



**Universidad Autónoma de Querétaro
Facultad de Ingeniería
Campus San Juan del Río**



“Mejora de contraste para imágenes de microscopio”

TESIS

Que para obtener el título de:

Ingeniero Electromecánico con especialidad en Mecatrónica

Presenta:

Marco Antonio Garduño Ramón

Asesor:

M. I. Hayde Peregrina Barreto

Co-Asesor:

Dr. Luis Alberto Morales Hernández

San Juan del Río, Qro. Marzo de 2012

Resumen

Para la metalografía óptica, contar con microscopios con una cámara digital montada es de utilidad para analizar características diferentes de muestras metálicas. Sin embargo, debido a factores como la iluminación, es difícil de controlar la calidad de las imágenes. Esto afecta directamente al contraste, el cual permite distinguir entre dos regiones con diferente luminancia, característica de vital importancia en aplicaciones metalográficas. La aplicación de métodos de mejora de contraste ofrece una opción para la corrección de imágenes con bajo nivel de iluminación. En el presente trabajo se demuestra cómo, a través de diferentes mejoras de contraste, es posible mejorar los resultados de medición de microdureza en metales haciendo uso del procesamiento de imágenes.

Palabras Clave: mejora de contraste, luminancia, metalografía.

Abstract

In optical metallography the use of a microscope and a camera is helpful for analyzing different characteristics of metallic samples. However, factors such as illumination make it difficult to control the quality of the image samples. This directly affects to the contrast level of the image. Contrast is the characteristic which allows distinguishing between two regions with different luminance. When there is a barely noticeable difference, it is difficult to determine where one region ends and another begins. The application of contrast improvement methods is a solution for the correction of images with low contrast level. In this work, it is shown how through the contrast improvement it is possible to improve the results of the micro-hardness measurement.

Keywords: contrast enhancement, luminance, metallography.

A la memoria de Cristina Ramón Trejo (†).

Mi madre.

Agradecimientos

A mis padres, el Sr. Eugenio Garduño Rocha y la Sra. Cristina Ramón Trejo (†), por toda su confianza y apoyo. A mi hermano, Carlos Eugenio, por ser mi mejor amigo. A mi asesora, la M. en I. Hayde Peregrina Barreto por su paciencia y dedicación para la buena culminación de este proyecto. Al Dr. Luis Alberto Morales Hernández por su amistad y asesoría brindada. A mis amigos Iván y Néstor y a compañeros de generación...

ÍNDICE

1. INTRODUCCIÓN	1
1.1. Antecedentes	2
1.2. Objetivos	5
1.2.1. Objetivo general.....	5
1.2.2. Objetivos particulares	5
1.3. Justificación	5
1.4. Planteamiento general.....	6
2. REVISIÓN DE LITERATURA	8
2.1. Estado del arte	10
2.2. Técnicas de mejora de contraste	11
2.2.1. Ecuación del histograma	11
2.2.2. Balance de blancos.....	12
2.2.3. Retinex.....	14
2.3. Espacios de color.....	14
2.4. El formato TIFF	16
2.5. Licencia pública general GNU (GNU GPL)	16
3. METODOLOGÍA.....	18
3.1. Análisis de luminancia en imágenes.....	18
3.2. Conversión de espacios de color.....	23
3.3. Estudio de los métodos de mejora de contraste	23
3.3.1. Estudio de ecuación de histograma.....	24
3.3.2. Estudio de balance de blancos	25
3.3.3. Balance de blancos con valores neutros.....	26

3.3.4. Estudio de Retinex.....	27
4. RESULTADOS	29
4.1. Análisis de muestras a tratar.....	29
4.2. Mejora de contraste.....	31
4.3. Aplicación	41
4.4. Comparativa de aplicación de métodos en canal <i>Y</i> y <i>RGB</i>	48
5. CONCLUSIONES	51
6. REFERENCIAS.....	53
7. ANEXOS	56
7.1. Código Cont_Ench.c de llamado de las funciones de la librería Cont_Ench.h	56
7.2. Librería Cont_Ench.h.....	60
7.3. Base de datos (imágenes con buen contraste)	83
7.4. Base datos (imágenes con mal contraste)	86
7.5. Retinex en Matlab	87

ÍNDICE DE FIGURAS

Figura 1.1 Diagrama a bloques del proyecto.....	7
Figura 2.1 (a), (b), (c) y (d) Diferentes niveles de contraste.	9
Figura 2.2 Muestra metalográfica tomada bajo pobres condiciones de iluminación.	10
Figura 2.3 Representación del espacio <i>RGB</i>	15
Figura 3.1 Análisis del valor medio de luminancia aplicado a las imágenes con buen contraste de la base de datos de prueba.	19
Figura 3.2 Análisis del valor medio de luminancia aplicado a las imágenes con bajo contraste de la base de datos de prueba.	19
Figura 3.3 (a), (c) y (e) Imágenes con buena luminancia. (b), (d) y (f) Sus histogramas de luminancia correspondientes.....	20
Figura 3.3 (Continuación).	21
Figura 3.4 (a) y (c) Imágenes con mala luminancia. (b) y (d) Sus histogramas de luminancia correspondientes.	21
Figura 3.4 (Continuación).	22
Figura 3.5 Histograma de $G(g)$	24
Figura 3.6 Ejemplo de porcentaje de pixeles blancos y oscuros tomados para un balance de blancos con valores neutros.....	27
Figura 4.1 (a), (b), (c) y (d) Muestras tomadas bajo pobres condiciones de iluminación.	30
Figura 4.2 (a), (b), (c) y (d) Histograma de luminancia de la Figura 4.1 (a), (b), (c) y (d), respectivamente.	31
Figura 4.3 (a) Muestra utilizada para medición de microdureza tomada bajo pobres condiciones de iluminación(Figura 4.1 (a)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).....	32
Figura 4.3 (Continuación).	33

Figura 4.4 (a) Muestra utilizada para medición de tamaño de grano tomada bajo pobres condiciones de iluminación (Figura 4.1 (b)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).	34
Figura 4.4 (Continuación).	35
Figura 4.4 (Continuación).	36
Figura 4.5 (a) Muestra de fibras tomada bajo pobres condiciones de iluminación (Figura 4.1 (c)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).....	37
Figura 4.5 (Continuación).	38
Figura 4.6 (a) Muestra utilizada para medición de tamaño de grano tomada bajo pobres condiciones de iluminación (Figura 4.1 (d)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).	39
Figura 4.6 (Continuación).	40
Figura 4.7 Muestra bien preparada para medición de microdureza (a) y diferentes variantes de luminancia de la misma (b) y (c).	42
Figura 4.8 (a) Figura 4.7 (a) introducida al sistema de medición de microdureza.	43
Figura 4.9 (a) (Figura 4.7 (b)), modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i), introducidas al sistema de medición de microdureza (b, d, f, h, y j).	43
Figura 4.9 (Continuación).	44
Figura 4.9 (Continuación).	45
Figura 4.10 (a) (Figura 4.7 (c)), modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i), introducidas al sistema de medición de microdureza (b, d, f, h, y j).	46
Figura 4.10 (Continuación).	47

Figura 4.11 (a) Muestra de fibras tomada bajo pobres condiciones de iluminación, modificada mediante ecualización del histograma en Y' (b) y RGB (c), balance de blancos en Y' (d) y RGB (e), balance de blancos con valores neutros en Y' (f) y RGB (g) y Retinex en Y' (h) y RGB (i).....	49
Figura 4.11 (Continuación).	50
Figura 7.1 Retinex en Matlab de Frankle-McCann.	87

ÍNDICE DE TABLAS

Tabla 7.1 Información estadística de la base de datos de imágenes con buen contraste.	85
Tabla 7.2 Información estadística de la base de datos de imágenes con mal contraste.	86

1. INTRODUCCIÓN

Durante los últimos años las diferentes ramas científicas se han apoyado de diversas herramientas tecnológicas y técnicas computacionales para llevar a cabo su objetivo. La ciencia de los materiales se ha valido de un sinnfn de equipos y sistemas para lograr una visión más amplia de la estructura y propiedades de los materiales. Dichas herramientas se han ido actualizando y modernizado con el pasar de los años y hoy es común ver sistemas automatizados en los laboratorios metalográficos para obtener información acerca de los diferentes tipos de metales y aleaciones. Anteriormente se requería de personal entrenado para llevar a cabo las pruebas y el análisis de muestras. Sin embargo, dado el factor humano, el proceso llevaba más tiempo y se producía cierto grado de incertidumbre en los resultados. Así, el uso de herramientas tecnológicas ha llegado a ser no solo de gran apoyo para el investigador y científico, sino que también ha mejorado la precisión y tiempo del análisis de muestras.

Algunos equipos hacen uso de cámaras fotográficas y permiten analizar una muestra en base a su imagen digital desde una computadora personal. Aplicando diversas técnicas computacionales y de procesamiento de imágenes, es posible lograr un ahorro de tiempo y obtener resultados más precisos. Además, esto abre la posibilidad de cuantificar, sobre una misma muestra, diversas características como: número de granos, composición estructural, calidad del acabado, etc. Estos sistemas están cobrando auge tanto a nivel industrial como de investigación por las características e innovaciones que presentan, esto implica una renovación de equipos y materiales dejando sin uso aparentemente a equipos con varios años en servicio.

La visión por computadora y el procesamiento de imágenes son dos áreas estrechamente relacionadas con investigaciones particulares dentro de la ciencia de los materiales. Las posibilidades que ofrece esta relación, simplificada a un

microscopio con una cámara montada, consiste en optimizar el análisis y estudio de materiales, para lograr obtener una visión más cercana a sus propiedades bajo ciertas condiciones de uso, con el fin de entender y prevenir los mecanismos de desgaste y deterioro que presentan los componentes metálicos en un sistemas mecánico.

1.1. Antecedentes

Particularmente en la Universidad Autónoma de Querétaro, se han desarrollado algunos sistemas que permiten caracterizar diferentes propiedades de los materiales ferrosos. Estas investigaciones incluyen:

Medición de la microdureza de la capa endurecida de un acero (Cayetano, 2010). Este sistema analiza la imagen de la huella dejada por el indentador del durómetro calibrado según la norma que lo rige, para determinar sus bordes y área que se puede relacionar con su profundidad y así determinar su dureza, haciendo uso de procesamiento de imágenes implementado en un software propio desarrollado en Visual Studio C++ con el formato de imágenes BMP. El sistema ofrece resultados satisfactorios y es un proceso semiautomático ofreciendo un considerable ahorro de tiempo. Sin embargo, una vez montada la muestra es necesario hacer un ajuste de contraste a fin de mejorar la calidad de la imagen y disminuir el error en los cálculos. Dicho ajuste es hecho de forma manual por el operador y requiere tomar un número considerable de imágenes de una misma muestra para después elegir la que represente mejor a la información de la muestra. Esto implica un mayor costo de almacenamiento, por el formato utilizado, tiempo de análisis y procesamiento.

Detección automática del límite de grano usando una aproximación de interceptos (Benítez et al., bajo revisión). Desarrollado para realizar la medición del tamaño y número de granos de una muestra que corresponde al campo de visión de un microscopio, mediante el uso de tres imágenes, que son: un patrón normado (como ejemplo se menciona una circunferencia), la imagen binarizada (donde se ven los bordes de los granos) y la muestra preparada en que se diferencia cada grano con un tono de gris. El proceso consiste en sobreponer el patrón con la imagen binarizada

para que el software automáticamente encuentre una intersección entre el patrón y el borde para después verificar con la tercera imagen si hay un cambio de tono de gris indicando una intercepción, pintando el punto de contacto y así sucesivamente hasta recorrer todo el contorno del patrón. Finalmente se cuenta el número de intersecciones que corresponderá al número de puntos de contacto pintados. Este sistema es aplicable no solo a materiales ferrosos sino también a materiales cerámicos, y ofrece un mejor nivel de exactitud y ahorro de tiempo. El sistema usa el formato BMP de imágenes lo cual implica una gran cantidad de espacio digital utilizado en sus muestras.

Medición automática de grafito en acero de fundición por procesamiento de imágenes (Morales et al., 2007). Lograron un proceso automático que lleva a cabo la cuenta de las inclusiones de grafito en la microestructura del acero. Esto sustituye el proceso manual llevado a cabo por un operador que hace sobreponer una rejilla a la muestra del microscopio para realizar la cuenta. Dicha tarea requiere de alrededor de 250 mediciones y su precisión está relacionada con el factor humano. Este sistema, realizado en Visual Studio C++, requiere de una buena preparación de la muestra para después digitalizarla, mejorarla y así facilitar su umbralización. La umbralización ayuda a contrastar el grafito del acero y mediante el conteo de píxeles blancos y negros se obtiene el porcentaje de grafito por área de la muestra. El método requiere de alrededor de 50 muestras.

Caracterización de metalografías de aluminio por medio de visión de computadora (Morales et al., 2009). Desarrollaron un sistema que permite caracterizar las propiedades del aluminio mediante el conteo de poros. Dichas características entregan información suficiente para relacionarlas con propiedades mecánicas y térmicas en aleaciones con Magnesio. Para esto se analiza la imagen digital de la muestra a través de diversos tratamientos a fin de ajustar su umbralización y obtener una imagen segmentada. En esta imagen se aprecian los poros en color negro sobre un fondo blanco, permitiendo implementar un sistema automático que mida su proporción con respecto al tamaño de la muestra. Con esto se evalúa el porcentaje volumétrico de porosidad, se cuenta el número de poros, y su

distribución. Este sistema fue implementado mediante el uso de Visual Studio C++ y usa el formato BMP.

Análisis de la microestructura para detectar incrustaciones de grafito en el acero por medio de análisis de imágenes (Solís, 2008). Se desarrolló un software para medir el porcentaje de grafito incrustado en una muestra metalográfica, aplicando técnicas de umbralización y filtros morfológicos a fin de facilitar la identificación del grafito. Estas técnicas permiten diferenciar estructuras de grafito, contarlas y relacionarla con el tamaño de la muestra. El sistema hace uso de Visual C++ y formato BMP. Ofrece una opción viable en contraste a otras opciones en el mercado que tiene un alto precio, además de un ahorro de tiempo en la realización de las pruebas y mayor exactitud en las mediciones. Se hace énfasis en que se requiere de una buena preparación de las muestras, para evitar ruido, y procurar una buena calidad en la captura de imágenes.

Efecto de la austenita retenida sobre el esfuerzo residual en acero SAE 8620 carbonitrurado (Granados, 2008). Se aplicaron técnicas como análisis químico, metalografía óptica, ensayos de microdureza y difracción de Rayos X, con el fin de encontrar la relación entre la austenita retenida y los altos esfuerzos residuales, ambas características encontrados en los análisis de engranes que han presentado fallas en pruebas de laboratorio y en servicio. Se analizó el engrane de manera visual buscando deformación plástica y al microscopio pudiéndose observar la austenita retenida como pequeñas trazas de color blanco. Si bien su cuantificación se realizó mediante otros métodos, al final de este trabajo se propone la implementación de un sistema de procesamiento de imágenes para llevar a cabo la medición del porcentaje volumétrico de austenita residual mediante imágenes de los dientes de engrane, esto requerirá de cierto número de muestras así como de una buena calidad de imagen.

En general, se puede apreciar que los trabajos de metalografía óptica, requieren de imágenes en que se pueda apreciar detalladamente o lo mejor posible la muestra a analizar, esto conlleva a la obtención de mejores resultados, así como la necesidad de un menor número de fotos para realizar los estudios.

1.2. Objetivos

1.2.1. Objetivo general

Incorporar algoritmos de mejora de contraste a un sistema de medición metalográfico, con la finalidad de mejorar la calidad de la imagen obtenida de cualquier microscopio o equipo especializado y facilitar dicha medición.

1.2.2. Objetivos particulares

- Implementar algoritmos de mejora de contraste.
- Aplicar dichos algoritmos a muestras digitales de metales a fin de mejorar su calidad visual.
- Comparación de los resultados obtenidos con diversos métodos.
- Comparación de las mediciones obtenidas antes y después de mejorar la imagen.
- Creación de una librería que contenga los procesamientos aplicados.

1.3. Justificación

La importancia de mejorar el nivel de contraste de una imagen, además de mejorar su apariencia visual, radica en poder hacer una mejor discriminación entre sus elementos. Esto facilita, tanto al observador como a diferentes algoritmos de procesamiento de imágenes, distinguir de manera más sencilla los límites entre una región y otra. Uno de los factores que afectan el contraste de una imagen es su calidad de iluminación. De manera que, si se controlan las condiciones de iluminación se pueden evitar fallas en el contraste; sin, embargo, esto implica la adecuación de un espacio y el equipo necesario para ajustar la iluminación. Algunos de los equipos más

recientes para el análisis de muestras ya incluyen herramientas de autoajuste para mejorar la calidad de la imagen a capturar. Sin embargo, debido a su costo no siempre es posible tener acceso a uno de estos equipos. Otra solución viable es el hacer uso de técnicas de procesamiento de imágenes que permiten mejorar el contraste en base a la información aportada por la imagen y que pueden facilitar tareas posteriores como la segmentación. El uso de técnicas de mejora de contraste aplicadas a la medición de microdureza es la propuesta de la presente tesis.

Actualmente existen herramientas que son capaces de llevar a cabo la mejora de contraste en imágenes digitales e incluso algunos fabricantes desarrollan módulos propios de software para ser utilizados en los equipos más recientes de su marca. Sin embargo, si se toma en cuenta que se puede contar con equipo de diferentes fabricantes y que la creación de herramientas propias a menudo hace uso de entornos de desarrollo propietarios, el costo sigue siendo el principal problema.

A fin de evitar estos inconvenientes, una solución viable es el uso de entornos de desarrollo de software libre que permiten la portabilidad de la aplicación y la reducción del costo de desarrollo. El contar con una herramienta propia para la mejora del análisis de muestras permite, no solo reducir el costo de tecnológico que implica la adquisición de equipo nuevo, sino además alargar el tiempo de vida del equipo con que ya se cuenta y así evitar dependencia tecnológica.

1.4. Planteamiento general

En la Figura 1.1 se muestra un diagrama a bloques del trabajo a realizar y la secuencia a seguir para que el sistema sea implementado de manera adecuada. A continuación se procede a explicar de manera general en qué consiste cada bloque.

Muestra digital: Se obtiene una serie de muestras de la pieza preparada mediante una cámara montada en un microscopio de uso general o en algún equipo especializado como un durómetro.

Análisis de luminancia: A partir del histograma de luminancia de la muestra digital es obtenida la información de contraste de la imagen para determinar si debe ser procesada.

Mejora de contraste: La muestra digital es procesada con diferentes métodos para mejorar su contraste, estos son elegidos a libre criterio por el operador de tal modo que pueda seleccionar el que más se adecue a sus necesidades y requerimientos para su aplicación específica.

Técnica metalográfica: Con la imagen mejorada se procede a aplicarle alguna técnica metalográfica programada en algún software como medición del tamaño de grano, medición de microdureza, etc.

Resultado: Se procede a comparar los resultados obtenidos entre los sistemas con y sin mejora de contraste. En este punto se verifica el nivel de eficacia que logra la aplicación de la técnica metalográfica cuando se le añade mejora de contraste a las muestras tomadas, de tal modo que se pueda mejorar dicho sistema, así como comparar los resultados obtenidos con diferentes técnicas de mejora de contraste a un proceso específico.

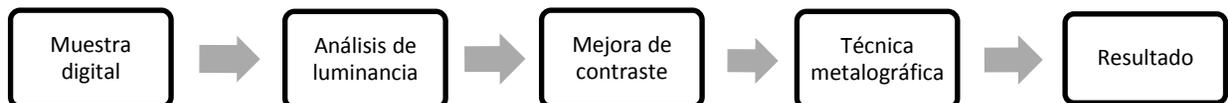


Figura 1.1 Diagrama a bloques del proyecto.

2. REVISIÓN DE LITERATURA

En este capítulo se tratan los conceptos necesarios para comprender la problemática planteada en este trabajo, como el concepto, trabajos y métodos de mejora de contraste, la importancia de los espacios de color, el formato de imágenes TIFF y la Licencia Pública General GNU.

De acuerdo con la Comisión Internacional de Iluminación (CIE), brillo es la cantidad de luz producida por una fuente luminosa y luminancia es la intensidad con la que la luz impacta el ojo humano, de tal manera que la luminancia nos permite distinguir si una región es más brillante u oscura que otra (<http://www.cie.co.at/>). La importancia de poder distinguir entre dos regiones siempre es importante para facilitar la interpretación visual y la característica relacionada a esto se le llama contraste. Por ejemplo, en la Figura 2.1 (a) se varía el nivel de intensidad del marco posterior, de tal manera que entre más alejado esté del nivel de luminancia del cuadro central es más fácil distinguir los límites entre uno y otro (alto contraste). Sin embargo entre más cercamos son sus valores de luminancia (Figura 2.1 (b), (c) y (d)) se hace más difícil distinguir las dos regiones (bajo contraste). La mayoría de las aplicaciones ópticas requieren cuidar del contraste, y en el caso de la metalografía óptica no es la excepción (Figura 2.2), ya que está relacionado con la exactitud de los resultados.

Existen, en la actualidad, varias formas para tratar de mejorar el contraste de una imagen, y puede ser realizado tanto en el momento previo a la adquisición, así como después, una vez que se tiene almacenada la imagen en algún formato digital. Para mejorar el contraste de una imagen previa a ser tomada, se debe tener cuidado de las condiciones de iluminación a través de instrumentos que midan dicho nivel (p. ej., luxómetro) y si es necesario se corrija mediante el uso de herramientas de iluminación.

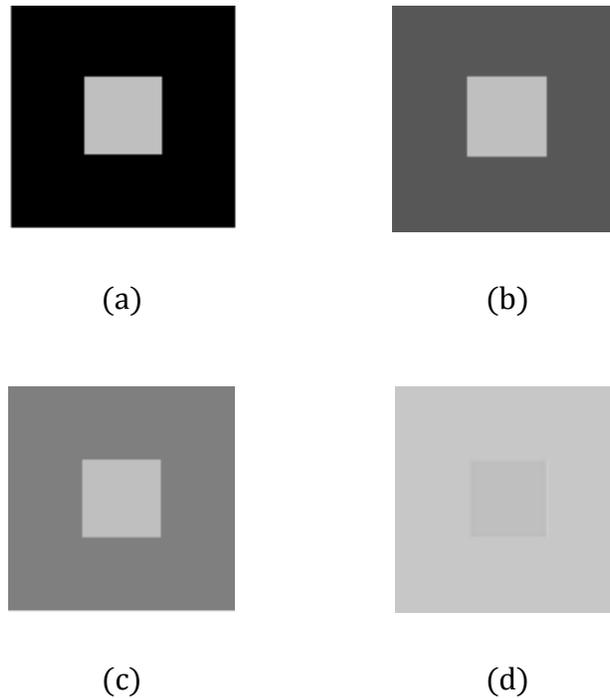


Figura 2.1 (a), (b), (c) y (d) Diferentes niveles de contraste.

Si bien hoy en día se han desarrollado equipos que proveen un ajuste automático de iluminación mediante el uso de sensores y lámparas, estos requieren de una calibración según las características propias del local o área de trabajo. Esto implica que se dependa del experto para realizar este ajuste, agregando cierto grado de incertidumbre ya que aspectos humanos como cansancio, sueño, hambre o estrés, merman la calidad del ajuste y por ende de la muestra. Sin embargo, no siempre es posible contar con los equipos antes mencionados debido a sus altos costos, y en caso de contar con alguno, no hay que olvidar que son sensibles a las variaciones eléctricas presentes, p. ej., en un taller de máquinas herramientas.

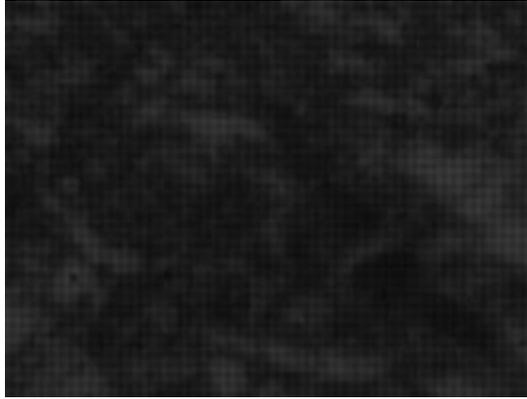


Figura 2.2 Muestra metalográfica tomada bajo pobres condiciones de iluminación.

Por otra parte, es posible mejorar la imagen una vez que ésta ya ha sido adquirida, mediante el uso de técnicas de mejora de contraste. Esta es una corrección común debido a que gran cantidad de imágenes no pueden ser tomadas bajo condiciones de iluminación controladas. Existen una gran cantidad de técnicas destinadas a mejorar la calidad de la imagen. Cada una de ellas tiene ciertas características que presentan ventajas y desventajas sobre otras. Esto abre la posibilidad de que el experto pueda elegir la que más se adecue a su aplicación.

2.1. Estado del arte

Numerosos trabajos dedicados a la mejora y estudio de contraste en imágenes de diversas áreas y para diferentes aplicaciones han sido desarrollados tanto a nivel nacional como internacional, por ejemplo:

Mukhopadhyay y Chanda (2000) proponen la aplicación de filtros morfológicos multiescala para mejora de contraste local en imágenes de tomografías cerebrales con zonas oscuras, permitiendo observar de manera más clara los detalles en esas áreas una vez tratada la muestra. De manera similar Serrano et al., (2005) desarrollaron un algoritmo para calibrar imágenes color de muestras de quemaduras tomadas bajo condiciones de iluminación no controladas, dada la situación de emergencia con que fueron adquiridas y la iluminación inadecuada, así como con una

configuración desconocida de la cámara; dichas imágenes pasan del espacio de color RGB al XYZ para buscar los factores de corrección que compensen la iluminación y la falta de uniformidad espacial de la sensibilidad de la cámara.

Otros trabajos enfocados a mejora de contraste de imágenes en general incluyen el propuesto por Mendiola y Terol (2005), que mediante la diferencia de contraste y de gradiente proponen un método para poder cuantificar el nivel de contraste. Esto permite elegir en base a un factor la imagen con mejor contraste dentro de una serie de muestras. Espino y Terol (2007), proponen la mejora de contraste a través de la aplicación de transformaciones morfológicas multiescala basadas en la percepción visual humana. Peregrina y Terol (2010), implementan operaciones para mejora de contraste en imágenes con diferentes condiciones de iluminación.

En la presente tesis serán estudiados y utilizados algunos de estos métodos aplicados a muestras metalográficas, cuya descripción se muestra en las siguientes secciones.

2.2. Técnicas de mejora de contraste

2.2.1. Ecuación del histograma

El histograma de una imagen es una función discreta que representa el número de píxeles en la imagen con respecto a los niveles de intensidad (Pajares y de la Cruz, 2004). En general, todas las imágenes poseen un histograma único, incluso imágenes de áreas diferentes de la misma escena o muestra, tendrá diferentes histogramas, dependiendo del área cubierta. Cambiar las condiciones de iluminación o la configuración de la cámara, da como resultado una modificación del histograma (Russ, 2007).

El método de ecualización del histograma permite realzar la luminancia de una imagen mediante una modificación a la distribución de su histograma. Se busca

una función $F(g)$ que realce el contraste general de la imagen original expandiendo la distribución de los niveles de gris. Dicha expansión debe ser lo más suave posible, es decir, se deben distribuir los niveles de gris de una manera uniforme a lo largo de todo el rango de valores de niveles de gris (Pajares y de la Cruz, 2004). Para una distribución uniforme como se presenta en la Ecuación 2.1, g_{max} y g_{min} son el rango de valores máximo y mínimo en que queremos que se distribuya el histograma y $P_g(g)$ es la función de densidad de probabilidad mostrada en la Ecuación 2.2.

$$F(g) = [g_{max} - g_{min}]P_g(g) + g_{min} \quad \text{Ecuación 2.1}$$

$$P_g(g) = \sum_{g=0}^g p(g) \quad \text{Ecuación 2.2}$$

2.2.2. Balance de blancos

Es utilizado frecuentemente en los dispositivos de adquisición de imágenes y consiste en balancear el nivel de iluminación de manera que este sea uniforme en toda la imagen. El balance de blancos toma, como referencia de corrección, la parte con mayor luminancia en la imagen (blanco de referencia) y la modifica hasta que parezca de color blanco, así mismo, toma la parte más oscura y la modifica hasta que parezca negro. De esta forma establece los rangos en los que será distribuida la luminancia de la imagen.

El método de balance de blancos, también llamado *White Patch*, asume que los valores máximos de cada banda de color *RGB*, espacio de color explicado en la sección 2.3, representan al blanco de referencia (Cardei et al., 1999). Dicho método debe encontrar una porción blanca o cercana al blanco dentro de la imagen. Esta región blanca permite dar una idea del iluminante que tuvo la escena y en base a esto hacer el balance de luminancia. El iluminante original puede ser estimado más fácilmente si al menos un pixel refleja la mayor cantidad de luminancia posible

(Ebner, 2003). Para esto es necesario mediante la Ecuación 2.3 recorrer la imagen hasta encontrar el más claro.

$$RGB_{\max}(\lambda_i) = \max\{I(x,y)\} \quad \text{Ecuación 2.3}$$

Se utilizan los valores del máximo (RGB_{\max}) para reescalar las bandas de color en un rango [0,1]. Estos valores sirven para determinar los coeficientes correspondientes a cada banda de color y son los que hacen posible la corrección al multiplicar al valor original de cada pixel tal y como se muestra en la Ecuación 2.4.

$$\begin{bmatrix} L_{\text{Deseado}} \\ M_{\text{Deseado}} \\ S_{\text{Deseado}} \end{bmatrix} = \begin{bmatrix} L/L_{\text{Max}} & 0 & 0 \\ 0 & M/M_{\text{Max}} & 0 \\ 0 & 0 & S/S_{\text{Max}} \end{bmatrix} \begin{bmatrix} L_{\text{MaxDeseado}} \\ M_{\text{MaxDeseado}} \\ S_{\text{MaxDeseado}} \end{bmatrix} \quad \text{Ecuación 2.4}$$

Para determinar el pixel más cercano al blanco, en este caso se utiliza como base la distancia euclidiana que existe entre cada pixel y el blanco. El pixel con distancia mínima a este es el punto más claro de la imagen. Puesto que el espacio en que se encuentra la imagen es RGB , y en procesamiento de imágenes el rango de valor de un pixel es [0,255], donde 0 es negro y 255 blanco, la distancia euclidiana de un pixel blanco está dada por la Ecuación 2.5.

$$D_{\text{white}} = \sqrt{(255 - R_i)^2 + (255 - G_i)^2 + (255 - B_i)^2} \quad \text{Ecuación 2.5}$$

Una vez obtenido el pixel con distancia mínima, $P_{D_{\min}} = [R_x, G_x, B_x]$ este valor dividirá a cada uno de los pixeles de la imagen. En este caso las operaciones se aplican sustituyendo directamente los valores LMS de la Ecuación 2.4 por RGB como se observa en la Ecuación 2.6.

$$\begin{bmatrix} R_{\text{Deseado}} \\ G_{\text{Deseado}} \\ B_{\text{Deseado}} \end{bmatrix} = \begin{bmatrix} R/R_x & 0 & 0 \\ 0 & G/G_x & 0 \\ 0 & 0 & B/B_x \end{bmatrix} \begin{bmatrix} R_{\text{MaxDeseado}} \\ G_{\text{MaxDeseado}} \\ B_{\text{MaxDeseado}} \end{bmatrix} \quad \text{Ecuación 2.6}$$

2.2.3. Retinex

Es un algoritmo que mejora la representación visual de las imágenes cuando las condiciones de iluminación son pobres y que se muestra en la Ecuación 2.7. Se basa en el mecanismo biológico del ojo para la adaptación cromática y luminancia local de cada pixel para aproximar el realismo de la escena en observación directa.

El algoritmo calcula la luminancia (L) de un punto (x_p) influenciado por N puntos (x_i), los cuales son escogidos aleatoriamente, utilizando la siguiente fórmula:

$$L = \frac{1}{N} \sum_{i=1}^N \left(\log(I(x_p)) - \log(I(x_i)) \right) \quad \text{Ecuación 2.7}$$

El proceso es aplicado a cada una de las bandas de color. Retinex brinda una mejor percepción de zonas oscuras (McCann et al, 2004). Si bien existen varias versiones de este método, la que se va utilizar en este trabajo es la propuesta por Frankle-McCann.

2.3. Espacios de color

Sirven para facilitar la especificación de los colores de una manera estándar en un sistema de coordenadas 3-D donde cada color se representa por un punto (Figura 2.3). En el espacio de color *RGB* cada color aparece en sus componentes espectrales primarios rojo, verde y azul, y se basa en el sistema de coordenadas cartesianas. *RGB* es un espacio de color aditivo ya que para dar origen a un color específico es necesario mezclar determinadas cantidades de rojo, verde y azul (Pajares y de la Cruz, 2004).

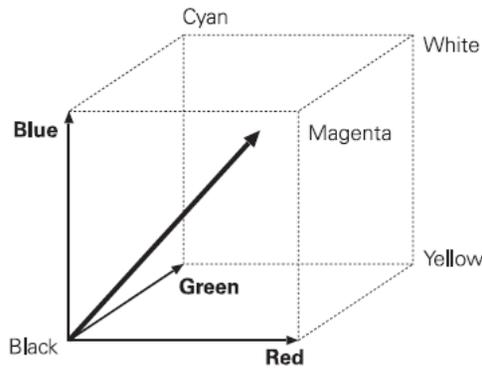


Figura 2.3 Representación del espacio *RGB*.

Los valores *RGB* se hallan en tres vértices, cyan, magenta y amarillo en otros tres, el negro corresponde al origen, y el blanco está en el vértice más alejado del origen. La escala de grises se extiende desde el negro al blanco a lo largo de la diagonal que une esos puntos, y los colores son puntos dentro del tetraedro, definidos por vectores desde el origen. Se asume que todos los vectores han sido normalizados, de modo que el tetraedro es unitario por lo que todos los valores están en el rango [0,1]. Este es el espacio de color más utilizado en imágenes digitales debido a su fácil representación en dispositivos electrónicos.

El espacio de color $u'v'Y'$ es un espacio que permite separar la información cromática de la acromática (Lucchese y Mitra, 2004). Esto es de suma importancia ya que el contraste puede ser mejorado cambiando los valores de luminancia que corresponden al canal Y' sin afectar el matiz de los canales $u'v'$. Es vital conservar la representación cromática original de una imagen ya que el cambio de luminancia no debe crear nuevos colores. Para usar este espacio es necesario cambiar los valores *RGB* a $u'v'Y'$ usando la matriz de transformación mostrada en la Ecuación 2.8.

$$\begin{bmatrix} Y' \\ u' \\ v' \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.14713 & -0.28886 & 0.436 \\ 0.615 & -0.51499 & -0.10001 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \text{Ecuación 2.8}$$

2.4. El formato TIFF

TIFF (*Tagged Image File Format* por sus siglas en inglés) es un formato de archivo de gráficos de mapa de bits. Fue desarrollado por *Aldus Corporation* en el año de 1986, actualmente el formato pertenece a *Adobe Systems Incorporated*. El formato permite almacenar imágenes de mapa de bits muy grandes, más de 4 GB comprimidos y sin considerar las plataformas o periféricos utilizados. Permite almacenar imágenes binarias, en colores verdaderos (hasta 32 bits por pixel). Además permite que se utilicen varios espacios de color: *RGB*, *CMYK* (*Cyan*, *Magenta*, *Yellow and Key (black)*), *CIE*L*a*b*, *YUV / YCrCb*. El principio del formato TIFF consiste en definir etiquetas que describen las características de la imagen, estas nos dicen las dimensiones de la imagen, la cantidad de colores utilizados, el tipo de compresión, etc. TIFF es uno de los formatos más utilizados en publicaciones científicas debido a su buena representación y compresión (Adobe, 1992).

2.5. Licencia pública general GNU (GNU GPL)

El modo más sencillo de hacer un programa libre es ponerlo en el dominio público, es decir, sin *copyright*. Esto permitirá que la gente comparta el programa y sus mejoras, pero también permitirá que quienes no compartan las modificaciones puedan convertirlo en software privativo. Esto lo logra modificándolo tanto como desee y posteriormente distribuyendo el resultado como un producto comercial. Aquellos que reciban la versión modificada no tendrán la libertad que el autor original tenía en mente debido al intermediario que se la privo. Por lo tanto en lugar de poner software libre bajo dominio público se distribuye mediante *copyleft*. El *copyleft* es la forma general de hacer un software libre y requerir que todas las modificaciones y versiones extendidas del programa sean también software libre. Significa que cualquiera que distribuya el software, con o sin cambios, no podrá restringir a nadie la libertad de copiarlo, redistribuirlo o modificarlo. De tal manera que *copyleft* garantiza que el usuario mantenga su libertad. En el proyecto GNU

(<http://www.gnu.org/home.es.html>) las cláusulas específicas de distribución están contenidas en la Licencia Pública General (GPL por sus siglas en inglés). La licencia está incluida en muchos manuales y en cada distribución de código fuente GNU (<http://www.gnu.org/licenses/licenses.es.html>). La GPL de GNU se diseñó de tal forma que pueda ser aplicada fácilmente en los programas; si se es el titular de los derechos de autor, basta poner anuncios en su programa que haga la referencia apropiada a la GPL de GNU.

En general las licencias GNU implican cuatro libertades para los usuarios de software. Primero, la libertad de usar el programa con cualquier propósito, segundo la libertad de estudiar el funcionamiento del programa y adaptarlo a las necesidades, tercero la libertad de distribuir copias con lo que se ayuda a otros, por último la libertad de mejorar el programa y hacer públicas las mejoras.

3. METODOLOGÍA

3.1. Análisis de luminancia en imágenes

Mediante el uso de herramientas computacionales usadas con el fin de editar imágenes, se obtuvo la información del histograma de luminancia de algunas muestras almacenadas en la PC usada en el laboratorio de metalografía a fin de encontrar algún parámetro o característica que compartan las imágenes que a nuestro juicio, por inspección visual, tengan buen contraste. Se hizo uso del software GIMP en su versión 2.6.11.

Se analizaron las propiedades estadísticas de media, mediana y desviación estándar de la distribución de la luminancia, dicha información es proporcionada por el software antes mencionado, de las muestras organizadas en dos grupos, contraste bajo y buen contraste. Para los fines de este estudio se analizaron un total de 50 imágenes con buen contraste y 17 muestras con bajo contraste consideradas como las que mejor representan estas propiedades (Anexo 7.3 y 7.4). En general fue posible observar cierta relación entre las características de contraste de las imágenes con su media, siendo un factor confiable para los fines de este trabajo permitiendo determinar la calidad de una muestra. Dicha información estadística es mostrada en la Figura 3.1 y en la Figura 3.2, primeramente se observa la información obtenida de las imágenes de buen contraste y después las de aquellas con bajo contraste.

De esta manera, mediante la información proporcionada en la Figura 3.1 y en la Figura 3.2 se pueden establecer los rangos de bajo contraste entre 4.7 y 58.1, y las de buen contraste entre 76.9 y 204.5.



Figura 3.2 Análisis del valor medio de luminancia aplicado a las imágenes con buen contraste de la base de datos de prueba.

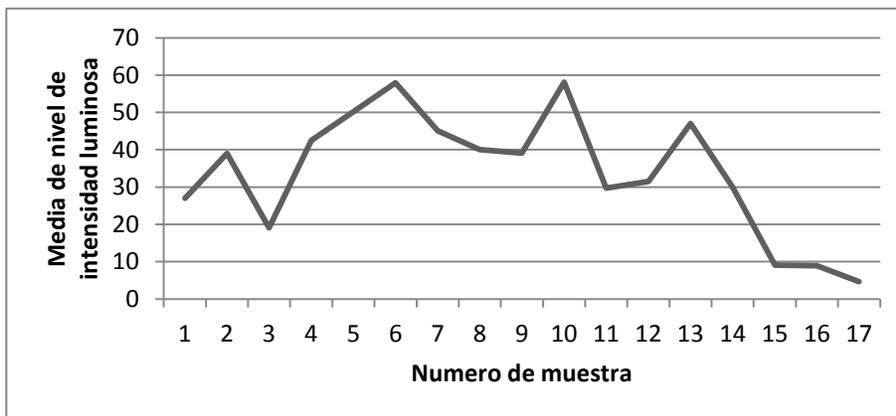
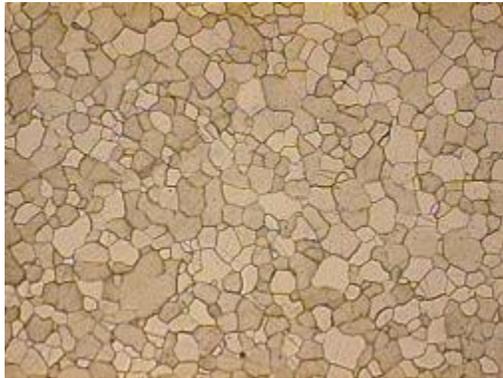


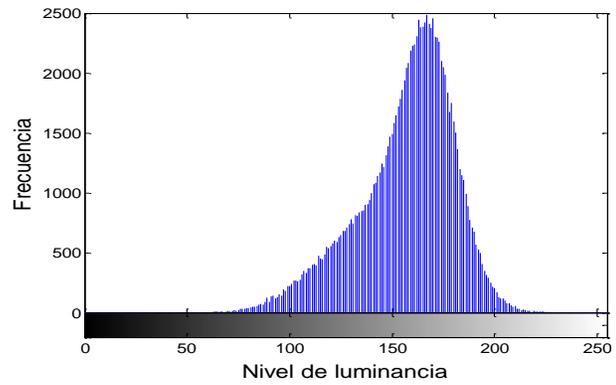
Figura 3.2 Análisis del valor medio de luminancia aplicado a las imágenes con bajo contraste de la base de datos de prueba.

Además de la información anterior, la interpretación del histograma de luminancia también es útil, ya que mediante la forma y distribución de éste podríamos inferir las características de contraste que posee. Por ejemplo, en las imágenes de la Figura 3.3 (b), (d) y (f), podemos observar que los histogramas abarcan una amplia cantidad de niveles de gris representada con picos extendidos. Generalmente este tipo de distribución se presenta en imágenes con buen contraste, cómo se muestra en la Figura 3.3 (a), (c) y (e), donde se distinguen con facilidad las regiones de la imagen.

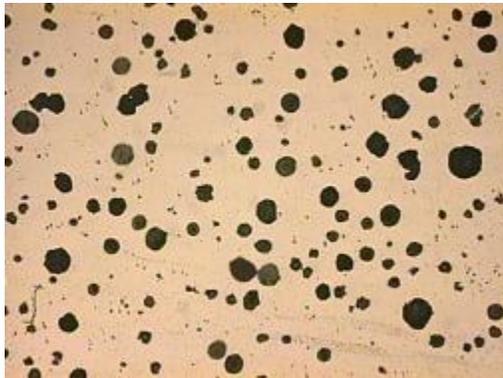
De la misma manera se analizó el histograma de las muestras con bajo contraste a fin de encontrar alguna forma de histograma que, de manera similar al caso anterior, defina la baja calidad de iluminación bajo la que fue adquirida la imagen.



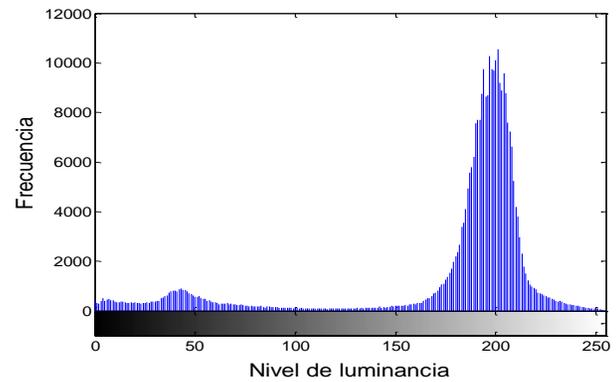
(a)



(b)



(c)

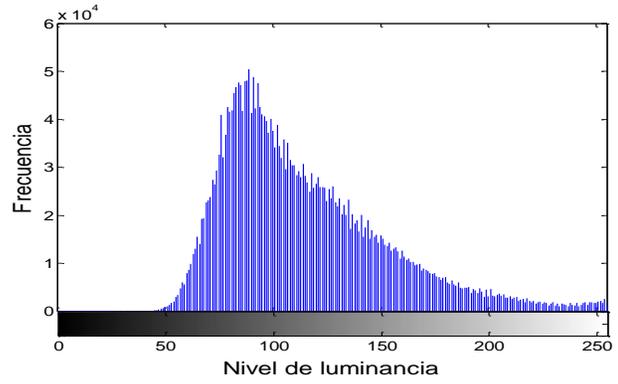


(d)

Figura 3.3 (a), (c) y (e) Imágenes con buena luminancia. (b), (d) y (f) Sus histogramas de luminancia correspondientes.



(e)

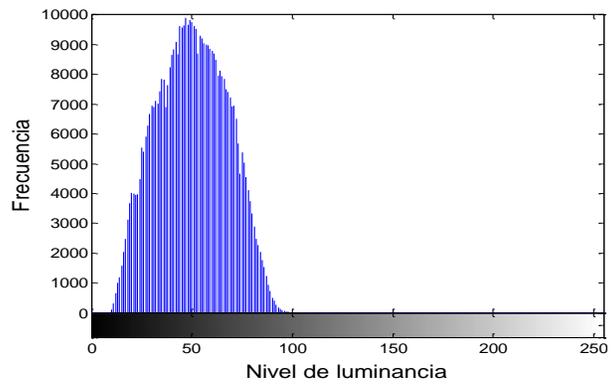


(f)

Figura 3.3 (Continuación).



(a)

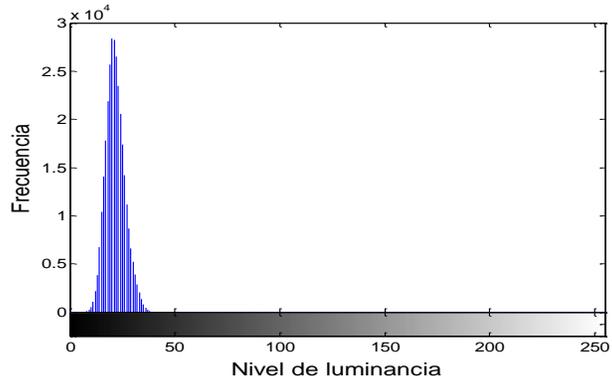


(b)

Figura 3.4 (a) y (c) Imágenes con mala luminancia. (b) y (d) Sus histogramas de luminancia correspondientes.



(c)



(d)

Figura 3.4 (Continuación).

En los histogramas mostrados en la Figura 3.4 (b) y (d) se observa la concentración de valores en la parte izquierda, que corresponde a los niveles oscuros, además se observan pocos o nulos píxeles en la región correspondientes a tonos claros del histograma. A diferencia del caso anterior, los histogramas de las imágenes con bajo contraste abarcan una cantidad menor de niveles de gris y sus picos son estrechos. A juicio de un observador, las pequeñas diferencias de contraste parecieran no ser suficiente para diferenciar claramente los elementos que componen las imágenes, sin embargo un sistema de medición computacional podría ser suficiente para la discriminación de regiones. En la Figura 3.4 (a) es posible diferenciar, con cierto esfuerzo, los detalles que la componen, mientras que en la muestra de la Figura 3.4 (c) la discriminación entre elementos es menos probable.

Los valores estadísticos de media obtenidos en las imágenes de la Figura 3.3 son 181, 204.5 y 152.9 respectivamente. De manera similar se observa que los valores de media para las imágenes de la Figura 3.4 son 50.2 y 30. Esto permite verificar que los rangos establecidos a partir de la Figura 3.1 y la Figura 3.2 son aceptables para nuestro objeto de estudio.

3.2. Conversión de espacios de color

El espacio RGB es el espacio de color que utilizan la mayoría de los dispositivos de despliegado y adquisición de imágenes. Sin embargo, si bien es posible aplicar los métodos de mejora de contraste directamente sobre este espacio, podríamos obtener como resultado una alta modificación del color de las muestras, por eso se pasa del espacio de color RGB al $u'v'Y'$, en el cual sólo modificaremos el canal de luminancia que corresponde al canal Y' , para poder realizar esta conversión de espacios basta con aplicar la matriz de transformación mostrada en la Ecuación 3.1.

$$\begin{bmatrix} Y'_i \\ u'_i \\ v'_i \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.14713 & -0.28886 & 0.436 \\ 0.615 & -0.51499 & -0.10001 \end{bmatrix} \begin{bmatrix} R_i \\ G_i \\ B_i \end{bmatrix} \quad \text{Ecuación 3.1}$$

El canal de luminancia Y' se mueve en el rango de 0 a 255, por lo que podemos implementar nuestros métodos de mejora de contraste sin problema alguno. Una vez llevado a cabo el proceso de mejora de contraste, debemos regresar la información hacia el espacio de color RGB a fin de poder desplegar el resultado, para ello se utiliza la matriz de transformación mostrada en la Ecuación 3.2.

$$\begin{bmatrix} R_i \\ G_i \\ B_i \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.135983 \\ 1 & -0.39465 & -0.58060 \\ 1 & 2.03211 & 0 \end{bmatrix} \begin{bmatrix} Y'_i \\ u'_i \\ v'_i \end{bmatrix} \quad \text{Ecuación 3.2}$$

3.3. Estudio de los métodos de mejora de contraste

A fin de lograr un mejor entendimiento de los métodos de mejora de contraste a utilizar para facilitar su implementación en cualquier espacio de color (RGB y $u'v'Y'$) y ayudar a la creación de la librería que contenga dichos métodos, se explica el funcionamiento de cada uno de ellos de una manera más amigable y entendible.

3.3.1. Estudio de ecualización de histograma

El objetivo de la ecualización del histograma es realizar una distribución uniforme de los valores de luminancia de nuestra imagen a lo largo de todos los valores posibles. Por ejemplo, considérese un vector de longitud 10 con valores de luminancia entre 0 y 9.

$$G(g) = (5, 1, 7, 6, 4, 4, 1, 7, 0, 1)$$

Se calcula el número de píxeles por nivel de luminancia que tenemos, es decir, el histograma de la imagen ($N(g)$) Figura 3.5.

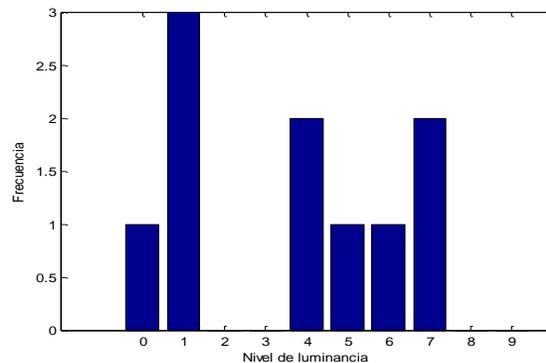


Figura 3.5 Histograma de $G(g)$.

Teniendo el histograma de la imagen calculamos la probabilidad de cada nivel de luminancia ($P(g)$), que es $N(g)$ entre el número de píxeles de nuestra imagen.

$$P(g) = \left(\frac{1}{10}, \frac{3}{10}, \frac{0}{10}, \frac{0}{10}, \frac{2}{10}, \frac{1}{10}, \frac{1}{10}, \frac{2}{10}, \frac{0}{10}, \frac{0}{10} \right)$$

Posteriormente se calcula la probabilidad acumulada ($Pp(g)$) que es la suma de probabilidades desde el nivel 0 hasta el 9 es este caso.

$$Pp(g) = \left(\frac{1}{10}, \frac{4}{10}, \frac{4}{10}, \frac{4}{10}, \frac{6}{10}, \frac{7}{10}, \frac{8}{10}, \frac{10}{10}, \frac{10}{10}, \frac{10}{10} \right)$$

Finalmente se calcula $F(g)$ que resulta de multiplicar $Pp(g)$ por el valor máximo de luminancia posible, para este ejemplo 9, y redondeando al valor entero más cercano. De tal modo que $F(g)$ sea el nuevo valor de luminancia.

$$F(g) = (1, 4, 4, 4, 5, 6, 7, 9, 9, 9)$$

Finalmente se hace la asignación de los nuevos valores de luminancia.

$$\text{Imágen ecualizada} = (6, 4, 9, 7, 5, 5, 4, 9, 1, 4)$$

En este ejemplo podemos observar como los pixeles con bajo nivel de luminosidad son realzados a valores centrales del rango posible, mientras que los valores altos son modificado muy ligeramente, esto podría traer como resultado una modificación importante de la imagen original.

3.3.2. Estudio de balance de blancos

El balance de blancos trata de hacer el ajuste de luminancia a partir del pixel con mayor valor de la imagen a procesar. Para este ejemplo retomamos nuestra imagen vector original del ejemplo anterior.

$$\text{Imágen vector} = (5, 1, 7, 6, 4, 4, 1, 7, 0, 1)$$

El método dice que busquemos el pixel con mayor valor P_{Max} en este caso en el tercer y octavo pixel que tienen un valor de 7, el cual será nuestro blanco de referencia. Posteriormente recorreremos todo el vector pixel a pixel y aplicamos la formula de balance de blancos mostrada en la Ecuación 3.3, la cual se deriva de la matriz mostrada en el capítulo anterior en la Ecuación 2.6.

$$P_i = \frac{P_{i\text{ original}}}{P_{Max}} P_{Max\text{ Deseado}} \quad \text{Ecuación 3.3}$$

De la formula, $P_{i\text{ original}}$ corresponde al valor de luminancia del pixel actual a modificar siendo P_i su nuevo valor, $P_{Max\text{ Deseado}}$ corresponde al máximo valor de nivel

posible que corresponde a 9 para nuestro caso. Aplicamos la formula de balanceo de blancos y obtenemos:

$$\textit{Imágen resultante} = (6, 1, 9, 8, 5, 5, 1, 9, 0, 1)$$

En este caso podemos observar como el cambio es sutil, con respecto al ecualizado de histograma. Este método, de hecho, se aplica a los tres canales del espacio *RGB*, por lo que se hace necesario buscar aquel pixel cuya suma de valores R, G y B sea mayor, y a partir de estos valores se aplica el procedimiento anterior a cada una de las bandas para finalmente volverse a sumar y obtener la imagen modificada.

3.3.3. Balance de blancos con valores neutros

Frecuentemente las cámaras llevan a cabo un ajuste automático al momento de tomar una imagen. Dicho ajuste acerca las regiones más claras al 255 y las más oscuras a 0, por lo que se hace conveniente, tomar las regiones menos modificadas como referencia. Este método funciona de la misma manera que el balance de blancos explicado en la sección anterior, sólo que en este caso el blanco de referencia se estima tomando la información proporcionada por cierto un cierto número de los pixeles más blancos junto con los más oscuros. Se establece un rango buscando el valor máximo y mínimo del histograma, a partir de ahí se toma cierto porcentaje de valores claros y oscuros, específicamente un 20% de pixeles blancos y un 10 % de pixeles negros Figura 3.6, estos porcentajes se determinaron después de llevar a cabo una serie de pruebas y observar que a un mayor valor de porcentaje de ambos tipos de pixeles no se obtenía una variación más significativa. Posteriormente se suman los niveles de intensidad de dichos pixeles y se promedian para obtener lo que será nuestro blanco de referencia.

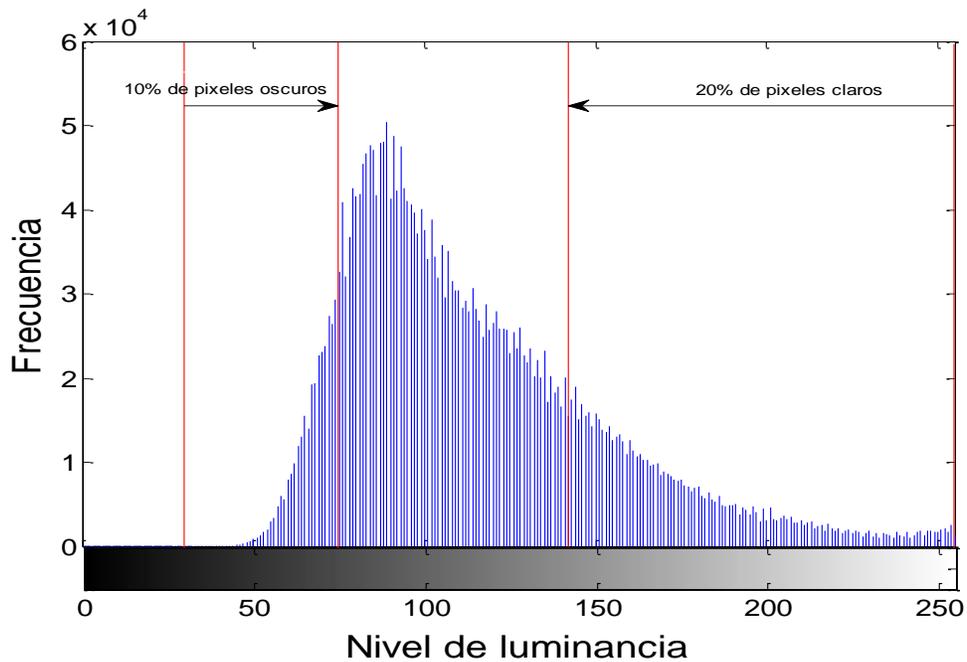


Figura 3.6 Ejemplo de porcentaje de pixeles blancos y oscuros tomados para un balance de blancos con valores neutros.

3.3.4. Estudio de Retinex

El algoritmo de Retinex utilizado en este trabajo fue propuesto por Flankle-McCann (Anexo 7.5) Este método es aplicado individualmente a cada canal de la imagen, sea Y' en el caso de luminancia o R, G y B para imágenes a color. El método especifica como entrada la imagen en forma logarítmica en el rango de 0 a 1, y un número de iteraciones, que generalmente es 4.

Una vez teniendo claro el funcionamiento de cada uno de los métodos anteriores se procedió a implementarlos bajo código C (Anexo 7.1), generando además la librería que contenga dichas funciones facilitando implementar dichos métodos en otras aplicaciones (Anexo 7.2). Dado que las imágenes utilizadas en este trabajo se encuentran en formato TIFF, es necesario también descargar e instalar la librería TIFF, disponible de manera gratuita. Esta librería tiene las funciones

requeridas para leer, modificar y escribir imágenes TIFF. Además recordemos que este formato hace uso de una serie de etiquetas donde almacena información de la imagen, p. ej., alto, ancho, orientación, espacio de color utilizado. Todas las funciones y etiquetas pueden ser consultadas en el manual de uso de la librería disponible en línea (<http://www.libtiff.org/index.html>).

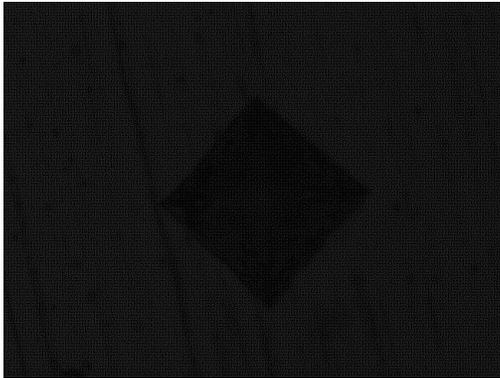
4. RESULTADOS

4.1. Análisis de muestras a tratar

Como se mencionó anteriormente, las condiciones de iluminación pueden afectar el contraste de una imagen. Para comprobar lo anterior, se experimentó con el cambio de iluminación al capturar imágenes de muestras metalográficas con ayuda de un experto. Debido a que la microdureza y el tamaño de grano son dos de los análisis más comunes en el área de metalografía, se consideraron imágenes de estos tipos para las pruebas.

Se capturaron imágenes de muestras metalográficas bajo condiciones de iluminación pobre, como se muestra en la Figura 4.1 (a), (b), (c) y (d). Se observa que el contraste en estas imágenes es muy bajo puesto que resulta muy difícil distinguir entre regiones. Estos casos de iluminación son extremos, sin embargo, son posibles. Se debe tomar en cuenta que, aun cuando una escena tenga baja iluminación, la percepción del sistema visual humano siempre es mejor que la de un dispositivo de adquisición. Es decir, las condiciones de la escena real varían, en mayor o menor grado, cuando se le representa en una imagen digital dependiendo de diversos factores como la calidad del dispositivo o el sensor de calibración. El propósito de estas muestras, oscuras en extremo, es mostrar el grado de mejora que se puede obtener cuando se aplica un método de mejora de contraste evitando el volver a adquirirlas. La información estadística proporcionada por las imágenes de la Figura 4.1 (a), (b), (c) y (d), indica que tienen una media de 9.1, 4.7, 50.2 y 44.2 respectivamente, esto se amolda de manera acertada a los parámetros fijados en el Capítulo 3 quedando clasificadas como imágenes con bajo contraste. La Figura 4.2 (a), (b), (c) y (d) muestra los histogramas de la Figura 4.1 (a), (b), (c) y (d), en estos es posible observar que todos están cargados hacia la izquierda, lo cual significa que en dichas imágenes hay una gran cantidad de píxeles oscuros y, al ser unos picos no muy

anchos, no hay gran variedad de niveles que permitan diferenciar fácilmente unas regiones de otras.



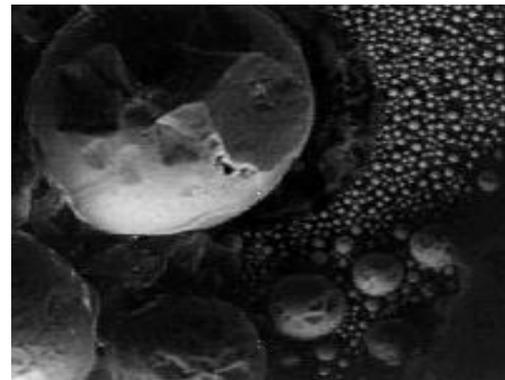
(a)



(b)

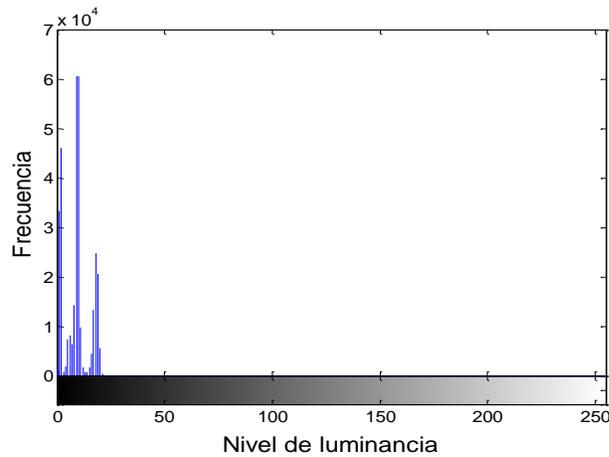


(c)

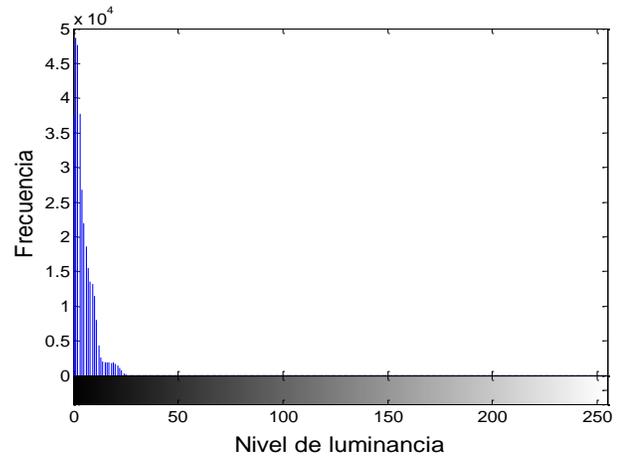


(d)

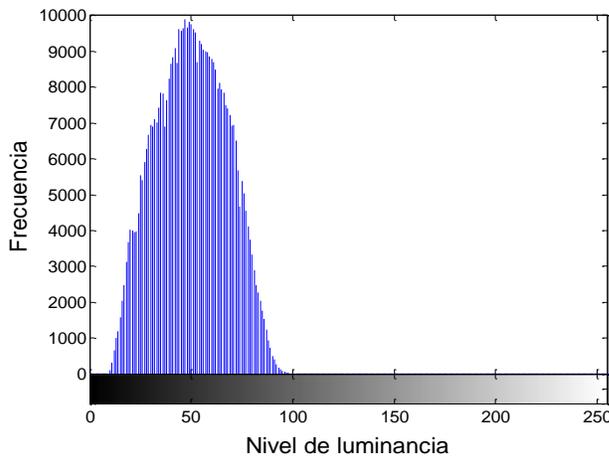
Figura 4.1 (a), (b), (c) y (d) Muestras tomadas bajo pobres condiciones de iluminación.



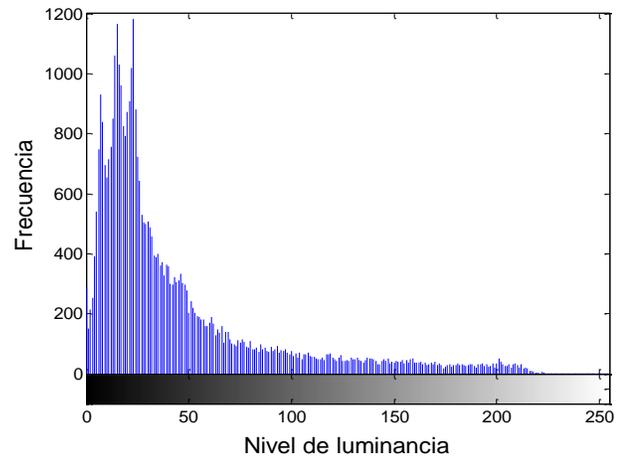
(a)



(b)



(c)



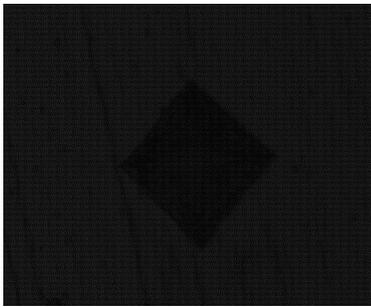
(d)

Figura 4.2 (a), (b), (c) y (d) Histograma de luminancia de la Figura 4.1 (a), (b), (c) y (d), respectivamente.

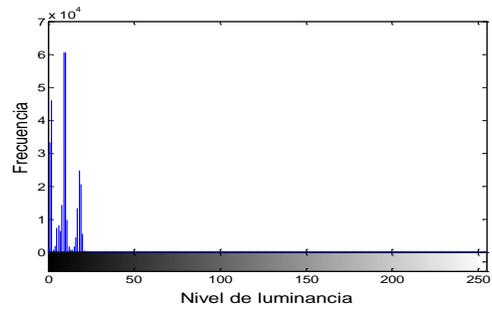
4.2. Mejora de contraste

Se aplicaron los métodos de mejora de contraste sobre el canal Y' mencionados en la metodología a las muestras metalográficas de la Figura 4.1 (a), (b), (c) y (d), a continuación se describen los resultados obtenidos.

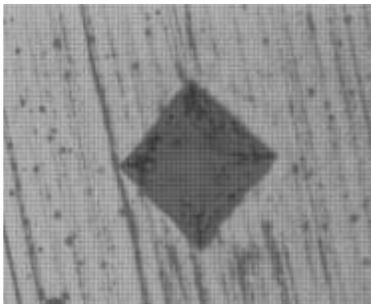
La primera muestra corregida corresponde a una imagen utilizada para realizar mediciones de microdureza Figura 4.1 (a). En dicho tipo de imágenes se busca que la indentación tenga un color oscuro desde el centro hasta el extremo tratando de que los bordes no se vean difusos, la parte metálica sin indentación debe tener un color claro.



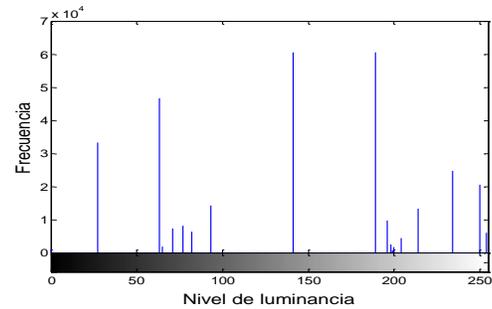
(a)



(b)

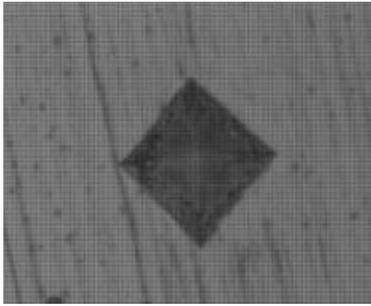


(c)

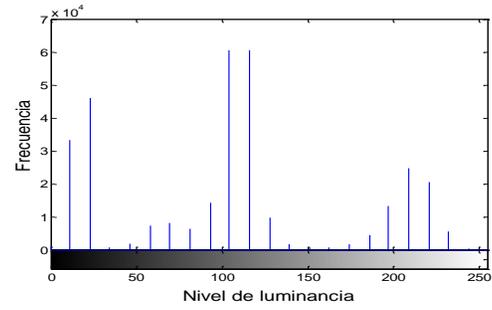


(d)

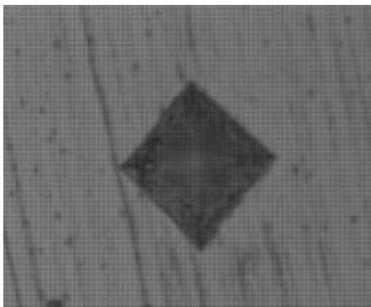
Figura 4.3 (a) Muestra utilizada para medición de microdureza tomada bajo pobres condiciones de iluminación(Figura 4.1 (a)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).



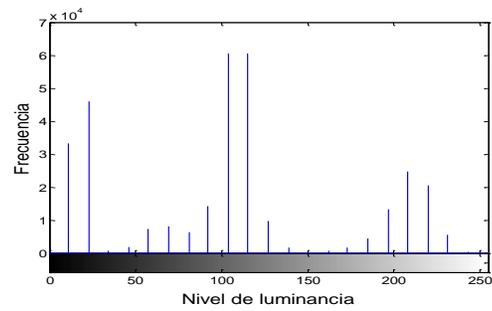
(e)



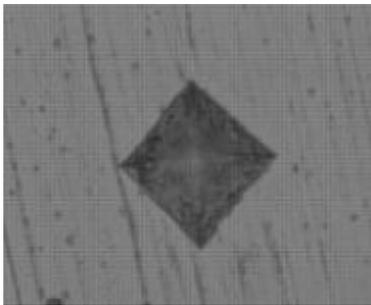
(f)



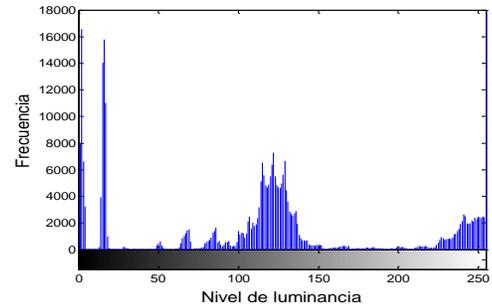
(g)



(h)



(i)

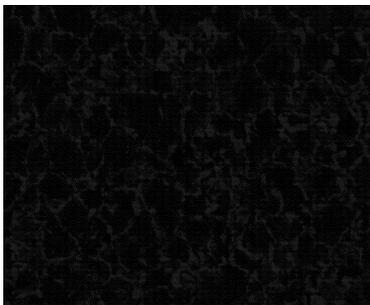


(j)

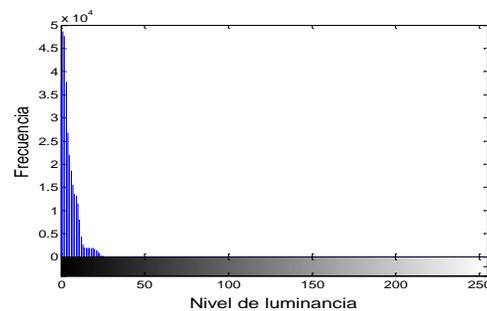
Figura 4.3(Continuación).

Los valores de media de la Figura 4.3 (c), (e), (g) e (i), obtenidos después de la aplicación de los mejora de contraste son de 142.1, 105.8, 105.3 y 118.6 respectivamente, de modo que en base a los criterios de clasificación establecidos en la sección 3.1, todas las imágenes mejoradas entran dentro de la clasificación de buen

contraste, siendo la mejor la Figura 4.3 (c) realizada con ecualización del histograma. El análisis de los histogramas Figura 4.3 (d), (f), (h) y (j), denota una clara mejora en comparación con el histograma original de la Figura 4.1 (a) mostrado en la Figura 4.2 (a), en todos es claramente visible como los valores de luminancia tienden a distribuirse a lo largo del rango posible, siendo en las muestra mejorada por Retinex, Figura 4.3 (i), donde se observa una mejor distribución de éste, Figura 4.3 (j). Visualmente, sin embargo, la muestra mejorada mediante ecualizado de histograma resalta en demasía líneas propias de la preparación de la muestra, que afectan de sobremanera su implementación en un método de mejora Figura 4.3 (c). Retinex por el contrario atenúa gran cantidad de estas líneas siendo, en este caso, el método que entrega mejores resultados Figura 4.3 (i). La segunda muestra corregida corresponde a una imagen utilizada para medición de tamaño de grano. Es importante para este tipo de muestras poder distinguir el límite de grano con respecto a la matriz férrica. Además de que los límites de grano estén continuos formando una malla completa sin interrupciones, lo cual se logra enfocando y buscando que la luz no adelgace las líneas del límite de grano.

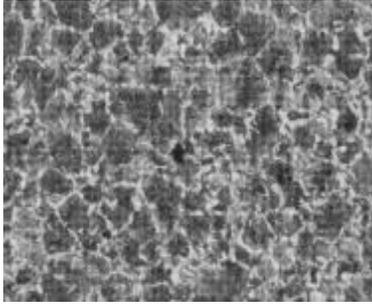


(a)

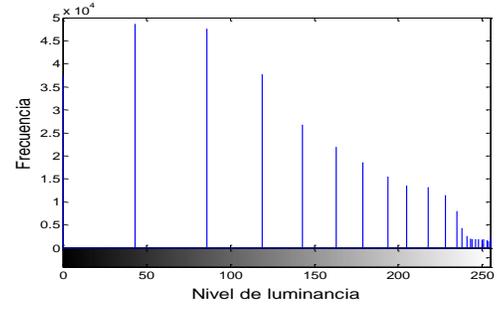


(b)

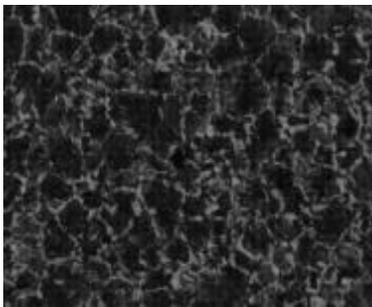
Figura 4.4 (a) Muestra utilizada para medición de tamaño de grano tomada bajo pobres condiciones de iluminación (Figura 4.1 (b)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).



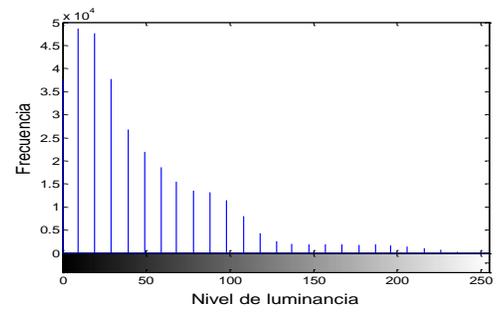
(c)



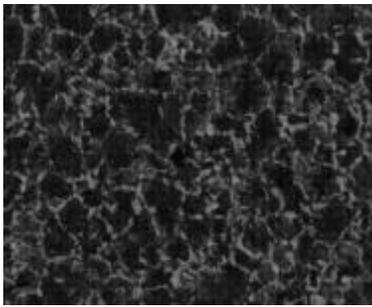
(d)



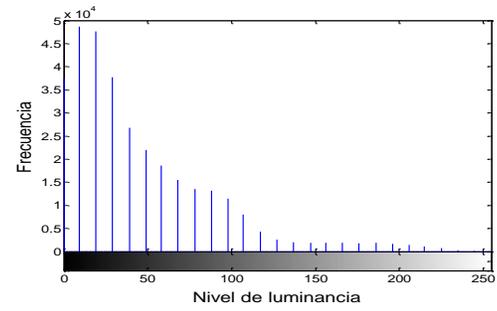
(e)



(f)

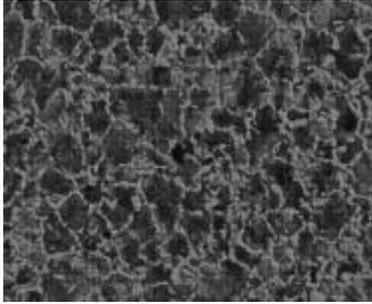


(g)

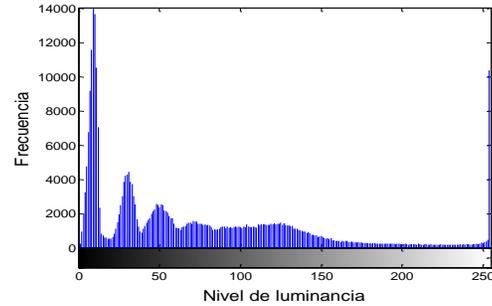


(h)

Figura 4.4 (Continuación).



(i)



(j)

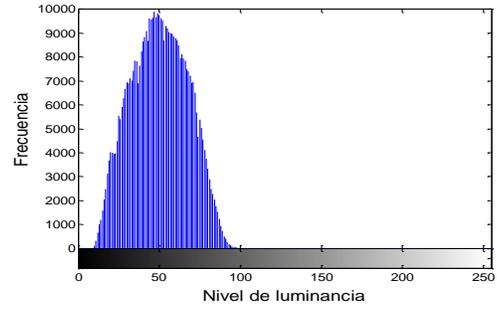
Figura 4.4(Continuación).

Llevando a cabo el análisis estadístico de las muestras mejoradas encontramos que la muestra mejora por ecualizado de histograma entrega el valor de media más alto Figura 4.4 (c) con un valor de 124.6, las demás imágenes corregidas, sin embargo, quedan por debajo de los criterios de clasificación para un buen contraste con 45.4, 45.3 y 72.5 para la Figura 4.4 (e), (g) e (i), respectivamente. De manera análoga, el histograma de la imagen tratada mediante ecualizado de histograma Figura 4.4 (d), es la que se distribuye de una manera más uniforme a lo largo del rango posible, mientras que los histogramas de los demás métodos Figura 4.4 (f), (h) y (j), siguen presentando altas concentraciones de valores de luminancia en los rangos de tonos bajos. Visualmente el realce mediante la ecualización del histograma es mejor que la de los otros métodos, donde sigue siendo complicado distinguir claramente algunos bordes de los diferentes granos, además es importante mencionar que en este caso la muestra de bajo contraste original Figura 4.1 (b) es la de más bajo nivel de media de las cuatro a analizar en esta sección.

La tercera muestra procesada corresponde a muestras de fibras. Aquí se busca que las fibras puedan apreciarse claramente de manera individual.



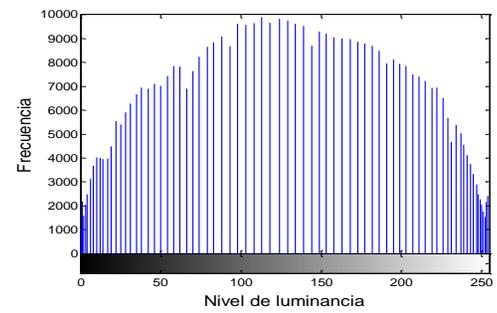
(a)



(b)



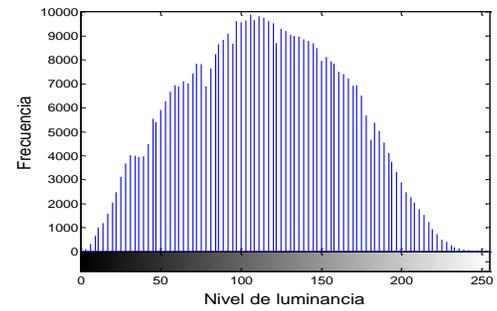
(c)



(d)



(e)

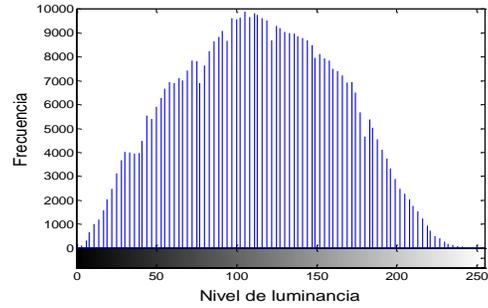


(f)

Figura 4.5 (a) Muestra de fibras tomada bajo pobres condiciones de iluminación (Figura 4.1 (c)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).



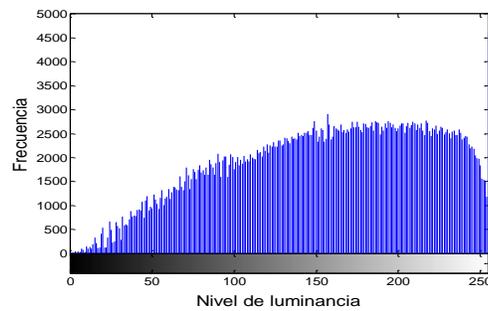
(g)



(h)



(i)



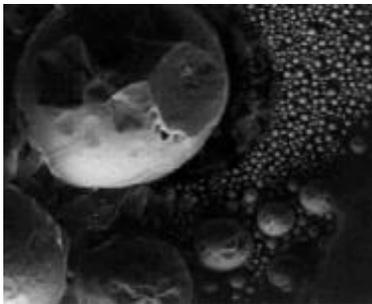
(j)

Figura 4.5(Continuación).

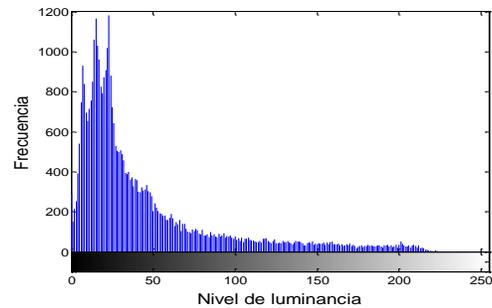
Estadísticamente todas las imágenes cumplen con los parámetros de nivel de media establecidos, siendo Retinex (Figura 4.5 (i)), el método que obtiene el valor de media más alto con 156. Le siguen ecualización de histograma con 129, balance de blancos con 114.7 y balance de blancos con valores neutros 114.1, Figura 4.5 (c), (e) y (g). Los histogramas en los cuatro casos muestran un pico muy amplio que se extiende a lo largo de todo el rango posible, estando el de Retinex, Figura 4.5 (j), el más cargado a la derecha, de ahí el alto valor de media obtenido. El histograma de la imagen mejorada mediante ecualizado tiene unas ligeras concentraciones en los extremos siendo claramente visibles en la Figura 4.5 (d), lo cual denota regiones muy blancas y otras muy oscuras. Para ambos casos de balanceo de blancos se obtiene un histograma más o menos centrado sin mostrar regiones muy claras, Figura 4.5 (f) y (h), proporcionando el mejor resultado según el análisis de histograma. Este es el caso

donde la muestra de bajo contraste original presenta el mejor contraste, y visualmente se obtienen mejores resultados tanto con balance de blancos como con balance de blancos con valores neutros.

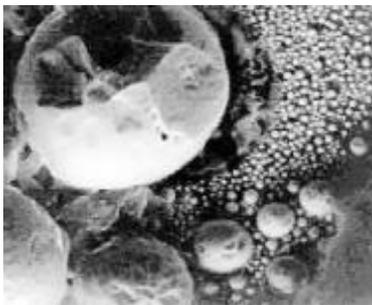
Finalmente, otra muestra para tamaño de grano.



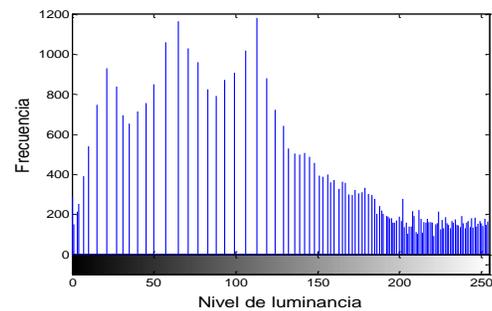
(a)



(b)

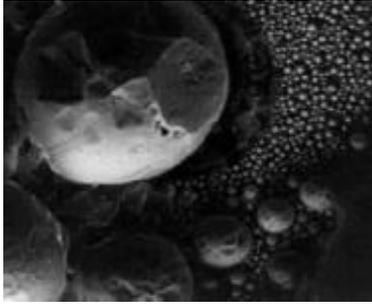


(c)

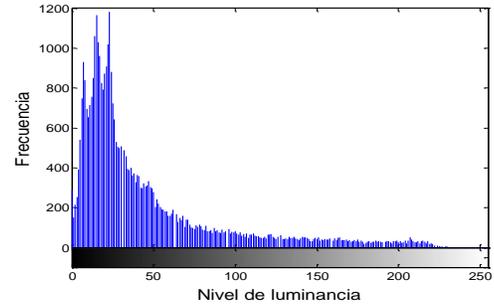


(d)

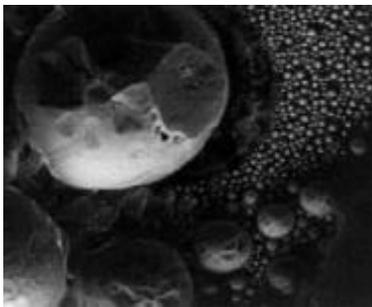
Figura 4.6(a) Muestra utilizada para medición de tamaño de grano tomada bajo pobres condiciones de iluminación (Figura 4.1 (d)), imagen modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i) y sus histogramas de luminancia correspondientes (b, d, f, h, y j).



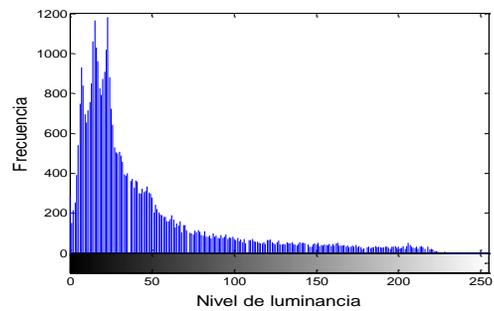
(e)



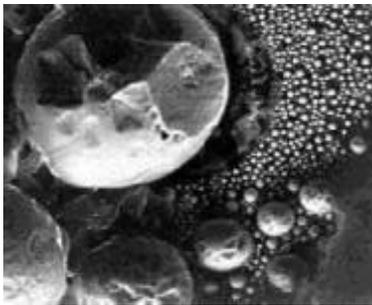
(f)



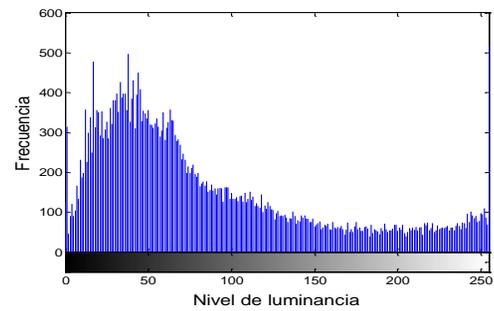
(g)



(h)



(i)



(j)

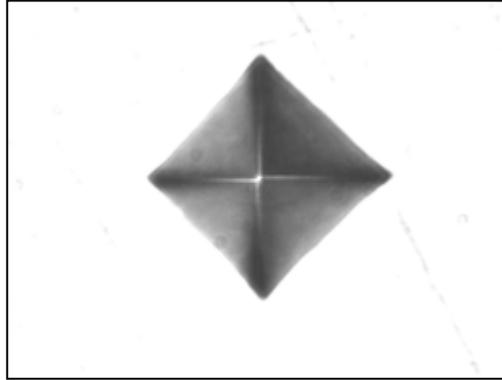
Figura 4.6 (Continuación).

En este caso, solamente las imágenes mejoradas mediante ecualizado de histograma, Figura 4.6 (c), y Retinex, Figura 4.6 (i), quedan dentro de los rangos establecidos anteriormente para buen contraste con 128.2 y 86.0 respectivamente. Es de notar que en este caso los balances de blancos no logran modificar de manera

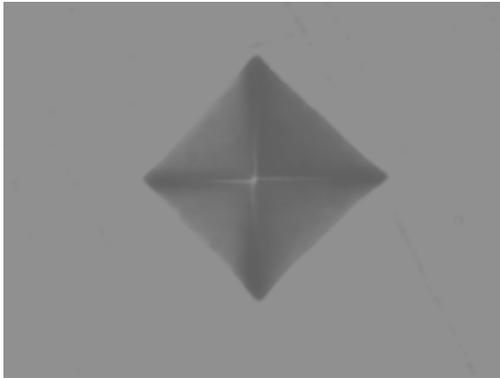
significativa la media resultante, entregando 45.1 para el balance de blancos, Figura 4.6 (e), y 45 para el balanceo con valores neutros, Figura 4.6 (g). De manera similar los histogramas obtenidos por estos últimos dos métodos, Figura 4.6 (f) y (h), dejan de manifiesto que no se realiza una modificación significativa al histograma original, el histograma obtenido mediante ecualizado, Figura 4.6 (d), muestra una distribución a lo largo del rango posible con una gran cantidad de valores en la región brillante, Retinex, por su parte, Figura 4.6 (j), de igual manera distribuye los niveles de intensidad a lo largo del rango posible, mostrando una distribución más uniforme. Visualmente, en las imágenes corregidas por los balanceos de blancos, sigue siendo difícil distinguir entre las diferentes regiones, Figura 4.6 (e) y (g), mientras que en el ecualizado la imagen se vuelve muy brillante, Figura 4.6 (c). La imagen de la corrección hecha por Retinex, Figura 4.6 (i), es en donde se puede diferenciar de mejor manera los diferentes elementos que la componen sin llevar a cabo un esfuerzo mayor.

4.3. Aplicación

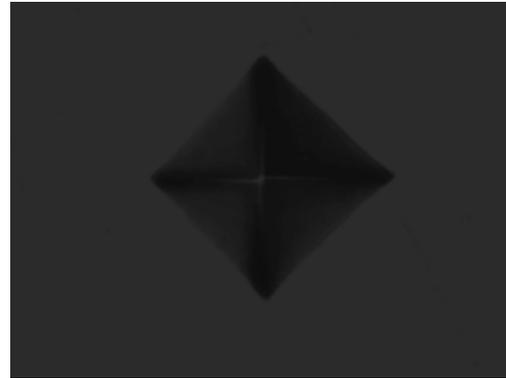
Para comparar los efectos que tiene una muestra tomada bajo malas condiciones de iluminación en un sistema de medición, contra una serie de muestras corregidas mediante los métodos de mejora de contraste en el canal Y' se optó por usar el sistema de medición de microdureza desarrollado por Cayetano (2010) explicado en la sección 1.1. La primer prueba introducida al sistema fue la Figura 4.1 (a) y sus respectivas mejoras, sin embargo, dicha muestra estaba mal preparada y el sistema de medición fallaba a cada intento. Para corregir dicho problema se tomó una muestra bien preparada, Figura 4.7 (a), y se le modificaron sus características de luminancia en un nivel medio y en un nivel extremo, Figura 4.7 (b) y (c). Se procedió a corregir las muestras con los métodos de mejora de contraste y se introdujeron al sistema de medición de microdureza tanto la muestra bien preparada, Figura 4.8 (b), como las muestras corregidas, Figura 4.9 y Figura 4.10.



(a)

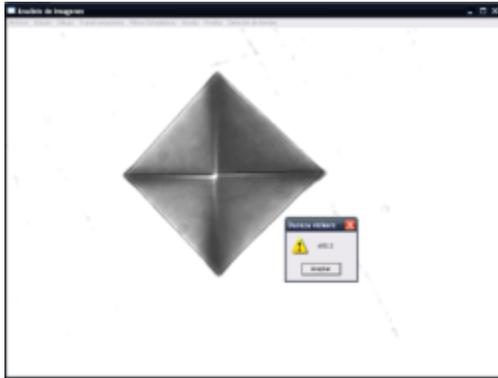


(b)



(c)

Figura 4.7 Muestra bien preparada para medición de microdureza (a) y diferentes variantes de luminancia de la misma (b) y (c).



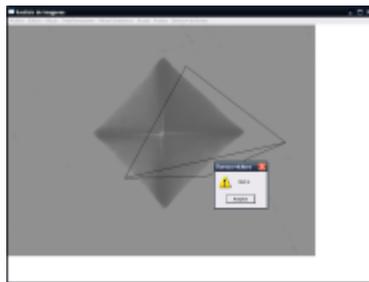
(a)



(b)

Figura 4.8 (a) Figura 4.7 (a) introducida al sistema de medición de microdureza.

La muestra original bien preparada, Figura 4.7 (a), entrega un resultado de medición de 693.3 dureza Vickers, y es visible que el sistema detecta de manera muy eficiente la marca dejada por el indentador, Figura 4.8 (a).

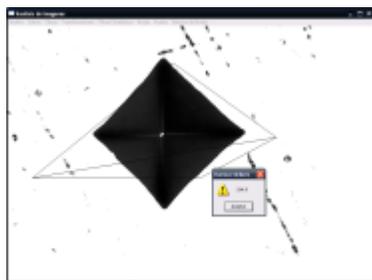


(a)



(b)

Figura 4.9 (a) (Figura 4.7 (b)), modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i), introducidas al sistema de medición de microdureza (b, d, f, h, y j).



(c)



(d)



(e)



(f)

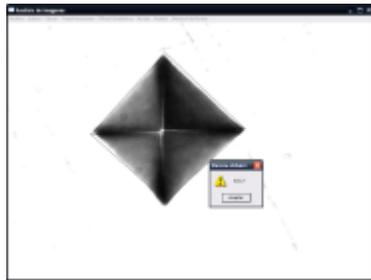


(g)



(h)

Figura 4.9 (Continuación).



(i)

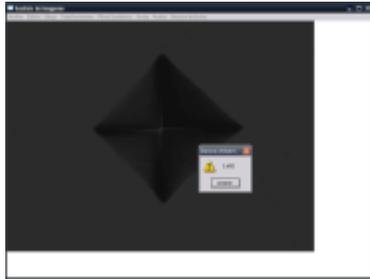


(j)

Figura 4.9 (Continuación).

Para la Figura 4.7 (b) al ser introducida al sistema de medición de microdureza se obtiene un valor de 552.5 dureza Vickers, en el caso de las muestras mejoradas de la Figura 4.9 (c), (e), (g) y (i) al ser introducidas al sistema de medición de microdureza, Figura 4.9 (d), (f), (h) y (j), se obtiene como resultado los valores de 244.9, 631.7, 632 y 631.7 dureza Vickers respectivamente.

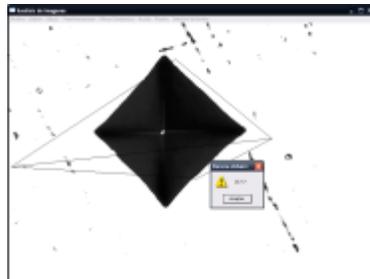
Es notable el hecho de que la imagen mejora mediante ecualizado de histograma, Figura 4.9 (c), al ser introducida al sistema de medición de microdureza, Figura 4.9 (d), entrega un resultado de 244.9 dureza Vickers, el cual es inclusive más bajo que el de la imagen oscurecida de la Figura 4.7 (b). Las muestras mejoradas mediante balance de blancos, Figura 4.9 (e), balance de blancos con valores neutros, Figura 4.9 (g), y Retinex, Figura 4.9 (i), dan resultados aceptables con una diferencia de 61.6 a 61.3 de dureza Vickers respecto al valor entregado por la muestra original de la Figura 4.7 (a). En este caso la muestra mejora mediante balance de blancos con valores neutros entrega un resultado mejor.



(a)



(b)



(c)



(d)



(e)



(f)

Figura 4.10 (a) (Figura 4.7 (c)), modificada mediante ecualización del histograma en Y' (c), balance de blancos en Y' (e), balance de blancos con valores neutros en Y' (g), Retinex en Y' (i), introducidas al sistema de medición de microdureza (b, d, f, h, y j).



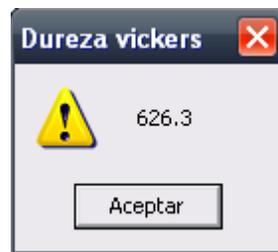
(g)



(h)



(i)



(j)

Figura 4.10 (Continuación).

Quando se introduce la Figura 4.7 (c) al sistema de medición de microdureza no le es posible entregar un resultado debido a que no detecta borde alguno, en el caso de las muestras mejoradas de la Figura 4.10 (c), (e), (i) y (g) al ser introducidas al sistema de medición de microdureza, Figura 4.10 (d), (f), (h) y (j), se obtiene como resultado los valores de 217.7, 626.3, 619.9 y 626.3 dureza Vickers respectivamente.

Una vez más la muestra mejorada mediante ecualización de histograma, Figura 4.10 (c), entrega un resultado bajo de tan solo 217.7, Figura 4.10 (d), en este caso las mejoras realizada mediante balance de blancos y Retinex, Figura 4.10 (e) e (i), entregan ambas un resultado de 626.3 dureza Vickers, Figura 4.10 (f) y (j), lo cual significa un diferencia de 67 dureza Vickers respecto al valor entregado por la muestra original de la Figura 4.7 (a). En el caso del resultado entregado por el balanceo de blancos con valores neutros, Figura 4.10 (h), la diferencia es de 73.4 dureza Vickers.

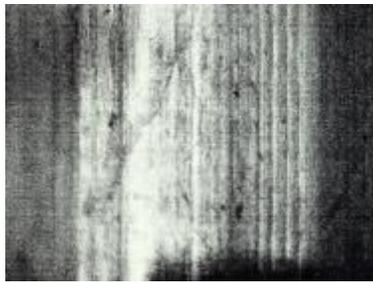
Este ejercicio permite ver como la aplicación de los métodos de mejora de contraste resulta una buena opción para lograr darle uso a aquellas imágenes descartadas por haber sido tomadas bajo pobres condiciones de iluminación, siendo los métodos de balance de blancos, balance de blancos con valores neutros y Retinex las mejores opciones para este tipo de muestras y métodos de medición, quedando relegado la ecualización de histograma que no ofrece buenos resultados.

4.4. Comparativa de aplicación de métodos en canal Y' y RGB

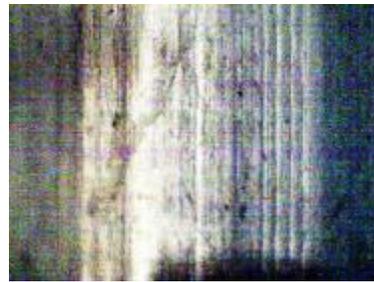
Como se ha venido mencionando, la implementación de los métodos de mejora de contraste en el canal Y' evita que modifiquemos de manera significativa los colores de una muestra como sucede al aplicarlos en los canales RGB . Para ejemplificar lo anterior se tomo una muestra de fibras, Figura 4.11 (a), y se aplico los métodos de mejora de contraste en ambos espacios de color, Figura 4.11 (b-i). En el caso de las mejoras en RGB se observa como el color es modificado sobre todo en la Figura 4.11 (g) e (i), donde se obtiene un color verduzco y rojizo respectivamente.



(a)



(b)



(c)

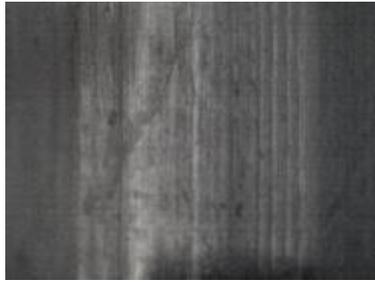


(d)

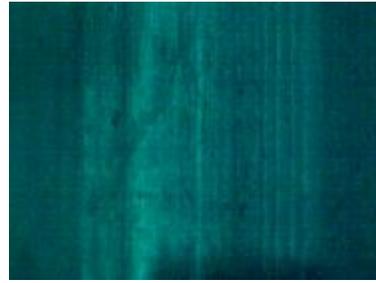


(e)

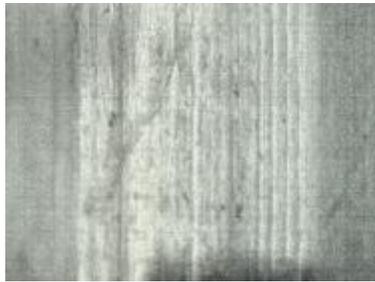
Figura 4.11 (a) Muestra de fibras tomada bajo pobres condiciones de iluminación, modificada mediante ecualización del histograma en Y' (b) y RGB (c), balance de blancos en Y' (d) y RGB (e), balance de blancos con valores neutros en Y' (f) y RGB (g) y Retinex en Y' (h) y RGB (i).



(f)



(g)



(h)



(i)

Figura 4.11 (Continuación).

5. CONCLUSIONES

La búsqueda de un parámetro que nos permitiera identificar algún indicio de buen o bajo contraste de una manera analítica mediante información estadística, trajo consigo la necesidad de buscar la ayuda de un experto que ayudara a diferenciar esas características en imágenes de muestras metalográficas, sin embargo, esto sigue siendo un aspecto difícil de determinar dado que lo importante es que el sistema de medición sea lo suficientemente sensible para poder distinguir esas pequeñas diferencias que para el ojo humano pudiesen no serlo tanto.

De igual forma el análisis de contraste mediante el uso del histograma requiere de poder tener acceso directo con la muestra para poder hacer un análisis en conjunto, si bien nos ofrece un buen indicio de la cantidad de luminancia de una muestra, también se requiere de ciertos criterios personales para poder llevar a cabo este análisis.

Además, se comprobó el hecho de que unas malas condiciones de iluminación influyen grandemente en los resultados entregados por un método de medición metalográfico, sin embargo con la aplicación los métodos de mejora de contraste se pueden llegar a salvar, hasta cierto punto y bajo ciertas condiciones, las muestras con bajo contraste.

El poder trabajar ambos los métodos de mejora de contraste tanto en el espacio de color RGB y el espacio $u'v'Y'$, permite poder observar claramente las ventajas de trabajar sobre el canal de luminancia en lugar de las bandas RGB , donde que si bien se podía observar un histograma de luminancia similar, al trabajar en RGB se modifica el color de nuestra imagen original.

Si se parte del hecho de que a toda imagen es posible aplicarle algún método de mejora de contraste, no en todos los casos se obtienen buenos resultados, esto depende de la naturaleza de la imagen que si bien puede ser totalmente oscura o

uniforme como fue en el caso de nuestro estudio, pudiera presentar combinación de regiones con bajo y buen contraste con lo cual los métodos de mejora de contraste aplicados tienen ciertos problemas.

6. REFERENCIAS

Adobe Developers Association, 1992. TIFF Revision 6.0. Adobe Systems Incorporated.

American Anthropological Association, 2009, AAA Style Guide.

Benítez-Rangel J. P., Morales Hernández L. A., Domínguez González A., Herrera Ruíz G., (Bajo Revisión). Detección Automática del límite de grano usando una aproximación de interceptos. Artículo de Ingeniería. Universidad Autónoma de Querétaro-Facultad de Ingeniería.

Cardei V. C., Funt B. y Barnadr K., 1999. White point estimation for uncalibrated images. In Proceedings of the Seventh Color Imaging Conference (CIC'99) Imaging Science and Technology Society. Pág. 97-100.

Cayetano García R., 2010, Sistema de Medición para determinar la microdureza en acero por visión artificial. Tesis de ingeniería. Universidad Autónoma de Querétaro-Facultad de Ingeniería.

Ebner M., 2003. Combining White-patch Retinex and the gray world assumption to achieve color constancy for multiple illuminants. *Pattern Recognition*, Volume 2781/2003, Pág. 60-67.

Espino Gudiño Ma. del C., Santillán I. y Terol Villalobos I. R., 2007. *Morphological multiscale contrast approach for gray and color images consistent with human visual perception. Optical Engineering* 46(6):067003-067003-14. Universidad Autónoma de Querétaro-Facultad de Ingeniería.

Granados Morales H. E., Morales Hernández L. A., Delgado Rosas M., Ramírez Baltazar J. C., 2008. Efecto de la austenita retenida sobre el esfuerzo residual en acero SAE 8620 carbonitrurado. Artículo de Ingeniería. Universidad Autónoma de Querétaro-Facultad de Ingeniería.

Lucchese L. y Mitra S. K., 2004. A new class of chromatic filters for color image processing. - Theory and applications. IEEE Transactions on Image Processing. Papers, 13(4), 534-538.

McCann J, Funt B., y Ciurea F., 2004. *Retinex in Matlab*. *Journal of the Electronic Imaging*, Pág 48-57.

Mendiola Santibañez J. D. y Terol Villalobos I. R., 2005. *Quantifying Contrast Methods Through Morphological Gradient*. *Computación y Sistemas*, abril-junio, año/vol. 8, numero 004, Instituto Politécnico Nacional, Distrito Federal, México Pág 317-333.

Morales Hernández L. A., Benítez Rangel J. P., Domínguez González A. y Herrera Ruiz G., 2007. Medición Automática de grafito en aceros de fundición por procesamiento de imágenes. Tercer Congreso Internacional de Ingeniería. Querétaro, México.

Morales Hernández L. A., Barajas Alarcón M. R., Benítez Rangel J. P., Ortiz Vargas W., 2009. Caracterización de metalografías de aluminio por medio de visión por computadora. Quinto Congreso Internacional de Ingeniería. Querétaro, México.

Mukhopadhyay Susanta y Chanda Bhabatosh, 2000. A multiscale morphological approach to local contrast enhancement. *Signal Processing*. Volume 80, Issue 4. Pág. 685–696, ISSN: 0165-1684.

Pajares Martinsanz G. y de la Cruz García J. M., 2004. *Visión por Computador Imágenes Digitales y Aplicaciones* (Primera reimpresión). Editorial Alfaomega, Colombia.

Peregrina Barreto H. y Terol Villalobos I., 2010. *Morphological rational multi-scale algorithm for color contrast enhancement*. Proc. SPIE Vol. 7532(1).Pág. 75320Q.

Russ J. C., 2007. *The Image Processing Handbook* (Fifth Edition). Taylor & Francis. USA.

Serrano Gotarredona M. del C., Acha Piñero B. y Fondón García I., 2005. Calibración de Imágenes en Color en Condiciones de Iluminación no Controladas. XX Simposium Nacional de la Unión Científica Internacional de Radio: Programa, Libro de Resúmenes. Unión Científica Internacional de Radio. Comité Español. Simposium Nacional. Núm. 20. Gandía. Editorial de la Universidad Politécnica de Valencia. Pág. 184-188. ISBN: 84-9705-859-3.

Solís Arellano J. O., 2008. Análisis de la microestructura para detectar incrustaciones de grafito en el acero por medio de análisis de imágenes. Tesis de Ingeniería. Universidad Autónoma de Querétaro-Facultad de Ingeniería.

<http://www.cie.co.at/>, Consultada el 23 de Febrero de 2010

<http://www.gnu.org/home.es.html>, Consultada el 23 de Febrero de 2010.

<http://www.gnu.org/licenses/licenses.es.html>, Consultada el 23 de Febrero de 2010.

<http://www.libtiff.org/index.html>, Consultada el 23 de Febrero de 2010.

7. ANEXOS

7.1. Código Cont_Ench.c de llamado de las funciones de la librería Cont_Ench.h

```
// Programa principal que hace el llamado a los diferentes metodos de mejora de
// contraste contenidos en el archivo Cont_Ench.h
```

```
// Entrada: Imagen en formato TIFF (Nombre.extension *.tif)
// Salida: Imagen mejorada en formato TIFF con nombre que hace referencia al
// metodo empleado
```

```
#include "rw_tiff.h" // Libreria para lectura y escritura de imagenes TIFF
#include "Cont_Ench.h" // Libreria de mejora de contraste
```

```
char fname[30]; // Variable para el nombre de la imagen de entrada
int opcion; // Variable para la opcion del menu elegida
```

```
int inicio(); // Funcion para la presentacion del proyecto
char getfname(); // Funcion para obtener nombre de la imagen a mejorar
int menu(); // Funcion del menu con los metodos de mejora de contraste
void eleccion(); // Funcion para eleccion del metodo de mejora de contraste
// deseado
```

```
//////////////////// Programa principal //////////////////////
int main (void)
{
    inicio();
    getfname();
    menu(fname);
    do{
```

```

    getoption();
    eleccion(opcion);
}while(opcion<1 || opcion>8);

return 0;
}
////////////////////////////////////
////
////////////////////////////////////      Presentacion      //////////////////////////////////
int inicio(void)
{
    printf("\n\t Universidad Autonoma de Queretaro");
    printf("\n\t Facultad de Ingenieria");
    printf("\n\t Campus San Juan del Rio");
    printf("\n\t UAQ FI SJR");
    printf("\n\n\tMejora de contraste de imagenes de microscopio");
    printf("\n\t Marco Antonio Garduno Ramon");

return 0;
}
////////////////////////////////////
////
//////////      Nombre de la imagen a mejorar      //////////
char getfname()
{
    printf("\n\n\tNombre del archivo (*.tif): ");
    scanf("%s",&fname);

return *fname;
}
////////////////////////////////////
////
//////////      Menu de seleccion      //////////
int menu(void)
{
    printf("\n\n\tMetodos de mejora de contraste");

```

```

printf("\n\n\t(1) Ecualizacion del histograma RGB");
printf("\n\t(2) Ecualizacion del histograma Y");
printf("\n\t(3) Balance de blancos RGB");
printf("\n\t(4) Balance de blancos Y");
printf("\n\t(5) Balance de blancos valores neutros RGB");
printf("\n\t(6) Balance de blancos valores neutros Y");
printf("\n\t(7) Retinex RGB");
printf("\n\t(8) Retinex Y");

return 0;
}
////////////////////////////////////
////
//////////////////////////////////// Nombre de la imagen a mejorar //////////////////////////////////
int getopcion()
{
printf("\n\n\tElige una opcion: ");
scanf("%i",&opcion);

return opcion;
}
////////////////////////////////////
////
//////////////////////////////////// Eleccion de metodo de mejora de contraste
////////////////////////////////////
void eleccion()
{
switch(opcion)
{
case 1:
hist_equa_rgb(fname);
break;

case 2:
hist_equa_y(fname);
break;
}
}

```

```
case 3:
    white_patch_rgb(fname);
break;

case 4:
    white_patch_y(fname);
break;

case 5:
    balance_neutros_rgb(fname);
break;

case 6:
    balance_neutros_y(fname);
break;

case 7:
    retinex_rgb(fname);
break;

case 8:
    retinex_y(fname);
break;

default:
    printf("\n\tOpcion erronea");
break;
}
}
////////////////////////////////////
/////
```

7.2. Librería Cont_Ench.h

```
// Libreria de metodos de mejora de contraste
// Incluye los métodos de ecualización del histograma, balance de blancos,
// balance de blancos con valores neutros y retinex, tanto para el espacio de
// color RGB como para u'v'Y'

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <tiffio.h>
#define MinAB(A,B) (((A)<(B))?A):(B));

/*----- METODOS DE MEJORA DE CONTRASTE -----*/
int hist_equa_rgb(char fname[]);
int hist_equa_y(char fname[]);
int white_patch_rgb(char fname[]);
int white_patch_y(char fname[]);
int balance_neutros_rgb(char fname[]);
int balance_neutros_y(char fname[]);
int retinex_rgb(char fname[]);
int retinex_y(char fname[]);

int gmax=255, gmin=0;
ImgColor img;

/*-----*/
int maximos(float vect[], int n);
int minimos(float vect[], int n);
float ajuste(float div, int min, float i);
void CompareWith(int s_row, int s_col, float *OP, float *bufferRGB, float Maximum, int
size, int w, int h);
void CompareWith2(int s_row, int s_col, float *OP, float *bufferRGB, float Maximum,
int size, int w, int h);
inline float Max(float *vector, int vsize);
/*-----*/
```

```

////////// Ecuacion de histograma RGB //////////
int hist_equa_rgb(char fname[])
{
    int i, j, k;
    int fg[256];
    float ng[256], pg[256], ppg[256];

    printf("\n\tEcuacion del histograma RGB.");
    ReadTIFF(fname, &img);

//////////          Calculo de n(g)          //////////
    for(k=0; k<=2; k++){
        for(j=gmin; j<=gmax; j++){
            ng[j]=0;
        }
        if(k==0){
            for(i=0; i<img.size; i++){
                j=img.R[i];
                ng[j]+=1;
            }
        }
        else if(k==1){
            for(i=0; i<img.size; i++){
                j=img.G[i];
                ng[j]+=1;
            }
        }
        else if(k==2){
            for(i=0; i<img.size; i++){
                j=img.B[i];
                ng[j]+=1;
            }
        }
}
//////////          Calculo de p(g)          //////////
    for(j=gmin; j<=gmax; j++){
        pg[j]=(ng[j]/img.size);
    }
}

```

```

    }
    ////////////      Calculo de Pp(g)      ////////////
    ppg[gmin]=pg[gmin];

    for(j=gmin+1; j<=gmax; j++){
        ppg[j]=pg[j]+ppg[j-1];
    }
    ////////////      Calculo de F(g)      ////////////
    for(j=gmin; j<=gmax; j++){
        fg[j]=(gmax*ppg[j]);
    }
    ////////////      Asignacion de F(g)      ////////////
    if(k==0){
        for(i=0; i<img.size; i++){
            j=img.R[i];
            img.R[i]=fg[j];
        }
    }
    else if(k==1){
        for(i=0; i<img.size; i++){
            j=img.G[i];
            img.G[i]=fg[j];
        }
    }
    else if(k==2){
        for(i=0; i<img.size; i++){
            j=img.B[i];
            img.B[i]=fg[j];
        }
    }
}

WriteTIFF("hist_equa_rgb.tif", &img);
FreeImgColor(&img);
}
////////////////////////////////////

```

```

////////// Ecuacion de histograma Y //////////
int hist_equa_y(char fname[])
{
    int i, j;
    int fg[256];
    int Ymax, Rmax, Gmax, Bmax, Ymin, Rmin, Gmin, Bmin;
    float divY, divR, divG, divB;
    float ng[256], pg[256], ppg[256];

    printf("\n\tEcuacion del histograma Y.");
    ReadTIFF(fname, &img);

    DinamicAsign_float(&img.Y, img.size, "Y");
    DinamicAsign_float(&img.U, img.size, "U");
    DinamicAsign_float(&img.V, img.size, "V");

    for(i=0; i<img.size; i++){
        img.Y[i]=(( 0.299*img.R[i])+( 0.587*img.G[i])+( 0.114*img.B[i]));
        img.U[i]=((-0.14713*img.R[i])-(0.28886*img.G[i])+( 0.436*img.B[i]));
        img.V[i]=(( 0.615*img.R[i])-( 0.51499*img.G[i])-( 0.10001*img.B[i]));
    }

    for(j=gmin; j<=gmax; j++){
        ng[j]=0;
    }

    for(i=0; i<img.size; i++){
        j=img.Y[i];
        ng[j]+=1;
    }

    for(j=gmin; j<=gmax; j++){
        pg[j]=(ng[j]/img.size);
    }
}

```

```

ppg[gmin]=pg[gmin];

for(j=gmin+1; j<=gmax; j++){
    ppg[j]=pg[j]+ppg[j-1];
}

for(j=gmin; j<=gmax; j++){
    fg[j] = (gmax*ppg[j]);
}

for(i=0; i<img.size; i++){
    j=img.Y[i];
    img.Y[i]=fg[j];
}

Ymax = maximos(img.Y,img.size);
Ymin = minimos(img.Y,img.size);

divY=Ymax-Ymin;

for(i=0; i<img.size; i++){
    img.Y[i]=ajuste(divY,Ymin,img.Y[i]);
}

for(i=0; i<img.size; i++){
    img.R[i]=(( 1*img.Y[i])-( 0*img.U[i])+( 1.135983*img.V[i]));
    img.G[i]=(( 1*img.Y[i])-( 0.39465*img.U[i])-( 0.58060*img.V[i]));
    img.B[i]=(( 1*img.Y[i])+( 2.03211*img.U[i])+( 0*img.V[i]));
}

Rmax=maximos(img.R,img.size);
Gmax=maximos(img.G,img.size);
Bmax=maximos(img.B,img.size);

Rmin=minimos(img.R,img.size);
Gmin=minimos(img.G,img.size);

```

```

Bmin=minimos(img.B,img.size);

divR=Rmax-Rmin;
divG=Gmax-Gmin;
divB=Bmax-Bmin;

for(i=0; i<img.size; i++){
    img.R[i]=ajuste(divR,Rmin,img.R[i]);
    img.G[i]=ajuste(divG,Gmin,img.G[i]);
    img.B[i]=ajuste(divB,Bmin,img.B[i]);
}

for(i=0; i<img.size; i++){
    if(img.R[i]>255)
        img.R[i]=gmax;
    if(img.G[i]>255)
        img.G[i]=gmax;
    if(img.B[i]>255)
        img.B[i]=gmax;
}

WriteTIFF("hist_equa_y.tif", &img);
FreeImgColor(&img);
}
////////////////////////////////////

//////////      Balance de blancos RGB      //////////
int white_patch_rgb(char fname[])
{
    int i;
    int Rmax, Gmax, Bmax, Rmin, Gmin, Bmin;
    float divR, divG, divB, d, dmin, Rx, Gx, Bx;

    printf("\n\tBalance de blancos RGB.");

    ReadTIFF(fname, &img);

```

```

//////////          Distancias          //////////
dmin=442.0;
for(i=0; i<img.size; i++){
    d=sqrt(pow((255.0-img.R[i]),2) + pow((255.0-img.G[i]),2) + pow((255.0-
img.B[i]),2) );
    if(d<dmin){
        dmin=d;
        Rx=img.R[i];
        Gx=img.G[i];
        Bx=img.B[i];
    }
}
//////////          Balance de blancos          //////////
for(i=0; i<img.size; i++){
    img.R[i] = ((float)img.R[i]/Rx)*gmax;
    img.G[i] = ((float)img.G[i]/Gx)*gmax;
    img.B[i] = ((float)img.B[i]/Bx)*gmax;
}

Rmax=maximos(img.R,img.size);
Gmax=maximos(img.G,img.size);
Bmax=maximos(img.B,img.size);

Rmin=minimos(img.R,img.size);
Gmin=minimos(img.G,img.size);
Bmin=minimos(img.B,img.size);

divR=Rmax-Rmin;
divG=Gmax-Gmin;
divB=Bmax-Bmin;

for(i=0; i<img.size; i++){
    img.R[i]=ajuste(divR,Rmin,img.R[i]);
    img.G[i]=ajuste(divG,Gmin,img.G[i]);
    img.B[i]=ajuste(divB,Bmin,img.B[i]);
}

```

```

WriteTIFF("white_patch_rgb.tif", &img);
FreeImgColor(&img);
}
////////////////////////////////////

//////////      Balance de blancos Y      //////////
int white_patch_y(char fname[])
{
    int i;
    int Ymax, Rmax, Gmax, Bmax, Ymin, Rmin, Gmin, Bmin;
    float Yx, divY, divR, divG, divB;

    printf("\n\tBalance de blancos Y.");

    ReadTIFF(fname, &img);

    DinamicAsign_float(&img.Y, img.size, "Y");
    DinamicAsign_float(&img.U, img.size, "U");
    DinamicAsign_float(&img.V, img.size, "V");

    for(i=0; i<img.size; i++){
        img.Y[i]=(( 0.299*img.R[i])+( 0.587*img.G[i])+( 0.114*img.B[i]));
        img.U[i]=((-0.14713*img.R[i])-(0.28886*img.G[i])+( 0.436*img.B[i]));
        img.V[i]=(( 0.615*img.R[i])-( 0.51499*img.G[i])-( 0.10001*img.B[i]));
    }

    Yx=maximos(img.Y, img.size);

    for(i=0; i<img.size; i++){
        img.Y[i] = ((float)img.Y[i]/Yx)*gmax;
    }

    Ymax=maximos(img.Y, img.size);
    Ymin=minimos(img.Y, img.size);

```

```

divY=Ymax-Ymin;

for(i=0; i<img.size; i++){
    img.Y[i]=ajuste(divY,Ymin,img.Y[i]);
}

for(i=0; i<img.size; i++){
    img.R[i]=(( 1*img.Y[i])-( 0*img.U[i])+( 1.135983*img.V[i]));
    img.G[i]=(( 1*img.Y[i])-( 0.39465*img.U[i])-( 0.58060*img.V[i]));
    img.B[i]=(( 1*img.Y[i])+( 2.03211*img.U[i])+( 0*img.V[i]));
}

Rmax=maximos(img.R,img.size);
Gmax=maximos(img.G,img.size);
Bmax=maximos(img.B,img.size);

Rmin=minimos(img.R,img.size);
Gmin=minimos(img.G,img.size);
Bmin=minimos(img.B,img.size);

divR=Rmax-Rmin;
divG=Gmax-Gmin;
divB=Bmax-Bmin;

for(i=0; i<img.size; i++){
    img.R[i]=ajuste(divR,Rmin,img.R[i]);
    img.G[i]=ajuste(divG,Gmin,img.G[i]);
    img.B[i]=ajuste(divB,Bmin,img.B[i]);
}

for(i=0; i<img.size; i++){
    if(img.R[i]>255)
        img.R[i]=gmax;
    if(img.G[i]>255)
        img.G[i]=gmax;
    if(img.B[i]>255)

```

```

        img.B[i]=gmax;
    }

    WriteTIFF("white_patch_y.tif", &img);
    FreeImgColor(&img);
}
////////////////////////////////////

int balance_neutros_rgb(char fname[])
{
    int i, j;
    int porc_black, porc_white, max, min, rango;
    int new_reference_R=0, new_reference_G=0, new_reference_B=0;
    int Ymax, Rmax, Gmax, Bmax, Ymin, Rmin, Gmin, Bmin;
    float Yx, Rx, Gx, Bx, divY, divR, divG, divB;

    printf("\n\tBalance de blancos con valores neutros RGB.");
    ReadTIFF(fname, &img);

    DinamicAsign_float(&img.Y, img.size, "Y");
    DinamicAsign_float(&img.U, img.size, "U");
    DinamicAsign_float(&img.V, img.size, "V");

    for(i=0; i<img.size; i++){
        img.Y[i]=(( 0.299*img.R[i])+( 0.587*img.G[i])+( 0.114*img.B[i]));
        img.U[i]=((-0.14713*img.R[i])-(0.28886*img.G[i])+( 0.436*img.B[i]));
        img.V[i]=(( 0.615*img.R[i])-( 0.51499*img.G[i])-( 0.10001*img.B[i]));
    }

    max=maximos(img.Y,img.size);
    min=minimos(img.Y,img.size);

    rango=max-min;

    porc_white=max - 0.20*rango;
    porc_black=min + 0.10*rango;

```

```

j = 1;
for(i=0; i<img.size; i++){
  if(img.Y[i]<= porc_black || img.Y[i]>= porc_white)
    new_reference_R += img.R[i];
    new_reference_G += img.G[i];
    new_reference_B += img.B[i];
  j++;
}

```

```

new_reference_R /=j;
new_reference_G /=j;
new_reference_B /=j;

```

```

Rx=new_reference_R;
Gx=new_reference_G;
Bx=new_reference_B;

```

```

for(i=0; i<img.size; i++){
  img.R[i] = ((float)img.R[i]/Rx)*gmax;
  img.G[i] = ((float)img.G[i]/Gx)*gmax;
  img.B[i] = ((float)img.B[i]/Bx)*gmax;
}

```

```

Rmax=maximos(img.R,img.size);
Gmax=maximos(img.G,img.size);
Bmax=maximos(img.B,img.size);

```

```

Rmin=minimos(img.R,img.size);
Gmin=minimos(img.G,img.size);
Bmin=minimos(img.B,img.size);

```

```

divR=Rmax-Rmin;
divG=Gmax-Gmin;
divB=Bmax-Bmin;

```

```

for(i=0; i<img.size; i++){
    img.R[i]=ajuste(divR,Rmin,img.R[i]);
    img.G[i]=ajuste(divG,Gmin,img.G[i]);
    img.B[i]=ajuste(divB,Bmin,img.B[i]);
}

for(i=0; i<img.size; i++){
    if(img.R[i]>255)
        img.R[i]=gmax;
    if(img.G[i]>255)
        img.G[i]=gmax;
    if(img.B[i]>255)
        img.B[i]=gmax;
}

WriteTIFF("balance_neutros_rgb.tif", &img);
FreeImgColor(&img);
}

int balance_neutros_y(char fname[])
{
    int i, j;
    int porc_black, porc_white, max, min, rango;
    int new_reference=0;
    int Ymax, Rmax, Gmax, Bmax, Ymin, Rmin, Gmin, Bmin;
    float Yx, divY, divR, divG, divB;

    printf("\n\tBalance de blancos con valores neutros Y.");

    ReadTIFF(fname, &img);

    DinamicAsign_float(&img.Y, img.size, "Y");
    DinamicAsign_float(&img.U, img.size, "U");
    DinamicAsign_float(&img.V, img.size, "V");

    for(i=0; i<img.size; i++){

```

```

img.Y[i]=(( 0.299*img.R[i])+( 0.587*img.G[i])+( 0.114*img.B[i]));
img.U[i]=((-0.14713*img.R[i])-(0.28886*img.G[i])+( 0.436*img.B[i]));
img.V[i]=(( 0.615*img.R[i])-( 0.51499*img.G[i])-( 0.10001*img.B[i]));
}

```

```

max=maximos(img.Y,img.size);
max=minimos(img.Y,img.size);

```

```

rango=max-min;

```

```

porc_white=max - 0.2*rango;
porc_black=min + 0.1*rango;

```

```

j = 1;
for(i=0; i<img.size; i++){
if(img.Y[i]<= porc_black || img.Y[i]>= porc_white)
new_reference += img.Y[i];
j++;
}
new_reference /=j;

```

```

Yx = new_reference;

```

```

for(i=0; i<img.size; i++){
img.Y[i] = ((float)img.Y[i]/Yx)*gmax;
}

```

```

Ymax=maximos(img.Y, img.size);
Ymin=minimos(img.Y, img.size);

```

```

divY=Ymax-Ymin;

```

```

for(i=0; i<img.size; i++){
img.Y[i]=ajuste(divY,Ymin,img.Y[i]);
}

```

```

for(i=0; i<img.size; i++){
    img.R[i]=(( 1*img.Y[i])-( 0*img.U[i])+( 1.135983*img.V[i]));
    img.G[i]=(( 1*img.Y[i])-( 0.39465*img.U[i])-( 0.58060*img.V[i]));
    img.B[i]=(( 1*img.Y[i])+( 2.03211*img.U[i])+( 0*img.V[i]));
}

Rmax=maximos(img.R,img.size);
Gmax=maximos(img.G,img.size);
Bmax=maximos(img.B,img.size);

Rmin=minimos(img.R,img.size);
Gmin=minimos(img.G,img.size);
Bmin=minimos(img.B,img.size);

divR=Rmax-Rmin;
divG=Gmax-Gmin;
divB=Bmax-Bmin;

for(i=0; i<img.size; i++){
    img.R[i]=ajuste(divR,Rmin,img.R[i]);
    img.G[i]=ajuste(divG,Gmin,img.G[i]);
    img.B[i]=ajuste(divB,Bmin,img.B[i]);
}

for(i=0; i<img.size; i++){
    if(img.R[i]>255)
        img.R[i]=gmax;
    if(img.G[i]>255)
        img.G[i]=gmax;
    if(img.B[i]>255)
        img.B[i]=gmax;
}

WriteTIFF("balance_neutros_y.tif", &img);
FreeImgColor(&img);

```

```
}
```

```
//////////          Retinex          //////////  
int retinex_rgb(char fname[]){  
    register int a, b, i, j;  
    int nIteraciones=4, iter, w2, pos, pos2, shift;  
    int Rmax, Gmax, Bmax, Rmin, Gmin, Bmin;  
    float l2, Maximum, *OP, *bufferRGB;  
    float divR, divG, divB;  
    uchar *outbuf;  
  
    printf("\n\tRetinex RGB.");  
    ReadTIFF(fname, &img);  
  
    DinamicAsign_float(&bufferRGB, img.size*3, "bufferRGB");  
    DinamicAsign_float(&OP, img.size*3, "OP");  
    w2 = img.W*3;  
  
    //pasando las bandas RGB al buffer RGB  
    for(i=0; i<img.H; i++){  
        for(j=0, a=img.W, b=img.W*2; j<img.W; j++, a++, b++){  
            pos = i * img.W + j;  
            pos2 = i * w2 + j;  
            *(bufferRGB + pos2) = *(img.R + pos);  
            pos2 = i * w2 + a;  
            *(bufferRGB + pos2) = *(img.G + pos);  
            pos2 = i * w2 + b;  
            *(bufferRGB + pos2) = *(img.B + pos);  
        }  
    }  
  
    //obtiene el buffer normalizado y logaritmico  
    for (i=0; i<img.size*3; i++){  
        if(*(bufferRGB + i) == 0.0) *(bufferRGB + i) = 1.0;  
        *(bufferRGB + i) /= 255.0;  
        *(bufferRGB + i) = log(*(bufferRGB + i));  
    }  
}
```

```

}

//valor maximo de la imagen
Maximum = Max(bufferRGB, img.size * 3);
shift = MinAB(w2, img.H);
shift = pow(2, (log(shift) / log(2)) - 1);
//printf("\n\tMaximum = %f\n\tShift = %d\n", Maximum, shift);

//inicializa buffer auxiliar con el maximo
for(i=0; i < img.size*3; i++) *(OP + i)=Maximum;

while(fabs(shift) >= 1){
    for(iter=0; iter<nIteraciones; iter++){
        CompareWith(0, shift, OP, bufferRGB, Maximum, img.size, img.W, img.H);
        CompareWith(shift, 0, OP, bufferRGB, Maximum, img.size, img.W, img.H);
    }
    shift = -shift / 2;
}

//pasa de buffer normalizado logaritmico a buffer RGB
for(i=0; i<img.size*3; i++) *(OP + i) = exp(*(OP + i)) * 255;

//descompone el bufferRGB en las tres bandas resultantes
for(i=0; i<img.H; i++){
    for(j=0, a=img.W, b=img.W*2; j<img.W; j++, a++, b++){
        pos = i * img.W + j;
        pos2 = i * w2 + j;
        *(img.R + pos) = *(OP + pos2);
        pos2 = i * w2 + a;
        *(img.G + pos) = *(OP + pos2);
        pos2 = i * w2 + b;
        *(img.B + pos) = *(OP + pos2);
    }
}

Rmax=maximos(img.R,img.size);

```

```

Gmax=maximos(img.G,img.size);
Bmax=maximos(img.B,img.size);

Rmin=minimos(img.R,img.size);
Gmin=minimos(img.G,img.size);
Bmin=minimos(img.B,img.size);

divR=Rmax-Rmin;
divG=Gmax-Gmin;
divB=Bmax-Bmin;

for(i=0; i<img.size; i++){
    img.R[i]=ajuste(divR,Rmin,img.R[i]);
    img.G[i]=ajuste(divG,Gmin,img.G[i]);
    img.B[i]=ajuste(divB,Bmin,img.B[i]);
}

for(i=0; i<img.size; i++){
    if(img.R[i]>255)
        img.R[i]=gmax;
    if(img.G[i]>255)
        img.G[i]=gmax;
    if(img.B[i]>255)
        img.B[i]=gmax;
}

//guarda las bandas como imagen TIFF
WriteTIFF("retinex_rgb.tif", &img);
free(OP);    OP = NULL;
free(bufferRGB); bufferRGB = NULL;
FreeImgColor(&img);
}
////////////////////////////////////

int retinex_y(char fname[])
{

```

```

register int a, b, i, j;
int nIteraciones=4, iter, pos, pos2, shift;
int Ymax, Rmax, Gmax, Bmax, Ymin, Rmin, Gmin, Bmin;
float l2, Maximum, *OP, *bufferY;
float divY, divR, divG, divB;
uchar *outbuf;

printf("\n\tRetinex Y.");
ReadTIFF(fname, &img);

DinamicAsign_float(&img.Y, img.size, "Y");
DinamicAsign_float(&img.U, img.size, "U");
DinamicAsign_float(&img.V, img.size, "V");

for(i=0; i<img.size; i++){
    img.Y[i]=(( 0.299*img.R[i])+ ( 0.587*img.G[i])+ ( 0.114*img.B[i]));
    img.U[i]=((-0.14713*img.R[i])-(0.28886*img.G[i])+ ( 0.436*img.B[i]));
    img.V[i]=(( 0.615*img.R[i])-( 0.51499*img.G[i])-( 0.10001*img.B[i]));
}

DinamicAsign_float(&bufferY, img.size, "bufferY");
DinamicAsign_float(&OP, img.size, "OP");

//pasando la banda Y al bufferY
for(i=0; i<img.size; i++){
    bufferY[i]=img.Y[i];
}
//obtiene el buffer normalizado y logaritmico
for (i=0; i<img.size; i++){
    if(*(bufferY + i) == 0.0) *(bufferY + i) = 1.0;
    *(bufferY + i) /= 255.0;
    *(bufferY + i) = log(*(bufferY + i));
}

//valor maximo de la imagen
Maximum = Max(bufferY, img.size);

```

```

shift = MinAB(img.W, img.H);
shift = pow(2, (log(shift) / log(2)) - 1);
//printf("\n\tMaximum = %f\n\tShift = %d\n", Maximum, shift);

//inicializa buffer auxiliar con el maximo
for(i=0; i < img.size; i++) *(OP + i)=Maximum;

while(fabs(shift) >= 1){
    for(iter=0; iter<nIteraciones; iter++){
        CompareWith2(0, shift, OP, bufferY, Maximum, img.size, img.W, img.H);
        CompareWith2(shift, 0, OP, bufferY, Maximum, img.size, img.W, img.H);
    }
    shift = -shift / 2;
}

//pasa de buffer normalizado logaritmico a bufferY
for(i=0; i<img.size; i++) *(OP + i) = exp(*(OP + i)) * 255;

//descompone el bufferY en la banda Y resultante
for(i=0; i<img.size; i++){
    img.Y[i]=OP[i];
}

Ymax=maximos(img.Y, img.size);
Ymin=minimos(img.Y, img.size);

divY=Ymax-Ymin;

for(i=0; i<img.size; i++){
    img.Y[i]=ajuste(divY,Ymin,img.Y[i]);
}

for(i=0; i<img.size; i++){
    img.R[i]=(( 1*img.Y[i])-( 0*img.U[i])+( 1.135983*img.V[i]));
    img.G[i]=(( 1*img.Y[i])-( 0.39465*img.U[i])-( 0.58060*img.V[i]));
    img.B[i]=(( 1*img.Y[i])+( 2.03211*img.U[i])+( 0*img.V[i]));
}

```

```

}

Rmax=maximos(img.R,img.size);
Gmax=maximos(img.G,img.size);
Bmax=maximos(img.B,img.size);

Rmin=minimos(img.R,img.size);
Gmin=minimos(img.G,img.size);
Bmin=minimos(img.B,img.size);

divR=Rmax-Rmin;
divG=Gmax-Gmin;
divB=Bmax-Bmin;

for(i=0; i<img.size; i++){
    img.R[i]=ajuste(divR,Rmin,img.R[i]);
    img.G[i]=ajuste(divG,Gmin,img.G[i]);
    img.B[i]=ajuste(divB,Bmin,img.B[i]);
}

for(i=0; i<img.size; i++){
    if(img.R[i]>255)
        img.R[i]=gmax;
    if(img.G[i]>255)
        img.G[i]=gmax;
    if(img.B[i]>255)
        img.B[i]=gmax;
}

//guarda las bandas como imagen TIFF
WriteTIFF("retinex_y.tif", &img);
free(OP);    OP = NULL;
free(bufferY); bufferY = NULL;
FreeImgColor(&img);
}

```

```

float ajuste(float div, int min, float i)
{
    float pixel;

    if (div==0.0)
        pixel=min;
    else
        pixel=(float)((i-min)/div)*255;

    return pixel;
}

```

```

int maximos(float vect[], int n)
{
    int i, max;

    max=vect[0];

    for(i=0; i<n; i++){
        if(vect[i]>max)
            max=vect[i];
    }
    return max;
}

```

```

int minimos(float vect[], int n)
{
    int i, min;

    min=vect[0];

    for(i=0; i<n; i++){
        if(vect[i]<min)
            min=vect[i];
    }
    return min;
}

```

```
}
```

```
void CompareWith(int s_row, int s_col, float *OP, float *bufferRGB, float Maximum, int size, int w, int h){
```

```
    register int i, j, k, l, m, n;
```

```
    int pos, pos2, w2, finh, finw, Tsize;
```

```
    float *IP;
```

```
    Tsize = size * 3;
```

```
    w2 = w * 3;
```

```
    DinamicAsign_float(&IP, Tsize, "IP");
```

```
    for(i=0; i < size * 3; i++) *(IP + i) = *(OP + i);
```

```
    if(s_row + s_col > 0){
```

```
        for(i=s_row, k=0; i<h; i++, k++)
```

```
            for(j=s_col, l=0; j<w2; j++, l++){
```

```
                pos = i * w2 + j;
```

```
                pos2 = k * w2 + l;
```

```
                *(IP + pos) = *(OP + pos2) + *(bufferRGB + pos) - *(bufferRGB + pos2);
```

```
            }
```

```
    }else{
```

```
        finh = h - 1 + s_row;
```

```
        finw = w2 - 1 + s_col;
```

```
        for(i=0, k=-s_row; i<finh; i++, k++)
```

```
            for(j=0, l=-s_col; j<finw; j++, l++){
```

```
                pos = i * w2 + j;
```

```
                pos2 = k * w2 + l;
```

```
                *(IP + pos) = *(OP + pos2) + *(bufferRGB + pos) - *(bufferRGB + pos2);
```

```
            }
```

```
    }
```

```
    for(i=0; i<Tsize; i++)
```

```
        if(*(IP + i) > Maximum)
```

```
            *(IP + i) = Maximum;
```

```
    for(i=0; i<Tsize; i++)
```

```

*(OP + i) = (*(IP + i) + *(OP + i)) / 2;

free(IP); IP = NULL;
}

void CompareWith2(int s_row, int s_col, float *OP, float *bufferY, float Maximum, int
size, int w, int h){
register int i, j, k, l, m, n;
int pos, pos2, w2, finh, finw, Tsize;
float *IP;

Tsize = size;
w2 = w;

DinamicAsign_float(&IP, Tsize, "IP");
for(i=0; i < size; i++) *(IP + i) = *(OP + i);

if(s_row + s_col > 0){
for(i=s_row, k=0; i<h; i++, k++){
for(j=s_col, l=0; j<w2; j++, l++){
pos = i * w2 + j;
pos2 = k * w2 + l;
*(IP + pos) = *(OP + pos2) + *(bufferY + pos) - *(bufferY + pos2);
}
}
}else{
finh = h - 1 + s_row;
finw = w2 - 1 + s_col;
for(i=0, k=-s_row; i<finh; i++, k++){
for(j=0, l=-s_col; j<finw; j++, l++){
pos = i * w2 + j;
pos2 = k * w2 + l;
*(IP + pos) = *(OP + pos2) + *(bufferY + pos) - *(bufferY + pos2);
}
}
}
for(i=0; i<Tsize; i++)
if(*(IP + i) > Maximum)

```

```

*(IP + i) = Maximum;

for(i=0; i<Tsize; i++)
*(OP + i) = (*(IP +i) + *(OP + i)) / 2;

free(IP); IP = NULL;
}

```

```

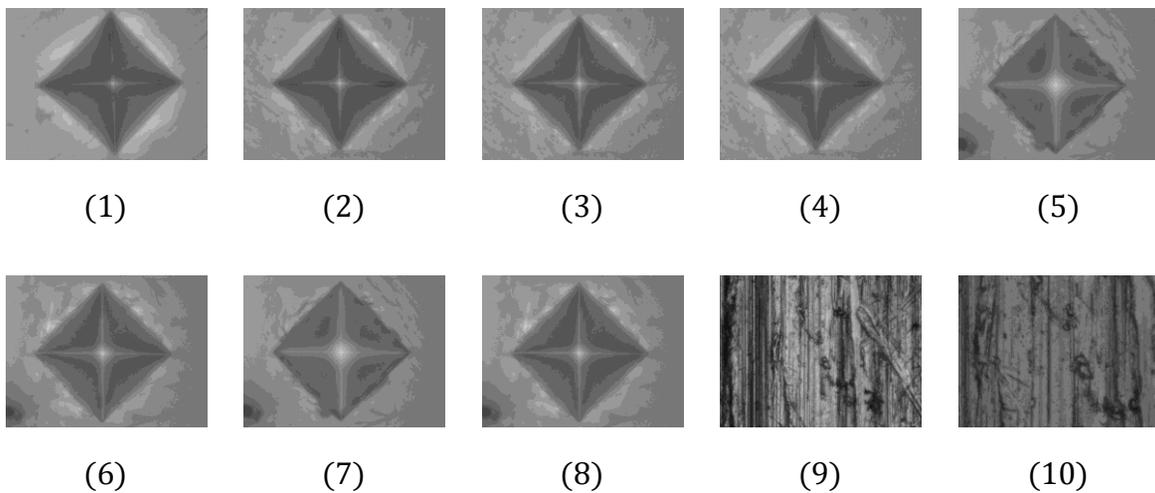
inline float Max(float *vector, int vsize){
register int i;
float max=-1024;

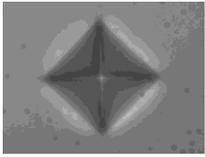
for(i=0; i<vsize; i++)
if(*(vector + i) > max)
max = *(vector + i);

return max;
}

```

7.3. Base de datos (imágenes con buen contraste)





(11)



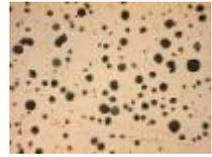
(12)



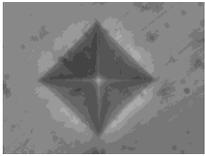
(13)



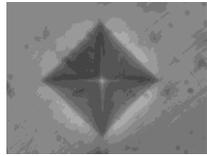
(14)



(15)



(16)



(17)



(18)



(19)



(20)



(21)



(22)



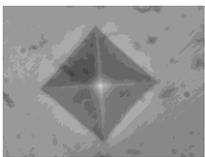
(23)



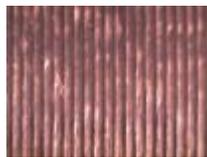
(24)



(25)



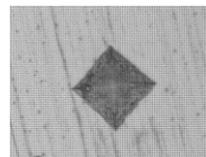
(26)



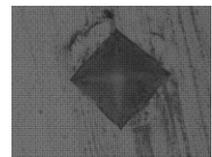
(27)



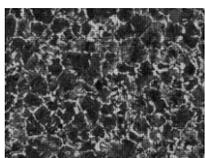
(28)



(29)



(30)



(31)



(32)



(33)



(34)



(35)



(36)



(37)



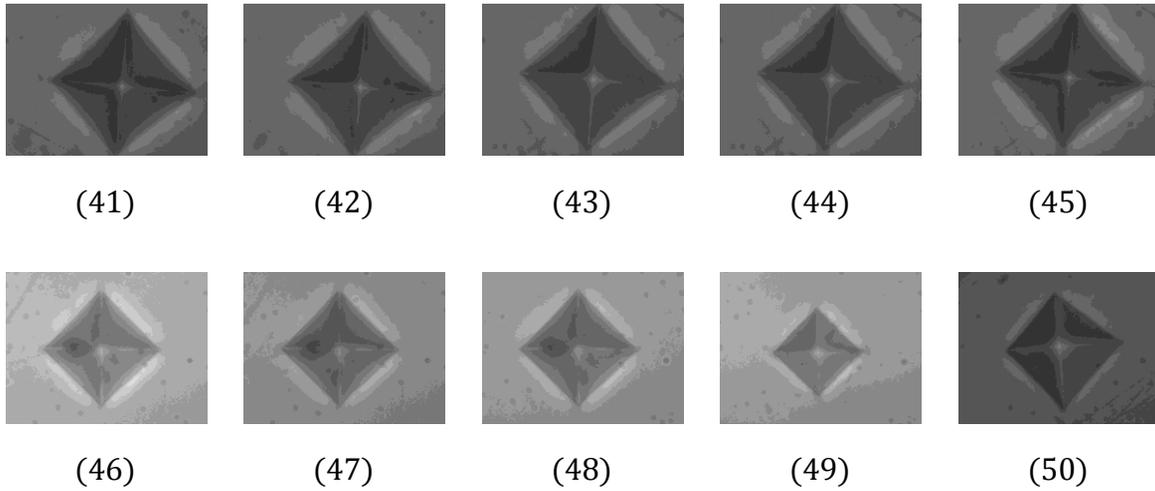
(38)



(39)



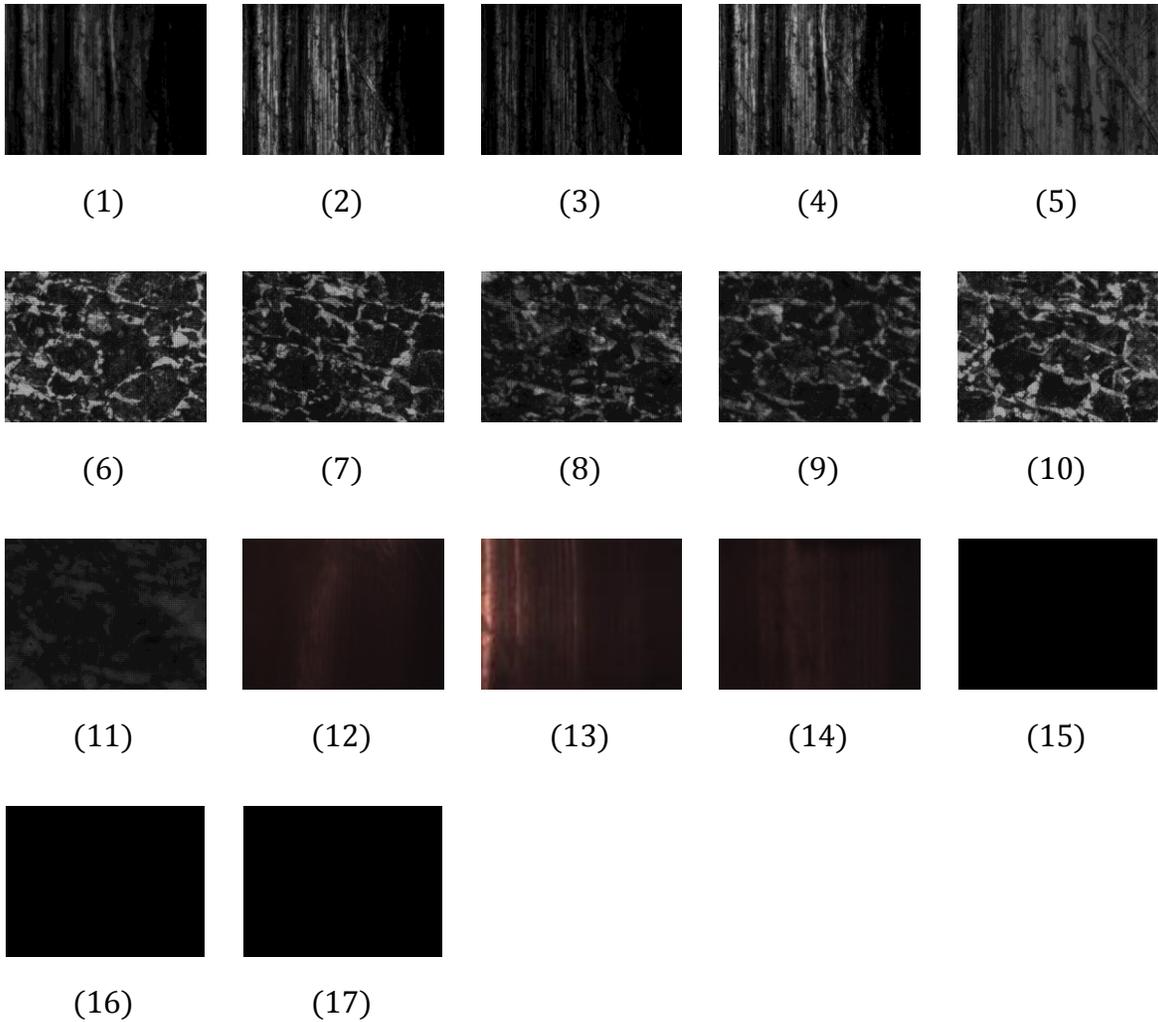
(40)



No. de muestra	Media	No. de muestra	Media
1	133.3	26	141.2
2	127.4	27	152.9
3	128.6	28	170
4	129	29	167.7
5	122.5	30	94.1
6	125.1	31	76.9
7	123.4	32	144.6
8	124.9	33	142.3
9	102.5	34	89.6
10	92.6	35	89.6
11	122.3	36	89.1
12	126.8	37	147.8
13	97.7	38	151.3
14	127.4	39	118.6
15	204.5	40	91.5
16	124.8	41	91.5
17	125.3	42	94.6
18	125.2	43	92.7
19	125.2	44	92.7
20	124.3	45	94
21	125.5	46	163.5
22	108	47	131.4
23	140.3	48	140.4
24	158.2	49	147.8
25	124.7	50	82.9

Tabla 7.1 Información estadística de la base de datos de imágenes con buen contraste.

7.4. Base datos (imágenes con mal contraste)



No. de muestra	Media	No. de muestra	Media
1	27	10	58.1
2	39	11	29.7
3	19.1	12	31.5
4	42.5	13	47
5	50.2	14	30
6	57.9	15	9.1
7	45.1	16	8.9
8	40	17	4.7
9	39.1		

Tabla 7.2 Información estadística de la base de datos de imágenes con mal contraste.

7.5. Retinex en Matlab

Matlab Implementation of Frankle-McCann retinex

```
function Retinex = retinex_frankle_mccann(L, nIterations)
global RR IP OP NP Maximum
RR = L;
Maximum = max(L(:)); % maximum color value in the image
[nrows, ncols] = size(L);

shift = 2^(fix(log2(min(nrows, ncols)))-1); % initial shift
OP = Maximum*ones(nrows, ncols) % initialize Old Product

while (abs(shift) >= 1)
    for i = 1:nIterations
        CompareWith(0, shift); % horizontal step
        CompareWith(shift, 0); % vertical step
    end
    shift = -shift/2; % update the shift
end
Retinex = NP;

function CompareWith(s_row, s_col)
global RR IP OP NP Maximum
IP = OP;
if (s_row + s_col > 0)
    IP((s_row+1):end, (s_col+1):end) = OP(1:(end-s_row), 1:(end-s_col)) + ...
    RR((s_row+1):end, (s_col+1):end) - RR(1:(end-s_row), 1:(end-s_col));
else
    IP(1:(end+s_row), 1:(end+s_col)) = OP((1-s_row):end, (1-s_col):end) + ...
    RR(1:(end+s_row), 1:(end+s_col)) - RR((1-s_row):end, (1-s_col):end);
end
IP(IP > Maximum) = Maximum; % The Reset operation
NP = (IP + OP)/2; % average with the previous Old Product
OP = NP; % get ready for the next comparison
```

Figura 7.1 Retinex en Matlab de Frankle-McCann.