# Capítulo 2

# 2 Marco Teórico

En el presente capítulo se explicarán ideas generales de teoría, técnicas y aplicaciones de recuperación de información visual. Esto es para entender más acerca del tema y tener una mejor visión de la literatura del área.

## *2.1 Características de alto nivel*

Recuperación de información visual es complejo, ya que diferentes tipos de información son asociados con imágenes o video. Esta información es la siguiente:

- Datos que no están directamente afectados con el contenido de imagen/video, pero de alguna forma relacionados a ello (referido como *meta datos independientes de contenido*). Ejemplos son: el formato, el nombre del autor, fecha, localización, dueño, etc.
- Datos que se refieren al contenido visual de las imágenes. Se pueden mencionar dos niveles:
    - Datos que se refieren a las características de bajo nivel, como color, textura, forma, relaciones espaciales, movimiento, etc. Y a sus combinaciones (referido también como *meta datos dependientes de contenido*). Generalmente estos datos son afectados por hechos preceptúales. Para imágenes sin movimiento las características

inmediatamente percibidas son color y textura mientras que para video son color, textura y movimiento.

- Datos que se refieren a contenido semántico. También llamados como *meta datos descriptivos de contenido*. Son afectadas con las relaciones de las entidades de imágenes con las entidades en el mundo real o eventos temporales, emociones y significado asociado con señales visuales y escenas. Estos datos son los denominados de alto nivel.

La nueva generación de sistemas de recuperación visual soporta recuperación completa por contenido visual. Accesar a información no es únicamente hecha por un nivel conceptual, usando palabras clave como en el dominio textual, si no por un nivel perceptual, usando medidas objetivas del contenido visual y modelos similares apropiados. En esos sistemas, procesamiento de imágenes, reconocimiento de patrones y visión computacional son una parte íntegra de su arquitectura y operación.

Contenido de imágenes sin movimiento incluye:

- Propiedades preceptúales, como color, textura, forma y relaciones espaciales.
- Semántica corresponde a abstracciones, como objetos, roles y escenas.
- Impresiones, emociones y significado asociado con la combinación de características preceptúales.

Para recuperar imágenes de acuerdo a propiedades preceptúales, el paradigma básico requiere que, para cada imagen, un conjunto de características distinguidas (parámetros del modelo) sean pre-informatizado. Para iniciar un query, el usuario selecciona con que

características y rangos de los parámetros del modelo son importantes y escoge una medida de similitud. El sistema checa la similitud de la imagen del query del usuario con la base de datos de imágenes. La mayoría de los sistemas carecen de una flexibilidad necesaria para tratar con conceptos y semántica de las imágenes. Por lo tanto, recuperar imágenes basándose en propiedades de alto nivel, los queries son expresados por texto.

## *2.2 Representación de imágenes*

Esta parte está dedicada a explicar varias técnicas que sirven para obtener información relevante a las imágenes, dichas técnicas son representaciones que se basan sobre alguna sus características principales. Las representaciones que se verán son: basadas en color, basadas en textura, basadas en forma, basadas en ontologías y basadas en agentes de Software.

### 2.2.1 Representación basada en color

Información que contiene color facilita nuestras vidas, por ejemplo en equipos de deportes o en la preferencia de color de nuestros autos. En la actualidad con el crecimiento del Internet, un nuevo campo de necesidades ha nacido, no solo necesitamos recuperar documentos de texto sino multimedia también. En sistemas de recuperación visual la información que nos brinda el color ha sido usada como una herramienta poderosa para dichos sistemas. Un ejemplo de un sistema que utiliza métodos basados en color es PicToSeek [8]. Este sistema fue desarrollado por la universidad de Ámsterdam.

Para el propósito de recuperar información basada en color, se define una taxonomia de acuerdo al criterio siguiente: [9]

- Uniformidad perceptual
- Linealidades
- Intuitividad
- Robustez en contra de las condiciones variantes de las imágenes
- Invarianza al cambio de la vista de dirección
- Invarianza al cambio de la geometría del objeto
- Invarianza al cambio de la dirección de la iluminación
- Invarianza al cambio de la intensidad de la iluminación
- Invarianza al cambio de la distribución del poder espectral de la iluminación.

Esta taxonomia puede ser usada para seleccionar el sistema de color apropiado para una aplicación específica.

### 2.2.2 Representación basada en textura

Junto con el color, la textura es una poderosa característica discriminatoria, presente en casi cualquier parte en la naturaleza. La textura puede ser descrita de acuerdo a su espacio, frecuencia o propiedades preceptúales. Periodicidad, asperosidad, dirección, grado de complejidad son algunos de los más preceptúales atributos de la textura. Características espaciales basadas en estos atributos son particularmente interesantes para recuperar imágenes por similitud de textura. Como los colores, la textura es conectada con efectos psicológicos.

Casi es imposible describir la textura por medio de palabras. Una efectiva descripción de textura puede estar basada en propiedades estadísticas y estructurales de patrones del brillo. Algunos investigadores han usado alguna de estas propiedades para buscar y recuperar. Existen muchas técnicas para extraer descriptores de textura significante de las imágenes, basadas en diferentes modelos y asunciones. Estos métodos se pueden clasificar de la siguiente manera: [10]

- Modelos basados en espacio
    - Función de auto-correlación
    - Matriz de co-ocurrencia
    - Dimensión fractal
    - Modelos auto-regresivos
    - Modelos estocásticos
- Modelos basados en frecuencia
    - Poder espectro
    - Transformación de Wavelets
- Firmas de textura
    - Energía textual
    - Contraste, asperosidad y direccionamiento
    - Repetición, direccionamiento, complejidad

Algunos ejemplos de sistemas que usan textura como principal descriptor para recuperar imágenes son Virage Engine, sistema CANDID y el sistema NeTra.

### 2.2.3 Representación basada en forma

No hay una definición universal de lo que es forma, pero es posible mencionar algunas definiciones bien aceptadas. Forma es la manera en que se ve un objeto definido por sus límites; forma es la apariencia externa de algo [9].

En general estos tipos de representaciones usan lo que se denomina *Shape Matching* donde a continuación se describe.

#### 2.2.3.1 Shape matching

Comparar formas es considerado como uno de los aspectos más difíciles en la recuperación de información visual basada en contenido, ya que la representación de formas frecuentemente es más compleja que el color y la textura, esto se debe porque una forma común necesita muchos parámetros para ser representados explícitamente. Determinar la similitud de dos formas es una tarea muy difícil en sistemas tanto humanos como de máquina. Procesamiento de imágenes y visión computacional han proveído diferentes formas de representar imágenes así como también diferentes medidas para calcular qué tan similar es una imagen de otra.

Es posible distinguir tres categorías diferentes de técnicas para representar la forma: característica de acercamiento de vectores, acercamiento relacional y acercamiento de transformación [10]. El acercamiento de vectores consiste en representar la forma como un vector numérico y la similitud entre dos formas es calculado usando alguna medida de distancia, como por ejemplo la distancia Euclidiana o distancia de Hausdorff.

$$d_E(f_1, f_2) = \sqrt{\Sigma(f_1 - f_2)^2}$$

*Distancia Euclidiana. $f_1$ y $f_2$ denotan dos vectores*

En el acercamiento relacional, una forma es dividida a un conjunto de componentes salientes; el descriptor total es hecho por descriptores de cada componente saliente y la relación entre ellos. Finalmente el acercamiento de transformación es usado para describir una forma basada en el esfuerzo necesitado para transformar la forma a otra. Distancia transformacional es usada para calcular la similitud entre dos formas. Factores como tolerancia a obstrucción y deformación, robustez al ruido, viabilidad para indexar así como también las necesidades de la aplicación deben ser consideradas para escoger el acercamiento adecuado.

### 2.2.4 Representación basada en ontologías

Una forma típica de recuperar una imagen de un repositorio es crear un query basado en una palabra clave. Estas búsquedas tienen muchas desventajas. Una palabra clave en un documento no necesariamente significa que el documento es relevante y documentos relevantes podrían no contener la palabra clave. Sinónimos hacen más difícil la tarea de recuperar documentos irrelevantes, así como los antónimos.

Con el uso de ontologías se eliminan muchos de estos errores. Las ontologías generalmente se usan para:

- Anotar términos

- Búsqueda basada en vista
- Dar semántica a las relaciones

Una ontología no puede ser usada solo para anotaciones y precisar la recuperación de información, sino también para ayudar al usuario a formular la información necesaria y generar el query correspondiente. La mayor dificultad de las ontologías es el trabajo extra para la creación de las mismas y las anotaciones detalladas. Sin embargo, en algunas ocasiones el hacer uso de ontologías vale la pena para una mejor recuperación de información.

### 2.2.5 Representación basada en arquitectura orientada a objetos y agentes de software

Navegar y recuperar imágenes o algún otro objeto multimedia de la Web se ha convertido en uno de los principales esfuerzos en el área de las aplicaciones. En el campo de la biomedicina esto es justificado por un gran número de bio-archivos distribuidos alrededor del mundo, conteniendo un alto rango de información como reportes clínicos, imágenes, etc. Los sistemas que usan este método (VHD-MMS por ejemplo) son construidos en un manejador de base de datos orientada a objetos. Las imágenes son mapeadas dentro de un modelo que contiene atributos y reglas para administrarlas. La interfaz gráfica del usuario le permite preguntar a la base de datos basada en palabras clave pertenecientes a fuentes de conocimiento bien definidas como UMLS y navegar visualmente en el repositorio basado en color, textura y otras propiedades. Con una tecnología como los agentes de software (SA), el funcionamiento del sistema aumenta ya que hace al sistema capaz de

proporcionar un Usuario-Vista del contenido de la base de datos según la especialidad del usuario. Mientras que siendo la estructura de datos sola, la arquitectura de los agentes del software permite presentar la información bajo diversos puntos de la visión. Los modelos virtuales de los datos se generan y se modifican para requisitos particulares a las habilidades del usuario [11].

## *2.3 Métodos existentes*

A continuación se analizarán algunos métodos existentes para recuperar información visual. Algunos de estos métodos son usados en ciertos sistemas existentes para dicho propósito.

### 2.3.1 Wavelets

Wavelets son funciones matemáticas que cortan datos a diferentes componentes de frecuencia, y entonces se estudia cada componente con una solución emparejada con su escala. Los Wavelets tienen ventajas sobre métodos tradicionales de Fourier en analizar situaciones físicas donde la señal contiene discontinuidades y puntos agudos. Los Wavelets fueron desarrollados independientemente en los campos de matemáticas, física quántica, ingeniería electrónica y geología sísmica. El intercambio entre estos campos durante los últimos diez años ha conducido a nuevas aplicaciones usando Wavelets como la compresión de imágenes, turbulencia, visión humana, radares y predicción de terremotos [12].

La idea fundamental detrás de los Wavelets es analizar de acuerdo a la escala. Los Wavelets son funciones que satisfacen ciertos requerimientos matemáticos y son usados para representar datos u otras funciones. Esta idea no es nueva. Usar superposiciones de funciones ha existido desde inicios de 1800, cuando Joseph Fourier descubrió que podía superponer senos y cósenos para representar otras funciones. Sin embargo, en el análisis de Wavelets, la escala que usamos para mirar los datos juega un rol especial. Los algoritmos de los Wavelets procesan datos en diferentes escalas o resoluciones. Por muchas décadas, científicos han querido funciones más apropiadas que la de los senos y cósenos, para aproximar señales. Por su definición, esas funciones no son locales y por ende hacen un trabajo pobre en aproximar formas puntiagudas. Pero usando Wavelets, podemos aproximar funciones que son contenidas en dominios infinitos. Wavelets son ideales para aproximar datos con forma puntiaguda.

El procedimiento de los Wavelets es adoptar una función prototipo, llamada un analizador Wavelet o Wavelet madre. Análisis temporal es ejecutado con una versión contractada de alta frecuencia del prototipo Wavelet, mientras que el análisis de la frecuencia es ejecutada por una versión dilatada de baja frecuencia del mismo Wavelet.

Es fácil explicar las funciones básicas si nos movemos de lo análogo (funciones) hacia lo digital (vectores). Cada vector 2-D (x, y) es una combinación de los vectores (1,0) y (0,1), ya que si multiplicamos x por (1,0) nos da (x, 0) y si multiplicamos a y por (0,1) nos el vector (0, y). Al hacer la suma de estos dos vectores nos resulta (x, y). Ahora regresemos al mundo análogo y vemos como se relacionan estos conceptos. En lugar del vector (x, y), tenemos la función f(x). Imaginemos que f(x) es un tono musical, decimos

la nota A en un particular octavo. Podemos construir A añadiendo senos y cósenos usando combinaciones de amplitudes y frecuencias. Estos senos y cósenos son las funciones básicas de este ejemplo, y los elementos de Fourier. Con los senos y cósenos seleccionados, podemos conjuntar los requerimientos adicionales que sean ortogonales. Esto se hace escogiendo la combinación apropiada de funciones senos y cósenos, donde su producto interno sea arriba de cero. El conjunto particular de funciones que son ortogonales y que construimos f(x) son nuestras funciones básicas ortogonales para este problema.

Los Wavelets son muy parecidos a las transformaciones de Fourier. Sin embargo hay varias ventajas al hacer uso de Wavelets, algunos son:

- Las funciones de Wavelets están localizadas en el espacio, las funciones de seno y coseno en Fourier no.
- Muchas operaciones y funciones utilizando Wavelets son escasas, lo cual resulta que este método se aplique en muchas aplicaciones.
- Wavelets no usa solo funciones de seno y coseno sino un conjunto infinito de posibles funciones básicas

A continuación se verá un ejemplo de una aplicación donde se ha usado Wavelets.

Compresión de huellas digitales del FBI

Desde 1924 a la fecha, la Agencia Federal de Investigación de los Estados Unidos de Norte América ha colectado cerca de 30 millones de conjuntos de huellas digitales. El

principal problema lleva una demanda por parte de la comunidad de justicia a una digitalización y un estándar de compresión. Una imagen de una huella digital tiene aproximadamente una resolución de 500 píxeles por pulgada con una información de 256 niveles de escalas de grises por píxel. Una sola huella digital tiene cerca de 700,000 píxeles y necesita cerca de 0.6 Mbytes de memoria para su almacenamiento. Un par de manos, entonces, requiere cerca de 6 Mbytes. Así que el archivo concurrente del FBI podría ser cerca de 200 terabytes de memoria. Obviamente, la compresión de datos es importante para disminuir esos números. Es aquí donde los Wavelets hacen su aparición.

*Figura 2.1 La imagen de la izquierda es la original, la de la derecha es la reconstruida con una compresión de 26:1.*

Como podemos ver, los Wavelets son muy interesantes, donde podemos implementarlos en una enorme cantidad de aplicaciones como lo puede ser la recuperación de información visual.

### 2.3.2 FFT

Fast Fourier Transform (FFT), es un algoritmo eficiente para calcular la transformación discreta de Fourier (DFT por sus siglas en inglés) y su inversa. Tiene una gran importancia puesto que hay una amplia variedad de aplicaciones, desde procesamiento digital de señales para resolver diferenciales parciales, hasta algoritmos para multiplicar números enteros grandes rápidamente [13].

La transformación rápida de Fourier no es una nueva transformación sino un algoritmo que permite reducir el tiempo de cálculo de la transformación discreta o DFT.

Un examen del algoritmo de la DFT permite comprobar que se precisarán $N$ multiplicaciones complejas y $N$ sumas por cada una de las rayas espectrales que permitan la representación en el dominio de la frecuencia, siendo $N$ el número de muestras. Como hay $N/2$ componentes distintas, contando solo las frecuencias positivas del espectro el número total de multiplicaciones es $N^2$.

La necesidad de precisión y el aumento de la banda de análisis conllevan el aumento de las frecuencias de muestreo y por tanto del número de muestras, multiplicaciones y sumas a realizar, lo cual retrasa la presentación en tiempo real de la señal. Sin embargo muchos de los coeficientes de los complejos son redundantes y el cálculo no puede ser tan exacto.

El método más famoso de FFT es el de Cooley-Turkey. El algoritmo es un divide y vencerás que recursivamente rompe un DFT de cualquier tamaño compuesto de n=n1n2 en más pequeños DFT´s de tamaños n1 y n2, junto con O(n) multiplicaciones por las

raíces complejas de la unidad tradicionalmente se denominan “twiddle factors”. Este método fue popularizado por J. W. Cooley y J. W. Turkey en 1965.

### 2.3.3 CSS

Representación de formas es uno de los aspectos más difíciles en la visión computacional. El problema ha sido difícil puesto que la forma es más compleja que el color y la textura. El color y la textura pueden ser representados por unos pocos parámetros pero en el caso de la forma se necesitan cientos de parámetros para que sea representada explícitamente. Sin embargo, los resultados obtenidos al hacer uso de técnicas de representación de formas son satisfactorios.

CSS por sus siglas en inglés (Curvature Scale Space) es un método de comparación de imágenes que utiliza la forma como principal descriptor. Básicamente el procedimiento de esté método es procesar cada límite de una imagen CSS y entonces encontrar el punto máximo de los contornos CSS que son usados como descriptores de formas para comparar objetos. Una imagen CSS es una organización multi-escala de los puntos de inflexión del contorno donde se desarrolla. La curvatura es una medida local de cuan rápidamente un contorno plano está dando vuelta. Las coordenadas de esos puntos junto con el *aspect radio* de la imagen CSS (número de columnas y número de renglones), la excentricidad, la circularidad y el nombre de la imagen original constituyen características que representan al objeto. Para recuperar imágenes similares de una base de datos el usuario puede poner una imagen y pedirle al sistema que recupere las imágenes similares. El sistema procesa la imagen CSS de entrada y encuentra su máximo,

y después compararla. Las características recuperadas de una imagen CSS por compararla son los puntos máximos de sus contornos de cero. Comparar dos imágenes CSS consiste en encontrar el cambio horizontal óptimo de los máximos en una de las imágenes CSS que rendirían el mejor traslapo posible con los máximos de las otras imágenes CSS. El costo de comparación entonces se define como la suma en parejas de distancias (en CSS) entre los pares correspondientes de máximos.

Una vez comparadas las imágenes, el sistema asigna un valor de comparación a cada imagen candidata en la base de datos que es similar a la de la entrada que dio el usuario. Una imagen candidata es aquella que tanto su *aspect radio*, excentricidad y circularidad caen en el intervalo de la imagen de entrada.

Digamos que $\Gamma$ es una curva plana y $u$ es el *arclenght* normalizado de $\Gamma$. Entonces

$$\Gamma = \{ ( x(u), y(u) ) \mid u \in [0, 1] \}$$

Para cada coordenada de $\Gamma$ esta es envuelta por un *kernel Gaussiano* de medida $\sigma$. La curvatura resultante $\Gamma_\sigma$ va a ser más ligera que $\Gamma$. Conforme $\sigma$ aumenta $\Gamma_\sigma$ se vuelve más ligera y el número de *zero crossings* decrementa. Cuando $\sigma$ se vuelve lo suficientemente grande $\Gamma_\sigma$ será una curva convexa sin ningún *zero crossing* (figura1). El proceso puede terminar aquí y el resultado puede ser mapeado en un plano $(u, \sigma)$. El resultado de este proceso será una imagen binaria llamada imagen CSS de la curva [14].

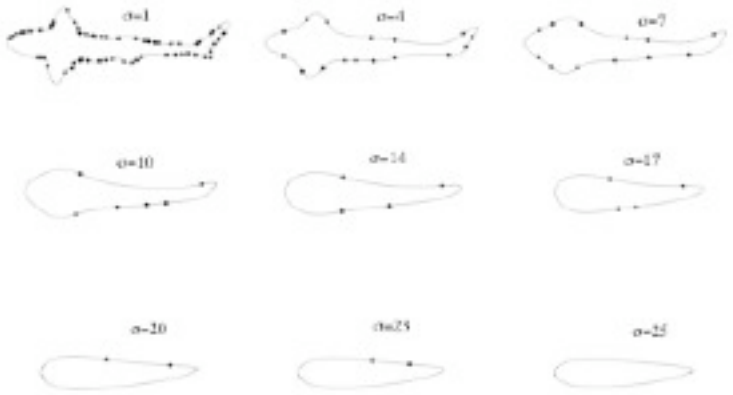

*Figura 2.2. Evolución de curvas con $\sigma$ =25*

Las ventajas que ofrece este método son las siguientes:

- Es robusto con respecto al ruido, escala y cambios de orientación.
- Retiene la información local de la forma de la imagen de entrada. Cada convexidad de la forma de la imagen tiene su propio contorno correspondiente en la imagen CSS.
- Es rápido. En respuesta a un query, el sistema procesa la imagen CSS de la entrada y extrae sus máximos. Entonces el sistema compra el vector característico extraído con todas aquellas imágenes candidatas en la base de datos.
- Es confiable. Varios sistemas como SQUID utilizan este método y los resultados obtenidos son satisfactorios.

### 2.3.4 TSR

Otro método muy usado en el área de recuperación de información visual es TSR. TSR es por sus siglas en inglés, representación de espacio tangente. Usando TSR, una curva *C* es

representada por la longitud del grafo como una función, los pasos siguientes en el eje *X* representa la longitud del arco de cada segmento *C*, y el eje de las *Y´s* representa el ángulo de giro entre dos segmentos consecutivos de *C*. La representación de la tangente es invariante a la rotación, escalamiento y a la traslación [15]. La siguiente figura muestra un ejemplo de cómo se usa este método.

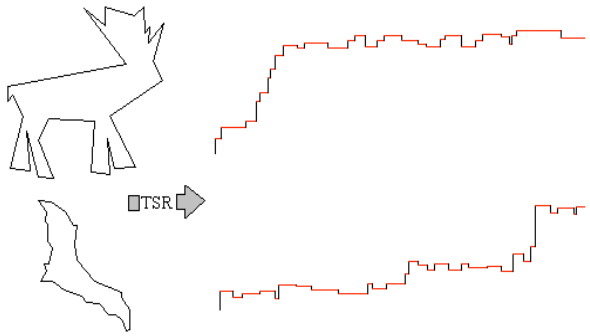


*Figura 2.3 Cada segmento de la figura es llamada paso, la dirección del ángulo de la línea de los segmentos es representada por el eje de las Y´s, y la longitud de cada línea normalizada del segmento con respecto a la longitud de la curva es representada por el eje de las X´s.*

### 2.3.5 Evolución discreta de curvas

Esta técnica reduce el conjunto de vértices de un polígono a un subconjunto de vértices [16], el cual contiene información relevante del polígono original. Para reducir el número de vértices es necesario asignar una medida de relevancia a cada vértice, para que el vértice menos importante sea removido. Una vez que el vértice se remueve sus vértices vecinos deben ser conectados. Este proceso se repite hasta que obtengamos una forma simplificada deseada. La siguiente figura nos muestra esta técnica.

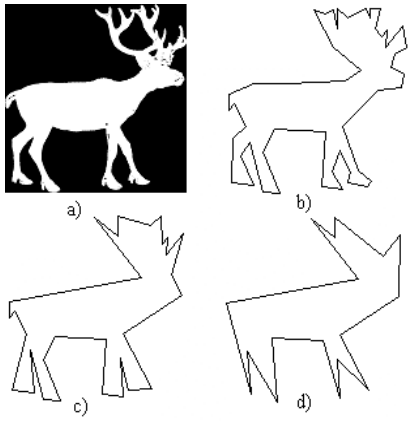


*Figura 2.4 La imagen original es etiquetada como a. Las letras b, c y d nos muestran el proceso de evolución con 50, 30 y 7 vértices respectivamente.*

El orden de las sustituciones de los vértices es dado por la medida de relevancia *K* donde $\beta(s_1, s_2)$ es el ángulo del vértice común de los segmentos $s_1$ $s_2$, y *l* es la longitud de la función normalizada con respecto a la longitud total de la curva poligonal *C*. Entre más bajo es el valor de $K(s_1, s_2)$ menos es la contribución de la forma de la curva del arco $s_1$ $U$ $s_2$. Para detener la evolución es necesario un parámetro que define el número de iteraciones.

$$\boldsymbol{K}(s_1, s_2) = \frac{\beta(s_1, s_2) l(s_1) l(s_2)}{l(s_1) + l(s_2)}$$

Ya que una representación poligonal no es una forma conveniente para calcular la similitud entre dos figuras, es necesario hacer uso de alguna representación alterna como TSR vista en el punto anterior.

## *2.4 Conclusiones.*

En este capítulo se estudiaron características de alto nivel de las imágenes, tipos de representación más comunes sobre su contenido, las cuales fueron: representación basada en color, representación basada en textura, representación basada en forma, representación basada en ontologías y finalmente representación basada en agentes de software. También se analizaron algunas técnicas que pueden ser implementadas en estos sistemas de recuperación de información visual, en particular se vieron: Wavelets, Fast Fourier Transform, Curvature Scale Space y Tangent Space Representation.