



Universidad Central de Venezuela  
Facultad de Ciencias  
Escuela de Computación  
Centro de Computación Gráfica

## **Calidad perceptual en imágenes en escala de grises empleando Decoloración Espectral**

Trabajo Especial de Grado  
presentado ante la ilustre  
Universidad Central de Venezuela  
para optar al título de  
Licenciada en Computación

**Autor:** Irena Cabanach Dresden

**Tutor:** Esmitt Ramírez

Caracas, 25 de septiembre de 2019

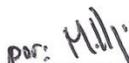
Universidad Central de Venezuela  
Facultad de Ciencias  
Escuela de Computación  
Centro de Computación Gráfica

### ACTA DEL VEREDICTO

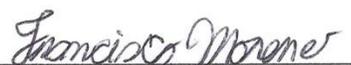
Quienes suscriben, Miembros del Jurado designado por el Consejo de la Escuela de Computación para examinar el Trabajo Especial de Grado presentado por la bachiller Irena Cabanach Dresden, C.I.: V-18.809.387, con el título *Calidad perceptual en imágenes en escala de grises empleando Decoloración Espectral*, a los fines de cumplir con el requisito legal para optar al título de Licenciada en Computación, dejan constancia de lo siguiente:

Leído el trabajo por cada uno de los Miembros del Jurado, se fijó el día 14 de octubre de 2019, a las 12:00 p.m., para que su autor lo defendiera en forma pública, en el *Centro de Computación Gráfica*, lo cual se realizó mediante una exposición oral de su contenido, y luego respondió satisfactoriamente a las preguntas que le fueron formuladas por el Jurado, todo ello conforme a lo dispuesto en la Ley de Universidades y demás normativas vigentes de la Universidad Central de Venezuela. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobarlo.

En fe de lo cual se levanta la presente acta, en Caracas a los catorce días del mes de octubre del año dos mil diecinueve, dejándose también constancia de que actuó como Coordinador del Jurado el Profesor Esmitt Ramírez, que estuvo presente por videoconferencia. Firma en su lugar el Director de la Escuela de Computación Prof. Robinson Rivas.

por:   
Prof. Esmitt Ramírez



  
Prof. Francisco Moreno

  
Prof. Miguel Astor

*A Oma.*

# AGRADECIMIENTOS

Desde antes del inicio de esta tesis y durante todo su desarrollo, no solo debo de agradecerle a la Universidad Central de Venezuela, a la Escuela de Computación, por permitirme el acceso a estas puertas del conocimiento y brindarme experiencias para la vida, sino también a todas las personas que hicieron este camino posible. El agradecimiento que tampoco puede faltar es al universo, a la vida, por permitirme existir, y colocar estas oportunidades a mi alcance, y al alcance de muchos.

En mi lista de agradecimientos no puedo dejar de indicar a varias personas, cuyo apoyo ha sido determinante en este trabajo, personalmente importante:

Al profesor, Esmitt Ramírez, por recibirme como tutorada, por ayudarme a encontrar un tema de trabajo especial de grado que me motivara, por guiarme y apoyarme, por su comprensión y paciencia, a pesar de mi largo plazo en este proceso.

A todos los profesores que durante mi formación, se dedicaron a transmitir conocimientos de calidad, con entrega incondicional, y que veo como inspiradores de la ciencia, en pro de los valores humanos.

Al profesor, Carlos Ayesta, por presentarme a la fotografía con tanta pasión, por compartir el conocimiento de la luz, de la comunicación, de la imagen... Por recibirme en el laboratorio de fotografía.

A mi amigo, Pavel Bastidas, por trasladarme incansablemente a través del arte y la fotografía, por tantas horas de conversaciones y enseñanzas, que me inspiraron a no ponerle límites a la imaginación.

A la profesora, Alejandra Camacho, por motivarme a seguir adelante a pesar de los tropiezos que aparezcan en el camino y a verle el lado positivo a los acontecimientos.

A mis amigos, Víctor Felipe, por su amistad y apoyo en varias etapas de mi carrera y vida personal; a Luis Carlos, por su amistad incondicional y de gran apoyo en momentos difíciles; a Luis Miguel, por sus conversaciones distractoras y su gran amistad; a Víctor Niebla, por su amistad, apoyo y disposición ante cualquier circunstancia; a Istvan, por su amistad a lo largo de toda la carrera, por su compromiso al estudio, su ayuda y su entrega a explicarme con paciencia.

A todas las personas que me apoyaron en distintos momentos de la carrera y en este proceso de investigación, como Tony, Audel, Anabel, Mercedes y Mayela.

A mis padres, Elisabeth y David, por apoyarme incondicionalmente y creer en que culminaría mi formación de pre-grado satisfactoriamente.

A mis suegros, Eleonora y Luis Alfredo, por su apoyo y palabras motivacionales.

A mi esposo, Luis Federico, por su gran amor, apoyo incondicional y constancia, por estar junto a mí en todos los momentos difíciles, cuando pensé que no podría más, y por su creencia en mis proyectos y sueños.

Y de última, pero no menos importante, a mi abuela, Oma, quien compartió palabras de aliento, historias motivacionales y me brindó apoyo pleno en todo lo referente con mis estudios, quien nunca dejó de creer en que podría lograr cualquier cosa que me propusiera, y que sigue apoyándome con su recuerdo.

# RESUMEN

La conversión de una imagen digital de color a escala de grises conlleva a una pérdida inevitable de información ya que se transforman los datos de un espacio de tres dimensiones a una dimensión. Una de las técnicas para resolver esta pérdida consiste en considerar no solamente el canal de luminancia sino también los canales cromáticos para la transformación.

El algoritmo de decoloración propone emplear la transformada de Fourier para calcular las magnitudes de los cambios entre los canales de color utilizados, en todas las escalas espaciales, y realizar modificaciones con operaciones aritméticas sencillas, como agregar contrastes cromáticos a la intensidad de grises.

En este trabajo se presenta una implementación del algoritmo de decoloración basándose en el estudio mencionado, modificando el cálculo automático de los coeficientes de control que actúan sobre el resultado en escala de grises. Además, se cuenta con una interfaz gráfica que permite al usuario manipular los valores de los coeficientes y mostrar su efecto en tiempo real sobre la imagen. Los resultados de calidad perceptual de este algoritmo son obtenidos a través de evaluaciones perceptuales, mediante encuestas, utilizando el paradigma 2AFC, y mediante el índice C2G-SSIM, utilizando veinte imágenes a color y diez conversiones a escala de grises para ambas; resultantes de aplicar los distintos valores de coeficientes en el algoritmo (manuales, automáticos y la combinación de ambos) y la luminosidad del canal L del espacio CIELab.

**Palabras claves:** Procesamiento digital de imágenes, PDI, color a escala de grises, C2G, decoloración del espectro, Transformada de Fourier, FFT, DFFT, luminancia, CIELab, automatización, evaluación subjetiva, evaluación objetiva, 2AFC, Thurstone, C2G-SSIM.

# TABLA DE CONTENIDO

ÍNDICE DE FIGURAS.....	9
ÍNDICE DE TABLAS.....	12
INTRODUCCIÓN.....	13
<b>1 TRANSFORMACIÓN DE IMÁGENES DE COLOR A ESCALA DE GRISES.....</b>	<b>15</b>
<b>1.1 Conceptos básicos .....</b>	<b>15</b>
1.1.1 El color.....	15
1.1.2 Términos que especifican el color .....	17
1.1.2.1 Intensidad .....	17
1.1.2.2 Tono .....	17
1.1.2.3 Saturación.....	17
1.1.2.4 Brillo.....	18
1.1.2.5 Luminancia.....	18
1.1.2.6 Crominancia .....	18
1.1.2.7 Luminosidad.....	19
<b>1.2 Modelos de color .....</b>	<b>19</b>
1.2.1 RGB .....	20
1.2.2 CIELab .....	22
<b>1.3 Algoritmos de transformación de color a escala de grises .....</b>	<b>24</b>
1.3.1 Luminosidad HSL ( <i>Lightness HSL</i> ).....	24
1.3.2 Promedio.....	24
1.3.3 Luminancia ( <i>Luminance</i> ).....	25
1.3.4 Luminosidad ( <i>Lightness</i> ) .....	25
1.3.5 Decoloración espectral.....	25
1.3.5.1 Optimización del contraste.....	26
1.3.5.2 Control de parámetros .....	26
<b>2 EVALUACIÓN DE CALIDAD DE LAS IMÁGENES .....</b>	<b>29</b>
<b>2.1 Métrica subjetiva de comparación por pares.....</b>	<b>29</b>
2.1.1 Elección forzada de dos alternativas (2AFC) .....	31
2.1.2 Recuento de votos vs. escalamiento.....	31
2.1.3 Ley del juicio comparativo de Thurstone .....	32
2.1.4 Caso V de Thurstone.....	36
2.1.5 Método de mínimos cuadrados de Thurstone-Mosteller .....	37
<b>2.2 Métrica objetiva C2G-SSIM.....</b>	<b>38</b>
2.2.1 Índice C2G-SSIM .....	38
2.2.1.1 Transformación del espacio de color .....	39
2.2.1.2 Medida de similitud.....	39
2.2.1.3 Medida de calidad general.....	40
<b>2.3 Coeficientes de correlación .....</b>	<b>42</b>

2.3.1 SRCC .....	42
2.3.2 KRCC.....	43
<b>3 PLANTEAMIENTO DEL PROBLEMA .....</b>	<b>44</b>
<b>3.1 Planteamiento del problema .....</b>	<b>44</b>
<b>3.2 Justificación.....</b>	<b>44</b>
<b>3.3 Objetivos.....</b>	<b>45</b>
3.3.1 Objetivo general.....	45
3.3.2 Objetivos específicos .....	45
<b>3.4 Metodología.....</b>	<b>45</b>
3.4.1 Diseño e implementación.....	46
3.4.2 Selección de estrategia de evaluación de calidad .....	46
3.4.3 Generación de pruebas.....	47
3.4.4 Elaboración de las encuestas.....	47
3.4.5 Cálculo de las métricas seleccionadas .....	47
<b>3.5 Esquema del proyecto .....</b>	<b>47</b>
<b>3.6 Ambiente de implementación y diseño de pruebas.....</b>	<b>49</b>
3.6.1 Hardware.....	49
3.6.2 Software .....	49
<b>4 DISEÑO DE LA SOLUCIÓN .....</b>	<b>50</b>
<b>4.1 Algoritmo de transformación .....</b>	<b>50</b>
4.1.1 Control de parámetros $\theta$ y $\phi$ .....	50
<b>4.2 Diseño de la GUI.....</b>	<b>51</b>
4.2.1 Abrir imagen .....	52
4.2.2 Despliegue de la imagen .....	52
4.2.3 Valores manuales .....	52
4.2.4 Valores automáticos.....	52
4.2.5 Restablecer valores .....	52
4.2.6 Mostrar imagen original.....	53
4.2.7 Resolución.....	53
4.2.8 Guardar imagen y salir.....	53
<b>5 IMPLEMENTACIÓN.....</b>	<b>54</b>
<b>5.1 Algoritmo de transformación .....</b>	<b>55</b>
5.1.1 Programas para el desarrollo.....	56
5.1.2 Biblioteca .....	56
5.1.2.1 Imagen a color .....	56
5.1.2.2 Obtención de canales.....	56
5.1.2.3 Transformada de Fourier (FFT) en cada canal.....	57
5.1.2.4 Magnitud de los valores complejos .....	59
5.1.2.5 Cálculo de $\theta$ y $\phi$ .....	59
5.1.2.6 Cálculo de la ecuación $\ddot{E}$ .....	61
5.1.2.7 Transformada inversa de Fourier .....	62
5.1.2.8 Transformar resultado al espacio CIELab.....	62

5.1.2.9 Transformar resultado al espacio BGR .....	63
<b>5.2 GUI.....</b>	<b>63</b>
5.2.1 Herramientas para el desarrollo .....	65
5.2.2 Bibliotecas.....	65
5.2.2.1 Signals y slots.....	65
5.2.2.2 Diálogo de selección de imagen.....	65
5.2.2.3 Despliegue de la imagen .....	67
5.2.2.4 Valores manuales .....	67
5.2.2.5 Valores automáticos .....	69
5.2.2.6 Restablecer valores.....	70
5.2.2.7 Mostrar imagen original .....	71
5.2.2.8 Resolución .....	71
5.2.2.9 Guardar imagen y salir .....	72
<b>6 PRUEBAS Y RESULTADOS .....</b>	<b>73</b>
<b>6.1 Imágenes a evaluar .....</b>	<b>74</b>
6.1.1 Tipos de coeficientes.....	75
<b>6.2 Aplicación de encuestas.....</b>	<b>76</b>
6.2.1 Resultados de las encuestas .....	77
<b>6.3 Métrica C2G-SSIM.....</b>	<b>79</b>
6.3.1 Resultados de la métrica C2G-SSIM .....	80
<b>6.4 Coeficientes de correlación .....</b>	<b>81</b>
<b>7 CONCLUSIONES Y TRABAJOS A FUTURO .....</b>	<b>84</b>
<b>REFERENCIAS .....</b>	<b>85</b>
<b>ANEXOS .....</b>	<b>87</b>

# ÍNDICE DE FIGURAS

Figura 1: Luz, objeto y observador. [2].....	15
Figura 2: Espectro electromagnético remarcando el espectro visible al ojo humano. [3] .....	16
Figura 3: Dispersión de la luz a través de un prisma. [3].....	16
Figura 4: Taxonomía de una selección de modelos de color. [4].....	20
Figura 5: (a) Cubo RGB. (b) Parte externa del cubo RGB. [6].....	21
Figura 6: Representación cartesiana del espacio CIEL*a*b*. [4] [12].....	23
Figura 7: Correspondencia en escala de grises de una imagen de paisaje por satélite de los incendios.....	27
Figura 8: Comparación cualitativa de los algoritmos de transformación de color a escala de grises. (a) Imágenes a color, (b) Promedio ingenuo, (c) Luminosidad HSL, (d) Luminancia, (e) Luminosidad y (f) Decoloración espectral. [14].....	28
Figura 9: Escalado de datos de comparación por pares para evaluar la calidad percibida de la imagen. [17].....	30
Figura 10: Representación del continuo psicológico y la posición (desconocida) de cuatro objetos ficticios (O1 a O4), según el grado del atributo de interés. [21].....	33
Figura 11: Funciones de densidad de probabilidad de A y B, la calidad aleatoria de dos opciones. El eje x representa la escala de calidad, donde cada opción se coloca en la escala de calidad en su media. [22].....	34
Figura 12: La diferencia de calidad aleatoria $A - B$ es una gaussiana con media $\mu A - \mu B$ .....	34
Figura 13: Gaussiana FDA. [22] .....	36
Figura 14: Estructura de tres etapas de C2G-SSIM. [25].....	39
Figura 15: Imágenes C2G y sus mapas de índice C2G-SSIM. (a) es la imagen a color de referencia. (b) y (c) son las imágenes C2G creadas por distintos métodos de decoloración. (d) y (e) son los correspondientes mapas C2G-SSIM de (b) y (c), respectivamente. En todos los mapas de índice C2G-SSIM, más brillante indica mejor calidad. [25] .....	41
Figura 16: Demostración visual de C2G-SSIM. CIEY, Bala [26], Rasche [27], Gooth [28], Kim [29] y Gland [30], son los algoritmos de decoloración aplicados a la imagen original y Q es el índice C2G-SSIM. [25] .....	41
Figura 17: Metodología de desarrollo. ....	46
Figura 18: Esquema general de la interacción de la aplicación con el usuario. ....	48
Figura 19: Esquema general de la evaluación de calidad.....	48
Figura 20: GUI propuesta, con el despliegue de la imagen a color seleccionada a transformar....	51
Figura 21: Diagrama de flujo de interacción entre el usuario y el algoritmo de transformación propuesto. ....	54

Figura 22: Diagrama de flujo del algoritmo de transformación de una imagen de color a escala de grises implementado.....	55
Figura 23: Captura del algoritmo de lectura de la imagen seleccionada.....	56
Figura 24: Captura del algoritmo de la declaración de la matriz imagen. ....	56
Figura 25: Captura del algoritmo para la obtención de los canales B, G y R. ....	57
Figura 26: Captura del algoritmo para obtención de los canales L, a y b. ....	57
Figura 27: Captura del algoritmo para el cálculo de la FFT en los canales B, G y R. ....	58
Figura 28: Captura del algoritmo para el cálculo de la FFT en los canales L, a y b. ....	58
Figura 29: Captura del algoritmo del cálculo de las magnitudes de las matrices complejas arrojadas por la FFT en los canales B, G y R. ....	59
Figura 30: Captura del algoritmo del cálculo de $\theta$ individual.....	60
Figura 31: Captura del algoritmo del cálculo de $\theta$ promedio. ....	60
Figura 32: Captura del algoritmo del cálculo de $\phi$ individual.....	60
Figura 33: Captura del algoritmo del cálculo de $\phi$ promedio.....	60
Figura 34: Captura del algoritmo del cálculo de la ecuación $\ddot{E}$ .....	61
Figura 35: Captura del algoritmo para la obtención de la inversa de la transformada de Fourier. ....	62
Figura 36: Captura del algoritmo de transformación del resultado obtenido al espacio CIELab. ....	62
Figura 37: Captura del algoritmo de transformación del espacio CIELab al espacio BGR.....	63
Figura 38: Diagrama de secuencia de la interacción entre el usuario, la GUI y el algoritmo de transformación.....	64
Figura 39: Capture del algoritmo de conexión con <i>signal</i> y <i>slot</i> . ....	65
Figura 40: Capture del algoritmo que contiene la invocación al elemento del diálogo de selección de la imagen a transformar. ....	66
Figura 41: Captura del cuadro de diálogo para la selección de la imagen a transformar.....	66
Figura 42: Captura del algoritmo de transformación de Qstring a string.....	66
Figura 43: Captura del algoritmo de despliegue de la imagen transformada en el QLabel nombrado "label_pic".....	67
Figura 44: Captura e indicación de widgets de la sección de control de valores de coeficientes de forma manual de la interfaz de usuario implementada.....	68
Figura 45: Captura de la implementación de conexión entre la edición de línea y el slider del coeficiente $\theta$ . ....	68
Figura 46: Captura de parte del algoritmo que modifica la edición de línea del coeficiente $\theta$ .....	68
Figura 47: Captura de parte del algoritmo que modifica la posición de la barra deslizante del coeficiente $\theta$ . ....	69
Figura 48: Captura e indicación de widgets de la sección de control de valores de coeficientes de forma automática de la GUI implementada.....	69
Figura 49: Captura de la implementación de conexión de los botones del cálculo automático (promedio e individual) de los coeficientes $\theta$ y $\phi$ . ....	70

Figura 50: Captura del botón que restablece valores de los coeficientes de la interfaz de usuario implementada. ....	70
Figura 51: Captura de la implementación de conexión del botón de restablecimiento de valores y su función respectiva. ....	70
Figura 52: Captura de parte del algoritmo que se ejecuta al seleccionar la opción de restablecimiento de valores. ....	70
Figura 53: Captura del botón de la GUI implementada que ejecuta el despliegue de la imagen original a color por pantalla.....	71
Figura 54: Captura de la conexión implementada para el botón "Mostrar imagen original".....	71
Figura 55: Captura de la GUI implementada para permitir la modificación de la resolución de la imagen. ....	71
Figura 56: Captura de la implementación de conexión entre la edición de línea y el slider de cambio de resolución.....	72
Figura 57: Captura de la GUI implementada de los botones "Guardar" y "Salir", e identificación del widget de casilla de verificación. ....	72
Figura 58: Captura de la implementación de conexión del botón “Guardar imagen” y del botón “Salir”.....	72
Figura 59: Comparación entre los resultados de los diferentes métodos de transformación de imágenes de color a escala de grises. [19].....	73
Figura 60: Imágenes de prueba para el algoritmo implementado. ....	74
Figura 61: Ejemplo de una evaluación de comparación por pares mostrada al sujeto de prueba. .	77
Figura 62: Gráfica de posiciones escalares que ocupan los resultados de las comparaciones realizadas del algoritmo propuesto y sus distintos coeficientes para la Imagen 1. El mejor resultado ocupa el lado derecho. ....	79
Figura 63: Gráfica de posiciones escalares generales que ocupan los resultados de las comparaciones realizadas del algoritmo propuesto y sus distintos parámetros para todas las imágenes. ....	79
Figura 64: Coeficientes de correlación SRCC y KRCC para todas las imágenes evaluadas.....	83

# ÍNDICE DE TABLAS

Tabla 1: Clasificación de cuatro objetos en dos descriptores, $x$ y $y$ . [31] .....	42
Tabla 2: Características de la imágenes a color utilizadas para la evaluación del algoritmo propuesto. ....	75
Tabla 3: Tabla que identifica los coeficientes a evaluar por imagen. ....	76
Tabla 4: Matriz de frecuencias obtenida de las encuestas de la Imagen 1. ....	77
Tabla 5: Matriz de proporciones referente a la Imagen 1. ....	78
Tabla 6: Matriz de puntuaciones típicas $z$ referente a la Imagen 1. ....	78
Tabla 7: Promedios de las puntuaciones típicas $z$ , transformados linealmente y ordenados de forma creciente de la Imagen 1. ....	78
Tabla 8: Índices C2G-SSIM de la Imagen 1. ....	80
Tabla 9: Índices C2G-SSIM promedios. ....	80
Tabla 10: Posiciones de clasificación obtenidas para la Imagen 1 tanto para los resultados de las puntuaciones típicas $z$ promediadas (“THURSTONE”) como de los índices C2G-SSIM. ....	81
Tabla 11: Resultados de los coeficientes de correlación entre las evaluaciones subjetivas y objetivas, junto con las transformaciones en escala de grises por imagen a color ordenadas según las respectivas clasificaciones obtenidas. ....	82

# INTRODUCCIÓN

Hoy en día, el uso de las imágenes se ha convertido en un elemento de constante presencia e importancia en nuestra sociedad, con diferentes fines, que pueden ser muy variados; dentro de ellos podemos mencionar objetivos recreativos, comerciales, académicos, artísticos, científicos o médicos. Su alcance a través de los medios informáticos y su manipulación digital se ha vuelto una tendencia común, dadas las facilidades de acceso, manipulación y bajos costos, por lo que su estudio y desarrollo están en constante crecimiento.

Se puede decir que la mayoría de las imágenes adquiridas por cámaras, escáneres y computadoras son a color, mientras que las imágenes en escala de grises son obtenidas a través de la fotografía profesional o por una posterior manipulación digital de una imagen a color. Las imágenes en escala de grises son muy utilizadas en sistemas médicos, periódicos, revistas, fotocopias, impresiones, reconocimiento de patrones, entre otros.

La transformación de una imagen de color a escala de grises conlleva a una pérdida inevitable de información, ya que descarta la información cromática mientras que mantiene la luminancia; llevando una asignación de un conjunto tridimensional (3D) a un espacio unidimensional (1D); en donde dos colores perceptivamente contrastados pueden convertirse en un valor de luminancia similar, desapareciendo la diferencia cromática en los resultados. Una técnica de conversión efectiva mantiene la mayor cantidad de información original del color de la imagen, buscando así el “mejor resultado” o perceptualmente acertada, emulando tanto las impresiones locales como globales de la imagen, en donde los contrastes locales no son ni exagerados ni subestimados, los valores de grises son ordenados según la apariencia del color y las diferencias en los detalles espaciales son imperceptibles.

Los algoritmos más conocidos o triviales para la transformación de imágenes de color a escala de grises no arrojan resultados óptimos y no permiten la manipulación de los resultados en escala de grises, por lo que en este trabajo se estudia un algoritmo que utiliza la transformada de Fourier y que permite controlar, a través de una interfaz gráfica con visualización de los cambios en tiempo real, dos parámetros, el contraste cromático y el de luminancia que se aplica al resultado final de la imagen.

El contenido de este documento se organiza de la siguiente manera:

**Capítulo 1:** Debido a que las imágenes de entrada para la transformación a escala de grises son a color, se realiza una breve presentación de los conceptos básicos del color y a las estructuras que comúnmente se manejan en esta área. Además, se explican los algoritmos de transformación que se consideran importantes para el desarrollo de este trabajo.

**Capítulo 2:** Se explica en qué consisten las evaluaciones subjetivas y objetivas, necesarias para la comparación y análisis de los resultados arrojados por el algoritmo implementado.

**Capítulo 3:** Se presentan de forma puntual el planteamiento del problema y los objetivos a cumplir en el presente trabajo.

**Capítulo 4:** Se expone el diseño de la solución planteada explicando el flujo del funcionamiento tanto del algoritmo de transformación como de la interfaz gráfica, utilizando sus respectivos diagramas.

**Capítulo 5:** Se detallan las características del software y hardware utilizado, los lenguajes de programación y bibliotecas; así como, los programas utilizados para la elaboración y ejecución de las pruebas.

**Capítulo 6:** Se muestran los resultados obtenidos tras haber aplicado las respectivas evaluaciones con sus respectivos análisis.

**Capítulo 7:** Conclusiones, recomendaciones y posibles trabajos futuros.

# 1 TRANSFORMACIÓN DE IMÁGENES DE COLOR A ESCALA DE GRISES

Antes de comenzar a trabajar en los algoritmos de transformación de imágenes de color a escala de grises no triviales se considera de importancia tener claros algunos conceptos relacionados con el color y las estructuras utilizadas para su almacenamiento y procesamiento, así como conocer los algoritmos básicos o triviales que sirven para comprender el problema que se enfrenta al momento de desarrollar los algoritmos complejos.

## 1.1 Conceptos básicos

Se considera que conocer ciertos conceptos relacionados al color, permiten comprender el funcionamiento de los algoritmos que los transforman.

### 1.1.1 El color

Según Levkowitz [1] el color es una sensación producida en el cerebro en respuesta a la incidencia de la luz en la retina del ojo. La sensación de color es causada por las diferentes cualidades de la luz emitida por las fuentes de luz o reflejada por los objetos. El color resulta de una interacción entre la luz, el objeto y el espectador. Es la luz que ha sido modificada por un objeto de tal manera que el espectador, como el sistema visual humano, percibe la luz modificada como un color distinto (Figura 1). Los tres elementos deben estar presentes para que exista el color como lo conocemos.



Figura 1: Luz, objeto y observador. [2]

La luz es una parte del espectro de las ondas electromagnéticas, que incluye desde los rayos X y ultravioleta (UV) hasta los rayos infrarrojos (IR), microondas (MO), radio, etc. La franja de energía radiante comprendida entre los rayos ultravioleta y los rayos infrarrojos es aquella a la cual el ojo humano es sensible, y por eso se denomina espectro visible (del arco iris) o luz (Figura 2). El parámetro básico que diferencia estas radiaciones es la longitud de onda  $\lambda$ , que en el caso de la luz comprende desde los 380nm hasta los 780nm.

El patrón de visión en el ojo humano está proporcionado por la distribución discreta de receptores de luz sobre la superficie de la retina. Hay dos clases de receptores, los conos y los bastones. Los conos son

fuertemente sensibles al color (sensible a los rojos, verdes y azules); aproximadamente el 65% de los conos son sensibles a la luz roja, 33% son sensibles a la luz verde y el 2% son sensibles a la luz azul (son los conos más sensibles); y los bastones no están involucrados en la visión del color y son sensibles a bajos niveles de iluminación.

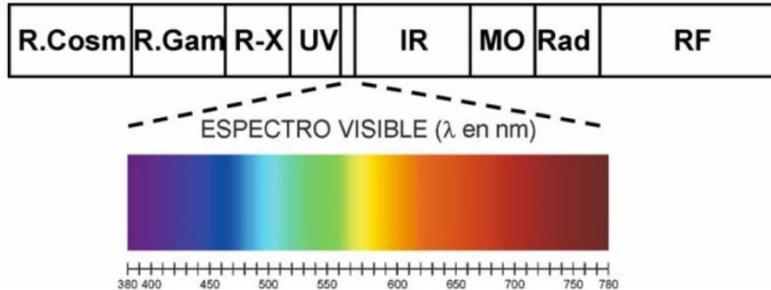


Figura 2: Espectro electromagnético remarcando el espectro visible al ojo humano. [3]

Exactamente qué color es percibido depende de la composición de longitudes de onda en las ondas de luz. Por ejemplo, si los sensores detectan todas las longitudes de onda visibles balanceadas a la vez, el cerebro percibe la luz blanca. Si se detecta un rango limitado del espectro visible se perciben algunos tonos de color. Si no se detectan longitudes de onda, no hay presencia de luz y el cerebro percibe el negro.

En 1666, Isaac Newton descubrió que cuando un haz de luz del sol pasa a través un prisma de cristal, el haz que emerge de la luz no es blanca, sino que consiste en un espectro continuo de colores que van del violeta al rojo. A través de este experimento podemos ver cómo nuestros ojos responden a cada longitud de onda individual y demuestra que las diferentes longitudes de onda nos hacen ver diferentes colores. Se pueden reconocer las regiones dominantes del espectro visible de color rojo, anaranjado, amarillo, verde, azul y violeta; y el "arco iris" de otros colores que se mezclan en el medio, como se puede observar en la Figura 3.

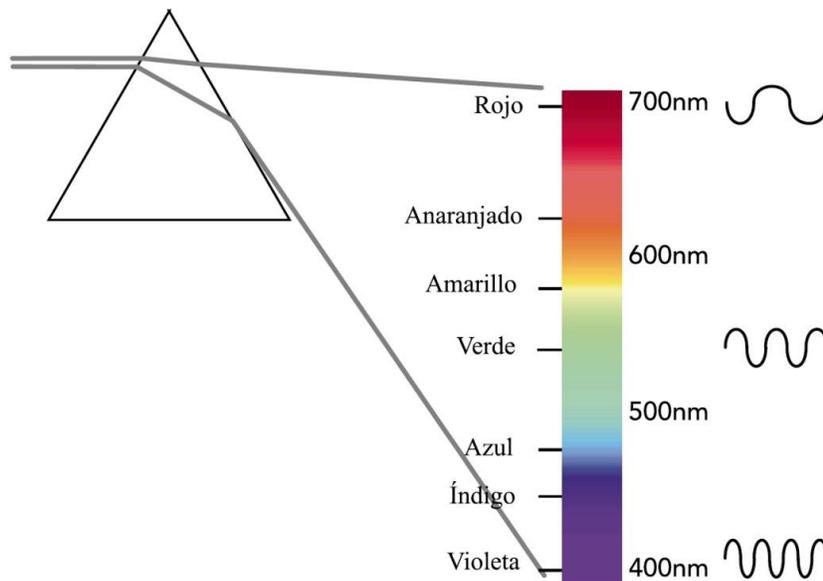


Figura 3: Dispersión de la luz a través de un prisma. [3]

### 1.1.2 Términos que especifican el color

Antes de proceder con los sistemas que especifican el color (modelos de color), es apropiado definir algunos términos. Bajo el punto de vista subjetivo o intuitivo, la CIE (*commission internationale de l'éclairage* - Comisión internacional de la iluminación) ha definido y recomendado las siguientes características para especificar el color percibido; estas características son los atributos, cualidades o propiedades del color.

Los colores tienen unas propiedades inherentes que les permiten distinguirse unos de otros, que les hacen variar de aspecto y que definen su apariencia final. También se les llaman: dimensiones, parámetros o variables del color. Estas propiedades del color son: intensidad (*intensity* - I), tono (matiz, tinte, *hue* - H), saturación (contenido de color, croma, *saturation* - S), brillo (valor, *brightness* - Br), luminancia (*luminance* - Y), crominancia, y luminosidad (claridad, *lightness* -  $L^*$ ).

Los humanos interpretan un color basándose en su luminancia, tono y saturación.

#### 1.1.2.1 Intensidad

La intensidad es una medida sobre un intervalo del espectro electromagnético, del flujo de poder que se irradia, o incide en una superficie; a menudo se llama una medida de luz lineal y por lo tanto se expresa en unidades, tales como vatios por metro cuadrado. El término intensidad se refiere a colores acromáticos.

#### 1.1.2.2 Tono

El tono es el estado puro del color, sin blanco o negro agregados, es un atributo del color asociado con la longitud de onda dominante en una mezcla de ondas de luz. Por lo tanto, la tonalidad representa el color dominante que es percibido por un observador; cuando se dice que un objeto es rojo, anaranjado o amarillo, el tono está siendo especificado. En otras palabras, es el atributo de una sensación visual según el cual un área parece ser similar a uno de los colores percibidos: rojo, amarillo, verde o azul, o una combinación de dos de ellos. Por ejemplo, todos los rojos tienen un valor de tono similar ya sea con luz, oscuro, intenso o pastel y coloca el color en su correcta posición en el espectro. Esta definición divide a los colores percibidos en dos clases: acromáticos (color percibido sin tono), cromáticos (color percibido con tono).

#### 1.1.2.3 Saturación

La saturación se refiere a la pureza relativa de la cantidad de luz blanca mezclada con la tonalidad, o la viveza del tono. Según Levkowitz [1], la saturación es el grado de diferencia de un gris de la misma luminosidad o brillo. La saturación es el colorido relativo a la luminosidad del color mientras que el tono es el colorido comparado con blanco. Cuando la luminosidad varía se percibe un cambio en la saturación percibida. Una mayor luminosidad causa la percepción de menor saturación, y viceversa. El tono no cambia con la luminosidad.

Los colores del espectro puro son completamente saturados y no contienen ninguna luz blanca. El color rosado (rojo y blanco) es menos saturado, con el grado de saturación inversamente proporcional a la cantidad de luz blanca añadida. Un color puede ser desaturado añadiendo luz blanca que contenga energía en todas las longitudes de onda; a medida que la saturación decrece, el color se torna más pálido hasta que finalmente se desvanece a neutral. Un gris neutral es considerado que tiene saturación cero. Cualquier cambio hecho a un color puro automáticamente baja su saturación.

La pureza especifica la distribución espectral que produce un color específico de luz. La pureza corresponde con la noción perceptual de saturación.

#### **1.1.2.4 Brillo**

El brillo o valor es definido como el atributo de una sensación visual conforme en que una zona parece emitir más o menos luz. El brillo de un color identifica la claridad u oscuridad del color. Es un descriptor subjetivo de la percepción de la luz. Cualquier color cuyo brillo sea cero es negro, sin importar su tono o saturación. Hay diferentes esquemas para especificar el brillo de un color, y dependiendo del que se utilice el resultado del aclarado de un color puede variar considerablemente.

El término brillo es utilizado para fuentes de luz, y está asociado con la luz emitida. Puede tomar términos como: muy tenue, tenue, intermedio, brillante, muy brillante. El término de valor fue primeramente utilizado en el sistema de color de Munsell [4]. Se refiere a la oscuridad o luminosidad relativa del color, en el sistema Munsell.

Los términos luminosidad y brillo tienden a intercambiarse en su uso, por lo que es recomendable detallar las especificaciones utilizadas en cada bibliografía. Algunas veces se hace referencia a colores brillantes, pero se recomienda hacer uso de términos vivos o saturados.

#### **1.1.2.5 Luminancia**

La luminancia especifica la cantidad de luz o reflexión que proviene de alguna región del espacio. Para una luz acromática es la intensidad de la luz. Para un color cromático corresponde a la noción subjetiva de luminosidad o brillo.

La luminancia no es una cantidad perceptual, es una medida física utilizada para definir la cantidad de luz en la región visible del espectro electromagnético. A diferencia de la luminosidad y brillo, la luminancia se puede leer directamente de un instrumento de medición científico. Es una medida de la energía luminosa ponderada por la función de sensibilidad espectral del sistema visual humano (medida en lúmenes); tomando en cuenta el hecho de que el ojo humano es más sensible a ciertos colores (como el amarillo-verde) que a otros (como el azul). Por ejemplo, la luz emitida por una fuente que opera en la región infrarroja del espectro puede tener energía significativa (radiancia), pero un observador difícilmente la percibiría, ya que su luminancia sería casi cero.

En general, la intensidad, luminosidad y brillo, especifican el componente acromático (luminancia). Se puede pensar como “qué cantidad de negro es mezclado en el color”.

#### **1.1.2.6 Crominancia**

La crominancia se define como el atributo de la sensación visual según el cual una superficie parece mostrar o contener más o menos color cromático. Es un concepto complementario de luminancia; es un espacio de color de dos dimensiones que representa el tono y la saturación, independientemente del brillo. Por ejemplo, la señal de televisión trabaja con dos componentes, una imagen en blanco y negro que representa la luminancia y una señal de color que contiene información de crominancia. La percepción del color es básicamente determinada por la luminancia y la crominancia.

### **1.1.2.7 Luminosidad**

La visión humana tiene una respuesta no lineal a la percepción de luminancia que se llama luminosidad. La luminosidad de un color es una medida de su brillo percibido.

Según Muñoz [5], la luminosidad es la cantidad de luz reflejada por una superficie (u objeto) en comparación con la reflejada por una superficie blanca en iguales condiciones de iluminación. Solamente los colores dependientes (aquellos que son percibidos al mismo tiempo que otros colores vecinos pertenecientes a una misma o cercana superficie) poseen luminosidad.

Los grados verbales para la luminosidad son: bastante claro, claro, intermedio, oscuro y muy oscuro.

## **1.2 Modelos de color**

El concepto de modelo de color se puede definir como:

Según Levkowitz [1], un modelo de color es un cuerpo tridimensional utilizado para representar alguna organización de color de acuerdo con una elección particular de tres coordenadas que describen el color; también llamado sólido de color o espacio de color.

Según Gonzalez y Woods [6], un modelo de color es una especificación de un sistema de coordenadas y un subespacio dentro de ese sistema en donde cada color es representado por un solo punto. Mediante la definición de diferentes colores primarios para la representación del sistema se pueden idear diferentes modelos de color.

En cada uno de ellos el color real (del espectro) es reconstruido mediante la combinación de los elementos base de los espacios vectoriales, también llamados colores primarios.

Los espacios de color proporcionan un método racional para especificar, ordenar, manipular y reproducir efectivamente los colores del objeto tomados en consideración. El proceso de selección de la mejor representación de color implica saber cómo se generan las señales de color y qué información se necesita de estas señales. Aunque los espacios de color imponen restricciones a la percepción y representación del color, también ayudan a los seres humanos a realizar tareas importantes; señalan Plataniotis y Venetsanopoulo [7]. En particular, los modelos de color pueden utilizarse para definir colores, discriminar entre colores, juzgar la similitud entre el color e identificar categorías de color para una serie de aplicaciones.

Para utilizar el color como señal visual en aplicaciones multimedia, procesamiento de imágenes, gráficos y visión por computador, se necesita un método apropiado para representar la señal del color. Los diferentes modelos de color responden a esta necesidad.

Los espacios de color de tres dimensiones son los más empleados. Un color se especifica utilizando tres coordenadas, que representan su posición dentro de un espacio de color específico. Estas coordenadas no indican cuál es el color, sino que muestran dónde se encuentra un color en un espacio de color en particular.

Un aspecto importante es la transformación del color, el cambio de coordenadas de un sistema de color a otro. Tal transformación asocia a cada color en un sistema un color en otro modelo. Actualmente, no existe una técnica para determinar el modelo de coordenadas óptimo para todas las aplicaciones de procesamiento de imágenes.

Las relaciones entre los modelos de color y la posibilidad de transformación entre ellos, a través de diferentes manipulaciones, se pueden diagramar de forma resumida como se muestra en la Figura 4.

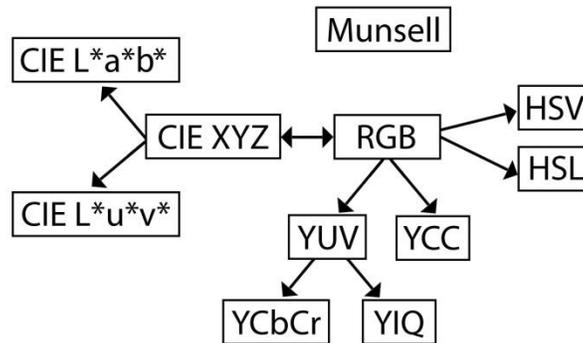


Figura 4: Taxonomía de una selección de modelos de color. [4]

Se explicarán con detalle exclusivamente los modelos RGB y CIELab ya que son los utilizados en el algoritmo desarrollado para este trabajo.

### 1.2.1 RGB

El hecho de que se pueda combinar cualquier color con una mezcla de no más de tres luces primarias es la base de la colorimetría, indica Ware [8]. Una comprensión de la colorimetría es esencial para cualquier persona que desee especificar colores precisamente para la reproducción.

El color se puede describir mediante la ecuación:

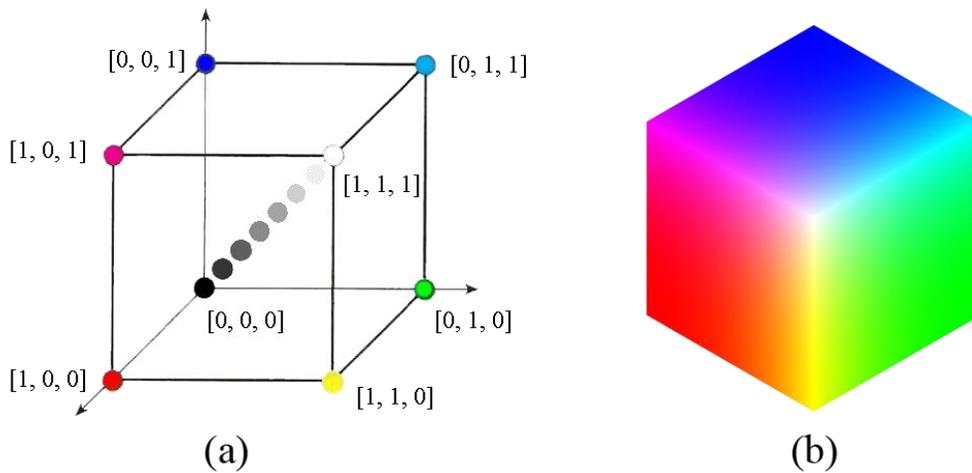
$$C \equiv rR + gG + bB$$

En donde C es el color a combinar, R, G y B son las fuentes primarias de luz a ser utilizadas para crear la combinación, y r, g, y b representan las cantidades de cada luz primaria. El símbolo  $\equiv$  es utilizado para denotar una combinación perceptual.

En este modelo de color, como especifican las investigaciones de Gonzalez y Woods [6], cada color aparece en sus componentes espectrales primarios de rojo, verde y azul. El modelo se basa en el sistema de coordenadas cartesianas de tres dimensiones, donde el subespacio de color de interés es el cubo mostrado en la Figura 5 en el cual los valores primarios RGB están en tres esquinas; los secundarios, cian, magenta y amarillo se sitúan en los otros tres vértices; el negro corresponde al origen; y el blanco se sitúa en el vértice más alejado del origen. En este modelo, la escala de grises (puntos con valores iguales de RGB) ese extiende desde el negro al blanco a lo largo de la recta que une esos dos puntos. Los diferentes colores son puntos dentro o sobre el tetraedro, definidos por los vectores que se extienden desde el origen. Por conveniencia, se asume que todos los vectores han sido normalizados, de modo que el tetraedro de la figura es el cubo unitario, es decir, todos los valores de R, G y B están en el rango [0,1].

Las imágenes representadas en el modelo de color RGB consisten en tres imágenes componentes, una para cada color primario. Cuando se introducen en una pantalla RGB, estas tres imágenes se combinan en la pantalla para producir una imagen de color compuesto. El número de bits utilizados para representar a cada píxel en el espacio RGB es llamado **profundidad del píxel**. Considerando una imagen RGB en la que cada imagen roja, verde y azul es una imagen de 8 bits, bajo esta condición cada píxel de color RGB, que es una tripleta de valores (R,G,B), se dice que tiene una profundidad de 24 bits. El término de imagen

full-color es utilizado comúnmente para denotar una imagen de color RGB de 24 bits. El número total de colores en una imagen RGB de 24 bits es  $(2^8)^3 = 16.777.216$ . En la Figura 5 (b) se muestra el cubo correspondiente de color RGB de 24 bits correspondiente al diagrama de la Figura 5 (a).



**Figura 5: (a) Cubo RGB. (b) Parte externa del cubo RGB. [6]**

El modelo RGB no proporciona un estándar para la especificación exacta del color, ya que el color producido por una especificación RGB depende de la distribución espectral de los primarios y las características de la gama de la pantalla.

Aunque no todos los colores que existen pueden ser mezclados utilizando cantidades no negativas de rojo, verde y azul, la gama es lo adecuadamente grande como para ser suficiente para la mayoría de los propósitos prácticos, indica Levkowitz. Es considerado el modelo de color base para la mayoría de las aplicaciones de imágenes ya que la imagen adquirida no necesita ninguna transformación adicional para mostrarse en pantalla.

La relación en la ecuación del color definida en este modelo es una relación lineal. Esto tiene la consecuencia de que si se duplica la cantidad de luz a la izquierda (C), se puede duplicar la cantidad de luz de cada una de las primarias y la combinación se seguirá manteniendo. Para una matemática más simple, es útil permitir el concepto de luz negativa. Así, se pueden permitir expresiones como:

$$C \equiv -rR + gG + bB$$

A pesar de que la luz negativa no exista en la naturaleza, es útil en la situación en la que se tenga una luz de color que no pueda ser igualada ya que se encuentra fuera de la gama de las tres fuentes primarias. Aún se puede alcanzar la combinación añadiendo parte de las primarias a la muestra.

Si el proyector de luz roja se redirecciona, añadiéndola al color de estudio, se obtiene:

$$C + rR \equiv gG + bB$$

Una vez permitido el concepto de los valores negativos para las primarias, es posible afirmar que cualquier luz de color puede ser igualada por una suma ponderada de tres primarias distintas.

En el sistema RGB existen muchos colores fuera del triángulo de los primarios, por lo que el principal problema de esta representación es que la mayoría de los colores tienen una de las coordenadas negativa, lo que dificulta los cálculos colorimétricos y el cálculo de la luminancia partiendo de las componentes tricromáticas.

### 1.2.2 CIELab

Algunos espacios de color pueden expresar el color de una forma independiente del dispositivo. Mientras que los colores RGB varían con las características de la pantalla o escáner, los colores independientes del dispositivo no dependen de ningún dispositivo en particular y están destinados a ser representaciones verdaderas de los colores que percibe el ojo humano. Estas representaciones de color resultan del trabajo llevado a cabo por la CIE.

Los espacios de color independientes del dispositivo pueden ser utilizados como un espacio de color de intercambio para convertir los datos de color de un espacio de color nativo de un dispositivo al espacio de color nativo de otro dispositivo.

La CIE desarrolló algoritmos para derivar tres componentes primarios imaginarios de color (X, Y, Z) que pueden ser combinados en diferentes niveles para producir todos los colores que el ojo humano puede percibir, definiendo un observador estándar.

Los modelos de color de CIE forman la base de todos los sistemas de gestión de color. El objetivo de este estándar es que para una especificación de color dada se produzca un resultado consistente en diferentes dispositivos, a pesar de las limitaciones del mismo.

Dentro de los espacios de color basados en CIE se encuentran: XYZ, Yxy,  $L^*a^*b^*$  y  $L^*u^*v^*$ .

Los espacios de color CIE XYZ y xyY son perceptualmente no lineales, por lo que no se hace posible evaluar de forma precisa la cercanía perceptual entre colores basándose en sus posiciones relativas en el espacio XYZ o xyY. Los colores cercanos en el espacio xyY pueden ser muy diferentes para el observador, y los colores que se ven muy similares para un observador pueden estar ampliamente separados en el espacio xyY; por lo que en 1978 la CIE produce un conjunto de recomendaciones con el uso de dos espacios de color uniformes que son transformaciones del espacio de color XYZ, como explica C.Ware [9]. Éstos son llamados los espacios CIELab y CIELuv. La razón de existencia de estos dos, en lugar de uno, se debe a la adopción de uno u otro por diferentes industrias. Además, los dos estándares tienen diferentes propiedades que los hacen útiles para diferentes tareas.

En la recopilación realizada por Gaurav [10] indica que los espacios de color  $L^*u^*v^*$  y  $L^*a^*b^*$  son transformaciones no lineales del espacio XYZ. Estos espacios están diseñados para tener una correspondencia más uniforme entre distancias geométricas y distancias perceptuales entre colores que son vistos bajo el mismo iluminante de referencia. Ambos sistemas se basan en la escala de luminosidad,  $L^*$ , y un conjunto de colores oponentes, aproximadamente rojo-verde versus amarillo-azul. Son ligeramente diferentes debido a los diferentes enfoques para su formulación. Las distancias euclidianas en cualquier espacio proporcionan una fórmula de diferencia de color para evaluar las diferencias de color en unidades perceptualmente relevantes.

La luminosidad (tanto para  $CIEL^*a^*b^*$  como  $CIEL^*u^*v^*$ ) se define por la CIE mediante la siguiente ecuación:

$$L^* = \begin{cases} 116(Y/Y_n)^{1/3} - 16, & \text{si } Y/Y_n > 0.008856 \\ 7.787(Y/Y_n) + 16/116, & \text{si } Y/Y_n \leq 0.008856 \end{cases}$$

En donde  $Y_n$  es la luminancia de un blanco de referencia (usualmente normalizada a 1.0 o 100.0). Los valores de  $L^*$  oscilan entre 0 y 100 representando, respectivamente, al negro y al blanco de referencia.

El espacio de color  $CIE L^*a^*b^*$  se define por las siguientes ecuaciones, para valores triestímulo normalizados al blanco que sea mayor que 0.008856:

$$L^* = 116(Y/Y_n)^{1/3} - 16$$

$$a^* = 500[(X/X_n)^{1/3} - (Y/Y_n)^{1/3}]$$

$$b^* = 200[(Y/Y_n)^{1/3} - (Z/Z_n)^{1/3}]$$

$$C_{ab}^* = \sqrt{(a^{*2} + b^{*2})}$$

$$h_{ab} = \tan^{-1}(b^*/a^*)$$

En estas ecuaciones X, Y y Z son los valores triestímulo del estímulo y  $X_n, Y_n, Z_n$  son los valores triestímulo del blanco de referencia.  $L^*$  que representa la luminosidad,  $a^*$  indica la coordenada de la sensación rojo-verde (si  $a^* > 0$  se percibirá con parte de rojo, si  $a^* < 0$  se percibirá con parte de verde),  $b^*$  indica la coordenada de la sensación amarillo-azul (si  $b^* > 0$  se percibirá con parte de amarillo, si  $b^* < 0$  se percibirá con parte azul),  $C_{ab}^*$  es el cromatismo y  $h_{ab}$  el tono. Las coordenadas  $L^*, a^*$  y  $b^*$  son utilizadas para construir un espacio de color cartesiano como se ilustra en la Figura 6 (izquierda). En aquellos casos en los que  $a^* = b^* = 0$ , se encuentran los colores que son acromáticos. Las coordenadas  $L^*, C_{ab}^*$  y  $h_{ab}$  son las coordenadas cilíndricas para la representación del mismo espacio, como se observa en la Figura 6 (derecha).

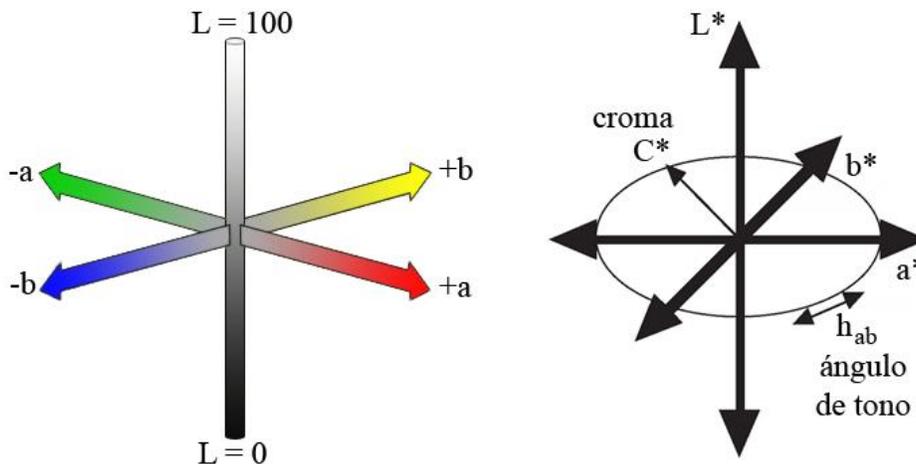


Figura 6: Representación cartesiana del espacio  $CIE L^*a^*b^*$ . [4] [11]

La distancia euclídea entre dos estímulos de color en el espacio  $CIE L^*a^*b^*$  es denotado por  $\Delta E_{ab}^*$  y es el total de la medida de la diferencia de color entre dos vectores, m y p:

$$\Delta E_{ab}^* = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2}$$

En donde:

$$\Delta L^* = L_p^* - L_m^*$$

$$\Delta a^* = a_p^* - a_m^*$$

$$\Delta b^* = b_p^* - b_m^*$$

### 1.3 Algoritmos de transformación de color a escala de grises

La conversión de imágenes digitales de color a escala de grises se puede realizar a través de diferentes cálculos que conllevan a la reducción de los datos del color de tres dimensiones a una dimensión, por lo que la pérdida de información durante la conversión es inevitable.

Dentro de los numerosos algoritmos que trabajan la transformación de color a escala de grises se procede a seleccionar cuatro de ellos, tres triviales (definidos por Kanan y Cottrell [12] en sus investigaciones) y uno optimizado. Los triviales son los más utilizados en las aplicaciones de edición de imágenes por su simplicidad pero sus resultados pueden no ser los mejores por la pérdida de información de color que conllevan, y el optimizado es escasamente utilizado pero utiliza un método para estimar la mejor correspondencia y para preservar ciertos aspectos de la información de color, como la información espacial (por ejemplo, los píxeles vecinos) y parámetros globales (como luminancia o contraste). El resultado de este procesamiento busca mejoras perceptuales, mejor brillo, contraste y mayor realce de detalles que los métodos convencionales.

#### 1.3.1 Luminosidad HSL (*Lightness HSL*)

El valor de la escala de grises ( $G_{Luster}$ ) se toma de la luminosidad de la representación HSL de la imagen. Este valor es la media entre el máximo y el mínimo de los valores del color. En este método un valor del color es descartado de cada píxel, los valores restantes se promedian y la información se pierde en términos del valor de color que se descarta del píxel. Es llamado *Luster* para diferenciarlo del canal de luminosidad de CIELab.

$$G_{Luster} = \frac{1}{2}(\max(R, G, B) + \min(R, G, B))$$

#### 1.3.2 Promedio

Este método se denomina algoritmo de intensidad y calcula el valor de la escala de grises ( $G_{Intensity}$ ) como el promedio de los canales RGB de la imagen:

$$G_{Intensity} = \frac{1}{3}(R + G + B)$$

### 1.3.3 Luminancia (*Luminance*)

Es un método que está diseñado para que el resultado de la escala de grises ( $G_{Luminance}$ ) coincida con la percepción humana del brillo mediante el uso de una combinación ponderada de los canales RGB:

$$G_{Luminance} = 0.3R + 0.59G + 0.11B$$

### 1.3.4 Luminosidad (*Lightness*)

La luminosidad es una representación perceptualmente uniforme de la escala de grises utilizada en los espacios de color de CIELab y CIELUV. Esto significa que la luminosidad se corresponde estrechamente a la percepción humana, y que se consigue mediante la transformación no lineal del espacio de color RGB.

$$G_{Lightness} = \frac{1}{100} (116f(Y) - 16)$$

En donde  $Y = 0.2126R + 0.7152G + 0.0722B$ , y

$$f(t) = \begin{cases} t^{1/3}, & \text{si } t > 0.008856 \\ 7.787 t + 16/116, & \text{otro caso} \end{cases}$$

Se ha normalizado el rango de luminosidad entre 0 y 1, en lugar del rango usual de 0 a 100.

### 1.3.5 Decoloración espectral

En este trabajo, Zhao y Tamimi [13] proponen un método para decolorar una imagen en color a escala de grises, preservando el contraste cromático. La conversión se realiza en el dominio espectral después de aplicar transformadas de Fourier en los canales cromáticos y de luminancia, y la intensidad resultante se recupera con una transformada inversa.

Desviándose del espacio de píxeles de una imagen, la preservación del contraste se basa en una imagen transformada en el espacio de frecuencia. La transformación calcula inherentemente la magnitud de los cambios (por ejemplo, el contraste) entre los canales, en todas las escalas espaciales. Por consiguiente, solo se necesitan operaciones aritméticas sencillas para añadir diferencias cromáticas a la intensidad final de escala de grises.

Este algoritmo propone una decoloración que provee la preservación controlable del contraste en todas las escalas espaciales. Se utiliza el espacio CIELab, ya que provee una buena base operacional. Además, los usuarios pueden controlar de forma flexible los efectos de la decoloración mediante el ajuste interactivo de parámetros visualmente intuitivos, además de los valores predefinidos.

Empleando este método, se calcula la transformada de Fourier en el canal de luminancia ( $L$ ), y en los dos canales cromáticos ( $a$  y  $b$ ) de una imagen, respectivamente. Los resultados  $\hat{L}$ ,  $\hat{a}$  y  $\hat{b}$ , están directamente relacionados con los índices espaciales del cambio de intensidad en todas las escalas espaciales. Cada frecuencia del espectro (ejemplo, la magnitud) refleja de forma inherente el nivel de contraste en cada escala correspondiente, entre los tres componentes de la imagen. Una imagen en escala de grises de la correspondencia de luminancia puede ser recuperada de una transformación inversa de Fourier de  $\hat{L}$ .

Este método implementa la decoloración aumentada modificando  $\hat{L}$  incorporando compensación de  $\hat{a}$  y  $\hat{b}$ . Posteriormente la transformada inversa logra una imagen en escala de grises deseada, preservando las

diferencias visuales de luminancia y crominancia. Se provee un esquema predefinido para calcular dos coeficientes utilizados en la adición del contraste cromático con el contraste de luminancia: un coeficiente para definir el grado del contraste cromático a ser incorporado, y otro para determinar los diferentes niveles de contraste cromático de dos canales diferentes, respectivamente. Además del cálculo automático, los coeficientes pueden ser definidos de forma flexible por los usuarios para las distintas tareas.

El método solo implica operaciones aritméticas simples, excepto numerosas transformadas de Fourier. Las transformaciones son aplicadas por un algoritmo de transformada rápida de Fourier (FFT – *Fast Fourier Transform*), considerando el número de píxeles  $N$ .

### 1.3.5.1 Optimización del contraste

Realizando la transformada de Fourier de cada canal (L,a,b) de una imagen a color  $I$ , se obtienen tres imágenes espectrales:  $\hat{L}$ ,  $\hat{a}$  y  $\hat{b}$ , con valores complejos. Una imagen en escala de grises convencional  $G_I$ , puede lograrse a través de una transformación inversa de  $\hat{L}$ :  $G_I = IF(\hat{L})$ , donde  $IF(\ )$  representa la inversa de la transformada de Fourier. Se introduce el contraste cromático en una imagen en escala de grises mejorada,  $\tilde{G}_I$ , en el dominio de Fourier.  $\tilde{G}_I$  es calculada por la transformada inversa de Fourier como:

$$\begin{aligned}\tilde{G}_I &= IF(\hat{E}) \\ \hat{E} &= H(\hat{L}, \hat{a}, \hat{b})\end{aligned}\quad (1)$$

La función  $H$  calcula una intensidad de la escala de grises modificada en el dominio de Fourier  $\hat{E}$ , a partir de las contrapartes del dominio de Fourier del canal de luminancia original y los dos canales cromáticos.  $H$  se implementa en cada frecuencia como:

$$H(\hat{L}, \hat{a}, \hat{b}) = (1 - \theta)\hat{L} + \theta(\phi\hat{a} + (1 - \phi)\hat{b})\quad (2)$$

Donde  $\theta$  controla el grado del contraste cromático incorporado al resultado en escala de grises, y  $\phi$  es un coeficiente para determinar las contribuciones relativas de los canales  $a$  y  $b$ . En la ecuación (2), todos los valores de Fourier y coeficientes son dependientes de la frecuencia  $w$ , que se omite por claridad.

### 1.3.5.2 Control de parámetros

Dos coeficientes controlables,  $\theta$  y  $\phi$ , determinan diversos efectos de aumento de contraste en los resultados en escala de grises. Pueden calcularse automáticamente basándose en el hecho de los datos de los espectros de Fourier.  $\theta$  modela el grado del contraste cromático incorporado, que puede ser determinado vinculándolo con la pérdida relativa de conversión medida por la comparación de la diferencia RGB y la diferencia de luminancia. En este esquema, estas diferencias son modeladas por la operación del espectro en cada frecuencia:

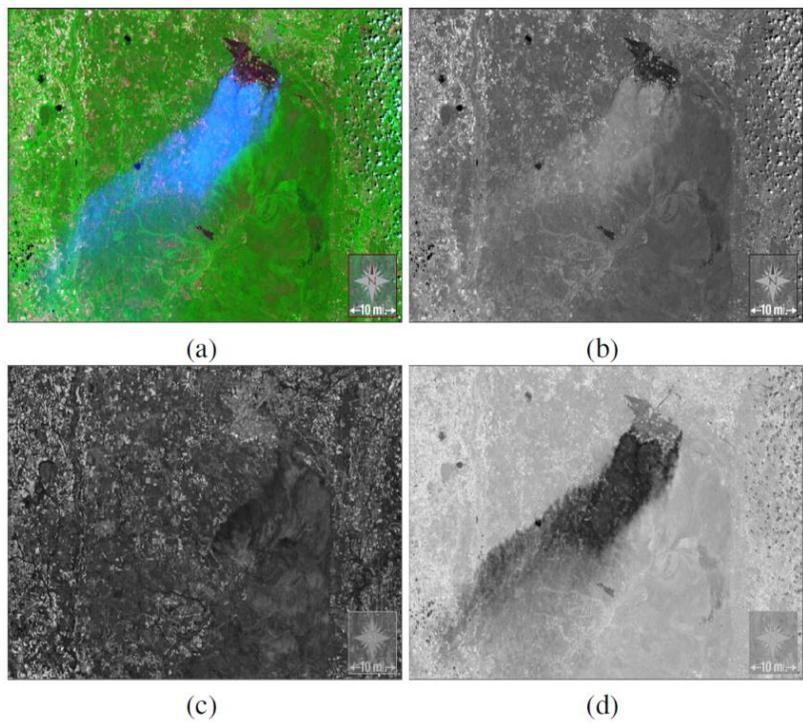
$$\theta = \frac{|\hat{R}| + |\hat{G}| + |\hat{B}| - |\hat{L}|}{|\hat{R}| + |\hat{G}| + |\hat{B}|}\quad (3)$$

Donde  $| \ |$  representa el espectro de los valores complejos.  $\hat{R}$ ,  $\hat{G}$ ,  $\hat{B}$  son los resultados de la transformada de Fourier de los canales R, G, B. Mientras tanto,  $\phi$  es calculada por la proporción relativa del espectro  $a$  y  $b$ :

$$\phi = \frac{|\hat{a}|}{|\hat{a}|+|\hat{b}|} \quad (4)$$

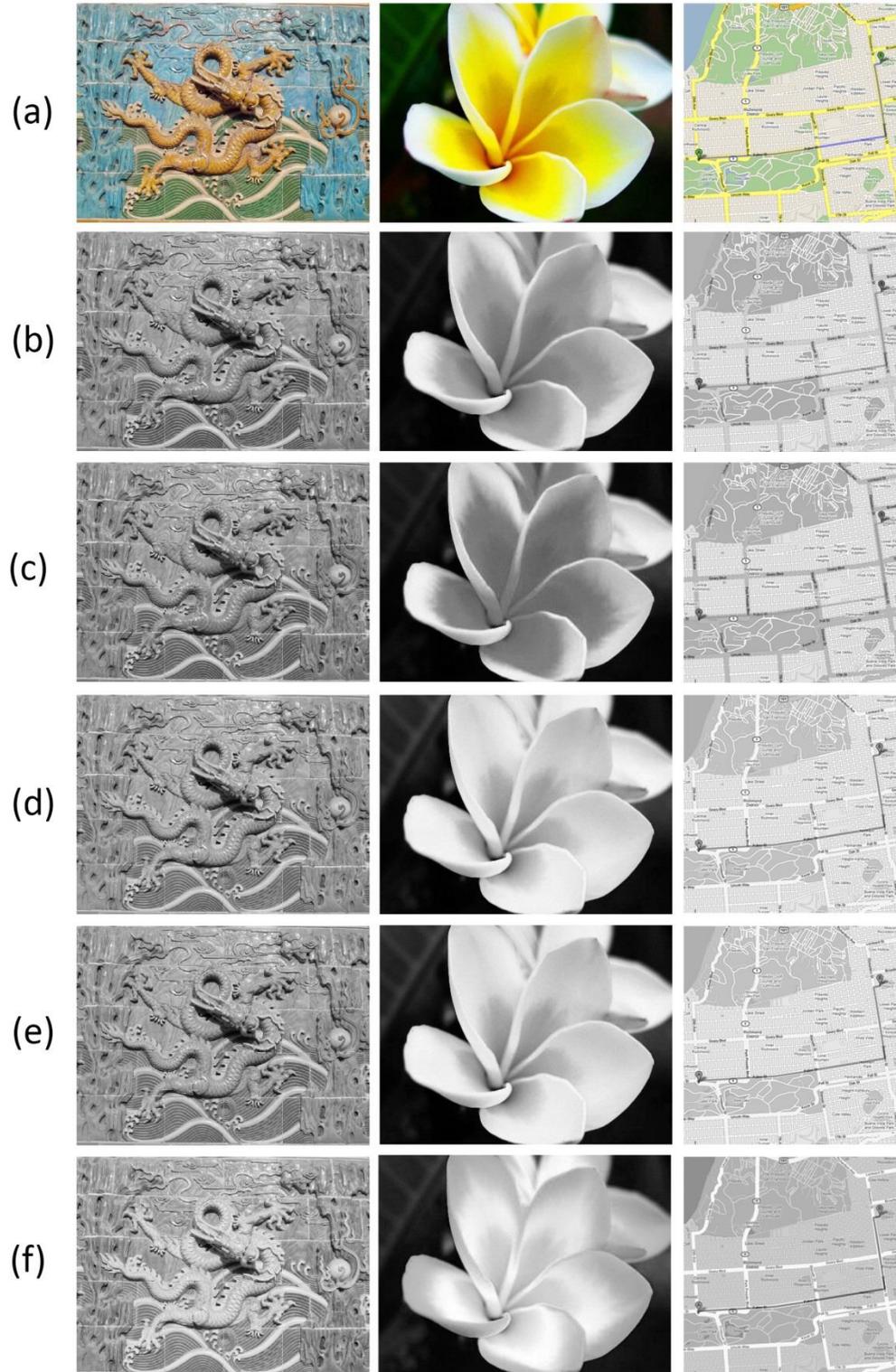
$\theta$  y  $\phi$  pueden ser calculadas automáticamente en cada frecuencia  $w$  y aplicarse en la ecuación (2). En la práctica, el uso de un promedio de  $\theta$  y un promedio de  $\phi$  desde todas las frecuencias, generalmente puede proporcionar resultados claros, sin artefactos causados por operaciones individuales de frecuencia. De esta manera, los resultados satisfacen la consistencia global, por ejemplo, píxeles con el mismo color corresponden a la misma escala de gris, debido a la linealidad de la transformada de Fourier aplicada en la ecuación (2).

En la Figura 7, se muestran los efectos de diferentes valores de  $\phi$  de la correspondencia de una imagen de paisaje por satélite de los incendios por una grave sequía. Dos casos extremos son utilizados: (c) aumenta considerablemente los puntos rojos no obvios en luminancia con  $\phi = 0.1$ , y (d) aumenta las regiones azules del humo, con  $\phi = 0.9$ .



**Figura 7: Correspondencia en escala de grises de una imagen de paisaje por satélite de los incendios récord por una sequía en Georgia: (a) imagen a color; (b) luminancia; (c) decoloración del espectro con  $\theta = 0.6$ ,  $\phi = 0.1$ ; (d) decoloración del espectro con  $\theta = 0.6$ ,  $\phi = 0.9$ . [13]**

En la Figura 8 se puede observar una comparación cualitativa de los algoritmos descritos en esta sección. Se muestra los resultados de las conversiones comparando las imágenes de un dragón amarillo, una flor y mapa de rutas digital; en donde el algoritmo de decoloración espectral genera un dragón visualmente separable del fondo, regiones amarillas en los pétalos blancos se tornan divisibles y muestra una mayor información de la ruta, junto con la mejora de las carreteras, que desempeña un papel significativo de referencia en el entendimiento de mapas de rutas y su uso.



**Figura 8: Comparación cualitativa de los algoritmos de transformación de color a escala de grises. (a) Imágenes a color, (b) Promedio ingenuo, (c) Luminosidad HSL, (d) Luminancia, (e) Luminosidad y (f) Decoloración espectral. [13]**

## 2 EVALUACIÓN DE CALIDAD DE LAS IMÁGENES

La evaluación de la calidad de la imagen (*image quality assessment* - IQA) proporciona una herramienta útil para evaluar el efecto visual de una amplia gama de artefactos impuestos en las imágenes digitales en el proceso de adquisición, procesamiento, transporte, compresión, reproducción y almacenamiento.

El uso de la IQA juega un papel importante dentro de la evaluación comparativa de un sistema de procesamiento de imágenes y algoritmos. Por ejemplo, la métrica de calidad se utiliza para seleccionar uno de varios sistemas de procesamiento de imágenes que proporcionan las imágenes de mejor calidad. Los enfoques existentes de la IQA, planteados por Gupta [14], se dividen en dos categorías: evaluación subjetiva y evaluación objetiva.

En este trabajo se ha seleccionado específicamente, en el caso de la evaluación subjetiva, la comparación por pares, junto con el método de análisis de los resultados obtenidos, y, de la evaluación objetiva, el algoritmo C2G-SSIM.

### 2.1 Métrica subjetiva de comparación por pares

La evaluación subjetiva de la calidad se considera un método confiable de evaluación de la calidad y, a menudo, se emplea para recopilar puntajes de calidad. Dos de los principales métodos de evaluación de calidad subjetiva para contenido multimedia, estudiados por E. Zeman et al. [15], son la clasificación directa (*rating*) y la clasificación (*ranking*). Los métodos de calificación directa solicitan a los observadores que asignen puntajes a los estímulos observados. Los métodos de clasificación solicitan a los observadores que comparen dos o más estímulos y los ordenen según su calidad. El método de clasificación más comúnmente empleado son las comparaciones por pares, debido a la simplicidad de la tarea y la consistencia de los resultados.

Una forma de medir un atributo perceptual de interés (por ejemplo: más bello, mejor formado, más aceptable, mejor calidad de imagen, o cualquiera que sea el atributo subjetivo que se mide) según la investigación presentada por Pérez-Ortiz y Mantiuk [16], es pedir a los sujetos de prueba ( $m$ ) del experimento que clasifiquen un conjunto de condiciones, por ejemplo, imágenes. La clasificación más simple es a través de las combinaciones por pares, donde solo se muestran dos condiciones a la vez y se le solicita a un participante que elija una de ellas de acuerdo con algunos criterios específicos, permitiendo de esta forma medir un atributo perceptivo de interés, como la calidad de imagen.

Cada par de comparación presentado resulta en una elección binaria, indican Brown y Peterson [17], donde se juzga un conjunto de estímulos, o elementos, al presentar todos los pares posibles de elementos a cada encuestado que elige para cada par el elemento que mejor satisface el criterio de elección especificado (por ejemplo, más preferido, más grave, más hermoso). Las elecciones permiten el cálculo de un conjunto de *valores de escala* que indican la posición de los elementos a lo largo de la dimensión especificada. Los valores de escala pueden estimarse para un encuestado individual o para un grupo de encuestados. Cabe destacar que a los encuestados no se les ofrece una opción de indiferencia cuando se les presenta un par de elementos.

En la planificación de los experimentos de comparación por pares el número de comparaciones requeridas depende del número de elementos,  $t$ , a evaluar, siendo en total  $t(t - 1)/2$  comparaciones. El resultado de las encuestas se organiza en una matriz  $t$  por  $t$ . Las matrices de respuesta de todos los encuestados en la muestra se pueden sumar para suministrar una matriz ( $F$ ) llamada *matriz de frecuencia*, cuyo elemento general  $f_{ij}$  representa el número de veces que la opción  $O_j$  se prefirió sobre  $O_i$ , cuando  $i \neq j$ , y cuando

$i = j$ , su diagonal, toma los valores de  $m/2$ . Estableciendo, en este trabajo,  $i$  como el identificador de la fila y  $j$  como el identificador de la columna de dicha matriz. Las variaciones de este orden no son consideradas en este trabajo.

Por ejemplo, en un experimento con tres condiciones, la matriz resultante podría ser la siguiente:

$$F = \begin{bmatrix} 0 & 3 & 0 \\ 27 & 0 & 7 \\ 30 & 23 & 0 \end{bmatrix}$$

En donde,  $c_{12} = 3$  indica que la condición  $O_2$  ha sido seleccionada 3 veces como mejor que la condición  $O_1$ , y  $c_{21} = 27$  indica que la condición  $O_1$  ha sido seleccionada 27 veces como mejor que  $O_2$ .

Por ejemplo, para analizar cuál de los tres métodos de rasterización (A, B y C) produce los resultados de mayor calidad, se presentan las imágenes producidas por estos métodos en pares (AB, BC, AC) y luego se pregunta a los observadores cuál imagen en cada par tiene mejor calidad. Si se recopilan suficientes datos, se pueden clasificar los algoritmos de inferior a superior y escalar las puntuaciones de clasificación para que puedan interpretarse fácilmente en términos de probabilidad de una mejor calidad percibida. Una representación de esta estrategia se puede ver en la Figura 9, donde se comparan 4 condiciones (tipos de distorsión), lo que resulta en 6 comparaciones diferentes ( $4(4 - 1)/2$ ), cada comparación repetida 30 veces. Los algoritmos de escalado producen una escala de calidad a partir de la matriz de comparaciones, en la que las distancias entre condiciones pueden interpretarse como la probabilidad de una mejor calidad percibida.

#### Objetos comparados



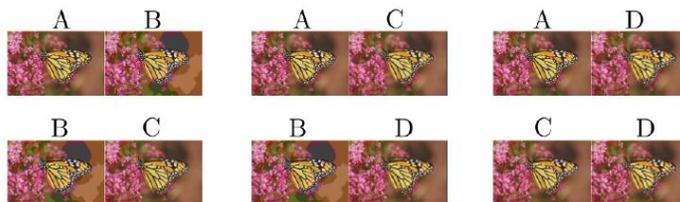
A: Imagen de referencia

B: JPEG

C: Desenfoque

D: Desvanecimiento

#### Experimentos de comparación por pares



$$\begin{matrix} & \begin{matrix} A & B & C & D \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \end{matrix} & \begin{pmatrix} 15 & 0 & 1 & 0 \\ 30 & 15 & 27 & 20 \\ 29 & 3 & 15 & 25 \\ 30 & 10 & 5 & 15 \end{pmatrix} \end{matrix}$$

Matriz de comparaciones

#### Escalado

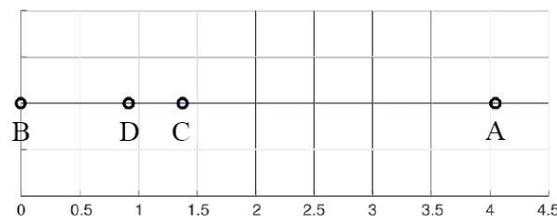


Figura 9: Escalado de datos de comparación por pares para evaluar la calidad percibida de la imagen. [16]

La comparación por pares presenta numerosas ventajas: i) conduce a una tarea experimental muy simple y, por lo tanto, es adecuada para participantes no expertos, ii) evita problemas de calibración que se encuentran con frecuencia en las mediciones cardinales, iii) en general proporciona una mayor sensibilidad y un error de medición más bajo en comparación con la calificación directa y iv) puede ser más rápido de ejecutar que la escala directa (particularmente porque hacer comparaciones por pares es más fácil y más rápido para los participantes).

### **2.1.1 Elección forzada de dos alternativas (2AFC)**

En el trabajo llevado a cabo por Ćadík [18] se especifica un experimento subjetivo perceptual para evaluar las conversiones de imágenes de color a escala de grises, llamado experimento de precisión, donde las imágenes en escala de grises se presentan junto con la imagen en color original (referencia). Dicho experimento se realiza utilizando la técnica psicofísica de comparaciones por pares conocida como el paradigma de experimentos 2AFC (*two alternatives forced choice* - Elección forzada de dos alternativas).

El experimento consiste en que cada vez se van presentando dos imágenes en escala de grises junto con la de color original en el centro. Los observadores deben de seleccionar una de las dos imágenes en escala de grises que esté más cerca en apariencia de la imagen en color original, es decir, la imagen que mejor reproduzca a la original.

Las imágenes en escala de grises presentadas pueden ser el resultado de diferentes algoritmos de conversión o un mismo algoritmo con variaciones de los parámetros (si aplica); lo cual se define según la evaluación que se desee realizar.

Idealmente, todas las imágenes del estudio de evaluación de calidad subjetivo se deben de realizar en una misma sesión, de modo que los desajustes de escala entre los sujetos sean minimizados. Una forma de incrementar el número de imágenes en el experimento es usar varias sesiones utilizando diferentes conjuntos de imágenes. Cada sesión de evaluación se realiza en un tiempo que no debería superar los treinta minutos por observador, para evitar su fatiga.

Las imágenes a evaluar se deben de presentar en un contorno controlado, con una serie de características establecidas y descritas; como es la pantalla, su calibración y tamaño, el color de fondo de las imágenes presentadas, su tamaño y resolución, la iluminación del salón donde se realizan las pruebas, la distancia del observador a la pantalla, entre otras. Además, se debe especificar el número de observadores, su género, edad, deficiencias visuales (si aplica), entre otras.

Los resultados de este experimento podrán ser estudiados a través de distintas técnicas, aunque para este trabajo se selecciona específicamente la metodología presentada por Thurstone.

### **2.1.2 Recuento de votos vs. escalamiento**

La forma más sencilla de informar el resultado de un experimento de comparación por pares es calcular los conteos de votos: el número de veces que se seleccionó una condición como mejor que cualquier otra condición. Los conteos de votos, sin embargo, presentan los resultados en una escala ordinal, que generalmente produciría la clasificación correcta de las condiciones, pero no captura correctamente la magnitud de las diferencias entre las condiciones. Por otro lado, la escala de comparación por pares coloca esas condiciones en una escala de intervalo continuo, que captura tanto el orden de las condiciones como la magnitud de la diferencia. Zerman et al. [15] confirman que las magnitudes de calidad se capturan mejor cuando se escalan los datos de comparación por pares. A diferencia de los conteos de votos, los

métodos de escala introducen un paso adicional para convertir las probabilidades de preferencia en una escala de intervalo de calidad.

En la literatura psicométrica, la investigación realizada por diferentes científicos ha demostrado que la capacidad de las personas para juzgar consistentemente los pares de estímulos se relaciona con la proximidad de los estímulos en la dimensión del interés: cuanto más cercanos están los elementos, menor es la capacidad de las personas para compararlos de manera consistente. Esta inconsistencia resulta muy útil, ya que proporciona información acerca de qué tan cerca están los estímulos en el continuo de valores correspondiente a la dimensión de interés. Los métodos de escala psicométricos, diseñados para hacer uso de la inconsistencia, ofrecen, en teoría, mejoras sobre las puntuaciones de las preferencias medias simples.

Dentro de los modelos comunes para analizar las comparaciones por pares se encuentra el modelo establecido por Thurstone, quien propone el escalamiento psicológico y proporciona un método para estimar la diferencia de puntaje de calidad para dos opciones utilizando la Ley del Juicio Comparativo; seleccionado para este trabajo y explicado en la siguiente sección.

### 2.1.3 Ley del juicio comparativo de Thurstone

La Ley del Juicio Comparativo, explicada con detalle por Thurstone [19], es aplicable no solo a la comparación de las intensidades de estímulos físicos, con magnitud física medible, sino también a los juicios comparativos cualitativos, o “subjetivos” de atributos no físicos, como por ejemplo “la excelencia de escritura a mano”, y se ha aplicado en la medición de tales valores psicológicos como una serie de opiniones.

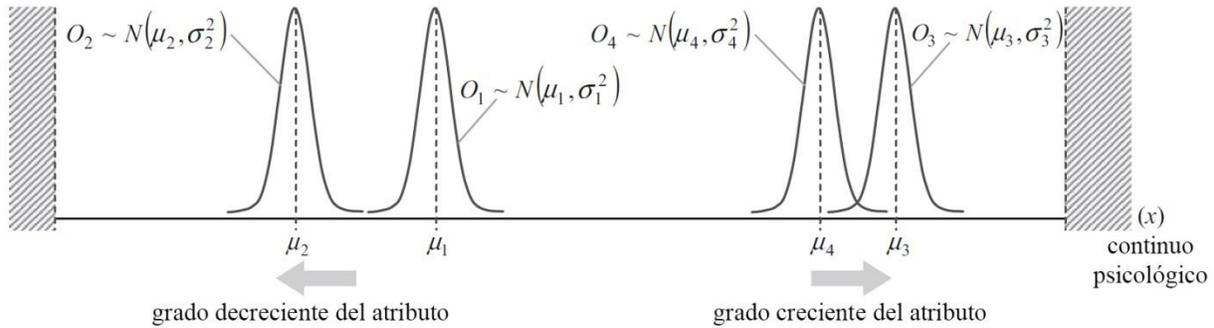
Thurstone postuló la existencia de un *continuo psicológico*, es decir, una escala unidimensional abstracta y desconocida, en la que los objetos se posicionan según el grado de un determinado atributo, es decir, una característica específica de los objetos, que evoca una respuesta subjetiva en cada uno de los jueces. La posición de un objeto es directamente proporcional al valor del atributo, es decir, aumenta a la derecha y disminuye a la izquierda de la escala.

Los datos necesarios para iniciar este procedimiento son los obtenidos mediante el método de las comparaciones por pares, que contienen los juicios de los sujetos que evaluaron los estímulos. Para analizar estos datos y elaborar con ellos la escala psicológica, Thurstone propone un modelo matemático partiendo de una serie de conceptos y supuestos.

En concreto parte de la idea de que el juicio comparativo de dos estímulos puede variar entre distintos sujetos u ocasiones debido a fluctuaciones momentáneas del organismo. Se asume que el estímulo que más frecuentemente es juzgado superior a otro tiene un valor más alto en el continuo psicológico. Así es posible relacionar la probabilidad de cada estímulo de ser juzgado “mayor que” con sus valores en el continuo.

Al efecto psicológico que provoca una reacción en concreto en un sujeto, se le denomina proceso discriminativo, que no es fijo sino que varía aleatoriamente. Los distintos procesos discriminativos dan lugar a una distribución discriminativa que se asume continua y normal. La media constituye el llamado *proceso discriminativo modal*, que se hará corresponder con el valor escalar (la puntuación) del estímulo psicológico en el continuo, y la desviación típica es la *dispersión discriminativa*; se puede observar un ejemplo de estas definiciones en la Figura 10, en donde se representan cuatro condiciones  $O_n$ . De esta forma, la proximidad entre dos estímulos, en este caso psicológicos, dependerá del solapamiento de sus distribuciones discriminativas. Una mayor confusión en los juicios acerca de qué estímulo del par

presenta más cantidad de un atributo, indicará un mayor solapamiento entre las distribuciones. Por ejemplo, si el estímulo A es juzgado mayor que B en un 40% de los casos, y A es juzgado mayor que C en un 90% de los casos, las distribuciones de A y B estarán más solapadas que las de A y C, lo que implicará que A y B están más próximos en el continuo psicológico y tienen valores escalares más similares.



**Figura 10: Representación del continuo psicológico y la posición (desconocida) de cuatro objetos ficticios ( $O_1$  a  $O_4$ ), según el grado del atributo de interés. [20]**

El concepto fundamental detrás de esta ley es que la proporción de veces que se estimó que un estímulo tiene más de un atributo dado que otro, se relaciona con el número de unidades que separan las dos sensaciones en una escala psicológica que representa esa calidad.

Partiendo en que Thurstone propone un modelo que asume que la calidad de una opción es una variable aleatoria Gaussiana y que el nivel de calidad de cada opción se toma como la calidad media del gaussiano correspondiente, Tsukida y Gupta [21] demuestran que se puede considerar el caso básico de dos opciones, donde las variables aleatorias Gaussianas A y B representan la calidad de la opción A y de la opción B, respectivamente,

$$A \sim N(\mu_A, \sigma_A^2), \quad B \sim N(\mu_B, \sigma_B^2)$$

Sus funciones de densidad de probabilidad (FDP) son,

$$p_A(A = a) = \frac{1}{\sigma_A} \phi\left(\frac{a - \mu_A}{\sigma_A}\right), \quad p_B(B = b) = \frac{1}{\sigma_B} \phi\left(\frac{b - \mu_B}{\sigma_B}\right)$$

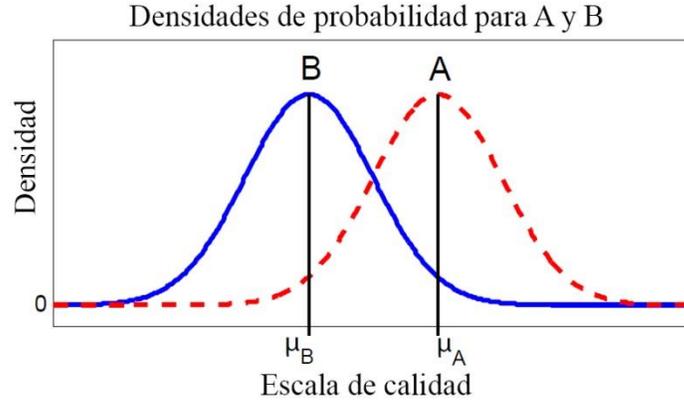
Donde  $\phi$  es la normal estándar FDP (media cero, varianza unitaria),

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$$

Como se muestra en la Figura 11, la opción A se coloca en la escala de calidad en  $\mu_A$ , y la opción B se coloca en la escala de calidad en  $\mu_B$ .

El modelo de Thurstone selecciona la opción A sobre la opción B si la diferencia de calidad aleatoria  $A - B$  es mayor que cero,

$$P(A > B) = P(A - B > 0)$$



**Figura 11:** Funciones de densidad de probabilidad de A y B, la calidad aleatoria de dos opciones. El eje x representa la escala de calidad, donde cada opción se coloca en la escala de calidad en su media. [21]

Dado que  $A - B$  es la diferencia de dos Gaussianas,  $A - B$  es una variable aleatoria Gaussiana:

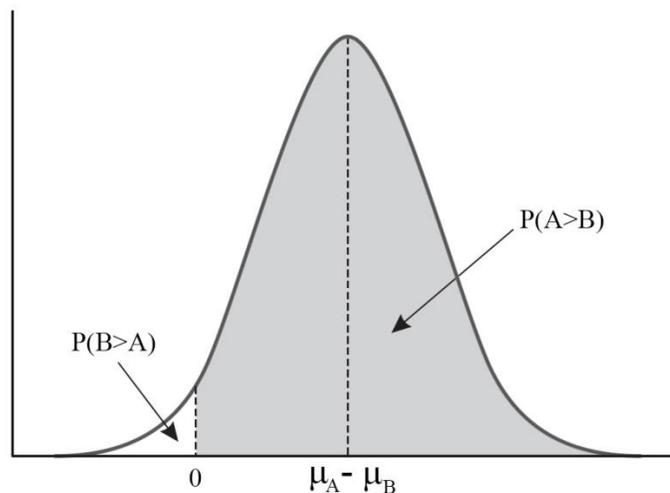
$$A - B \sim N(\mu, \sigma)$$

$$\mu_{AB} = \mu_A - \mu_B$$

$$\sigma_{AB}^2 = \sigma_A^2 + \sigma_B^2 - 2\rho_{AB}\sigma_A\sigma_B \quad (5)$$

Donde  $\mu_{AB}$  es la media de la diferencia de calidad de  $A - B$ ,  $\sigma_{AB}$  es la desviación estándar de la diferencia de calidad aleatoria  $A - B$  y  $\rho_{AB}$  es la correlación entre A y B.

El resultado del juicio de comparación por pares se relaciona, por lo tanto, con la distribución de la diferencia entre las dos distribuciones de procesos discriminiales para la prueba A y la prueba B. Si esta diferencia es positiva, tenemos el juicio “A vence a B”, y si es negativo tenemos el juicio “B vence a A”. La distribución de las diferencias se muestra en la Figura 12.



**Figura 12:** La diferencia de calidad aleatoria  $A - B$  es una gaussiana con media  $\mu_A - \mu_B$ . El área sombreado bajo la curva FDP de  $A - B$  es la  $P(A > B)$ . [21] [22]

Por lo tanto, la probabilidad de elegir la opción A sobre la opción B, es:

$$\begin{aligned}
 P(A > B) &= P(A - B > 0) \\
 &= \int_0^{\infty} \frac{1}{\sqrt{2\pi\sigma_{AB}^2}} e^{-(x-\mu_{AB})^2/(2\sigma_{AB}^2)} dx \\
 &= \int_{-\mu_{AB}}^{\infty} \frac{1}{\sqrt{2\pi\sigma_{AB}^2}} e^{-x^2/(2\sigma_{AB}^2)} dx
 \end{aligned}$$

Por la simetría de las Gaussianas,

$$\begin{aligned}
 &= \int_{-\infty}^{\mu_{AB}} \frac{1}{\sqrt{2\pi\sigma_{AB}^2}} e^{-x^2/(2\sigma_{AB}^2)} dx \\
 &= \int_{-\infty}^{\mu_{AB}} \frac{1}{\sigma_{AB}} \phi\left(\frac{x}{\sigma_{AB}}\right) dx \\
 &= \int_{-\infty}^{\frac{\mu_{AB}}{\sigma_{AB}}} \phi(t) dt \\
 &= \Phi\left(\frac{\mu_{AB}}{\sigma_{AB}}\right) \tag{6}
 \end{aligned}$$

Donde  $\Phi(z)$  es la función de distribución acumulativa (FDA) de la normal estándar,

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-t^2/2} dt = \int_{-\infty}^z \phi(t) dt$$

Al invertir (6), se puede calcular la diferencia de calidad media  $\mu_{AB}$ , (asignando probabilidades en diferencias de calidad) como,

$$\mu_{AB} = \sigma_{AB} \Phi^{-1}(P(A > B))$$

Donde  $\Phi^{-1}(x)$  es la FDA inversa de la normal estándar (conocida como *probit*, mostrada en la Figura 13). La inversa de la FDA de la normal estándar es conocida comúnmente como la puntuación  $z$  ( $z$  - *score*) o puntuación estándar ya que provee el número de desviaciones estándar que está  $x$  de la media, que serán positivas para todos los valores de  $P(A > B)$  sobre 0.5 y negativas para todos los valores de  $P(A > B)$  por debajo de 0.5.

Thurstone propuso estimar la  $P(A > B)$  mediante la proporción empírica, o probabilidad, de personas que prefieren A sobre B,  $f_{B,A}/f_{B,A} + f_{A,B}$ . Asumiendo que se conoce (o que se puede estimar) la desviación  $\sigma_{AB}$ , el estimador para la diferencia de calidad  $\hat{\mu}_{AB}$  es,

$$\hat{\mu}_{AB} = \sigma_{AB} \Phi^{-1}\left(\frac{f_{B,A}}{f_{B,A} + f_{A,B}}\right) \tag{7}$$

La estimación (7) se conoce como la Ley del Juicio Comparativo de Thurstone.

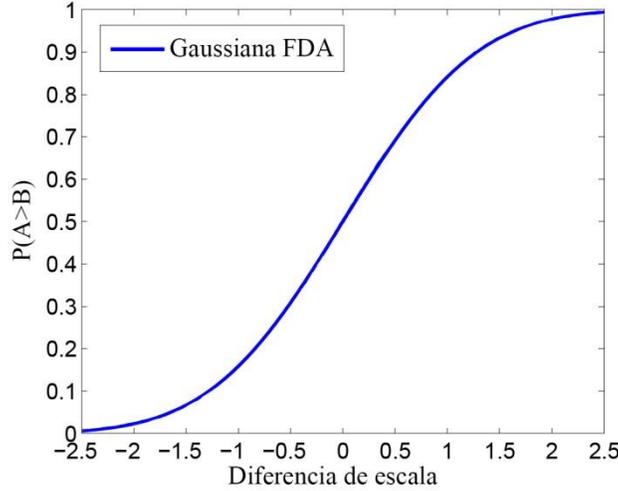


Figura 13: Gaussiana FDA. [21]

Cuando hay valores extremos en los juicios, la proporción  $f_{B,A}/(f_{B,A} + f_{A,B})$  es 0 o 1, y sucede que  $\Phi^{-1}(0) = -\infty$  y  $\Phi^{-1}(1) = +\infty$ , por lo que una solución a este problema es modificar las entradas de la matriz de frecuencias  $F$ , siendo  $m$  el número de personas encuestadas para esa comparación:

$$\tilde{f}_{ij} = \begin{cases} \frac{1}{2} & \text{si } f_{ij} = 0 \\ f_{ij} - \frac{1}{2} & \text{si } f_{ij} = m \\ f_{ij} & \text{otro caso} \end{cases}$$

### 2.1.4 Caso V de Thurstone

El caso I asume que la correlación  $\rho_{AB}$  es constante en todas las comparaciones. El Caso II agrega la suposición de que el modelo general puede aplicarse a los juicios de un grupo de observadores (a diferencia de los múltiples juicios del mismo observador). El caso III además supone que A y B no están correlacionados, de modo que  $\rho_{AB} = 0$ . El caso IV adicionalmente supone que las varianzas son aproximadamente iguales,  $\sigma_A = \sigma_B + \epsilon$ , donde  $\epsilon$  es pequeña. El caso V además supone que las varianzas son exactamente iguales,  $\sigma_A = \sigma_B$ .

El modelo general representado por (5) requiere que la correlación  $\rho_{AB}$  y la desviación estándar  $\sigma_{AB}$  (o  $\sigma_A$  y  $\sigma_B$ ) sean estimadas. En su artículo original, Thurstone realizó una serie de simplificaciones del modelo para su manejabilidad. La simplificación más sencilla y popular es el modelo del Caso V, que asume que cada opción tiene la misma varianza y correlación cero (o menos restrictiva, igual correlación en lugar de correlaciones cero):

$$\sigma_A^2 = \sigma_B^2$$

$$\rho_{AB} = 0$$

Sin pérdida de generalidad, establece las desviaciones a la mitad  $\sigma_A^2 = \sigma_B^2 = \frac{1}{2}$  para que la varianza de  $A - B$  sea uno,

$$\sigma_{AB}^2 = \sigma_A^2 + \sigma_B^2 = 1$$

Esto establece la unidad de escala para la escala de intervalo (eliminando un grado de libertad) de modo que una diferencia de escala de calidad de 1 implica que la media de  $A - B$  es una desviación estándar de  $A - B$ . Esto simplifica la Ley de Thurstone dada en (7) para el Caso V a,

$$\hat{\mu}_{AB} = \Phi^{-1}\left(\frac{f_{B,A}}{f_{B,A}+f_{A,B}}\right) \quad (8)$$

### 2.1.5 Método de mínimos cuadrados de Thurstone-Mosteller

El modelo de Thurstone proporciona un método para estimar la diferencia de escala para cualquier par de opciones mediante la estimación de  $P(A > B)$  por la proporción empírica de personas que prefieren A sobre B. Sin embargo, al considerar más de dos opciones, este enfoque se rompe porque estos valores necesitan ser adaptados para que quepan en una escala unidimensional. En esta sección se detalla el enfoque para estimar los puntajes de calidad, dadas más de dos opciones para el modelo Thurstone.

Mosteller [23] ofrece una solución, aplicando el método de mínimos cuadrados, al mostrar que la mejor estimación que podemos hacer del valor escalar de un estímulo es el promedio de las distancias entre el estímulo de interés y el resto de estímulos; es decir, el promedio de las puntuaciones típicas correspondientes. Veamos un ejemplo con 3 estímulos A, B y C. El valor escalar de A lo obtendremos promediando las puntuaciones típicas  $z$  asociadas a la proporción de veces en que A ha sido juzgado mayor que B y la de veces en que ha sido juzgado mayor que C, es decir  $\hat{\mu}_{AB}$  y  $\hat{\mu}_{AC}$ . Del mismo modo, el valor escalar de B lo obtendremos promediando  $\hat{\mu}_{BA}$  y  $\hat{\mu}_{BC}$ , y el valor escalar de C lo obtendremos promediando  $\hat{\mu}_{CA}$  y  $\hat{\mu}_{CB}$ .

Para determinar las puntuaciones de calidad para un conjunto de opciones  $t$ , se define el vector de las puntuaciones de calidad  $\mu = [\mu_1, \mu_2, \dots, \mu_t]$  y la matriz  $D$  como una matriz  $t \times t$ , donde  $D_{ij} = \Phi^{-1}\left(\frac{f_{j,i}}{f_{j,i}+f_{i,j}}\right)$ , definida en (8), para la diferencia de calidad entre la opción  $i$  y la opción  $j$ . Se calcula entonces una posición en la escala psicológica para cada objeto  $j$  como,

$$\mu_j = \frac{1}{t} \sum_{i=1}^t D_{ij}$$

En donde se suman los valores de cada columna  $j$  de la matriz  $D$  y se dividen entre el número de objetos ( $t$ ).

Al tratarse de puntuaciones típicas, pueden aparecer valores negativos. Para evitarlos se suele hacer una transformación lineal de la escala. Se suma una constante a todos los estímulos  $\mu_t$  de manera que el valor del estímulo con el valor escalar más bajo pase a ser cero. Con ello se fija el origen de la escala y se obtiene la escala de intervalo buscada,

$$\tilde{\mu}_j = \mu_j - \min(\mu)$$

## 2.2 Métrica objetiva C2G-SSIM

Dado que los seres humanos son los receptores finales en la mayoría de las aplicaciones de procesamiento de imágenes, la manera más confiable de evaluar la calidad de una imagen es mediante la evaluación subjetiva, sin embargo es un método lento, costoso y poco práctico para los sistemas de procesamiento de imágenes en tiempo real. Por lo tanto, en los últimos años ha habido un mayor interés en desarrollar métricas objetivas de la IQA, y se ha realizado un gran esfuerzo para diseñar nuevos métodos objetivos de evaluación de la calidad que sean consistentes con las medidas de calidad perceptiva.

En la evaluación objetiva de la calidad, se utilizan algoritmos automáticos o ecuaciones matemáticas para la evaluación de la calidad que podría analizar imágenes y reportar su calidad sin la participación humana, dando medidas cuantitativas. Este método reduce el costo y agiliza la evaluación de la calidad. Según la disponibilidad de una imagen original, las métricas objetivas de calidad de imagen se clasifican en, referencia completa (*Full-Reference - FR*), sin referencia (*No-Reference - NR*) y referencia reducida (*Reduced-Reference - RR*). Donde, la métrica de referencia completa exige que una imagen de referencia completa sea conocida; sin referencia es cuando la imagen de referencia no está disponible; y, por último, referencia reducida indica que la imagen de referencia se conoce parcialmente en forma de un conjunto de características extraídas como información aparte que ayuda a la evaluación.

Los enfoques convencionales de FR como el error cuadrático medio (*mean squared error - MSE*) y el índice de similitud estructural (*structural similarity - SSIM*) no son aplicables en este escenario, ya que las imágenes de referencia y distorsionadas no tienen la misma dimensión (diferentes números de canales de color, siendo una a color y la otra en escala de grises). La aplicación de medidas de RR y NR también es conceptualmente inapropiada porque la imagen de origen está completamente disponible y contiene más información que la imagen de prueba.

A pesar del amplio uso de los algoritmos de transformación de una imagen en color en una escala de grises (*color to gray - C2G*), se ha dedicado poca investigación a comparar el rendimiento de estos algoritmos de conversión. Zhou, Zeng, et. al. [24], en el trabajo titulado “Objective Quality Assessment for Color-to-Gray Image Conversion”, desarrollan un modelo de calidad objetivo que predice automáticamente la calidad percibida de las imágenes convertidas en C2G; y cuya implementación se encuentra disponible para MATLAB<sup>1</sup>. Inspirados en la filosofía del índice SSIM, realizan la propuesta de un índice de similitud estructural C2G (C2G-SSIM), que evalúa las similitudes de luminancia, contraste y estructura entre la imagen de color de referencia y la imagen convertida C2G. Los tres componentes se combinan para obtener una medida de calidad general.

### 2.2.1 Índice C2G-SSIM

El diagrama propuesto del índice C2G-SSIM se muestra en la Figura 14. Primero, se transforma la imagen de color de referencia y la imagen de prueba C2G en un espacio de color. A continuación, se miden las distorsiones de la luminosidad, el contraste y la estructura para capturar los cambios de calidad percibidos, introducidos por la conversión C2G. Finalmente, se combinan las tres medidas anteriores en una medida de calidad general.

A continuación se explican brevemente cada una de las etapas que conforman el modelo del índice C2G-SSIM. La explicación en detalle no forma parte del alcance de este trabajo.

---

<sup>1</sup> <https://ece.uwaterloo.ca/~k29ma/>

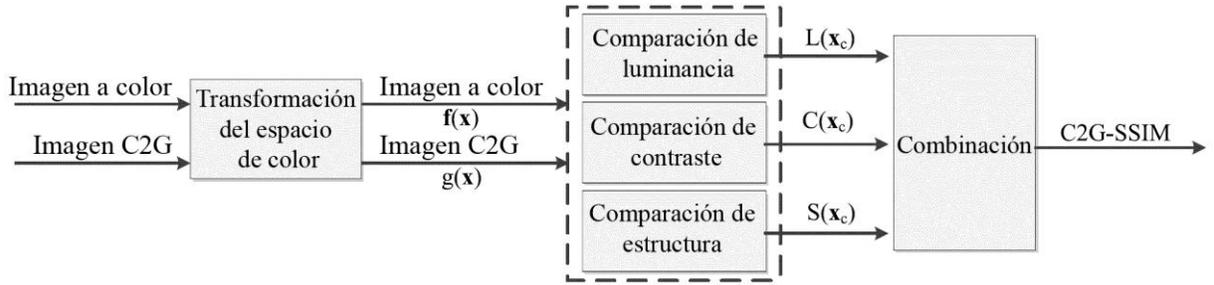


Figura 14: Estructura de tres etapas de C2G-SSIM. [24]

### 2.2.1.1 Transformación del espacio de color

Para capturar la pérdida de calidad percibida durante la conversión de C2G, se trabaja en un espacio de color de uniformidad perceptiva, CIELab, donde la distancia euclidiana entre dos puntos de color es proporcional a la diferencia de color percibida, denotada por  $\Delta E$ .

### 2.2.1.2 Medida de similitud

Sea  $x$  la coordenada espacial de la imagen y  $f(x)$  y  $g(x)$  denotan las imágenes a color y C2G, respectivamente. En cualquier ubicación espacial particular  $x$ ,  $f(x)$  es un vector de tres dimensiones y  $g(x)$  es un escalar. Se comienza con la evaluación de similitud de la imagen en cada ubicación espacial. Un enfoque útil para lograr esto es definir una función de proximidad geométrica centrada en cualquier ubicación espacial dada  $x_c$ . La función de proximidad se denota por  $p(x, x_c)$ .

Para comparar  $f(x)$  y  $g(x)$  en  $x_c$ , se combinan tres medidas de similitud distintas de luminancia, contraste y estructura. Específicamente, la medida de luminancia  $L(x_c)$  evalúa la consistencia de la luminancia local entre  $f(x)$  y  $g(x)$ ; la medida de contraste  $C(x_c)$  indica la similitud de contraste local entre  $f(x)$  y  $g(x)$ ; y la medida de la estructura  $S(x_c)$  evalúa la similitud de la estructura local entre  $f(x)$  y  $g(x)$ . Al combinar los tres componentes relativamente independientes, definimos la medida de calidad general en  $x_c$  como,

$$q(x_c) = \mathfrak{F}(L(x_c), C(x_c), S(x_c))$$

Donde  $\mathfrak{F}(\cdot)$  es una función de combinación que aumenta monótonamente con los tres componentes, de modo que cualquier pérdida de luminancia, contraste o estructura provoca una degradación de la calidad general. Los tres componentes de similitud se describen a continuación.

La medida de luminancia es definida como,

$$L(x_c) = \frac{2\mu_f(x_c)\mu_g(x_c) + C_1}{\mu_f(x_c)^2 + \mu_g(x_c)^2 + C_1}$$

Donde  $\mu_f(x_c)$  y  $\mu_g(x_c)$  son la luminancia media ponderada, calculados de los componentes de luminancia  $l_{f(x)}$  y  $l_{g(x)}$  extraídos de  $f(x)$  y  $g(x)$ , respectivamente, y  $C_1$  es una pequeña constante de estabilización positiva.

La medida de contraste es definida como,

$$C(x_c) = \frac{2d_f(x_c)d_g(x_c) + C_2}{d_f(x_c)^2 + d_g(x_c)^2 + C_2}$$

Donde,  $d_f(x_c)$  es la diferencia de color media ponderada de su entorno, que permiten evaluar el contraste de color local en la ubicación espacial  $x_c$ ,  $d_g(x_c)$  es la diferencia de tono gris media de la imagen C2G y  $C_2$  es una pequeña constante positiva para evitar la inestabilidad cuando el denominador está cerca de cero.

La medida de la estructura es definida como,

$$S(x_c) = \frac{\sigma_{fg}(x_c) + C_3}{\sigma_f(x_c)\sigma_g(x_c) + C_3}$$

Donde,  $C_3$  es una pequeña constante positiva,  $\sigma_f(x_c)$ ,  $\sigma_g(x_c)$  y  $\sigma_{fg}(x_c)$  son las desviaciones estándar de  $\phi(\|f(x) - f(x_c)\|)$ ,  $\phi(|g(x) - g(x_c)|)$  y correlación entre  $\phi(\|f(x) - f(x_c)\|)$  y  $\phi(|g(x) - g(x_c)|)$ , respectivamente.

Donde, por simplificación,  $\phi(\|f(x) - f(x_c)\|)$  y  $\phi(|g(x) - g(x_c)|)$  son una asignación no lineal de la distancia euclídea ( $\Delta E$ ).

Se establece  $C_1 = 10$ ,  $C_2 = 0.1$  y  $C_3 = 0.01$ . Empíricamente, se encuentra que el rendimiento general de C2G-SSIM es robusto a las variaciones de estos parámetros.

### 2.2.1.3 Medida de calidad general

La medida de luminancia  $L(x_c)$ , la medida de contraste  $C(x_c)$  y la medida de estructura  $S(x_c)$  describen tres aspectos diferentes de la calidad perceptiva de la imagen C2G.  $L(x_c)$  cuantifica la consistencia de la luminancia, mientras que  $C(x_c)$  y  $S(x_c)$  están más relacionados con la preservación de los detalles estructurales de la conversión de C2G.

Por lo tanto, el índice C2G-SSIM, que permite combinaciones flexibles de los tres componentes, se define:

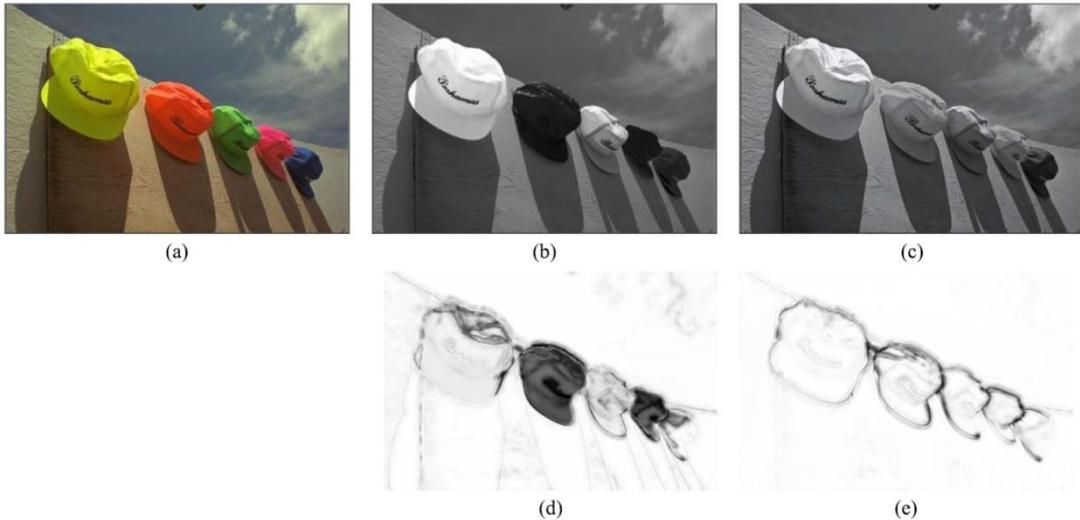
$$q(x_c) = L(x_c)^\alpha \cdot C(x_c)^\beta \cdot S(x_c)^\gamma$$

Donde  $\alpha > 0$ ,  $\beta > 0$  y  $\gamma > 0$ , son parámetros de control definidos por el usuario para ajustar la importancia relativa de los tres componentes.

La comparación local se aplica mediante una ventana deslizante en toda la imagen, lo que da como resultado un mapa de calidad que indica cómo se conservan la coherencia de la luminancia, el contraste y el detalle estructural en cada ubicación espacial.

En la Figura 15 se muestra un ejemplo visual, donde el brillo indica la magnitud del valor C2G-SSIM local. Como se puede ver, los mapas de calidad reflejan las variaciones espaciales de la calidad de la imagen percibida de diferentes imágenes C2G. Específicamente, la imagen C2G en la Figura 15 (c) muestra una mejor consistencia de luminancia que la imagen C2G en la Figura 15 (b), donde la luminancia de los sombreros está muy alterada. Además, se otorga una penalización aún mayor (marcada

como píxeles negros en el mapa C2G-SSIM) a las regiones de letras en las partes delanteras del segundo y cuarto sombrero, donde los detalles estructurales desaparecen.

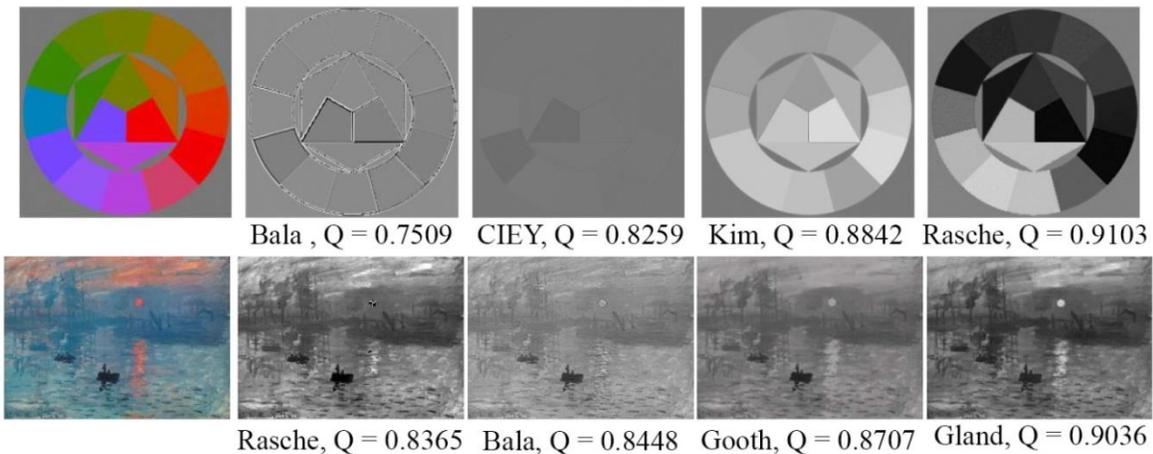


**Figura 15: Imágenes C2G y sus mapas de índice C2G-SSIM. (a) es la imagen a color de referencia. (b) y (c) son las imágenes C2G creadas por distintos métodos de decoloración. (d) y (e) son los correspondientes mapas C2G-SSIM de (b) y (c), respectivamente. En todos los mapas de índice C2G-SSIM, más brillante indica mejor calidad. [24]**

En la práctica, se desea una puntuación única para la calidad general de toda la imagen. Se puede obtener una única puntuación C2G-SSIM tomando el promedio del mapa C2G-SSIM:

$$Q(f, g) = \frac{\int q(x_c) dx_c}{\int dx_c}$$

Como el máximo de  $q(x_c)$  es 1,  $Q$  también está delimitado por 1. Un ejemplo de este procedimiento se muestra en la Figura 16, en la cual las imágenes se presentan en orden ascendente de izquierda a derecha en términos de C2G-SSIM.



**Figura 16: Demostración visual de C2G-SSIM. CIEY, Bala [25], Rasche [26], Gooth [27], Kim [28] y Gland [29], son los algoritmos de decoloración aplicados a la imagen original y  $Q$  es el índice C2G-SSIM. [24]**

## 2.3 Coeficientes de correlación

Dentro del ámbito específico de las pruebas de ranking, para medir el grado de correlación (la asociación o interdependencia) entre dos valoraciones dadas, es decir, entre la evaluación de calidad subjetiva y objetiva, se aplican métricas de rendimiento objetivo, como son el coeficiente de correlación de Spearman (*Spearman rank correlation coefficient* - SRCC) y el coeficiente de correlación de Kendall (*Kendall rank correlation coefficient* - KRCC), donde ambos cuantifican la relación entre dos descriptores. Dichos algoritmos son presentados en detalle en el libro de Legendre P. y L. Legendre [30].

### 2.3.1 SRCC

El coeficiente de correlación de Spearman, también conocido como  $\rho$  (rho), es utilizado cuando ambas variables están en el tipo de datos ordinales (de ranking). Siendo  $X$  y  $Y$  dos variables, ambas de tamaño  $t$ , entonces para determinar el coeficiente  $\rho$ , los  $t$  puntajes crudos  $X_i$  y  $Y_i$  se convierten en su rango  $x_i$  y  $y_i$ . Entonces, la ecuación de  $\rho$  se calcula como,

$$\rho = 1 - \frac{6 \sum_{i=1}^t d_i^2}{t^3 - t}$$

Donde,  $d_i = x_i - y_i$  denota la diferencia entre rangos. El coeficiente de correlación de Spearman varía entre +1 y -1. Los descriptores que están perfectamente emparejados, en términos de rangos, muestran valores de  $\rho = +1$  (relación directa) o  $\rho = -1$  (relación inversa), mientras que  $\rho = 0$  indica la ausencia de una asociación entre los descriptores.

Si dos o más objetos llegan a tener el mismo orden en un descriptor dado, lo que se conoce como un empate, a cada uno de ellos se les asigna el promedio de los rangos que se hubieran fijado si no hubiera empates. Si la proporción de empates es alta se deberán aplicar factores de corrección, no considerados en este trabajo.

Tomando como referencia el ejemplo numérico de la Tabla 1, se calcula,

$$\rho = 1 - \frac{6 \times ((3 - 3)^2 + (4 - 1)^2 + (2 - 4)^2 + (1 - 2)^2)}{4^3 - 4} = -0.4$$

Objetos (unidades de observación)	Rangos de objetos en los dos descriptores	
	$x$	$y$
$O_1$	3	3
$O_2$	4	1
$O_3$	2	4
$O_4$	1	2

Tabla 1: Clasificación de cuatro objetos en dos descriptores,  $x$  y  $y$ . [30]

Para este trabajo se calcula el coeficiente SRCC mediante la función CORREL( ) facilitada por Microsoft Excel<sup>2</sup>.

<sup>2</sup> <https://support.office.com/en-us/article/correl-function-995dcef7-0c0a-4bed-a3fb-239d7b68ca92>

### 2.3.2 KRCC

El coeficiente de correlación de Kendall, también conocido como  $\tau$  (tau). Al igual que Spearman, tiene como objetivo evaluar la asociación entre dos variables ordinales.

Sea  $(x_i, y_i)$  un conjunto de observaciones de las variables aleatorias conjuntas  $X$  y  $Y$  respectivamente, de manera que todos los valores de  $(x_i)$  y  $(y_i)$  sean únicos. Cualquier par de rangos  $(x_i, y_i)$  y  $(x_j, y_j)$  se supone que son concordantes si los rangos de ambos elementos son  $x_i > x_j$  y  $y_i > y_j$  o ambos son  $x_i < x_j$  y  $y_i < y_j$ . Se supone que son discordantes, si  $x_i > x_j$  y  $y_i < y_j$  o si  $x_i < x_j$  y  $y_i > y_j$ . El par no es ni concordante ni discordante, si  $x_i = x_j$  o  $y_i = y_j$ . La ecuación para el coeficiente de Kendall,  $\tau$  se da como,

$$\tau = \frac{\sum_{i=1}^t \sum_{j=1}^t \text{sgn}(x_i - x_j) \text{sgn}(y_i - y_j)}{t(t-1)/2}$$

Donde,  $\text{sgn}(\cdot)$  es la función de Signo de su argumento.

Tomando como referencia el ejemplo numérico de la Tabla 1, se calcula,

$$\tau = \frac{(-1) + (-1) + (+1) + (-1) + (-1) + (+1)}{6} = \frac{-4 + 2}{6} = -0.33$$

En el caso de un acuerdo perfecto entre dos descriptores, todos los pares reciben una puntuación positiva, por lo tanto  $\tau = +1$ . Cuando hay un desacuerdo completo  $\tau = -1$ . Cuando los descriptores no están relacionados, las puntuaciones positivas y negativas se cancelan, por lo que  $\tau$  resulta cercano a 0.

Otra forma simplificada para calcular el coeficiente KRCC se puede consultar en el capítulo 5 del libro “*Numerical ecology*” [30].

## **3 PLANTEAMIENTO DEL PROBLEMA**

Basándose en los conocimientos teóricos expuestos a lo largo del documento, que abarcan desde el proceso de decoloración de imágenes hasta las técnicas para evaluar los resultados obtenidos, se presentan una serie de puntos que definen la propuesta a realizar en este Trabajo Especial de Grado.

### **3.1 Planteamiento del problema**

Las imágenes digitales tienen una participación diaria en la sociedad a través de las diferentes tecnologías modernas en dispositivos electrónicos de consumo actuales, pero a pesar de esto existen dispositivos monocromáticos que todavía juegan papeles importantes. Por ejemplo, dispositivos de impresión monocromáticos, que se utilizan para exportar las imágenes en color a escala de grises, con diversos propósitos, tales como la impresión de forma económica de libros de texto, catálogos, periódicos o revistas. Aparte, las imágenes en escala de grises se utilizan por otras razones, como para su reproducción en dispositivos monocromáticos (como dispositivos médicos), procesamiento posterior (como detección de bordes), para fines estéticos (como es el caso de la fotografía a color con la transformación a blanco y negro), entre otros.

Las conversiones de color a escala de grises realizan una reducción de los datos de color tridimensionales a una sola dimensión, haciéndose evidente que cierta pérdida de información durante la conversión es inevitable. Dado que los enfoques clásicos de conversión de color a escala de grises no buscan preservar los aspectos de contraste y luminancia e ignoran el análisis de la información general de color, recientemente, se han propuesto varias aproximaciones a este problema de conversión, centrándose principalmente en preservar magnitudes de los colores y así reducir la pérdida de información. Éstos son algoritmos complejos, cuyo objetivo es almacenar tanta información de la imagen de color original como sea posible y, al mismo tiempo, producir resultados perceptiblemente aceptables en escala de grises.

La automatización del proceso de decoloración es compleja, debido a que aún no existe un algoritmo que arroje resultados visualmente acertados para todas las imágenes a color dadas, dado que cada imagen es única en sus características generales, como color, brillo, contrastes, etc.

### **3.2 Justificación**

Actualmente las impresiones de libros, folletos y demás material con apoyo visual de imágenes que deben mantener alta calidad de representación, como las reproducciones de pinturas, son generadas con altos costos debido a la necesidad de su impresión a color para preservar sus detalles. Los elevados costos se pueden disminuir con la impresión en escala de grises, pero para esto es necesario procesar las imágenes a esta escala, manteniendo los detalles necesarios, de forma que pueda obtenerse una impresión monocromática de calidad.

Existen diversos enfoques de algoritmos complejos de conversión de color a escala de grises, al alcance de cualquier persona y de libre distribución, que tienen como objetivo principal la obtención de una imagen final con una transformación óptima, que busca obtener “el mejor” resultado de representación en escala de grises, dada una imagen a color. Estos algoritmos, a diferencia de las transformaciones clásicas, no son sencillos de implementar, y la automatización para el cálculo del mejor resultado no siempre está disponible, ni son populares entre las aplicaciones libres de tratamiento de imágenes, por lo que su alcance en la sociedad se ha visto limitado.

Mediante el desarrollo de este trabajo se pretende reducir los costos de impresión, obtener la mejor calidad perceptual posible de los resultados al transformar la imagen de color a escala de grises, permitir el control de parámetros (tanto de forma automatizada como manual) para obtener la imagen en escala de grises deseada y alcanzar a la población a través de una aplicación libre y con código abierto.

### **3.3 Objetivos**

Dentro de los objetivos que se cumplen para alcanzar con satisfacción la propuesta planteada en el punto anterior, se encuentran los descritos a continuación.

#### **3.3.1 Objetivo general**

Desarrollar una implementación de la transformación de imágenes de color a escala de grises maximizando la percepción visual.

#### **3.3.2 Objetivos específicos**

- Diseñar una solución basada en una selección específica de los algoritmos del estado del arte, que permita una ejecución automatizada de sus parámetros de control sobre la imagen.
- Diseñar e implementar una interfaz de usuario (GUI) que permita principalmente, dentro de sus interacciones, la selección de la imagen a transformar, el control de parámetros y la ejecución del algoritmo implementado.
- Seleccionar la estrategia de evaluación de calidad de los resultados del algoritmo implementado, a través de las metodologías conocidas como las métricas subjetivas y objetivas.
- Generar un conjunto de datos de prueba, que contenga los parámetros específicos y sus variaciones, para poder aplicar las métricas de calidad.
- Elaborar las encuestas subjetivas de las imágenes a evaluar.
- Calcular las métricas de evaluación de calidad de las imágenes obtenidas a través de las metodologías seleccionadas.

### **3.4 Metodología**

Las etapas de desarrollo del proyecto han sido llevadas a cabo siguiendo el modelo de desarrollo en cascada, por la modularidad y la dirección del flujo de dependencia entre las partes que lo componen. El esquema del flujo de la metodología a seguir se puede observar en la Figura 17. A continuación se detallan las partes que lo componen.

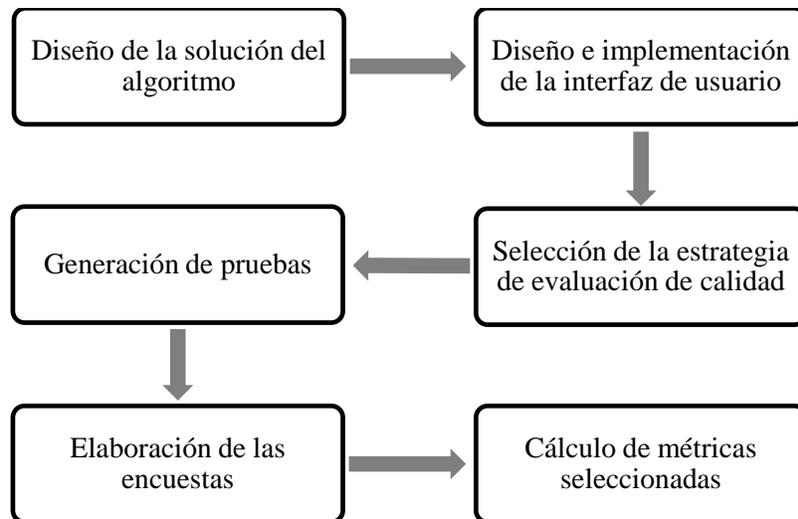


Figura 17: Metodología de desarrollo.

### 3.4.1 Diseño e implementación

Luego de estudiar los distintos algoritmos del estado del arte se optó por diseñar una solución basándose en el algoritmo propuesto por Zhao y Tamimi [13], quienes proponen el uso de la transformada de Fourier, sobre el espacio CIELab de la imagen a color, y dos parámetros de control para modificar la transformación en escala de grises. Más adelante se detallan las características del diseño de software.

El software utilizado para implementar el algoritmo propuesto es Visual Studio 2017, debido a que el ambiente de desarrollo se encuentra en Windows 7. El lenguaje de programación utilizado es C++, y se utilizan bibliotecas que facilitan las estructuras necesarias para la implementación del algoritmo; en este caso GNU OpenCV versión 2.4.13.

Para el diseño de la interfaz de usuario se deben considerar todas las interacciones que el algoritmo requiera con el usuario para permitir el control de parámetros previos a la transformación definitiva de la imagen; esto conlleva a la decisión de diseñar una interfaz interactiva con resultados en tiempo real y un control de parámetros con estructuras sencillas de comprender y utilizar. Para un diseño más específico se estudian las estructuras de diseño disponibles en la biblioteca Qt.

La biblioteca Qt versión 5 permite el acceso a estructuras de despliegue de gráficos que se adaptan a las necesidades propuestas en el diseño. Para la manipulación del diseño se utiliza el IDE (*Integrated development environment* – entorno de desarrollo integrado) Qt Creator versión 4.5. Debido a que está implementado en C++ esto permite una interacción sin inconvenientes con la implementación del algoritmo propuesto. Finalmente, el algoritmo implementado es cargado en la plataforma online, de libre acceso, GitHub, a través de la dirección: [https://github.com/irenita/Espectral\\_II](https://github.com/irenita/Espectral_II).

### 3.4.2 Selección de estrategia de evaluación de calidad

Para evaluar la calidad los resultados del algoritmo implementado se consideran dos metodologías, la subjetiva y la objetiva. Por lo tanto en esta etapa se deben de diseñar dos estrategias. En relación a la

evaluación subjetiva, donde los sujetos de prueba se someten a la visualización y análisis de las imágenes, se sigue la metodología de las combinaciones por pares (2AFC), lo que conlleva a la elaboración de encuestas. Y, en relación a la evaluación objetiva, donde un algoritmo analiza la calidad, se sigue la métrica C2G-SSIM.

Para la evaluación de los resultados se debe determinar el grupo de imágenes a color a utilizar, las cuales se obtienen de páginas web que permiten la descarga y uso de imágenes en distintas resoluciones, de forma libre para fines no remunerados; estas imágenes son almacenadas de forma local.

### **3.4.3 Generación de pruebas**

El cálculo de las métricas de evaluación de calidad necesita disponer de los resultados del algoritmo propuesto, por lo que se deben de realizar todas las ejecuciones del mismo para obtener cada una de las imágenes que se desean someter a prueba. Se deben de considerar todas las variaciones de parámetros de control que se quieran evaluar del algoritmo.

### **3.4.4 Elaboración de las encuestas**

Las encuestas presentadas a los sujetos de prueba, como métrica subjetiva y siguiendo la metodología 2AFC, contienen combinaciones por pares de las imágenes a evaluar. Cada una de las presentaciones de combinaciones por pares se elabora con la ayuda del software de edición de imágenes, Adobe Photoshop CC. El control de los resultados de la encuesta se lleva a cabo a través de una hoja impresa, que contiene todas las combinaciones por pares presentadas al sujeto de prueba y permite el registro de su selección.

### **3.4.5 Cálculo de las métricas seleccionadas**

En el caso de la métrica objetiva, C2G-SSIM, su cálculo se hace mediante el software MATLAB. Las combinaciones por pares evaluadas con esta métrica deben ser las mismas que las presentadas en la métrica subjetiva, para posteriormente comparar el resultado de ambas, a través de las métricas de correlación SRCC y KRCC. En el caso del cálculo de los resultados de las encuestas, se realiza aplicando la Ley del Juicio Comparativo de Thurstone Caso V. Tanto la escala de Thurstone como el cálculo de los coeficientes de correlación se llevan a cabo a través de hojas de cálculo en Microsoft Excel.

Los resultados obtenidos de las métricas de evaluación de calidad se organizan a través de tablas y gráficos que permiten una visualización intuitiva de los resultados arrojados. El análisis de éstos permite determinar si el algoritmo propuesto cumple las expectativas de calidad perceptual propuestas y permite obtener conclusiones sobre cada grupo de imágenes a color evaluadas con sus respectivas transformaciones en escala de grises, y por lo tanto de los distintos parámetros utilizados.

## **3.5 Esquema del proyecto**

Este proyecto se divide en dos tareas fundamentales: la relacionada al algoritmo de transformación y la relacionada a las métricas de evaluación de calidad de los resultados que arroja el algoritmo propuesto; detalladas en los próximos capítulos.

En la Figura 18 se presenta el esquema general de la interacción del usuario final con el algoritmo implementado, a través de la interfaz gráfica propuesta. Dentro de las interacciones permitidas están: la selección de la imagen a transformar de color a escala de grises, el establecimiento de los parámetros para

la transformación y la ejecución del algoritmo implementado (con visualización de resultados en tiempo real).

En la Figura 19 se presenta el esquema general de la evaluación de calidad de las imágenes, donde un grupo de sujetos de prueba evalúan los resultados que arroja el algoritmo, posteriormente se escalan a una dimensión dichos resultados para analizar y establecer conclusiones sobre el algoritmo implementado.

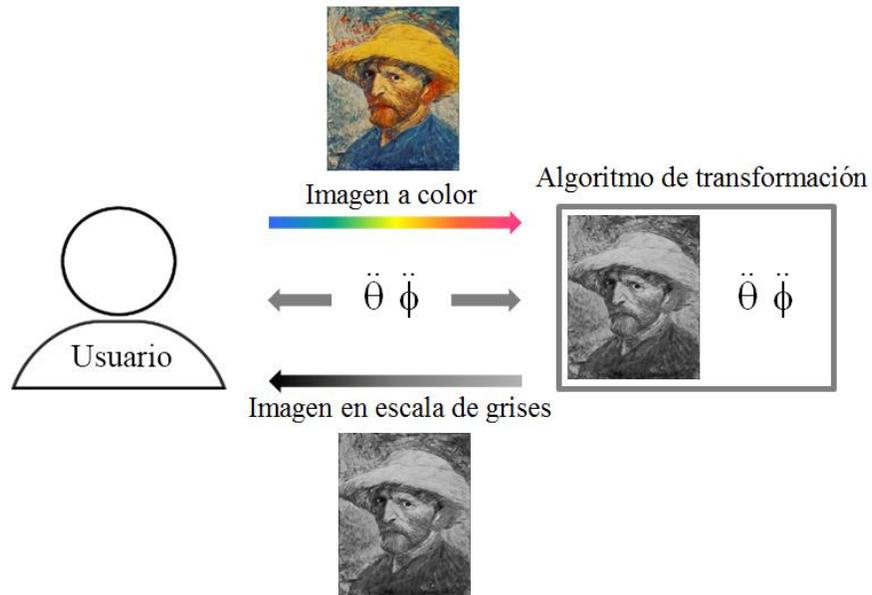


Figura 18: Esquema general de la interacción de la aplicación con el usuario.

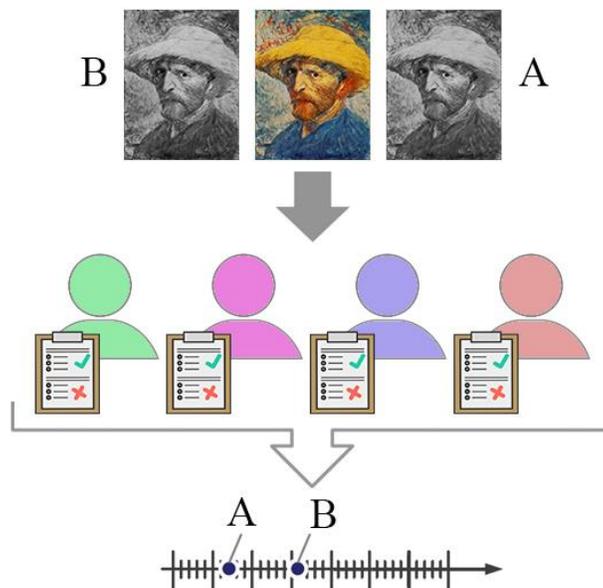


Figura 19: Esquema general de la evaluación de calidad.

## **3.6 Ambiente de implementación y diseño de pruebas**

La implementación del algoritmo, el diseño de las pruebas y el cálculo de las métricas de evaluación, se llevan a cabo siguiendo las siguientes características de hardware y software.

### **3.6.1 Hardware**

Computador portátil DELL Inspiron 5558 con las siguientes características: Sistema operativo Windows 7 de 64 bits, Procesador Intel Core i7 de 2.40GHz, Memoria RAM de 8 GB, Tarjeta de video NVidia GeForce 920M, Pantalla de 15.6" LCD, Disco duro de 1TB.

### **3.6.2 Software**

Sistema operativo Windows 7, IDE Microsoft Visual Studio 2017, Biblioteca OpenCV 2.4.13.4, Biblioteca Qt versión 5.10.0, IDE Qt Creator 4.5, MATLAB 9.0, Photoshop CC.

## 4 DISEÑO DE LA SOLUCIÓN

El diseño de este proyecto se divide en dos partes fundamentales: el diseño del algoritmo de transformación de imágenes de color a escala de grises y el diseño de la interfaz gráfica de interacción con el usuario, para la ejecución, manipulación y visualización del algoritmo diseñado.

### 4.1 Algoritmo de transformación

El algoritmo propuesto en este proyecto para transformar una imagen de color a escala de grises está basado en la teoría planteada en el trabajo de Ye Zhao y Zakiya Tamimi, titulado “Decoloración espectral” (sección 1.3.5). Se opta por mantener la estructura básica de transformación propuesta en dicho trabajo, donde se introduce el contraste cromático en una imagen en escala de grises mejorada,  $\tilde{G}_I$ , en el dominio de Fourier, calculada como,

$$\tilde{G}_I = IF(\tilde{E})$$

Donde,  $I$  representa la imagen a color original,  $IF()$  representa la inversa de la transformada de Fourier y  $\tilde{E}$  representa la escala de grises modificada en el dominio de Fourier, dada por,

$$\tilde{E} = \tilde{H}(\hat{L}, \hat{a}, \hat{b})$$

Donde la función  $\tilde{H}$  calcula una intensidad de la escala de grises modificada en el dominio de Fourier,  $\tilde{E}$ , a partir de las contrapartes del dominio de Fourier del canal de luminancia original y los dos canales cromáticos.  $\tilde{H}$  se implementa en cada frecuencia como:

$$\tilde{H}(\hat{L}, \hat{a}, \hat{b}) = (1 - \tilde{\theta})\hat{L} + \tilde{\theta}(\tilde{\phi}\hat{a} + (1 - \tilde{\phi})\hat{b}) + \beta\hat{L}$$

Donde  $\hat{L}$ ,  $\hat{a}$  y  $\hat{b}$  son las tres imágenes espectrales, con valores complejos, obtenidas a través de la transformada de Fourier de cada canal  $L$ ,  $a$  y  $b$ , respectivamente, del espacio CIELab de la imagen  $I$ . El coeficiente  $\tilde{\theta}$  controla el grado del contraste cromático incorporado al resultado en escala de grises,  $\tilde{\phi}$  es el coeficiente para determinar las contribuciones relativas de los canales  $a$  y  $b$  y  $\beta$  controla la proporción de luminosidad  $\hat{L}$  que se incorpora sobre el resultado en escala de grises.

El cálculo de los coeficientes  $\tilde{\theta}$  y  $\tilde{\phi}$  son alterados para este trabajo y se definen en la siguiente sección.

#### 4.1.1 Control de parámetros $\tilde{\theta}$ y $\tilde{\phi}$

Los coeficientes controlables,  $\tilde{\theta}$  y  $\tilde{\phi}$ , que determinan diversos efectos de aumento de contraste en los resultados en escala de grises, pueden calcularse automáticamente basándose en el hecho de los datos de los espectros de Fourier.

$\tilde{\theta}$  modela el grado del contraste cromático incorporado, que puede ser determinado vinculándolo con la pérdida relativa de conversión, medida por la comparación de la diferencia ponderada de RGB y la diferencia de luminancia. En este esquema, estas diferencias son modeladas por la operación del espectro en cada frecuencia:

$$\ddot{\theta} = 1 - \frac{|\hat{L}|}{|\hat{R}| * 0.299 + |\hat{G}| * 0.587 + |\hat{B}| * 0.114}$$

Donde  $|\cdot|$  representa el espectro de los valores complejos.  $\hat{R}$ ,  $\hat{G}$  y  $\hat{B}$  son los resultados de la transformada de Fourier de los canales R, G, B.

$\ddot{\phi}$  es calculada por la diferencia entre las proporciones relativas del espectro a y b:

$$\ddot{\phi} = \frac{|\hat{a}| - |\hat{b}|}{|\hat{a}| + |\hat{b}|}$$

Los coeficientes, tanto  $\ddot{\theta}$  como  $\ddot{\phi}$ , pueden ser calculados de forma automática en cada frecuencia ( $w$ ) a través de las ecuaciones presentadas; pueden ser promediadas desde todas sus frecuencias; o, también, pueden ser establecidas de forma manual. Estas tres opciones se consiguen controlar a través de una GUI que permita al usuario establecer la opción que satisfaga sus necesidades, permitiendo algunas permutaciones entre ellas.

## 4.2 Diseño de la GUI

La interfaz gráfica propuesta permite una interacción en tiempo real entre el algoritmo de transformación presentado en la sección anterior y el usuario final. En la Figura 20 se puede observar la vista de la interfaz en su primera interacción: con una imagen a color seleccionada por el usuario para su posterior transformación.

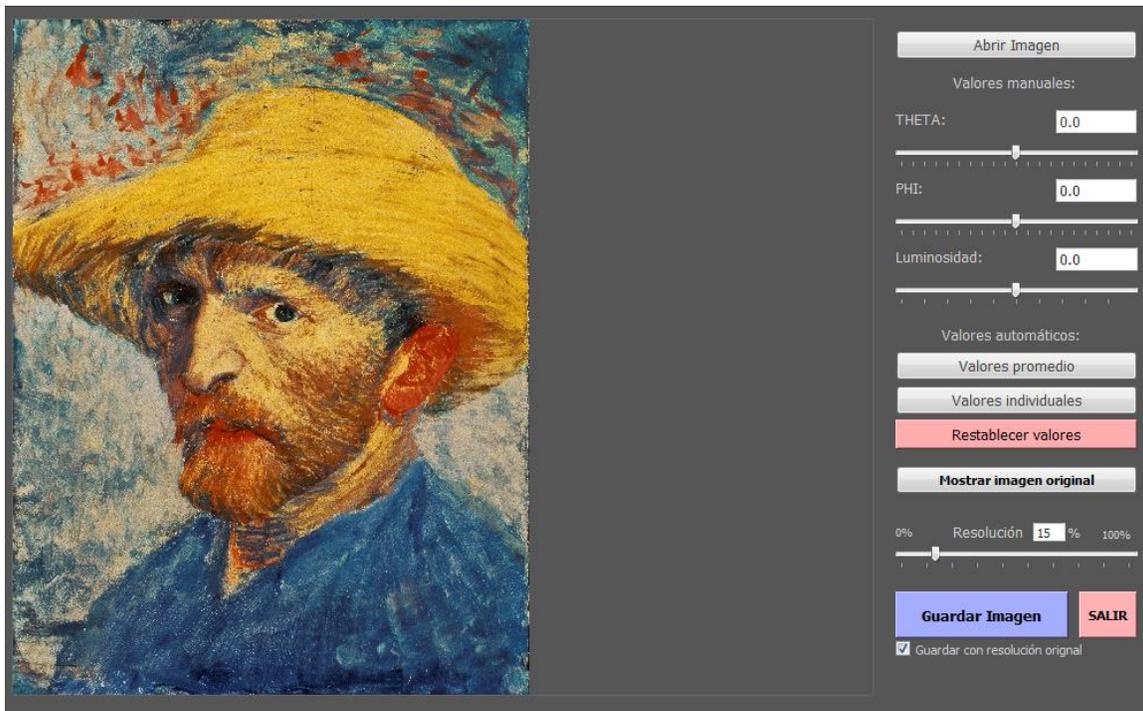


Figura 20: GUI propuesta, con el despliegue de la imagen a color seleccionada a transformar.

Las opciones que se presentan en la GUI se detallan a continuación, por orden de aparición de arriba hacia abajo.

#### 4.2.1 Abrir imagen

Permite seleccionar la imagen a transformar, ubicada dentro del directorio local. Los formatos permitidos son JPG, PNG o BMP. Se permite procesar una imagen a la vez, y no es posible procesar imágenes por lotes.

#### 4.2.2 Despliegue de la imagen

Una vez seleccionada la imagen a transformar, ésta se muestra en la sección destinada de la interfaz. El tamaño de presentación de la imagen es escalado a conveniencia de la ventana desplegada; en caso de tener mayor resolución, se redimensiona previamente a su carga en la aplicación, sin modificar el tamaño real de la imagen para la transformación final, más sí para su visualización temporal. La resolución máxima permitida no tiene restricciones, aunque los tiempos de ejecución dependen del hardware en donde se ejecute la aplicación; por ejemplo, en la configuración probada una imagen con un máximo de resolución de 840x840 píxeles arroja una respuesta en 0,6 segundos en promedio.

#### 4.2.3 Valores manuales

Dentro de los campos modificables manualmente se encuentran los coeficientes  $\check{\theta}$  y  $\check{\phi}$ , y además se activa un campo de luminosidad que se manipula con el coeficiente  $\beta$ . Estos valores manuales pueden modificarse a través de la casilla de edición correspondiente sin límites de rango, o a través de los deslizadores, cuyos límites son  $[-3, +3]$  para  $\check{\theta}$ ,  $[-5, +5]$  para  $\check{\phi}$  y  $[-1, +1]$  para  $\beta$ . Al editar cualquiera de estos valores su efecto sobre la imagen se ve reflejado en el espacio de despliegue de la imagen. En el caso de haber seleccionado con anterioridad alguna opción automatizada, este coeficiente se modifica a su opción manual.

#### 4.2.4 Valores automáticos

Los coeficientes  $\check{\theta}$  y  $\check{\phi}$  pueden ser calculados de forma automatizada a través de dos opciones, como se detalla en la sección 4.2.1, de forma individual (para cada frecuencia) o promediada. Al seleccionar cualquiera de estas opciones su efecto sobre la imagen se ve reflejado en el espacio de despliegue de la imagen. En el caso de haber seleccionado con anterioridad alguna opción de forma manual, este coeficiente se modifica a su opción automática.

#### 4.2.5 Restablecer valores

La opción de retornar la imagen a un estado inicial, sin intervención de los coeficientes  $\check{\theta}$ ,  $\check{\phi}$  y  $\beta$ , se logra mediante esta opción, en la que todos los coeficientes descritos se colocan en 0; de esta forma la imagen resultante es valor de luminancia L del espacio CIELab de la imagen dada.

#### **4.2.6 Mostrar imagen original**

Esta opción permite, mientras se mantenga presionado el botón, visualizar la imagen a color original, con la finalidad de facilitarle al usuario una referencia temporal de las modificaciones realizadas a la imagen a través de los distintos coeficientes.

#### **4.2.7 Resolución**

Dado que en algunos casos se desea transformar una imagen a color en alta resolución, se facilita la opción de modificar la resolución de la imagen de entrada, para permitir una visualización y ejecución del algoritmo en tiempo real. Por defecto se establece que cualquier imagen cargada por primera vez tenga una resolución del 15% de su tamaño original. Cabe señalar que al modificar la resolución de la imagen, el valor de los coeficientes puede verse modificado.

#### **4.2.8 Guardar imagen y salir**

Al finalizar el conjunto de selecciones pertinentes para la transformación de la imagen de color a escala de grises, el usuario puede almacenar el resultado final en un disco local. Antes de seleccionar la opción de almacenar el resultado, se permite “Guardar con resolución original”, en caso de haber cambiado la resolución a una más baja para su manipulación en tiempo real y desear almacenarla con la resolución de origen. Los formatos de almacenamiento permitidos son JPG, PNG o BMP. Por defecto los valores de los coeficientes finales son anexados junto al nombre del archivo definitivo. Al culminar el proceso de almacenamiento se muestra un texto, debajo del área de despliegue de la imagen, indicando que la imagen ha sido guardada.

Al momento de desear salir de la aplicación se dispone de la opción de forma legible y directa, con una ventana que despliega la confirmación de salida.

## 5 IMPLEMENTACIÓN

La ejecución de toda la aplicación propuesta utiliza una serie de entornos de desarrollo integrado (*integrated development environment* - IDE), bibliotecas y clases específicas, las cuales se detallan a continuación; abarcando desde el algoritmo de transformación hasta la interfaz de usuario. En la Figura 21 se puede observar un diagrama de flujo de la aplicación de forma general, es decir, del algoritmo y su interacción con el usuario.

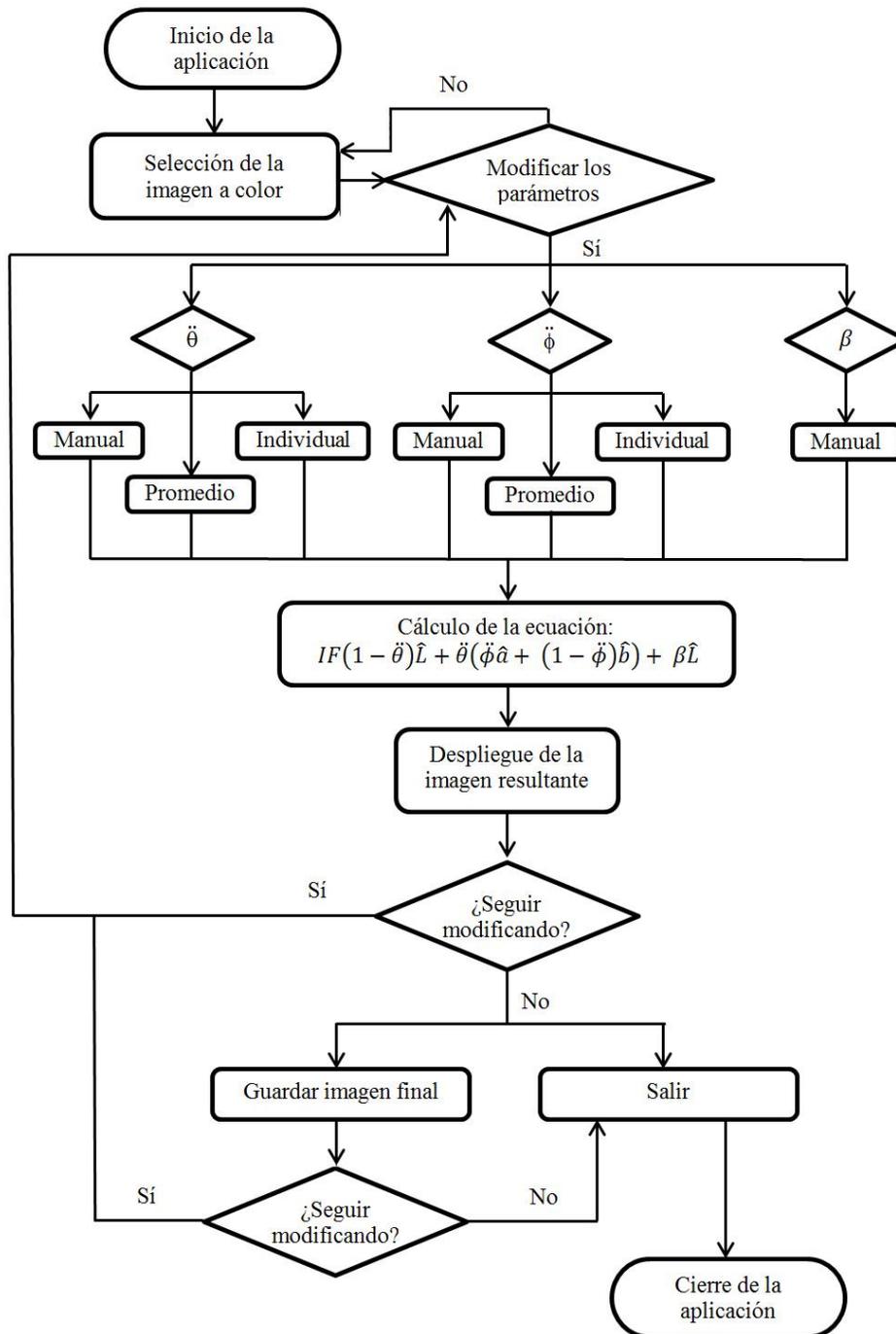


Figura 21: Diagrama de flujo de interacción entre el usuario y el algoritmo de transformación propuesto.

## 5.1 Algoritmo de transformación

En la Figura 22 se puede observar el diagrama de flujo del algoritmo de transformación de imágenes de color a escala de grises implementado, sin detallar las interacciones de éste con el usuario, en donde  $FFT()$  es la transformada de Fourier,  $||$  representa la magnitud de los valores complejos, donde  $\hat{R}$ ,  $\hat{G}$ ,  $\hat{B}$ ,  $\hat{L}$ ,  $\hat{a}$  y  $\hat{b}$  son los resultados de la transformada de Fourier en cada canal e  $IF()$  es la transformada inversa de Fourier.

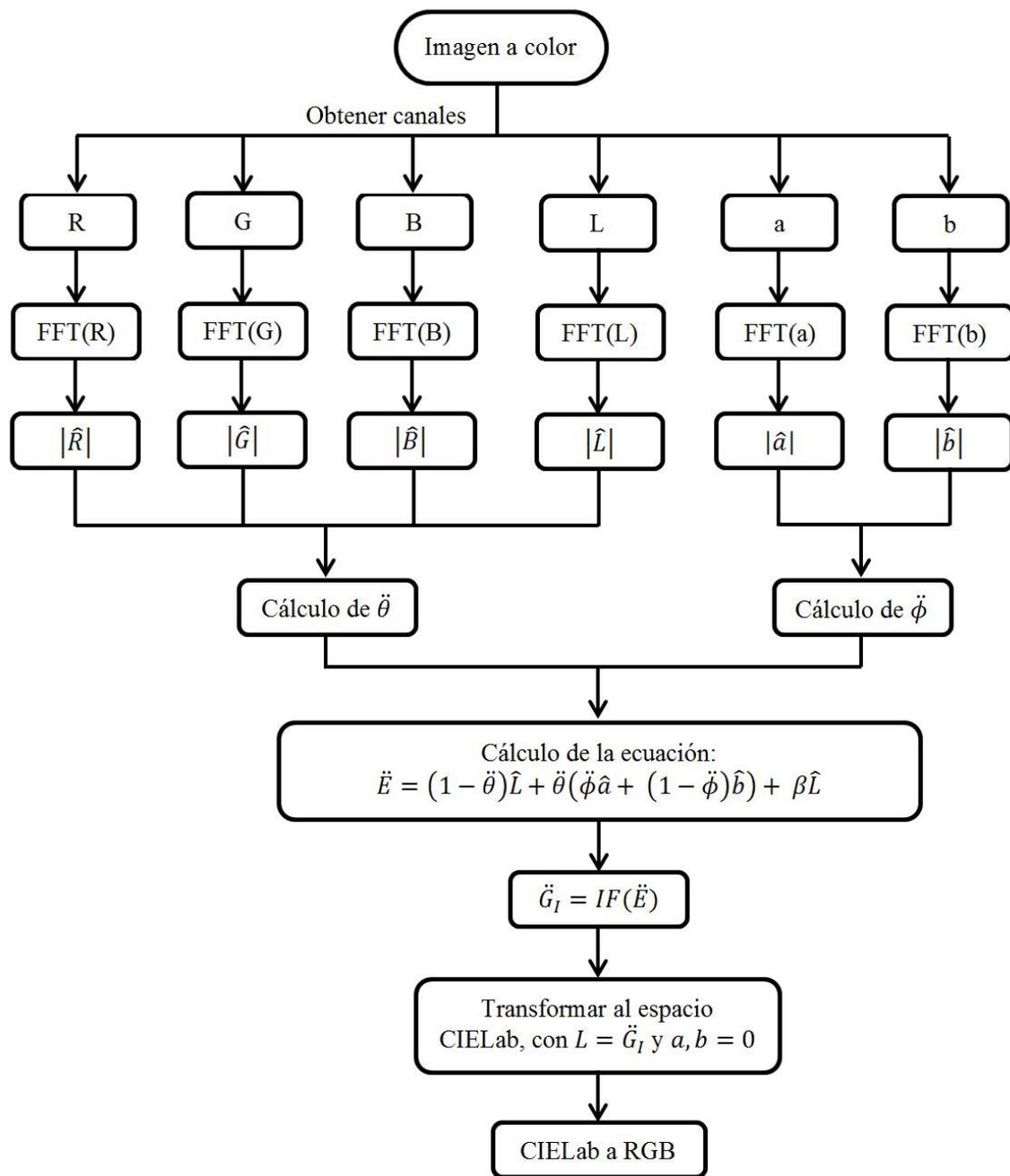


Figura 22: Diagrama de flujo del algoritmo de transformación de una imagen de color a escala de grises implementado.

### 5.1.1 Programas para el desarrollo

La implementación del algoritmo descrito que se llevó a cabo en el lenguaje C++, utilizando el IDE Microsoft Visual Studio 2017, versión 15.6.0. Este entorno de desarrollo permite editar, depurar y compilar código y, después, publicar una aplicación. Además incluye herramientas de finalización de código, diseñadores gráficos y muchas más características para facilitar el proceso de desarrollo de software, como la extensión de GitHub que permite una interacción con este repositorio.

### 5.1.2 Biblioteca

Dada la complejidad del algoritmo se necesita la inclusión de la biblioteca OpenCV (*Open source computer vision library*), que es una biblioteca de visión por computador de código abierto, multiplataforma y escrita en los lenguajes C y C++, diseñada para ser eficiente en cuanto al uso de recursos computacionales y con un enfoque hacia las aplicaciones en tiempo real. Uno de los objetivos de OpenCV es proveer una infraestructura de visión por computador fácil de utilizar que ayude a los programadores a desarrollar aplicaciones “sofisticadas” de CV (*Computer vision*). Es un producto con licencia BSD (permite el uso del código fuente en software no libre).

La biblioteca cuenta con más de 2500 algoritmos optimizados, que incluyen un conjunto de completo de algoritmos de visión por computador tanto clásicos como del estado del arte.

Siguiendo el flujo de la transformación propuesta (ver Figura 22), se especifican las funciones utilizadas de la biblioteca OpenCV.

#### 5.1.2.1 Imagen a color

Para la lectura y almacenamiento de la imagen a color indicada por el usuario, a través de una dirección del directorio local, se utiliza la función indicada en la Figura 23.

```
671         imagenCargada.imagen = imread(imagenCargada.nombreArchivo, IMREAD_COLOR);
```

Figura 23: Captura del algoritmo de lectura de la imagen seleccionada.

Donde, “nombreArchivo” indica la ruta completa de la imagen a color seleccionada, IMREAD\_COLOR, es una bandera que indica la carga de una imagen en color, donde cualquier transparencia de imagen es despreciada, e “imagen” es una matriz definida con la estructura Mat de OpenCV (ver Figura 24).

```
69         Mat imagen;
```

Figura 24: Captura del algoritmo de la declaración de la matriz imagen.

#### 5.1.2.2 Obtención de canales

Las imágenes en color cargadas por OpenCV están en modo BGR (*Blue Green Red* - BGR), en lugar del común RGB. Por lo que al momento de desplegar o trabajar las imágenes se debe de considerar con atención este orden.

La imagen cargada es dividida en los tres canales tanto del modelo RGB como del modelo CIELab. Para la obtención de los canales RGB se realizan las instrucciones descritas en la Figura 25.

```
682     Mat imagenBGR[3];
683     imagenCargada.imagen.convertTo(imagenCargada.imagen, CV_8U);
684     split(imagenCargada.imagen, imagenBGR);
```

Figura 25: Captura del algoritmo para la obtención de los canales B, G y R.

Donde, por orden de aparición:

La matriz “imagenBGR[3]” posee tres canales.

La función *convertTo( )* convierte la matriz a otro tipo de datos con escala opcional; en este caso al tipo de datos CV\_8U, lo que significa que cada elemento de la matriz “imagen” es de 8 bits de profundidad, sin signo, es decir, que tomarán valores en el rango de [0, 255].

La función *split( )* divide una matriz multicanal en tres matrices de un solo canal. En este caso el resultado es almacenado en la matriz “imagenBGR”, en donde cada canal contendrá un canal de color en específico, en este caso en el orden B, G y R, respectivamente.

Para la obtención de los canales L, a y b del espacio CIELab se realizan las instrucciones presentadas en la Figura 26.

```
689     Mat imagenLab_temp;
690     cvtColor(imagenCargada.imagen, imagenLab_temp, CV_BGR2Lab);
691     Mat imagenLab[3];
692     split(imagenLab_temp, imagenLab);
```

Figura 26: Captura del algoritmo para obtención de los canales L, a y b.

Donde, *cvtColor( )* convierte una imagen de un espacio de color a otro; en este caso del espacio BGR a CIELab. Dado que “imagenCargada.imagen” es de 8 bits por canal, la conversión de los canales Lab toman un rango de [0,255], respectivamente.

El resultado de la división, por la función *split( )* de la matriz “imagenLab\_temp”, es almacenado en la matriz “imagenLab”, en donde cada canal contendrá un canal del espacio Lab en específico, en este caso en el orden L, a y b respectivamente.

### 5.1.2.3 Transformada de Fourier (FFT) en cada canal

La FFT deberá ser aplicada en cada uno de los canales de color obtenidos en el punto anterior. La función *dft( )* realiza una transformada de Fourier discreta directa o inversa de una matriz de 1D o 2D de punto flotante; por lo que se debe de transformar cada canal de color, definido en su matriz respectiva, es decir, imagenBGR e imagenLab, al rango de valores de punto flotante (de 0 a 1).

En la Figura 27 se puede observar dicha implementación para los canales BGR, y para los canales Lab en la Figura 28. En donde, las matrices para uso temporal; tempBlue, tempGreen, tempRed, tempL, tempa y tempb, contienen cada canal, B, G, R, L, a y b, respectivamente. Estas matrices se convierten a punto flotante a través de la función *convertTo()* y su parámetro CV\_32F.

```

700     Mat tempBlue, tempGreen, tempRed;
701     tempBlue = imagenBGR[0];
702     tempGreen = imagenBGR[1];
703     tempRed = imagenBGR[2];
704     tempBlue.convertTo(tempBlue, CV_32F);
705     tempGreen.convertTo(tempGreen, CV_32F);
706     tempRed.convertTo(tempRed, CV_32F);
707
708     Mat complexBlue, complexGreen, complexRed;
709
710     dft(tempBlue, complexBlue, DFT_COMPLEX_OUTPUT);
711     dft(tempGreen, complexGreen, DFT_COMPLEX_OUTPUT);
712     dft(tempRed, complexRed, DFT_COMPLEX_OUTPUT);

```

Figura 27: Captura del algoritmo para el cálculo de la FFT en los canales B, G y R.

Dado que se utiliza la bandera DFT\_COMPLEX\_OUTPUT en la función *dft()*, el resultado es una matriz compleja, que se almacena en cada matriz correspondiente, declarada como complexBlue, complexGreen, complexRed, complexL, complexa y complexb, para cada canal B, G, R, L, a y b, respectivamente.

```

724     Mat tempL, tempa, tempb;
725     tempL = imagenLab[0];
726     tempa = imagenLab[1];
727     tempb = imagenLab[2];
728     tempL.convertTo(tempL, CV_32F);
729     tempa.convertTo(tempa, CV_32F);
730     tempb.convertTo(tempb, CV_32F);
731     .
732     .
733     .
744     Mat complexL, complexa, complexb;
745
746     dft(tempL, complexL, DFT_COMPLEX_OUTPUT);
747     dft(tempa, complexa, DFT_COMPLEX_OUTPUT);
748     dft(tempb, complexb, DFT_COMPLEX_OUTPUT);

```

Figura 28: Captura del algoritmo para el cálculo de la FFT en los canales L, a y b.

#### 5.1.2.4 Magnitud de los valores complejos

Al tener las matrices complejas del resultado de la FFT en cada canal de color, se prosigue con el cálculo de sus magnitudes, como se muestra en la Figura 29. En este caso se muestra el algoritmo del cálculo de las magnitudes de las matrices complejas de los canales B, G y R, aunque se sigue el mismo método para las matrices complejas de los canales L, a y b.

```
763     Mat complexBlueSplit[2];
764     split(complexBlue, complexBlueSplit);
765     magnitud(complexBlueSplit[0], complexBlueSplit[1], imagenCargada.magnitudBlue);
766     ...
769     Mat complexGreenSplit[2];
770     split(complexGreen, complexGreenSplit);
771     magnitud(complexGreenSplit[0], complexGreenSplit[1], imagenCargada.magnitudGreen);
772     ...
774     Mat complexRedSplit[2];
775     split(complexRed, complexRedSplit);
776     magnitud(complexRedSplit[0], complexRedSplit[1], imagenCargada.magnitudRed);
```

Figura 29: Captura del algoritmo del cálculo de las magnitudes de las matrices complejas arrojadas por la FFT en los canales B, G y R.

Se definen las matrices de dos canales cada una, `complexBlueSplit`, `complexGreenSplit`, `complexRedSplit`, `complexLSplit`, `complexASplit` y `complexbSplit`, las cuales sirven para almacenar los valores reales e imaginarios, en cada canal, de las matrices complejas arrojadas por la FFT, a través de la función `split( )`. Posteriormente se calculan las magnitudes con la función `magnitud(x, y, resultado)` definida en OpenCV, que calcula la magnitud de los vectores 2D formados a partir de los elementos correspondientes de las matrices x e y:

$$resultado(I) = \sqrt{x(I)^2 + y(I)^2}$$

Y cuyo resultado se almacena en la matriz definida para cada canal, como `magnitudBlue`, `magnitudGreen` y `magnitudRed`.

#### 5.1.2.5 Cálculo de $\tilde{\theta}$ y $\tilde{\phi}$

El cálculo de los coeficientes  $\tilde{\theta}$  y  $\tilde{\phi}$  (ver sección 1.3.5.2) puede realizarse en varias modalidades; manual, individual y promedio. En este caso se abarcan solo los casos de promedio e individual, dado que el manual se genera a través de la GUI, aunque su comportamiento es como la modalidad de promedio.

Tanto para el cálculo individual como promedio de  $\tilde{\theta}$  y  $\tilde{\phi}$ , se utilizan inicialmente matrices, específicamente para este trabajo se declaran Mat TITA y Mat PHI, dado que todo coeficiente en su modalidad de promedio resulta de promediar los valores de dichas matrices (explicado más adelante).

El cálculo de  $\tilde{\theta}$  individual se realiza siguiendo las instrucciones indicadas en la Figura 30, donde se opera individualmente en cada frecuencia y el resultado obtenido para este coeficiente resulta en una matriz.

```

862     if (imagenCargada.valIndTITA == true) {
      .
      .
869     TITA = 1 - ((imagenCargada.magnitudL /
870         ((imagenCargada.magnitudBlue*0.114 + imagenCargada.magnitudGreen*0.587 + imagenCargada.magnitudRed*0.299)));

```

Figura 30: Captura del algoritmo del cálculo de  $\tilde{\theta}$  individual.

Para el cálculo de  $\tilde{\theta}$  promedio, se realizan las instrucciones indicadas en la Figura 31, donde se opera individualmente en cada frecuencia (al igual que en el cálculo de  $\tilde{\theta}$  individual) y posteriormente el resultado para este coeficiente se calcula promediando la suma de todos los valores de la matriz, obteniendo un valor escalar.

```

824     if (imagenCargada.valPromTITA == true) {
      .
      .
832     TITA = 1 - (imagenCargada.magnitudL /
833         ((imagenCargada.magnitudBlue*0.114) + (imagenCargada.magnitudGreen*0.587) + (imagenCargada.magnitudRed*0.299)));
      .
      .
842     int m, n;
843     m = imagenCargada.imagen.rows;
844     n = imagenCargada.imagen.cols;
845     double TITA2 = sum(TITA).val[0] / (m*n);
846
847     TITA = TITA2;

```

Figura 31: Captura del algoritmo del cálculo de  $\tilde{\theta}$  promedio.

El cálculo de  $\tilde{\phi}$  individual se realiza siguiendo las instrucciones indicadas en la Figura 32, donde se opera individualmente en cada frecuencia y el resultado obtenido para este coeficiente resulta en una matriz.

```

929     if (imagenCargada.valIndPHI == true) {
      .
      .
934     PHI = (imagenCargada.magnituda - imagenCargada.magnitudb) / (imagenCargada.magnituda + imagenCargada.magnitudb);

```

Figura 32: Captura del algoritmo del cálculo de  $\tilde{\phi}$  individual.

Para el cálculo de  $\tilde{\phi}$  promedio, se realizan las instrucciones indicadas en la Figura 33, donde se opera individualmente en cada frecuencia (al igual que en el cálculo de  $\tilde{\phi}$  individual) y posteriormente el resultado para este coeficiente se obtiene promediando la suma de todos los valores de la matriz, obteniendo un valor escalar.

```

899     if (imagenCargada.valPromPHI == true) {
      .
      .
906     PHI = (imagenCargada.magnituda - imagenCargada.magnitudb) / (imagenCargada.magnituda + imagenCargada.magnitudb);
      .
      .
911     int mm, nn;
912     mm = imagenCargada.imagen.rows;
913     nn = imagenCargada.imagen.cols;
914     double PHI2 = sum(PHI).val[0] / (mm*nn);
915     PHI = PHI2;

```

Figura 33: Captura del algoritmo del cálculo de  $\tilde{\phi}$  promedio.

### 5.1.2.6 Cálculo de la ecuación $\ddot{E}$

La implementación de la ecuación definida como  $\ddot{E}$  se muestra en la Figura 34, en donde se dividen las operaciones para trabajar con la parte real y la parte imaginaria de los números complejos, y, además, se resuelve la ecuación por partes. La matriz compleja “sumaAmasE” contiene el resultado final de la ecuación  $\ddot{E}$ .

```
961 //IMAGEN_MODIFICADA = ((1 - TITA).*FFT_L_Shift) + (TITA).*(PHI.* (FFT_a_Shift)+((1 - PHI).*FFT_b_Shift));
962
963
964 // *** ((1 - TITA).*FFT_L_Shift) ----- A
965 // Mat complexLSplit_temp[2];
966 complexLSplit_temp[0] = imagenCargada.complexLSplit[0];
967 complexLSplit_temp[1] = imagenCargada.complexLSplit[1];
968
969 Mat unoMenosTita = 1 - TITA; //1 - TITA
970 cv::multiply(imagenCargada.complexLSplit[0], unoMenosTita, complexLSplit_temp[0], 1.0);
971 cv::multiply(imagenCargada.complexLSplit[1], unoMenosTita, complexLSplit_temp[1], 1.0);
972
973 // *** PHI.* (FFT_a_Shift) ----- B
974 // Mat complexaSplit_temp[2];
975 complexaSplit_temp[0] = imagenCargada.complexaSplit[0];
976 complexaSplit_temp[1] = imagenCargada.complexaSplit[1];
977 cv::multiply(imagenCargada.complexaSplit[0], PHI, complexaSplit_temp[0], 1.0);
978 cv::multiply(imagenCargada.complexaSplit[1], PHI, complexaSplit_temp[1], 1.0);
979
980 // *** ((1 - PHI).*FFT_b_Shift) ----- C
981 // Mat complexbSplit_temp[2];
982 complexbSplit_temp[0] = imagenCargada.complexbSplit[0];
983 complexbSplit_temp[1] = imagenCargada.complexbSplit[1];
984
985 Mat unoMenosPhi = 1 - PHI;
986 cv::multiply(imagenCargada.complexbSplit[0], unoMenosPhi, complexbSplit_temp[0], 1.0);
987 cv::multiply(imagenCargada.complexbSplit[1], unoMenosPhi, complexbSplit_temp[1], 1.0);
988
989 // *** B + C = D ----- D
990 // Mat sumaBmasC[2];
991 cv::add(complexaSplit_temp[0], complexbSplit_temp[0], sumaBmasC[0]);
992 cv::add(complexaSplit_temp[1], complexbSplit_temp[1], sumaBmasC[1]);
993
994 // *** (TITA)*D = E ----- E
995 // Mat titaPorD[2];
996 cv::multiply(sumaBmasC[0], TITA, titaPorD[0], 1.0);
997 cv::multiply(sumaBmasC[1], TITA, titaPorD[1], 1.0);
998
999 // *** A + E
1000 // Mat sumaAmasE[2];
1001 cv::add(titaPorD[0], complexLSplit_temp[0], sumaAmasE[0]);
1002 cv::add(titaPorD[1], complexLSplit_temp[1], sumaAmasE[1]);
```

Figura 34: Captura del algoritmo del cálculo de la ecuación  $\ddot{E}$ .

La función *multiply()* calcula el producto escalado por elemento de dos matrices y la función *add()* calcula la suma por elemento de dos matrices o una matriz y un escalar. Ambas funciones forman parte de la biblioteca OpenCV.

### 5.1.2.7 Transformada inversa de Fourier

El resultado de la ecuación  $\ddot{E}$  es sometido a la transformada inversa de Fourier, aunque antes las matrices de valores reales e imaginarios deben unirse a una matriz multicanal. Se puede observar en la Figura 35 dicha implementación.

```
1016     Mat invDFTFinal, imagenFinal;
1017     Mat mergeAE;
1018     cv::merge(sumaAmasE, 2, mergeAE);
1019     cv::idft(mergeAE, invDFTFinal, DFT_SCALE | DFT_REAL_OUTPUT);
```

Figura 35: Captura del algoritmo para la obtención de la inversa de la transformada de Fourier.

Las funciones de OpenCV aplicadas son *merge()*, para crear una matriz multicanal a partir de varias de un solo canal (en este caso genera una matriz de dos canales), y *idft()*, para calcular la transformada inversa de Fourier discreta de una matriz de 1D o 2D; en donde la bandera *DFT\_SCALE* escala el resultado, dividiéndolo por el número de elementos de la matriz, y *DFT\_REAL\_OUTPUT* realiza una transformación inversa de una matriz compleja de 1D o 2D; dado que la transformación hacia adelante (*dft*) utilizó el indicador *DFT\_COMPLEX\_OUTPUT*, la salida es una matriz real.

El resultado final de la transformada inversa es almacenado en la matriz “invDFTFinal”.

### 5.1.2.8 Transformar resultado al espacio CIELab

El algoritmo propuesto resulta en la modificación del canal de luminancia original, en este caso el canal L del espacio de CIELab de la imagen original, por lo que la matriz “invDFTFinal” contiene los valores del canal L modificado, cuyo rango de los valores, al realizar la transformada inversa de Fourier, viene dado por el definido al inicio del algoritmo (ver sección 5.1.2.2), siendo de [0,255], previo a ser sometido a la transformada de Fourier hacia adelante.

El objetivo de esta etapa del algoritmo de transformación es reconstruir el espacio CIELab final de la transformación propuesta de la imagen de color a escala de grises. Esto se logra mediante una serie de instrucciones que se pueden observar en la Figura 36. El resultado de la transformación se considera como el canal L del espacio definido para obtener la respectiva escala de grises planteada.

```
1047     invDFTFinal.convertTo(imagenFinal, CV_32FC1);
1048     Mat matrizLCerosCeros[3], imagenLabBGR_final;
1049     imagenFinal = imagenFinal * 100 / 255;
1050
1051     matrizLCerosCeros[0] = imagenFinal;
1052     matrizLCerosCeros[1] = Mat::zeros(imagenCargada.imagen.rows, imagenCargada.imagen.cols, CV_32FC1);
1053     matrizLCerosCeros[2] = Mat::zeros(imagenCargada.imagen.rows, imagenCargada.imagen.cols, CV_32FC1);
1054     Mat imagenLabBGR(imagenCargada.imagen.rows, imagenCargada.imagen.cols, CV_32FC3);
1055     cv::merge(matrizLCerosCeros, 3, imagenLabBGR);
```

Figura 36: Captura del algoritmo de transformación del resultado obtenido al espacio CIELab.

El primer paso es convertir los valores de la matriz “invDFTFinal” de CV\_8U a CV\_32FC1, cuyo resultado de la conversión se almacena en la matriz nombrada “imagenFinal”, para su posterior uso en el

manejo de los canales del espacio de color CIELab, donde al utilizar el tipo de datos de 32 bits, los rangos de los valores definidos para cada canal son de [0,100] para L, [-127,127] para a y [-127,127] para b. Posteriormente se escalan los valores de la matriz “imagenFinal” al rango de [0,100] (futuro canal L de la transformación propuesta), para la manipulación del espacio CIELab del tipo CV\_32F a obtener.

Las matrices de un solo canal nombradas como “matrizCerosCeros”, contienen, respectivamente desde la matriz[0] a la matriz[2], el canal L (con los valores de la matriz “imagenFinal”), y los canales a y b, declarados como matrices de ceros, a través de la función *zeros()*, con el mismo tipo de datos del canal L; para este caso CV\_32FC1. Dado que en el espacio CIELab de 32 bits el valor de 0 en los canales a y b indican que no hay cambios cromáticos, se estarían representando únicamente los cambios de luminancia indicados por el canal L.

El conjunto de matrices “matrizCerosCeros” se unen en una sola matriz de 3 canales a través de la función *merge()*, y el resultado se almacena en la matriz nombrada “imagenLabBGR”, conteniendo la imagen final obtenida por la transformación propuesta en el espacio CIELab.

### 5.1.2.9 Transformar resultado al espacio BGR

La imagen obtenida en el espacio CIELab no puede ser desplegada o almacenada por lo que su conversión al espacio BGR (ver Figura 37) es necesaria para cualquiera de estas dos acciones, siempre y cuando se utilicen las clases facilitadas por OpenCV. En otros casos puede ser necesaria la conversión al espacio RGB.

```
1056 cv::cvtColor(imagenLabBGR, imagenLabBGR_final, CV_Lab2BGR);
1057 imagenLabBGR_final.convertTo(imagenLabBGR_final, CV_8U, 255.0);
```

Figura 37: Captura del algoritmo de transformación del espacio CIELab al espacio BGR.

La función *cvtColor()* convierte la imagen “imagenLabBGR” del espacio CIELab al espacio BGR, al utilizar la bandera CV\_Lab2BGR. Dado que la matriz “imagenLabBGR” contiene valores del tipo de dato de 32 bits (CV\_32FC3), el rango de valores obtenido en esta conversión es de [0,1]. El resultado es almacenado en la matriz “imagenLabBGR\_final”.

La función *convertTo()*, en este caso, a través del parámetro CV\_8U convierte la matriz “imagenLabBGR\_final” del tipo de dato de CV\_32F al tipo de datos de 8 bits, sin signo, de profundidad por canal, y el indicador de escala de 255.0, escala los valores en el rango de [0, 255].

La imagen obtenida en este paso es utilizada por bibliotecas de GUI, con algunas modificaciones que se explicarán en la siguiente sección.

## 5.2 GUI

La interacción del algoritmo de transformación propuesto con el usuario se realiza a través de una GUI (cuya vista con una imagen seleccionada se puede ver en la Figura 20), en la cual la interacción permitida incluye: seleccionar la imagen a color a transformar, modificar los coeficientes  $\theta$  y  $\phi$ , restablecer valores, mostrar la imagen original, manipular la resolución de la imagen y guardar la imagen transformada. Las interacciones que generen cambios en la imagen implican el despliegue del resultado de la imagen en

tiempo real. El diagrama de secuencia de dicha interacción se puede observar en la Figura 38, donde se muestran las interacciones de los tres objetos de la aplicación (el usuario, la GUI y el algoritmo planteado).

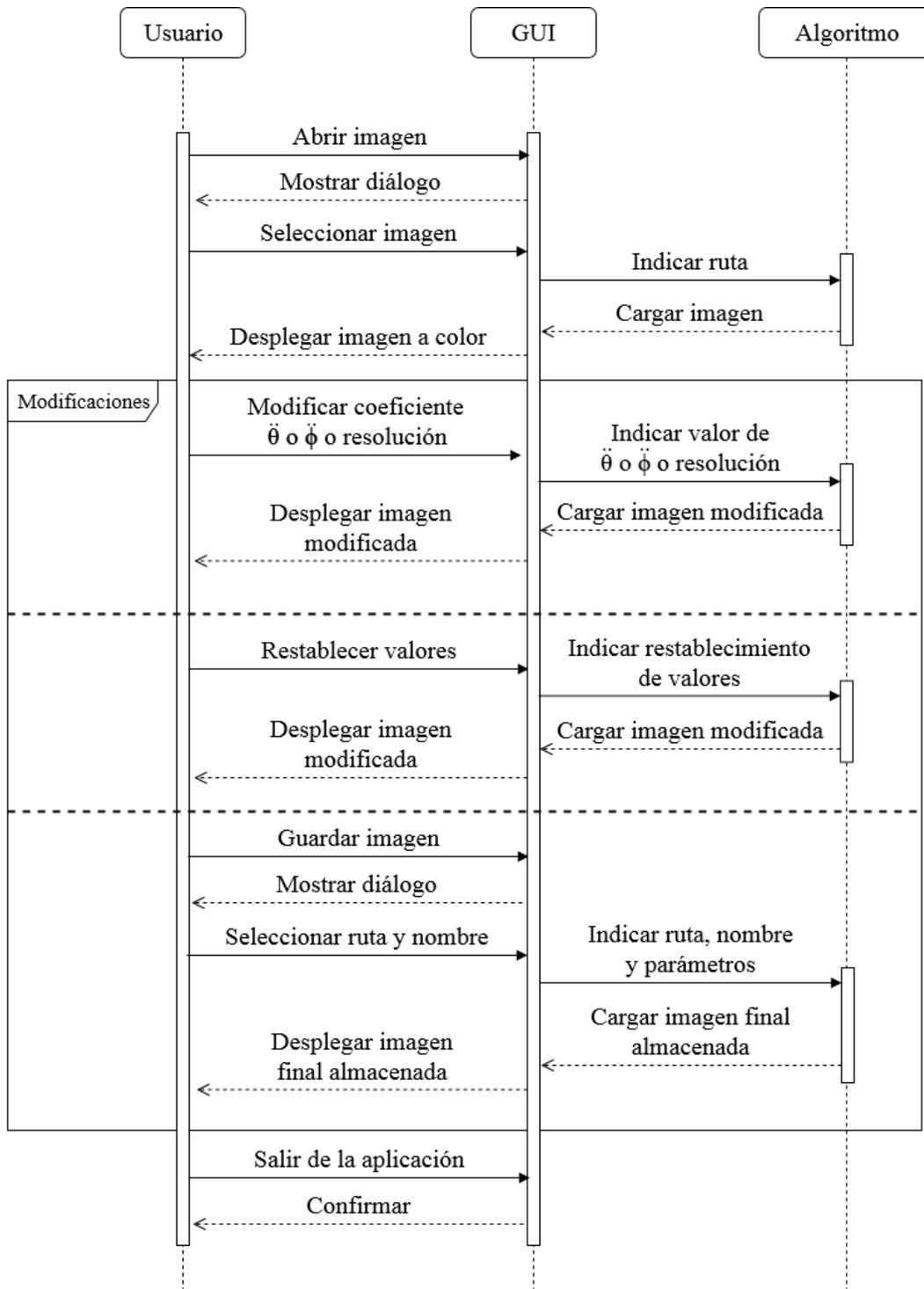


Figura 38: Diagrama de secuencia de la interacción entre el usuario, la GUI y el algoritmo de transformación.

## 5.2.1 Herramientas para el desarrollo

Una herramienta que facilita el desarrollo de la GUI es el IDE Qt Creator, que permite crear aplicaciones C++ y QML (*Qt modeling language* - Lenguaje de modelado Qt) multiplataforma. Incluye un editor de código y un espacio para diseñar y construir interfaces gráficas de usuario (GUI) a partir de Qt widgets (elementos de control gráficos). Qt Creator es parte del SDK (*software development kit* - Kit de desarrollo de software) del framework Qt, en este caso, versión 5.10.0, disponible tanto bajo licencias comerciales como de código abierto. Es posible componer y personalizar los widgets o diálogos y probarlos usando diferentes estilos y resoluciones directamente en el editor.

La integración del código, e interacción, entre el algoritmo implementado (que utiliza funciones de OpenCV) y los elementos de la GUI (diseñada en Qt Creator) se implementa mediante el IDE Visual Studio 2017, utilizando el lenguaje C++.

## 5.2.2 Bibliotecas

Las bibliotecas utilizadas para la implementación de la GUI y que forman parte de los módulos del SDK de Qt, son QtCore y QtWidgets. Donde QtCore es una biblioteca base que proporciona contenedores, gestión de subprocesos, gestión de eventos, entre otros; y QtWidgets proporciona una biblioteca de elementos de la interfaz de usuario para crear interfaces de usuario del estilo de escritorio clásico. La clase QWidget proporciona la capacidad básica para renderizar en la pantalla y para manejar eventos de entrada de usuario. Todos los elementos de la interfaz de usuario que proporciona Qt son subclases de QWidget o se utilizan en conexión con una subclase de QWidget.

### 5.2.2.1 Signals y slots

En la programación de GUI con Qt, cuando se quiere que el cambio de un widget notifique a otro widget, se utilizan señales (*signals*) y ranuras (*slots*). Este mecanismo es una característica central de Qt. Un *signal* se emite por un objeto cuando cambia su estado de una manera que puede ser interesante para otros objetos. Un *slot* es una función que se invoca en respuesta a un *signal* particular. Todas las clases que heredan de QObject o una de sus subclases (como QWidget) pueden contener *signals* y *slots*.

Los widgets implementados para esta aplicación con Qt Creator se integran con el código programado, utilizando este mecanismo a través de la función `connect()`. Un ejemplo de la implementación con esta conexión se puede observar en la Figura 39, en donde al liberar (`SIGNAL(released())`) el botón “Abrir imagen” (elemento identificado como “pushButtonAbrir”) se invoca a la función `botonCargarImagen()`.

```
38 connect(ui.pushButtonAbrir, SIGNAL(released()), this, SLOT(botonCargarImagen()));
```

Figura 39: Capture del algoritmo de conexión con *signal* y *slot*.

### 5.2.2.2 Diálogo de selección de imagen

La selección de la imagen a transformar de color a escala de grises se realiza, posterior a presionar el botón “Abrir imagen”, a través de la clase QFileDialog, que proporciona un cuadro de diálogo que permite a los usuarios seleccionar archivos o directorios. En la Figura 40 se puede observar el algoritmo implementado y en la Figura 41 el resultado gráfico.

```

55 void QtGuiApplication3::botonCargarImagen()
56 {
57
58     QString fileName;
59     fileName = QFileDialog::getOpenFileName(this, tr("Abrir imagen..."),
60         "/home", tr("Imágenes (*.png *.jpg *.tif *.bmp)"));
61
62     if (fileName.isEmpty()) {
63         ...

```

Figura 40: Capture del algoritmo que contiene la invocación al elemento del diálogo de selección de la imagen a transformar.

La clase `QString`, proporcionada por Qt, permite crear cadenas de caracteres de tipo Unicode. En este caso la cadena llamada "fileName" servirá en esa sección para almacenar la ruta de la imagen seleccionada, necesaria para siguientes secciones.

La función estática `getOpenFileName()` devuelve un archivo existente seleccionado por el usuario. Si el usuario presiona Cancelar, devuelve una cadena nula. Solo se muestran los archivos que coinciden con el filtro indicado.

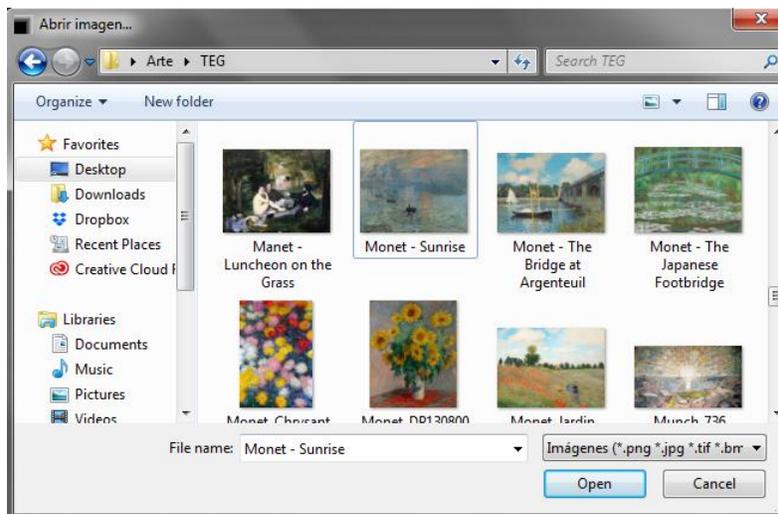


Figura 41: Captura del cuadro de diálogo para la selección de la imagen a transformar.

Dado que OpenCV trabaja con la clase `string` en lugar de `QString` de Qt, se transforma de la codificación Unicode a la codificación UTF8, para permitir el uso de la ruta del archivo seleccionada por el usuario por OpenCV. En la Figura 42 se puede observar el algoritmo implementado para esta transformación.

```

1076 void QtGuiApplication3::direccionEntrada(QString direccion)
1077 {
1078     imagenCargada.nombreArchivo = direccion.toUtf8().constData();

```

Figura 42: Captura del algoritmo de transformación de `QString` a `string`.

### 5.2.2.3 Despliegue de la imagen

Las modificaciones que se realicen a la imagen, es decir, cualquier cambio de valor de coeficientes o resolución, de forma manual o automática, deberán ser visualizadas por el usuario a través de una pantalla, en un área destinada para tal función. Esto se logra a través del widget QLabel, que es una clase utilizada para mostrar texto o una imagen, no proporciona ninguna funcionalidad de interacción del usuario y puede contener diferentes tipos de contenido, como, para este caso, mapa de píxeles ( *pixmap* ).

La visualización de la imagen seleccionada por primera vez es a color, a través del diálogo de selección, el resto de las modificaciones realizadas a esta imagen generará resultados en escala de grises. Es importante destacar que Qt trabaja con el espacio de color RGB mientras que OpenCV trabaja en BGR, por lo que luego de aplicar las modificaciones a la imagen en OpenCV se deberá transformar su espacio de color.

En la implementación realizada (que se puede observar en la Figura 43), para la visualización de la imagen final de la transformación de color a escala de grises propuesta, se convierte la matriz obtenida en BGR al espacio RGB, luego se convierte en un objeto QPixmap (a través de la función *fromImage( )*) para que pueda ser desplegado por pantalla utilizando el widget QLabel nombrado para esta aplicación “label\_pic”. Para poder visualizar la imagen completa en “label\_pic”, la imagen es escalada por el alto y el ancho de éste, manteniendo su relación de aspecto original.

```
1086 void QtGuiApplication3::cvMat2QPixmap(Mat matrizImagen) {
1087
1088     int matFilas = matrizImagen.rows;
1089     int matColumnas = matrizImagen.cols;
1090     ...
1092     Mat imagenLabBGRtoRGB(matFilas, matColumnas, CV_32FC3);
1093     cv::cvtColor(matrizImagen, imagenLabBGRtoRGB, CV_BGR2RGB);
1094
1095     int al = ui.label_pic->width();
1096     int an = ui.label_pic->height();
1097     ...
1101     QPixmap pixMapFinal(QPixmap::fromImage(QImage((uchar*)imagenLabBGRtoRGB.data,
1102         imagenLabBGRtoRGB.cols,
1103         imagenLabBGRtoRGB.rows,
1104         imagenLabBGRtoRGB.step,
1105         QImage::Format_RGB888)));
1106     ...
1109     ui.label_pic->setPixmap(pixMapFinal.scaled(al, an, Qt::KeepAspectRatio));
1110     ui.label_pic->repaint();
1111
1112 }
```

Figura 43: Captura del algoritmo de despliegue de la imagen transformada en el QLabel nombrado “label\_pic”.

### 5.2.2.4 Valores manuales

La modificación de forma manual de los coeficientes  $\theta$ ,  $\phi$  y  $\beta$ , se puede llevar a cabo a través de dos tipos de elementos, QLineEdit y QSlider, que se pueden observar en la Figura 44.

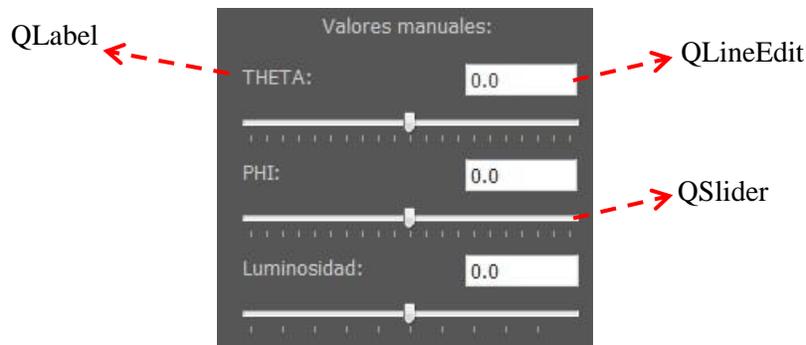


Figura 44: Captura e indicación de widgets de la sección de control de valores de coeficientes de forma manual de la interfaz de usuario implementada.

El widget QLineEdit es un editor de texto de una línea. Una edición de línea permite al usuario ingresar y editar una sola línea de texto sin formato.

El widget QSlider proporciona un control deslizante vertical u horizontal. Permite al usuario mover un control deslizante (*slider*) a lo largo de un canal horizontal o vertical y traduce la posición del controlador en un valor entero dentro del rango legal; dentro del valor acotado definido para el control deslizante.

Para cada coeficiente, tanto su slider como su edición de línea están interconectados en la implementación realizada, por lo que el cambio en el slider (*SIGNAL(valueChanged( ))*) afecta el valor presentado en la edición de línea y un cambio del valor indicado en la edición de línea (*SIGNAL(editingFinished( ))*) afecta la posición del slider. La implementación de la interconexión se muestra en la Figura 45.

```

34 connect(ui.horizontalSliderTITA, SIGNAL(valueChanged(int)), this, SLOT(cambiarLabelTITA(int)));
    ...
36 connect(ui.lineEditTITA, SIGNAL(editingFinished()), this, SLOT(cambiarSliderTITA()));

```

Figura 45: Captura de la implementación de conexión entre la edición de línea y el slider del coeficiente  $\theta$ .

Una parte del algoritmo de la función *cambiarLabelTITA( )* se presenta en la Figura 46. Donde se calcula el valor a mostrar en la edición de línea, correspondiendo a la posición que tenga en indicador de la barra deslizante. Cabe destacar que la barra deslizante toma valores enteros entre -9999 y 9999, que al llevarlos al rango de  $[-3, +3]$  permiten obtener cuatro decimales para la manipulación del coeficiente  $\theta$ . Lo mismo sucede con el coeficiente  $\phi$  y  $\beta$ , con la diferencia en que sus rangos son de  $[-5, +5]$  y  $[-1, +1]$ , respectivamente.

```

124 void QtGuiApplication3::cambiarLabelTITA(int valor) {
125
126     float valueTITA = 3.0*((float)valor / (float)ui.horizontalSliderTITA->maximum());
127     int enteroTITA = (int)(valueTITA * 10000.0);
128     double flotanteTITA = (double)(0.0);
129     flotanteTITA = enteroTITA / 10000.0;
130     ui.lineEditTITA->setText(QVariant(flotanteTITA).toString());

```

Figura 46: Captura de parte del algoritmo que modifica la edición de línea del coeficiente  $\theta$ .

Una fracción del algoritmo de la función *cambiarSliderTITA()* se presenta en la Figura 47. Donde se calcula la posición final que tendrá el indicador de la barra deslizante, correspondiendo al valor indicado por la edición de línea. Si el valor indicado en la edición de línea excede al rango permitido por el *slider* correspondiente, se posiciona su barra deslizante en el extremo positivo o negativo, según sea el caso.

```

239 void QtGuiApplication3::cambiarSliderTITA() {
240
241     QString valueTITA2 = ui.lineEditTITA->text();
242     double valorTITAint = valueTITA2.toDouble();
243     valorTITAint = valorTITAint * 10000 / 3;
244     ui.horizontalSliderTITA->setValue(valorTITAint);

```

**Figura 47:** Captura de parte del algoritmo que modifica la posición de la barra deslizante del coeficiente  $\theta$ .

Al momento de utilizar en OpenCV los valores de los coeficientes indicados de forma manual, se transforma a flotante el valor indicado, dado que el elemento QLineEdit contiene el texto como QString.

Posterior a cada modificación realizada en cualquiera de los elementos descritos, se invoca la función de transformación de la imagen de color a escala de grises propuesto; activando una bandera que indica una modificación manual, y a continuación se genera el despliegue por pantalla del resultado sobre la imagen. La manipulación de las banderas utilizadas a lo largo de la aplicación no es detallada a fondo por razones de simplificación.

### 5.2.2.5 Valores automáticos

El usuario puede seleccionar el cálculo automático de los coeficientes  $\theta$  y  $\phi$  a través de los botones destinados para ello, definidos como elementos QPushButton, que luego de ser presionados invocan a las funciones propuestas en la sección 5.1.2.5. El resultado de la implementación gráfica se puede observar en la Figura 48, en donde el botón “Valores promedio” invoca a la función (*selecValProm()*) que indica el cálculo del promedio de ambos coeficientes y “Valores individuales” invoca a la función (*selecValInd()*) que indica el cálculo de ambos coeficientes como operaciones individuales en cada frecuencia.



**Figura 48:** Captura e indicación de widgets de la sección de control de valores de coeficientes de forma automática de la GUI implementada.

La implementación de la conexión entre cada botón indicado y la invocación a la función respectiva se muestra en la Figura 49.

Posterior a cada modificación realizada en cualquiera de los elementos descritos, se invoca al algoritmo de transformación de la imagen de color a escala de grises propuesto; activando la bandera que indica el tipo cálculo de coeficiente (individual o promedio), y consecutivamente se genera el despliegue por pantalla del resultado sobre la imagen.

```

38     connect(ui.pushButtonValProm, SIGNAL(released()), this, SLOT(selecValProm()));
39     connect(ui.pushButtonValInd, SIGNAL(released()), this, SLOT(selecValInd()));

```

Figura 49: Captura de la implementación de conexión de los botones del cálculo automático (promedio e individual) de los coeficientes  $\theta$  y  $\phi$ .

### 5.2.2.6 Restablecer valores

Para obtener la transformación de color a escala de grises dada por el canal L de CIELab, sin intervención de los coeficientes  $\theta$ ,  $\phi$  y  $\beta$ , se presiona el botón “Restablecer valores”. El resultado de la implementación gráfica del botón se muestra en la Figura 50.

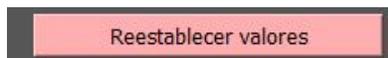


Figura 50: Captura del botón que restablece valores de los coeficientes de la interfaz de usuario implementada.

La conexión entre este botón, nombrado “pushButtonReestVal”, y la invocación a la función *reestValores()* (propuesta en la sección 4.2.5) se muestra en la Figura 51.

```

41     connect(ui.pushButtonReestVal, SIGNAL(released()), this, SLOT(reestValores()));

```

Figura 51: Captura de la implementación de conexión del botón de restablecimiento de valores y su función respectiva.

Parte de la función *reestValores()*, que restablece los coeficientes mencionados y las banderas respectivas sobre el tipo de cálculo de coeficientes, se muestra en la Figura 52, donde se ejecuta la modificación de los widgets respectivos (*sliders*) y se indican los valores de las banderas relacionadas al tipo de cálculo de los coeficientes (quedando seleccionado el tipo de cálculo manual). Posterior a la modificación de los coeficientes, se procede a ejecutar el algoritmo propuesto y se genera el despliegue por pantalla del resultado sobre la imagen.

```

638     void QtGuiApplication3::reestValores() {
        .
        .
633         ui.horizontalSliderTITA->setValue(0.0);
634         ui.horizontalSliderPHI->setValue(0.0);
635         ui.horizontalSliderLuminancia->setValue(0.0);
        .
        .
638         imagenCargada.valPromTITA = false;
639         imagenCargada.valIndTITA = false;
640         imagenCargada.valPromPHI = false;
641         imagenCargada.valIndPHI = false;
642         imagenCargada.valManualTITA = true;
643         imagenCargada.valManualPHI = true;
644         .
        .

```

Figura 52: Captura de parte del algoritmo que se ejecuta al seleccionar la opción de restablecimiento de valores.

### 5.2.2.7 Mostrar imagen original

Al desplegar la imagen en escala de grises, resultante del algoritmo de transformación propuesto, puede ser necesaria la visualización momentánea de la imagen original a color, por razones de soporte para el usuario al momento de seleccionar los coeficientes que mejor se adapten al resultado deseado. Para esto se implementa el botón “Mostrar imagen original” (ver Figura 53), que despliega la imagen a color original sobre la imagen en escala de grises, únicamente mientras se mantenga presionado el botón.



Figura 53: Captura del botón de la GUI implementada que ejecuta el despliegue de la imagen original a color por pantalla.

En la Figura 54 se indica la implementación de la conexión del botón, en donde dependiendo si se encuentra presionado (*pressed()*) o liberado (*released()*), se muestra o se oculta, respectivamente, el QLabel que contiene la imagen a color.

```
42     connect(ui.pushButtonImagenOriginal, SIGNAL(pressed()), this, SLOT(mostrarOriginal()));
43     connect(ui.pushButtonImagenOriginal, SIGNAL(released()), this, SLOT(ocultarOriginal()));
```

Figura 54: Captura de la conexión implementada para el botón "Mostrar imagen original".

### 5.2.2.8 Resolución

La interacción del usuario con el algoritmo para con imágenes de gran tamaño (que superen los 840x840 píxeles para esta aplicación) puede tornarse en una espera de procesamiento que no permita la interacción en tiempo real, para solucionar este inconveniente se permite modificar la resolución de la imagen, independientemente si ésta ya ha sido transformada a escala de grises, a través de una edición de línea o un *slider*, como se muestra en la Figura 55. La modificación de la resolución siempre se hará manteniendo la relación de aspecto original.

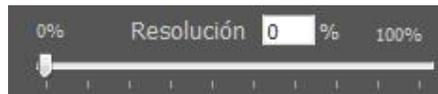


Figura 55: Captura de la GUI implementada para permitir la modificación de la resolución de la imagen.

Si se modifica la resolución con algún coeficiente  $\theta$ ,  $\phi$  y  $\beta$  establecido de forma manual, se ejecuta el algoritmo de transformación propuesto con dichos parámetros y la imagen escalada. En caso de que el valor de los coeficientes sea automatizado, éste puede variar al cambiar la resolución.

Al igual que en la sección 5.2.2.4, la edición de línea y el *slider* relacionados a la resolución, se encuentran conectados, por lo que al momento de modificar uno de ellos el otro se verá afectado. La implementación de esta conexión se puede observar en la Figura 56.

```

46 connect(ui.horizontalSliderResolucion, SIGNAL(valueChanged(int)), this, SLOT(cambiarLabelResolucion()));
47 connect(ui.lineEditResolucion, SIGNAL(editingFinished()), this, SLOT(cambiarSliderResolucion()));

```

Figura 56: Captura de la implementación de conexión entre la edición de línea y el slider de cambio de resolución.

### 5.2.2.9 Guardar imagen y salir

La decisión por parte del usuario de salir de la aplicación o de guardar la imagen obtenida de la transformación, se puede generar en cualquier momento que lo desee, mientras no se esté ejecutando otra tarea de la aplicación. La implementación de la GUI de estos botones se puede observar en la Figura 57.



Figura 57: Captura de la GUI implementada de los botones "Guardar" y "Salir", e identificación del widget de casilla de verificación.

En la Figura 57 se puede observar que se permite una casilla de verificación (implementada con el widget QCheckBox) con la etiqueta de “Guardar con resolución original”, lo que permite, antes de ejecutar el algoritmo de transformación, determinar si se aplicará sobre la imagen en su tamaño original o sobre la imagen escalada.

La conexión de los botones “Guardar Imagen” y “SALIR” con las funciones que invocan se muestran en la Figura 58.

```

33 connect(ui.pushButtonGuardar, SIGNAL(released()), this, SLOT(botonGuardarImagen()));
...
40 connect(ui.pushButtonSalir, SIGNAL(released()), this, SLOT(mensajeSalir()));

```

Figura 58: Captura de la implementación de conexión del botón “Guardar imagen” y del botón “Salir”.

La función *botonGuardarImagen()* permite especificar la ruta y el nombre final del archivo a través del cuadro de diálogo (*QFileDialog::getSaveFileName()*); dado que se estudiarán los resultados obtenidos con este algoritmo, se decide mantener por defecto el nombre original de la imagen y agregar datos como el tipo de coeficientes utilizados ( $\theta$  y  $\phi$ , individual, promedio o manual, y  $\beta$ ) con su valor respectivo, el tiempo de ejecución del algoritmo en segundos y porcentaje de resolución utilizado. El resultado se puede almacenar en formato JPG o PNG.

## 6 PRUEBAS Y RESULTADOS

El algoritmo implementado es evaluado para determinar si genera una escala de grises perceptualmente aceptable, que maximice la percepción visual. Por lo que sus resultados se someten a evaluaciones subjetivas y objetivas para así realizar los análisis y establecer conclusiones pertinentes.

Para una variedad de imágenes a color, en la Figura 59 se muestra su resultado en escala de grises por diferentes métodos de distintas investigaciones en el área de decoloración de la imagen, (a) CIE Y, (b) Bala y Braun, (c) Gooch et al., (d) Grundland et al., (e) Neumann et al., (f) Smith et al. y (g) Método propuesto (con  $\bar{\theta}$  y  $\bar{\phi}$  promedios). Para mayor información referente a los autores y los parámetros utilizados, consultar los trabajos de Čadík [18] y Kim et al. [28].

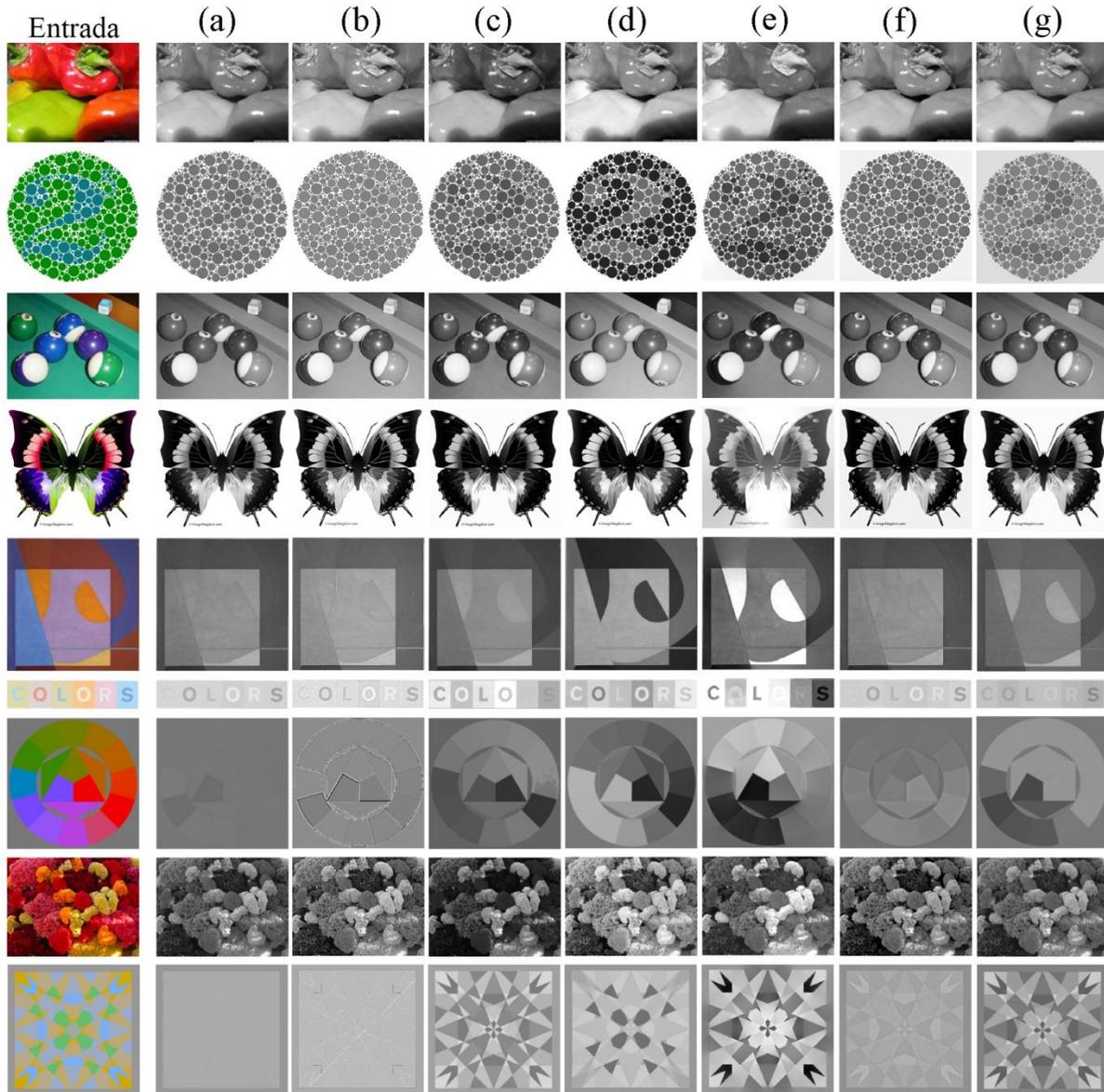


Figura 59: Comparación entre los resultados de los diferentes métodos de transformación de imágenes de color a escala de grises. [18]

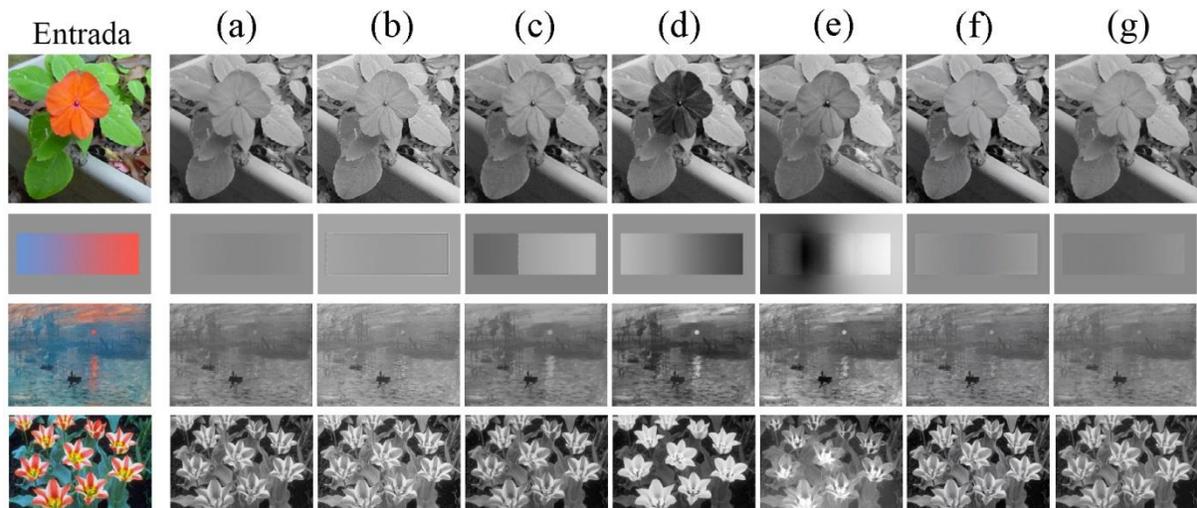


Figura 5.9 (Cont.): Comparación entre los resultados de los diferentes métodos de transformación de imágenes de color a escala de grises. [18]

## 6.1 Imágenes a evaluar

El conjunto de imágenes a color seleccionadas y almacenadas en el disco local, sometidas a evaluación para estudiar el algoritmo propuesto se presentan en la Figura 60, donde se muestran veinte imágenes de



Figura 60: Imágenes de prueba para el algoritmo implementado.

pinturas de diversos artistas, obtenidas de distintas páginas web dedicadas al arte, que permiten la descarga de reproducciones digitales en diversas resoluciones, libres al público para su descarga y uso sin fines de lucro. Entre los sitios web se encuentran: [images.nga.gov](http://images.nga.gov), [artgallery.yale.edu](http://artgallery.yale.edu), [en.gallerix.ru](http://en.gallerix.ru), entre otros.

Las imágenes presentadas en la Figura 60 se detallan en la Tabla 2, con su identificador de imagen utilizado para este trabajo, el nombre de la obra, el autor, la resolución en píxeles de la digitalización utilizada y el tamaño de almacenamiento que ocupa.

<b>Id.</b>	<b>Nombre</b>	<b>Autor</b>	<b>Resolución (píxeles)</b>	<b>Tamaño</b>
Imagen 1	In the Garden of Bellevue	Edouard Manet	3203x4125	3,15 MB
Imagen 2	Cafe-Concert: The Song of the Dog	Edgar Degas	810x1000	2,16 KB
Imagen 3	The Gulf of Marseilles Seen from L'Estaque	Paul Cézanne	3705x2696	2,04 MB
Imagen 4	Fruits	Paul Cézanne	3799x2996	2,27 MB
Imagen 5	Color Study	Wassily Kandinsky	1067x797	154 KB
Imagen 6	Martyrdom of the Ten Thousand	Alberto Durero	2898x3329	3,03 MB
Imagen 7	Bouquet of Sunflowers	Claude Monet	3238x4000	6,03 MB
Imagen 8	The Scream (1893)	Edvard Munch	1397x1759	342 KB
Imagen 9	Haystacks in Brittany	Paul Gauguin	4000x3178	16,3 MB
Imagen 10	L'atelier (Deux personnages)	Pablo Picasso	1064x854	225 KB
Imagen 11	Buste de Dora Maar	Pablo Picasso	736x865	139 KB
Imagen 12	Les Coquelicots	Claude Monet	2500x1870	1,11 MB
Imagen 13	The circus	Georges Seurat	1184x1528	398 KB
Imagen 14	Wheat Field with Cypresses	Vincent Van Gogh	3811x3016	3,93 MB
Imagen 15	Vincent's Bedroom in Arles	Vincent Van Gogh	4433x3500	5,68 MB
Imagen 16	Self-Portrait with Straw Hat	Vincent Van Gogh	3204x4200	4,75 MB
Imagen 17	Sábado	Carlos Cruz-Diez	720x720	248 KB
Imagen 18	La adoración de la virgen	Diego Rivera	1608x1993	346 KB
Imagen 19	Las dos Fridas	Frida Kahlo	729x733	84,8 KB
Imagen 20	Impression Sunrise	Claude Monet	1600x1245	1,90 MB

**Tabla 2: Características de las imágenes a color utilizadas para la evaluación del algoritmo propuesto.**

### 6.1.1 Tipos de coeficientes

El algoritmo propuesto permite la manipulación de la imagen resultante en escala de grises a través de distintos coeficientes, como se describe en la sección 4.1.1. Para este caso se evalúa solamente  $\theta$  y  $\phi$ , tanto en su forma automática como manual, obteniendo entonces las combinaciones de coeficientes mostradas en la Tabla 3, donde “Imagen X” identifica la tabla de la imagen a la cual se le aplica los valores de los coeficientes indicados, la columna de letras “A, B, ... J” identifican el par de coeficientes  $\theta$  y  $\phi$  aplicados y la columna de “Tiempo” identifica los segundos que tardó la aplicación en obtener el resultado en escala de grises con dichos parámetros en la imagen con la resolución original.

Para cada una de las veinte imágenes a color a evaluar se lleva el control de coeficientes con su tabla respectiva, dado que los coeficientes utilizados para cada imagen son únicos, es decir, que no se utiliza el mismo valor de coeficiente para todas las imágenes; por ejemplo, el valor automático de coeficiente de tipo promedio no será el mismo para la “Imagen 1” que para la “Imagen 2”. En el caso de los valores automatizados de tipo individual, que genera una matriz como coeficiente, no se detallada dicha matriz sino que se identifica dicho coeficiente con el nombre “Individuales”.

Imagen X			
	$\theta$	$\phi$	Tiempo (seg)
<b>A</b>	Promedio	Promedio	
<b>B</b>	Individuales	Individuales	
<b>C</b>	Promedio	Individuales	
<b>D</b>	Individuales	Promedio	
<b>E</b>	Manual	Manual	
<b>F</b>	Promedio	Manual	
<b>G</b>	Manual	Promedio	
<b>H</b>	Individuales	Manual	
<b>I</b>	Manual	Individuales	
<b>J</b>	L de CIELab		

**Tabla 3: Tabla que identifica los coeficientes a evaluar por imagen.**

Las imágenes en escala de grises evaluadas son las obtenidas con el algoritmo propuesto al aplicarles los distintos tipos de coeficientes mostrados en la Tabla 3 y el resultado de luminosidad (sección 1.3.4), canal L de CIELab, transformado al espacio RGB; dado que es un algoritmo trivial de transformación de imágenes de color a escala de grises que arroja resultados perceptualmente acertados, con rápidos tiempos de ejecución y de fácil implementación.

Las tablas de los resultados de coeficientes aplicados a cada una de las imágenes se pueden observar en el Anexo 1.

## 6.2 Aplicación de encuestas

Los resultados de las imágenes en escala de grises obtenidas a través del algoritmo propuesto con los distintos coeficientes que éste utiliza y el canal L de CIELab, se evaluaron de forma subjetiva a través de encuestas realizadas utilizando comparaciones por pares, descrita en la sección 2.1, donde se le solicita a los sujetos de prueba que seleccionen la imagen en escala de grises que consideren como favorita de un par de imágenes en escala de grises y su imagen a color como referencia.

Una muestra de la evaluación que se le presentó a los sujetos se puede observar en la Figura 61, donde las letras A y B en la parte inferior de las imágenes identifican el par de valores de coeficientes utilizados para esa imagen (como se detalla en la sección anterior), y en la parte superior se identifica la referencia del número de la imagen sometida a evaluación. El orden de presentación de las imágenes a evaluar y el par de selección se establecieron de forma aleatoria, aunque se diseñaron únicamente cinco modelos de prueba que abarcan todas las comparaciones necesarias para poder evaluar los resultados, sin obviar ningún par.

En resumen, se cuenta con 20 imágenes a color, 10 resultados en escala de grises por cada una de ellas y seis 6 evaluaciones por combinación de pares (entre distintos sujetos). Por imagen a color, se tienen  $\frac{10(10-1)}{2} = 45$  permutaciones de combinaciones por pares de resultados en escala de grises, dando un total de  $20 * 45 * 6 = 5400$  evaluaciones de combinaciones por pares. Estas 5400 comparaciones se distribuyeron de forma aleatoria entre 30 sujetos de prueba, con un total de 180 comparaciones por participante. Dado que el tiempo recomendado para que cada sujeto evalúe las imágenes sin presentar signos de fatiga es de 30 minutos, se realizan las evaluaciones en varias sesiones.



**Figura 61: Ejemplo de una evaluación de comparación por pares mostrada al sujeto de prueba.**

La decisión de cada una de las comparaciones tomada por cada sujeto son registradas en un formulario previamente entregado, y cuyo modelo se puede observar en el Anexo 2. Los resultados finales de todas las encuestas recopiladas se trasladaron a una hoja de cálculo (en este caso se utiliza Microsoft Excel) donde se llevan a cabo el conjunto de cálculos pertinentes, aplicando la Ley de Juicio Comparativo de Thurstone Caso V (explicados a lo largo de la sección 2.1) para determinar los índices de posiciones que ocupan cada uno de los resultados en escala de grises.

### 6.2.1 Resultados de las encuestas

Los pasos y cálculos de escalamiento descritos en el procedimiento de la Ley del Juicio Comparativo de Thurstone Caso V (explicados en las secciones 2.1.3, 2.1.4 y 2.1.5) son aplicados a los datos obtenidos a través de las encuestas y se pueden resumir en:

Primero se genera una matriz de frecuencias (descritas en la sección 2.1) de los resultados de las encuestas por cada imagen a color y una matriz de frecuencias de la suma de todas las frecuencias de las veinte imágenes; se puede visualizar un ejemplo de la matriz de frecuencias de la Imagen 1 en la Tabla 4.

	A	B	C	D	E	F	G	H	I	J
A	3,0	2,0	4,0	0,5	4,0	3,0	5,0	0,5	2,0	3,0
B	4,0	3,0	3,0	2,0	5,5	4,0	4,0	0,5	5,5	1,0
C	2,0	3,0	3,0	3,0	4,0	3,0	4,0	2,0	2,0	3,0
D	5,5	4,0	3,0	3,0	4,0	3,0	3,0	2,0	2,0	5,0
E	2,0	0,5	2,0	2,0	3,0	5,0	2,0	4,0	3,0	4,0
F	3,0	2,0	3,0	3,0	1,0	3,0	3,0	2,0	4,0	2,0
G	1,0	2,0	2,0	3,0	4,0	3,0	3,0	1,0	4,0	1,0
H	5,5	5,5	4,0	4,0	2,0	4,0	5,0	3,0	4,0	5,0
I	4,0	0,5	4,0	4,0	3,0	2,0	2,0	2,0	3,0	2,0
J	3,0	5,0	3,0	1,0	2,0	4,0	5,0	1,0	4,0	3,0

**Tabla 4: Matriz de frecuencias obtenida de las encuestas de la Imagen 1.**

Segundo, se generan las matrices de proporciones correspondientes a cada una de las matrices de frecuencias; el ejemplo de una matriz de proporciones perteneciente a la Imagen 1 se puede observar en la Tabla 5.

	A	B	C	D	E	F	G	H	I	J
A	0,5000	0,3333	0,6667	0,0833	0,6667	0,5000	0,8333	0,0833	0,3333	0,5000
B	0,6667	0,5000	0,5000	0,3333	0,9167	0,6667	0,6667	0,0833	0,9167	0,1667
C	0,3333	0,5000	0,5000	0,5000	0,6667	0,5000	0,6667	0,3333	0,3333	0,5000
D	0,9167	0,6667	0,5000	0,5000	0,6667	0,5000	0,5000	0,3333	0,3333	0,8333
E	0,3333	0,0833	0,3333	0,3333	0,5000	0,8333	0,3333	0,6667	0,5000	0,6667
F	0,5000	0,3333	0,5000	0,5000	0,1667	0,5000	0,5000	0,3333	0,6667	0,3333
G	0,1667	0,3333	0,3333	0,5000	0,6667	0,5000	0,5000	0,1667	0,6667	0,1667
H	0,9167	0,9167	0,6667	0,6667	0,3333	0,6667	0,8333	0,5000	0,6667	0,8333
I	0,6667	0,0833	0,6667	0,6667	0,5000	0,3333	0,3333	0,3333	0,5000	0,3333
J	0,5000	0,8333	0,5000	0,1667	0,3333	0,6667	0,8333	0,1667	0,6667	0,5000

Tabla 5: Matriz de proporciones referente a la Imagen 1.

Tercero, por cada matriz de proporciones se genera su respectiva matriz de puntuaciones típicas z; continuando con los ejemplos, se puede observar en la Tabla 6, la matriz de puntuaciones típicas z de la Imagen 1.

	A	B	C	D	E	F	G	H	I	J
A	0,0000	-0,4307	0,4307	-1,3830	0,4307	0,0000	0,9674	-1,3830	-0,4307	0,0000
B	0,4307	0,0000	0,0000	-0,4307	1,3830	0,4307	0,4307	-1,3830	1,3830	-0,9674
C	-0,4307	0,0000	0,0000	0,0000	0,4307	0,0000	0,4307	-0,4307	-0,4307	0,0000
D	1,3830	0,4307	0,0000	0,0000	0,4307	0,0000	0,0000	-0,4307	-0,4307	0,9674
E	-0,4307	-1,3830	-0,4307	-0,4307	0,0000	0,9674	-0,4307	0,4307	0,0000	0,4307
F	0,0000	-0,4307	0,0000	0,0000	-0,9674	0,0000	0,0000	-0,4307	0,4307	-0,4307
G	-0,9674	-0,4307	-0,4307	0,0000	0,4307	0,0000	0,0000	-0,9674	0,4307	-0,9674
H	1,3830	1,3830	0,4307	0,4307	-0,4307	0,4307	0,9674	0,0000	0,4307	0,9674
I	0,4307	-1,3830	0,4307	0,4307	0,0000	-0,4307	-0,4307	-0,4307	0,0000	-0,4307
J	0,0000	0,9674	0,0000	-0,9674	-0,4307	0,4307	0,9674	-0,9674	0,4307	0,0000

Tabla 6: Matriz de puntuaciones típicas z referente a la Imagen 1.

Cuarto, se promedian las puntuaciones típicas z, por matriz y por columna que indica el coeficiente utilizado, se ordenan de forma creciente dichos resultados (de menor preferencia a mayor preferencia en el rango de calidad obtenido, en donde el mejor resultado se encuentra a la derecha) y si es necesario se realiza una transformación lineal de la escala obtenida para desaparecer los valores negativos; en la Tabla 7 se pueden observar dichos promedios y su transformación lineal para los resultados de la Imagen 1.

H	D	B	J	C	E	A	I	F	G
0,0000	0,3643	0,4716	0,5562	0,6424	0,7270	0,7792	0,7807	0,7822	0,8895

Tabla 7: Promedios de las puntuaciones típicas z, transformados linealmente y ordenados de forma creciente de la Imagen 1.

Finalmente, se generan las gráficas de la escala obtenida por imagen para visualizar de forma más asimilable los índices (y órdenes) de preferencia que ocupa cada resultado al aplicar un coeficiente con respecto al resto. En la Figura 62 se puede observar la gráfica generada con los resultados escalares obtenidos para la Imagen 1, donde el mejor resultado ocupa el lado derecho y la separación entre los puntos indican la similitud perceptual; en este caso particular, la imagen en escala de grises generada a partir de la Imagen 1 a color con los coeficientes identificados con el par H ( $\theta$  individual y  $\phi$  manual) resulta clasificada con la peor escala perceptual, mientras que la imagen en escala de grises generada con los coeficientes identificados con el par G ( $\theta$  manual y  $\phi$  promedio) obtienen la mejor clasificación en la escala perceptual.

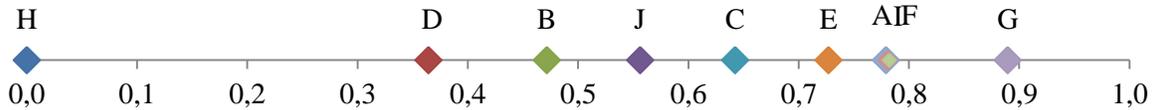


Figura 62: Gráfica de posiciones escalares que ocupan los resultados de las comparaciones realizadas del algoritmo propuesto y sus distintos coeficientes para la Imagen 1. El mejor resultado ocupa el lado derecho.

El resto de los resultados de las comparaciones por cada imagen evaluada se pueden observar en el Anexo 3.

La escala general de todos los resultados, calculada aplicando la misma metodología que en el caso anterior pero utilizando una matriz de frecuencias resultante de la suma de todas las matrices de frecuencias generadas por cada imagen (con un total de 120 evaluaciones por par de imágenes en escala de grises), se puede visualizar en la Figura 63, donde la mejor clasificación es obtenida utilizando el algoritmo de transformación propuesto con los coeficientes identificados como E ( $\theta$  manual y  $\phi$  manual).

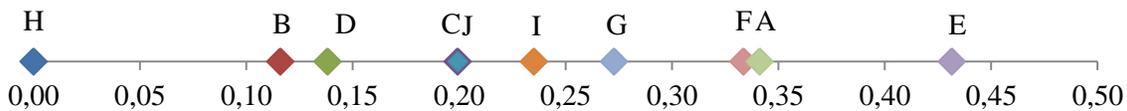


Figura 63: Gráfica de posiciones escalares generales que ocupan los resultados de las comparaciones realizadas del algoritmo propuesto y sus distintos parámetros para todas las imágenes.

Los cálculos para obtener la escala de calidad de cada uno de los resultados obtenidos al aplicar el algoritmo de transformación de color a escala de grises con sus respectivos coeficientes, se realizan con cada imagen a color por separado, para poder establecer el análisis de resultados de forma más acertada. Se puede observar, por los resultados de los índices de calidad obtenidos en la Figura 63, que hay un grado de discrepancia de coincidencia entre los resultados calculados de forma individual y el general.

### 6.3 Métrica C2G-SSIM

La evaluación a través de la métrica C2G-SSIM (descrita en la sección 2.2) es realizada para cada imagen a color y sus respectivas transformaciones en escala de grises (que llamaremos grupo). Para obtener la escala general de los índices de todas las imágenes se suman todos los resultados C2G-SSIM obtenidos por cada uno de los distintos tipos de coeficientes aplicados (Tabla 3) y se promedian entre el número de grupos evaluados (en total veinte para este trabajo).

Los resultados obtenidos permiten clasificar de forma ascendente o descendente las imágenes transformadas en escala de grises (en donde el mayor índice C2G-SSIM se traduce en una mejor calidad perceptual).

### 6.3.1 Resultados de la métrica C2G-SSIM

El resultado de los índices C2G-SSIM arrojados para la Imagen 1 se muestran en la Tabla 8. Los resultados de las veinte imágenes a color se pueden visualizar en el Anexo 4.

	<b>C2G-SSIM</b>
<b>Color vs. A</b>	0,93610
<b>Color vs. B</b>	0,92970
<b>Color vs. C</b>	0,93610
<b>Color vs. D</b>	0,93030
<b>Color vs. E</b>	0,93280
<b>Color vs. F</b>	0,93610
<b>Color vs. G</b>	0,93700
<b>Color vs. H</b>	0,91600
<b>Color vs. I</b>	0,93680
<b>Color vs. J</b>	0,93590

Tabla 8: Índices C2G-SSIM de la Imagen 1.

El índice general de C2G-SSIM obtenido mediante el promedio de todos los resultados obtenidos por cada grupo de imágenes se puede observar en la Tabla 9.

	<b>C2G-SSIM Promedio</b>
<b>Color vs. A</b>	0,879396
<b>Color vs. B</b>	0,876429
<b>Color vs. C</b>	0,880256
<b>Color vs. D</b>	0,876140
<b>Color vs. E</b>	0,884883
<b>Color vs. F</b>	0,878257
<b>Color vs. G</b>	0,878669
<b>Color vs. H</b>	0,867380
<b>Color vs. I</b>	0,877983
<b>Color vs. J</b>	0,881208

Tabla 9: Índices C2G-SSIM promedios.

Al igual que en la escala subjetiva, los resultados obtenidos con esta métrica objetiva es dependiente de la imagen de entrada y no del tipo de coeficiente utilizado. En la siguiente sección se comparan los resultados subjetivos y objetivos para determinar la relación que existe entre ambos resultados y así establecer las conclusiones al respecto.

## 6.4 Coeficientes de correlación

Los coeficientes de correlación (SRCC y KRCC), detallados en la sección 2.3, son aplicados entre los resultados obtenidos en las encuestas (sección 6.2) y los índices C2G-SSIM (sección anterior). Antes de aplicar el cálculo de los coeficientes de correlación, se le asigna a cada resultado obtenido en escala de grises (identificados con las letras desde la A hasta la J) correspondientes a cada grupo generado por una imagen a color, y por cada tipo de evaluación realizada (subjetiva y objetiva), un escalero del 1 al 10, que indica la posición de preferencia que ocupa cada uno de estos resultados, de forma descendente; donde 1 identifica los de mayor preferencia (es decir, mejor calidad perceptual) y 10 los de menor preferencia. Un ejemplo de esta asignación se puede observar en la Tabla 10, donde se muestran las posiciones de clasificación obtenidas para la “Imagen 1” por cada tipo de evaluación a través de las columnas nombradas “THURSTONE” y “C2G-SSIM” agrupadas en la columna “RANK”, que corresponden, respectivamente, a los resultados crudos obtenidos de los promedios de las puntuaciones típicas  $z$  y a los índices C2G-SSIM, agrupados en la columna “RAW”.

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
<b>Color vs. A</b>	0,7792	0,93610	4	3
<b>Color vs. B</b>	0,4716	0,92970	8	9
<b>Color vs. C</b>	0,6424	0,93610	6	4
<b>Color vs. D</b>	0,3643	0,93030	9	8
<b>Color vs. E</b>	0,7270	0,93280	5	7
<b>Color vs. F</b>	0,7822	0,93610	2	5
<b>Color vs. G</b>	0,8895	0,93700	1	1
<b>Color vs. H</b>	0,0000	0,91600	10	10
<b>Color vs. I</b>	0,7807	0,93680	3	2
<b>Color vs. J</b>	0,5562	0,93590	7	6

**Tabla 10: Posiciones de clasificación obtenidas para la Imagen 1 tanto para los resultados de las puntuaciones típicas  $z$  promediadas (“THURSTONE”) como de los índices C2G-SSIM.**

Una vez identificadas todas las posiciones de clasificación que ocupa cada resultado en escala de grises por cada imagen a color (ver el Anexo 5), se procedió a calcular los respectivos coeficientes de correlación.

En la Tabla 11 se muestran los resultados obtenidos junto con las transformaciones en escala de grises (identificadas con la letra correspondiente a los coeficientes de transformación utilizados y detallados en la Tabla 3) por imagen a color ordenadas según las respectivas clasificaciones obtenidas para los dos tipos de evaluaciones realizadas.

Para una mejor visualización de los resultados de los coeficientes de correlación obtenidos, su gráfica se muestra en la Figura 64, donde se puede detallar que para las imágenes 6, 11 y 17 la relación entre la evaluación subjetiva y objetiva es inversa o indistinguible, es decir, que hay una discrepancia muy alta entre los resultados subjetivos y objetivos para esos casos; que podría deberse, en caso de: la imagen 6, a su complejidad visual, con numerosos detalles y alta resolución; la imagen 11, por sus pocos colores y pocos detalles; la imagen 17 por su baja resolución y calidad de compresión para la complejidad de los detalles que presenta, originando confusión visual en la percepción de las líneas; lo que podría originar selecciones aleatorias por parte de los encuestados, impidiendo una relación directa con el cálculo de calidad objetivo de C2G-SSIM.

IMAGEN		SRCC	KRCC	Ordenando los resultados									
				mejor									peor
Imagen 1	(THURSTONE)	0,86667	0,73333	G	F	I	A	E	C	J	B	D	H
Imagen 1	(C2G-SSIM)			G	I	A	C	F	J	E	D	B	H
Imagen 2	(THURSTONE)	0,64848	0,46667	A	E	G	F	J	I	C	B	H	D
Imagen 2	(C2G-SSIM)			E	J	C	F	A	G	I	H	D	B
Imagen 3	(THURSTONE)	0,46667	0,33333	A	E	I	J	G	F	B	C	D	H
Imagen 3	(C2G-SSIM)			G	J	I	F	C	A	E	D	B	H
Imagen 4	(THURSTONE)	0,63636	0,42222	A	F	D	B	C	J	G	H	E	I
Imagen 4	(C2G-SSIM)			J	F	C	A	B	D	I	G	H	E
Imagen 5	(THURSTONE)	0,73333	0,6	C	G	I	J	A	E	F	D	B	H
Imagen 5	(C2G-SSIM)			E	G	I	J	C	A	F	D	H	B
Imagen 6	(THURSTONE)	-0,15152	-0,15556	F	E	D	A	G	C	B	I	J	H
Imagen 6	(C2G-SSIM)			J	G	I	A	C	F	D	B	E	H
Imagen 7	(THURSTONE)	0,93939	0,82222	E	J	C	A	F	B	D	H	G	I
Imagen 7	(C2G-SSIM)			E	J	F	C	A	D	B	H	I	G
Imagen 8	(THURSTONE)	0,75758	0,55556	E	G	I	A	J	F	H	B	D	C
Imagen 8	(C2G-SSIM)			E	I	G	J	F	C	A	D	B	H
Imagen 9	(THURSTONE)	0,6	0,46667	E	B	G	I	D	F	A	H	C	J
Imagen 9	(C2G-SSIM)			E	G	I	F	J	D	B	C	A	H
Imagen 10	(THURSTONE)	0,61212	0,46667	E	A	C	I	G	F	D	J	B	H
Imagen 10	(C2G-SSIM)			G	I	A	C	F	J	E	D	B	H
Imagen 11	(THURSTONE)	0,01216	0,02222	C	F	D	A	B	G	I	J	H	E
Imagen 11	(C2G-SSIM)			J	D	B	G	I	F	H	A	C	E
Imagen 12	(THURSTONE)	0,35758	0,28889	F	E	H	J	I	A	G	C	B	D
Imagen 12	(C2G-SSIM)			E	I	G	J	A	F	C	D	B	H
Imagen 13	(THURSTONE)	0,87879	0,73333	E	I	F	J	A	G	B	C	D	H
Imagen 13	(C2G-SSIM)			E	I	G	J	F	A	C	D	B	H
Imagen 14	(THURSTONE)	0,4303	0,33333	E	B	G	A	C	D	I	H	J	F
Imagen 14	(C2G-SSIM)			E	G	I	A	C	F	J	D	B	H
Imagen 15	(THURSTONE)	0,32121	0,28889	E	I	J	G	F	A	D	H	B	C
Imagen 15	(C2G-SSIM)			J	G	I	C	F	A	H	D	E	B
Imagen 16	(THURSTONE)	0,5183	0,46667	E	D	G	I	F	A	H	J	B	C
Imagen 16	(C2G-SSIM)			E	G	I	J	F	C	A	D	B	H
Imagen 17	(THURSTONE)	-0,62424	-0,46667	I	G	J	F	A	C	D	E	H	B
Imagen 17	(C2G-SSIM)			E	B	D	C	G	A	H	I	J	F
Imagen 18	(THURSTONE)	0,38182	0,24444	A	F	H	C	D	E	G	J	I	B
Imagen 18	(C2G-SSIM)			F	C	A	I	J	G	D	B	H	E
Imagen 19	(THURSTONE)	0,52888	0,42222	F	C	B	A	H	D	E	J	G	I
Imagen 19	(C2G-SSIM)			J	F	C	A	E	D	B	H	I	G
Imagen 20	(THURSTONE)	0,35758	0,24444	D	H	B	F	I	A	J	E	C	G
Imagen 20	(C2G-SSIM)			J	D	B	C	A	F	H	I	G	E

Tabla 11: Resultados de los coeficientes de correlación entre las evaluaciones subjetivas y objetivas, junto con las transformaciones en escala de grises por imagen a color ordenadas según las respectivas clasificaciones obtenidas.

La Tabla 11 pone en evidencia que ningún tipo de coeficiente aplicado al algoritmo de transformación propuesto arroja la misma calidad perceptual para todas las imágenes a color dadas; dependiendo de la imagen a color dada la clasificación de cada tipo de coeficiente varía al menos en una posición, por lo que es necesario analizar el comportamiento de los resultados para cada imagen a color con sus correspondientes transformaciones en escala de grises.

Se puede decir que las transformaciones en escala de grises obtenidas aplicando los coeficientes identificados como E (es decir,  $\bar{\theta}$  manual y  $\bar{\phi}$  manual), presentan el resultado con mayor calidad perceptual para las imágenes 2, 7, 8, 9, 12, 13, 14 y 16, mientras que presenta el caso opuesto (peor calidad perceptual) para la imagen 4; también se observa que para la imagen 15 se obtiene la mayor calidad perceptual en la métrica subjetiva y la peor calidad perceptual en la métrica objetiva.

Dentro de los resultados de los tipos de coeficientes automatizados se obtiene que:

El tipo de coeficiente identificado como A (es decir,  $\bar{\theta}$  promedio y  $\bar{\phi}$  promedio), nunca es clasificado con la peor calidad perceptual y obtiene las mejores clasificaciones para las evaluaciones subjetivas en las imágenes 2, 3, 4 y 18, y supera las clasificaciones de los tipos de coeficientes identificados como B y J, tanto para las evaluaciones subjetivas como objetivas, en las imágenes 1, 10 y 18.

El tipo de coeficiente identificado como B (es decir,  $\bar{\theta}$  individuales y  $\bar{\phi}$  individuales), a pesar de no obtener ninguna clasificación sobresaliente, para las imágenes 9 y 20 supera la obtenida por tipo de coeficiente identificado como A, tanto en las evaluaciones subjetivas como objetivas.

El tipo de coeficiente identificado como C (es decir,  $\bar{\theta}$  promedio y  $\bar{\phi}$  individuales) logra superar las clasificaciones obtenidas por el tipo de coeficiente identificado como A en las imágenes 5, 7 y 19.

En relación a la transformación identificada como J (canal L de CIELab), en las imágenes 7, 12, 13 y 15, supera las clasificaciones obtenidas por los tipos de coeficientes automatizados identificados como A, B, C y D, tanto en las evaluaciones subjetivas como objetivas.

El tipo de coeficiente identificado como H (es decir,  $\bar{\theta}$  individuales y  $\bar{\phi}$  manual) es el peor clasificado en prácticamente todas las imágenes menos en la evaluación subjetiva de la imagen 20.

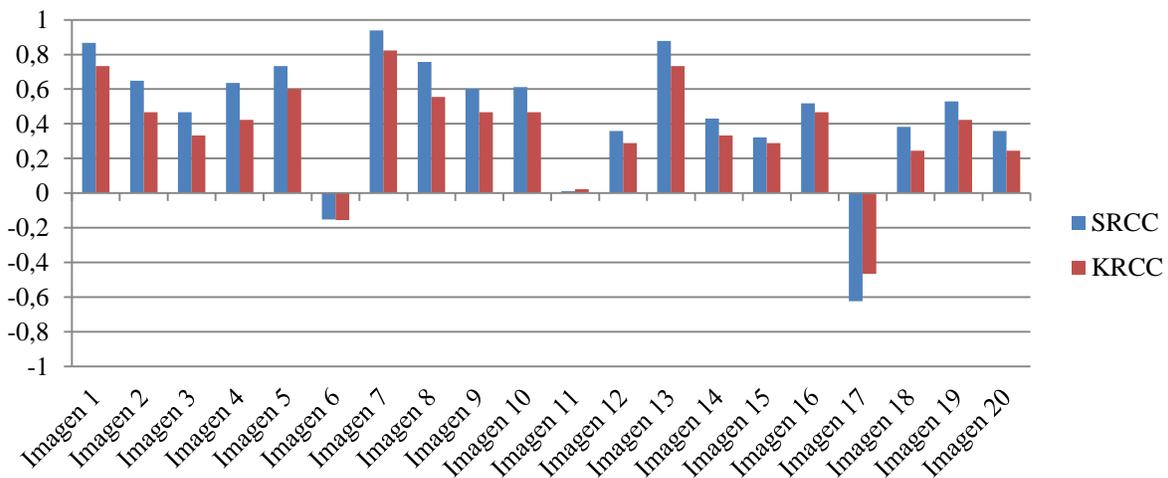


Figura 64: Coeficientes de correlación SRCC y KRCC para todas las imágenes evaluadas.

## 7 CONCLUSIONES Y TRABAJOS A FUTURO

En este trabajo se propone un algoritmo de transformación que utiliza distintos tipos de coeficientes para generar resultados en escala de grises con distintas variaciones en sus características. A través de las evaluaciones subjetivas (encuestas) y objetivas (algoritmo C2G-SSIM), aplicadas a una muestra de 20 imágenes a color con distintas características, se puede medir la calidad perceptual que se genera como resultado de la transformación, obteniendo que los coeficientes controlados de forma manual arrojan la máxima calidad perceptual, aunque para algunas imágenes esta selección no es la mejor, dado que el control manual puede representar un reto para el sujeto que manipula los valores, con gran consumo de tiempo y que la percepción subjetiva de esta sola persona no garantiza el mejor resultado para el resto de las personas que evalúan la imagen.

Dado que el coeficiente de correlación nunca alcanza su máximo valor (es decir, 1, donde ambas métricas de evaluación resultan exactamente iguales) pero se mantiene en la mayoría de los casos por encima de cero (existiendo una relación directa entre los resultados de las evaluaciones subjetivas y objetivas), se opta por seleccionar como resultado final de calidad perceptual, para cada tipo de coeficiente, las clasificaciones ocupadas por las posiciones escalares generales obtenidas de las evaluaciones subjetivas (graficadas en la Figura 63), dada la credibilidad que se le ha atribuido a este tipo de evaluaciones a lo largo del tiempo en trabajos similares.

Tomando los tipos de coeficientes automatizados (A, B, C y D) que forman parte de los objetivos específicos a alcanzar en este trabajo, se puede decir que A ( $\bar{\theta}$  promedio y  $\bar{\phi}$  promedio) logra obtener una calidad perceptual mayor al canal L del espacio CIELab (J), mientras que B ( $\bar{\theta}$  individuales y  $\bar{\phi}$  individuales), C ( $\bar{\theta}$  promedio y  $\bar{\phi}$  individuales) y D ( $\bar{\theta}$  individuales y  $\bar{\phi}$  promedio) no logran obtener una calidad perceptual mayor a la obtenida por J, ni por alguna combinación, aparte de A, que contenga algún valor de coeficiente controlado de forma manual, como F, G e I.

A pesar de que los controles manuales de los valores de los coeficientes no es la opción óptima al momento de transformar una imagen de color a escala de grises, el estudio de los valores seleccionados podría brindar pistas hacia una solución automatizada. Aparte, la FFT ofrece un espacio de trabajo que se puede aprovechar de muchas maneras en el área de edición de imágenes.

La optimización del código implementado, para manipulaciones en tiempo real en imágenes con alta resolución podría ser un trabajo a futuro, al igual que su comparación con otros algoritmos que forman parte del estado del arte de la transformación de imágenes de color a escala de grises. Del mismo modo, dado que uno de los enfoques de este trabajo es su alcance a través del software libre, se podría implementar el algoritmo propuesto como un plugin para alguna aplicación de edición de imágenes de uso común.

Para futuras evaluaciones se recomienda añadir, al repertorio de imágenes de prueba, algunas del grupo de imágenes más utilizadas por los trabajos de los algoritmos del estado del arte, para facilitar sus comparaciones y análisis de datos.

En relación al coeficiente  $\beta$ , incluido en el diseño e implementación del algoritmo, más no en las evaluaciones realizadas, por razones de tiempo, podría ser estudiado en otros trabajos.

El número de encuestados se recomienda que sea elevado para garantizar una mayor exactitud en los resultados, así como aplicar distintas metodologías para el descarte de respuestas no consistentes, que indiquen discrepancia en las selecciones realizadas por algún sujeto de prueba. Además se pueden ampliar las evaluaciones con muestras impresas, llevando los resultados del espacio RGB al CMYK.

## REFERENCIAS

- [1] Levkowitz Haim, *Color theory and modeling for computer graphics, visualization, and multimedia applications*, Springer, Ed. Massachusetts, USA: Kluwer Academic Publishers, 1997.
- [2] X-Rite, *The color guide and glossary*. USA, 1998.
- [3] Rebeca Azorín Montesinos, *Especificación cromática de gamas de colores usadas en la industria del calzado*. Alicante, España, 2003, Trabajo de investigación.
- [4] Ibraheem Noor, Hasan Mokhtar, Khan Rafiqul, and Mishra Pramod, *Understanding Color Models: A Review.*, 2012, vol. 2.
- [5] Antonio Valero Muñoz, *Principio de color y holopintura*, Editorial Club Universitario (ECU), Ed. Alicante, España, 2013.
- [6] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Tercera ed., Tom Robbins, Ed. Upper Saddle River, New Jersey, USA: Prentice Hall, 2007.
- [7] Konstantinos N. Plataniotis and Anastasios N. Venetsanopoulos, *Color Image Processing and Applications.*: Springer Science & Business Media, 2000.
- [8] Colin. Ware, *Information visualization - Perception for design*, Segunda ed. San Francisco, USA: Morgan Kaufmann, 2004.
- [9] Maureen C. Stone, William B. Cowan, and John C. Beatty, *Color Gamut Mapping and the Printing of Digital Color Images.*: ACM, Octubre 1988, vol. 7.
- [10] Gaurav Sharma, *Digital Color Imaging Handbook*. New York: CRC Press, 2003.
- [11] Mark D. Fairchild, *Color Appearance Models*, Segunda Edición ed. USA, 2005.
- [12] Christopher Kanan and Garrison W. Cottrell, *Color-to-Grayscale: Does the Method Matter in Image Recognition?*, Eshel Ben-Jacob, Ed. California, USA, Enero 2012, vol. 7.
- [13] Ye Zhao and Zakiya Tamimi, *Spectral Image Decolorization*. Las Vegas, NV, USA, Diciembre 2010, vol. 6454.
- [14] Raman Gupta, Dipti Bansal, and Charanjit Singh, *A Survey on various objective Image Quality Assessment Techniques*. India, 2014, vol. 2.
- [15] Emin Zerman, Vedad Hulusic, Giuseppe Valenzise, Rafal Mantiuk, and Frédéric Dufaux , *The Relation Between MOS and Pairwise Comparisons and the Importance of Cross-Content Comparisons.*, 2018.
- [16] María Pérez-Ortiz and Rafal Mantiuk, *A practical guide and software for analysing pairwise comparison experiments*. Reino Unido, 2017.

- [17] Thomas C. Brown and George L. Peterson, *An Enquiry Into the Method of Paired Comparison*. Estados Unidos, 2009.
- [18] Martin Čadík, *Perceptual Evaluation of Color-to-Grayscale Image Conversions*, T. Igarashi, N. Max, and F. Sillion, Eds. USA: Blackwell Publishing, Octubre 2008, vol. 27.
- [19] Louis Leon Thurstone , *A law of comparative judgment*. Chicago, 1927.
- [20] Fiorenzo Franceschini and Domenico A. Maisano, *Adapting Thurstone's Law of Comparative Judgment to fuse preference orderings in manufacturing applications.*, 2018.
- [21] Kristi Tsukida and Maya R. Gupta, *How to Analyze Paired Comparison Data.*, 2011.
- [22] Tom Bramley, *Paired comparison methods.*, 2008.
- [23] Frederick Mosteller, *Remarks on the method of paired comparisons: I. The least squares solution assuming equal standard deviations and equal correlations.*, 1951.
- [24] Kede Ma, Tiesong Zhao, Kai Zeng, and Zhou Wang, *Objective Quality Assessment for Color-to-Gray Image Conversion.*, 2015.
- [25] Raja Bala and Reiner Eschbach, *Spatial Color-to-Grayscale Transform Preserving Chrominance Edge Information.*, 2004.
- [26] Karl Rasche, Robert Geist, and James Westall, *Re-coloring Images for Gamuts of Lower Dimension*, 2005, Clemson University.
- [27] Amy A. Gooch, Sven C. Olsen, Jack Tumblin, and Bruce Gooch, *Color2Gray: Saliency-Preserving Color Removal*. New York, NY, USA: ACM, 2005, vol. 24, Northwestern University.
- [28] Yongjin Kim, Cheolhun Jang, Julien Demouth, and Seungyong Lee, *Robust Color-to-gray via Nonlinear Global Mapping*. New York, USA, Diciembre 2009, vol. 28.
- [29] Mark Grundland and Neil A. Dodgson, *The decolorize algorithm for contrast enhancing, color to grayscale conversion*. UK, Octubre 2005, University of Cambridge.
- [30] P. Legendre and Loic F Legendre, *Numerical Ecology.*, 1998, vol. 24.

# ANEXOS

Imagen 1			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	-0,0021	-0,024	275,827
B	Ind	Ind	274,987
C	-0,0021	Ind	276,208
D	Ind	-0,024	276,617
E	-0,08	4,5	252,062
F	-0,0021	-2,5002	270,834
G	-0,09	-0,024	270,54
H	Ind	-2,5002	240,943
I	-0,09	Ind	277,903
J	L	L	—

Imagen 2			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0006	-0,065	0,315
B	Ind	Ind	0,314
C	0,0006	Ind	0,296
D	Ind	-0,065	0,305
E	-0,03	5	0,281
F	0,0006	0	0,305
G	-0,03	-0,065	0,308
H	Ind	0	0,314
I	-0,03	Ind	0,308
J	L	L	—

Imagen 3			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0291	-0,0155	16,407
B	Ind	Ind	16,291
C	0,0291	Ind	16,226
D	Ind	-0,0155	16,312
E	-0,18	1,2501	15,346
F	0,0291	-1,5	16,677
G	-0,09	-0,0155	16,23
H	Ind	-1,5	17,172
I	-0,09	Ind	16,381
J	L	L	—

Imagen 4			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	-0,0591	-0,01	19,425
B	Ind	Ind	19,361
C	-0,0591	Ind	19,428
D	Ind	-0,01	19,349
E	-0,15	0,25	17,738
F	-0,0591	5	19,378
G	-0,09	-0,01	19,344
H	Ind	5	19,399
I	-0,09	Ind	19,271
J	L	L	—

Imagen 5			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0081	-0,0115	2,996
B	Ind	Ind	3,1
C	0,0081	Ind	3,037
D	Ind	-0,0115	3,009
E	-0,12	0,9	3,035
F	0,0081	0,8	2,957
G	-0,06	-0,0115	3,05
H	Ind	0,8	2,911
I	-0,06	Ind	3,004
J	L	L	—

Imagen 6			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	-0,0735	-0,0075	108,304
B	Ind	Ind	107,574
C	-0,0735	Ind	111,801
D	Ind	-0,0075	107,064
E	-0,09	5	102,715
F	-0,0735	3,5	103,947
G	-0,04	-0,0075	103,927
H	Ind	3,5	113,689
I	-0,04	Ind	106,979
J	L	L	—

Imagen 7			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,006	-0,1155	71,55
B	Ind	Ind	69,22
C	0,006	Ind	71,422
D	Ind	-0,1155	70,636
E	-0,06	4,5	65,508
F	0,006	-1,5	71,786
G	0,12	-0,1155	70,286
H	Ind	-1,5	75,12
I	0,12	Ind	72,774
J	L	L	—

Imagen 8			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0177	0,0775	18,14
B	Ind	Ind	16,571
C	0,0177	Ind	16,482
D	Ind	0,0775	16,454
E	-0,09	2,5	15,189
F	0,0177	-2	16,356
G	-0,03	0,0775	16,361
H	Ind	-2	16,489
I	-0,03	Ind	16,357
J	L	L	—

Imagen 9			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0687	-0,102	14,386
B	Ind	Ind	14,495
C	0,0687	Ind	14,452
D	Ind	-0,102	14,09
E	-0,18	4,5	12,797
F	0,0687	-1,5	14,308
G	-0,06	-0,102	14,287
H	Ind	-1,5	14,007
I	-0,06	Ind	14,156
J	L	L	—

Anexo 1: Tablas de valores de coeficientes utilizados por imagen.

Imagen 10			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	-0,0105	-0,135	0,615
B	Ind	Ind	0,599
C	-0,0105	Ind	0,591
D	Ind	-0,135	0,603
E	-0,09	4	0,56
F	-0,0105	-2	0,59
G	-0,06	-0,135	0,596
H	Ind	-2	0,603
I	-0,06	Ind	0,587
J	L	L	—

Imagen 11			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	-0,0718	-0,0294	0,595
B	Ind	Ind	0,595
C	-0,0718	Ind	0,589
D	Ind	-0,0294	0,588
E	0,09	-5	0,562
F	-0,0718	5	0,61
G	-0,06	-0,0294	0,597
H	Ind	5	0,602
I	-0,06	Ind	0,601
J	L	L	—

Imagen 12			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0255	0,002	2,231
B	Ind	Ind	2,226
C	0,0255	Ind	2,282
D	Ind	0,002	2,333
E	-0,06	3,6	2,214
F	0,0255	-1,5	2,289
G	-0,05	0,002	2,278
H	Ind	-1,5	2,279
I	-0,05	Ind	2,257
J	L	L	—

Imagen 13			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0222	-0,0935	1,897
B	Ind	Ind	1,906
C	0,0222	Ind	1,917
D	Ind	-0,0935	1,916
E	-0,06	5	1,717
F	0,0222	-1	1,888
G	-0,04	-0,0935	1,882
H	Ind	-1	1,884
I	-0,04	Ind	1,902
J	L	L	—

Imagen 14			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	-0,0081	-0,0735	16,053
B	Ind	Ind	15,629
C	-0,0081	Ind	15,497
D	Ind	-0,0735	15,639
E	-0,03	4	14,961
F	-0,0081	-1,5	15,6
G	-0,09	-0,0735	15,58
H	Ind	-1,5	15,516
I	-0,09	Ind	15,53
J	L	L	—

Imagen 15			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0102	-0,092	13,531
B	Ind	Ind	12,98
C	0,0102	Ind	12,641
D	Ind	-0,092	12,547
E	-0,15	-0,5	12,477
F	0,0102	1,5	12,709
G	-0,09	-0,092	13,28
H	Ind	1,5	13,062
I	-0,09	Ind	12,929
J	L	L	—

Imagen 16			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0219	0,013	9,945
B	Ind	Ind	9,23
C	0,0219	Ind	9,13
D	Ind	0,013	9,192
E	-0,06	2,6	9,207
F	0,0219	-1,5	9,224
G	-0,03	0,013	9,172
H	Ind	-1,5	9,157
I	-0,03	Ind	9,086
J	L	L	—

Imagen 17			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,222	0,011	0,177
B	Ind	Ind	0,183
C	0,222	Ind	0,203
D	Ind	0,011	0,172
E	0,06	-5	0,165
F	0,222	0,2	0,18
G	-0,21	0,011	0,182
H	Ind	0,2	0,174
I	-0,21	Ind	0,175
J	L	L	—

Imagen 18			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	-0,0138	-0,075	21,749
B	Ind	Ind	21,676
C	-0,0138	Ind	22,139
D	Ind	-0,075	21,478
E	0,09	-2,5	22,85
F	-0,0138	1,5	21,618
G	0,03	-0,075	23,877
H	Ind	1,5	21,796
I	0,03	Ind	21,851
J	L	L	—

Imagen 19			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0105	-0,047	1,509
B	Ind	Ind	1,469
C	0,0105	Ind	1,484
D	Ind	-0,047	1,473
E	0,07	-5	1,523
F	0,0105	-1,5	1,467
G	0,1	-0,047	1,445
H	Ind	-1,5	1,544
I	0,1	Ind	1,47
J	L	L	—

Imagen 20			
	TITA $\theta$	PHI $\phi$	Tiempo seg
A	0,0228	-0,077	1,294
B	Ind	Ind	1,235
C	0,0228	Ind	1,254
D	Ind	-0,077	1,229
E	0,16	3	1,245
F	0,0228	3	1,247
G	0,15	-0,077	1,258
H	Ind	3	1,256
I	0,15	Ind	1,244
J	L	L	—

Anexo 1 (Cont.): Tablas de valores de coeficientes utilizados por imagen.



Evaluación de los resultados obtenidos de las conversiones

Nombre: \_\_\_\_\_ Edad: \_\_\_\_\_ Sexo:  F  M

Deficiencias visuales: \_\_\_\_\_

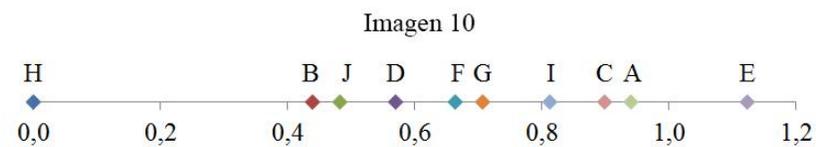
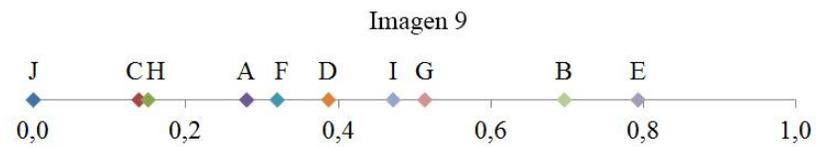
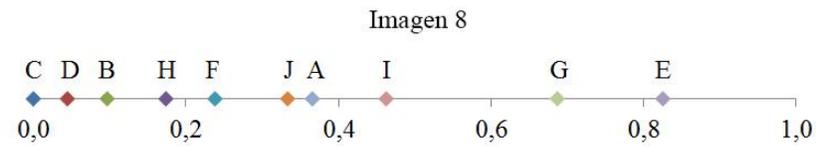
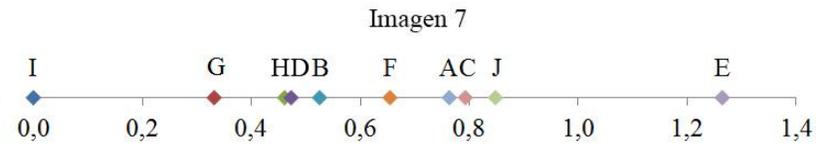
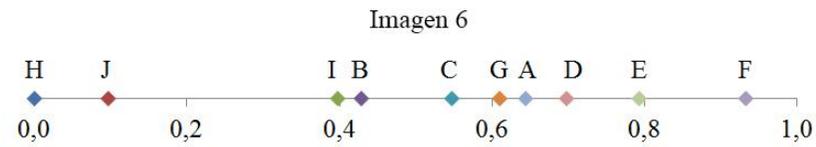
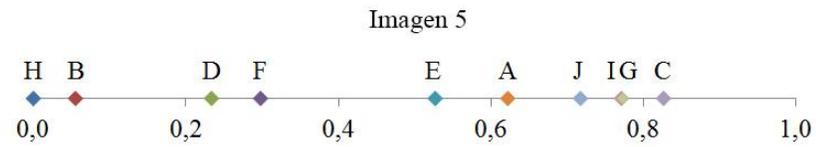
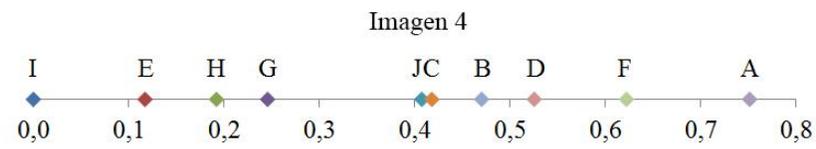
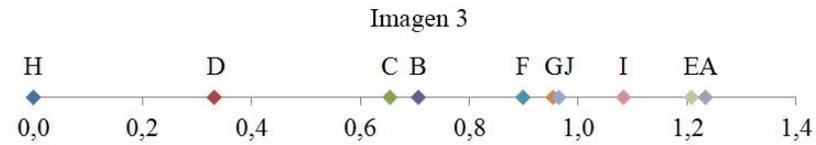
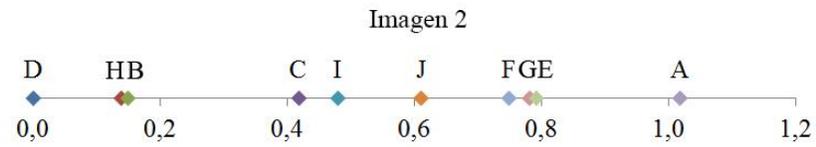
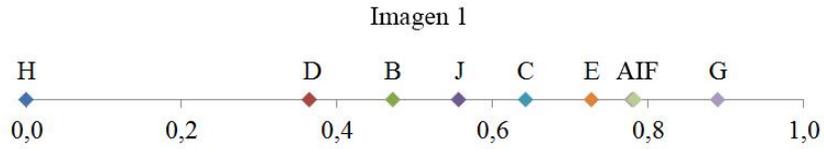
Tipo de prueba:  Precisión (con referencia a color)

Preferencia

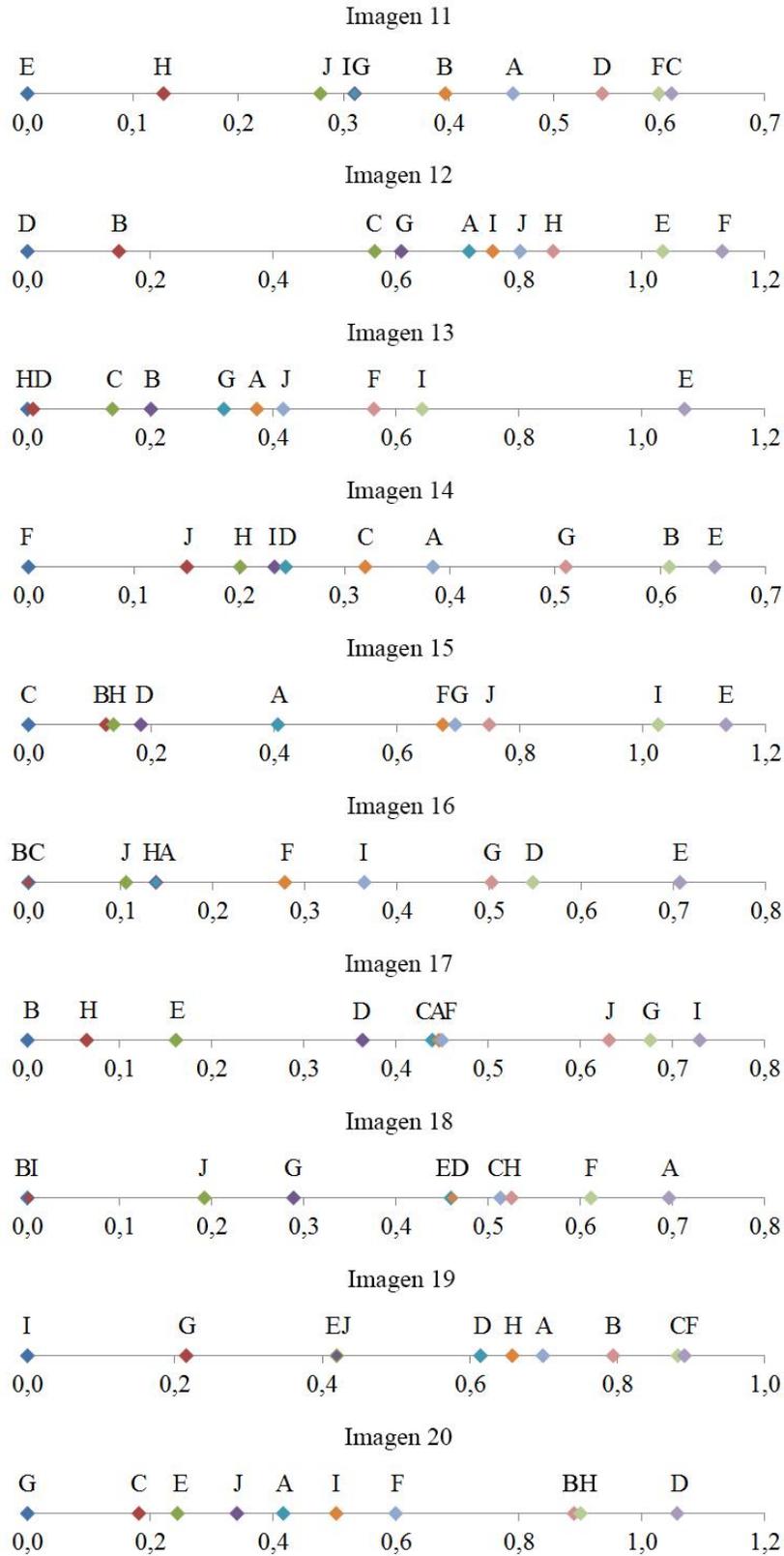
**Actividad:** Rellene la casilla con una X donde la letra coincida con la imagen que seleccione como favorita. No se permite marcar ambas opciones, debe seleccionar solo una opción.

Ref.	Par:	Ref.	Par:	Ref.	Par:	Ref.	Par:
1r	A <input type="checkbox"/> <input type="checkbox"/> B	6r	A <input type="checkbox"/> <input type="checkbox"/> B	11r	A <input type="checkbox"/> <input type="checkbox"/> B	16r	A <input type="checkbox"/> <input type="checkbox"/> B
1r	B <input type="checkbox"/> <input type="checkbox"/> C	6r	B <input type="checkbox"/> <input type="checkbox"/> C	11r	B <input type="checkbox"/> <input type="checkbox"/> C	16r	B <input type="checkbox"/> <input type="checkbox"/> C
1r	C <input type="checkbox"/> <input type="checkbox"/> D	6r	C <input type="checkbox"/> <input type="checkbox"/> D	11r	C <input type="checkbox"/> <input type="checkbox"/> D	16r	C <input type="checkbox"/> <input type="checkbox"/> D
1r	D <input type="checkbox"/> <input type="checkbox"/> E	6r	D <input type="checkbox"/> <input type="checkbox"/> E	11r	D <input type="checkbox"/> <input type="checkbox"/> E	16r	D <input type="checkbox"/> <input type="checkbox"/> E
1r	E <input type="checkbox"/> <input type="checkbox"/> F	6r	E <input type="checkbox"/> <input type="checkbox"/> F	11r	E <input type="checkbox"/> <input type="checkbox"/> F	16r	E <input type="checkbox"/> <input type="checkbox"/> F
1r	F <input type="checkbox"/> <input type="checkbox"/> G	6r	F <input type="checkbox"/> <input type="checkbox"/> G	11r	F <input type="checkbox"/> <input type="checkbox"/> G	16r	F <input type="checkbox"/> <input type="checkbox"/> G
1r	G <input type="checkbox"/> <input type="checkbox"/> H	6r	G <input type="checkbox"/> <input type="checkbox"/> H	11r	G <input type="checkbox"/> <input type="checkbox"/> H	16r	G <input type="checkbox"/> <input type="checkbox"/> H
1r	H <input type="checkbox"/> <input type="checkbox"/> I	6r	H <input type="checkbox"/> <input type="checkbox"/> I	11r	H <input type="checkbox"/> <input type="checkbox"/> I	16r	H <input type="checkbox"/> <input type="checkbox"/> I
1r	I <input type="checkbox"/> <input type="checkbox"/> J	6r	I <input type="checkbox"/> <input type="checkbox"/> J	11r	I <input type="checkbox"/> <input type="checkbox"/> J	16r	I <input type="checkbox"/> <input type="checkbox"/> J
2r	A <input type="checkbox"/> <input type="checkbox"/> C	7r	A <input type="checkbox"/> <input type="checkbox"/> C	12r	A <input type="checkbox"/> <input type="checkbox"/> C	17r	A <input type="checkbox"/> <input type="checkbox"/> C
2r	B <input type="checkbox"/> <input type="checkbox"/> D	7r	B <input type="checkbox"/> <input type="checkbox"/> D	12r	B <input type="checkbox"/> <input type="checkbox"/> D	17r	B <input type="checkbox"/> <input type="checkbox"/> D
2r	B <input type="checkbox"/> <input type="checkbox"/> E	7r	B <input type="checkbox"/> <input type="checkbox"/> E	12r	B <input type="checkbox"/> <input type="checkbox"/> E	17r	B <input type="checkbox"/> <input type="checkbox"/> E
2r	C <input type="checkbox"/> <input type="checkbox"/> E	7r	C <input type="checkbox"/> <input type="checkbox"/> E	12r	C <input type="checkbox"/> <input type="checkbox"/> E	17r	C <input type="checkbox"/> <input type="checkbox"/> E
2r	D <input type="checkbox"/> <input type="checkbox"/> F	7r	D <input type="checkbox"/> <input type="checkbox"/> F	12r	D <input type="checkbox"/> <input type="checkbox"/> F	17r	D <input type="checkbox"/> <input type="checkbox"/> F
2r	E <input type="checkbox"/> <input type="checkbox"/> G	7r	E <input type="checkbox"/> <input type="checkbox"/> G	12r	E <input type="checkbox"/> <input type="checkbox"/> G	17r	E <input type="checkbox"/> <input type="checkbox"/> G
2r	F <input type="checkbox"/> <input type="checkbox"/> H	7r	F <input type="checkbox"/> <input type="checkbox"/> H	12r	F <input type="checkbox"/> <input type="checkbox"/> H	17r	F <input type="checkbox"/> <input type="checkbox"/> H
2r	G <input type="checkbox"/> <input type="checkbox"/> I	7r	G <input type="checkbox"/> <input type="checkbox"/> I	12r	G <input type="checkbox"/> <input type="checkbox"/> I	17r	G <input type="checkbox"/> <input type="checkbox"/> I
2r	H <input type="checkbox"/> <input type="checkbox"/> J	7r	H <input type="checkbox"/> <input type="checkbox"/> J	12r	H <input type="checkbox"/> <input type="checkbox"/> J	17r	H <input type="checkbox"/> <input type="checkbox"/> J
3r	A <input type="checkbox"/> <input type="checkbox"/> D	8r	A <input type="checkbox"/> <input type="checkbox"/> D	13r	A <input type="checkbox"/> <input type="checkbox"/> D	18r	A <input type="checkbox"/> <input type="checkbox"/> D
3r	A <input type="checkbox"/> <input type="checkbox"/> E	8r	A <input type="checkbox"/> <input type="checkbox"/> E	13r	A <input type="checkbox"/> <input type="checkbox"/> E	18r	A <input type="checkbox"/> <input type="checkbox"/> E
3r	B <input type="checkbox"/> <input type="checkbox"/> F	8r	B <input type="checkbox"/> <input type="checkbox"/> F	13r	B <input type="checkbox"/> <input type="checkbox"/> F	18r	B <input type="checkbox"/> <input type="checkbox"/> F
3r	B <input type="checkbox"/> <input type="checkbox"/> G	8r	B <input type="checkbox"/> <input type="checkbox"/> G	13r	B <input type="checkbox"/> <input type="checkbox"/> G	18r	B <input type="checkbox"/> <input type="checkbox"/> G
3r	C <input type="checkbox"/> <input type="checkbox"/> F	8r	C <input type="checkbox"/> <input type="checkbox"/> F	13r	C <input type="checkbox"/> <input type="checkbox"/> F	18r	C <input type="checkbox"/> <input type="checkbox"/> F
3r	C <input type="checkbox"/> <input type="checkbox"/> G	8r	C <input type="checkbox"/> <input type="checkbox"/> G	13r	C <input type="checkbox"/> <input type="checkbox"/> G	18r	C <input type="checkbox"/> <input type="checkbox"/> G
3r	D <input type="checkbox"/> <input type="checkbox"/> G	8r	D <input type="checkbox"/> <input type="checkbox"/> G	13r	D <input type="checkbox"/> <input type="checkbox"/> G	18r	D <input type="checkbox"/> <input type="checkbox"/> G
3r	E <input type="checkbox"/> <input type="checkbox"/> H	8r	E <input type="checkbox"/> <input type="checkbox"/> H	13r	E <input type="checkbox"/> <input type="checkbox"/> H	18r	E <input type="checkbox"/> <input type="checkbox"/> H
3r	F <input type="checkbox"/> <input type="checkbox"/> I	8r	F <input type="checkbox"/> <input type="checkbox"/> I	13r	F <input type="checkbox"/> <input type="checkbox"/> I	18r	F <input type="checkbox"/> <input type="checkbox"/> I
3r	G <input type="checkbox"/> <input type="checkbox"/> J	8r	G <input type="checkbox"/> <input type="checkbox"/> J	13r	G <input type="checkbox"/> <input type="checkbox"/> J	18r	G <input type="checkbox"/> <input type="checkbox"/> J
4r	A <input type="checkbox"/> <input type="checkbox"/> F	9r	A <input type="checkbox"/> <input type="checkbox"/> F	14r	A <input type="checkbox"/> <input type="checkbox"/> F	19r	A <input type="checkbox"/> <input type="checkbox"/> F
4r	A <input type="checkbox"/> <input type="checkbox"/> G	9r	A <input type="checkbox"/> <input type="checkbox"/> G	14r	A <input type="checkbox"/> <input type="checkbox"/> G	19r	A <input type="checkbox"/> <input type="checkbox"/> G
4r	B <input type="checkbox"/> <input type="checkbox"/> H	9r	B <input type="checkbox"/> <input type="checkbox"/> H	14r	B <input type="checkbox"/> <input type="checkbox"/> H	19r	B <input type="checkbox"/> <input type="checkbox"/> H
4r	B <input type="checkbox"/> <input type="checkbox"/> I	9r	B <input type="checkbox"/> <input type="checkbox"/> I	14r	B <input type="checkbox"/> <input type="checkbox"/> I	19r	B <input type="checkbox"/> <input type="checkbox"/> I
4r	C <input type="checkbox"/> <input type="checkbox"/> G	9r	C <input type="checkbox"/> <input type="checkbox"/> G	14r	C <input type="checkbox"/> <input type="checkbox"/> G	19r	C <input type="checkbox"/> <input type="checkbox"/> G
4r	C <input type="checkbox"/> <input type="checkbox"/> H	9r	C <input type="checkbox"/> <input type="checkbox"/> H	14r	C <input type="checkbox"/> <input type="checkbox"/> H	19r	C <input type="checkbox"/> <input type="checkbox"/> H
4r	D <input type="checkbox"/> <input type="checkbox"/> H	9r	D <input type="checkbox"/> <input type="checkbox"/> H	14r	D <input type="checkbox"/> <input type="checkbox"/> H	19r	D <input type="checkbox"/> <input type="checkbox"/> H
4r	E <input type="checkbox"/> <input type="checkbox"/> I	9r	E <input type="checkbox"/> <input type="checkbox"/> I	14r	E <input type="checkbox"/> <input type="checkbox"/> I	19r	E <input type="checkbox"/> <input type="checkbox"/> I
4r	F <input type="checkbox"/> <input type="checkbox"/> J	9r	F <input type="checkbox"/> <input type="checkbox"/> J	14r	F <input type="checkbox"/> <input type="checkbox"/> J	19r	F <input type="checkbox"/> <input type="checkbox"/> J
5r	A <input type="checkbox"/> <input type="checkbox"/> H	10r	A <input type="checkbox"/> <input type="checkbox"/> H	15r	A <input type="checkbox"/> <input type="checkbox"/> H	20r	A <input type="checkbox"/> <input type="checkbox"/> H
5r	A <input type="checkbox"/> <input type="checkbox"/> I	10r	A <input type="checkbox"/> <input type="checkbox"/> I	15r	A <input type="checkbox"/> <input type="checkbox"/> I	20r	A <input type="checkbox"/> <input type="checkbox"/> I
5r	A <input type="checkbox"/> <input type="checkbox"/> J	10r	A <input type="checkbox"/> <input type="checkbox"/> J	15r	A <input type="checkbox"/> <input type="checkbox"/> J	20r	A <input type="checkbox"/> <input type="checkbox"/> J
5r	B <input type="checkbox"/> <input type="checkbox"/> J	10r	B <input type="checkbox"/> <input type="checkbox"/> J	15r	B <input type="checkbox"/> <input type="checkbox"/> J	20r	B <input type="checkbox"/> <input type="checkbox"/> J
5r	C <input type="checkbox"/> <input type="checkbox"/> I	10r	C <input type="checkbox"/> <input type="checkbox"/> I	15r	C <input type="checkbox"/> <input type="checkbox"/> I	20r	C <input type="checkbox"/> <input type="checkbox"/> I
5r	C <input type="checkbox"/> <input type="checkbox"/> J	10r	C <input type="checkbox"/> <input type="checkbox"/> J	15r	C <input type="checkbox"/> <input type="checkbox"/> J	20r	C <input type="checkbox"/> <input type="checkbox"/> J
5r	D <input type="checkbox"/> <input type="checkbox"/> I	10r	D <input type="checkbox"/> <input type="checkbox"/> I	15r	D <input type="checkbox"/> <input type="checkbox"/> I	20r	D <input type="checkbox"/> <input type="checkbox"/> I
5r	D <input type="checkbox"/> <input type="checkbox"/> J	10r	D <input type="checkbox"/> <input type="checkbox"/> J	15r	D <input type="checkbox"/> <input type="checkbox"/> J	20r	D <input type="checkbox"/> <input type="checkbox"/> J
5r	E <input type="checkbox"/> <input type="checkbox"/> J	10r	E <input type="checkbox"/> <input type="checkbox"/> J	15r	E <input type="checkbox"/> <input type="checkbox"/> J	20r	E <input type="checkbox"/> <input type="checkbox"/> J

Anexo 2: Modelo de encuesta a rellenar por cada sujeto de prueba.



Anexo 3: Gráficas de posiciones escalares que ocupan los resultados de las comparaciones realizadas del algoritmo propuesto y sus distintos coeficientes para todas las imágenes. El mejor resultado ocupa el lado derecho.



Anexo 3 (Cont.): Gráficas de posiciones escalares que ocupan los resultados de las comparaciones realizadas del algoritmo propuesto y sus distintos coeficientes para todas las imágenes. El mejor resultado ocupa el lado derecho.

Imagen 1

	C2G-SSIM	RANK-C2G
Color vs. A	0,93610	3
Color vs. B	0,92970	9
Color vs. C	0,93610	4
Color vs. D	0,93030	8
Color vs. E	0,93280	7
Color vs. F	0,93610	5
Color vs. G	0,93700	1
Color vs. H	0,91600	10
Color vs. I	0,93680	2
Color vs. J	0,93590	6

Imagen 2

	C2G-SSIM	RANK-C2G
Color vs. A	0,851329	5
Color vs. B	0,792725	10
Color vs. C	0,851381	3
Color vs. D	0,798939	9
Color vs. E	0,860448	1
Color vs. F	0,851363	4
Color vs. G	0,850282	6
Color vs. H	0,800079	8
Color vs. I	0,849697	7
Color vs. J	0,851425	2

Imagen 3

	C2G-SSIM	RANK-C2G
Color vs. A	0,959408	6
Color vs. B	0,956854	9
Color vs. C	0,959488	5
Color vs. D	0,957364	8
Color vs. E	0,958745	7
Color vs. F	0,959595	4
Color vs. G	0,960095	1
Color vs. H	0,954408	10
Color vs. I	0,959762	3
Color vs. J	0,959902	2

Imagen 4

	C2G-SSIM	RANK-C2G
Color vs. A	0,976076	4
Color vs. B	0,971644	5
Color vs. C	0,976619	3
Color vs. D	0,971275	6
Color vs. E	0,944600	10
Color vs. F	0,976622	2
Color vs. G	0,964453	8
Color vs. H	0,963984	9
Color vs. I	0,965114	7
Color vs. J	0,987817	1

Imagen 5

	C2G-SSIM	RANK-C2G
Color vs. A	0,745675	6
Color vs. B	0,730271	10
Color vs. C	0,745803	5
Color vs. D	0,734361	8
Color vs. E	0,749140	1
Color vs. F	0,745411	7
Color vs. G	0,747658	2
Color vs. H	0,731745	9
Color vs. I	0,746857	3
Color vs. J	0,746097	4

Imagen 6

	C2G-SSIM	RANK-C2G
Color vs. A	0,950481	4
Color vs. B	0,946292	8
Color vs. C	0,950478	5
Color vs. D	0,946810	7
Color vs. E	0,940077	9
Color vs. F	0,947355	6
Color vs. G	0,955685	2
Color vs. H	0,937027	10
Color vs. I	0,955649	3
Color vs. J	0,956921	1

Imagen 7

	C2G-SSIM	RANK-C2G
Color vs. A	0,864893	5
Color vs. B	0,859721	7
Color vs. C	0,864919	4
Color vs. D	0,860976	6
Color vs. E	0,877765	1
Color vs. F	0,865178	3
Color vs. G	0,839146	10
Color vs. H	0,848622	8
Color vs. I	0,839342	9
Color vs. J	0,865580	2

Imagen 8

	C2G-SSIM	RANK-C2G
Color vs. A	0,883792	7
Color vs. B	0,870974	9
Color vs. C	0,883830	6
Color vs. D	0,872120	8
Color vs. E	0,893350	1
Color vs. F	0,884233	5
Color vs. G	0,890919	3
Color vs. H	0,863625	10
Color vs. I	0,890936	2
Color vs. J	0,886943	4

Imagen 9

	C2G-SSIM	RANK-C2G
Color vs. A	0,777124	9
Color vs. B	0,781574	7
Color vs. C	0,777588	8
Color vs. D	0,782617	6
Color vs. E	0,820261	1
Color vs. F	0,784303	4
Color vs. G	0,785478	2
Color vs. H	0,775109	10
Color vs. I	0,785004	3
Color vs. J	0,782756	5

Imagen 10

	C2G-SSIM	RANK-C2G
Color vs. A	0,924777	3
Color vs. B	0,918315	9
Color vs. C	0,924764	4
Color vs. D	0,918955	8
Color vs. E	0,919112	7
Color vs. F	0,924686	5
Color vs. G	0,924994	1
Color vs. H	0,908705	10
Color vs. I	0,924779	2
Color vs. J	0,924013	6

Imagen 11

	C2G-SSIM	RANK-C2G
Color vs. A	0,974176	8
Color vs. B	0,979527	3
Color vs. C	0,973850	9
Color vs. D	0,979814	2
Color vs. E	0,967925	10
Color vs. F	0,976861	6
Color vs. G	0,977467	4
Color vs. H	0,975285	7
Color vs. I	0,977253	5
Color vs. J	0,985030	1

Imagen 12

	C2G-SSIM	RANK-C2G
Color vs. A	0,922875	5
Color vs. B	0,915855	9
Color vs. C	0,922838	7
Color vs. D	0,916774	8
Color vs. E	0,926848	1
Color vs. F	0,922851	6
Color vs. G	0,926579	3
Color vs. H	0,913188	10
Color vs. I	0,926691	2
Color vs. J	0,924736	4

Anexo 4: Índices C2G-SSIM de cada una de las transformaciones en escala de grises para cada imagen a color.

Imagen 13

	C2G-SSIM	RANK-C2G
Color vs. A	0,752060	6
Color vs. B	0,739681	9
Color vs. C	0,751979	7
Color vs. D	0,741043	8
Color vs. E	0,762321	1
Color vs. F	0,752228	5
Color vs. G	0,760955	3
Color vs. H	0,737300	10
Color vs. I	0,761001	2
Color vs. J	0,755709	4

Imagen 14

	C2G-SSIM	RANK-C2G
Color vs. A	0,922662	4
Color vs. B	0,918333	9
Color vs. C	0,922601	5
Color vs. D	0,919363	8
Color vs. E	0,925828	1
Color vs. F	0,922420	6
Color vs. G	0,923406	2
Color vs. H	0,913747	10
Color vs. I	0,922676	3
Color vs. J	0,922026	7

Imagen 15

	C2G-SSIM	RANK-C2G
Color vs. A	0,974647	6
Color vs. B	0,968359	10
Color vs. C	0,974798	4
Color vs. D	0,970614	8
Color vs. E	0,970036	9
Color vs. F	0,974725	5
Color vs. G	0,975317	2
Color vs. H	0,970950	7
Color vs. I	0,975042	3
Color vs. J	0,975386	1

Imagen 16

	C2G-SSIM	RANK-C2G
Color vs. A	0,965590	7
Color vs. B	0,962337	9
Color vs. C	0,965661	6
Color vs. D	0,963289	8
Color vs. E	0,968560	1
Color vs. F	0,966038	5
Color vs. G	0,968441	2
Color vs. H	0,956378	10
Color vs. I	0,968362	3
Color vs. J	0,967176	4

Imagen 17

	C2G-SSIM	RANK-C2G
Color vs. A	0,400940	6
Color vs. B	0,492791	2
Color vs. C	0,416909	4
Color vs. D	0,461830	3
Color vs. E	0,537804	1
Color vs. F	0,372009	10
Color vs. G	0,409811	5
Color vs. H	0,399259	7
Color vs. I	0,398552	8
Color vs. J	0,389717	9

Imagen 18

	C2G-SSIM	RANK-C2G
Color vs. A	0,938675	3
Color vs. B	0,930057	8
Color vs. C	0,938679	2
Color vs. D	0,931631	7
Color vs. E	0,921571	10
Color vs. F	0,938804	1
Color vs. G	0,934519	6
Color vs. H	0,928072	9
Color vs. I	0,934569	5
Color vs. J	0,937859	4

Imagen 19

	C2G-SSIM	RANK-C2G
Color vs. A	0,936909	4
Color vs. B	0,932519	7
Color vs. C	0,936950	3
Color vs. D	0,933443	6
Color vs. E	0,933711	5
Color vs. F	0,937052	2
Color vs. G	0,927516	10
Color vs. H	0,929252	8
Color vs. I	0,927710	9
Color vs. J	0,937455	1

Imagen 20

	C2G-SSIM	RANK-C2G
Color vs. A	0,929733	5
Color vs. B	0,931050	3
Color vs. C	0,929884	4
Color vs. D	0,931274	2
Color vs. E	0,886763	10
Color vs. F	0,927296	6
Color vs. G	0,913666	9
Color vs. H	0,924860	7
Color vs. I	0,913859	8
Color vs. J	0,931716	1

Anexo 4 (Cont.): Índices C2G-SSIM de cada una de las transformaciones en escala de grises para cada imagen a color.

IMAGEN 1

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,7792	0,93610	4	3
Color vs. B	0,4716	0,92970	8	9
Color vs. C	0,6424	0,93610	6	4
Color vs. D	0,3643	0,93030	9	8
Color vs. E	0,7270	0,93280	5	7
Color vs. F	0,7822	0,93610	2	5
Color vs. G	0,8895	0,93700	1	1
Color vs. H	0,0000	0,91600	10	10
Color vs. I	0,7807	0,93680	3	2
Color vs. J	0,5562	0,93590	7	6

IMAGEN 2

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	1,0172	0,851329	1	5
Color vs. B	0,1489	0,792725	8	10
Color vs. C	0,4179	0,851381	7	3
Color vs. D	0,0000	0,798939	10	9
Color vs. E	0,7913	0,860448	2	1
Color vs. F	0,7482	0,851363	4	4
Color vs. G	0,7807	0,850282	3	6
Color vs. H	0,1383	0,800079	9	8
Color vs. I	0,4792	0,849697	6	7
Color vs. J	0,6099	0,851425	5	2

IMAGEN 3

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	1,2334	0,959408	1	6
Color vs. B	0,7066	0,956854	7	9
Color vs. C	0,6545	0,959488	8	5
Color vs. D	0,3318	0,957364	9	8
Color vs. E	1,2077	0,958745	2	7
Color vs. F	0,8986	0,959595	6	4
Color vs. G	0,9538	0,960095	5	1
Color vs. H	0,0000	0,954408	10	10
Color vs. I	1,0830	0,959762	3	3
Color vs. J	0,9644	0,959902	4	2

IMAGEN 4

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,7512	0,976076	1	4
Color vs. B	0,4701	0,971644	4	5
Color vs. C	0,4179	0,976619	5	3
Color vs. D	0,5253	0,971275	3	6
Color vs. E	0,1171	0,944600	9	10
Color vs. F	0,6220	0,976622	2	2
Color vs. G	0,2456	0,964453	7	8
Color vs. H	0,1920	0,963984	8	9
Color vs. I	0,0000	0,965114	10	7
Color vs. J	0,4073	0,987817	6	1

IMAGEN 5

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,6220	0,745675	5	6
Color vs. B	0,0552	0,730271	9	10
Color vs. C	0,8261	0,745803	1	5
Color vs. D	0,2335	0,734361	8	8
Color vs. E	0,5268	0,749140	6	1
Color vs. F	0,2978	0,745411	7	7
Color vs. G	0,7724	0,747658	2	2
Color vs. H	0,0000	0,731745	10	9
Color vs. I	0,7709	0,746857	3	3
Color vs. J	0,7172	0,746097	4	4

IMAGEN 6

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,6439	0,950481	4	4
Color vs. B	0,4285	0,946292	7	8
Color vs. C	0,5471	0,950478	6	5
Color vs. D	0,6976	0,946810	3	7
Color vs. E	0,7928	0,940077	2	9
Color vs. F	0,9326	0,947355	1	6
Color vs. G	0,6099	0,955685	5	2
Color vs. H	0,0000	0,937027	10	10
Color vs. I	0,3976	0,955649	8	3
Color vs. J	0,0967	0,956921	9	1

IMAGEN 7

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,7633	0,864893	4	5
Color vs. B	0,5253	0,859721	6	7
Color vs. C	0,7928	0,864919	3	4
Color vs. D	0,4731	0,860976	7	6
Color vs. E	1,2644	0,877765	1	1
Color vs. F	0,6545	0,865178	5	3
Color vs. G	0,3318	0,839146	9	10
Color vs. H	0,4610	0,848622	8	8
Color vs. I	0,0000	0,839342	10	9
Color vs. J	0,8480	0,865580	2	2

IMAGEN 8

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,3658	0,883792	4	7
Color vs. B	0,0967	0,870974	8	9
Color vs. C	0,0000	0,883830	10	6
Color vs. D	0,0446	0,872120	9	8
Color vs. E	0,8253	0,893350	1	1
Color vs. F	0,2381	0,884233	6	5
Color vs. G	0,6870	0,890919	2	3
Color vs. H	0,1738	0,863625	7	10
Color vs. I	0,4625	0,890936	3	2
Color vs. J	0,3333	0,886943	5	4

Anexo 5: Posiciones de clasificaciones obtenidas de las imágenes evaluadas, tanto para los resultados de las puntuaciones típicas z promediadas (“THURSTONE”) como de los índices C2G-SSIM.

IMAGEN 9

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,2796	0,777124	7	9
Color vs. B	0,6960	0,781574	2	7
Color vs. C	0,1383	0,777588	9	8
Color vs. D	0,3870	0,782617	5	6
Color vs. E	0,7928	0,820261	1	1
Color vs. F	0,3197	0,784303	6	4
Color vs. G	0,5132	0,785478	3	2
Color vs. H	0,1504	0,775109	8	10
Color vs. I	0,4716	0,785004	4	3
Color vs. J	0,0000	0,782756	10	5

IMAGEN 10

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,9402	0,924777	2	3
Color vs. B	0,4391	0,918315	9	9
Color vs. C	0,8986	0,924764	3	4
Color vs. D	0,5699	0,918955	7	8
Color vs. E	1,1231	0,919112	1	7
Color vs. F	0,6636	0,924686	6	5
Color vs. G	0,7066	0,924994	5	1
Color vs. H	0,0000	0,908705	10	10
Color vs. I	0,8125	0,924779	4	2
Color vs. J	0,4822	0,924013	8	6

IMAGEN 11

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,4610	0,974176	4	8
Color vs. B	0,3967	0,979527	5	3
Color vs. C	0,6114	0,973850	1	9
Color vs. D	0,5456	0,979814	3	2
Color vs. E	0,0000	0,967925	10	10
Color vs. F	0,5993	0,976861	2	6
Color vs. G	0,3106	0,977467	6,5	4
Color vs. H	0,1292	0,975285	9	7
Color vs. I	0,3106	0,977253	6,5	5
Color vs. J	0,2781	0,985030	8	1

IMAGEN 12

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,7188	0,922875	6	5
Color vs. B	0,1489	0,915855	9	9
Color vs. C	0,5653	0,922838	8	7
Color vs. D	0,0000	0,916774	10	8
Color vs. E	1,0339	0,926848	2	1
Color vs. F	1,1306	0,922851	1	6
Color vs. G	0,6084	0,926579	7	3
Color vs. H	0,8555	0,913188	3	10
Color vs. I	0,7573	0,926691	5	2
Color vs. J	0,8019	0,924736	4	4

IMAGEN 13

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,3733	0,752060	5	6
Color vs. B	0,2011	0,739681	7	9
Color vs. C	0,1383	0,751979	8	7
Color vs. D	0,0091	0,741043	9	8
Color vs. E	1,0694	0,762321	1	1
Color vs. F	0,5638	0,752228	3	5
Color vs. G	0,3197	0,760955	6	3
Color vs. H	0,0000	0,737300	10	10
Color vs. I	0,6424	0,761001	2	2
Color vs. J	0,4164	0,755709	4	4

IMAGEN 14

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,3839	0,922662	4	4
Color vs. B	0,6084	0,918333	2	9
Color vs. C	0,3197	0,922601	5	5
Color vs. D	0,2441	0,919363	6	8
Color vs. E	0,6515	0,925828	1	1
Color vs. F	0,0000	0,922420	10	6
Color vs. G	0,5101	0,923406	3	2
Color vs. H	0,2011	0,913747	8	10
Color vs. I	0,2335	0,922676	7	3
Color vs. J	0,1504	0,922026	9	7

IMAGEN 15

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,4058	0,974647	6	6
Color vs. B	0,1262	0,968359	9	10
Color vs. C	0,0000	0,974798	10	4
Color vs. D	0,1829	0,970614	7	8
Color vs. E	1,1352	0,970036	1	9
Color vs. F	0,6742	0,974725	5	5
Color vs. G	0,6945	0,975317	4	2
Color vs. H	0,1383	0,970950	8	7
Color vs. I	1,0248	0,975042	2	3
Color vs. J	0,7497	0,975386	3	1

IMAGEN 16

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,1383	0,965590	6,5	7
Color vs. B	0,0000	0,962337	9,5	9
Color vs. C	0,0000	0,965661	9,5	6
Color vs. D	0,5471	0,963289	2	8
Color vs. E	0,7066	0,968560	1	1
Color vs. F	0,2781	0,966038	5	5
Color vs. G	0,5026	0,968441	3	2
Color vs. H	0,1383	0,956378	6,5	10
Color vs. I	0,3643	0,968362	4	3
Color vs. J	0,1058	0,967176	8	4

Anexo 5 (Cont.): Posiciones de clasificaciones obtenidas de las imágenes evaluadas, tanto para los resultados de las puntuaciones típicas z promediadas (“THURSTONE”) como de los índices C2G-SSIM.

IMAGEN 17

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,4467	0,400940	5	6
Color vs. B	0,0000	0,492791	10	2
Color vs. C	0,4391	0,416909	6	4
Color vs. D	0,3636	0,461830	7	3
Color vs. E	0,1610	0,537804	8	1
Color vs. F	0,4497	0,372009	4	10
Color vs. G	0,6757	0,409811	2	5
Color vs. H	0,0643	0,399259	9	7
Color vs. I	0,7293	0,398552	1	8
Color vs. J	0,6311	0,389717	3	9

IMAGEN 18

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,6960	0,938675	1	3
Color vs. B	0,0000	0,930057	10	8
Color vs. C	0,5132	0,938679	4	2
Color vs. D	0,4610	0,931631	5	7
Color vs. E	0,4595	0,921571	6	10
Color vs. F	0,6114	0,938804	2	1
Color vs. G	0,2887	0,934519	7	6
Color vs. H	0,5253	0,928072	3	9
Color vs. I	0,0015	0,934569	9	5
Color vs. J	0,1920	0,937859	8	4

IMAGEN 19

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,6991	0,936909	4	4
Color vs. B	0,7943	0,932519	3	7
Color vs. C	0,8820	0,936950	2	3
Color vs. D	0,6144	0,933443	6	6
Color vs. E	0,4194	0,933711	7,5	5
Color vs. F	0,8910	0,937052	1	2
Color vs. G	0,2154	0,927516	9	10
Color vs. H	0,6575	0,929252	5	8
Color vs. I	0,0000	0,927710	10	9
Color vs. J	0,4194	0,937455	7,5	1

IMAGEN 20

	RAW		RANK	
	THURSTONE	C2G-SSIM	THURSTONE	C2G-SSIM
Color vs. A	0,4164	0,929733	6	5
Color vs. B	0,8895	0,931050	3	3
Color vs. C	0,1814	0,929884	9	4
Color vs. D	1,0573	0,931274	1	2
Color vs. E	0,2441	0,886763	8	10
Color vs. F	0,5993	0,927296	4	6
Color vs. G	0,0000	0,913666	10	9
Color vs. H	0,9001	0,924860	2	7
Color vs. I	0,5026	0,913859	5	8
Color vs. J	0,3409	0,931716	7	1

**Anexo 5 (Cont.): Posiciones de clasificaciones obtenidas de las imágenes evaluadas, tanto para los resultados de las puntuaciones típicas z promediadas (“THURSTONE”) como de los índices C2G-SSIM.**