



**INSTITUTO POLITÉCNICO NACIONAL
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN**

No. 236

Serie: Azul

Fecha: Abril 2008

**LA CUANTIFICACIÓN DEL COLOR:
UNA APROXIMACIÓN AL
ESTADO DEL ARTE**

Sergio A. Márquez De Silva¹
Edgardo M. Felipe Riverón²

RESUMEN

En este reporte recopilamos los aspectos más importantes y describimos las principales herramientas del tratamiento digital de imágenes en el ámbito de la cuantificación del color, así como su estado actual, alcances y aplicaciones.

Palabras Clave: Cuantificación del color; Análisis de imágenes; Segmentación; Agrupamiento.

¹ Alumno de la Maestría en Ciencias de la Computación del C.I.C.

² Profesor-Investigador del laboratorio de reconocimiento de patrones del C.I.C.

ADVERTENCIA

“Este reporte contiene información desarrollada por el Centro de Investigación en Computación del Instituto Politécnico Nacional a partir de datos y documentos con derechos de propiedad y por lo tanto su uso queda restringido a las aplicaciones que explícitamente se convenga.

La aplicación no convenida exime al Centro de su responsabilidad técnica y da lugar a las consecuencias legales que para tal efecto se determinen.

Información adicional sobre este reporte podrá obtenerse recurriendo a la Unidad de Publicaciones y Reportes Técnicos del Centro de Investigación en Computación del I.P.N. Av. Juan de Dios Batiz s/n, teléfono 729-60-00 ext. 56403, 56608 y 56610”.

Índice

I.	INTRODUCCIÓN.....	4
II.	CUANTIFICACIÓN DEL COLOR.....	4
	II.1 Métodos basados en la optimización iterativa.....	7
	II.2 Métodos heurísticos.....	7
	II.3 Métodos basados en cuantificación escalar mejorada.....	8
III.	ALGORITMOS.....	8
	III.1 Cuantificación uniforme.....	8
	III.2 Cuantificación de mínima varianza.....	9
	III.3 Algoritmo de Popularidad.....	10
	III.4 Algoritmo de Corte de mediana (<i>Median-Cut</i>).....	11
	III.5 Algoritmo <i>Octree</i>	11
	III.6 Algoritmo cuantificador principal óptimo (<i>Optimal Principal Quantizer</i>).....	12
	III.7 Algoritmo K-media (<i>K-means</i>).....	12
	III.8 Algoritmo K - medias difuso (<i>Fuzzy K-means</i>).....	13
IV.	RECONSTRUCCIÓN.....	15
	IV. 1 Indexación.....	15
	IV.II Difuminado (<i>Dithering</i>).....	15
V.	DETECCIÓN AUTOMÁTICA DE LOS COLORES ESENCIALES EN UNA IMAGEN DIGITAL.....	15
	V.1 Objetivo general.....	16
	V.2 Descripción.....	16
	V.3 Evaluación de los resultados.....	18
VI.	CONCLUSIONES.....	30
	Referencias.....	30

Lista de figuras

Fig. 1 Esquema del proceso de cuantificación	5
Fig. 2 Ampliación de la imagen "Mandrill"	6
Fig. 3 Cubo de colores <i>RGB</i>	7
Fig. 4 Ejemplo de cuantificación uniforme para el plano $R = 0$ del cubo de colores <i>RGB</i>	9
Fig. 5 Ejemplo de cuantificación de mínima varianza para el plano $R = 0$ del cubo de colores <i>RGB</i>	9
Fig. 6 Imágenes reconstruidas	16
Fig. 7 Diagrama de bloques del método propuesto	17
Fig. 8 Tipos de barridos de píxeles.....	17
Fig. 9 Etapas de la metodología propuesta.....	18
Fig. 10 Caricatura 1	20
Fig. 11 Caricatura 2	21
Fig. 12 Fotografía 1 (Lena)	22
Fig. 13 Fotografía 2 (Mandrill).....	23
Fig. 14 Pintura 1	24
Fig. 15 Pintura 2	25
Fig. 16 Mapa 1	26
Fig. 17 Mapa 2.....	27
Fig. 18 Texto 1	28
Fig. 19 Texto 2	29

I. INTRODUCCIÓN

El primer paso crucial en el análisis de imágenes consiste generalmente en segmentar la imagen. La *segmentación* subdivide la imagen en sus partes constituyentes, que se denominan regiones o segmentos. El nivel al que se lleva a cabo esta subdivisión depende del problema a resolver. Esto es, la segmentación deberá detenerse cuando los objetos de interés de una aplicación hayan sido aislados.

En general, la segmentación autónoma es una de las tareas más difíciles del procesamiento de imágenes y, a su vez, determina el éxito o el fracaso del análisis. Por tanto, habrá que intentar obtener resultados satisfactorios en esta etapa [1].

La segmentación de imágenes es un aspecto muy importante del sistema de percepción visual humana. Las personas utilizan su sentido visual para particionar su entorno en diferentes objetos. Es un proceso complicado que implica un análisis del color, la forma, el movimiento y la textura de los objetos en la imagen. Sin embargo, para el sistema visual humano esta tarea es espontánea y natural. Desafortunadamente no es fácil crear algoritmos artificiales cuyas prestaciones sean comparables a las del sistema visual humano.

El mayor obstáculo para el éxito de dichos algoritmos está en que realmente el sistema visual humano utiliza métodos subconscientes para segmentar [2].

Idealmente los investigadores intentan encontrar la técnica universal de segmentación que sirva para cualquier tipo de imágenes. Esto, si no imposible, puede ser, al menos, un trabajo de toda una vida. En general, los investigadores se suelen centrar en un número limitado de aplicaciones que comparten una serie de características, de forma que los métodos que desarrollan suelen funcionar bien solo para dichas aplicaciones.

A lo largo de los años se ha centrado mucha más atención a la segmentación de imágenes en escala de grises que de imágenes en color. Son las nuevas aplicaciones, como las bases de datos multimedia, fotografías digitales y procesamiento de datos visuales de Internet, lo que ha hecho que aparezca un nuevo interés en la segmentación, y en particular, en la de las imágenes en color.

Existe un gran número de algoritmos de segmentación distintos y cada día aparecen nuevos. Los diferentes métodos de segmentación mezclan técnicas básicas de procesamiento de imágenes, tales como el filtrado y la cuantificación del color; en este reporte se presentarán las distintas técnicas de cuantificación del color que se manejan actualmente.

II. CUANTIFICACIÓN DEL COLOR

El objetivo de la cuantificación de color es realizar una conversión entre una imagen de color real y una imagen definida por una paleta de colores (imagen indexada) [3], [4] (Fig. 1).

El diseño de la cuantificación consiste básicamente en dos pasos:

- 1) Diseño de la paleta de colores.
- 2) Asignación de cada píxel a un color de la paleta.

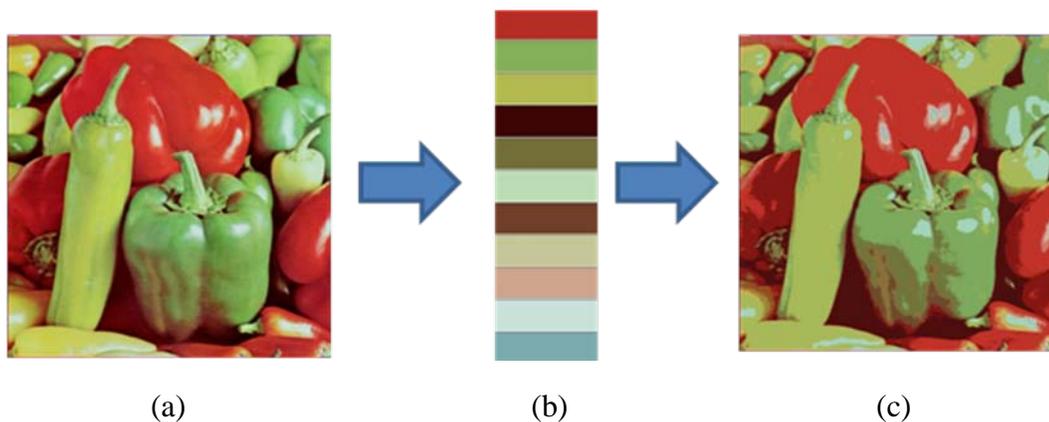


Fig. 1 Esquema del proceso de cuantificación
 (a) Imagen original; (b) Paleta de colores (PC); (c) Imagen generada con la PC.

La mayoría de los métodos para el diseño de la paleta de colores se basan en la *cuantificación vectorial*. De este modo se puede interpretar a cada color de la paleta como una palabra de código y al proceso que le asigna a cada píxel un color, como la codificación.

La tarea de encontrar un conjunto óptimo de colores representativos puede ser vista como un problema de *búsqueda de agrupaciones (clustering en inglés)* a gran escala [5].

Formalmente, la cuantificación del color se refiere a que dada una imagen I , encontrar las particiones óptimas del conjunto de N colores de I en K subconjuntos donde $K < N$ de acuerdo a un criterio de error de cuantificación dado. Los centroides de las agrupaciones definirán, por tanto, las palabras de código que son óptimas en el sentido de mínimo error de cuantificación.

Como error de cuantificación se suele escoger el error cuadrático medio. Dado un píxel x , con tres componentes, el error cuadrático medio de cuantificación d de dicho píxel respecto al verdadero \hat{x} viene dado por:

$$d(x, \hat{x}) = \sum_{i=1}^3 (x_i - \hat{x}_i)^2 \quad \text{Ec. 1}$$

Donde los x_i ($i = 1, 2, 3$) representan las componentes del centroide al que se asocia el píxel x .

Para la imagen completa el error de cuantificación se calcula promediando los errores individuales de cada píxel que forman la imagen.

La cuantificación de colores reduce significativamente el número de colores de la imagen, lo cual es muy deseable para evitar que los métodos de búsqueda de agrupaciones lleven mucho tiempo de cálculo. Es decir, la cuantificación es un paso dentro del preprocesamiento de la imagen (preparación de la imagen) para su posterior segmentación basada en la búsqueda de agrupaciones o en otras estrategias.

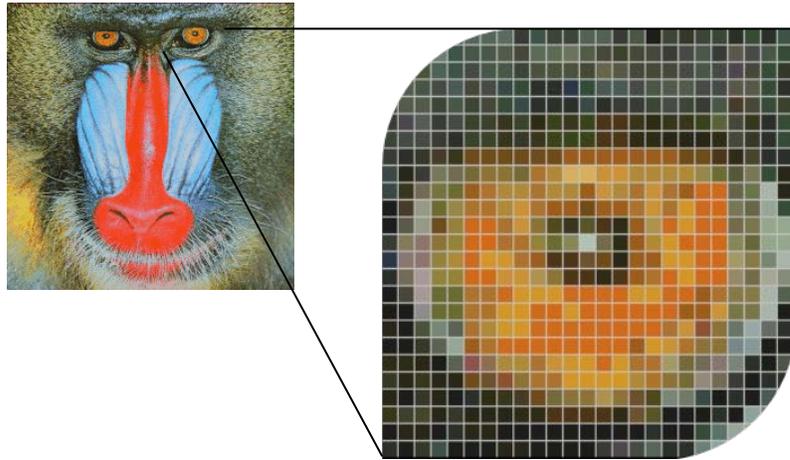


Fig. 2 Ampliación de la imagen "Mandrill"
 Aquí podemos apreciar la gran cantidad de colores que están presentes y que no son discriminados por el ojo humano en la imagen original.

En general, los algoritmos de cuantificación de color pueden ser clasificados en dos categorías

- Métodos independientes de la imagen
- Métodos dependientes de la imagen

Los métodos independientes de la imagen pueden ser vistos como de cuantificación uniforme, dado que el espacio de color es dividido en k regiones del mismo tamaño, mientras que en los métodos dependientes de la imagen las regiones tienen tamaños distintos. En la mayoría de los métodos que entran en esta categoría, el espacio de color es dividido de acuerdo a la distribución estadística de colores en la imagen. Los algoritmos de cuantificación uniforme son, en general, simples, pero sus resultados no son buenos. Por este hecho son más comunes los métodos de cuantificación basados en la imagen. Estos algoritmos de cuantificación pueden ser subdivididos por la metodología empleada para encontrar las agrupaciones de colores adecuadas.

Los métodos de diseño de paletas de colores se pueden dividir en tres categorías [18] (Tabla 1):

Tabla 1 Categorías y métodos de cuantificación

Optimización iterativa	Heurísticos	Cuantificación Escalar
<ul style="list-style-type: none"> • Kmeans • Fuzzy kmeans • Genético kmeans • Redes neuronales 	<ul style="list-style-type: none"> • Divisorios • Acumulativos • Popularidad • Median cut • División binaria 	<ul style="list-style-type: none"> • Uniforme • Mínima varianza

II.1. Métodos basados en la optimización iterativa

Estos métodos, dado un conjunto de colores iniciales, tratan de ajustar la paleta de colores de acuerdo a un criterio de error de cuantificación global. Entre los más conocidos cabe destacar los siguientes:

- a) *Algoritmo LBG (Linde-Buzo-Gray)* [6], también llamado *Algoritmo de K-medias (K-means)* o *Algoritmo generalizado de Lloyd*. El algoritmo se propuso para hallar un diccionario sub-óptimo para aplicaciones de cuantificación vectorial. Representa uno de los algoritmos clásicos y más utilizados de búsqueda de agrupaciones.
- b) *Algoritmo K-medias difuso (Fuzzy K-means)*. Es una extensión difusa del anterior, consistente en la relajación de la estricta regla de clasificación dentro del paso de recálculo de centroides del anterior algoritmo.
- c) *Algoritmo K-media genético* [7]. Es una aproximación híbrida que combina el algoritmo de *K-medias* con un algoritmo genético para evitar la convergencia a un mínimo local.
- d) *Algoritmos de redes neuronales* [3]. Parten de una paleta dada, que es una versión submuestreada de la imagen de color original, con la que van entrenando la red iterativamente de forma que se produzca un ajuste en las neuronas ganadoras, representantes de un color de la paleta. Son métodos que llevan mucho tiempo de computación.

II.2. Métodos heurísticos

En lugar de encontrar los mínimos locales llevando a cabo numerosas iteraciones, intentan obtener una solución aceptable en mucho menos tiempo. La mayoría de los algoritmos se basan en un análisis estadístico de la distribución de los colores de los píxeles. Se clasifican en *métodos divisorios* y *métodos acumulativos*. Los primeros subdividen el espacio de color tridimensional en celdas siguiendo distintos criterios, mientras que los segundos aplican un algoritmo de agrupación jerárquico uniendo iterativamente los centroides según ciertas reglas.

El cubo de colores es una matriz tridimensional de todos los posibles colores que se definen para un tipo de datos determinado de la imagen. Si se escoge un tipo de datos de la imagen de ocho bits por cada componente de color, el cubo de color representará 2^{24} colores posibles. A esta distribución de los colores se le llama cubo de colores *RGB* (Fig. 3)

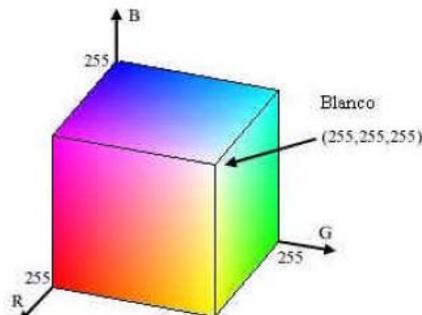


Fig. 3 Cubo de colores *RGB*

La cuantificación implica dividir el cubo de colores *RGB* en un número pequeño de regiones y después asignar todos los colores que caen dentro de una misma región al valor del color central de la misma. La diferencia entre los métodos radica en la aproximación utilizada para dividir el cubo de colores *RGB*.

Otros algoritmos heurísticos que se pueden encontrar en la literatura son el *algoritmo de Popularidad (Popularity algorithm)* [8], que utiliza el histograma de la imagen para seleccionar los *K* colores más frecuentes en la misma; el *algoritmo del corte de mediana (Median-Cut algorithm)* [8], el *algoritmo de división binaria* [5], el algoritmo de Wu [19] y el algoritmo de mejoramiento selectivo condicionado [20].

II.3. Métodos basados en cuantificación escalar mejorada

Estos métodos explotan la dependencia estadística y la dimensionalidad de la imagen. Aquí encontramos los métodos de cuantificación uniforme y de mínima varianza; cabe destacar la *cuantificación escalar secuencial* [9], donde se calculan y analizan las distribuciones marginales de los colores. Cada componente de color se cuantifica individualmente teniendo en cuenta condicionalmente la información de la cuantificación de la componente anterior.

Para eliminar posibles cambios y desplazamientos en los colores, la cuantificación se debe llevar a cabo en los planos *RGB*, en lugar de aplicarla a otros sistemas de representación. Los trabajos [4] [10] [11] han investigado la asignación óptima de los niveles de decisión del cuantificador para imágenes en colores, de forma que se minimicen las distancias entre el color original y los centroides de los grupos que los representan. Los autores de dichos trabajos encontraron que la cuantificación de las coordenadas *RGB* daban los mejores resultados en comparación con otros sistemas de representación del color. La principal razón es que en el sistema *RGB* se utiliza la mayor parte de los niveles de cuantificación, mientras que para otros espacios de color, muchos de estos niveles no se utilizan. También se han encontrado estudios que mantienen el criterio de que existen espacios de color más adecuados para cuantificar los colores que el *RGB*, según la aplicación [3] [12].

III. ALGORITMOS

Un paso fundamental para la posterior segmentación mediante la localización de agrupaciones es el de convertir los datos de la imagen en el formato más adecuado para su representación, es decir, se tendrán que encontrar los descriptores o las transformaciones de los espacios de color que mayores diferencias entre agrupaciones den y más correlación tengan con el sistema visual humano a la hora de distinguir los diferentes colores. A continuación explicaremos los métodos más comunes utilizados para la cuantificación del color.

III.1. Cuantificación uniforme

Para llevar a cabo la cuantificación uniforme de los colores se divide el cubo de colores *RGB* en pequeñas cajas de igual tamaño (pequeños cubos), asignándosele el color del píxel del centro de cada caja a todos los píxeles con los colores que caigan dentro de la misma. Una vez que el cubo

de colores *RGB* se ha dividido y se hayan hecho las correspondientes asignaciones, todas las cajas vacías (sin píxeles con colores dentro de ellas) se eliminan (Fig. 4).

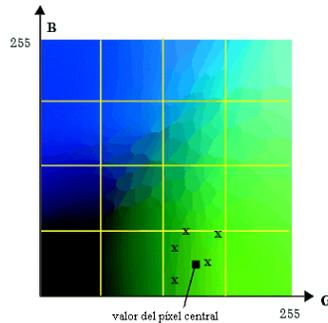


Fig. 4 Ejemplo de cuantificación uniforme para el plano $R = 0$ del cubo de colores *RGB*

III.2. Cuantificación de mínima varianza

En este tipo de cuantificación el cubo de colores *RGB* se divide en cajas, no necesariamente con forma de cubo, sino pueden ser esféricas o de alguna otra forma, y pueden ser de distintos tamaños, con o sin solapamientos. Las dimensiones de las cajas dependen de cómo estén distribuidos los colores en la imagen y no tienen por qué cubrir el cubo de color completamente. Si en algunas áreas de la imagen no existen píxeles con colores diferentes a los ya asignados, no habrá cajas asociadas a ellos.

El cuantificador de mínima varianza trabaja asignando los píxeles a distintos grupos según las varianzas entre sus valores de color, respecto al valor medio de la agrupación. Se establecerá más agrupaciones en las regiones con mayores errores de cuantificación, es decir, donde los datos (colores de los píxeles) tengan mayor varianza [3] [8] [13]. Tras indicar el número de colores a los que se quiere cuantificar, el algoritmo determina el emplazamiento de las cajas analizando los datos de color de la imagen. Una vez dividida la imagen, a los píxeles con colores que caen dentro de una misma caja se les asigna el valor del color del píxel central, como ocurría con la cuantificación uniforme (Fig. 5).

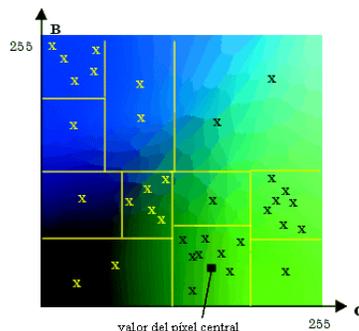


Fig. 5 Ejemplo de cuantificación de mínima varianza para el plano $R = 0$ del cubo de colores *RGB*

Para un número de colores dado, la cuantificación de mínima varianza produce mejores resultados que la cuantificación uniforme, debido a que tiene en cuenta los datos reales. La cuantificación de mínima varianza asigna más entradas del mapa de colores a los colores que aparecen frecuentemente en la imagen original, y menos a los que aparecen con menor frecuencia. Como resultado, la precisión de los colores es mayor que en el caso de la cuantificación uniforme. El tiempo de computación en la cuantificación de mínima varianza es mayor que en la uniforme.

Los siguientes métodos en este apartado se centran en la propiedad de que generalmente los colores en una imagen dada tienden a formar *agrupaciones* en el histograma.

Estas técnicas calculan primero el histograma a partir de los colores de los píxeles que componen la imagen y posteriormente se halla cada agrupación. Entonces, cada píxel de la imagen se asocia a la agrupación más cercana en color. Por tanto, los algoritmos de agrupaciones proveen un método para representar un conjunto de datos de N puntos a través de un conjunto de K puntos, donde $K < N$. El nuevo conjunto de datos es, de alguna forma, representativo del conjunto completo. La mayoría de los algoritmos de agrupaciones son iterativos, donde se parte de un conjunto inicial de puntos y por división, fusión o promediado, van ajustándose hasta obtener el conjunto óptimo de puntos representativos de los colores de toda la imagen.

III.3. Algoritmo de Popularidad

El algoritmo de Popularidad [8] Fue inventado por Tom Boyle y Andy Lippman en el MIT en el verano de 1978, cuando Tom Boyle escribió su software llamado “*Color Maker*”. Subsecuentemente, Paul Heckbert reescribió aquel software con el nombre “*IMAGE*” y publicó el algoritmo.

Es una simple y rápida manera de crear una paleta de color, así como también es uno de los primeros algoritmos creados en el campo de la cuantificación del color.

La idea principal es contar las ocurrencias de cada color en la imagen y entonces seleccionar los M colores más frecuentes para conformar la paleta de color. El algoritmo trabaja en un tiempo $O(N \log N)$, para escoger los M colores más frecuentes que pueden ser detectados para crear un primer ordenamiento de los valores de píxeles y entonces después escanear la lista ordenada. Algunos valores de píxel son entradas consecutivas en la lista ordenada, así que escanearlos tomará solo el tiempo $O(N)$.

El principio detrás del algoritmo de popularidad es que los picos del histograma corresponden a los centroides de los *clusters*. Desafortunadamente este principio no se mantiene en la práctica.

En algunos casos la interacción humana es necesaria. Cuando hay partes importantes de la imagen, pero con colores con poca frecuencia de ocurrencia, el algoritmo por sí solo las descartaría y reemplazaría, como por ejemplo, los ojos de una persona en una fotografía de su rostro. El remedio para esto es usar un método para que el usuario manualmente elija las zonas que se quieren enfatizar. El número de los píxeles de estas zonas serán multiplicados por un número mayor que uno en el conteo antes de incorporarlos a la tabla de frecuencias, garantizando así una mejor representación en la paleta de colores.

III.4. Algoritmo de Corte de mediana (Median-Cut)

Este algoritmo fue el primero propuesto por Heckbert en su trabajo relacionado con la cuantificación de color clásica [8]. Su idea es partir recurrentemente el espacio de color en particiones rectangulares, esto es, subdividir en varias ocasiones el cubo de colores en cajas rectangulares cada vez más pequeñas. Primero, el algoritmo proyecta todos los vectores a los ejes coordenados. El punto medio de los valores proyectados es el punto de corte de esa partición. Esto conduce a igualar el número de colores S_1 y S_2 en ambas particiones nuevas.

El proceso continúa recurrentemente hasta que se crea el número de M particiones. Entonces los centroides de las particiones forman la paleta de color. El algoritmo original es recursivo, pero la variante iterativa es fácil de definir. A continuación se presenta la variante iterativa.

$S_1 \leftarrow$ Cubo *RGB* más pequeño que cubre todos los vectores
 $M \leftarrow 1$
Repetir
 $S_i \leftarrow$ Selecciona el que maximice $(\max(X_i) - \min(X_i))$
 $X_i \leftarrow$ Encuentra el eje de S_i que maximice.
 $X_j \leftarrow$ Encuentra la mediana de X_i
 $S_{2i}, S_{2i+1} \leftarrow$ Particiona S_i de la ortogonal X_j a X_i
 Calcula los covectores C_{2i} y C_{2i+1}
 $M \leftarrow M + 1$
Hasta ($M = |C|$)

Existen muchas variaciones del algoritmo, por ejemplo, el *Center-Cut* que divide por el centro las particiones en vez de por la mediana.

III.5. Algoritmo Octree

El algoritmo de cuantificación *Octree* subdivide repetidamente el cubo en ocho cubos más pequeños con una estructura arborescente de grado ocho. Entonces los cubos adyacentes con el menor número de píxeles se combinan. Se repite esto hasta que el número requerido de colores se obtiene. *Octree* produce resultados similares al algoritmo de *Median-Cut*, pero con una velocidad más alta y con menos requisitos para la memoria.

El algoritmo de cuantificación *Octree* es rápido y produce paletas de color de calidad. La publicación original fue reimpressa más adelante en “Graphics Gems I” lo que dio lugar a la adopción amplia del algoritmo en diversas implementaciones. Muchos paquetes comerciales del procesamiento de imágenes todavía están utilizando el algoritmo *Octree*.

Este se basa en una estructura de datos consistente en un árbol donde cada nodo contiene indicadores a ocho nodos hijos. Los colores se almacenan en los nodos de la hoja correspondiente. Este produce la paleta de colores en un solo paso y no utiliza ningún tipo de pre cuantificación de los vectores de la entrada. La idea es que la imagen es procesada píxel por píxel. Se atraviesa el árbol hasta que se alcanza un nodo de la hoja y si el nodo de la hoja está

vacío, entonces el color del píxel se inserta en la estructura de datos del octree; si no, la cuenta de la frecuencia de ese color se aumenta.

Los colores del modelo de color *RGB* tienen 8 bits por componente, así que podemos separar el color *RGB* en 8 planos de 3 bits. Por ejemplo, supongamos que en la posición del bit i R tiene su bit con valor 1, G en 0 y B en 1. Esto conduce al número binario $101_2 = 5_{10}$. Está claro que con 3 bits podemos representar números del cero al siete. Este número se utiliza como índice del nodo hijo. Primero tomamos el plano más significativo y utilizamos el índice para atravesar a uno de los ocho nodos hijos. Entonces tomamos el siguiente plano y lo utilizamos otra vez como índice de un nodo hijo. Esto continúa hasta que hemos recorrido todo el árbol. Cuando se han procesado todos los píxeles, entonces el proceso de reducir las hojas del árbol comienza.

El algoritmo busca un nodo para el cual la suma de las frecuencias es mínima. Calcula el vector medio de todos los nodos de la hoja situados bajo este nodo. Quita todos los nodos de la hoja y almacena el nuevo vector medio con las frecuencias sumadas. El proceso continúa hasta que solamente queden M hojas.

El operador de la reducción divide el espacio de color en diversos tamaños. El tamaño del cubo depende del nivel en el cual la reducción del octree fue realizada.

III.6. Algoritmo cuantificador principal óptimo (Optimal Principal Quantizer)

Es bien sabido que los colores en el espacio de color están separados por el eje de la luminiscencia. Por lo tanto el eje de la luminiscencia es también un eje principal. Wu [19] propuso un algoritmo que primero proyecta todos los vectores al eje principal de luminiscencia y entonces clasifica los vectores según sus proyecciones.

Ahora bien, si consideramos los valores de la proyección, tenemos una lista de los valores escalares que representan el espacio de color entero. El algoritmo *Optimal Principal Quantizer* utiliza el cuantificador escalar óptimo en los valores de la proyección para repartir el sistema de proyecciones en los intervalos de k , donde $k < M$.

La función de distorsión usada en la cuantificación escalar de los valores de la proyección es la distorsión que tiene lugar al cuantificar los vectores en esa partición. De esa manera el cuantificador principal óptimo se puede considerar como método híbrido de dos métodos existentes: la cuantificación escalar y la cuantificación vectorial.

El valor de k se determina dinámicamente a medida que el proceso dinámico continúa. Si después de que el cuantificador principal es encontrado $k < M$ entonces todavía M/k particiones necesitan ser formadas. El algoritmo entonces usa un esquema de partición local óptima para dividir las particiones creadas en pasos anteriores.

III.7. Algoritmo K-media (K-means)

Este método de búsqueda de agrupaciones en n dimensiones produce K vectores de medias que representan K clases de datos. Este algoritmo también se conoce con las siglas LBG, provenientes de las siglas de los apellidos de sus autores (Linde, Buzo y Gray) [6]. Se basa en una medida de distorsión $d(x, y)$ entre puntos de R^n .

Sea el conjunto de datos de entrenamiento $X = \{x_1, x_2, \dots, x_N\}$. Dado un *centroide* (representante de una agrupación) y_i , el conjunto de puntos de X que esté más cercano a y_i que a cualquier otro centroide, según la medida de distorsión $d(x, y)$, se denomina *región de Voronoi* de y_i la que se denota por

$$V_i = \{x \in X: d(x, y_i) < d(x, y_j), i \neq j\}. \quad \text{Ec. 2}$$

El número de vectores en una región de Voronoi se representa por $|V_i|$. El centroide de los vectores de una región de Voronoi viene dado por

$$\frac{1}{|V_i|} \sum_{x \in V_i} x \quad \text{Ec. 3}$$

Sea $y(x)$ el centroide y_i más cercano a x . La distorsión promedio para todo el conjunto de datos X es:

$$d(X) = \frac{1}{|X|} \sum_{x \in X} d(x, y(x)) \quad \text{Ec. 4}$$

Con esta notación, el algoritmo *K-means* se resume en los siguientes pasos:

- 1) Elegir un conjunto inicial de centroides y_1, y_2, \dots, y_K .
- 2) Determinar la región de Voronoi para cada y_i .
- 3) Calcular el centroide de cada región de Voronoi.
- 4) Si el algoritmo no ha convergido, ir al paso 2; en cualquier otro caso, parar.

Un posible criterio para saber si el algoritmo ha convergido es que la distorsión promedio sea suficientemente pequeña. Otro criterio es calcular el cambio en la distorsión de una iteración a otra y parar si ese cambio es suficientemente pequeño.

La determinación de los centroides iniciales juega un papel crucial debido a que cuanto mejor sea la partición inicial, más rápido convergerá el algoritmo. Es más, aunque se asegura la convergencia, ésta no tiene por qué ser a un mínimo global. Una forma de llevar esta elección a cabo es calcular el histograma N -dimensional de la imagen y determinar los picos dominantes del mismo. Los K picos dominantes se corresponderán con los K centroides iniciales [2]. En otros algoritmos, las agrupaciones iniciales se escogen aleatoriamente [14], con lo que disminuyen la dificultad, pero aumentan el tiempo de convergencia del algoritmo.

III.8. Algoritmo K - medias difuso (Fuzzy K-means)

El algoritmo *K-means* explicado en el apartado anterior puede extenderse para incluir reglas de inferencia difusa, formando lo que se denomina el algoritmo *Fuzzy K-means* [15]. En el algoritmo *K-means* anterior un vector x puede pertenecer solamente a una región descrita por el centroide y_i . No existe la posibilidad de pertenecer parcialmente a una agrupación y parcialmente a otra, y sin embargo, en muchos problemas de clasificaciones de patrones sería deseable que el

sistema tuviera esto en cuenta. Para ello, se puede utilizar una función de pertenencia que indique el grado en el que un vector pertenece a una agrupación.

El hecho de que el grado de pertenencia a un conjunto determinado sea explícitamente representado, es el que lleva al uso de la palabra *borroso o difuso* (en inglés, *fuzzy*) para describir este tipo de agrupaciones.

Sea u_{ij} un indicador de la pertenencia de un vector de datos x_j a la agrupación i . Para el algoritmo *K-means*, $u_{ij} \in \{0, 1\}$, es decir, sólo toma dos posibles valores.

$$u_{ij} = \begin{cases} 1, & \text{si } X_j \in \text{agrupación } i \\ 0, & \text{en otro caso.} \end{cases} \quad \text{Ec. 5}$$

Como todo elemento tiene que pertenecer a una agrupación, se cumple que:

$$\begin{aligned} \sum_{i=1}^K u_{ij} &= 1, \quad j=1,2,\dots,N, \\ \sum_{i=1}^K \sum_{j=1}^N u_{ij} &= N \end{aligned} \quad \text{Ec. 6}$$

Para el algoritmo *Fuzzy K-means* se puede generalizar el concepto del indicador u_{ij} a que tome un valor comprendido en el rango $[0, 1]$. Bajo esta especificación, u_{ij} indica el grado en el que x_j pertenece a la agrupación i , y se dice que es una *función de pertenencia al conjunto*.

Para describir el algoritmo de búsqueda de agrupaciones basado en esta generalización, se define una *función objetivo* basada en una medida ponderada de similitud entre los píxeles de la imagen y cada uno de los K centroides. Un extremo local de dicha función objetivo indica una agrupación óptima de los datos de entrada. La función objetivo a minimizar viene dada por:

$$J(U, Y) = \sum_{j=1}^N \sum_{i=1}^K (u_{ij})^m d^2(x_j, y_i) \quad \text{Ec. 7}$$

Donde $U = [u_{ij}]$ es la matriz de las funciones de pertenencia, $Y = [y_1, y_2, \dots, y_K]$ es el conjunto de centroides, y m ($1 \leq m \leq \infty$) varía la naturaleza de la agrupación (para $m = 1$ se tiene una agrupación “dura” y para valores altos de m se tienen agrupaciones con la borrosidad en aumento). La medida de distancia se toma, por razones analíticas, como

$$d^2(x_j, y_i) = (x_j - y_i)^T (x_j - y_i) \quad \text{Ec. 8}$$

La funcional $J(U, Y)$ mide el error al representar K agrupaciones borrosas indicadas por U con centroides representados por Y . El objetivo es minimizarla, sujeto a la restricción:

$$\begin{aligned} \sum_{i=1}^K u_{ij} &= 1, \quad j = 1, 2, \dots, N \\ u_{ij} &\geq 0, \quad i = 1, 2, \dots, K \quad j = 1, 2, \dots, N \end{aligned} \quad \text{Ec. 9}$$

Cuando $m = 1$, el algoritmo se reduce al de *K-means*. Si $m > 1$, hay que resolver un problema de optimización con restricciones. Es decir, el problema de búsqueda de agrupaciones por el algoritmo *Fuzzy K-means* se resuelve eligiendo un conjunto inicial de K centroides o medias, e iterando las dos ecuaciones hasta que converjan.

Los algoritmos de agrupaciones de píxeles encuentran problemas en las situaciones siguientes:

- 1) A menudo las agrupaciones adyacentes se superponen en el espacio de color.
- 2) La segmentación es más difícil cuando el número de agrupaciones es desconocido, lo que ocurre con frecuencia.
- 3) Al ser técnicas de segmentación basadas en el píxel, no tienen en cuenta las restricciones espaciales, lo cual las hace susceptibles al ruido.

IV. RECONSTRUCCIÓN

Una vez obtenida la Paleta de Color (PC), se procede a la reconstrucción de la imagen, donde el método a usar varía según la aplicación. Si lo que se desea es segmentar es muy conveniente que los bordes entre los objetos estén muy bien definidos y reducir los degradados que pueda tener la imagen. Pero si la aplicación requiere que la imagen conserve sus propiedades perceptuales se requiere otra técnica.

IV.1. Indexación

Para el primer caso es común emplear el método de indexación por similitud, donde simplemente se reemplaza el color original con el color que tenga más parecido en la PC. Esta similitud es calculada por algún tipo de distancia en un espacio de representación del color. En nuestro caso empleamos la formula CIE2000 para calcular las diferencias del color [21].

IV.2. Difuminado (*Dithering*)

En caso de que se desee conservar la mayor similitud con la imagen original, se emplea algún método de difuminado (*dithering*), Generalmente se usa el método de Floyd-Steinberg [16].

V. DETECCIÓN AUTOMÁTICA DE LOS COLORES ESENCIALES EN UNA IMAGEN DIGITAL

Se han descrito los algoritmos más comunes para la cuantificación del color. Ahora nos encontramos frente al reto de la detección automática de la cantidad de colores. Debido a que todos los métodos clásicos de agrupación de colores previamente expuestos reciben como parámetro inicial la cantidad de conjuntos a los que agrupar los colores (*clusters*), esto se traduce en la cantidad de colores que uno desea obtener como resultado final después de la cuantificación. Estos algoritmos pueden ser eficientemente empleados cuando la aplicación recibe siempre el mismo tipo de imágenes, por lo que una cantidad de colores a priori preestablecida puede ser determinada manualmente sin problemas. Pero cuando es de interés el

encontrar la cantidad de colores representativos para cualquier tipo de imagen estos algoritmos simplemente no son los adecuados.



(a)

(b)

Fig. 6 Imágenes reconstruidas

(a) Imagen reconstruida por indexación; (b) Imagen reconstruida por *dithering*

El trabajo de tesis que lleva por título “Detección automática de los colores esenciales de una imagen digital”, presenta un algoritmo para lograr una cuantificación de colores sin la necesidad de conocer a priori la cantidad de colores a la cual se debe reducir; por lo que para una imagen en la que encontramos una gran variedad de colores el resultado será mucho mayor que en aquella que podemos etiquetar menos colores sin importar la cantidad real de colores muestreados.

V.1. Objetivo general

Desarrollar una metodología para la detección automática de los colores esenciales de una imagen digital, entendiendo como colores esenciales aquellos que son necesarios para reconstruir la imagen sin pérdida de los objetos representativos que componen la imagen.

V.2. Descripción

Se ha desarrollado un algoritmo que, a diferencia de los métodos clásicos de cuantificación de color, no requiere conocer a priori el tamaño deseado de la paleta de colores (cantidad de colores) o la cantidad de agrupaciones. Esto nos conduce a un método de cuantificación del color no supervisado con paleta de colores dinámica. El diagrama de bloques se muestra en la Fig. 7.



Fig. 7 Diagrama de bloques del método propuesto

En la primera fase convertimos los colores de la imagen que se encuentran en el espacio RGB al espacio $L^*a^*b^*$. Este espacio de color es utilizado debido a que es perceptualmente uniforme, por lo que se ajusta de una manera más precisa a la forma en la que el sistema visual humano diferencia los colores.

La segunda fase está dedicada a obtener regiones homogéneas mediante un proceso de crecimiento donde partiendo de un píxel no procesado, se aglomeran sucesivamente los píxeles vecinos que cumplen con un criterio de homogeneidad, el que en nuestro caso depende de un parámetro al que hemos llamado *nivel de tolerancia*. El valor de este parámetro equivale al límite que se acepta para la diferencia de color, la que se calcula por la fórmula de distancia CIE2000; esta es una métrica aplicada en el espacio $L^*a^*b^*$ para mejorar la percepción en ese espacio de color. Los colores de las diversas regiones se van modificando en busca del centroide entre los colores de los píxeles que las forman.

Es común explorar las imágenes por filas o columnas sucesivas. Cabe destacar que en nuestro caso la exploración de la imagen se hizo en zigzag, dado que con esto se evitó el efecto de escurrimiento encontrado en los resultados iniciales creados con una exploración por filas o columnas (Fig. 8).

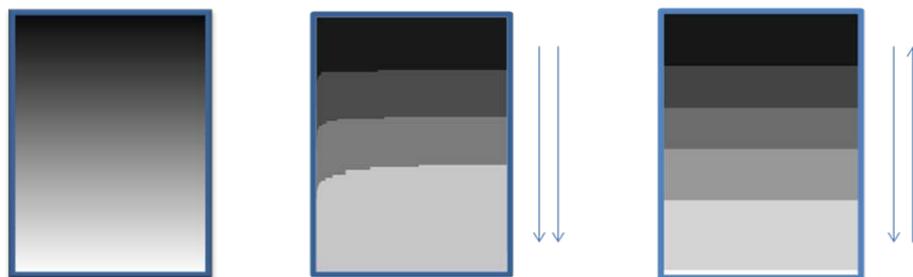


Fig. 8 Tipos de barridos de píxeles

Una imagen de prueba (izquierda) y sus versiones de exploración, usando el recorrido por columnas (centro) y usando el recorrido en zigzag (derecha).

Después de esta fase, ya se cuenta con una paleta de colores reducida por regiones, a la que en la fase 3 le aplicamos otro agrupamiento donde los colores con menor frecuencia de ocurrencia se agrupan con los colores de mayor frecuencia, siempre y cuando estén dentro de los márgenes del nivel de tolerancia establecido. Así se genera la paleta de colores resultantes.

Una vez obtenida la paleta se procede a reconstruir la imagen. La metodología propuesta brinda como uno de sus resultados una imagen con las regiones creadas en la fase 2. Este resultado colateral nos lleva a un posterior análisis de sus ventajas para la segmentación y el reconocimiento de patrones.



Fig. 9 Etapas de la metodología propuesta

De izquierda a derecha y de arriba hacia abajo: Imagen original con 35,121 colores; Imagen reconstruida por indexación; Imagen reconstruida por difuminación; y finalmente las regiones encontradas por el algoritmo. Las últimas tres imágenes tienen sólo 33 colores.

V.3. Evaluación de los resultados

Para determinar la eficiencia de nuestro algoritmo, hemos dividido el cúmulo de imágenes digitales en colores en cinco tipos diferentes, a saber:

- Caricaturas
- Fotografías
- Pinturas
- Mapas
- Textos

Algunas imágenes dentro de cada tipo, en el orden siguiente, serán cuantificadas con los algoritmos clásicos *Median-Cut* [8], el de componentes principales desarrollado por Wu [19] y desde luego con el algoritmo propuesto.

Para lograr una comparación congruente y válida se siguió el siguiente procedimiento:

1.- Cuantificar la imagen con el método propuesto y anotar la cantidad de colores que se obtuvo.

2.- Ajustar los otros algoritmos de cuantificación (clásicos) a la cantidad de colores que se obtuvo con el método propuesto.

3.- Mostrar en un histograma tridimensional la paleta de colores construida con cada método y para todos los casos mostrar la imagen regenerada mediante el algoritmo de difuminación de Floyd-Steinberg.

Por lo tanto, para cada imagen mostramos:

- Cada uno de los resultados obtenidos con los algoritmos clásicos;
- El histograma 3D de frecuencia (probabilidad de ocurrencia) en el espacio de color *RGB*.

Las imágenes y sus histogramas son presentados de izquierda a derecha y de arriba hacia abajo en el siguiente orden: Original, algoritmo propuesto, *Median-Cut* y componentes principales. Se consideró conveniente probar los algoritmos con dos imágenes en colores de cada tipo. (Figuras 10-19).

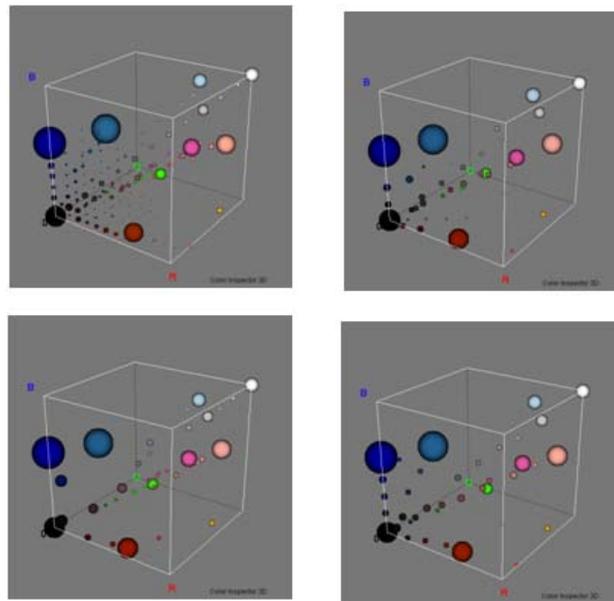
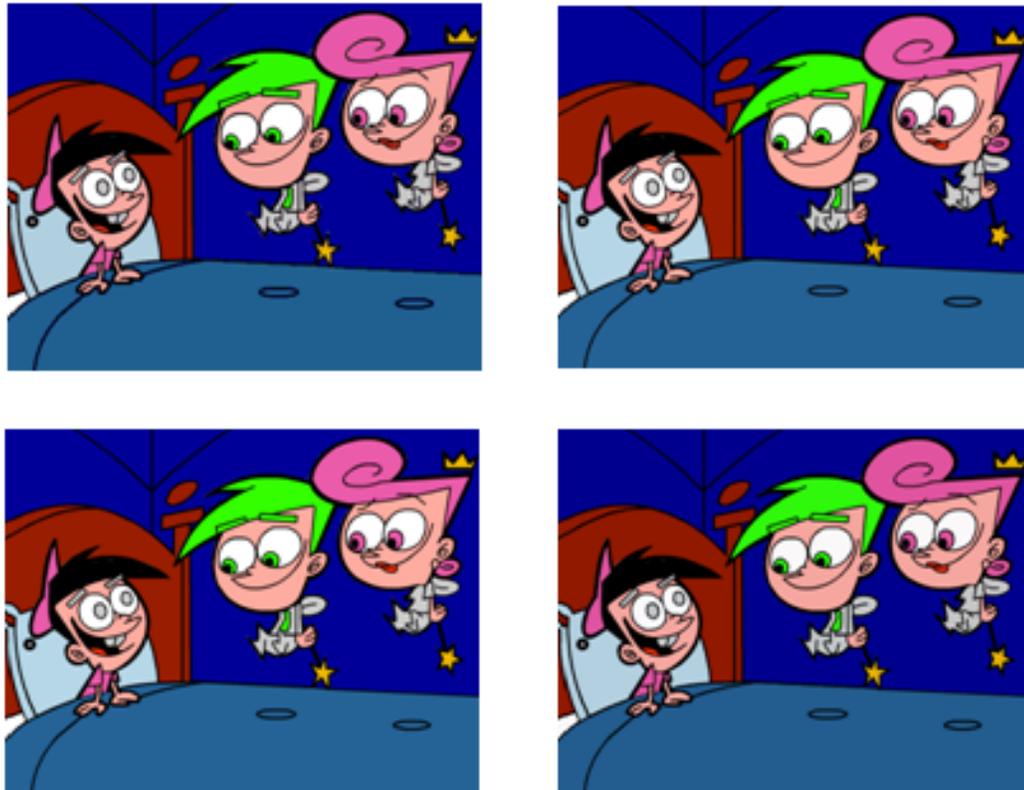


Fig. 10 Caricatura 1

De 1,398 colores se redujo a 41 colores. No apreciamos algún tipo de distorsión en las imágenes resultado con respecto a la imagen original. Entre los 3 métodos no hay una diferencia significativa en los resultados.



Fig. 11 Caricatura 2

De 63,421 colores se redujo a 68 colores. En esta imagen, cuya cantidad de colores inicial es muy grande, reconocemos el poder de la cuantificación. Podemos observar cómo el método propuesto agrupa la mayoría de los azules en un solo azul, mientras que los otros métodos mantienen una gran redundancia en los azules.

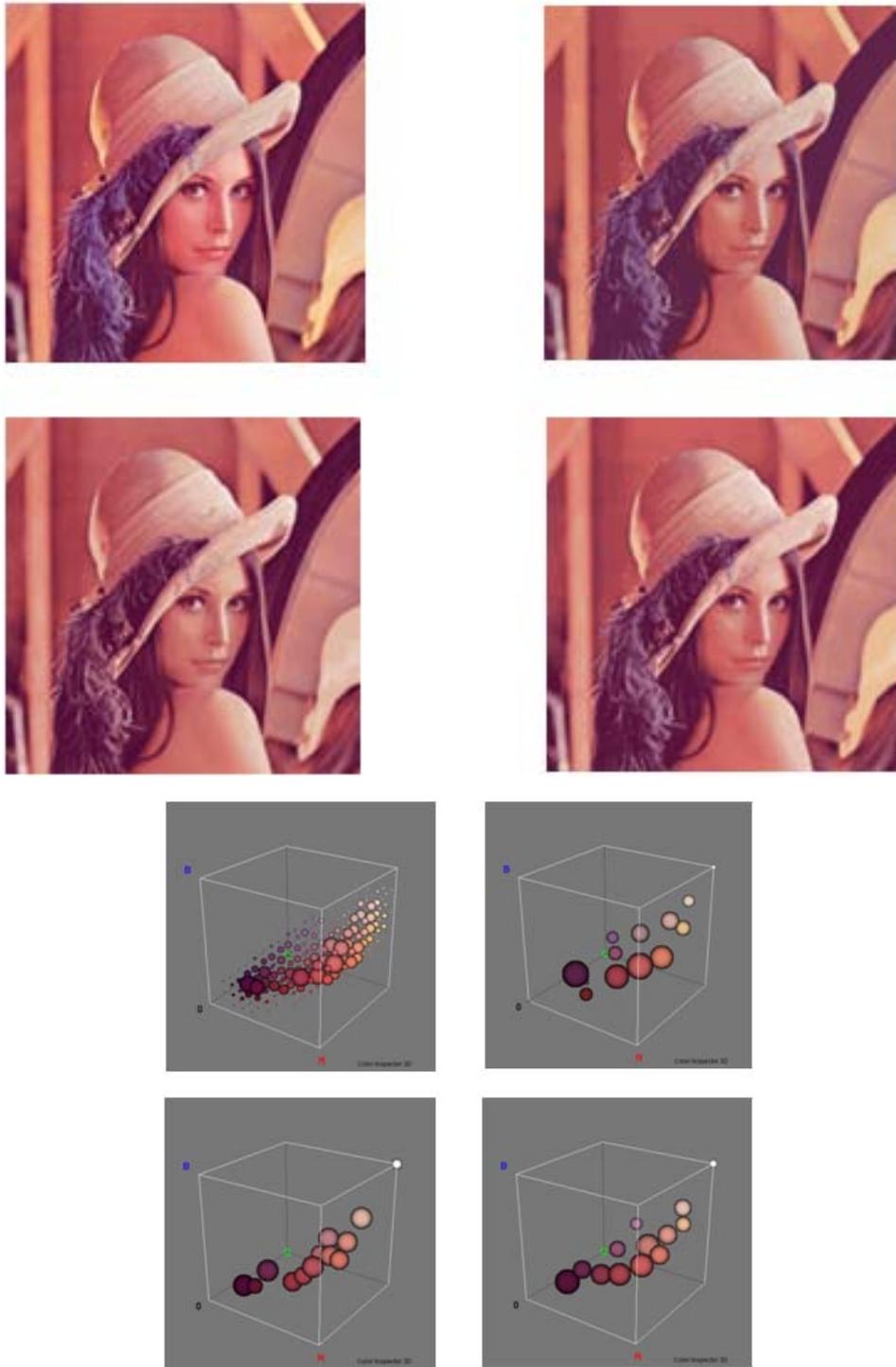


Fig. 12 Fotografía 1 (Lena)

Lena [17], de 71,068 colores se redujo a 12 colores. En esta imagen, el algoritmo reduce automáticamente a 12 colores, debido a la poca variedad de colores que el ojo humano puede apreciar en la imagen original; por ello es que su comportamiento es el esperado.

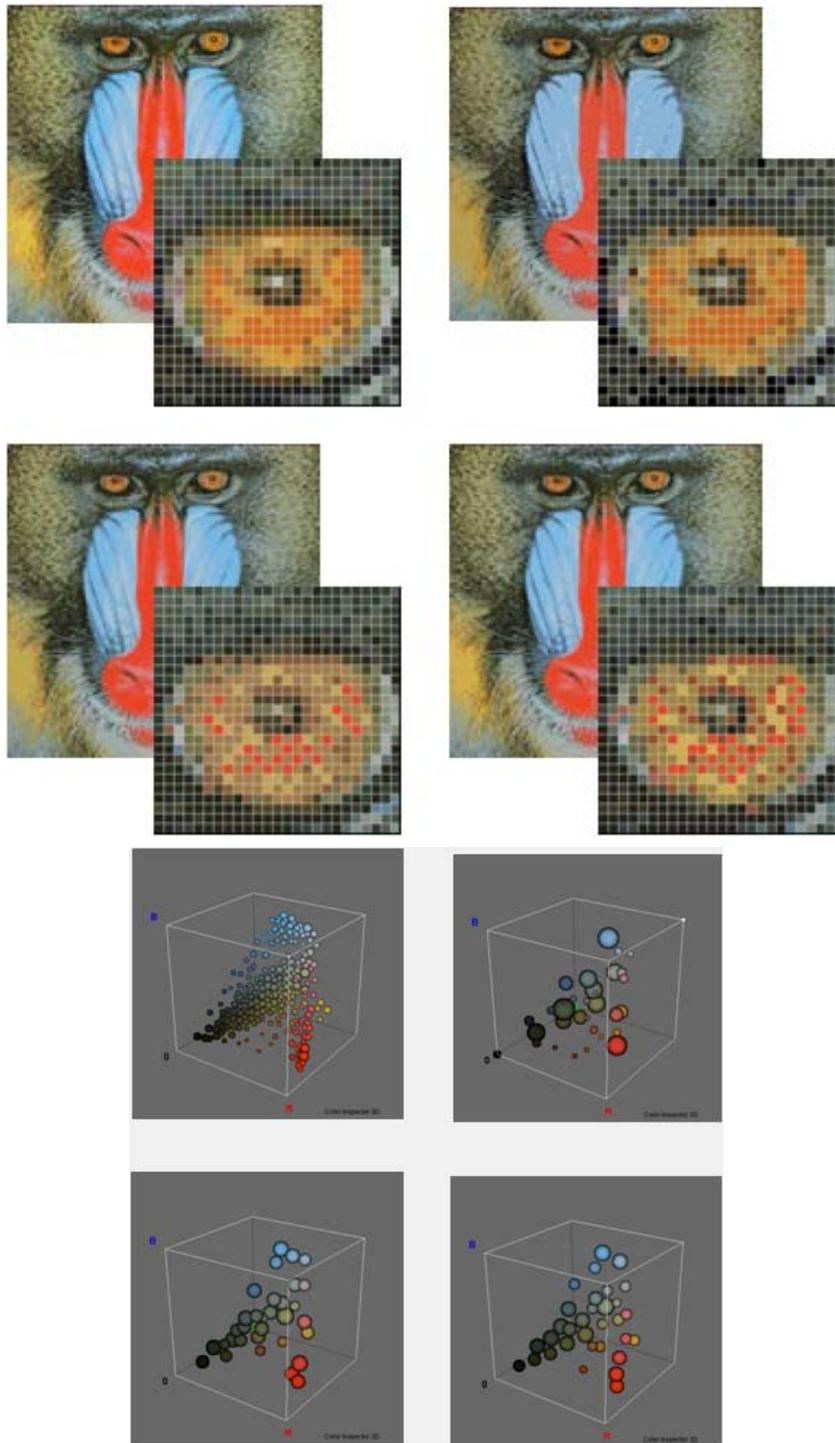


Fig. 13 Fotografía 2 (Mandril)

Mandril [17], de 256 colores se redujo a 32 colores. En esta imagen se hizo una ampliación del ojo del mandril para que se pueda apreciar cómo el método propuesto es más noble con los detalles y conserva mejor la relación de tonos que los otros métodos.

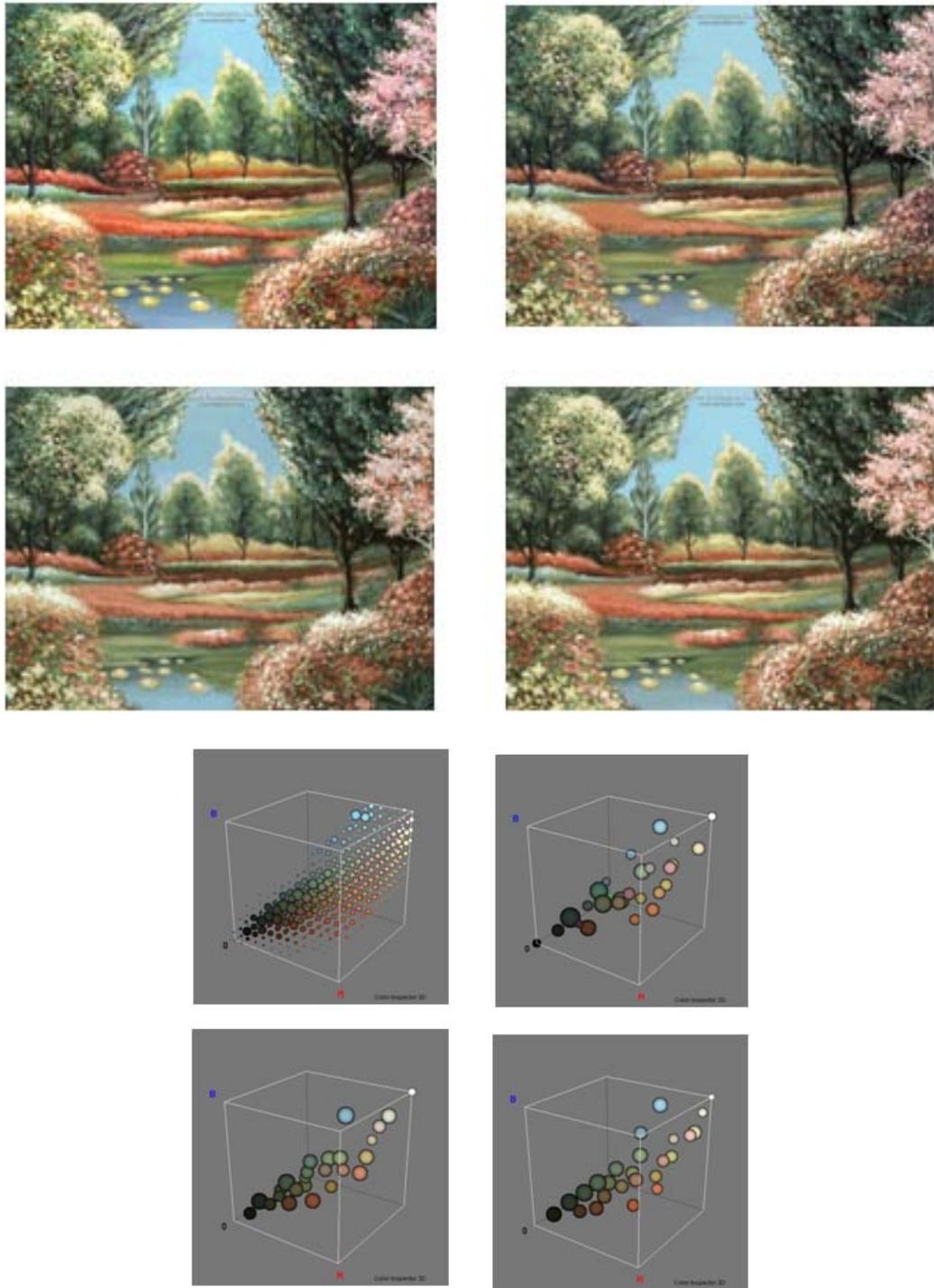


Fig. 14 Pintura 1

De 306,234 colores se redujo a 24 colores. Una imagen con una gran cantidad de colores en la imagen original reducida automáticamente a 24 colores. El *Median-Cut* presenta una variación importante de tonalidades, mientras que el de componentes principales y el propuesto mantiene una gran similitud con la imagen original.

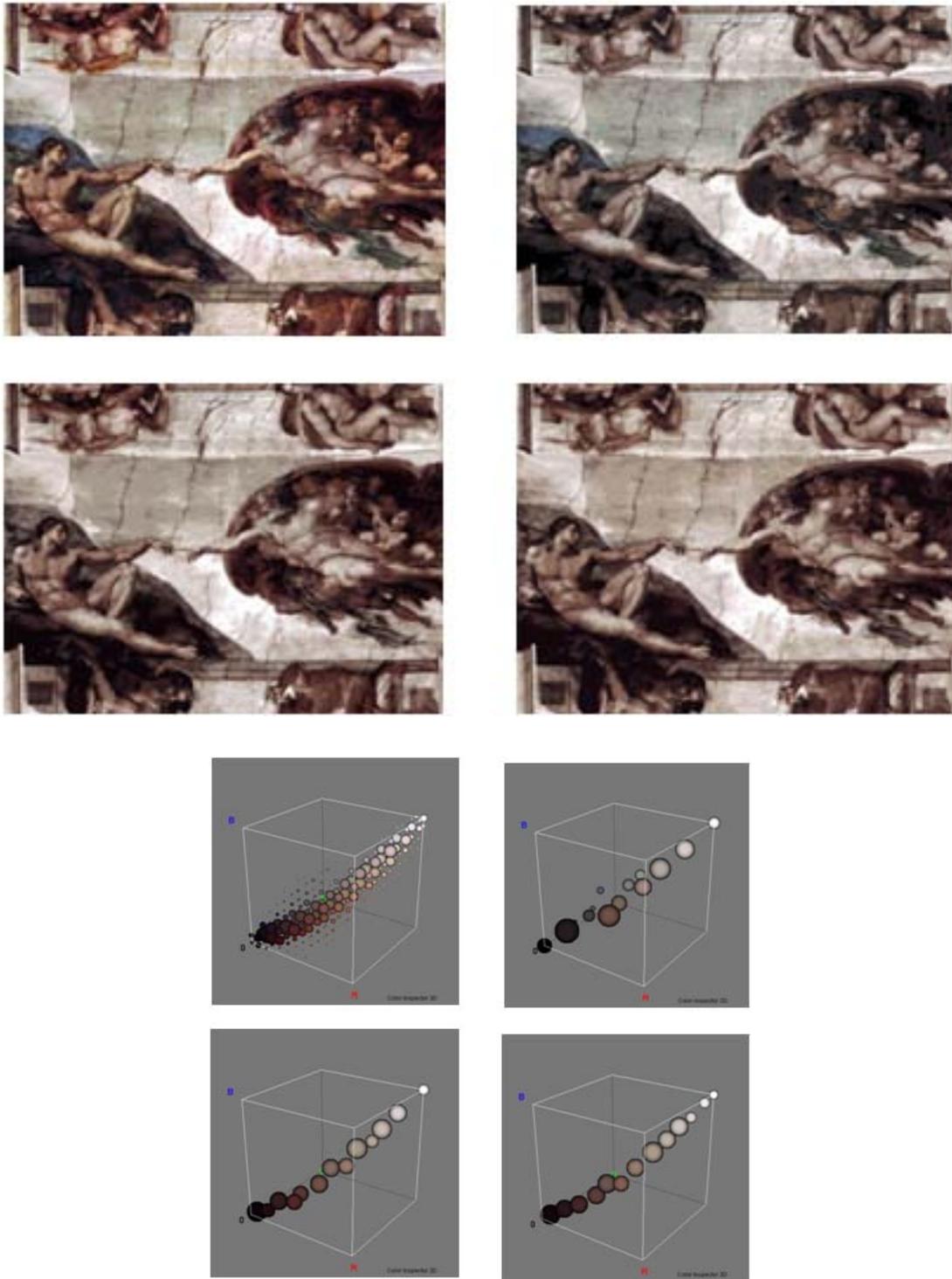


Fig. 15 Pintura 2

De 103,317 colores se redujo a 12 colores. En esta imagen se puede observar que el método propuesto es el único que conserva la zona azul en la parte central izquierda de la imagen original.

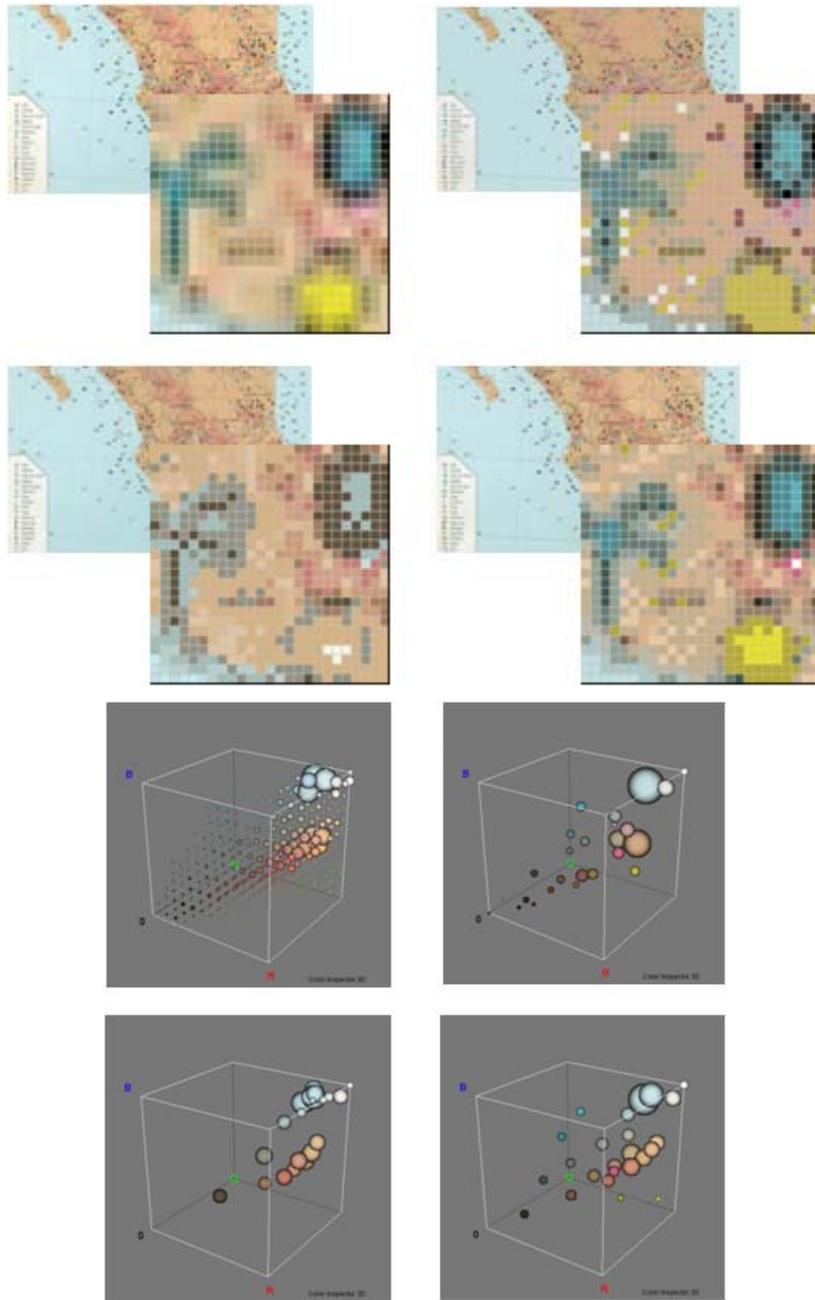


Fig. 16 Mapa 1

De 144,998 colores se redujo a 23 colores. El mapa de México es una imagen representativa de un problema que pretende ser resuelto con la metodología propuesta: el reconocimiento de los símbolos cartográficos. El algoritmo Median-Cut nos muestra un deterioro grande en la imagen al desaparecer el símbolo amarillo, mientras que los otros métodos lo mantienen, pero con mejores resultados con el método propuesto. Puede verse que arroja una imagen con los bordes de los objetos más definidos, lo cual es útil para el reconocimiento y/o la segmentación de los objetos gráficos.

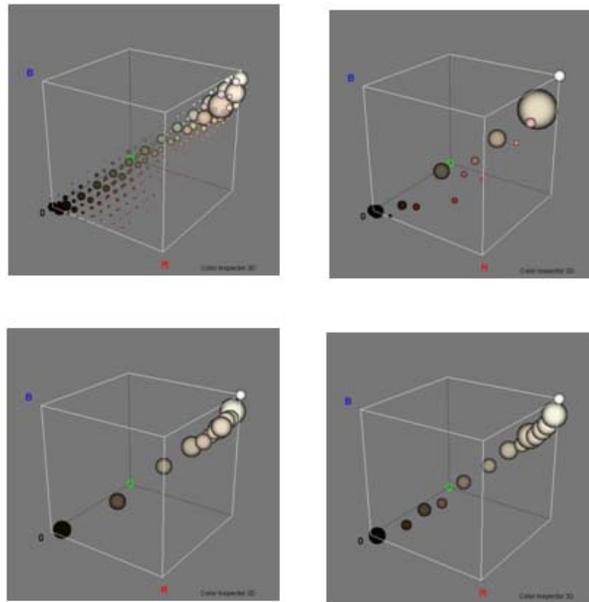
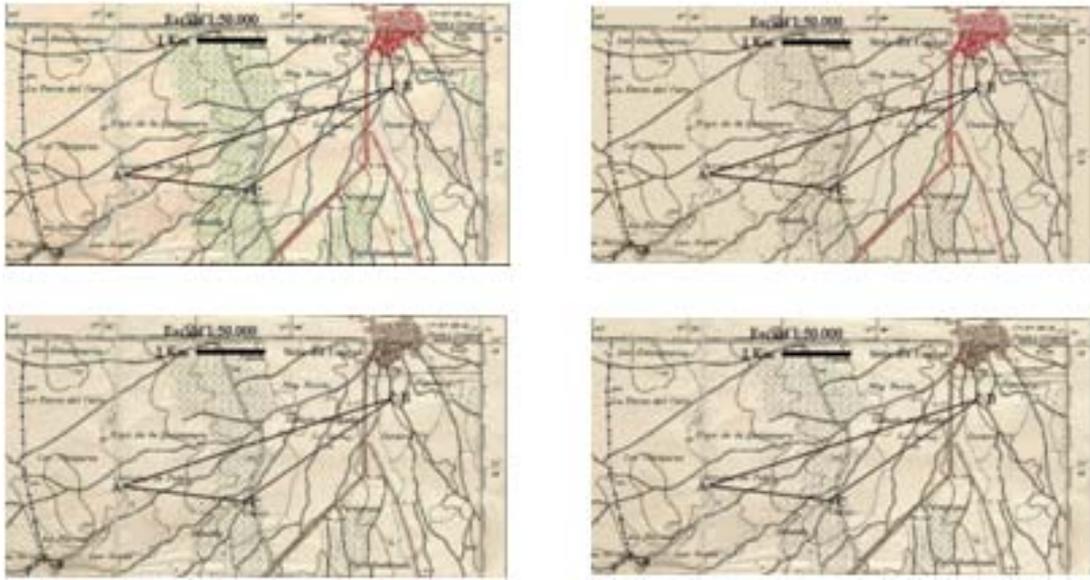
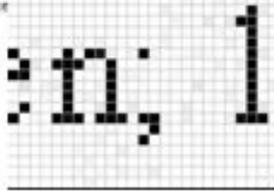


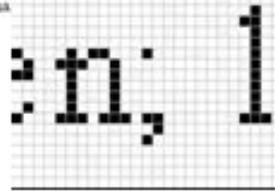
Fig. 17 Mapa 2

De 36,085 colores se redujo a 13 colores. Se puede observar claramente cómo el Median-Cut y el de componentes principales pierden el color rojo que indica la población, mientras que el método propuesto lo mantiene. Los tres métodos presentan pérdida de la tonalidad verde.

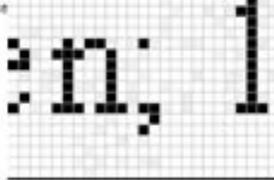
Resumen. Este capítulo se presentan los resultados experimentales del método descrito en el capítulo anterior. Los resultados demuestran la capacidad del método para obtener los colores más significativos de la imagen, llamados éstos los estudiamos, nuestro método mejoramos las técnicas clásicas de la escena.



Resumen. Este capítulo se presentan los resultados experimentales del método descrito en el capítulo anterior. Los resultados demuestran la capacidad del método para obtener los colores más significativos de la imagen, llamados éstos los estudiamos, nuestro método mejoramos las técnicas clásicas de la escena.



Resumen. Este capítulo se presentan los resultados experimentales del método descrito en el capítulo anterior. Los resultados demuestran la capacidad del método para obtener los colores más significativos de la imagen, llamados éstos los estudiamos, nuestro método mejoramos las técnicas clásicas de la escena.



Resumen. Este capítulo se presentan los resultados experimentales del método descrito en el capítulo anterior. Los resultados demuestran la capacidad del método para obtener los colores más significativos de la imagen, llamados éstos los estudiamos, nuestro método mejoramos las técnicas clásicas de la escena.

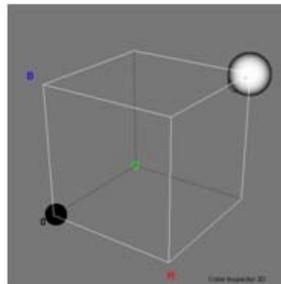
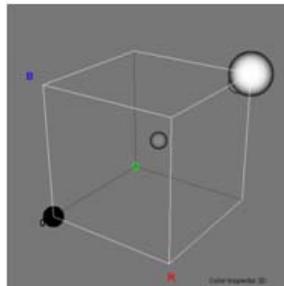
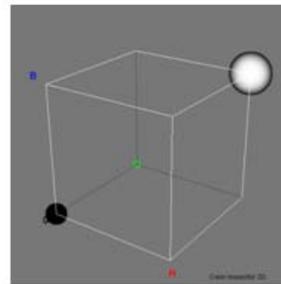
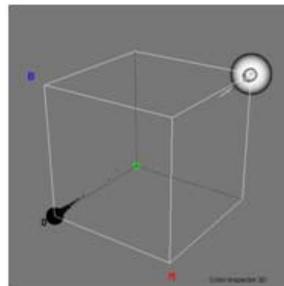


Fig. 18 Texto 1

De 159 colores se redujo a 5 colores, Como era de esperar, el método propuesto casi alcanza la binarización de la imagen que es lo que el ojo humano esperaría. En las ampliaciones hechas, se observa una mayor uniformidad del fondo con el método propuesto que con los otros dos métodos.

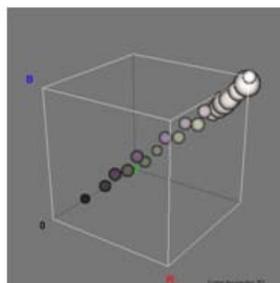
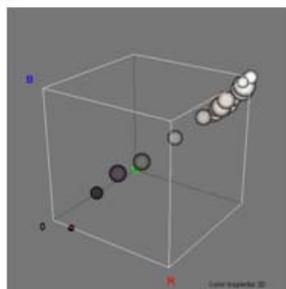
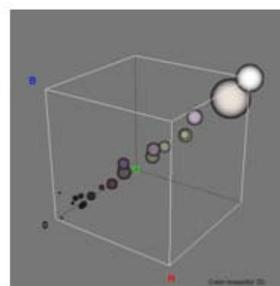
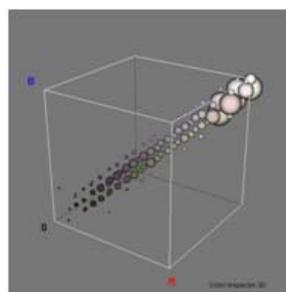
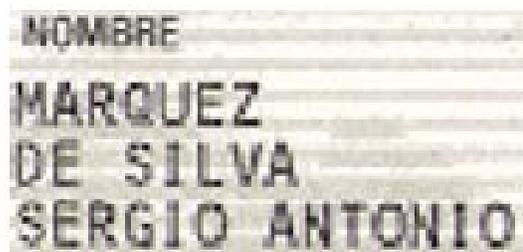
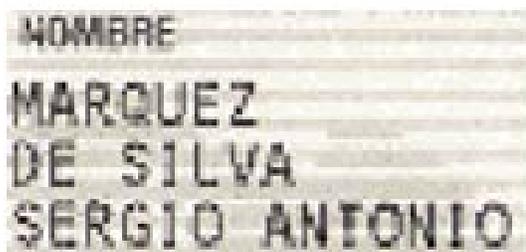
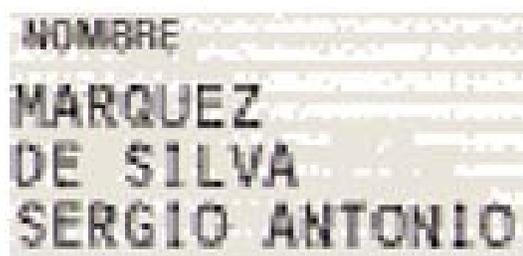
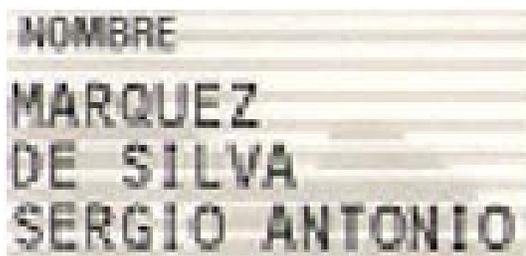


Fig. 19 Texto 2

De 398 colores se redujo a 41 colores. Como hemos observado, los métodos clásicos tienden a uniformizar el histograma de la imagen, lo que no es deseable para todo tipo de aplicaciones y de imágenes, pues las frecuencias de ocurrencia de los colores se “nivelan”, mientras que por el método propuesto se reconocen los colores más dominantes, sin perder aquellos con baja frecuencia de ocurrencia en la imagen, siempre y cuando estén concentrados en una región.

VI. CONCLUSIONES

En este reporte hemos presentado un resumen de los algoritmos de cuantificación del color más empleados en la actualidad y se ha hecho la introducción de un nuevo algoritmo que ha demostrado ventajas sobre los mismos. Sin embargo, hacemos notar que el tiempo de cómputo de este último es mayor que el de los otros. Las ventajas del nuevo método son:

Automatización: La cantidad de colores obtenidos en la paleta de colores son independientes del nivel de tolerancia elegido según las propiedades intrínsecas en la imagen, propiedades que escapan del alcance de los métodos clásicos.

Eficacia: La descripción cualitativa de una imagen en colores hecha por el sistema visual humano es muy precisa. Los pequeños detalles de una imagen, como los ojos del mandril o los símbolos en el Mapa 1, son claramente percibidos por nuestro sistema visual.

Agradecimientos

Los autores de este reporte técnico agradecen al Centro de Investigación en Computación (CIC), México; a la Secretaria de Investigación y Posgrado (SIP), México; y al Instituto Politécnico Nacional (IPN), México, por su apoyo en la realización de la presente investigación.

Referencias

- [1] R.C. Gonzalez, R.E. Woods, *Tratamiento Digital de Imágenes*, Addison – Wesley /Díaz de Santos, Wilmington (EE.UU.), 1996.
- [2] K.N. Plataniotis, A.N. Venetsanopoulos, *Color Image Processing and Applications*, Springer, Berlin (Alemania), 2000.
- [3] M. Domański, M. Bartkowiak, Compression, *The Colour Image Processing Handbook*, editado por S.J. Sangwine, R.E.N. Horne, Chapman & Hall, Cambridge (Gran Bretaña), 1998.
- [4] A.K. Jain, W.K. Pratt, Color Image Quantization, *Proceedings National Telecommunications Conference*, IEEE Publication, Houston (EE.UU.), 1972.
- [5] M. Orchard, C. Bouman, Color Quantization of Images, *IEEE Trans. On Signal Processing*, vol. 39, nº 12, p. 2677-2690, diciembre 1991.
- [6] Y. Linde, A. Buzo, R. Gray, An Algorithm for Vector Quantizer Design, *IEEE Trans. on Communications*, vol. COM-28, nº 1, p. 84-95, enero 1980.
- [7] P. Sheunders, A Genetic Approach towards Optimal Color Image Quantization, *IEEE Proceedings of Int. Conf. on Image Processing*, p. 1031-1034, Lausanne (Suiza), 1996.
- [8] P. Heckbert, Color Image Quantization for Frame Buffer Display, *Computer Graphics*, vol. 16, nº 3, p. 297-307, 1982.
- [9] R. Balasubramanian, C.A. Bouman, J.P. Allebach, Sequential Scalar Quantization of Vectors, an Analysis, *IEEE Trans. on Image Processing*, vol. 4, nº 9, p. 1282-1295, septiembre 1995.

- [10] L. Lucchese, S.K. Mitra, Color Segmentation Based on Separate Anisotropic Diffusion of Chromatic and Achromatic Channels, *IEE Proceedings Vision, Image and Signal Processing*, vol. 148, n° 3, p. 141-150, junio 2001.
- [11] W.K. Pratt, *Digital Image Processing*, 3ª edición, Wiley, Nueva York (EE.UU.), 2001.
- [12] B.W. Kolpatzik, C.A. Bouman, Optimized Universal Palette Design for Error Diffusion, *Journal of Electronic Imaging*, vol. 4, n° 2, p. 131-143, 1995.
- [13] S. Wan, S. Wong, P. Prusinkiewicz, An Algorithm for Multidimensional Data Clustering, *ACM Trans. on Mathematical Software*, vol. 14, n° 2, p. 153-162, 1988.
- [14] A.R. Weeks, G.E. Hague, Color Segmentation in the HIS Color Space Using the K-means Algorithm, *Proc. of the SPIE*, vol. 3026, p.143-154, 1997.
- [15] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, Nueva York (EE.UU.), 1981.
- [16] Floyd, R. W. and Steinberg, L., An adaptive algorithm for spatial gray scale, in SID, Ed., *Int. Symp. Dig. Tech. Papers*, 36, 1975.
- [17] <http://sipi.usc.edu/services/database/Database.html>
- [18] Begoña Acha Piñero. Segmentación y clasificación de imágenes en color. Aplicación al diagnóstico de quemaduras Sevilla, 2002.
- [19] X. Wu. Color quantization by dynamic programming and principal analysis. *ACM Transactions on Graphics*, 11(4):348–372, October 1992.
- [20] Ramos García Mario Eduardo y Felipe Riverón Edgardo, Mejoramiento de imágenes digitales impresas en semitonos, Tesis de Maestría, CIC México D.F. 2002.
- [21] Luo M. R., Cui G., and Rigg B.: The development of the CIE 2000 colour difference formula: CIEDE2000, *Color Res. Appl.*, 26(5), (2001) 340–350.