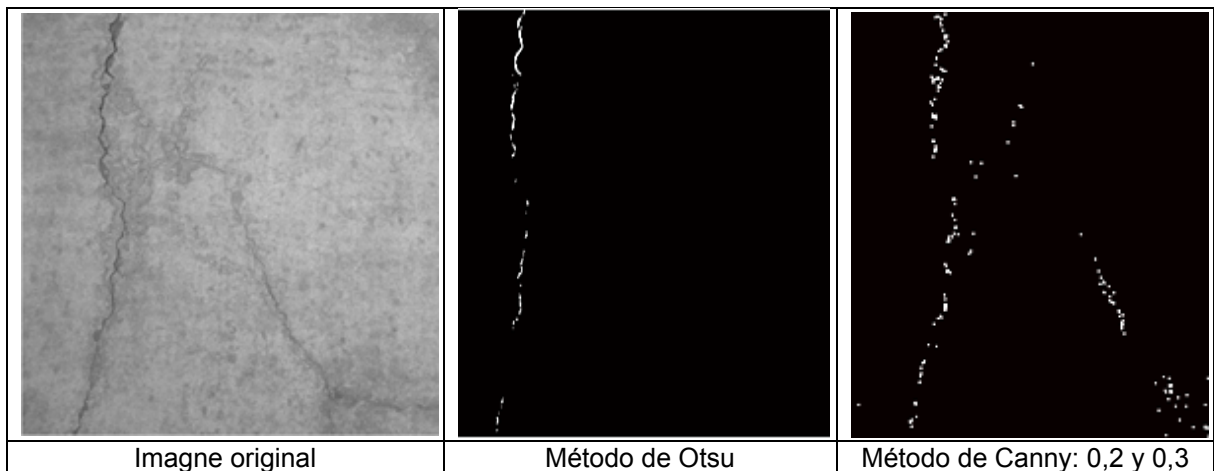
Fuente: Los autores.

## 4. SEGMENTACIÓN

La segmentación de las fisuras es considerada por muchos autores como el paso crucial en la identificación y clasificación de fisuras mediante imágenes digitales, de allí que se hayan hecho múltiples intentos por avanzar en el uso de técnicas que realicen la segmentación de forma totalmente automática, sin que aún se haya tenido un resultado definitivo.

Para determinar el algoritmo a implementar se evaluaron varias técnicas de segmentación presentes en la literatura y se compararon los resultados obtenidos para las imágenes recolectadas. En primera instancia, se emplearon las técnicas de segmentación por umbral y se corroboró que los resultados obtenidos no son los esperados, esto se debe a que estas técnicas están enfocadas a la solución de problemas donde el histograma de intensidades es bimodal, mientras que el histograma de las imágenes de fisuras en el pavimento rígido es unimodal. Posteriormente, se evaluó el comportamiento de la técnica de detección de bordes de Canny, y pese a que los resultados fueron satisfactorios, se encontró que los parámetros que requiere este procedimiento varían notablemente en cada imagen, haciéndose difícil emplearlo para una segmentación automatizada<sup>34</sup>.

Figura 27. Resultados de las distintas técnicas de segmentación.



Fuente: Los autores.

Por otro lado, del estudio realizado por Chang, Y. et al (2010) se concluyó que de las seis técnicas evaluadas (técnica de relajación, de Canny, de Wavelet multiescala, de recorte sucesivo, de verificación de fisuras semilla y de optimización dinámica) los mejores resultados se obtuvieron para la técnica de recorte sucesivo, teniendo en cuenta el

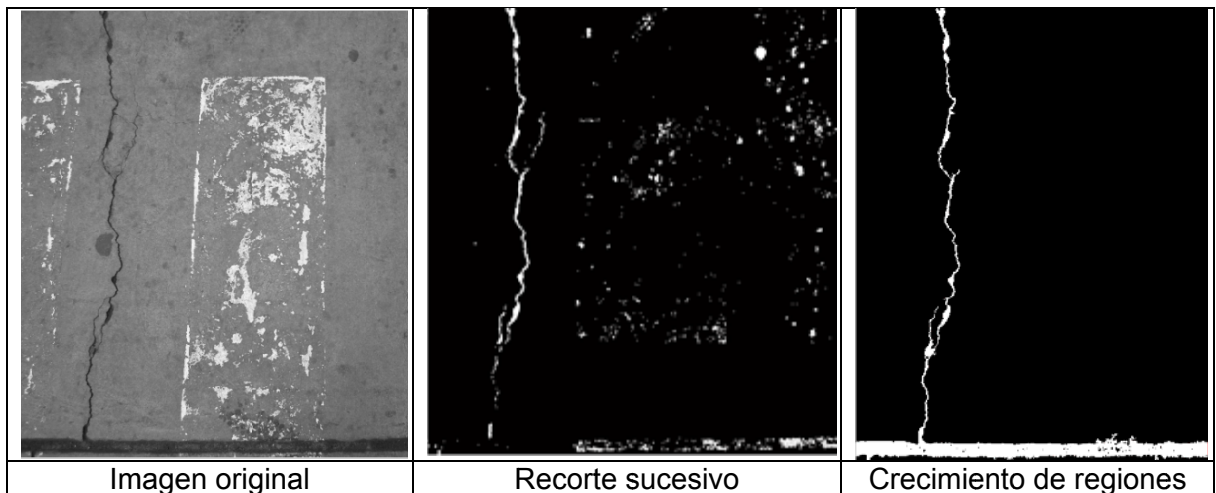
<sup>34</sup>Chang, Y. et al. Critical assessment of pavement distress segmentation method. En: Journal of transportation engineering. Vol. 136, no. 11; 2010. Pág. 12-18.

porcentaje de fisuras detectado y el tiempo de ejecución empleado. Adicionalmente, se implementó la técnica de crecimiento por regiones y se compararon, por inspección visual, los resultados obtenidos con los dos métodos, concluyendo que para las imágenes capturadas los mejores resultados se obtuvieron con el algoritmo de crecimiento por regiones.

#### 4.1 CRECIMIENTO POR REGIONES

En el presente proyecto se implementó la técnica de crecimiento por regiones para llevar a cabo la segmentación de las fisuras en el pavimento rígido, el procedimiento que se implementó es una modificación del presentado por Gonzalez, R. et al. (2009):

Figura 28. Comparación de las técnicas de segmentación de recorte sucesivo y crecimiento por regiones.



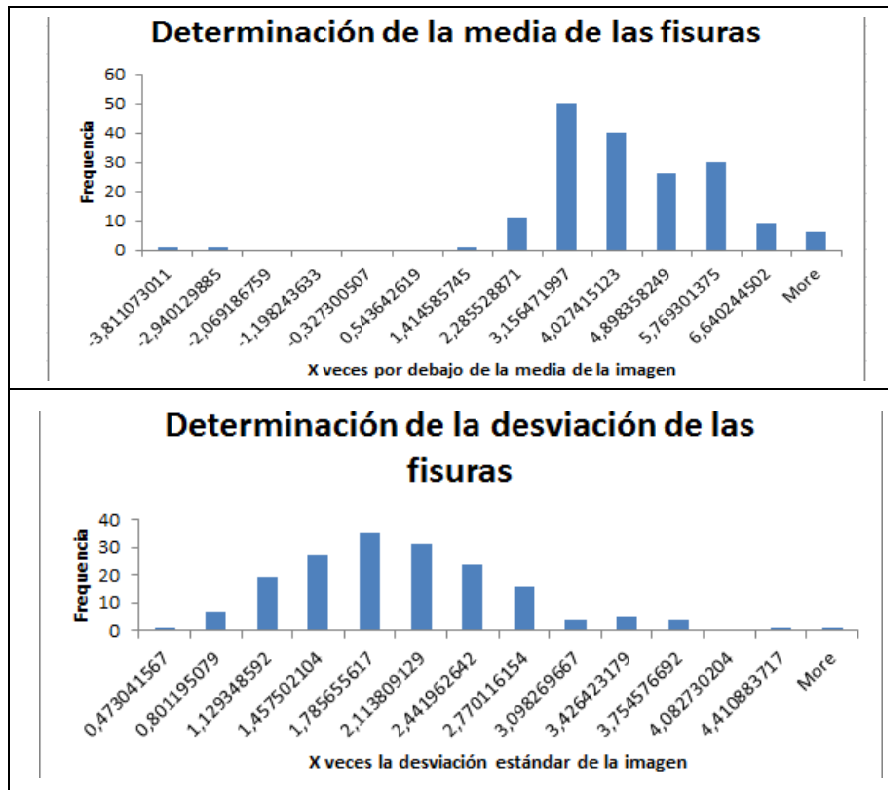
Fuente: Los autores.

Para comenzar, se definieron los parámetros de valor semilla (S) y tolerancia (T) que requiere el algoritmo: el valor semilla se definió como la intensidad promedio de las fisuras, mientras que la tolerancia se definió como la desviación estándar de la intensidad de las fisuras. Con el ánimo de automatizar la estimación de estos valores, se optó por establecer una relación entre los parámetros requeridos entre la media y la desviación estándar de la imagen, para ello se realizó un análisis estadístico descriptivo sobre la relación entre la media de la intensidad de la imagen y la media de intensidad de las fisuras, y entre la desviación de las intensidades en la imagen y la desviación de las intensidades en las fisuras para las 175 imágenes capturadas. Los resultados de dicho estudio (ver figura 29) concluyeron que la media de las fisuras es igual, aproximadamente, la media de la imagen menos cuatro veces la desviación estándar de la imagen, y que la desviación de las fisuras es dos veces y media la desviación estándar de la imagen, esto es:

$$S = \mu_{fisuras} \approx \mu_{imagen} - 4 \cdot \sigma_{imagen}$$

$$T = \sigma_{fisuras} \approx 2,5 \cdot \sigma_{imagen}$$

Figura 29. Histogramas empleados para definir los parámetros del algoritmo de crecimiento por regiones.



Fuente: Los autores.

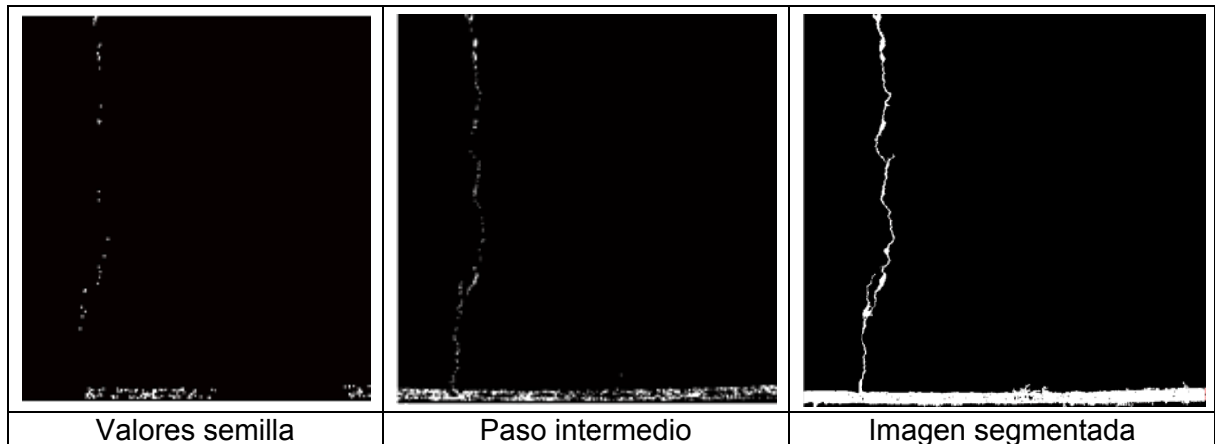
Una vez se automatizó el cálculo de los dos parámetros, el procedimiento que se siguió fue:

1. Encontrar los píxeles que tuviesen una intensidad igual a la media de las fisuras (en caso de no haber ningún pixel que coincidiera se tomó como valor semilla a la intensidad más cercana a la media de las fisuras).
2. Conectar los píxeles semilla con los píxeles que estuvieran próximos en cualquiera de las ocho direcciones y cuya intensidad no excediera el rango de tolerancia, definido desde  $S-T$  hasta  $S+T$ . Se continuó con este procedimiento hasta que no hubo más píxeles por conectar.

El resultado fue una imagen donde al primer conjunto de píxeles encontrando (región) se les asignó el valor de uno, al siguiente dos y así sucesivamente (buscando de arriba hacia abajo y de izquierda a derecha en la imagen), mientras que a los píxeles que no pertenecían a ninguna región se les asignó un valor de cero.

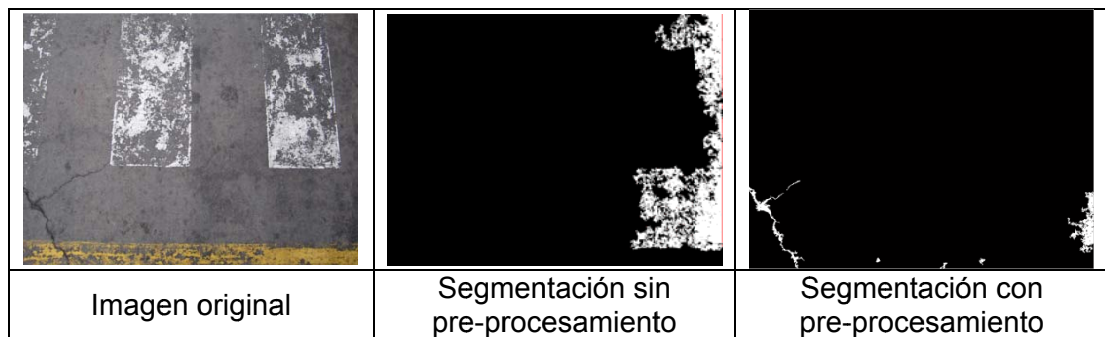
El resultado del procedimiento fue la imagen binaria, donde cada objeto se representó como una región. Finalmente, para resaltar la influencia del pre-procesamiento de la imagen en la segmentación, en la Figura 32 se muestran los resultados de la segmentación aplicando este algoritmo y omitiéndolo. El código en Matlab para el algoritmo de crecimiento de regiones está disponible en el Anexo B, titulado **cregiones**.

Figura 30. Representación del funcionamiento del algoritmo de crecimiento por regiones.



Fuente: Los autores.

Figura 31. Influencia del pre-procesamiento de la imagen en la segmentación.



Fuente: Los autores.

#### 4.2 ALGORITMO PARA ELIMINAR EL RUIDO Y LAS JUNTAS DE LAS LOSAS

Los resultados obtenidos del algoritmo de segmentación por regiones presentaron tres inconvenientes: las juntas de las losas también fueron segmentadas y conectadas a las fisuras, además, son segmentados otros objetos distintos a las fisuras tales como las manchas en el pavimento, y por último, las fisuras quedaron separadas por tramos, lo que

dificultaba su caracterización. Para dar solución a los dos primeros problemas se empleó el siguiente algoritmo:

En primer lugar, para eliminar las juntas de las imágenes binarias se aplicó dos veces la operación morfológica de *tophat*: una vez para eliminar las juntas horizontales y otra para eliminar las juntas verticales. Las estructuras planteadas para emplear el operador *tophat* fueron dos rectángulos: uno de 0.5 cm de ancho y 25 cm de largo para eliminar las juntas verticales, y otro de 25 cm de ancho y 0.5 cm de alto para eliminar las juntas horizontales, cuyas se definieron considerando que el ancho de una junta es de entre 0.5 cm y 1.2 cm<sup>35</sup>, y que éstas no están totalmente orientadas en dirección horizontal o vertical en las imágenes digitales.

Posteriormente, se caracterizó la forma de las fisuras y se eliminó de la imagen binaria todas aquellas regiones que no tuviesen una forma similar a la de una fisura o que correspondiesen a tramos de juntas que no fueron del todo removidos con la operación morfológica. Los descriptores de forma empleados para este procedimiento fueron, en el siguiente orden, la redondez, la excentricidad y la orientación, condicionando esta última alalongitud del eje mayor de la región. En la Tabla 1 se presenta la caracterización realizada.

Tabla 1. Caracterización morfológica de las fisuras, tramos de juntas y otras regiones.

Descriptores	Fisuras	Tramos de juntas	Otras regiones
Redondez ( $A/P^2$ )	<0,025	<0,025	>0,025
Excentricidad ( $e$ )	>0,95	>0,95	<0,95
Orientación ( $\phi$ )	0-90°	0°-5° o 85°-90°	0-90°
Eje mayor ( $E_M$ )	Cualquiera	<50 cm	Cualquiera

Fuente: Los autores.

A partir de la caracterización realizada, fue posible eliminar las regiones indeseadas mediante la condición enunciada a continuación. El código en Matlab para eliminar regiones se puede encontrar en el Anexo B con el nombre de **eregiones**.

$$P(x, y) \subseteq R = \begin{cases} 0 & \leftrightarrow \text{en } R: \frac{A}{P^2} > 0,025 \vee e < 0,95 \vee [(\phi \in [0, 5] \vee \phi \in [85, 90]) \wedge E_M < 50 \text{ cm}] \\ 1 & \text{en otro caso} \end{cases}$$

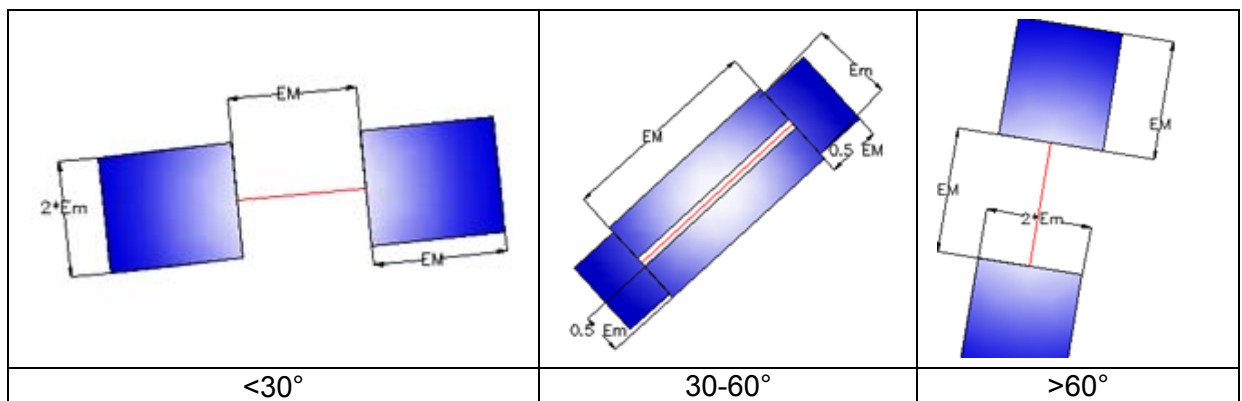
### 4.3 ALGORITMO PARA UNIR LAS REGIONES

Para culminar el proceso de segmentación se propuso el uso de un algoritmo que permitiera unir las regiones que hicieran parte de una misma fisura, para ello se adoptó y adaptó el algoritmo propuesto por Liu, F. et al (2008). El procedimiento desarrollado fue:

<sup>35</sup>Secretaría de Integración Económica Centroamericana. Manual centroamericano para diseño de pavimentos. 2002. Pág. 99.

1. Ordenar las regiones de mayor a menor según la longitud de su eje mayor, y calcular su orientación.
2. De acuerdo a la orientación de la primera región ordenada, definida como región base, establecer rectángulos de búsqueda (ver Figura 32) y definir como regiones cercanas aquellas que tuvieran al menos un pixel contenido en dichos rectángulos y cuya orientación no difiera a la de la región base en más de  $30^\circ$ .
3. Calcular la menor distancia entre la región base y todas las regiones cercanas, y realizar la conexión más corta.
4. Repetir el procedimiento para todas las regiones, evitando conectar aquellas que ya se hubieran considerado como regiones base.
5. Ordenar de nuevo las regiones de acuerdo a la longitud de su eje mayor tras haber realizado las conexiones y repetir el procedimiento hasta que no hubiese más regiones a unir.

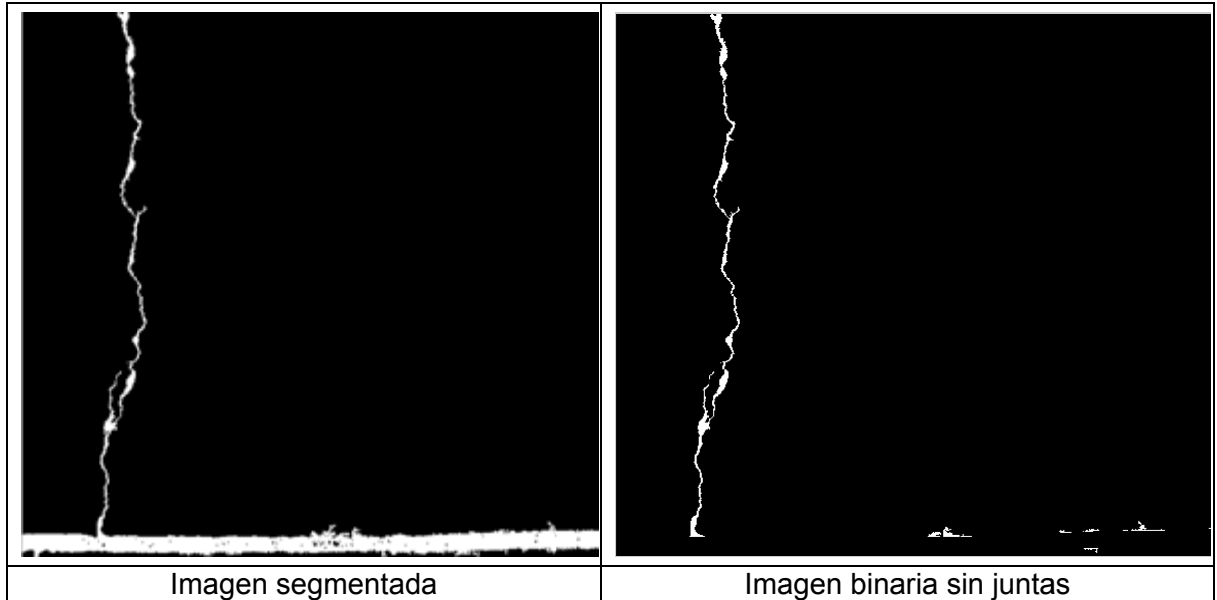
Figura 32. Rectángulos de búsqueda empleados en el algoritmo para unir regiones según la orientación de la región base.



6. Fuente: Los autores.

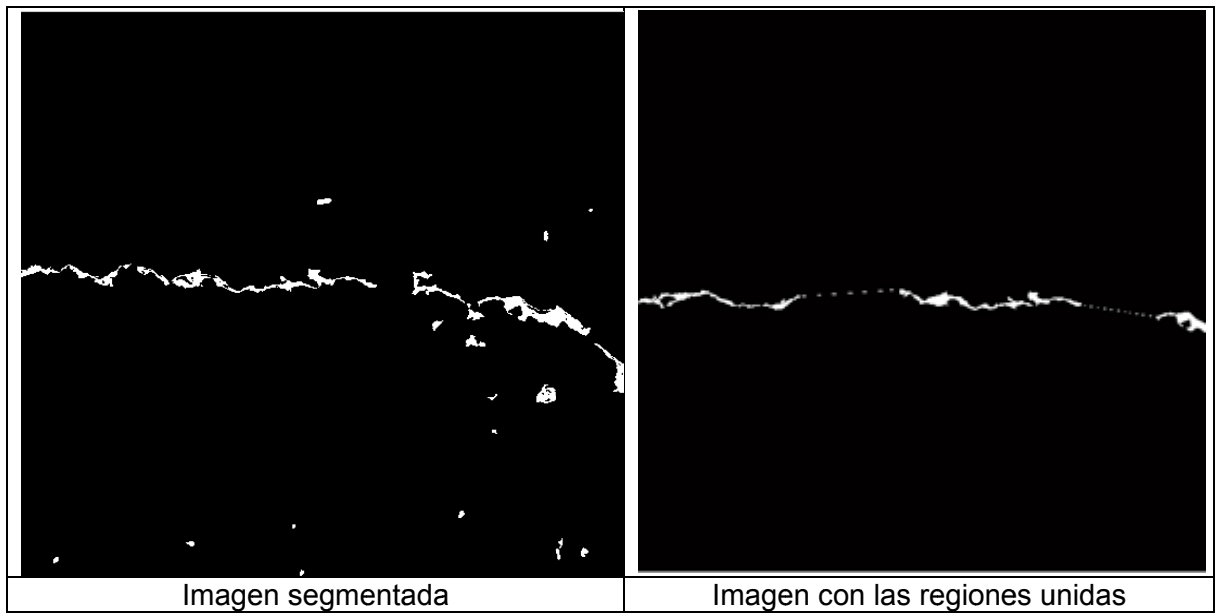
Cabe resaltar que se definió como orientaciones semejantes aquellas cuya diferencia relativa fuese menor a  $30^\circ$ , para evitar unir fisuras longitudinales con fisuras transversales o de borde y viceversa. Los resultados obtenidos con el algoritmo de eliminar las juntas de las losas y unir las regiones se presentan, respectivamente, en las figuras 33 y 34. Adicionalmente, el código en Matlab del algoritmo para unir regiones se encuentra disponible en el Anexo B con el nombre de **uregiones**.

Figura 33. Implementación del algoritmo para eliminar juntas de losas y otras regiones.



Fuente: Los autores.

Figura 34. Implementación del algoritmo para unir regiones.



Fuente: Los autores.

## **5. CARACTERIZACIÓN DE LAS FISURAS EN EL PAVIMENTO RÍGIDO**

### **5.1 CLASES DE FISURAS EN EL PAVIMENTO RÍGIDO**

Para el presente proyecto se definieron cuatro clases de fisuras, con base en lo establecido por el en el “Manual de inspección visual para pavimentos rígidos” (INVIAS-Universidad Nacional de Colombia 2006). Los cuatro tipos o clases de fisura fueron: fisuras longitudinales, fisuras transversales, fisuras de borde y fisuras de bloque. Adicionalmente, se estableció una quinta clase denominada como “otros objetos”, con el fin de tener una categoría para clasificar los patrones que no fuesen fisuras y que no se hubiesen eliminado en el procedimiento de segmentación, o aquellas fisuras que por medir menos de una pulgada no se consideran un daño considerable en la superficie del pavimento.

Después de haber definido las cinco clases, se clasificó en solo una de ellas a la totalidad de los patrones obtenidos tras haber segmentado las 175 imágenes recolectadas. Del total de 611 patrones 13 se clasificaron como fisuras de borde, 144 como fisuras longitudinales, 16 como fisuras de bloque, 56 como fisuras transversales y 382 como otros objetos.

### **5.2 DESCRIPTORES DE LAS FISURAS EN EL PAVIMENTO RÍGIDO**

En busca de establecer los descriptores que caracterizaran las fisuras, se planteó, en un comienzo, el uso de descriptores de forma y de textura, sin embargo, tras un estudio de los descriptores de textura se encontró que los valores de correlación, homogeneidad, varianza, etc., no permitieron discriminar las cinco clases de fisuras en las imágenes segmentadas, por lo que se descartó el uso de descriptores de textura y solo se recurrió al uso de descriptores de forma.

Según Theodoridis y Koutroumbas (2009), los descriptores seleccionados para el reconocimiento de patrones deben ser sencillos, ser pocos y su combinación debe ofrecer una gran capacidad de poder discriminar entre las diferentes clases, con el fin de que el clasificador empleado tenga grandes capacidades de generalización. Atendiendo a esta guía, se evaluó la capacidad de discriminación y la independencia de ocho descriptores de forma para regiones – área, perímetro, eje mayor, eje menor, extensión, excentricidad, elongación, y orientación-, de los cuales solo se optó por emplear tres, por ser los que presentaron valores más distantes entre clases y una menor varianza en cada clase, y porque su combinación era la que caracterizaba de forma más sencilla y correcta las

cinco clases ;los tres descriptores seleccionados fueron la extensión, la elongación y la orientación. En la Tabla 2 se presenta los resultados de un análisis estadístico descriptivo realizado para estimar la media, desviación estándar y rango de cada descriptor en cada clase de fisura planteada, los resultados obtenidos permiten concluir que los valores de los descriptores seleccionados, presentan grandes diferencias entre clases y poca varianza dentro de cada clase. Además, a continuación, se señala el aporte cualitativo de cada descriptor en la discriminación de las distintas clases:

- Extensión: permite diferenciar entre patrones asociados a una fisura (menor extensión) y los que no son fisuras o son fisuras muy cortas (mayor extensión).
- Elongación: distingue entre patrones de fisuras de bloque (menor elongación) y las demás fisuras (mayor elongación).
- Orientación: discrimina entre los patrones de las fisuras transversales, fisuras de borde y fisuras longitudinales.

Tabla 2. Comportamiento de los descriptores en cada clase.

<b>Clase</b>	<b>Descriptor</b>	<b>Media</b>	<b>Desviación estándar</b>	<b>Rango</b>
Fisura de borde	Extensión	0,06	0,03	0,13
Fisura de borde	Elongación	9,42	3,13	10,55
Fisura de borde	Orientación	48,95	19,20	40,63
Fisura longitudinal	Extensión	0,09	0,07	0,15
Fisura longitudinal	Elongación	12,61	6,60	35,06
Fisura longitudinal	Orientación	79,73°	10,90°	35,67°
Fisura de bloque	Extensión	0,03	0,01	0,03
Fisura de bloque	Elongación	3,60	2,22	5,96
Fisura de bloque	Orientación	35,99°	32,96°	80,25°
Fisura transversal	Extensión	0,08	0,07	0,11
Fisura transversal	Elongación	9,64	4,74	19,45
Fisura transversal	Orientación	12,41°	14,19°	29,32°
Otros objetos	Extensión	0,28	0,09	0,44
Otros objetos	Elongación	5,47	4,54	50,09
Otros objetos	Orientación	45,04°	29,79°	89,60°
Fuente: Los autores.				

## 6. CLASIFICACIÓN DE FISURAS EN EL PAVIMENTO RÍGIDO

Después de haber seleccionado los descriptores para caracterizar los patrones se debió elegir un clasificador que reconociera y clasificara las fisuras. Como se mencionó en el estado del arte, los clasificadores que se emplean comúnmente en el reconocimiento de fisuras son los algoritmos genéticos, las máquinas de soporte vectorial y las redes neuronales perceptrón multicapa; adicionalmente, del estudio adelantado por Rababaah, H. (2005) se concluyó que el mejor clasificador supervisado era la RNA perceptrón multicapa, razón por la cual éste último fue el algoritmo desarrollado.

### 6.1 ARQUITECTURA DE LA RED NEURONAL

La efectividad de una red neuronal perceptrón multicapa está condicionada a la arquitectura de la red, adicionalmente, dicha arquitectura puede tener infinitas configuraciones y no existe una metodología que determine la solución óptima. Por lo tanto, el problema de desarrollar la red óptima para el reconocimiento de fisuras se realizó mediante ensayo y verificación, seleccionando entre decenas de configuraciones probadas, las cinco mejores arquitecturas de acuerdo al porcentaje de acierto obtenido en la etapa de entrenamiento, y posteriormente, de éstas cinco configuraciones, se seleccionó la arquitectura que presentó el mayor acierto en la etapa de evaluación.

En general, las cinco redes neuronales tenían tres neuronas en la capa de entrada (una para cada descriptor) y cinco neuronas en la capa de salida (una para cada clase), diferenciándose únicamente las arquitecturas en el número de capas ocultas y las neuronas en dichas capas. A continuación, se presentan las cinco configuraciones planteadas, empleando la notación [3 x1 x2... 5] para representar las distintas arquitecturas, donde el primer elemento es el número de neuronas en la capa de entrada, x1 es el número de neuronas en la capa oculta número uno, x2 el número de neuronas en la capa oculta número dos, etc., y el último elemento es el número de neuronas de la capa de salida.

Tabla 3. Configuraciones de las redes neuronales evaluadas

Red neuronal	1	2	3	4	5
Arquitectura	[3 8 5]	[3 6 10 5]	[3 7 8 5]	[3 5 3 5]	[3 4 5]
Fuente: Los autores.					

## 6.2 ENTRENAMIENTO DE LAS REDES NEURONALES

Cada una de las redes neuronales planteadas fue entrenada en Matlab con 96 de los 611 patrones obtenidos en la segmentación. El número de patrones para el entrenamiento se definió teniendo en cuenta dos aspectos: primero, que únicamente se tenían 13 patrones para las fisuras de borde, lo que constituía una limitante en la cantidad de patrones a emplear de las demás clases, en aras de que la red neuronal no aprendiera a reconocer unos patrones más que otros, y segundo, que los patrones seleccionados para cada clase debían representar la variación de los descriptores en dicha clase, de forma tal que no se afectará la capacidad de generalización del algoritmo al reducir el número de patrones en el entrenamiento. En definitiva, para el entrenamiento de la red se emplearon 13 patrones de borde, 22 longitudinales, 15 de bloque, 21 transversales y otros 25 de otros objetos.

Por otro lado, para entrenar la red se optó por emplear el algoritmo *backpropagation* de Levenberg-Marquardt, el cual se caracteriza por emplear el Jacobiano para ajustar los pesos de la red, generar redes con un alto rendimiento durante su entrenamiento y por ser el algoritmo más rápido de los disponible en Matlab para el entrenamiento de las RNA<sup>36</sup>. Adicionalmente, para el cálculo del error se empleó la función de la media del error cuadrado por ser la que conlleva a una convergencia más rápida de la red. Finalmente, de los 96 patrones seleccionados, 60% se empleó para la etapa de entrenamiento, 20% para la etapa de validación y 20% para la etapa de evaluación, la selección de qué patrones conformaban la etapa de entrenamiento, validación o evaluación la realizó aleatoriamente Matlab.

## 6.3 RESULTADOS DEL ENTRENAMIENTO DE LAS REDES NEURONALES

El entrenamiento de las cinco RNA propuestas permitió determinar el rendimiento alcanzado por cada una de ellas, expresado en términos del número de iteraciones requeridas para el entrenamiento (épocas), el tiempo que tomó el entrenamiento y el menor valor del error cuadrado medio (MSE), como se muestra en la Tabla 4. Además, se obtuvo el porcentaje de acierto global (para todas las clases de fisuras) de cada clasificador en cada etapa del entrenamiento, presentado en la Tabla 5. A partir de la información consolidada en las dos tablas, se seleccionó a la configuración [3 6 10 5] como la indicada para ser empleada en el algoritmo de reconocimiento de fisuras en el pavimento rígido, por ser la que presentó mayor porcentaje de acierto en el reconocimiento de patrones a lo largo de todo el entrenamiento (95,8%), y en especial, en la etapa de evaluación de la red (94,7%), además, de haber sido la red que presentó el menor error cuadrado medio durante su entrenamiento (0,0084).

---

<sup>36</sup>MathWorks. Ayuda del producto para Matlab 7.9.0. Función *trainlm*

Tabla 4. Parámetros del progreso de la red neuronal durante el entrenamiento.

Parámetros	Red 1	Red 2	Red 3	Red 4	Red 5
Arquitectura	[3 8 5]	[3 6 10 5]	[3 7 8 5]	[3 5 3 5]	[3 4 5]
Épocas	15	14	14	14	11
Tiempo [s]	< 1	<1	1	<1	<1
Menor MSE	0,0120	0,0084	0,027	0,0449	0,0157
Gradiente del error	0,011 e <sup>-10</sup>	0,00012 e <sup>-10</sup>	0,004 e <sup>-10</sup>	0,034 e <sup>-10</sup>	0,00057 e <sup>-10</sup>
Validaciones	6	6	6	6	6

Fuente: Los autores.

Tabla 5. Resultados globales del entrenamiento de las redes neuronales.

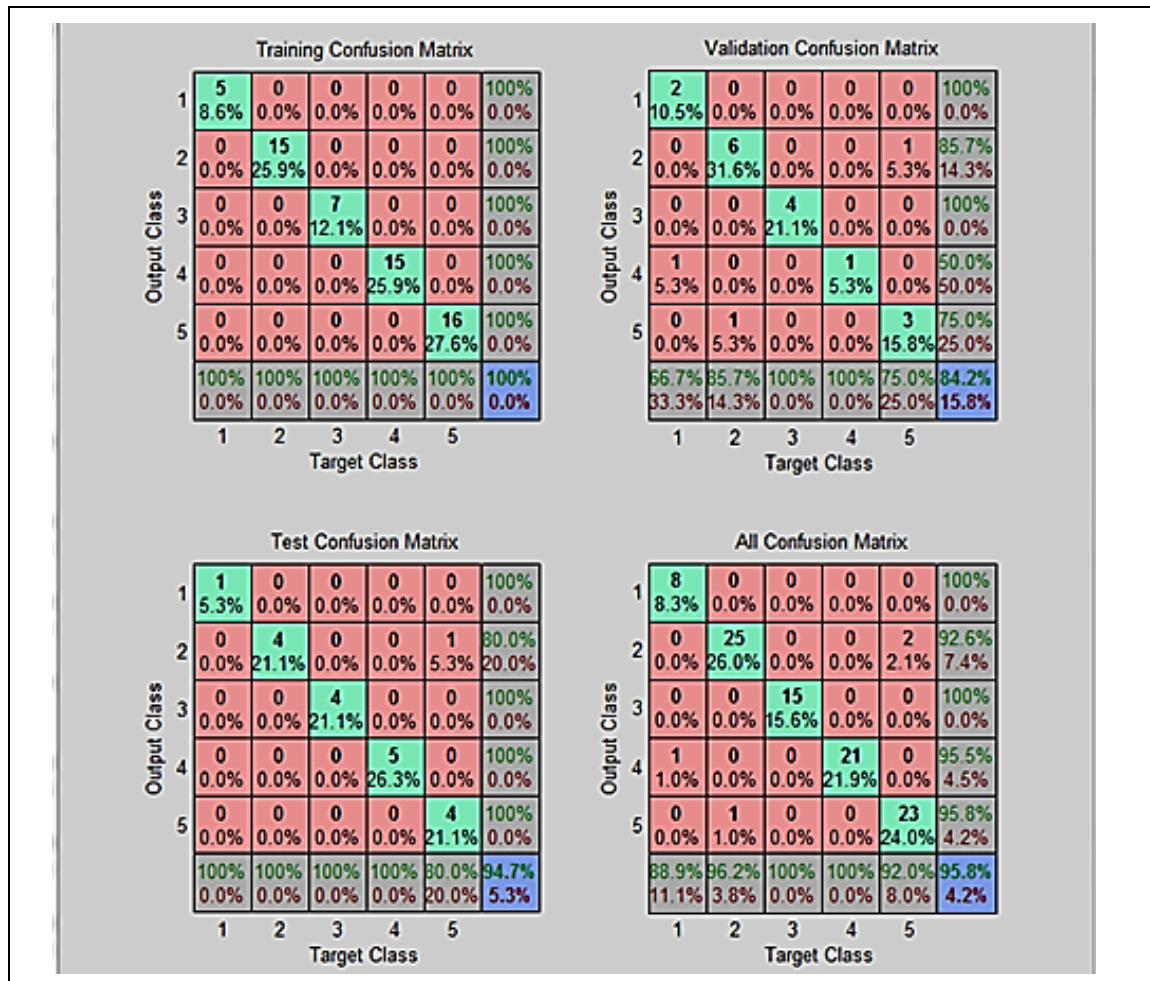
Acierto por etapas (%)	Red 1	Red 2	Red 3	Red 4	Red 5
Entrenamiento	98,3	<b>100</b>	96,6	91,4	98,3
Validación	94,7	<b>84,2</b>	73,7	57,9	84,2
Evaluación	78,9	<b>94,7</b>	94,7	94,7	94,7
Total	93,8	<b>95,8</b>	91,7	85,4	94,8

Fuente: Los autores.

Adicionalmente, a partir del entrenamiento de la RNA con configuración [3 6 10 5] se estableció el porcentaje de acierto obtenido en cada etapa del entrenamiento para cada clase de fisura, información recopilada en la Figura 35 bajo el nombre de Gráfica de confusión. Ésta gráfica presenta en la diagonal principal de cada matriz (una para cada etapa del entrenamiento), el número de patrones clasificados correctamente para cada clase y el porcentaje que representan estos patrones del total de su clase empleados en el entrenamiento. Asimismo, en la fila inferior de cada matriz se señalan los porcentajes de acierto (verde) y error (rojo) para cada clase. En el presente trabajo se empleó la siguiente notación: uno (1) para fisuras de borde, dos (2) para fisuras longitudinales, tres (3) para fisuras de bloque, cuatro (4) para fisuras transversales y cinco (5) para otros objetos.

De la interpretación de la gráfica de confusión, se resalta que en la etapa de evaluación de la RNA perceptrón multicapa [3 6 10 5] se obtuvo un 100% de eficacia a la hora de clasificar los patrones de fisuras de borde, longitudinales, transversales y de borde, mientras que para los patrones asociados a otros objetos solo se obtuvo el 80% de acierto, clasificando, erróneamente, el 20% restante como fisuras longitudinales.

Figura 35. Gráfica de confusión de la red neuronal [3 6 10 5]



Fuente: Los autores.

## 7. VALIDACIÓN DEL ALGORITMO PROPUESTO

Para evaluar la capacidad de generalización del algoritmo planteado, se realizó la validación del algoritmo determinando el porcentaje de acierto y de error obtenido por el clasificador desarrollado, para cada clase y en dos conjuntos de patrones: uno correspondiente a los patrones de las imágenes que se recopilaban siguiendo la metodología propuesta y que no fueron empleados en ninguna etapa del entrenamiento de la red, y otro correspondiente a los patrones identificados en imágenes que se recopilaban en tres escenarios de captura por fuera de la metodología establecida.

En general, se establecieron dos tipos de errores: el patrón no era de la clase y se clasificó en ella (No es- Se clasificó) o el patrón era de la clase y no se clasificó en ella (Es- No se clasificó), y dos tipos de aciertos: el patrón era de la clase y se clasificó en ella (Es- Se clasificó) y no era de la clase y no se clasificó (No es- No se clasificó). Los resultados obtenidos de la validación son presentados en las Tabla 6 y 7, donde se indican - en porcentaje- los aciertos y errores de la red.

### 7.1 VALIDACIÓN DENTRO DE LA METODOLOGÍA PROPUESTA

Como se señaló anteriormente, primero se validó el algoritmo planteado para patrones identificados en imágenes que se recopilaban con la metodología propuesta y que no se emplearon en el entrenamiento de la red neuronal. Es importante destacar que dentro de éste conjunto de patrones, no se contó con patrones de borde ni de bloque, de allí que los porcentajes correspondientes de “Es- No se clasificó” y “Es- Se clasificó” en la Tabla 6 sean nulos.

Tabla 6. Resultados de la validación de la RNA dentro de la metodología propuesta.

Fisura Clasificación	De borde		Longitudinal		De bloque		Transversal		Otra región	
	Es	No es	Es	No es	Es	No es	Es	No es	Es	No es
<b>Se clasificó</b>	---	1%	85%	2%	---	3%	88%	13%	83%	8%
<b>No se clasificó</b>	---	99%	16%	98%	---	97%	9%	91%	18%	92%

Fuente: Los autores.

Los resultados obtenidos presentan un nivel de error general de clasificación del 16%, que evidencian la gran capacidad del clasificador para reconocer patrones desconocidos en imágenes capturadas según la metodología propuesta.

## 7.2 VALIDACIÓN DEL ALGORITMO EN TRES ESCENARIOS DE CAPTURA

Adicionalmente, se validó el desempeño del algoritmo para reconocer patrones en imágenes del pavimento rígido que se hubiesen tomado en escenarios de captura por fuera de la metodología propuesta. En particular, se definieron tres escenarios de captura: uno correspondiente a imágenes digitales donde el pavimento rígido estuviera en condiciones húmedas (Escenario 1), otro donde no hubiese fisuras en las losas de concreto (Escenario 2), y un último donde la superficie del pavimento estuviera desgastada o sucia (Escenario 3). Los porcentajes de acierto y error para cada tipo de fisura y en cada escenario se presentan en la Tabla 7.

Tabla 7. Resultados de la validación en distintos escenarios de captura.

Fisura Clasificación	De borde		Longitudinal		De bloque		Transversal		Otra región	
	Es	No es	Es	No es	Es	No es	Es	No es	Es	No es
Escenario 1										
<b>Se clasificó</b>	---	0%	60%	8%	---	0%	67%	0%	100%	25%
<b>No se clasificó</b>	--	100%	40%	92%	---	100%	33%	100%	0%	75%
Escenario 2										
<b>Se clasificó</b>	---	0%	---	4%	---	1%	---	3%	92%	---
<b>No se clasificó</b>	---	100%	---	96%	---	99%	---	97%	8%	---
Escenario 3										
<b>Se clasificó</b>	67%	1%	67%	1%	100	1%	75%	13%	80%	11%
<b>No se clasificó</b>	33%	99%	33%	99%	0%	99%	25%	87%	20%	89%
Fuente: Los autores.										

Los resultados obtenidos evidencian que el algoritmo propuesto es capaz de reconocer de forma correcta, aproximadamente, el 61% de los patrones identificados cuando la imagen se toma en superficies húmedas, el 98% en imágenes sin fisuras (se espera que en estas imágenes todos los patrones sean de la clase “Otros objetos”) y el 64% en imágenes donde el pavimento se encuentra desgastado; evidenciando la capacidad del algoritmo de reconocer patrones en imágenes tomadas por fuera de la metodología propuesta. Adicionalmente, dentro de los resultados obtenidos de esta validación se concluyó que el algoritmo tiene un desempeño satisfactorio sin importar el tamaño de la imagen a estudiar.

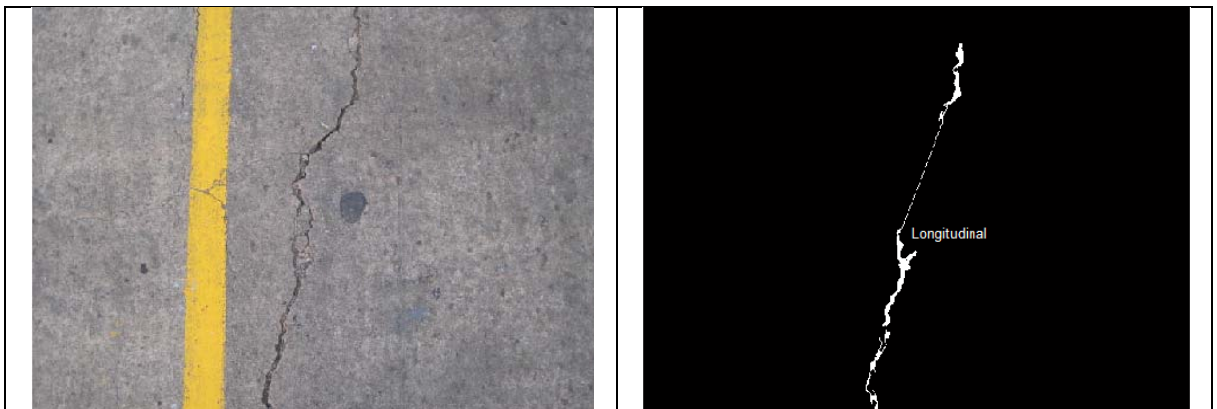
En las Figuras 36 a 38 se presentan tres resultados obtenidos con el algoritmo propuesto, una para cada escenario de captura validado

Figura 36. Imagen del pavimento sin fisuras



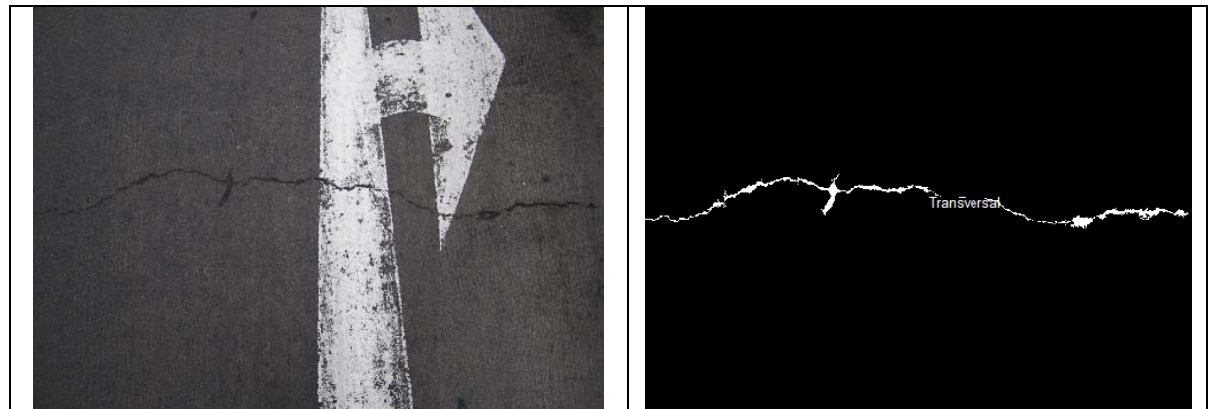
Fuente: Los autores.

Figura 37. Imagen de resolución de 480x640



Fuente: Los autores.

Figura 38. Imagen del pavimento en condiciones húmedas



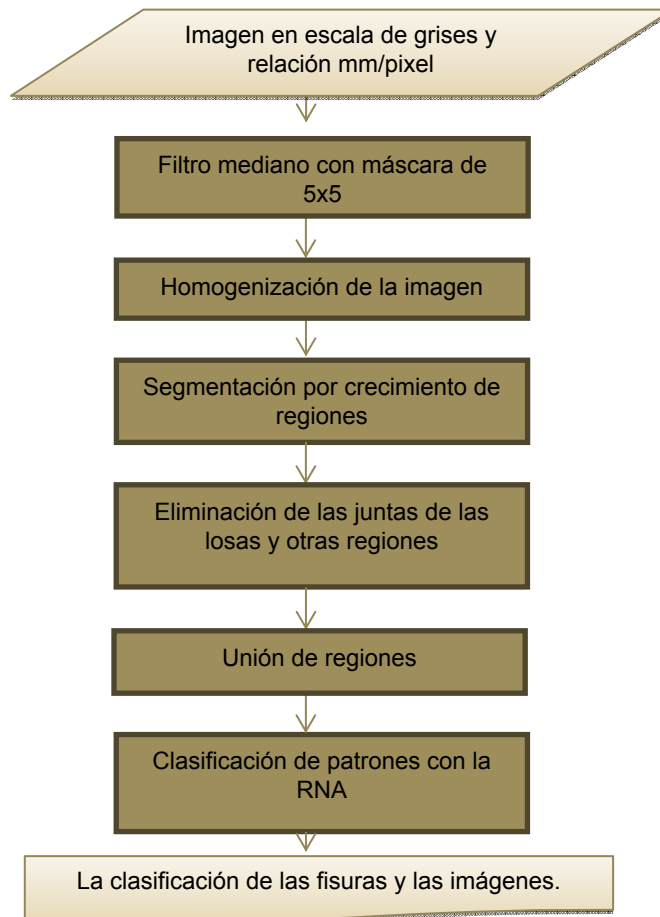
Fuente: Los autores.

## 8. ALGORITMO PARA EL RECONOCIMIENTO DE FISURAS EN PAVIMENTOS DE LOSA DE CONCRETO

La reunión de las técnicas de pre-procesamiento y segmentación implementadas y el clasificador desarrollado, componen el algoritmo planteado para el reconocimiento de fallas tipo fisuras en el pavimento rígido mediante imágenes digitales (denominado ALFIL). A continuación, se presenta el diagrama de flujo correspondiente, la evaluación del tiempo de ejecución del algoritmo una vez se implementó en Matlab, y la interfaz gráfica del usuario desarrollada para el algoritmo, así como los resultados obtenidos para cuatro imágenes digitales representativas. El código del algoritmo ALFIL está disponible en el Anexo B, mientras que la interfaz gráfica se encuentra en el Anexo C.

### 8.1 DIAGRAMA DE FLUJO DEL ALGORITMO PROPUESTO

Figura 39. Diagrama de flujo del algoritmo ALFIL.



Fuente: Los autores.

## 8.2 TIEMPO DE EJECUCIÓN DEL ALGORITMO

Al implementar el algoritmo planteado en el *software* Matlab, se pudo establecer el tiempo de ejecución de cada procedimiento empleado y el tiempo total de ejecución del algoritmo ALFIL. En la Tabla 6 se resumen los tiempos promedios para 10 imágenes de 3240x4320 y 10 imágenes de 480x640, medidos con dos computadores con características distintas; los dos equipos empleados se denominaron como Equipo 1 y Equipo 2. El Equipo 1 cuenta con un procesador AMD Phenom II X4 B95 de 3 GHz de frecuencia y cuatro núcleos, memoria RAM de 4 GB, memoria en caché de 6 GB y tarjeta de video ATI Radeon HD 4200; mientras que el Equipo 2 cuenta con un procesador Intel Core i5 M 450 de 2.4 GHz de frecuencia y cuatro núcleos, 4 GB de memoria RAM y 7,6 GB de memoria en caché y tarjeta de video Inter HD Graphics. Los resultados obtenidos permitieron concluir que el tiempo de ejecución del algoritmo planteado está en función del tamaño de la imagen y que el procedimiento que mayor demanda tiempo (independientemente del tamaño de la imagen y el equipo) es el de eliminar las juntas de las losas de concreto y otras regiones (aproximadamente el 50% en imágenes de 3240x4320 y el 80% en imágenes de 480x640).

Tabla 8. Tiempos de ejecución de los algoritmos en Matlab.

Tiempo de procedimiento [s]	Imagen de 3240x4320		Imagen de 480x640	
	Equipo 1	Equipo 2	Equipo 1	Equipo 2
<b>Filtrado</b>	1,53	6,01	0,04	0,13
<b>Homogeneizado</b>	0,62	0,81	0,01	0,02
<b>Segmentación</b>	0,42	1,35	0,01	0,03
<b>Eliminación de juntas</b>	5,52	9,72	0,42	1,23
<b>Unión de regiones</b>	1,54	3,42	0,02	0,06
<b>Clasificación</b>	0,50	0,77	0,01	0,03
<b>Total</b>	10,13	22,08	0,51	1,5

Fuente: Los autores.

## 8.3 INTERFAZ GRÁFICA DEL USUARIO ALFIL

Finalmente, se desarrolló una interfaz gráfica de usuario para el algoritmo ALFIL, la cual, además de reconocer las fisuras, presenta en una imagen la clasificación realizada, calcula el área mínima de reparación de la fisura - obtenida con la propiedad *Bounding Box*-y estima cuanto representa esta área en porcentaje del área total de la imagen. La interfaz desarrollada carga la imagen indicada por el usuario, realiza el reconocimiento de las fisuras según los parámetros y procedimientos establecidos por el usuario, y permite almacenar la imagen con la clasificación y un archivo de texto con los descriptores calculados, la clasificación y las áreas y los porcentajes de reparación de cada patrón clasificado como fisura. En la Figura 40 se presenta la interfaz desarrollada - cuyo manual puede consultarse en el Anexo C-, mientras que en las Figuras 41 a 44 se presentan la

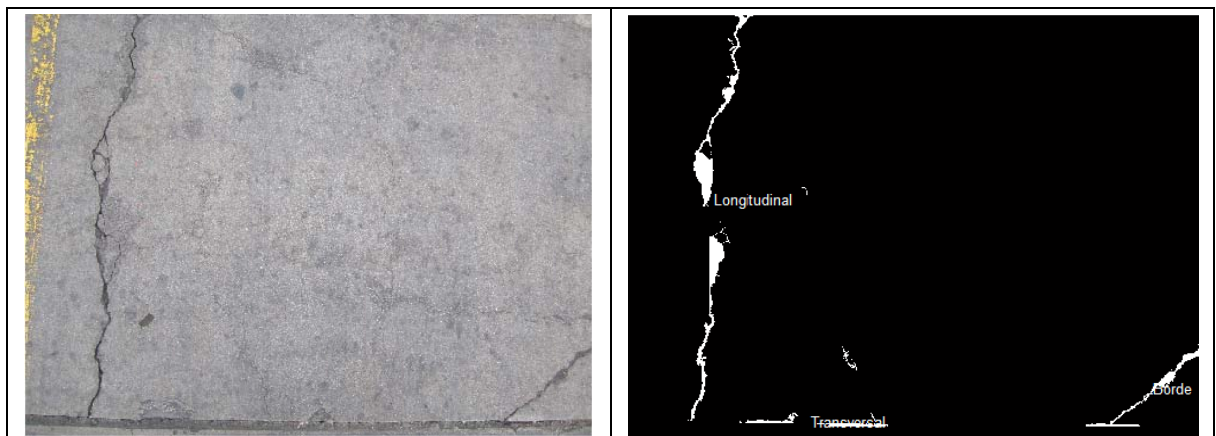
clasificación de cuatro imágenes representativas de las 200 imágenes recolectadas bajo la metodología propuesta.

Figura 40. Interfaz gráfica del usuario para el algoritmo ALFIL.



Fuente: Los autores.

Figura 41. Fisura longitudinal y de borde.



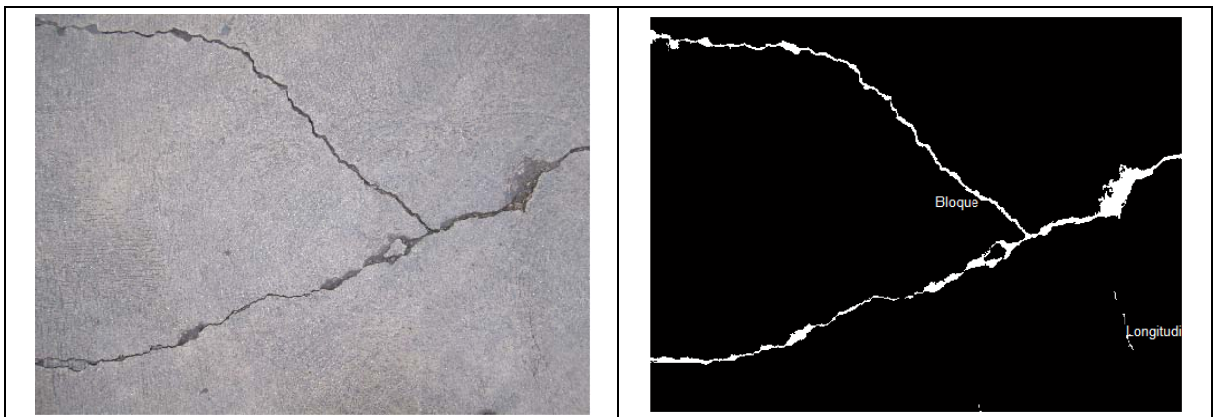
Fuente: Los autores.

Figura 42. Fisura de bloque y transversal.



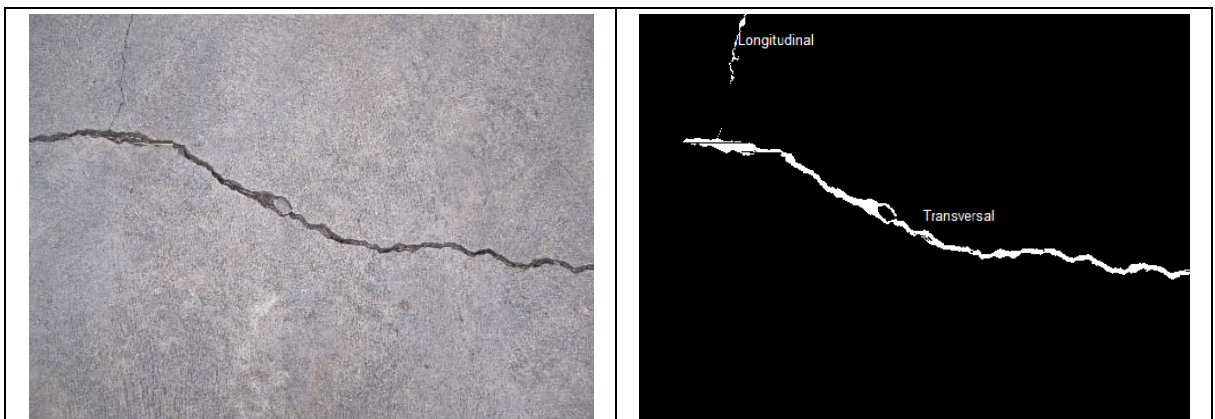
Fuente: Los autores.

Figura 43. Fisura de bloque y longitudinal.



Fuente: Los autores.

Figura 44. Fisura longitudinal y transversal.



Fuente: Los autores.

## 9. CONCLUSIONES

Al finalizar el proyecto se concluyó que el algoritmo desarrollado para el reconocimiento de fisuras en pavimento rígido mediante el tratamiento de imágenes digitales, constituye una propuesta innovadora en su tipo, ya que segmenta, describe y clasifica dichas fallas tras representarlas como regiones dentro de las imágenes digitales. El algoritmo propuesto - denominado como ALFIL-permite identificar las fisuras presentes en la superficie del pavimento en losas de concreto en condiciones heterogéneas de iluminación, clasificarlas en cinco categorías según su morfología, y medir el área mínima a restaurar en el pavimento rígido; sentando las bases para la creación de un sistema automatizado de evaluación del estado del pavimento en vías urbanas, y reducir así los costos y tiempos de los planes de gestión de pavimentos.

Tras revisar las distintas metodologías presentes en la literatura para la recolección de las imágenes digitales que presentan fisuras en el pavimento, se estableció un procedimiento manual para capturar las imágenes, acorde con las posibilidades económicas y el alcance del presente proyecto. Las imágenes digitales obtenidas con esta metodología presentaron un grado de detalle de las fisuras suficiente para su identificación mediante técnicas de tratamiento digital de imágenes, pese a que la iluminación de las mismas no fue homogénea, aun cuando se siguieron todas las recomendaciones establecidas en la literatura.

A partir de las técnicas de pre-procesamiento empleadas se logró eliminar el ruido de la imagen y se homogenizó la iluminación de las mismas, facilitando la diferenciación de las fisuras con respecto a los demás objetos presentes en las imágenes durante el procedimiento de segmentación. Al respecto, la implementación de la técnica de crecimiento por regiones permitió identificar las fisuras de forma rápida, precisa y automática – este último aspecto la convirtió en la herramienta más adecuada para ser incluida en el algoritmo ALFIL-, representando a los objetos segmentados como regiones. Posteriormente, como resultado de emplear la técnica de eliminación de las juntas de las losas y otras regiones, se obtuvieron imágenes binarias sin regiones cuya forma discrepara de la de una fisura, mientras que con el procedimiento de unir regiones, se logró la conexión de segmentos de fisuras que correspondían a una única falla, resultando en regiones más fáciles de describir y clasificar.

Como resultado de la caracterización de las fisuras en el pavimento rígido, se lograron establecer cinco categorías de fisuras que agruparon la totalidad de los tipos de grietas capturados en las imágenes digitales, y en las cuales se clasificaron las regiones segmentadas. Por otro lado, se concluyó que con solo tres descriptores de forma era suficiente para caracterizar de manera sencilla, pero con un alto grado de discriminación, cada una de las cinco categorías definidas.

Del desarrollo del clasificador para el algoritmo ALFIL, se concluyó que el mejor algoritmo para el reconocimiento de fisuras en el pavimento rígido correspondió a una red neuronal perceptrón multicapa con configuración [3 6 10 5] entrenada bajo el algoritmo *backpropagation* de Levenberg-Marquardt. Dicha red obtuvo un porcentaje global de acierto del 95,8% durante su etapa de entrenamiento y presentó un excelente

desempeño para clasificar cualquier tipo de fisura, esto último fue posible gracias a que durante la etapa de aprendizaje de la red neuronal se seleccionó un número semejante de patrones en cada clase, evitando que la red aprendiera a reconocer unos tipos de patrones más que otros.

Para validar el algoritmo planteado se determinaron los porcentajes de error y de acierto del clasificador en dos situaciones: con imágenes tomadas bajo la metodología propuesta y que no fueron empleadas en el entrenamiento, y otra con imágenes tomadas en tres escenarios de captura por fuera de la metodología planteada. En el primer caso, se concluyó que el algoritmo presentó un nivel de acierto del 85% y el 88% para fisuras longitudinales y transversales, respectivamente, y del 83% para otros objetos, lo que constituye un porcentaje de acierto global del 84%, un resultado que es altamente satisfactorio si se comparan con los reseñados en la literatura. En el segundo caso, se obtuvo un porcentaje de acierto general del 61% para imágenes en condiciones húmedas del pavimento (Escenario 1), del 98% para imágenes del pavimento rígido sin fisuras (Escenario 2), y del 64% para imágenes de fisuras tomadas sobre pavimentos con desgaste superficial (Escenario 3).

Los resultados obtenidos durante el entrenamiento y la validación del clasificador evidencian una gran capacidad de generalización y eficacia del algoritmo ALFIL para el reconocimiento de fisuras en el pavimento rígido en imágenes digitales tomadas según la metodología de recolección propuesta, y aun en imágenes capturadas por fuera de dicha metodología, para las cuales no fue diseñado el algoritmo. Finalmente, de la implementación del algoritmo ALFIL en el *software* Matlab, se pudo determinar que, para el tamaño de imagen empleado en la metodología, el tiempo de ejecución del algoritmo es de aproximadamente 15 segundos, del cual aproximadamente el 50% es empleado por el algoritmo para eliminar las juntas de las losas y otras regiones. El tiempo de ejecución y la eficacia del algoritmo ALFIL, sumado a sus capacidades de identificar, clasificar y medir las fisuras en el pavimento rígido de forma automática, permiten concluir que el algoritmo propuesto puede ser implementado en cualquier sistema semi-automático de evaluación del estado del pavimento en vías urbanas.

## 10. RECOMENDACIONES

A modo de recomendación para futuros trabajos en la materia se plantea, en primer lugar, la implementación de un sistema semi-automático para recolectar las imágenes digitales semejante al propuesto en el desarrollo de este trabajo, con el fin ampliar las capacidades del algoritmo propuesto, las cuales se encuentran en su mayoría restringidas por la metodología de captura empleada; incluyendo casos tales como el reconocimiento de fisuras en imágenes donde el pavimento esté en condiciones húmedas, en donde el pavimento presente desgaste superficial y donde se registren dos tipos de pavimentos en una misma imagen.

Adicionalmente, con el ánimo de incrementar la eficacia del algoritmo se propone el uso de al menos otro descriptor que permita discernir de una mejor forma entre la categoría de borde y las demás categorías, dado que es en este aspecto donde se observó que el algoritmo ALFIL presenta la mayor deficiencia a la hora de clasificar los patrones.

Por otro lado, considerando que la mayor parte del tiempo de ejecución del algoritmo ALFIL se emplea en el procedimiento para eliminar las juntas, se propone el desarrollo de otro procedimiento que demande menos tiempo pero que continúe eliminando las juntas de las losas de concreto como lo realiza el algoritmo desarrollado. Asimismo, dado que se presentan los códigos desarrollados en Matlab se espera que estos puedan ser optimizados para obtener resultados más precisos y en menor tiempo, y que resulten cada vez más comprensibles para el usuario.

Finalmente, se exhorta a las futuras investigaciones a proponer otros algoritmos para el reconocimiento de nuevos tipos de fallas en los distintos tipos de pavimentos, con el fin de contribuir a conformar un sistema automatizado de evaluación del estado del pavimento en vías urbanas.

## BIBLIOGRAFÍA

AEPO. Proyecto EVASIVA: equipo de video auscultación. Sistema de inspección alternativo. Descripción del equipo evasiva. Memoria Técnica. 11p.

Alcaldía de Bucaramanga – Universidad Industrial de Santander. Modelamiento para la gestión del mantenimiento, rehabilitación y mejoramiento de pavimentos urbanos. Bucaramanga: La Alcaldía - UIS, 2011. 63p.

Ali, Mohamed y Clausi, David. Using the canny edge detector for feature extraction and enhancement of remote sensing images. En: IEEE. 2001. p. 2298-2300.

Altamirano, Kauffmann, Luis F. Deterioro de pavimentos rígidos: metodología de medición, posibles causas de deterioro y reparaciones. Universidad Nacional de Ingeniería, 2007. 91p.

Austrroads. Review of specification for the collection of Digital Road Imaging. En: Austrroads Technical Report no. AP-T169-10. Sydney: Austrroads, 2010. 41p.

Consejo de directores de carreteras de Iberia e Iberoamérica. Catálogo de deterioros de pavimentos rígidos. DIRCAIBEA. 2002. 25p.

Cline, Gregory D., Shahin, Mohamed Y. and Burkhalter, Jeffrey A. Automated data collection for pavement condition index survey.[enlínea] disponible en: <http://pavement.wes.army.mil/papers/18/PAPER18.PDF> (julio 2011).

Chou, JaChing, Wende, O'Neill y Cheng, H.D. Pavement distress classification using neural networks. En: IEEE-94, 0-7803-2129-4/94. 1994.

Gonzalez, Rafael C., Woods, Richard E. y Eddins, Steven L. Digital Image Processing using Matlab. 2 ed. New York: Addison – Wiley, 2009. 597p.

Haralick, Robert M., Shanmugam, K. y Dinstein, ItsHak. Textural features for image classification. En: IEEE Transactions on Systems Man and Cybernetics. Nov. 1973. SMC 3, no. 6; p. 610-621.

Hsu, Chung-Jung, Chen, Chai-Fam, Lee, Chau y Huang, Shu-Meng. Airport pavement distress image classification using moment invariant neural network. 22<sup>nd</sup> Asian Conference on Remote Sensing and Processing ( CRISP), Singapore. 2001.

Hu, Yong, Zhao, Chun-xia y Wang, Hong-nan. Automatic pavement crack detection using texture and shape descriptors. En: IETE Technical Review. (Sep – Oct. 2010); Vol. 27, no. 5; p. 398-405. [Enlínea] disponible en: [www.tr.letejournals.org](http://www.tr.letejournals.org) (2010: oct. 27).

INVIAS – UNIVERSIDAD NACIONAL DE COLOMBIA. Manual de inspección visual para pavimentos rígidos. Estudio e investigación del estado actual de las obras de la red nacional de carreteras [convenio interadministrativo 0587-03], Octubre 2006. 65p.

Jing, Lou y Aiqin, Zang. Pavement crack distress detection based on image analysis (2010: International Conference on Machine Vision and Human Machine Interface); p. 576-579.

Kaseko.Mohamed S. y Ritchie Stephen G. A neural network based methodology for pavement crack detection and classification. En: Transp. Res. C. 1993, Vol. 1, no. 4; p.275-291.

Koutsopoulus, Haris N., El Sanhoury, Ibrahim y Downey, Allen B. Analysis of segmentation algorithms for pavement distress images.ASCE, 2011.900 p.

Lee, Byoung y Lee, Hosin D. A robust position invariant artificial neural network for digital pavement crack analysis.Technical Report. TRB 2003-000996.

Liu, Fanfan y otros. Novel approach to pavement cracking automatic detection based on segment extending. (2008: International Symposium on Knowledge Acquisition and Modeling) en: IEEE Computer Society 2008; p. 610-614.

Méndez, E. Mariño, J. Sistema automático de entrenamiento de redes neuronales artificiales basado en el ajuste genético de parámetros y variación de arquitectura. [trabajo de grado] Bucaramanga: UIS, 2008. 2 CD.

NATIONAL. COOPERATIVE. HIGHWAY. RESEARCH. PROGRAM. Automated pavement distress collection techniques. Synthesis 334.Washington: NCHRP, 2004. 85p.

Oh, Hyungkee, Garrick, Norman W.,Achenie, Luke E. K. Automated pavement evaluation system for pavement distress assessment. University of Connecticut. JHRAC, 1997. 50 p.

OMC. Informe sobre el comercio mundial 2004. II Coherencia. Capítulo B La infraestructura en el comercio y el desarrollo económico. Berna: OMC, 2004. p. 124 – 161.

Ouyang, Aiguo. Luo, Chagen, Zhou, Chao. Surface distress detection of pavement based on digital image processing. En: Computer and computing technologies in agriculture IV. Nanchang: IFIP, 2010. p. 368-374.

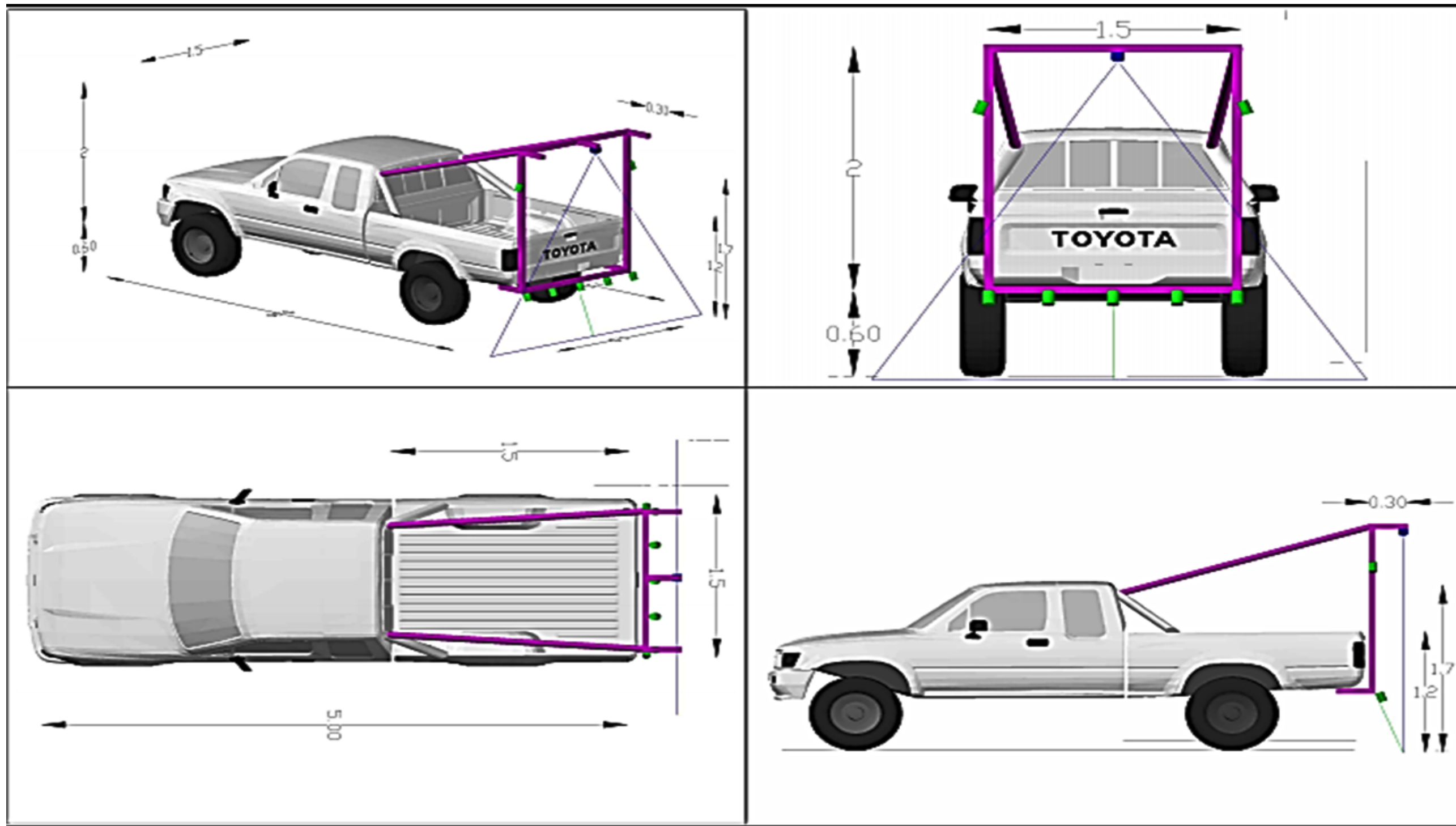
Rababaah, Haroun. Asphalt pavement crack classification: a comparative study of three AI approaches: multilayer perceptrón genetic algorithms and self-organizingmaps [maestría en ciencias] .Indiana : Indiana University South Bend, 2005. 135p.

Russ, J. The image processing handbook.5ª Edición.Broken Sound Parkway: CRC, 2008.832p.

Secretaria de Integración Económica Centroamericana. Manual centroamericano para diseño de pavimentos. 2002. 109p.

- Theodoridis, Sergios y Koutroumbas, Konstantinos. *Pattern Recognition*. 4 ed. London: Academic Press, 2009. 967p.
- Tsai, Yi Chang, Kaul, Vivek y Mersereau, Russell M. Critical assement of pavement distress segmentation methods. En: *Journal of Transportation Engineering ASCE*. No. 1, 2010. P. 11-19.
- U.S. Department of Transportation, Federal Highway Administration. *Distress identification manual for the long-term pavement performance program*. Washington: FHWA, 2003. 155p.
- Vasquez Varela, Luis Ricardo. *PavementConditionIndex (PCI): para pavimentos asfálticos y de concreto en carreteras*. Manizalez: Ingepav, 2002. 90p.
- Wang, Kelvin y Elliot, Robert. *Investigation of image archiving for pavement surface distress survey*. University of Arkansas, Department of Civil Engineering Fayetteville. 1999.
- Wang, Kelvin, Quintin, We-Yih T. y Kuchikulla, Subash. *Digital distress survey of airport pavement surface*. Federal Aviation Administration Airport Technology Transfer Conference. 2002.
- Wang, Kelvin, C.P. y otros. *Automated imaging technique for runway condition survey*. (FAA Worldwide Airport Technology Transfer Conference: Apr. 2007: Atlantic City, NJ). 10p.
- Xu, Guoai y otros. *Automatic recognition of pavement surface crack based on BP neural network*. (2008: International Conference on Computer Electrical Engineering). P. 19-22.
- Ying, L., Salari, E., Beamlet. *Transform based technique for pavement. Image Processing and Classification*, Department of Electrical Engineering and Computer Science, 2009, 141-145.
- Zahoyun, Sun, Chaofan, Wang y Aimin, Sha. *Study of image based pavement cracking measurement techniques*. (2009: 9: International Conference on Electronic Measurement and Instruments: IEEE); p. 140 – 143.
- Zhou, Huilin, Yang, Zhu y Zhu Jian. *Illumination invariant enhancement and threshold segmentation algorithm for asphalt pavement crack image*. En: *IEEE* (2010); 4p.

AnexoA.Esquema gráfico del sistema semi-automático propuesto para la recolección de las imágenes digitales



## AnexoB.Códigos en Matlab de los algoritmos propuestos

### Algoritmo para el reconocimiento de fisuras en pavimentos de losas de concreto (ALFIL)

```
function [ImágenesClasificación]=alfil(I,r,varargin)
%ALFIL permite clasificar y cuantificar las fisuras en una imagen a color
%del pavimento rígido.
%F=alfil(I,r,varargin) retorna las distintas imagen y la clasificación de las
regiones, ingresando la imagen a color (I), larelación mm/pixel de la imagen (r)
y los demás parámetros si es el caso.
Imágenes=struct;Clasificación=struct;
%Conversión a escala de grises y filtro mediano con una máscara de 5 x 5
if isempty(varargin)
Imágenes.Filtrada=medfilt2(rgb2gray(I), [5 5]);
else
Imágenes.Filtrada=medfilt2(rgb2gray(I), [varargin{1}(1) varargin{1}(1)]);
end
%Homogeneizar la imagen a escala de grises
if isempty(varargin) || numel(varargin{1})<3
Imágenes.Homogena=imhomogenea(Imágenes.Filtrada);
else
Imágenes.Homogena=imhomogenea(Imágenes.Filtrada,varargin{1}(2),varargin{1}(3));
end
%Segmentar la imagen por crecimiento de regiones.
if isempty(varargin) || numel(varargin{1})<5
Imágenes.Segmentada=cregiones(Imágenes.Homogena);
else
Imágenes.Segmentada=cregiones(Imágenes.Homogena,varargin{1}(4),varargin{1}(5));
end
%Eliminar las regiones correspondientes a juntas de losa u otros objetos
%que por su forma no se asemejan a una fisura.
[Imágenes.REliminadas,Imágenes.CCEliminadas]=eregiones(Imágenes.Segmentada,r);
%Unir las regiones que presenten igual orientación.
if isempty(varargin) || numel(varargin{1})<6
[Imágenes.RUnidas,Imágenes.CCUnidas]=uregiones(Imágenes.REliminadas,Imágenes.CCEl
iminadas);
else
[Imágenes.RUnidas,Imágenes.CCUnidas]=uregiones(Imágenes.REliminadas,...
Imágenes.CCEliminadas,varargin{1}(6));
end
%Obtener las características de las regiones y clasificarlas.
[Clasificación.SalidasClasificación.Fisuras.
Clasificación.Areas]=clasfisuras(Imágenes.CCUnidas,r);
```

### Algoritmo para homogeneizar la imagen (imhomogenea)

```
function G=imhomogenea(varargin)
% IMHOMOGENEA permite homogeneizar las intensidades de una imagen en esca
% la de grises, retornando la imagen homogeneizada siguiendo el algoritmo pl
% anteaado por Zhouet al.
% G=imhomogenea(varargin) retorna la imagen homogenea (G) a partir de la
% información suministrada en varargin. VARARGIN es un cellarray que puede
% contener: tan solo la imagen a homogeneizar (I) o I y las constantes
% A y f, en ese orden, según [1].
% REFERENCIAS:
% [1] Zhou, Huilin et al. Illumination invariant enhancement and threshold
```

```

% segmentation algorithm for asphalt pavement crack image.IEEE. 2010.
error(nargchk(1,3,nargin))
ifnargin==1
iptcheckinput(varargin{1},{'numeric'},{'real','nonempty','finite'},...
'func_name','var_name',2);
I=varargin{1};A=mean2(I);f=0.6;
else
iptcheckinput(varargin{1},{'numeric'},{'real','nonempty','finite'},...
'func_name','var_name',2);
I=varargin{1};A=varargin{2};f=varargin{3};
end
G=I;[r c]=size(I);paso=r/6;I(I>=A*ones(size(I)))=A;
whilemod(r,paso)~=0 || mod(c,paso)~=0
paso=round(paso)+1;
end
total=r*c/paso^2;k=1;media=ones(total,1);minimo=ones(total,1);
maximo=ones(total,1);Ls=ones(total,1);Li=ones(total,1);factor=ones(total,1);
mediamod=ones(total,1);Q=cell(total,1);
for i=1:paso:r;
for j=1:paso:c
Q{k}=I(i:i+paso-1,j:j+paso-1);
media(k)=mean2(Q{k});
minimo(k)=min(min(Q{k}));
maximo(k)=max(max(Q{k}));
Ls(k)=media(k)+(maximo(k)-media(k))*f;
Li(k)=media(k)-(media(k)-minimo(k))*f;
mediamod(k)=mean(Q{k}(logical(Q{k}<=Ls(k)*ones(size(Q{k})))&...
logical(Q{k}>=Li(k)*ones(size(Q{k})))));
factor(k)=A/mediamod(k);
G(i:i+paso-1,j:j+paso-1)=I(i:i+paso-1,j:j+paso-1).*factor(k);
k=k+1;
end
end
end

```

### Algoritmo de crecimiento por regiones (cregiones)

```

function [R,CC]=cregiones(varargin)
%CREGIONES segmenta una imagen en escala de grises homogenea empleando un
%algoritmo de crecimiento por regiones según Gonzales, Rafael et al [1].
%R=cregiones(varargin) retorna la imagen segmentada (R) en blanco y negro.
%VARARGIN es un cellarray que puede contener: tan solo la imagen en escala
%de grises (f) o la imagen (f), el valor semilla (S) y la tolerancia (T)
%definida según [1].
%REFERENCIAS:
%[1] Gonzales, Rafael et al. Digital image processing using Matlab.Pág.
%408-411.
error(nargchk(1,3,nargin))
ifnumel(varargin)==1
iptcheckinput(varargin{1},{'numeric'},{'real','nonempty','finite'},...
'func_name','var_name',2);
f=varargin{1};S=mean2(f)-4*std2(f);T=2.5*std2(f);
else
iptcheckinput(varargin{1},{'numeric'},{'real','nonempty','finite'},...
'func_name','var_name',2);
f=varargin{1};S=varargin{2};T=varargin{3};
end
ifnumel(S)==1

```

```

%Encontrar el valor semilla más cercano en caso de no coincidir con el
%calculado en ningun pixel de la imagen.
f=double(f);
    [~,C]=min(min(abs(f-round(S))));
    [~,F]=min(abs(f(:,C)-round(S)));
%%%
    S=f(F,C);SI=f==S;S1=S;
else
    SI=bwmorph(S,'shrink',Inf);
    J=SI;S1=f(J);
end
TI=false(size(f));
for k=1:length(S1)
    seedvalue=S1(k);
    S=abs(f-seedvalue)<=T;
    TI=TI | S;
end
CC=bwconncomp(imreconstruct(SI,TI));
R=labelmatrix(CC);

```

### Algoritmo para eliminar juntas y otras regiones (eregiones)

```

function [R,CC]=eregiones(BW,r)
%EREGIONES elimina juntas de losas de concreto y demás objetos distintos a
%una fisura dentro de la función FIPAVCO. Para eliminar las regiones se
%definen dos estructura de 5 mm x 25 cm (una vertical y una horizontal) que
%elimine las juntas aplicando dos veces la operación morfologicatophat,
%y posteriormente todo objeto que no tenga la redondez, excentricidad u
%orientación semejante a una fisura se remueve.
%[R2,CC]=eregiones(R,r) retorna la imagen en blanco y negro tras eliminar
%las regiones (R2) y su estructura de conectividad (CC). BW es la imagen a
%blanco y negro tras la segmentación y r la relación milímetro/pixel en la
%imagen original del pavimento rígido a color.
error(nargchk(1,2,nargin))
iptcheckinput(BW,{'numeric','logical'},{'nonempty'},'func_name','var_name',2);
SE1=strel('rectangle',round([250/r 5/r]));SE2=strel('rectangle',round([5/r
250/r]));
BW=imfill(BW);BW=imtophat(BW,SE1);BW=imtophat(BW,SE2);
CC=bwconncomp(BW);BW=labelmatrix(CC);
prop=regionprops(CC,'Area','Perimeter','Eccentricity','Extent',...
'MajorAxisLength','Orientation','MinorAxisLength');
BW(ismember(BW,find(logical(logical(abs([prop.Orientation])<5)...&logical([prop.M
inorAxisLength]<25/r))|logical(logical(abs([prop.Orientation])>85)...&logical([pr
op.MinorAxisLength]<25/r))|logical([prop.Area]./[prop.Perimeter].^2>0.025)...
|logical([prop.Extent]>0.45)|logical((logical([prop.MajorAxisLength]<500/r)...&lo
gical([prop.Eccentricity]<0.95)))))=0;
CC=bwconncomp(BW);R=labelmatrix(CC);

```

### Algoritmo para unir regiones (uregiones)

```

function [U CCU]=uregiones(R,CC,varargin)
%UREGIONES une regiones de fisuras con orientación semejante dentro de la
%función FIPAVCO, producto de modificar el algoritmo descrito por
%Liu, Fanfan et al [1].
%[U CCU]=uregiones(R,CC) retorna la imagen tras unir las regiones (U) y su
%estructura de conectividad (CCU). R es la imagen en blanco y negro tras
%eliminar las regiones en la función EREGIONES, CC su correspondiente

```

```

%estructura de conectividad y alfa el ángulo de tolerancia para unir dos
%fisuras.
%REFERENCIAS:
%[1] Liu, Fanfan et al. Novel approach to pavement cracking automatic
%detection based on segmentation extending. 2008 International Symposium on
%Knowledge Acquisition and Modeling.
error(nargchk(1,3,nargin))
iptcheckinput(R,{'logical','numeric'},{'nonempty','finite'},'func_name','var_name',
2);
if isempty(varargin)
alfa=45;
else
alfa=varargin{1};
end
while (true&&CC.NumObjects~=0)
prop=regionprops(CC, 'Centroid', 'BoundingBox', 'MajorAxisLength', 'Orientation', 'Ext
rema');
[~,orden]=sort([prop.MajorAxisLength], 'descend'); regcercanas=cell(length(orden),1
);
regcon=zeros(size(orden)); BB={prop.BoundingBox}; G=R; S=CC.ImageSize;
for i=orden
if abs(prop(i).Orientation)<=35
reg1=R(max(1,round(BB{i}(1,2)-0.5*BB{i}(1,4))):min(S(1),...
round(BB{i}(1,2)+1.5*BB{i}(1,4))),max(1,round(BB{i}(1,1)...
-BB{i}(1,3))):max(1,round(BB{i}(1,1)))));
reg2=R(max(1,round(BB{i}(1,2)-0.5*BB{i}(1,4))):min(S(1),...
round(BB{i}(1,2)+1.5*BB{i}(1,4))),min(S(2),round(BB{i}(1,1)...
+BB{i}(1,3))):min(S(2),round(BB{i}(1,1)+2*BB{i}(1,3)))));
regcercanas{i}=[reshape(reg1(reg1>0),1,[]) reshape(reg2(reg2>0),1,[])];
regcercanas{i}=setdiff(unique(regcercanas{i}),[regcon i]);
elseif abs(prop(i).Orientation)>=55
reg1=R(max(1,round(BB{i}(1,2)-
BB{i}(1,4))):max(1,round(BB{i}(1,2)...
)),max(1,round(BB{i}(1,1)-0.5*BB{i}(1,3))):min(S(2),...
round(BB{i}(1,1)+1.5*BB{i}(1,3)))));
reg2=R(min(S(1),round(BB{i}(1,2)+BB{i}(1,4))):min(S(1),...
round(BB{i}(1,2)+2*BB{i}(1,4))),max(1,round(BB{i}(1,1)...
-0.5*BB{i}(1,3))):min(S(2),round(BB{i}(1,1)+1.5*BB{i}(1,3)))));
regcercanas{i}=[reshape(reg1(reg1>0),1,[]) reshape(reg2(reg2>0),1,[])];
regcercanas{i}=setdiff(unique(regcercanas{i}),[regcon i]);
else
reg1=R(max(1,round(BB{i}(1,2)-0.5*BB{i}(1,4))):max(1,...
round(BB{i}(1,2))),max(1,round(BB{i}(1,1))):min(S(2),...
round(BB{i}(1,1)+BB{i}(1,3))));
reg2=R(min(S(1),round(BB{i}(1,2)+BB{i}(1,4))):min(S(1),...
round(BB{i}(1,2)+1.5*BB{i}(1,4))),max(1,round(BB{i}(1,1))...
:min(S(2),round(BB{i}(1,1)+BB{i}(1,3))));
reg3=R(max(1,round(BB{i}(1,2))):min(S(1),round(BB{i}(1,2)...
+BB{i}(1,4))),max(1,round(BB{i}(1,1)-0.5*BB{i}(1,3))...
:min(S(2),round(BB{i}(1,1))));
reg4=R(max(1,round(BB{i}(1,2))):min(S(1),round(BB{i}(1,2)...
+BB{i}(1,4))),min(S(2),round(BB{i}(1,1)+BB{i}(1,3))...
:min(S(2),round(BB{i}(1,1)+1.5*BB{i}(1,3))));
regcercanas{i}=[reshape(reg1(reg1>0),1,[]) reshape(reg2...
(reg2>0),1,[]) reshape(reg3(reg3>0),1,[])
reshape(reg4(reg4>0),1,[])];
regcercanas{i}=setdiff(unique(regcercanas{i}),[regcon i]);

```

```

end
regcercanas{i}=regcercanas{i}(logical(abs(abs([prop(regcercanas{i}).Orientation]
...-abs(prop(i).Orientation))<=alfa));
if ~isempty(regcercanas{i})
regionbase=cell(size(regcercanas{i}));[regionbase{:}]=...deal([prop(i).Extrema]);
regionesproximas={prop(regcercanas{i}).Extrema};
[k,d]=cellfun(@dsearchn,regionbase,regionesproximas,'UniformOutput',0);
[distposicion]=cellfun(@min,d);[~,lugar]=min(dist);
regcercanas{i}=regcercanas{i}(lugar);
xo=min(S(1),max(1,round(prop(i).Extrema(k{lugar}(posicion(lugar)),2))));
yo=min(S(2),max(1,round(prop(i).Extrema(k{lugar}(posicion(lugar)),1))));
xf=min(S(1),max(1,round(prop(regcercanas{i}).Extrema(posicion(lugar),2))));
yf=min(S(2),max(1,round(prop(regcercanas{i}).Extrema(posicion(lugar),1))));
xi=[xo,xf];yi=[yo,yf];[cx,cy,~]=improfile(R,xi,yi);...
indices=sub2ind(size(R),round(cx),round(cy));
R(indices)=i;
end
regcon=[setdiff(regcon,0) regcercanas{i} i];
end
CC=bwconncomp(bwmorph(R,'bridge'));R=labelmatrix(CC);
if all(G==R)
break
end

end
if ~exist('G','var')
U=zeros(size(R));
CCU=struct;
else
CCU=CC;U=R;
end
end

```

## Anexo C. Manual de la interfaz gráfica del usuario ALFIL

Con el fin de brindar una herramienta que emplee el algoritmo ALFIL, se desarrolló una interfaz gráfica del usuario (UI) en Matlab que permite identificar, clasificar y medir las fisuras. El uso de la interfaz es muy sencillo, pudiéndose resumir en los siguientes pasos:

1. Iniciar el programa denominado como ALFIL.
2. En el menú Inicio->Abrir se carga la imagen a color que se va a estudiar. La imagen aparecerá en el recuadro izquierdo de la pantalla (la imagen puede ser de cualquier tamaño).



3. Ingresar la relación mm/pixel de la imagen cargada en el cuadro de texto correspondiente.
4. Si se desea cambiar algún parámetro de los algoritmos empleados en ALFIL ingresar al menú Parámetros y realizar los cambios pertinentes para cada procedimiento.
5. Si se desean obviar ciertos procedimientos en el desarrollo del algoritmo (excepto la segmentación por crecimiento por regiones y la clasificación), esto se puede realizar deseleccionando los procedimientos desde el menú Procedimientos.
6. Dar clic en el botón ejecutar y esperar hasta que aparezca la imagen en blanco y negro en el recuadro derecho de la pantalla.



- Una vez se haya cargado la imagen en blanco y negro, en éstase etiquetará en el centroide de cada fisura reconocida la clase a la cual fue clasificada por el algoritmo ALFIL.

8. Si se desea tener una mejor visualización de la imagen clasificada o se desean realizar mediciones basta con hacer clic en el botón Herramienta de Imagen para cargar la herramienta *imtool* de Matlab.
9. En los cuadros de texto presentados en la parte inferior de la interfaz se presenta la información obtenida de medir el área fisurada y el porcentaje que ésta representa del total de la imagen. En la lista desplegable "Tipo de fisura" se puede seleccionar el tipo de fisura a analizar y en los cuadros "No. Fisuras Tipo", "Área fisurada por tipo (m2)" y "Porcentaje fisurado por tipo" aparecerán el número de fallas correspondientes a la selección realizada, el área fisurada para el total de dichas fisuras y el porcentaje que ésta área representa del total de la imagen., respectivamente Finalmente, en los cuadros "No. Total de fisuras", "Área fisurada total (m2)" y "Porcentaje fisurado total" aparecerán, respectivamente, el número total de fisuras reconocidas por Alfil en la imagen, el área fisurada que suman todas las fisuras y el porcentaje del área de la imagen que representa dicha área fisurada.



10. Para guardar los resultados obtenidos, se ingresa al menú Inicio->Guardar. Al guardar la imagen el archivo gráfico no salvará las etiquetas. Además de la imagen, se guarda un archivo de texto plano con la misma dirección y nombre que la imagen que contiene los descriptores, la clasificación y las mediciones para cada fisura reconocida en la imagen mediante ALFIL. El esquema del archivo de texto luce semejante a como se muestra en la siguiente imagen.

Archivo	Edición	Formato	Ver	Ayuda					
Elongación	7.220947	4.058497	3.220917	12.074317	7.214143	4.528745	5.181691	6.268454	
Orientación °	74.899212	12.149933	50.852427	6.173464	81.672773	42.130560	40.102302	81.237227	
Extensión	0.039136	0.375436	0.354489	0.120521	0.158257	0.212607	0.092644	0.189866	
Clasificación	2.000000	5.000000	5.000000	4.000000	2.000000	5.000000	5.000000	5.000000	
Área(m2)	0.963090	0.000000	0.000000	0.004451	0.002036	0.000000	0.000000	0.000000	
Porcentaje	27.523148	0.000000	0.000000	0.127215	0.058192	0.000000	0.000000	0.000000	

11. Finalmente, si se desea abrir otra imagen basta con ir a Inicio->Abrir y repetir el procedimiento enunciado anteriormente, en caso contrario, ir al menú Inicio->Salir o desde el ícono correspondiente ubicado en la pantalla,