

## **Capítulo 5. Diseño del modelo de indexación de imágenes basado en una ontología**

Como se vio anteriormente, en el Capítulo 2 se analizaron las características de diversos métodos de representación de imágenes y en el Capítulo 3, de técnicas de representación de formas. Las conclusiones de estos capítulos mencionan aquellos que se eligieron, evaluando su rapidez, eficiencia, facilidad de implementación y obtención de buenos resultados, para formar el modelo de indexación de imágenes que se sugiere en éste trabajo de investigación. En base a las ventajas de estas técnicas y métodos seleccionados, además de añadir la comparación por color, éstos se pueden integrar junto con el uso de una estructura ontológica (ver Capítulo 4) y facilitar el manejo de la representación y clasificación de la semántica de la imagen, para desarrollar un modelo de indexación de formas que arroje buenos resultados en una forma rápida, sencilla y fácil.

Ésta tesis se enfoca a la recuperación de imágenes basándose en la forma y el color, en donde cada imagen tiene un significado por sí misma utilizando una estructura ontológica.

### **5.1 Modelo de recuperación de imágenes sugerido**

El modelo sugerido en este trabajo de tesis, define la siguiente secuencia de procesos, que integran el uso de métodos y técnicas para la representación de imágenes y de formas, para realizar la recuperación de imágenes. El modelo tiene dos tipos de consultas, cuando la consulta es una imagen y cuando se da una consulta por texto.

En una consulta por imagen:

- A. Se da una imagen de entrada: La imagen no debe ser una escena (donde existen varios objetos) y debe estar preprocesada para así obtener fondo plano y un sólo objeto. Estas limitaciones no influyen en la calidad del funcionamiento del modelo.
- B. Se obtiene el histograma de color de la imagen, que forma el vector de características de color. (Como se explica en sección 5.4.1)
- C. Se comparan los histogramas de color de imágenes, para obtener las distancias de los vectores de características de color. En este paso es un filtro de color que reduce la cantidad de imágenes a procesar en base de datos de imágenes.
- D. Se detecta el contorno del objeto de la imagen binarizada o en tonos de grises, con el algoritmo basado en la tortuga de Papert (Descrito en sección 5.3.2.1).
- E. Se aplica la técnica de evolución de la curva para simplificar la complejidad del contorno del objeto.
- F. Se representa la curva evolucionada por medio de la función de ángulo de giro, del método de representación de espacio tangencial, obteniendo una gráfica escalonada, fácil de describir y utilizar para realizar las comparaciones. (Como se define en sección 5.3.2.3)

- G. Se obtienen los vectores de características de forma, la cual el usuario puede indexar y describir para su posterior almacenamiento en la estructura ontológica y en la base de datos de imágenes.
- H. Se realiza la correspondencia entre el vector de características de la imagen de entrada y los de vectores de características, de imágenes previamente indexadas y filtradas por color.
- I. Se obtienen las distancias Euclidianas de ambos vectores de características de forma, comparando las funciones de representación de espacio tangencial de las imágenes. Posteriormente estas distancias se ordenan para determinar el orden de similitud entre imágenes recuperadas. (Como se explica en sección 5.3.3).
- J. A partir de la imagen más similar que se recupera se obtiene su término ontológico y se recuperan las imágenes con ese término ontológico, ordenadas por su similitud.
- K. Se obtiene la salida: Visualización de todas imágenes similares.

Para una consulta por texto:

A partir de la consulta se obtiene el término ontológico, la secuencia de pasos se inicia desde A, tomando como imagen de entrada, la imagen indexada más representativa para ese término ontológico, obteniendo los vectores de características de forma del paso G y se omite el paso J.

Para probar el funcionamiento de la integración de los métodos y técnicas elegidas que se mencionaron se diseñó una aplicación, la cual implementa la secuencia de procesos que define el modelo especificado anteriormente. En las siguientes secciones de este capítulo se describe la manera en como se diseñó esta aplicación.

## **5.2 Implementación del modelo de recuperación de imágenes.**

Para iniciar el sistema, el usuario introduce una consulta ya sea por imagen o por texto. Si la consulta es por imagen, se extraen las características de forma y color de la imagen. Con esos vectores de características se busca la similitud con otros vectores de características que fueron obtenidos previamente de la colección de imágenes clasificadas en la ontología; el conjunto de imágenes con mayor similitud es indexado y posteriormente se visualizan las imágenes ordenadas ascendentemente, la más similar para el sistema, se muestra primero y así sucesivamente.

Para el caso de una consulta por texto, a través del vocabulario ontológico estructurado se hace la selección de imágenes representativas, dependiendo del término definido por el usuario que es también una clase en la ontología, estas imágenes representan a las subclases de la clase (término) que consultó el usuario. A partir de estas imágenes, se hace la comparación de sus vectores de características con los vectores de características de las imágenes clasificadas en estas mismas subclases. De esta manera se obtendrán las distancias de todas las imágenes

pertenecientes a la clase que el usuario consultó. Cada clase esta formada de subclases y estas pueden tener más subclases, las subclases más internas son las que definen las imágenes, es decir, contienen almacenadas las imágenes y sus respectivos vectores de características. A continuación se definen los casos de uso que se representan las acciones del sistema.

### 5.2.1 Diagramas de casos de uso

Para entender los requerimientos funcionales del sistema, los siguientes casos de uso especifican colecciones de posibles situaciones respecto al uso del sistema. En la Figura 5.1 se observa el diagrama de caso de usos del sistema y a continuación se describen cada uno de ellos.

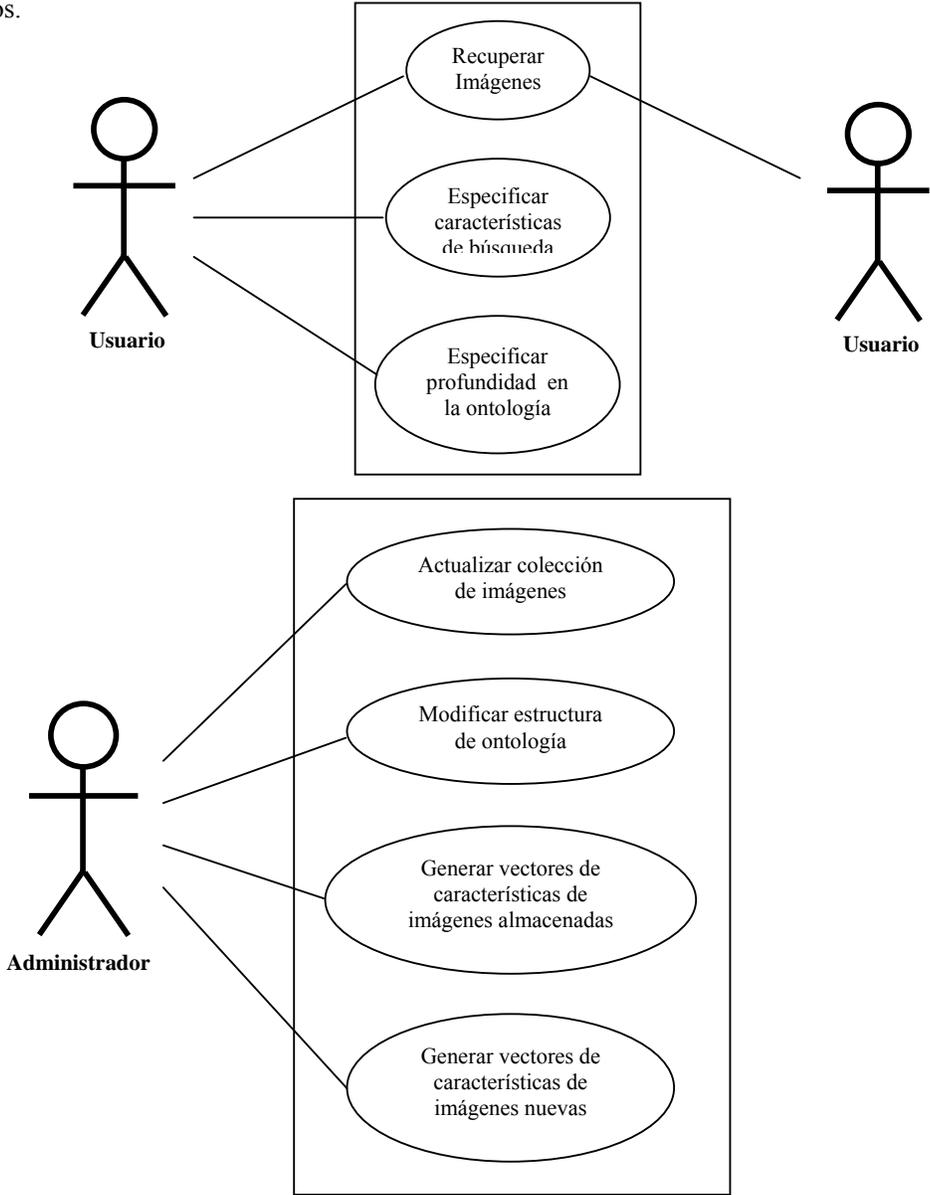


Figura 5.1: Diagrama de casos de uso del sistema

### **5.2.2 Recuperación de imágenes**

Éste caso de uso se inicia cuando un usuario realiza una búsqueda, ya sea por imagen o por texto. La consulta por medio de una imagen, representa un escenario, el usuario elige una imagen con los requisitos previos para ser analizada ó puede escribir una anotación representando la clase de imagen que quiere recuperar. El sistema retorna el conjunto de imágenes ordenadas de acuerdo a su similitud.

### **5.2.3 Especificar características de búsqueda**

Se inicia al especificar una búsqueda. El usuario define como hacer la comparación de la búsqueda ya sea por característica de forma, por color o por ambas características. Esto afecta la manera en que se realiza la comparación entre imágenes, para su recuperación.

### **5.2.4 Especificar profundidad en la ontología**

El usuario inicia este caso de uso, al realizar algún cambio en la profundidad definida de la ontología para buscar imágenes, en una consulta por imagen.

### **5.2.5 Modificar estructura de ontología**

El administrador inicia este caso de uso, cuando modifica la estructura de la ontología, ya sea agregando o eliminando clases y/o modificando los términos definidos en la estructura.

### **5.2.6 Actualizar colección de imágenes**

El actor de este caso de uso es el administrador del sistema, se inicia cuando éste agrega imágenes a una determinada clase de la ontología, dichas imágenes deben estar previamente procesadas con las características que se definieron anteriormente.

### **5.2.7 Generar vectores de características de imágenes almacenadas**

Este caso de uso se inicia por el administrador, cuando genera nuevamente los vectores de características de todas las imágenes almacenadas y clasificadas en la ontología.

### **5.2.8 Generar vectores de características de imágenes nuevas**

El administrador inicia este caso de uso, generando vectores de características de una imagen nueva que se va a almacenar en la ontología.

### 5.3 Diagrama de componentes

A continuación se muestra el diagrama de componentes (Figura 5.2), que describe el funcionamiento del sistema y el flujo del proceso a seguir, para realizar una búsqueda de imágenes.

Existen sistemas de recuperación de imágenes basados en características de bajo nivel (línea punteada roja en Figura 5.1). El modelo propuesto agrega la definición de semántica a nivel de descripción de objetos, (línea punteada azul Figura 5.2).

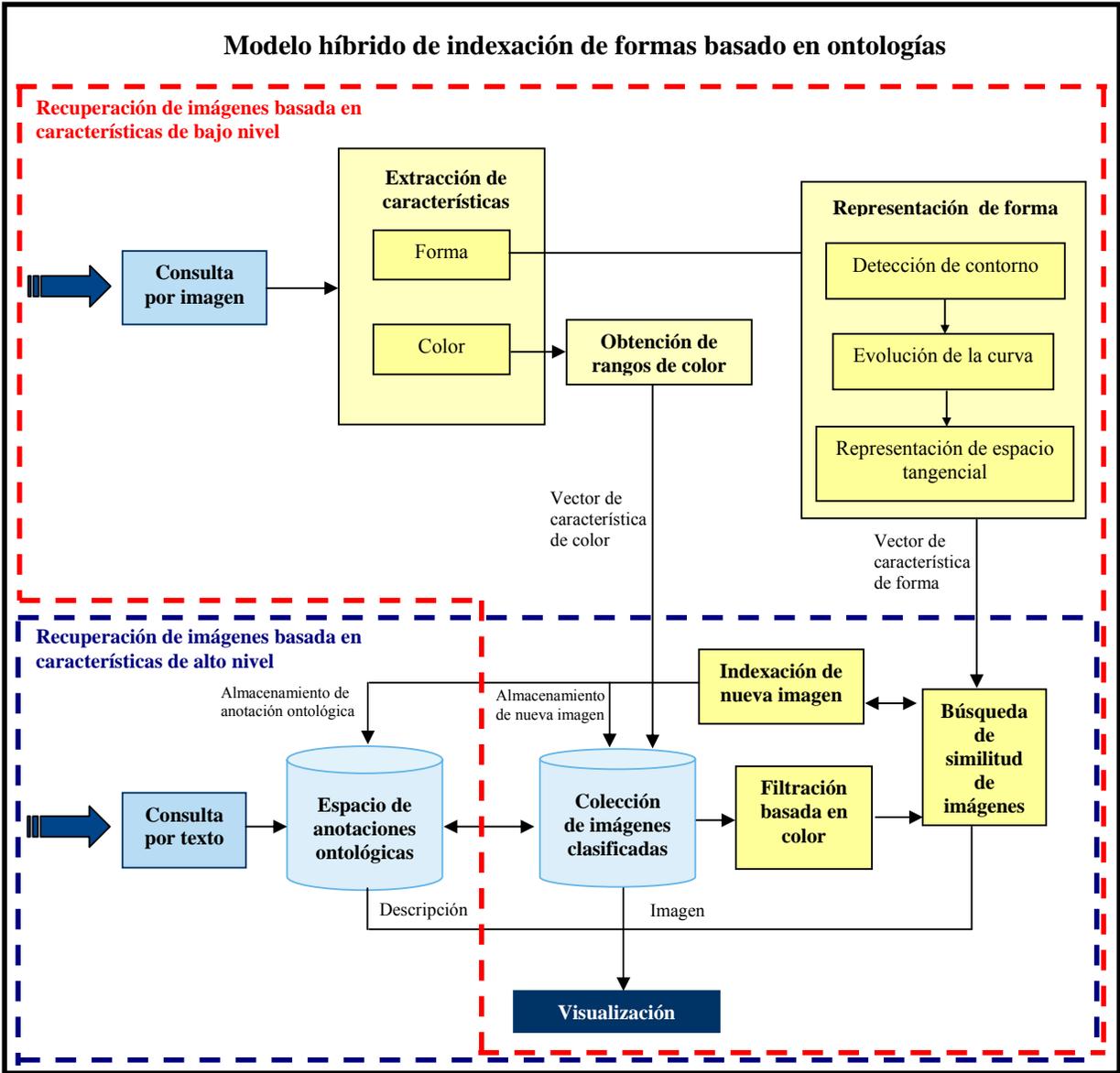


Figura 5.2: Diagrama de componentes del sistema.

## 5.4 Recuperación basada en forma

A continuación se hace la descripción de los procesos necesarios para la recuperación de imágenes al realizar una consulta.

Primero nos enfocaremos en la recuperación de forma y el proceso que se debe seguir para detectarla, representarla y finalmente compararla con otras.

### 5.4.1 Requerimientos de la imagen suministrada, previa a ser analizada.

- La imagen debe ser formato \*.jpg o \*.bmp.
- La imagen ya debe estar preprocesada. La imagen se debe encontrar en un estado conveniente para realizar los procesos, es decir, la imagen debe contener un solo objeto y tener fondo plano.

Las consultas se pueden especificar por medio de un ejemplo visual o por medio de texto. Generalmente en las consultas que se realizan por medio de un ejemplo visual, los resultados obtenidos son más exactos. Esto se debe a que no dependen de las anotaciones definidas en la ontología, que para un usuario pueden ser ambiguas. El tipo de búsqueda por ejemplo, depende totalmente de la forma de la imagen, los resultados generados basándose en las comparaciones con ésta imagen ejemplo.

El proceso de recuperación de formas abarca tres conceptos: *representación de formas*, *medida de similitud entre formas* y la *indexación de formas*.

### 5.4.2 Descripción de forma.

La representación de forma basada en el contorno, se define de la información del borde de la forma.

Para realizar la comparación de la similitud entre formas, primero se detectó el contorno de la imagen para después representarlo, posteriormente se aplicó el algoritmo de evolución de la curva con el que obtenemos una forma más simple y fácil de describir. Tomando esta curva, se genera la función de espacio tangencial. Con esta función se representa la curva “evolucionada” de la forma, de una manera más óptima para la comparación entre formas. A continuación se describen detalladamente estos procesos.

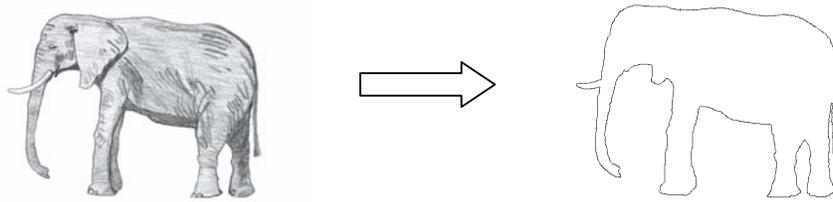
#### 5.4.2.1 Detección de contorno

La forma de las imágenes es el principal descriptor. Partiendo de una forma se genera el vector. De una imagen consulta, se obtienen las coordenadas del contorno aplicando el proceso de seguimiento de contorno, que está basado en el algoritmo de la tortuga de Papert.

La matriz que representa la imagen de entrada se obtiene calculando el RGB de cada píxel y almacenándolo en la matriz. El tamaño de la matriz es el mismo tamaño que el de la imagen a

analizar; a partir de esta matriz se obtendrá el contorno  $C$ . El objeto dentro de la imagen esta representado por el conjunto de píxeles consecutivos  $P$  que son diferentes del color de fondo de la imagen, por ejemplo el blanco, que en el sistema RGB está definido por la terna RGB (255, 255, 255).

El algoritmo de seguimiento de contorno se resume en una secuencia de pasos definida en el Apéndice A, sección A.1. La salida de éste algoritmo produce un conjunto de coordenadas  $C$ , que pertenecen al contorno de la imagen. Estas coordenadas (alrededor de 500 puntos) que representan al contorno se almacenan en una matriz, después se aplica un algoritmo de evolución de la curva, con el cual se reduce el conjunto de coordenadas (hasta 50 puntos).



**Figura 5.3: Ejemplo de detección de contorno aplicando el algoritmo basado en la tortuga de Papert.**

#### **5.4.2.2 Evolución de la curva**

Éste método se eligió por las características y ventajas que presenta (ver Capítulo 3). La idea del algoritmo de evolución de la curva es reducir el conjunto de vértices  $C$  de un polígono, a un conjunto  $A$  con la información más importante del contorno original (entonces  $A \subset C$ ). El contorno  $C$ , que se obtuvo a través de la búsqueda de contorno, contiene mucha información detallada e irrelevante de la forma, la cual se debe reducir para facilitar el reconocimiento de formas.

La base de éste método es asignar una medida de relevancia a cada vértice. Éste método se ha utilizado para reducir la influencia de ruido y simplificar las formas, removiendo características irrelevantes de la forma y manteniendo los puntos que contienen información relevante. Iterativamente compara la medida de relevancia de todos los vértices del polígono. La mayor relevancia significa que el vértice tiene una contribución mayor a la curva de forma. En cada iteración se repite el proceso, el vértice que tiene la menor relevancia es eliminado y sus vértices adyacentes son unidos formando un nuevo segmento. Se repite la eliminación hasta que obtenemos el estado de abstracción deseado. El algoritmo de evolución de la curva se define en el Apéndice A, sección A.2.

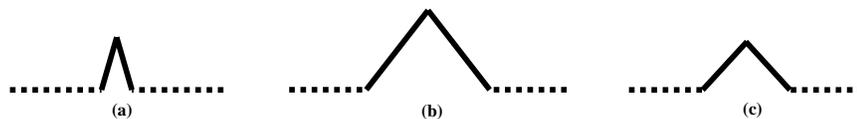
El conjunto de vértices es reducido de un polígono, a un subconjunto de vértices que contiene la información relevante del polígono original, formando otro polígono.

### 5.4.2.2.1 Medida de relevancia.

La medida de relevancia se basa en la longitud relativa y en el ángulo de giro de los segmentos adyacentes. Se puede expresar como:

$$K(S_1, S_2) = \frac{\beta(S_1, S_2)l(S_1)l(S_2)}{l(S_1) + l(S_2)} \quad (5.1)$$

Donde  $\beta(S_1, S_2)$ , es el ángulo de giro en el vértice común de los segmentos adyacentes  $S_1$  y  $S_2$ ,  $l$  es la longitud normalizada con respecto a la longitud total del polígono. La medida de relevancia es directamente proporcional al ángulo de giro y a la longitud del segmento de la curva. El valor menor de  $K(S_1, S_2)$ , significa la menor contribución a la curva, del punto común donde se unen los segmentos  $S_1$  y  $S_2$ .

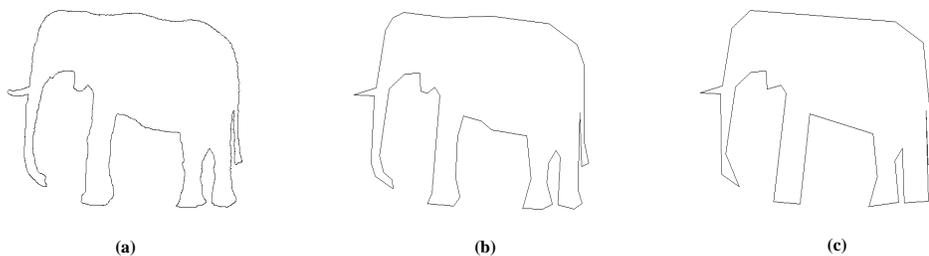


**Figura 5.4: Vértices con diferentes medidas de relevancia. (c) Vértice con menor relevancia ya que tiene menor longitud que (b) y presenta un ángulo de giro menor que (a). El vértice (a) tiene la mayor relevancia de éstos vértices por mayor su ángulo de giro.**

### 5.4.2.2.2 Análisis del algoritmo de evolución de la curva

Este algoritmo sólo funciona si los vértices son eliminados en el orden correcto, de aquí la importancia de la medida de relevancia definida en la ecuación 5.1.

Para verificar el orden correcto de eliminación de vértices, se probó con una figura pequeña de 5 vértices (ver Figura A.1 en Apéndice A). Se obtuvieron buenos resultados, ya que visualmente se pueden determinar que vértices se deberán eliminarán y cual será la figura final que se obtendrá. El resultado de aplicar el método de evolución de la curva, es una nueva curva  $A$  reducida, la cual contiene los vértices con información importante para definir la forma de la imagen, como se muestra en la Figura 5.5.



**Figura 5.5: Ejemplo de evolución de la curva, de (a) Contorno de la imagen original, (b) Evolución a 50 segmentos y (c) Evolución a 30 segmentos.**

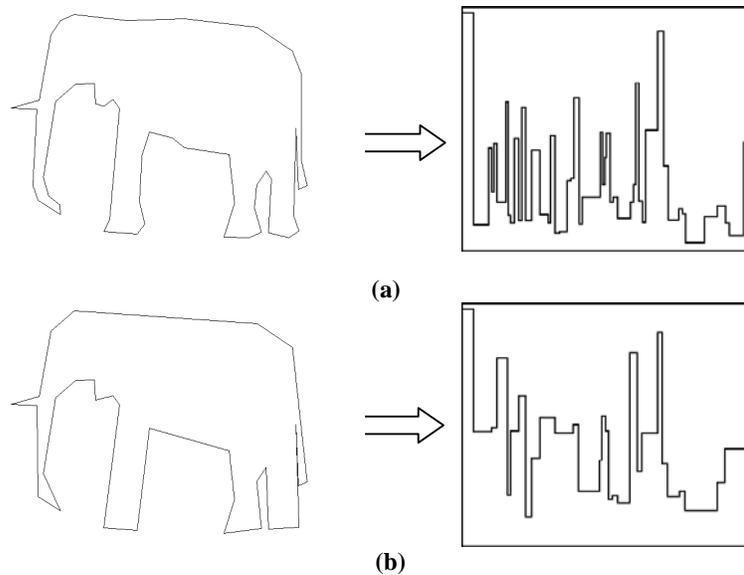
### 5.4.2.3 Representación del polígono a través del dominio de espacio tangencial

Para la representación de polígonos, se eligió la función de representación de espacio tangencial, utilizada en muchas otras investigaciones [79, 80] para geometría computacional y visión por computadora.

Esta función presenta muchas propiedades que la hacen adecuada para la comparación entre imágenes. Se parte de un polígono que se obtiene basándose en la forma del objeto como se mencionó en Capítulo 3. Esta función siempre es constante en los cálculos para los polígonos, lo que la hace fácil y rápida en su evaluación.

En el Apéndice A, sección A.4 se describe el método de obtención de la función de ángulo de giro (Turning Function), también llamada representación de espacio tangencial.

Esta representación de función se utiliza para calcular la distancia de comparación entre dos polígonos (A y B), buscando la distancias  $L_p$  entre sus respectivas funciones de ángulo de giro  $\theta_A(s)$  y  $\theta_B(s)$ .



**Figura 5.6: Ejemplo de funciones de espacio tangencial a partir de polígonos con (a) 50 segmentos y (b) 30 segmentos.**

En el Apéndice A en la sección A.5 se muestran algunos ejemplos de imágenes con sus correspondientes funciones de la representación tangencial.

#### 5.4.2.3.1 Invarianzas de la función de representación de espacio tangencial

La representación de la función tangencial tiene las características de ser invariante a traslación, escalamiento y rotación. A continuación se muestran algunos ejemplos de estas invarianzas.

- Invarianza a rotación.** Esta característica se presenta debido a que la selección del vértice de inicio siempre es la misma, no importando el sentido y magnitud de la rotación. El criterio de selección del primer vértice es el vértice con el mayor valor de relevancia obtenido con la ecuación 5.1. Las funciones de espacio tangencial de las Figuras 5.7 y 5.8 presentan diferencias, debido a que las rotaciones modifican la configuración de los píxeles, alterando de esta forma, la representación del contorno y las funciones obtenidas.

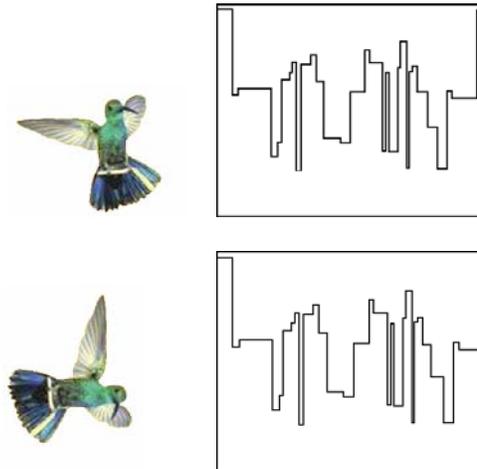


Figura 5.7: Ejemplo de invarianza a la rotación, del método de representación tangencial de figuras rotadas.

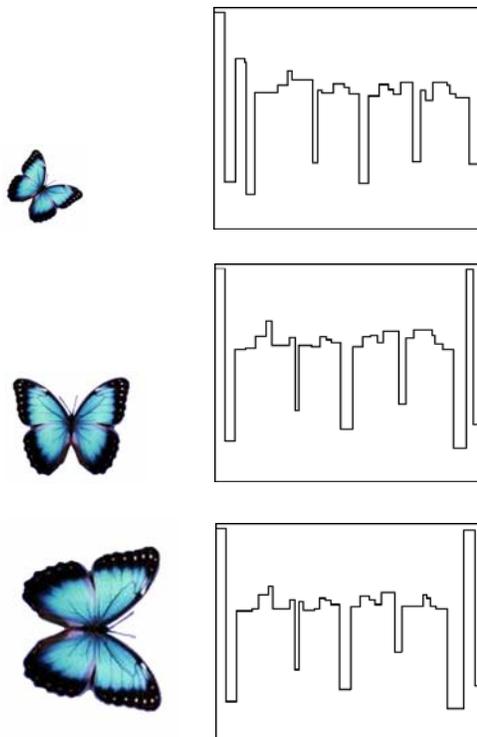
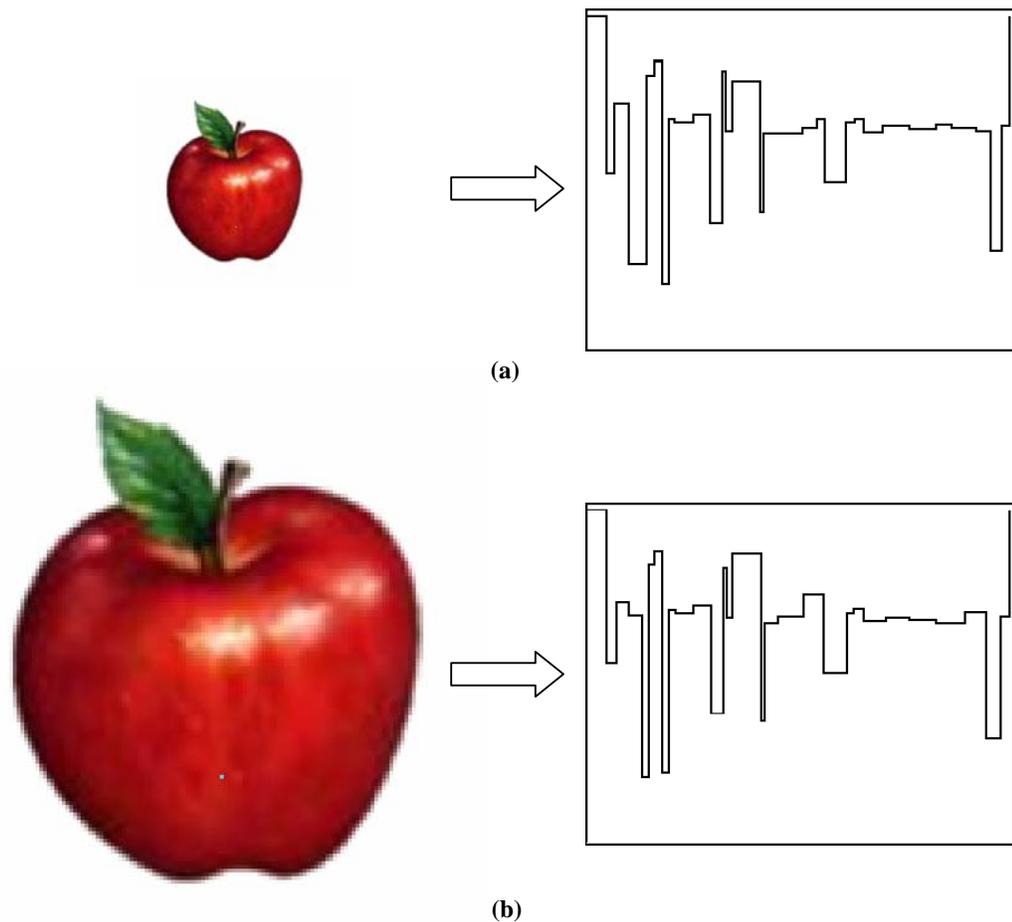


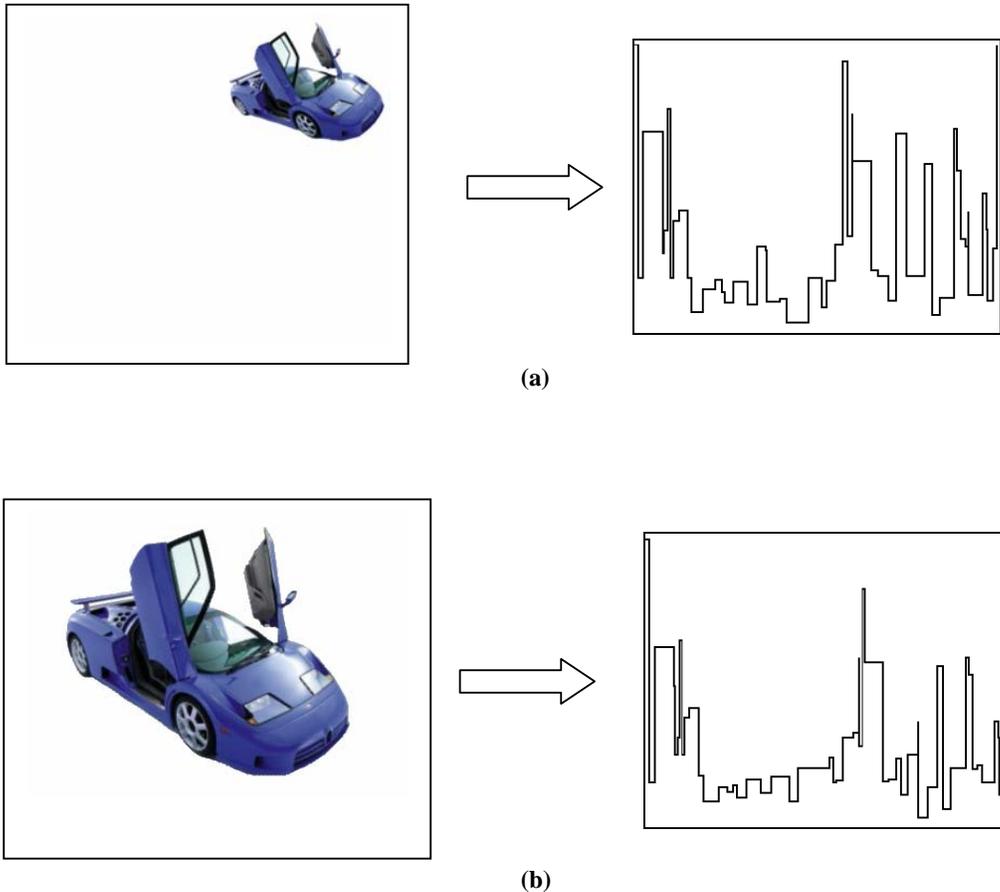
Figura 5.8: Ejemplo de invarianza a la rotación, del método de representación tangencial de figuras rotadas.

- **Invarianza a escalamiento.** Esto se debe a que la función de espacio tangencial se basa, tanto en el ángulo de giro entre dos segmentos consecutivos, como en la longitud de cada segmento. El ángulo de giro es el mismo, aún cuando los segmentos incrementan o reducen su tamaño, ya que la longitud del segmento es normalizada con respecto al perímetro total de la forma obtenida. Las variaciones de las funciones se deben al escalamiento, que modifica la configuración de los píxeles actuales. La longitud de toda la curva es igual a 1. A continuación se muestra un ejemplo en la Figura 5.9.



**Figura 5.9: Ejemplo de invarianza a escalamiento del método de representación tangencial, (a) imagen original, (b) imagen escalada en un 300%.**

- **Invarianza a traslación.** La función también es invariante a traslación, ya que es relativa a la posición del primer vértice. Esto permite obtener la misma representación de la función de espacio tangencial, para un conjunto de formas similares, aún cuando estas últimas se localicen en diferentes posiciones. Como se muestra en la Figura 5.10.



**Figura 5.10: Ejemplo de invarianza a traslación del método de representación tangencial, (a) Imagen escalada a un 30% , rotada 15° a la izquierda y trasladada a la esquina superior, (b) imagen original.**

### 5.4.3 Medida de similitud entre formas

Para el medir la similitud entre la forma A y la forma de consulta B, se comparan la forma A, la cual es almacenada como un modelo para un objeto en particular, con la forma B. Si A y B son parecidas, el sistema debe reportar la correspondencia que existe y la medida que define que tan parecidas son.

Para definir la función de *costo*, asociada con la distancia entre dos formas A y B, se obtiene  $d(A, B)$ , que mide su similitud.

Para comparar polígonos, basándose en la similitud de medida, se deben cumplir las siguientes propiedades con respecto a  $d(A, B)$ :

- Debe ser invariante a traslación, rotación y a escalamiento.
- Fácil de calcular.
- La correspondencia obtenida debe concordar con la intuición humana sobre la similitud de formas.
- Métrica:

**Tabla 5.1: Propiedades de la distancia obtenida.**

Propiedad	Descripción
$d(A, B) = 0$	Sólo si $A = B$ , cuando una forma se parece a sí misma
$d(A, B) \geq 0$	Para toda A y B.
$d(A, B) = d(B, A)$	Para toda A y B, el orden de comparación no debe importar.

Para obtener la medida de similitud ente dos funciones de espacio tangencial,  $\Theta A$  y  $\Theta B$ , existen dos métodos, uno es calculando el área mínima entre de las funciones y otro es a través del cálculo de la distancia Euclideana de cada uno de sus puntos.

Para obtener el área, las dos funciones primero deben ser alineadas, debido a las diferencias en la forma. La distancia se obtiene haciendo una búsqueda de dos dimensiones para encontrar la distancia mínima. El cálculo sería:

$$d(A, B) = \|\Theta A - \Theta B\|_2 = \sqrt{\left(\int_0^1 |\Theta A - \Theta B|^2 ds\right)} = \sqrt{\min_{\theta \in \mathbb{R}, t \in [0,1]} \left(\int_0^1 |\Theta A(s+t) - \Theta B(s) + \theta|^2 ds\right)} \quad (5.2)$$

Este método obtiene buenos resultados pero tiene la desventaja de ser más complejo y por lo tanto más lento en su ejecución. Además tiene la desventaja de obtener mismas áreas para funciones totalmente diferentes.

En el método de la distancia Euclideana, se tienen dos vectores que corresponden a los valores de Y para cada X, para ambas funciones de espacio tangencial. Se tienen dos vectores  $S_A = [a_1, a_2, a_3 \dots a_n]$  y  $S_B = [b_1, b_2, b_3 \dots b_n]$ , donde cada valor  $a_1, a_2, a_3 \dots a_n$  y  $b_1, b_2, b_3 \dots b_n$ , son los valores de y que corresponden a cada posición x, donde  $x = 1, 2, 3, \dots, n$ , de las respectivas funciones de espacio tangencial A y B gráficas. Estos valores de los vectores obtenidos, se introducen a la ecuación 5.3 para obtener la distancia Euclideana.

El método de distancia Euclideana tiene la ventaja de ser más rápido y menos complejo en su ejecución, es por esto que se eligió este método para el cálculo de distancias entre polígonos. A continuación se explica más a detalle.

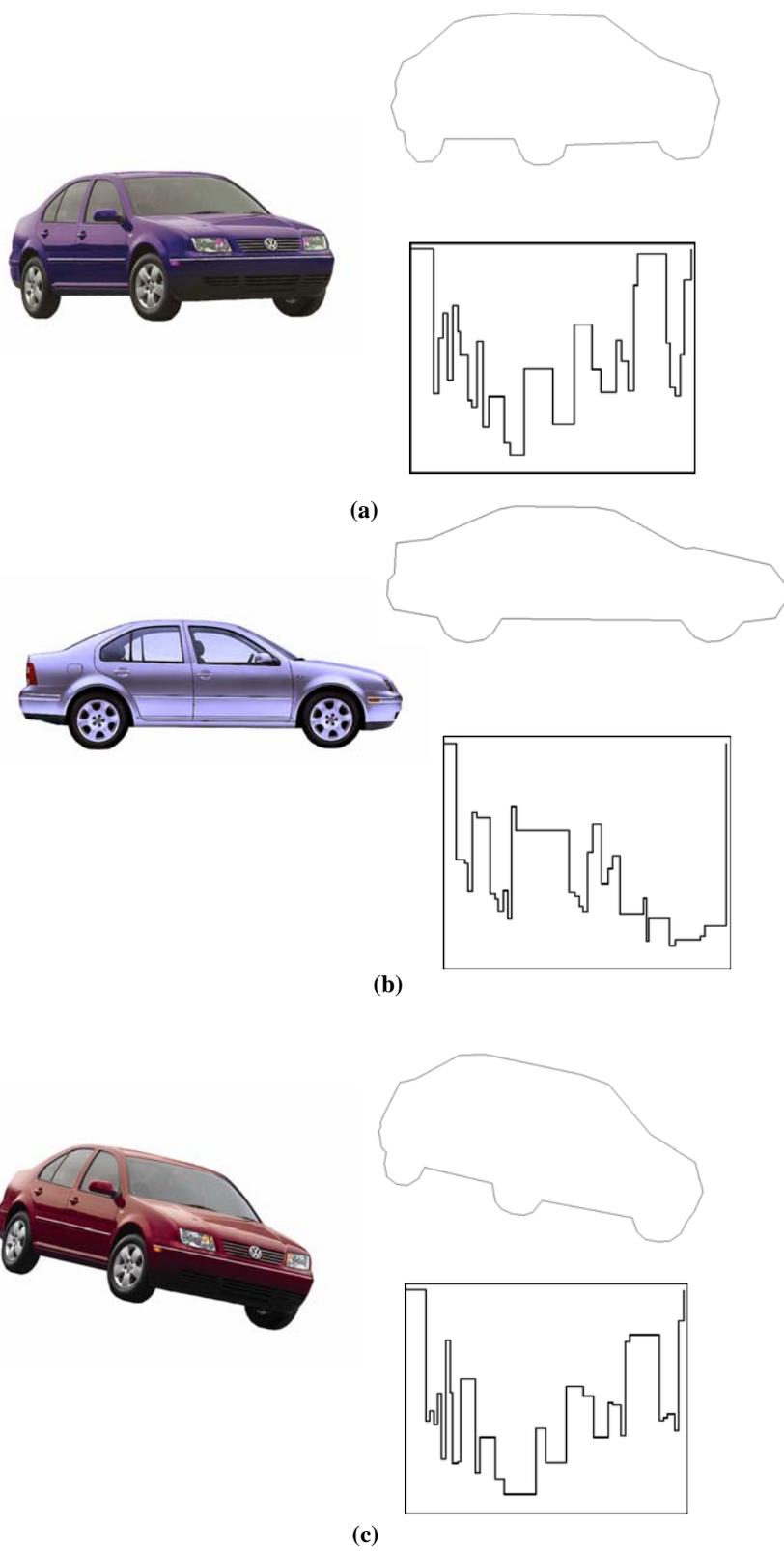
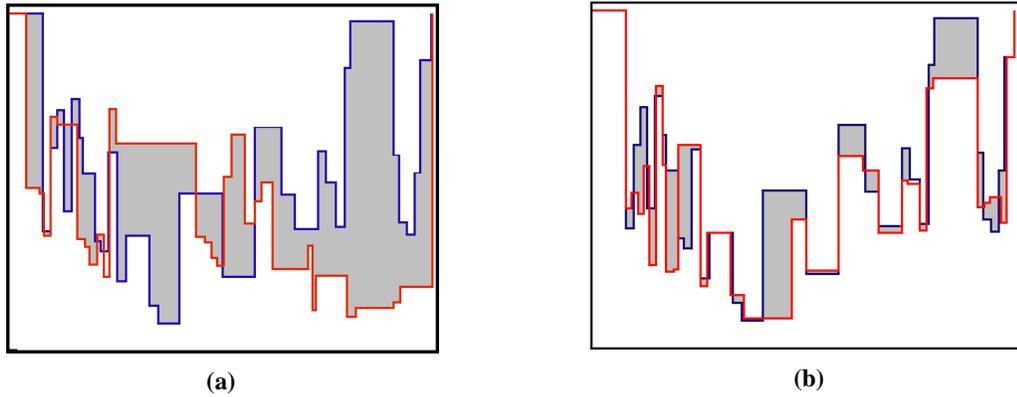


Figura 5.11: Ejemplos de imágenes y sus respectivas funciones de espacio tangencial.

En la figura anterior, visualmente las imágenes (a) y (b), son más parecidas que la combinación de otro par, mostrando que sus respectivas funciones de espacio tangencial, también presentan mayor similitud. La imagen (b) en la Figura 5.12, presenta menor distancia entre las formas, que la imagen (a). Se observa que existe menor cantidad de color gris, que representa la distancia entre formas.



**Figura 5.12: Comparación de imágenes por medio de sus funciones de espacio tangencial.**

Para hacer la correspondencia entre dos formas A y B, se almacenan sus características como 2 vectores unidimensionales que se expresan como  $S_A = [a_1, a_2, a_3, \dots]$  y  $S_B = [b_1, b_2, b_3, \dots]$ . Para comparar la distancia en el espacio de características de ambos vectores, se calcula la distancia Euclidiana, la cual determina que tan parecidas son las dos imágenes. Esta distancia se define como:

$$\|d(A, B)\| = |S_A - S_B|^2 = \sqrt{\sum_{i=1}^n |a_i - b_i|^2} \quad (5.3)$$

Los vectores de características de las formas A y B, que se obtienen a partir de sus correspondientes funciones de espacio tangencial, corresponden a los valores de sus coordenadas  $x$  y  $y$ . Estas coordenadas se refieren a la longitud del segmento y al ángulo de giro entre dos segmentos de la forma, respectivamente.

La comparación de las imágenes en las Figuras 5.11, se presentan en la Figura 5.1. (a) representa la comparación de las imágenes Figura 5.11(a) y (b). La Figura 5.12 (b) es la comparación de las imágenes en Figura 5.11 (a) y (c). Como se observa en la Figura 5.12 (b), la cual se obtuvo por la comparación entre las imágenes (a) y (c) de la Figura 5.11, son más parecidas, ya que las distancias entre funciones presenta la menor parte sombreada.

Las imágenes más parecidas, obtienen valores menores de  $d(A, B)$ .

Partiendo de Figura 5.11 (a), se obtuvo el siguiente vector de características que contiene las coordenadas de los vértices de la función de espacio tangencial de dicha imagen. Cada par de números representan un par de coordenadas (x, y), donde x representa el valor en la gráfica de representación tangencial, de la longitud del segmento de la forma y y representa el valor del ángulo en ese punto x, en el dominio de unidades del plano cartesiano donde se grafica.

0 5	53 207	103 177	207 194	264 167	306 70
30 5	63 207	124 177	219 194	280 167	346 70
30 173	63 167	124 274	219 251	280 172	346 65
37 173	70 167	150 274	239 251	287 172	355 65
37 144	70 232	150 282	239 156	287 166	355 151
42 144	79 232	179 282	245 156	294 166	367 151
42 131	79 116	179 207	245 196	294 170	367 196
48 131	95 116	192 207	257 196	301 170	371 196
48 174	95 256	192 262	257 257	301 20	371 35
53 174	103 256	207 262	264 257	306 20	380 35
					380 5

Figura 5.13: Vector de características de forma de la Figura 5.11 (a).

## 5.5 Recuperación basada en color

### 5.5.1 Búsqueda por imagen

Para la comparación de color se utilizó una variación al método de histogramas, que es uno de los métodos más utilizados para describir propiedades de imágenes. Un histograma como se definió en el Capítulo 2, es la frecuencia con que aparece cada color en una imagen. Se utilizó el sistema RGB (rojo, verde y azul), por lo que se obtienen 3 histogramas de los tres canales respectivamente. El histograma representa la frecuencia en que aparece la tonalidad de cada color. Para el formato RGB (0 a 255) existen 256 tonos por cada color.

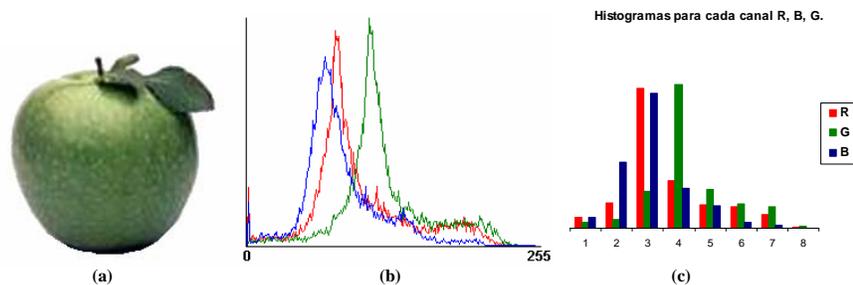
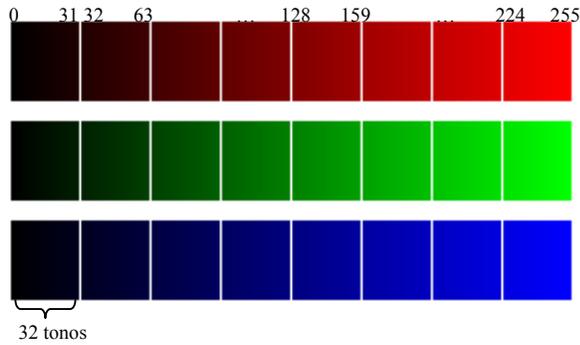


Figura 5.14: Histograma para los tres canales del sistema RGB.

Para una consulta por imagen, se obtiene el histograma respectivo de la imagen dividiendo cada canal RGB, como se observa en la Figura 5.14. A partir del histograma que se obtiene, se construye un vector representativo. Se divide el rango (0..255) de cada canal del sistema RGB, en 8 segmentos, cada uno con 32 tonos como se observa en Figura 5.15.

El rango para cada uno de los 8 segmentos en cada canal es: 0 – 31, 32 – 63, 64 – 95, 96 – 127, 128 – 159, 160 – 191, 192 – 223 y 224 – 255.

Posteriormente, se determina el número de tonos que coincidieron para cada uno de los 8 segmentos. El número de tonos representará al vector de características de color, que servirá para realizar la comparación entre estos vectores. El resultado de esta comparación indica la diferencia de color entre dos imágenes.



**Figura 5.15: División de los canales del sistema RGB para obtener el vector de características de color.**

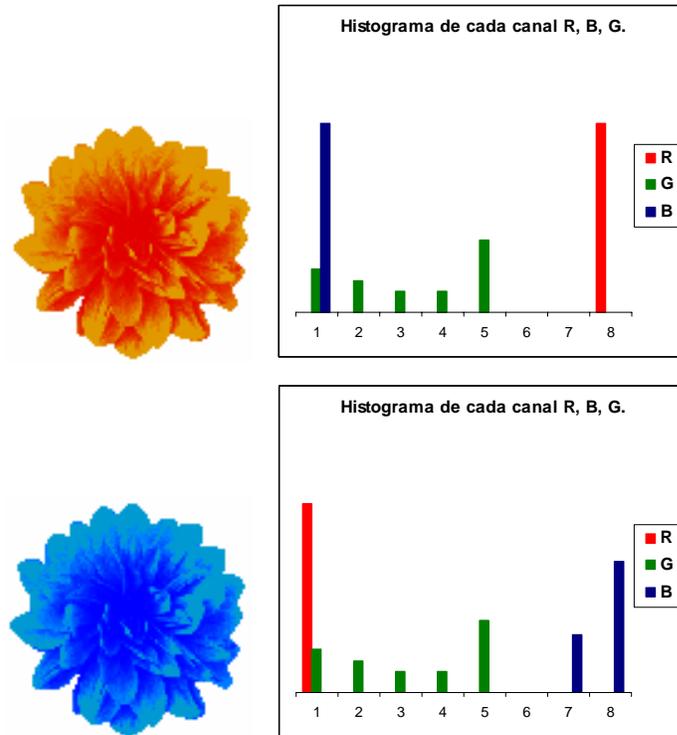
Partiendo de la Figura 5.14, se obtuvo el vector de características de color, el cual se muestra en la Figura 5.16. Cada columna indica un canal del sistema RGB y cada renglón, cada una de las 8 divisiones en los canales, como se muestra en la Figura 5.15.

0.0196921231507397	0.0112954818072771	0.0179928028788485
0.0303378648540584	0.0210915633746501	0.0375349860055978
0.20171931227509	0.0403338664534186	0.447321071571371
0.444722111155538	0.273840463814474	0.323970411835266
0.139894042383047	0.416533386645342	0.129548180727709
0.0783186725309876	0.115503798480608	0.0277389044382247
0.080217912834866	0.100909636145542	0.0158936425429828
0.00509796081567373	0.0204918032786885	0

**Figura 5.16: Vector de características de color de la Figura 5.14.**

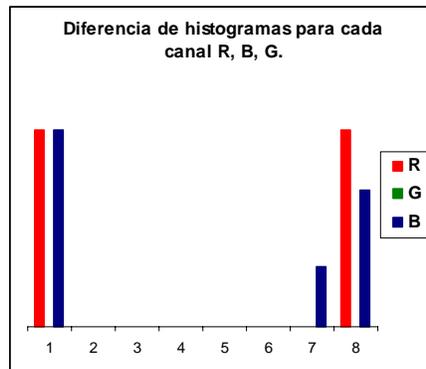
La distancia entre vectores de color se obtuvo de la misma manera que con la forma, en base a la distancia Euclidiana, calculada con la Ecuación 5.3. Los dos vectores de características a comparar se definen como  $S_A = [a_1, a_2, a_3, \dots]$  y  $S_B = [b_1, b_2, b_3, \dots]$ , cada término  $a_i$  y  $b_i$  se compone por 3 elementos que corresponden a los tres canales del sistema RGB. La distancia de los términos nos indica la distancia entre los canales de las imágenes comparadas. La menor distancia obtenida indica la imagen más parecida en color, respecto a la imagen consulta.

A continuación se muestra un ejemplo de comparación de gráficas de histogramas, para cada canal R, G, B, de las respectivas imágenes.



**Figura 5.17: Imágenes y su correspondiente gráfica de histogramas de cada canal R, G, B, dividida en 8 segmentos.**

Comparación entre gráficas de histogramas de las imágenes en la Figura 5.17.



**Figura 5.18: Distancias de segmento en que se dividen los valores de cada canal R, G, B de las imágenes en Figura 5.17.**

### 5.5.2 Búsqueda de color por texto

Para cada imagen, se obtienen los tres canales del sistema RGB de cada píxel y se determinaron rangos de canal para los tres colores primarios y las combinaciones entre estos: rojo, verde, azul, magenta, cian y amarillo. En una búsqueda por texto, el usuario especifica alguno de estos seis colores.

A continuación se definen los rangos que se deben cumplir para determinar el color del píxel analizado. Considerando que r = rojo, v = verde, a = azul.

Para el color rojo:

$$100 \leq r \leq 255, v \leq 70\% \text{ de } r, a \leq 70\% \text{ de } r.$$

Para el color verde:

$$100 \leq v \leq 255, r \leq 70\% \text{ de } v, a \leq 70\% \text{ de } v.$$

Para el color azul:

$$100 \leq a \leq 255, v \leq 70\% \text{ de } a, r \leq 70\% \text{ de } a.$$

Para el color amarillo:

$$200 \leq r \leq 255, 200 \leq v \leq 255, a \leq 100.$$

Para el color magenta:

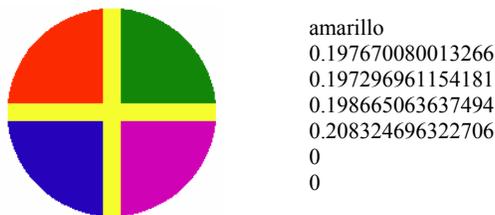
$$200 \leq r \leq 255, 200 \leq a \leq 255, v \leq 100.$$

Para el color cyan:

$$200 \leq a \leq 255, 200 \leq v \leq 255, r \leq 100.$$

Un contador, definido cada uno de los seis colores obtenidos, se incrementa cuando algún píxel analizado cumple las condiciones.

El contador con valor más grande, determinará el color que se asignará a la imagen. Para cada imagen analizada, los valores de los contadores y el nombre del color que se definió, se almacenan en un archivo, el cual define otro tipo de vector de características de color. Este tipo de vector de características se utilizará solo para las búsquedas por texto.



**Figura 5.19: Vector de características de color, utilizado para una búsqueda por texto.**

A partir de la circunferencia de la Figura 5.19, el primer renglón, indica el color que se le asignó a la imagen, los siguientes renglones definen la cantidad de colores de rojo, verde, azul, amarillo, magenta y cyan que tiene la imagen.

La comparación de estos vectores de características se realiza buscando el color que especificó un usuario, en el primer renglón de los archivos que representen a este vector de características. En una búsqueda primero se especifica qué es lo que se desea buscar y posteriormente, el término que defina al color de la imagen buscada. Por ejemplo: “Fruta amarillo”, “Transporte rojo”, “Polígono verde”, etc.

En este caso, las imágenes que se recuperan son aquellas, cuyo color almacenado, coincide con el color especificado en una consulta por texto por algún usuario.

## 5.6 Indexación de imágenes

La indexación de imágenes depende del tipo de búsqueda que se realice, si es una consulta visual o por texto, o también si se compara sólo la forma o forma y color.

En una *consulta por medio de un ejemplo visual*, se comparan los vectores de características (de forma y/o color) de la imagen del query, con los vectores de características de imágenes almacenadas y clasificadas en la ontología. Se seleccionan aquellas imágenes que tengan menor distancia, es decir, sean más similares en la(s) característica(s) seleccionadas (forma y/o color). Estas distancias se obtienen aplicando las Ecuaciones 5.4 y 5.5.

$$d_{EF}(f_1, f_2) = \sqrt{\sum (f_1 - f_2)^2} \quad \text{y} \quad d_{EC}(f_1, f_2) = \sqrt{\sum (f_1 - f_2)^2} \quad (5.4) \text{ y } (5.5)$$

$d_{EF}$  y  $d_{EC}$  son las distancias Euclidianas entre dos imágenes  $f_1$  y  $f_2$ , respecto a la forma y color respectivamente,

La distancia menor entre dos imágenes, determina la mayor similitud en los histogramas segmento y por consecuencia entre los colores. Además, si la imagen tiene exactamente la misma forma, la distancia será de 0.

Para calcular el porcentaje de similitud, se utilizó la distancia Euclideana. Al parámetro de forma se le asignó mayor relevancia, teniendo mayor peso que el parámetro de color. La distancia total se obtiene,

$$d_T = 2d_{EF} + d_{EC} \quad (5.6)$$

Una vez obtenidas todas las distancias Euclidianas, tanto de color como de forma, de la colección de imágenes, respecto a la imagen solicitada, éstas se combinan, dando mayor peso (el doble de su valor) como ya se mencionó, al parámetro de forma. En caso de que la búsqueda sea solamente por forma, el parámetro que mide el color tendrá un peso igual a 0. La determinación de darle mayor peso a la forma, fue que ésta, define en sí, al objeto de la imagen

con mayor precisión. El color por sí solo, no determina que objeto tiene una imagen. Las distancias se ordenan de mayor a menor peso, mediante el método Quicksort. Después se muestran las imágenes recuperadas, más parecidas a la consulta, el valor mayor en este caso indica la imagen más similar. Para calcular el porcentaje de similitud, primero se determina el valor (*max*) de mayor similitud que se obtuvo en la comparación de todas las imágenes recuperadas. Este valor representa el 100% de similitud con respecto a la consulta que se especificó, después se aplica la Ecuación 5.7, para obtener el porcentaje de similitud.

$$\% \text{ similitud} = d_T * 100 / \max \quad (5.7)$$

Para el caso de una *consulta sea por texto*, primero se obtienen los vectores de características de la imagen que represente el concepto que definió el usuario, dicha imagen ya esta previamente definida en la ontología. La recuperación se realiza de la siguiente manera, primero se obtienen las imágenes que pertenecen a clase en la ontología a la que se refiere el usuario en la consulta, obteniendo así un conjunto de imágenes relacionadas semánticamente. Si se especificó algún color, partiendo de este conjunto de imágenes relacionadas semánticamente, se buscan los vectores de características de color, que están almacenados y tengan definido el mismo color, aquí se obtiene un nuevo conjunto de imágenes relacionadas en semántica y en color. De este nuevo conjunto de imágenes, se obtienen los vectores de características de forma y se comparan con el o los vectores de características de forma de la o las imágenes representativas de cada clase(s) recuperada(s). Las distancias que se obtienen (al aplicar las Ecuaciones 5.4 y 5.5) se ordenan de mayor a menor, se obtienen los porcentajes con las Ecuaciones 5.6 y 5.7. y se muestran las imágenes ya ordenadas.

## 5.7 Diseño del módulo ontológico

La recuperación semántica se basa en el resultado de la descripción de contenido semántico, lo que permite buscar y recuperar imágenes respecto a su contenido. El modelo de indexación de imágenes de esta tesis se basó en la utilización de una ontología, para describir la semántica de la imagen y establecer un entendimiento común y compartido de dominio. Con el uso la ontología obtenemos una recuperación eficiente de imágenes similares, que satisfacer de forma eficiente, los requerimientos del usuario. Así, el repositorio de imágenes, se indexa de tal forma, que al hacer una consulta, sólo las imágenes relevantes serán analizadas, sin necesidad de explorar la colección de datos completa.

El uso de ontologías facilitó la implementación de un vocabulario, cuyos términos están orientados al entendimiento del usuario, con respecto a la relación entre el término semántico y

el significado de los objetos en las imágenes. Su representación se utilizará para decidir si un objeto detectado en otra imagen pertenece o no a la misma clase del modelo base.

La recuperación basada en ontologías, está formada de dos componentes: anotación semántica de la imagen y correspondencia de semántica de la imagen. La anotación semántica se enfoca principalmente a la descripción del contenido de la imagen, intenta describir el contenido de la imagen tanto como sea posible. Para esto se obtuvieron dos vectores de características para cada imagen, los cuales describen propiedades de estas imágenes y representan la descripción semántica de las mismas. Estos vectores de características se refieren al color y a la forma del objeto en la imagen.

En la correspondencia para la búsqueda por imagen visual, se comparan los vectores de características de la imagen consulta y los vectores de las imágenes representativas de cada clase de la ontología. La integración de elementos de bajo nivel junto (color y forma) con los elementos de alto nivel (significado de los objetos), permiten recuperar imágenes que tienen contenido similar.

### 5.7.1 Estructura de la ontología.

La ontología esta previamente se definió en el Capítulo 4, se tienen palabras establecidas para cada tipo de descripción (clases).

En la Figura 5.20, se presenta el árbol ontológico, que muestra las relaciones entre sus categorías.  $\mathbf{T}$  es el símbolo universal que identifica a toda la ontología y está representado por la categoría en la raíz del árbol, la cual contiene todas las características de la información de todos los nodos hojas.

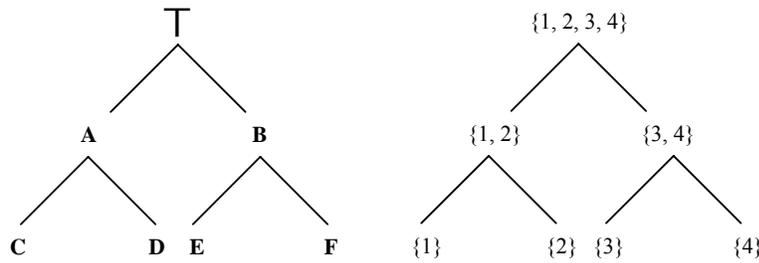


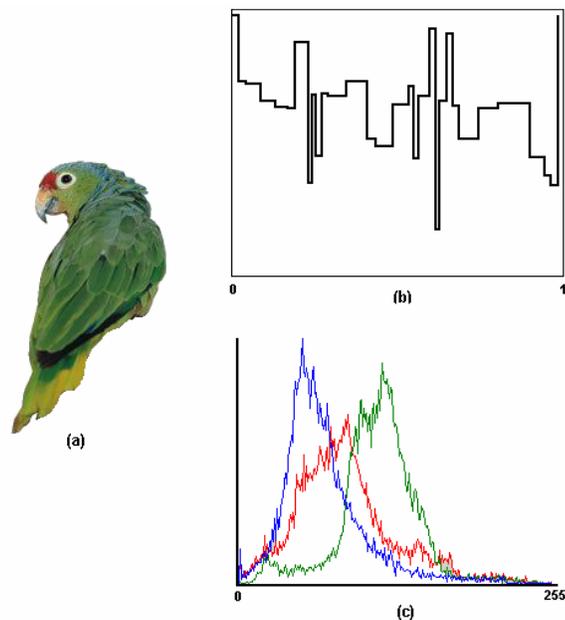
Figura 5.20: Categorías en un árbol ontológico.

Cuando categoría  $X$  tiene dos o más subcategorías, la unión de éstas genera un subconjunto de  $X$ . Por ejemplo  $A$  y  $B$  son subconjuntos de  $\mathbf{T}$ :  $A \subset \mathbf{T}$ ,  $B \subset \mathbf{T}$  y debido a que  $A$  y  $B$  son las únicas dos clases (subconjuntos) derivados de  $\mathbf{T}$ , podemos decir que  $A \cup B = \mathbf{T}$ ,  $C \cup D = A$ , debido a que  $C$  y  $D$  son los dos únicos subconjuntos derivados de  $A$ , que  $E \cup F = B$ , ya que  $E$  y  $F$  son los dos únicos subconjuntos derivados de  $B$  y por lo tanto tenemos que  $C \cup D \cup E \cup F = \mathbf{T}$ , en donde ningún subconjunto  $C, D, E$  y  $F$ , se intersecta.

La ontología se clasifica en relación a otras categorías, cada categoría describe diferentes tipos de propiedades. Una clase es identificada por todos los vectores de características que pertenecen a sus respectivas subclases. Así, en la Figura 5.20, A es identificada por los vectores de características de sus correspondientes subclases C y D.

Para la creación y modelado de la representación ontológica se definió una estructura de árbol. Cada nodo, representa una clase de objeto, los nodos hoja definen a las imágenes, las cuales se representan por sus dos vectores de características, forma y color (Figura 5.21). Existen varias subclases dentro de cada clase, cada una abarca un dominio restringido y limitado. La estructura ontológica contiene aproximadamente 250 imágenes clasificadas con términos definidos, referentes a sus respectivas clases.

Para la creación de la ontología, como ya se mencionó, se construyó un árbol ontológico. Se consideró que tan profunda y descriptiva debería de ser esta estructura ya que si existen tantas categorías como imágenes, la estructura será demasiado profunda, sin embargo, si hay sólo una categoría con muchas imágenes, la estructura será demasiado superficial. Lo adecuado es un término medio, considerando el número total de clases que van a existir dentro de la ontología. Además, se debe de considerar que, una sola categoría no debe de describir demasiados conceptos o a muchas imágenes y que si dos o más categorías describen cosas muy similares, éstas se deben combinarse en una sola.



**Figura 5.21: (a) Imagen original. (b) Función de espacio tangencial para el vector de características de forma. (c) Histograma para el vector de características de color.**

Las categorías de la ontología son simples. Las características en las que se basaron para realizar la clasificación, son intuitivas para el humano. La estructura de la ontología tiene 2 componentes:

- **Nombres para los conceptos importantes del dominio. Como:**
  - Insecto
  - Mamífero
  - Pez
  - ...
- **Conocimiento previo sobre el dominio. Se definen algunas relaciones que existen:**
  - Araña *es un* insecto definido por una forma.
  - Mariposa *es un* insecto definido por una forma.
  - Insecto *es un* animal definido por una forma.
  - ...

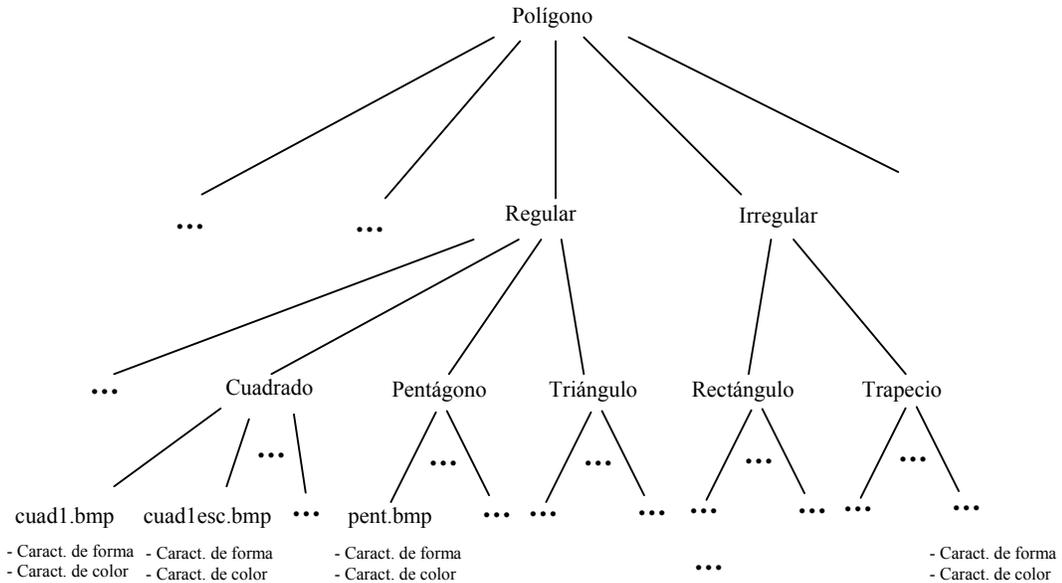
El conocimiento previo, lo definimos como taxonomías de términos, es decir, relaciones que existen entre conceptos de la ontología. Cada concepto es instanciado por un archivo imagen, que pertenece a la clase definida por el concepto. Existen clases generales y otras más específicas, además, todas las clases tienen instancias y no existen sinónimos entre ellas.

En el ejemplo de la estructura ontológica de la Figura 5.22, cada nodo es un concepto definido por su organización. Para cada concepto, existen una serie de atributos que lo definen.

Cada nodo hoja, como ya se mencionó, contiene dos atributos que corresponden a sus respectivos vectores de características de forma y color. Para el término *Cuadrado*, sus atributos serán sus respectivos vectores de características de forma y de color. El concepto de *Polígono Regular*, está formado por los atributos de todas sus subclases, éstas abarcan a los siguientes conceptos: *Circunferencia*, *Cuadrado*, *Pentágono* y *Triángulo*.

Cada nodo hoja, representa una imagen a un cierto nivel y los arcos entre los conceptos representan las relaciones entre ellos, generalmente representan relaciones del tipo “*es un*”.

Las categorías que están en un nivel de jerarquía más alto en la estructura ontológica (*Construcción*, *Naturaleza*, *Polígono*, *Transporte*, *Mueble*), son más generales que las que tienen una menor jerarquía (*Barroca*,... *Insecto*,... *Motocicleta*,... *Mesa*,... ,etc).



**Figura 5.22: Parte de la estructura ontológica, mostrando algunos de atributos (vectores de características).**

## 5.7.2 Relaciones en la ontología

Como se mencionó en el Capítulo 4, existen relaciones en las ontologías, las cuales representan la interacción y el enlace entre los conceptos del dominio. Por ejemplo: subclase-de, parte-de, etc.

Las relaciones que se establecen entre los elementos del dominio, clasifican las entidades del dominio, en base a las siguientes relaciones: *Es-un*, *Instancia-de* y *Parte-de*. La relación *Instancia-de*, es una membresía entre concepto, la relación *Parte-de* muestra relaciones de composición y la relación *Es-un*, muestra la propiedad de inclusión, éste último, especifica que el segundo concepto es más general que el primer concepto.

Los arcos que definidos entre los conceptos especificados en la estructura, determinan relaciones entre ellos. Un arco entre las clases del árbol, se puede ver como un “triple”, el cual consiste de un sujeto, un predicado y un objeto. El triple se puede definir como  $\langle \mathbf{B}, \text{subclase-de}, \mathbf{A} \rangle$ , lo que significa que el concepto  $B$  es una subclase de  $A$ , esto determina que los conceptos  $B$ , como clases inferiores de la clase  $A$ . Un ejemplo de estos triples, basándose en la Figura 5.22, se define como  $\langle \text{Pentágono}, \text{subclase-de}, \text{Polígono} \rangle$ .

Los arcos entre instancias y clases, definen triples  $\langle \mathbf{I}, \text{instancia-de}, \mathbf{C} \rangle$  donde  $I$  es una instancia de la clase  $C$ .

Algunos ejemplos de este tipo de triple, partiendo de la Figura 5.22 son:  $\langle \text{”cuad1.bmp”}, \text{instancia-de}, \text{Cuadrado} \rangle$ ,  $\langle \text{”pent.bmp”}, \text{instancia-de}, \text{Pentágono} \rangle$ ,  $\langle \text{”cuad1.bmp”}, \text{instancia-de}, \text{Polígono Regular} \rangle$ ,  $\langle \text{”pent.bmp”}, \text{instancia-de}, \text{Polígono Regular} \rangle$ , etc.

Se observa que las instancias pueden pertenecer a varias clases de la jerarquía, según el nivel en el que se quiera definir el triple.

En la Figura 5.22, la clase **Cuadrado** tiene una relación *es-un* hacia el concepto **Polígono Regular**, el cual es un concepto más general que **Cuadrado**. Una característica de una relación *es-un*, es que todos los atributos del concepto más general son heredados de sus correspondientes conceptos específicos.

En la recuperación de imágenes basada en ontologías, si queremos encontrar una imagen incluyendo el término general, (por ejemplo Polígono Irregular), las imágenes definidas por conceptos específicos (por ejemplo Trapecio, Rectángulo, etc), se consideran similares y se encuentran en el resultado de recuperación de imágenes. Las imágenes recuperadas al buscar Polígono Irregular, serán las instancias contenidas en las clases de Trapecio y Rectángulo, que son subclases de la clase Polígono Irregular. Las categorías (clases) de la ontología, se pueden considerar como directorios tradicionales en una computadora personal y las imágenes individuales que describen las categorías, son archivos almacenados en dichas carpetas. Una carpeta, a su vez contiene varias carpetas y así sucesivamente, al igual que una clase tiene varias subclases.

El fin de utilizar la ontología, es usar un vocabulario consistente, que facilite y ayude a una recuperación de imágenes más efectiva y eficiente.

Las consultas se hacen correspondiendo conceptos definidos en la ontología y sus combinaciones permitidas de clases y subclases correspondientes.

De cada concepto básico  $ac$  que equivale a un nodo hoja (archivo) en la estructura de la ontología, se obtienen dos atributos  $c_1$  y  $c_2$ .

$ac = (c_1, c_2)$ , donde los atributos  $c_1 = \text{vector de características de forma}$  y  $c_2 = \text{vector de características de color}$ . Se puede dar la combinación de varios conceptos,  $nc, (ac_1, ac_2, \dots, ac_k)$  definiendo un nuevo concepto  $nc$ , donde  $ac_i$ , se refiere a los conceptos que ahora definen atributos del nuevo concepto definido  $nc$ . Estos conceptos se pueden unir formando combinaciones de nuevos conceptos  $cnc(nc_1, nc_2, \dots, nc_m)$ . Cada combinación define un concepto que usa otros conceptos para describir sus atributos.

Ejemplos:

- Definición de nuevos **conceptos  $nc$**  combinando conceptos básicos.

*Regular(Circunferencia(vectorforma,vectorcolor),  
Cuadrado(vectorforma,vectorcolor),  
Triangulo(vectorforma,vectorcolor),  
Pentagono(vectorforma,vectorcolor))*

De la misma forma:

- Definición de combinación de **nuevos conceptos *cnc*** (Construcción(...)), resultado de combinar varios conceptos *nc* como Iglesia(..), Ruina(...).

*Construccion (Iglesia (Barroca (vectorforma,vectorcolor),  
Gotica (vectorforma,vectorcolor),  
Romantica (vectorforma,vectorcolor)),  
Ruina (vectorforma,vectorcolor))*

## 5.8 Conclusiones

En este capítulo, se implementó el modelo de indexación sugerido en esta tesis. Se explica cada proceso, que corresponde a una técnica y/o método de representación de imágenes (color, forma, semántica). La recuperación de imágenes del modelo, se basa en la forma y/o color de las imágenes. Primero se determinan los vectores de características de la imagen a comparar, recuperando un conjunto de imágenes que corresponden al término semántico (clase), que el usuario definió en una consulta por texto o se recuperó automáticamente al comparar la forma y/o color. Estos términos semánticos deberán estar previamente clasificados en la estructura ontológica, la cual se modeló en una estructura de árbol. El árbol ontológico, determina jerarquías necesarias para clasificar las diferentes clases que definen conceptos semánticos. Estas clases son simples y fáciles de entender y relacionar. También se dieron ejemplos de las relaciones que se pueden dar en la estructura. Las proposiciones de la ontología se pueden obtener a partir de los axiomas que se dieron en el Capítulo 4. Al realizar una consulta por color a partir del conjunto relacionado semánticamente, se eligen las imágenes que coincidan con el color o tengan mayor similitud con el color buscado, obteniendo así, un nuevo conjunto de imágenes que están relacionadas en semántica y color. El color nos servirá como filtro para obtener un conjunto menor de imágenes a comparar. La comparación de color también se implementa de una manera fácil aunque se tienen algunas desventajas en el método que se utilizó, debido a que pequeñas variaciones en los tonos pueden afectar la recuperación, éste problema se explica mejor en el Capítulo 7.

Finalmente, a partir de este nuevo conjunto de imágenes, se obtienen los vectores de características de forma para comparar su similitud de forma. Una vez obtenidas las distancias, se ordenan de mayor a menor similitud y se obtiene el porcentaje de similitud. Las técnicas y métodos de representación de forma empleados, fueron fáciles de implementar. Primero se realizaron pruebas con los métodos para determinar el valor correcto del parámetro, con el cual se obtuvieran buenos resultados, sin realizar muchas operaciones de comparación, por ejemplo, para la técnica de evolución de la curva, el valor de 50 segmentos resultó factible para representar y reconocer la forma sin tener demasiada información. También se especificó la secuencia de pasos sugeridos por el modelo, también se define el diagrama de bloques del sistema y la explicación de los casos de uso que presenta.