Universidad de las Ciencias Informáticas

Facultad 6

Título: Componente de software para la búsqueda y recuperación de imágenes por contenido.

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autores:

Mónica Vigil Martínez

Yoandri Medina Leyva

Tutor: Héctor Raúl González Diez

Junio 2012

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos al Centro GEYSED de la Universidad de las Ciencias Informáticas hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Mónica Vigil Martínez y Yoandri Medina Leyva

_____

Héctor Raúl González Diez

_____

DATOS DE CONTACTO

Tutor: Héctor Raúl González Diez.

Cargos: Dir. Centro de desarrollo, para la informatización de la seguridad ciudadana. Actual miembro de la sociedad cubana de reconocimiento de patrones.

Categoría docente: Profesor asistente.

Categoría Científica: Ms Informática aplicada.

Tema de investigación: Construcción de diccionarios visuales para la recuperación semántica de imágenes.

AGRADECIMIENTOS

Tras estos cinco años de trabajo intenso y momentos imborrables, quisiera dejar por escrito aquellas personas las cuales han hecho de mi vida en la Universidad una estancia placentera y fructífera, y a las que de una manera u otra han contribuido a la realización de este trabajo, que da fin a una etapa crucial y la más hermosa de mi vida.

Quisiera agradecer en primer lugar a mi familia, que ha estado a mi lado en cada batalla ganada o perdida, sobre todo a mi mamá, por ser mi fuente de inspiración, por saber conducirme por el camino en que hoy transito y por brindarme su apoyo incondicional que me ha levantado cuando he caído. Quisiera agradecer a Humbe, por quererme como una hija, por educarme y ayudarme a ser la persona que soy y a mi hermana Jessica y mi prima Liane, por su inmenso optimismo que me ha dado fuerzas en tiempos difíciles.

En segundo lugar quiero agradecer a mis amigos, los que estuvieron conmigo antes, en la Vocacional, que han dotado a mi vida de recuerdos y alegrías indestructibles y que sin querer me han hecho una mejor persona, en especial a Yudelkis por ser mi amiga desde entonces, y a los de ahora, que me han dado impulso y sostén, en los cuales me he refugiado para seguir adelante. A Mailyn y Yolayne por su incondicional apoyo, por los momentos compartidos y por su compañía que me han hecho muy feliz pese a algunas diferencias, que solo la amistad ha sabido ignorar. A Brown y Yoandy, por ser mis mejores compañeros de aula y de estudio, y a todos aquellos que desde sus inicios me han alentado y ayudado, los que vienen conmigo desde 1er año y los amigos que he ganado en el último.

Quiero darle gracias a mi tutor Héctor, por incentivarme en el estudio de esta temática de procesamiento de imágenes, por el tiempo dedicado en compartir las dudas surgidas y por la dedicación inigualable para lograr un trabajo de calidad. Quisiera agradecer por último a mi compañero de tesis, por soportar mi persistencia todo este curso, por elaborar conmigo esta investigación, por el esmero entregado y por su ayuda y compañía que han aplacado las tensiones impuesta entorno al trabajo de diploma.

Mónica

AGRADECIMIENTOS

A la hora de agradecer muchas personas vienen a mi pensamiento, intentaré no pecar olvidando alguna. Ante todo a mi familia, la que siempre mantuvo su fe en mí y me apoyó en cualquier circunstancia, en especial a mi madre, mi hermano y a mi prima Tania, que son los que en gran medida han hecho posible el que yo esté hoy aquí. Un agradecimiento especial quiero dar a mi esposa y a la vida, por inscribirme en el curso de padre, que es la próxima carrera que llena mis expectativas, al darme a ese pedacito sano y salvo que lleva mi sangre y hemos llamado Alex David, el motivo por el que siempre daré hasta mi último aliento.

En segundo lugar quiero agradecer a quienes también he considerado como mi familia, y de los cuales casi ya me tengo que despedir, pero esperemos que no sea por siempre. Ellos son mis mejores amigos, mis hermanos diría yo. Hiram, a quien desde los viejos tiempos del IPVCE he considerado un hermano que siempre me puede aconsejar. A quienes hace 4 años estaban y hoy aun están a mi lado, a Mauricio "el negro", a Daynovi "el flaco", a Dorgis "el yoyi" y a Yanary "nary". Al tocayo de mi primer heredero, Alex, quien espero haya aprendido que "los hombres no lloran". A "el rodo", "al reynier" y a Elisandra que han sabido brindarme lo mejor de sí aun cuando la distancia ya nos separa, demostrando los verdaderos lazos de amistad, de los que habla Naruto. Al "mostro de la 2", Yoanni, de quien me he proclamado discípulo. Especialmente quiero mencionar a quien superó su rol de tutor y se convirtió para mí en un gran personaje, Héctor y su familia, quienes me ayudaron y brindaron apoyo en todo y en mucho más y a quién le estaré eternamente agradecido.

Por último y sin poner nada de desprecio por ello, a quienes también han vivido mis preocupaciones y con quienes he compartido gratos momentos. A "el wilfre", de quien tomé la idea de ser papá. A mí querida compañera de tesis, Mónica, la que además de volverme "loco", supo poner todo su empeño para que lográramos esto, que es el fruto de todo un sacrificio. Al piquete del primer año, a los que hemos llegado hasta aquí y a los que no pudieron llegar pero que están orgullosos de nosotros. A la gente del viejo proyecto, en especial a Zori, que siempre estuvo dispuesta a ayudarme. A todos los que de una forma positiva o negativa me sirvieron para estar hoy listo para ejercer como Ingeniero en Ciencias Informáticas y como padre de familia.

Yoandri

DEDICATORIA

Le dedico este trabajo a mi mamá, por ofrecerme este presente, por ser mi mejor oponente y guiarme hacia el camino idóneo.

A Humbe por haberse comportado como un padre, por brindarme apoyo y su cariño incondicional.

A mi hermanita por el amor que me ha dado siempre y por querer ser para ella un ejemplo, con el anhelo de que siga un día mis pasos.

Mónica

Le dedico este trabajo a mi familia, en especial a mi madre y a ese tesoro que es mi hijo.

Yoandri

RESUMEN

El objetivo de este trabajo es desarrollar un componente de software para la búsqueda y recuperación de imágenes basado en contenido, bajo el paradigma de saco de características. Este componente permite dada una imagen de consulta, poder identificar las imágenes similares de un conjunto de imágenes de entrenamiento. Para el desarrollo del mismo se utilizó la librería OpenCV 2.2, como lenguaje base: C++, como ID de desarrollo: QT Creator y como herramienta case para el modelado de las clases: Visual Paradigm, lo que asegura un desarrollo eficaz de la aplicación.

En el desarrollo de la solución, la extracción de las características de las imágenes se realiza a partir del descriptor local SURF, en el agrupamiento de las misma interviene el algoritmo jerárquico ascendente, con el COP como función heurística para dar puntaje a la mejor partición en el árbol jerárquico, y como algoritmo de búsqueda se emplea el SEP-COP, que identifica los mejores grupos conformados, que luego son incluido en una estructura llamada vocabulario visual, elemento que materializa la conceptualización del saco de características. Permitiendo reducir los datos de entrada, llegándose a comparar a nivel de datos numéricos, debido a que tanto las imágenes de entrenamiento, como la imagen de consulta estarán representadas por un vector de términos. Las pruebas realizadas, demuestran que el algoritmo retorna respuestas favorables para la detección de imágenes similares, con un porciento elevado de aciertos.

PALABRAS CLAVE

Clúster, Descriptor, Saco de Características, Vocabulario Visual.

TABLA DE CONTENIDOS

TABLAS, IMÁGENES Y GRÁFICAS

INTRODUCCIÓN

El desarrollo de Internet aparejado con las mejoras en las conexiones, el aumento de ancho de banda y las nuevas formas de compresión de datos digitales han hecho posible que sea el medio de interacción más utilizado por la sociedad. Por ello es la plataforma común donde todos los usuarios se relacionan para transmitir información, siendo hoy el repositorio mundial de contenido, donde más allá de información textual, se maneja cualquier tipo de archivo incluyendo las imágenes y el video.

La utilización de las imágenes digitales, ha devenido por el aumento de la capacidad del hardware y la creación de interfaces gráficas, dejando atrás las computadoras que necesitaban de solo textos para ser utilizadas. La introducción de la interfaz gráfica significó el nacimiento de máquinas capaces de procesar y presentar información más compleja como la imagen fotográfica y el sonido(Ordoñez Santiago, 2005). Por ende la diversificación del uso de las computadoras, junto a Internet como medio gráfico y dinámico en propuestas de soluciones interactivas ha posibilitado la intensificación del uso de las imágenes, con propósitos de cautivar a los internautas.

La imagen se ha convertido en el elemento fundamental de los medios de comunicación. El respaldo en ellas para trasmitir ideas y enriquecer los textos planos han posibilitado la mejor percepción de la realidad, y el nacimiento de una cultura visual como una forma nueva de comunicación. Por ello se ha utilizado para manipular información, fabricar justificaciones históricas, elaborar propagandas publicitarias, política y en creaciones artísticas. Su uso masivo en Internet ha dotado este medio de comunidades para compartirlas y lugares profesionales para la comercialización de las mismas, sumándole el uso de las cámaras digitales, que ha impulsado la producción y venta de fotografías y la creación de archivos, centros de documentación y bancos de imágenes que gestionan gran cantidad de imágenes fotográficas y tienen como misión difundirlas, bien de forma gratuita o bien mediante el pago de algunos derechos, de las cuales se retroalimentan agencias de publicidad y editoriales (Marcos Recio, 2007).

El uso e importancia de las imágenes tiene su máxima expresión con la aparición de los primeros trabajos cinematográficos, que intentaba mostrar la realidad de una época y ponían al descubierto un conjunto de imágenes secuenciales en movimiento. Tanto fue la aceptación de este nuevo medio de comunicación, que empezaron a intensificarse los programas de televisión y la realización de videos más cortos, que intentaban transmitir ideas referentes a un contexto en específico. Hoy en día con

Internet no sólo se utilizan imágenes y audio, sino que cada vez el uso del video es más común en los sitios electrónicos.

¨El video es uno de los vehículos expresivos que mayor desarrollo ha adquirido en Internet en los últimos años¨ (García, 2010). Anteriormente pocos sitios ofrecían la posibilidad de descargar o visionar videos mediante el entorno virtual, no es hasta el 2008, que la cantidad de internautas que habían accedido a contenidos audiovisuales reportaron un aumento considerable. El 77% de los usuarios norteamericanos habían visionado videos en líneas, de ahí a que visualizar estos archivos, era la actividad más frecuente (82.9%). Debido a este auge, el contenido audiovisual comienza a expresarse mediante numerosas facetas en Internet, el llamado cibercine, que se revela en la descarga (legal o ilegal, gratuita o de pago) de películas completas, la denominada cibertelevisión, que supera la limitación espacial y extendida, además de poder fragmentarse temporalmente, y por último el cibervideo, formado por fragmentos de películas, programas, clips musicales y noticias, las cuales han permitido el crecimiento continuo de estos archivos en Internet (Arias, 2009). Por ello su uso se ha expandido en diferentes planos: en el entretenimiento, como propulsor de noticias en el campo informativo, como sistema de difusión de muchas empresas donde se enmarca el marketing y la promoción, en la educación, entre muchos otros y disímiles espacios.

De esta manera el video y las imágenes son los archivos que más predominan en Internet. Los mismos se manejan en las bibliotecas audiovisuales que almacenan cualquier tipo de información en formato digital, en las agencias de noticias que recopilan datos relevantes contenidos en fotografías, textos y videos para enviarlas posteriormente a portales, diarios, y otros clientes. También se manejan en las grandes base de datos como Youtube, Google Video y en las televisoras viéndose obligadas a través de los años a archivar grandes volúmenes de contenido audiovisual, que a veces ocupan más de una sala de almacenamiento, lo que dificulta su procesamiento y acceso. Es por estos factores que hoy abogan por la digitalización y las soluciones automáticas.

Debido a lo antes expuesto se ha hecho necesario automatizar la búsqueda y recuperación de información que permita dar facilidad a los usuarios en procesos de gran complejidad como la indexación, selección, búsqueda y recuperación de imágenes y videos a través de repositorios de datos distribuidos de gran tamaño. Para ello es necesario remitirse a las imágenes, que también son el elemento clave que conforma el video, que al reproducirse en un espacio de tiempo ínfimo causan sensación de movimiento percibida por el ojo humano. La realización de esta tarea permite trabajar

mejor en el sector multimedia para la gestión de contenido audiovisual y en las tecnologías de la información (TIC) permitiendo brindar un mejor servicio en la comunicación, y distribución de información a través de las redes y medios informáticos.

Anteriormente la recuperación de imágenes se hacía a base de textos mediantes palabras claves predeterminadas que eran subjetivas e imprecisas, producidas por expertos después de un análisis exhaustivo de la imagen. Este proceso traía consigo demora en las respuestas y muchas veces contradicción entre los lenguajes que utilizaba el experto y el que era nativo del usuario, además no se podían representar muchas características propias de la imagen como textura, color, luminosidad y forma (Robles Sánchez, 2004). Muchas investigaciones se llevaron a cabo a partir de la necesidad de encontrar una manera eficiente de recuperación, arrojando resultados con un porciento de certeza y seguridad elevada, surgiendo de esta manera un nuevo concepto: “la recuperación por contenido”.

Con la recuperación por contenido el acceso a la información de las imágenes no se hace a nivel conceptual, sino más bien perceptual, teniendo en cuenta la información visual que contiene la imagen, permitiendo de alguna manera descripciones más concretas.

El proyecto SCCM (Sistema de Captura y Catalogación de Media) de la Universidad de las Ciencias Informáticas tiene como objetivo además de la digitalización de materiales audiovisuales, el de encontrar una solución automática que permita el acceso, descripción, catalogación y recuperación de los mismos. Los especialistas de las entidades que utilizan soluciones como las que propone el proyecto continuamente se ven en la necesidad de hacer búsquedas de información visual asociadas a temas específicos, con el propósito de elaborar nuevos contenidos audiovisuales o como herramienta de ayuda en la captura de descripciones que sean útiles para la catalogación. Los sistemas de recuperación de imágenes basados en contenido satisfacen esta necesidad. Hoy existen soluciones que contemplan esta funcionalidad, pero que a su vez cumplen con otros requisitos propios de su institución, y por tanto no se podría integrar estos sistemas completos al proyecto para satisfacer un requerimiento puntual, además Cuba está abogando por la independencia informática, y cada vez requiere de sistemas informáticos desarrollado en su propio territorio, que le sean útil en distintas ramas.

Por lo antes expuesto se define como **problema a resolver**: ¿Cómo facilitar la búsqueda y recuperación de imágenes en grandes archivos de imágenes en ayuda a los usuarios de Captura y Catalogación de Media?

EL **objeto de estudio** estará centrado en los métodos de búsqueda y recuperación de imágenes por contenido y el **campo de acción** estará enfocado en los métodos de búsqueda y recuperación de imágenes por contenido que se basan en el vocabulario visual y descriptores locales.

El **objetivo general** del trabajo es desarrollar un componente de software para la búsqueda y recuperación de imágenes basadas en contenido para los usuarios de Captura y Catalogación de Media.

La **idea a defender** que guiará este trabajo es que la implementación del componente para la búsqueda y recuperación de imágenes por contenido a partir del vocabulario visual aumentará las opciones de búsqueda que actualmente se brinda a los usuarios del Sistema de Captura y Catalogación de Medias.

Las **tareas** por la cual se guiará la presente investigación se presentarán a continuación:

1. Describir el funcionamiento general de los sistemas para la búsqueda y recuperación de imágenes por contenido.

2. Caracterizar los descriptores espaciales invariantes a las diferentes características de los contenidos audiovisuales.

3. Describir el funcionamiento del paradigma del vocabulario visual para la búsqueda y recuperación de imágenes por contenido.

4. Describir los métodos de clasificación para la búsqueda y recuperación de imágenes por contenido.

5. Proponer los elementos que componen el algoritmo base para la búsqueda y recuperación de imágenes por contenidos.

6. Definición de la tecnología a usar para el desarrollo del componente propuesto.

7. Implementar el vocabulario visual a partir de una base de datos experimental de imágenes.

8. Implementar mecanismos de búsquedas y recuperación de contenidos en grande bases de datos de imágenes.

9. Validar el funcionamiento del algoritmo propuesto a partir bases de datos internacionales y métodos estadísticos.

10. Implementar el componente para la búsqueda y recuperación de imágenes por contenido.

Métodos Científicos

La búsqueda de la respuesta a la interrogante a resolver y la resolución de las tareas plantadas el proceso investigativo estará dirigida por varios métodos científicos de investigación:

Métodos Teóricos

Método Analítico - Sintético: Permite llegar a conclusiones a partir del análisis por separado de los componentes que integran el problema, como los descriptores locales y la representación del vocabulario visual, sobre la base de las proposiciones expuestas en la literatura, para luego sintetizar las soluciones en la confección de un algoritmo único que permita resolver la interrogante propuesta.

Método Inductivo – Deductivo: Permite inducir a partir de situaciones particulares, comportamiento a un nivel general, y deducir a partir de datos habituales, una conducta específica, lo que puede ser útil en la prueba de los algoritmos encontrados, para definir patrones comunes de comportamiento, que pueden ser ajustados en la implementación de nuevos algoritmos. Además se podrá generalizar los resultados de la investigación pudiéndose defender la idea propuesta.

Método Análisis-histórico-lógico: Permite evaluar los resultados obtenidos como parte de la investigación teniendo en cuenta el comportamiento y evolución de las soluciones dadas para facilitar la búsqueda y recuperación de imágenes por contenido, en la búsqueda de eficiencia y calidad.

Métodos empíricos

Método experimental: Permitirá comprobar la idea a defender y validar los resultados de los algoritmos implementados a partir de las métricas propuestas en la evaluación de la eficiencia de los mismos.

Principales Aportes

Se desarrollará un componente de software para la búsqueda y recuperación de imágenes por contenido que será integrado a la solución propuesta por el proyecto de Sistema de Captura y Catalogación de Media y que mejorará las opciones de búsqueda que ya se han implementado (simple y avanzada).

Además se propondrán algoritmos que satisfagan las necesidades antes expuestas y se obtendrá un vocabulario visual a partir de la información contenida en bases de datos experimentales de imágenes.

El documento ha sido estructurado de la siguiente manera:

Capítulo 1: Fundamentación teórica, constituye la base teórica de la investigación realizada. Se describen los principales conceptos relacionados que permitan entender cómo trabajan los sistemas de búsqueda y recuperación de imágenes por contenido.

Capítulo 2: Se describen las herramientas y lenguajes a utilizar para el desarrollo de la solución propuesta, justificando además la selección y uso de estas.

Capítulo 3: Construcción de la solución propuesta, selección de los algoritmos a utilizar**,** unido el diseño de clase.

Capítulo 4: Se hace un análisis de los resultados obtenidos a partir de la base de datos de pruebas definida y los criterios de medida que permiten evaluarlos.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En este capítulo se abordarán los principales conceptos relacionados con el dominio del problema como los son los conceptos de recuperación por contenido, los CBIR, información visual, descriptores e histogramas. Además se hará una descripción de los principales algoritmos de extracción de características, de agrupamiento y clasificación, se fundamentará la situación problemática y se valorará las soluciones existentes.

1.2 Conceptos asociados al dominio del problema

Para adentrarse al tema del procesamiento de imágenes es necesario tener un previo conocimiento de un conjunto de conceptos que facilitarán un mejor entendimiento de la investigación. En esta sección se abordarán algunas de estas definiciones.

1.2.1 Sistema de búsqueda y recuperación de imágenes por contenido (CBIR)

De manera informal un sistema de recuperación de imágenes por contenido es aquel que utiliza la información visual contenida en las imágenes para decidir los resultados de una búsqueda(Cruz, et al.). Una definición más formal es la que se plantea en (Yoo, et al., 2002) donde un CBIR se define como un sistema que a partir de una imagen de consulta obtiene de un repositorio un subconjunto de imágenes semejantes a la proporcionada.

1.2.2 La información visual

La información visual de una imagen comprende color, textura, forma, localización espacial y regiones de interés(Robles Sánchez, 2004)(La Serna, et al., 2010)(Boullosa, 2011). Las cuales constituyen características primitivas o de bajo nivel en contraste con las de alto nivel que describen objetos como montaña, personas, etc. La información visual puede ser tratada desde dos enfoques: global, si se analiza la imagen completa y local si se aplica a una región de la imagen. Las mimas pueden ser extraídas por descriptores.

1.2.3 Descriptores

Los descriptores de imágenes son un mecanismo de extracción de características o información visual, existen dos grupos de ellos: globales y locales. Los globales son los que resumen el contenido de la imagen en un único vector o matriz y los descriptores locales son aquellos que representan la información de la imagen por regiones de interés, por lo que están constituido por todos los vectores de características de las regiones identificadas en la imagen(Boullosa, 2011). Estas regiones poseen información distintiva, que suele ser importante representarlas en imágenes similares, tomadas bajo condiciones distintas.

Los descriptores tienen dos formas de representación: mediante vectores y distribuciones. Los vectores que son los más utilizados se pueden clasificar en Histogramas, descriptores basados en particiones y basados en regiones. Los histogramas son los más ampliamente usados, aunque se crean nuevos enfoques que superan la falta de información espacial de los mismos, que son los basados en particiones, donde la imagen se divide en fragmentos fijos, y por cada una se extrae las propiedades de bajo nivel. El enfoque basado en regiones es mucho más avanzado, se dirige a la detección de objetos, asemejándose a la forma en que los seres humanos interpretan las imágenes (La Serna, et al., 2010).

1.2.4 Histogramas

Un histograma según (Rodriguez, 2007) es una representación gráfica de una variable en forma de barras. La superficie de cada una de las barras mostradas es proporcional a la frecuencia de los valores representados. En el eje vertical se representan las frecuencias, y en el eje horizontal los valores de las variables, de modo que será más alta, o tendrá más superficie aquel valor que más se repite.

Según (Boullosa, 2011) un histograma representa la frecuencia de aparición de cada una de las intensidades de color presentes en la imagen, mediante la contabilidad de los pixeles que comparten dichos valores de intensidad de color. El histograma está compuesto por diferentes rangos o contenedores que representan un valor o conjuntos de valores de intensidad de color.

Se puede definir entonces que un histograma de una imagen M, es un vector v de dimensión N, tal que, v(n), con n = {0,1,…., N-1} representa la frecuencia con que un valor de intensidad n_i con i =

{0,1,…., N-1} de un pixel se manifiesta en una imagen. Pudiéndose concluir que el “histograma de color describe la distribución de color global en una imagen” (Huang, et al.).Ver Anexo 1

1.3 Objeto de Estudio

Descripción General

La búsqueda y recuperación de imágenes por contenido es una tendencia que data desde 1992 y supone que la información visual de una imagen debiera estar dada por su propio contenido y propiedades. A partir de entonces el desarrollo de sistemas basados en este principio (CBIR) con fines tanto comerciales como académicos ha tenido un gran auge. De ahí que los CBIR jueguen un papel significativo en instituciones como hospitales, televisoras, universidades, etc.(Robles Sánchez, 2004).

1.3.1 Enfoques de los Sistemas de Recuperación de Imagen

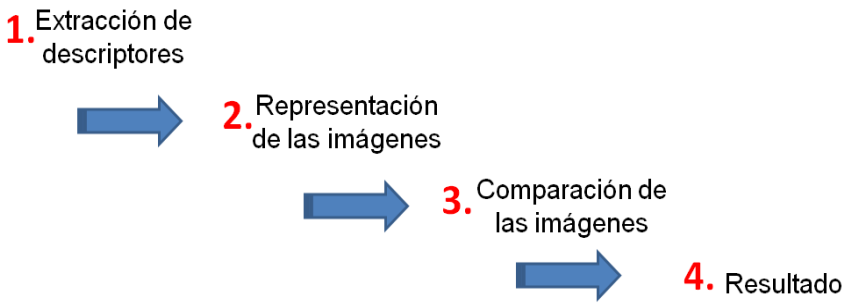
Los sistemas de búsqueda y recuperación han evolucionado desde las primeras definiciones por los años 70 (Rui, et al., 1999), referidas al campo de la recuperación de imágenes basado en texto, ya sea por una palabra clave, un conjunto de estas o una descripción textual del contenido de la imagen. Ver anexo 2

Las dificultades que para este tipo de sistemas trajo consigo el aumento considerable del volumen de información de los archivos, en los que se requerían las búsquedas, propició que para los 90s se introdujera con mayor auge la información visual como un elemento a considerar desde el punto de vista propio de una imagen, ya sea por sus colores, texturas o formas, iniciando así la búsqueda y recuperación de imágenes basado en el contenido. Ver anexo 3

La comunidad científica del campo del procesamiento de imágenes en la actualidad se enmarca en lo que se ha denominado “búsqueda semántica” que prevé una mayor efectividad de los CBIR al realizar búsquedas. El término “búsqueda semántica” trata de interpretar una imagen por su contenido, simulando o imitando la interpretación visual que un humano tendría de una imagen, esto un poco influenciado por los avances en ramas como la detección en imágenes de objetos, personas, texto, etc. Una definición más acertada es que “las características de las imágenes son mapeados a conceptos representativos que describen de una forma muy próxima la información contenida en la

imagen(Moreno, 2008)". Estos sistemas están influenciados por la temática específica del entorno para el que se diseñan. Ver anexo 4

Pasos que implementan los sistemas CBIR:



En los dos primeros pasos la especificidad de cada uno de los distintos sistemas para la búsqueda y recuperación de imágenes por contenido radica en los descriptores que se aplican a las imágenes tanto del repositorio de consulta como a la imagen de muestra. Otro punto distintivo podría ser la forma de representar los vectores de características que identificarán a cada una estas imágenes.

En el tercer punto los distintos algoritmos podrían diferir de la propia función de similitud o de distancia como también se conoce, que se emplee para determinar los candidatos a posibles resultados. Por lo general estos método emplean un mecanismo de umbral para determinar cuáles de los candidatos a resultados realmente lo son, y la definición de este umbral es un tema también distintivo de estos algoritmos.

El último paso puede estar aparejado a algún tipo de prioridad o de ordenamiento a la hora de mostrarle al usuario el resultado.

1.3.2 Representación en el Procesamiento de imágenes

En el procesamiento de imágenes que utilizan los CBIR una imagen es representada por un conjunto de características visuales siendo una etapa del procesamiento que describen el contenido de la misma. Para ello se utiliza dos procesos: la extracción de las características y la construcción de descriptores visuales para el almacenamiento y la recuperación. En el primero de los procesos, el

objetivo es enfocarse a la información visual de la imagen tanto características de bajo nivel como de alto nivel, utilizándose los descriptores, que además permiten en el segundo proceso construir un vector final de características (La Serna Palomino, et al., 2010).El enfoque que actualmente se utiliza para representar tales características está dado por el paradigma de vocabulario visual.

1.3.3 Vocabulario Visual

Un vocabulario visual se construye para representar el diccionario por grupos de características extraídos de un conjunto de imágenes de entrenamiento. Las características de la imagen representan las áreas locales de la misma, así como las palabras son las características locales de un documento. La agrupación es necesaria para que unos vocabularios discretos se puedan generar a partir de millones (o billones) de las características locales en la muestra de los datos de entrenamiento. Cada grupo de característica es una palabra visual. Dada una nueva imagen, son detectadas las características y se asigna a sus más cercanos términos coincidentes (centros de clústeres o centroides) del vocabulario visual. El término vector es simplemente el histograma normalizado de las características cuantificadas detectadas en la imagen.

Existen otras terminologías como codebooks o libro de códigos, que representan de igual forma el saco de características de la imagen. El histograma normalizado de los códigos es exactamente el mismo que el histograma normalizado de palabras visuales, sin embargo, está motivado desde puntos de vista diferentes.

1.3.4 Descriptores de imágenes

Modelos de colores o Correlogramas

Este descriptor surge como una variante más eficiente del histograma de color, que ha sido destinado en un principio a la comparación e indexación[1] de imágenes. Las variantes más generalizadas del método de histograma de color se centran en la partición de las imágenes trabajando el histograma por regiones estáticas o el refinamiento de histogramas a partir de la extracción de propiedades de forma local.

[1]La indexación de imágenes tiene como objetivo adjuntar a una imagen una serie de descriptores de contenido.

Se puede ver a un Correlograma en una definición informal, como una tabla indexada por pares de colores en la que el k-ésimo valor (i, j) especifica la probabilidad de encontrar un color j a la distancia k de un pixel de color i en una imagen (Huang, et al.).

El Correlograma es un método que no está sujeto a las características tanto locales como globales de una imagen sino que tiene en cuenta "tanto la correlación[2] del color espacial de forma local, junto con la distribución global de esta correlación espacial". Este modelo representa por lo tanto, el cambio de la correlación espacial de colores respecto de la distancia entre los pixeles (Boullosa, 2011).

Propiedades de los Modelos de Color o Correlogramas:

- Incluye la correlación espacial de los colores.
- Se utiliza para describir la distribución global de la correlación espacial de los colores locales.
- Es fácil de calcular.
- El tamaño del vector de característica es bastante pequeño.

Histograma de Gradientes Orientados (HOG)

El algoritmo Histograma de Gradientes[3] Orientados se usa con el propósito de encontrar objetos en una imagen. La esencia de dicho algoritmo es describir por medio de la distribución de los gradientes la forma de un objeto en una imagen (Ingelmo, 2009).

Se implementa dividiendo la ventana de la imagen en pequeñas regiones (celdas), por cada una se acumula un histograma de direcciones o de orientaciones de gradiente, teniendo en cuenta los pixeles

[2]La correlación trata de establecer la relación o dependencia que existe entre las dos variables que intervienen en una distribución bidimensional.

[3]El gradiente es una medida de la inclinación de una curva. Se define como la relación del cambio vertical (elevación) con respecto al cambio horizontal (recorrido) para una línea no vertical.

de cada celda. Los histogramas combinados forman la representación final. Para mejor invarianza a la iluminación, sombreado, y otras variaciones a las cuales puede estar sometida la imagen, se normaliza la respuesta antes de usarla, esto puede ser realizado mediante la acumulación de una medida del histograma local sobre algunas regiones más largas y utilizar el resultado para normalizar todas las celdas que se encuentran en estas regiones. Cada bloque normalizado es un histograma de gradiente. El análisis de orientación da robustez a los cambios de iluminación, y es importante destacar que los histogramas de por sí, son fáciles y rápidos de calcular (Triggs, 2005)(T. Freeman, et al., 1994).

Descriptor Scale Invariant Feature Transform (SIFT):

Descriptor originalmente desarrollado por Lowe (G. Lowe, 2004) para el reconocimiento de objetos de manera general, propuesto como un algoritmo capaz de detectar puntos característicos estables y además “invariantes frente a diferentes transformaciones como traslación, escala, rotación, iluminación y transformaciones afines” (Boullosa, 2011) en una imagen. Este descriptor tiene como desventaja más significativa que “en aras de lograr su robustez, es resultado de complejos y tardados procesos iterativos, lo que representa un alto costo computacional” (Chang, y otros).

El proceso de extracción de características de una imagen en este descriptor se define en 4 fases según Lowe(G. Lowe, 2004):

- Detección de Extremos en el Espacio Escala: tiene como objetivo la detección de los posibles puntos de interés, o sea los puntos candidatos.

- Localización de los Puntos de Interés: se discriminan los puntos candidatos cuya estabilidad se vea afectada, según Lowe “los puntos no firmemente situados sobre los bordes o aquellos con bajo contraste son bastante vulnerables al ruido y por lo tanto no podrán ser detectados bajo pequeños cambios de iluminación o variación del punto de vista de la imagen.”

- Asignación de la Orientación: se asigna a cada punto de interés una orientación basada en las propiedades locales de la imagen para garantizar la invarianza respecto a la rotación.

- Descriptor del Punto de Interés: en esta etapa “se crea un vector de 128 características por cada uno de los puntos de interés, que contiene una estadística local de las orientaciones del gradiente de la escala de espacio gaussiano[4] ” (Boullosa, 2011).Ver anexo 5

SURF (Speeded Up Robust Features)

El descriptor SURF es uno de los más utilizados para la extracción de puntos de interés[5]. El mismo utiliza la matriz hessiana[6] y su determinante para la localización de los puntos y la determinación de la escala, que produce la reducción del tiempo de computación, esto hace al algoritmo más preciso y rápido. Una vez obtenida la escala mediante la matriz y su determinante, se calcula la orientación del punto de interés, para posteriormente calcular el descriptor SURF. El cálculo de la orientación le otorga al descriptor invarianza ante la rotación. El vector obtenido por este algoritmo es distintivo y al mismo tiempo robusto al ruido, errores y deformaciones geométricas y fotométricas[7], su longitud tiene una dimensión de 64 valores, lo que supone una reducción de la mitad de la longitud del SIFT, además no permite que haya varios puntos invariantes en una misma posición, con distinta escala y/u orientación y utiliza siempre la misma imagen original. (Romero, y otros, 2009)(Boullosa, 2011). Para saber si hay similitud entre una imagen y otra, los puntos de interés de la imagen 1 en un instante de tiempo t se comparará con los puntos de interés de la imagen 2 en un tiempo mayor, si la distancia relativa entre ambos puntos es menor que el umbral, en un caso hipotético con valor de 0.7, entonces existe una semejanza entre ambos (Dominguez, 2009).

Híbridos

---

[4]El espacio de escala gaussiana es una teoría formal para manejar las estructuras de la imagen en diversas escalas de manera tal que las características puedan ser suprimidas sucesivamente asociadas a un parámetro t en cada nivel de la representación del espacio.

[5]Los puntos de interés son los puntos invariantes a la deformación de la imagen.

[6]La matriz hessiana de una función f de n variables, es la matriz cuadrada de n × n, de las segundas derivadas parciales.

[7]Las deformaciones geométricas y fotométricas están dada por cambios de desplazamiento, rotación, escala e iluminación.

La combinación de muchos de los descriptores caracterizados anteriormente, permite eficiencia y mejores resultados, posibilitando suprimir u opacar las deficiencias que tienen estos individualmente.

El HOG intenta representar de manera robusta las características locales de una imagen, pero suele ser sensible a los cambios de sombra puesto que dependen de la intensidad del gradiente. Para mejorar estas limitaciones, se toman algunos de los elementos del SIFT como la utilización de rejillas de 4 x 4 celdas y 8 orientaciones de gradiente por cada punto clave o interés. El método se basa en un modelo físico del proceso de formación de la imagen y se esfuerza por eliminar los efectos de las sombras, produciendo una imagen de contorno invariantes a estas modificaciones, aunque las características extraídas durante la fase de aprendizaje son importantes para la detección. Se hace posible entrenar el detector usando solo una condición de iluminación, que puede ser eficaz para detectar objetos o escenas con diferentes condiciones de iluminación. (Villamizar, y otros, 2009)

El descriptor HOG captura los bordes o la estructura de los gradientes, los cuales son características del contorno y de la forma local. Sin embargo al igual que otras características de contorno y textura una de las debilidades que presenta es que no puede representar con eficacia objetos o fondos con grandes regiones ya que los contornos de ellos pueden ser indistintos. Otra desventaja es qué la orientación es sensible lo que implica que al rotar un objeto la representación anterior del mismo puede ser invalida.

La combinación de un conjunto de características con la integración de los colores, el gradiente de orientación, los contornos locales y los descriptores SIFT propone un enfoque mejor a la sensibilidad de la orientación de las características de HOG, calculando la orientación dominante, lo que prueba la efectividad de la combinación de características.

En la combinación de las características se usa el HC o histograma de color en el espacio RGB y el HOG que se aplica a imágenes grises, con lo cual se construye el histograma HOGC. Estas características fueron escogidas porque pueden computar eficientemente y porque el cálculo de HC y HOG son simples estadísticas de gradiente de color y simples probabilidades de ocurrencia de orientación.

En el cálculo del histograma de color RGB que son robusto a la rotación y deformación, se obtiene un histograma de 48 dimensiones, por cada 16 niveles que contiene cada componente de color. El HOG

que se calcula posteriormente posee un bloque de detección constituido por 4 celdas de dos pixel, la cantidad de pixel que delimitan el contorno del bloque son 9, y por cada uno se calcula la orientación que suele ser de 8 posiciones. Por lo que el descriptor final de cada bloque posee 72 de dimensión. Para ser frente a la rotación del objeto se utiliza la orientación dominante del método SIFT, lo que puede producir hasta cierto punto una insensibilidad a la rotación(Han, et al., 2011).

Por otra parte el descriptor SIFT no es invariante a los cambios de luz aunque es mucho más insensible que el HOG, debido a que el canal de intensidad es una combinación de los canales R, G, B y una modificación en la apariencia de los colores puede traer consigo resultados erróneos. Para mejorar esta limitación Bostch computa el descriptor SIFT sobre los canales del modelo de color HSV. Esto brinda una dimensión de 3 X 128 por descriptor y 128 por canal. El modelo de color H es invariante a escalas, y movimiento con lo que respecta a la intensidad de la luz (Van de Sande, y otros, 2010).

En otras investigaciones la combinación de los descriptores se realiza mediante la ponderación de los resultados obtenido por cada uno, que se combina con una nueva ecuación de distancia euclidea donde los resultados de distancia por cada descriptor son normalizados. De esta manera se obtiene una nueva distancia euclidea que permite comparar la imagen original con la que imagen nueva a buscar. De los resultados obtenidos en esta investigación resultó que la unión más óptima, fue la del HSV con el SURF, que es, al cambio de luz, movimiento de objetos en la imagen y variación del zoom superior al descriptor HSV de manera individual, provocando que sus resultados mejoren los del SIFT que a su vez representa el mejor de los rendimientos individuales (Boullosa, 2011).

1.3.5 Operadores de puntos de interés

Como se había analizado, los descriptores locales se basan en la detección de zonas de interés. Estas zonas son determinadas a partir de un operador de punto de interés que operan directamente sobre los valores de intensidad de una imagen. La extracción de puntos de interés es la primera etapa en aplicaciones de búsqueda y recuperación, por lo que su importancia para facilitar establecer similitud entre imágenes distintas, es alta. Debido a ello mucha investigación se ha volcado en proponer detectores de puntos de interés que sean invariante a rotación, traslación, escala y a transformaciones afines.

Los detectores Harris-Affine y Hessian-Affine, están diseñados para que detecten puntos que no solo sean invariantes a transformaciones de escala y rotación, sino también a cambios de puntos de vista de la imagen. Para ello detectan los puntos en un espacio de escala y determinan una región elíptica por cada punto, que pueden ser determinados mediante el operador Harris o el Hessian. El primero de ellos procede buscando aquellos puntos donde la matriz de segunda derivada C alrededor del él tenga valores de gran tamaño. La matriz C se calcula a partir una ventana alrededor del punto, ponderada por una gaussiana que suma todos los pixeles en un vecindario circular. El detector Hessian por su parte calcula la segunda derivada por cada punto de la imagen, y luego escoge aquellos donde el determinante de la matriz hessiana obtiene un máximo. La búsqueda de los puntos la realiza a través de una ventana que hace un barrido por toda la imagen, manteniendo solo los pixeles cuyo valor es mayor a los valores de los 8 vecinos inmediatos dentro de la ventana. El detector devuelve la zona cuyo valor está por encima de un umbral predefinido (Grauman, y otros, 2008).

Los puntos de Harris son localizados con mayor precisión pues tiene en cuenta una zona mucho más grande en la imagen. Por lo tanto, los puntos de Harris son preferibles cuando se busca esquinas exactas o cuando se requiere precisión en la localización, mientras que los puntos de Hessian pueden proporcionar lugares de interés adicionales.

En ambos casos la selección de la escala está basada en una función de Laplaciano y la forma de la región elíptica es determinada con la matriz de gradiente de segundo orden, que describe la distribución del gradiente alrededor de cada punto. Dado estos puntos iníciales extraídos bajo condiciones de escalas, se aplica una estimación iterativa sobre la región elíptica, normalizándose a una región circular, y se procede nuevamente a estimar la forma de la región calculando la matriz, repitiéndose los dos pasos anteriores hasta que los valores de la misma para un nuevo punto sean iguales (Mikolajczyk, 2005).

Estos operadores intentan mejorar los resultados de los operadores invariantes a transformaciones de escala Harris-Laplacian y Hessian-Laplace. El primero de ellos fue propuesto por el poder discriminativo en comparación con los operadores Laplacian y Difference of Gaussian (DoG). Combinando la detección de las esquinas de Harris con un mecanismo de selección de escala. El método primeramente construye dos espacios separados a gran escala, con el uso de Harris localiza los puntos candidatos por cada nivel de escala y selecciona aquellos para los cuales el Laplaciano simultáneamente alcanza un valor extremo en las escalas. Los puntos resultantes son robustos a cambios en la escala, rotación de imagen, iluminación, y ruido de la cámara. El mismo retorna menor cantidad de puntos que los operadores Laplacian y DoG. Para muchas aplicaciones prácticas de reconocimiento de objetos, un menor número de regiones de interés puede ser una desventaja ya que reduce la solidez a la oclusión parcial.

Por esta razón se ha creado otra versión de Harris-Laplacian, en lugar de buscar los máximos, se selecciona la escala máxima del Laplaciano en los lugares para los cuales la función de Harris también alcanza un máximo a cualquier escala.

En el Hessian-Laplace la escala es seleccionada de acuerdo al máximo local obtenido por la traza y el determinante de la matriz Hessiana (H) simultáneamente. Específicamente, primero aplica el operador Hessian para localizar los puntos de mayor interés en cada nivel de escala, seleccionándose aquellos puntos que son máximos en el espacio de escalas determinados por la función de Laplaciano. El detector de Hessian-Laplace devuelve más regiones de interés que Harris-Laplace en una repetición ligeramente inferior.

Existen otros operadores que son referenciados como operadores de puntos de interés o regiones de interés, que han sido motivado para proporcionar información complementaria sobre alguna región detectada, que no puede ser obtenida a partir de otros operadores como los de esquina. Dentro de este conjunto se encuentra el Laplacian of Gaussian (LoG), que se basa en la búsqueda de extremos en un espacio de escala. Para ello utiliza primeramente un desenfoque gaussiano que provoca una opacidad en la imagen, intentando reducir el ruido, de esta manera se hace más fácil detectar los bordes a partir del filtrado Laplaciano. El mismo consta de una máscara circular con pesos positivos en el centro de la región y pesos negativos en el borde del anillo. Por lo tanto, se producen respuestas máximas si se aplica a una zona de imagen que contiene una similar aproximación circular en una escala correspondiente, de tal manera que una ubicación de un punto clave repetible también puede definirse como el centro del la región (Grauman, y otros, 2008).

Otro de los operadores de región el DoG que intenta aproximar el espacio de escala Laplaciano, a partir de la diferencia de dos escalas adyacentes. Las regiones de interés son definidas como ubicaciones de extremos simultaneaos en el plano de la imagen y a lo largo de las coordenadas de escala. Estos puntos son encontrados comparando la diferencia gaussiana por cada punto con sus 8 vecinos en la mismo nivel de escala, y los 9 vecinos más próximos en los dos niveles adyacentes. Los puntos finalmente escogidos, son los puntos máximos en un radio determinado, de los puntos detectados.

1.3.6 Clasificación en el Procesamiento de Imágenes.

En el proceso de clasificación los vectores de características de las imágenes de entrenamiento, son almacenados, conformando una matriz donde por cada fila o columna según se defina, se almacenan las características de una clase en específico. Las clases son determinadas a partir del agrupamiento o clustering de características similares, empleando algoritmos jerárquicos y particionales como el jerárquico descendente o k-mean respectivamente. Cuando se necesita comparar un conjunto de imágenes que no están previamente clasificadas, se extrae su vector de características para luego compararlos con los existentes haciendo uso de un algoritmo de clasificación como SVM o K-NN. De esta manera se determina a qué clase pertenece, pasando a formar parte de la base de datos del sistema, integrándose a la matriz y utilizándose en la clasificación de nuevas imágenes(Muños, 2010).

1.3.7 Algoritmos de Agrupamiento

K-mean

El objetivo fundamental de este tipo de análisis es clasificar N objetos en K grupos o clústeres, en los cuales estarán representados los atributos. El K-means es uno de los algoritmos de agrupamiento más populares y ampliamente utilizados, debido a que su implementación es relativamente fácil. El mismo consta de cuatro pasos:

- Inicialización: Se definen un conjunto de objetos a particionar, el número de grupos y un centroide[8] por cada grupo.

[8] Centro de masa de un objeto uniforme. Para un clúster suele ser la media de los puntos que lo conforman.

- Clasificación: Se clasifican los nuevos objetos que se procesen, de acuerdo a su cercanía respecto a los centroides de cada grupo existente, por lo que el mismo pertenecerá al grupo más cercano.

- Cálculo del centroide: Se vuelve a calcular el centroide, esta vez incluyendo a los nuevos objetos incorporados en el paso anterior.

- Condición de convergencia: Se determina cuando se deja de recalcular el centroide, para pasar a dar la respuesta final. La condición más utilizada es converger cuando no existe un intercambio de objetos entre los grupos (Pérez, y otros, 2007).

Algoritmos Jerárquicos

Los métodos jerárquicos se dividen en dos grandes grupos, en aglomerativos y divisivos o particionales, o también conocidos como los tipo bottom-up y top-down respectivamente. El primero de ellos parte de las hojas del árbol, y se van uniendo según las similitudes que existan entre ellas. Los elementos más cercanos se encontrarán en una misma agrupación o raíz de un subárbol hasta ir conformando un solo grupo, o raíz principal. El segundo método realiza el clustering de manera inversa, comienza desde la raíz y va particionando los conjuntos de forma recursiva hasta que alcanza algún criterio de parada, en la mayoría de los casos está dado por el número K de clústeres. Muchos investigadores reconocen que estos algoritmos de clustering son adecuados para el agrupamiento de grandes conjuntos de datos, pero se cuestiona su calidad de agrupación quedando por debajo de su contraparte, los algoritmos de aglomeración (Zhao, y otros, 2002).

1.3.8 Algoritmos de Clasificación

K-NN

El K-NN es un algoritmo de clasificación supervisado debido a que necesita datos de entrenamiento que son introducidos manualmente por algún usuario. Se basa en los vecinos más cercanos para poder identificar la clase a la cual pertenece la nueva instancia que se desee clasificar.

A cada clase estarán asociados muchos vectores de características, que estarán representados por valores numéricos. A la llegada de un nuevo vector se calcula la distancia con los otros ya existentes, y

de acuerdo con los K más próximos, se procede a distinguir cual es la clase que lo representa, en caso de que varios de estos puntos pertenezcan a la misma clase se le asigna al vector la más frecuente.

En este algoritmo el costo de la búsqueda depende de la cantidad de instancia que exista, por ello con una base de entrenamiento demasiado grande, el costo se incrementa elevadamente, sumándole que necesita de una memoria auxiliar que ordene para cada nuevo vector que se desea clasificar, las distancias entre los puntos. Además nunca se sabe con certeza que tan eficaz es la cantidad K, para clasificar, pues no existe ningún método que la determine. Además el rendimiento de este método baja si el número de descriptores crece (García, y otros, 2008).

SVM

La Máquina de Soporte Vectorial o SVM por sus siglas en inglés tiene su definición por Vapnik en(Vapnik, 2000). Es un clasificador lineal puesto que basa la asignación de un elemento a una clase según el valor de una combinación lineal de sus características, es además un clasificador supervisado pues requiere de un entrenamiento previo a su uso.

La SVM que tiene como objetivo determinar la pertenencia de un elemento a una de dos posibles clases, obtiene la solución óptima al problema de encontrar la separación de mayor margen entre dos conjuntos disjuntos de elementos. Ver anexo 5

En la actualidad, SVM es muy utilizado en la detección y el reconocimiento de objetos, recuperación de imágenes basado en el contenido, reconocimiento de texto, la biometría, reconocimiento de voz, etc., dado que ha mostrado resultados mejores que las redes neuronales (Vapnik, 2000).

1.3.9 Algoritmo para determinar número óptimo de clúster

AntClust (Colonia de Hormigas)

Propuesto en 1992 por Marco Dorigo en (Dorigo, 1992) es un algoritmo perteneciente a la clase de meta-heurísticos, que son algoritmos para obtener un número suficiente de soluciones a difíciles problemas de optimización combinatoria en un período razonable de tiempo de cálculo. Basado en la explicación biológica del comportamiento de una colonia de hormigas al obtener alimentos.

Según el comportamiento real de las hormigas, para obtener alimentos salen de la colonia en busca de alimentos, recorriendo un área determinada aleatoriamente. Al encontrar comida las hormigas retornan a la colonia, dejando un rastro desde el lugar donde encontró el alimento hasta el nido. En el rastro las hormigas van dejando una sustancia química llamada “feromona”, con una intensidad según la calidad y la cantidad encontrada del alimento. Las demás hormigas al regresar a la colonia con su alimento encontrado valorarán cual es el rastro de mayor intensidad de feromona y hacia ahí acudirán, estas nuevas hormigas que se incorporen a un rastro aumentaran la intensidad de la feromona del rastro al agregar el suyo, dándole mayor probabilidad al rastro en el que se encuentren la mayor cantidad de hormigas trabajando, trazando así lo que computacionalmente se le denomina un camino mínimo. En la colonia artificial que simula el algoritmo los alimentos no son más que los píxeles, el área de donde obtendrán estos alimentos será la imagen, la colonia será la estructura que guardará los clústeres.

El algoritmo AntClust adopta además una filosofía para acomodar los píxeles que traen las hormigas a la colonia, basada en la teoría de agrupamiento o clustering. El agrupamiento de los píxeles dependerá de un criterio de similitud, que garantizará que la imagen quede dividida en grupos de tal manera que los píxeles dentro de un mismo grupo sean lo más homogéneos posible, mientras que los grupos entre sí sean tan heterogéneos como sea posible con respecto a esta medida de similitud. Este criterio de similitud o grado de pertenencia de un pixel a un clúster estará determinado por la medida en la que su valor se aproxime al centro del agrupamiento. Al no semejarse un pixel candidato a ningún grupo se creará un nuevo grupo con ese único elemento.

1.4 Situación Problemática

Hoy el departamento de televisión de la Universidad, cuenta con un software escrito para el sistema operativo Windows, el cual tiene por nombre Where is it? que permite mantener y organizar las colecciones de imágenes y medias existentes. Esto ya indica una limitante debido a la necesidad de migración completa a software libre por la que está abogando la universidad y el país, lo que implica utilizar sistemas operativos como Ubuntu y Nova.

A su vez al no contar el departamento con un servicio online, donde los archivos audiovisuales se encuentren montados en un servidor, se necesita quemar periódicamente en un DVD los metadatos de cada media, que mediante el software podrá ser accedido, lo que permitirá cargar en una imagen virtual todas las fichas de información y guardarlas en un catálogo. Para buscar un contenido en específico el software permite realizar búsquedas avanzadas y básicas, usando categorías que filtren

la búsqueda (por archivos o carpetas, por un formato en específico, etc.), unido a la búsqueda por palabras clave que permite comparar los nombres y las descripciones de los archivos guardados con la indicaciones que se introduce.

Esta forma de búsqueda a veces no satisface en un 100% las necesidades del usuario. En muchas ocasiones, la urgencia de buscar imágenes predeterminadas, y saberlas ubicar, tanto en un minuto en específico de la cinta o en un conjunto grande de imágenes, provoca que el usuario pueda perder de vista lo que busca, debido a la dificultar de desplazar fragmento a fragmento el video, o buscar imagen a imagen dentro de un aglomerado conjuntos de ellas y por ende tendría que realizar esta operación una y otra vez para obtener el resultado deseado. Tales deficiencias puede ser erradicadas por un componente que sea capaz dada una imagen, encontrar aquella más semejante.

Este componente también agilizaría el proceso de catalogación en el departamento de televisión de la UCI, puesto que al digitalizar el material, varias personas tienen que trabajar en describir el contenido. Siendo tedioso visualizar todo el archivo para realizar una breve descripción. Sin embargo con el componente se podría buscar automáticamente algunas imágenes que infieran el contenido y otras especificidades de la media, por ejemplo en la sinopsis de los capítulos de alguna serie suele incluirse los actores que participan, pero en muchas ocasiones existen actores invitados que son obviados de la descripción, pudiendo ser importante para algún usuario. Debido a la dificultad de emplear tiempo en recorrer el video e identificar manualmente nuevas figuras en las escenas, un componente recuperación de imágenes permitiría comparar una serie de imágenes con los frames del video, indicando si hay correspondencia entre ellas de una manera fácil y automática.

Estos problemas no solo afectan a la UCI, sino también a empresas mayores como las televisoras de Cuba ICRT y de Venezolana VTV. Donde la búsqueda de archivos audiovisuales se realiza a través de palabras claves, y no mediante un análisis de imágenes que podría agilizar los procesos y automatizarlos. La necesidad de un componente que realice tal tarea pudiera no solo ser útil en la catalogación o la edición, sino también en la confección de contenidos nuevos que se alimente de imágenes procedentes de disímiles archivos de video, por ejemplo documentales, teleclases, videos publicitarios.

Actualmente no existe en Cuba ningún software que permita darle respuesta a estas necesidades y las empresas del país que manejan contenido tanto visual como audiovisual se ven afectado por esta limitante.

1.5 Análisis de otras soluciones existentes

Desde los años 90 los sistemas de recuperación de imágenes basados en contenido, han incorporado nuevas técnicas, y enfoques de búsqueda, examinando como mejorar las consultas y sus resultados, de tal manera que no se dependa de la intervención humana, en la elaboración de características textuales, que no son totalmente fiables debido a la dificultad de expresar mediante palabras las cualidades gráficas y las sensaciones estéticas que proporciona la percepción de una representación visual (Alvarez, 2003).

En la búsqueda del perfeccionamiento han surgido muchos sistemas que permiten hacer búsquedas de contenido visual. Uno de los primeros sistemas fue QBIC (Query By Image Content) desarrollado por la IBM[9] en 1995. Consistía en una interfaz web mediante la cual se permitía hacer consultas a través de las imágenes, dibujos realizados por el usuario y/o características de color o patrones de textura, estos dos últimos se podían escoger según las muestras que proporcionaba la interfaz y las barras de color ajustables (Yuste Cortés, 2009).

En el año 1999 sale un nuevo producto llamado Excalibur Visual RetrievalWare, desarrollado por Excalibur Technologies Corporation[10]. Las consultas se realizaban especificando la importancia de los atributos visuales como: color, forma, textura, brillantez y estructura de color (Yuste Cortés, 2009).

Via2 Platform es otro sistema diseñado para gestionar el contenido multimedia de manera óptima, cubriendo el proceso de digitalización, captura, catalogación y explotación de video, audio e imágenes (Via2 Platform, 2004). El proceso de búsqueda de información se realiza a partir de descriptores, palabras claves, tiempo, escenas, imágenes y personajes, para lo cual implementa el reconocimiento

[9]IBM (International Bussiness Machine) es una empresa multinacional estadounidense de tecnología y consultoría. Fabrica y comercializa hardware y software para computadoras, y ofrece servicios de infraestructura, alojamiento de Internet, y consultoría en una amplia gama de áreas relacionadas con la informática, desde computadoras centrales hasta nanotecnología. Fue fundada en 1911.

[10]Es una compañía privada que produce software, computadoras y periféricos.

facial a partir de una base de datos de rostros y el reconocimiento de imágenes se puede hacer a partir de nombres, imágenes similares, o dibujos esquemáticos de la misma (Visual Century, 2003).

En el 2006 Attrasoft Inc[11]desarrolla ImageFinder un sistema CBIR para Windows desarrollado por la compañía Attrasoft. Este sistema de recuperación de imágenes se comercializó en tres productos diferentes (ImageFinder, Internet ImageFinder e Image Hunt), que utilizaban la misma tecnología, y su única diferencia era el diseño de sus interfaces. Actualmente solo sobrevive ImageFinder. La cual es una herramienta compleja, no se utiliza solamente para realizar búsquedas, sino que también permite el tratamiento y el procesado de imágenes, por lo que el destinatario final es un usuario experto, familiarizado con estas técnicas. El método de consulta que implementa es por imagen ejemplo, donde el usuario puede escoger el ámbito de búsqueda y la fuente (un directorio, una base de datos, etc.) (Yuste Cortés, 2009).

GazoPa Similar Image Searcher fue otro producto desarrollado con el interés de conseguir búsquedas de imágenes más sofisticadas. Se desarrolló por Hitachi[12] en el 2008, es mucho más avanzado que los anteriores, pues ya con él se pueden hacer búsquedas mediantes imágenes, proporcionada por el usuario a partir de una URL**.** Además de imágenes, también puede analizar imágenes congeladas de un video y buscar en la web otros videos que sean similares (a partir de imágenes clave de video (keyframes)) (Yuste Cortés, 2009).

En el 2008 también se desarrolla Picollator, un buscador de imágenes especializado en el reconocimiento facial desarrollado por la empresa de origen ruso Recogmission LLC[13]. El mismo permite formular la búsqueda a través de texto, una imagen en particular proporcionada por un dirección o ambos métodos a la vez (Yuste Cortés, 2009).

---

[11]Empresa estadounidense fundada en 1995 que se dedica al desarrollo de software para la búsqueda y reconocimiento de imágenes.

[12]Es una empresa japonesa fundada en 1910 que produce electrónica de consumo y provee a otras fábricas de circuitos integrados y otros semiconductores.

[13]Compañía Rusa establecida en el 2006 que está enfocado en el desarrolló motores híbridos de búsqueda en Internet, que es capaz de utilizar cualquier tipo de datos, incluyendo texto y multimedia.

En este mismo año la empresa Idée Inc[14] lanza Piximilar, una herramienta especializada en búsqueda de imágenes pertenecientes a una colección. Trabaja con el método de búsqueda por imagen similar o por selección de varios colores (Yuste Cortés, 2009).

Existen otros sistemas como BLOBWORLD y QUICKLOOK que permiten buscar mediante los rasgos característicos de una imagen (color, forma, textura y distribución espacial). BLOBWORLD permite además la recuperación por palabras claves, y segmenta las imágenes en regiones (blobs) a las cuales se le asocian colores y descriptores textuales, cada región tratará de corresponderse de forma aproximada a los objetos que se quieren encontrar y es importante destacar que no está disponible comercialmente (Alvarez, 2003). El CIRE es otro sistema CBIR que es capaz de distinguir distintos objetos en una imagen, utiliza para ello agrupamiento jerárquico de las características de la imagen de bajo nivel, como el color y la textura, usa técnica de histograma de color y no realiza segmentación, ni representación detallada de los objetos (La Serna, y otros, 2010).

Dentro de estos sistemas aunque resuelven el problema planteado, muchos son motores de búsqueda web, como Picollator, CIRE, GazoPa Similar ImageSearcher que brindan servicio a toda la comunidad y no se especializan en entornos específicos, otros no están disponibles comercialmente y algunos como ImageFinder se desarrollaron para Windows, y muchos otros no brindan el código para poder modificarlos, como Via2Platform. Por ende la UCI junto al proyecto de Captura y Catalogación de Media, tienen como tarea desarrollar un software libre que satisfaga las necesidades antes expuestas.

1.6 Conclusiones Parciales

Como se pudo apreciar en el capítulo, los sistemas CBIR intentan adentrarse en el contenido de las imágenes para retornar resultados más precisos, que se acerquen más al criterio humano de percepción y diferenciación de distintos escenarios. Por lo que se necesita adquirir su método de trabajo para la realización de un componente eficaz que sea capaz de recuperar imágenes similares a una imagen de consulta. Para su conformación queda demostrado que la utilización de descriptores de alto nivel, robustos e invariantes a diferentes transformaciones de la imagen, indican mejores resultados, pues la capacidad de detectar imágenes afines bajo cambios de sombra, iluminación y rotación se precisa en la obtención de resultados favorables, pues ignora las condiciones a las que

[14]Campania canadiense fundada en 1999 dedicada al desarrollo de software para la identificación de imágenes y búsqueda visual.

hayan sido expuestas. El uso de un vocabulario visual reducirá los datos de entrada, debido al agrupamiento de características similares, lo que proporcionará mayor rapidez en la recuperación.

También queda expuesto en el capítulo que aunque existen muchas otras aplicaciones que aseguran la búsqueda y recuperación de imágenes, no se pueden incorporar a la solución del proyecto de Captura y Catalogación de media, por problemas de compatibilidad en cuanto a plataformas de desarrollo, y comercialización, sin contar que no pueden ser incorporada como parte del proyecto, ya que satisfacen otras especificidades de su propio negocio. Es por ello que se necesita implementar una solución propia, para la búsqueda y recuperación de imágenes basadas en contenido.

CAPÍTULO 2: HERRAMIENTAS Y TECNOLOGÍAS

En el presente capítulo se recogerán las tecnologías a utilizar, para el desarrollo del componente.

2.1 Arquitectura

La arquitectura en tres capas permite delegar responsabilidades a niveles específicos, donde un nivel superior solo tiene comunicación con el nivel inferior inmediato. La comunicación entre las capas es un proceso continuo, en forma de cascada, que se dispara en un único sentido.

Este tipo de arquitectura permite la reutilización posterior de las capas, y una mejor actualización de la misma sin tener que interferir en las capas adyacentes, esto hace que los procesos de refinamiento y cambio sean mucho más rápidos y por ende menos costosos. Además la determinación de funcionalidades a sus capas pertinentes, propicia a la mejor lucidez de la implementación y al encapsulamiento, donde una capa no tiene por qué conocer el comportamiento de las demás, solo solicitar y adquirir información, de las mismas.

El modelo de tres capas cuenta con:

- Capa de Presentación: Esta capa presenta las interfaces de usuario, e interactúa con ellos procesando sus solicitudes, que son a su vez manipuladas por la capa de negocio según se solicite.

- Capa de Negocio: Esta capa se encarga de automatizar los procesos, e implementar los algoritmos que le dan respuestas a las solicitudes de los usuarios.

- Capa de Acceso a Datos: Esta capa manipula (actualizar, borrar, adicionar) los datos persistentes, que pueden estar guardados en una base de datos, o en cualquier archivo XML, TXT, Excel, que permita registrar información, haciendo que no dependa del tiempo de vida de la propia aplicación.

2.2 Librería

Las librerías de visión por computador facilitan el manejo y procesado de las imágenes, como son The Matrox Image Library (MIL), Khoros, eVision, HIPS, Exbem, Aphelion. Sin embargo estas poseen ineficiencias plausibles: sus ciclos de actualización son largos y por ende lentos, algunos carecen de un entorno de desarrollo de alto nivel, dificultando su uso, y los que disponen de ellos, están limitados por la plataforma de desarrollo y el propio hardware de captura. Por otro lado se encuentra OpenCV, VXL, LTI-Lib. Este último es un producto escrito para Windows, siendo la característica clave que impide utilizarlo en este trabajo. VXL por su parte está diseñado para ser portátil en muchas plataformas, el mismo trabaja con matrices, vectores, descomposiciones y optimizadores, además de poder cargar, guardar y manipular imágenes en muchos formatos de archivo comunes, incluyendo imágenes muy grandes, sin mencionar la capacidad que posee de trabajar con puntos, curvas y otros objetos elementales de 2 o 3 dimensiones. Una de sus desventajas fundamentales que lo hace inferior a OpenCV es que no posee un marco de trabajo completo para el desarrollo de aplicaciones relacionadas con la visión por computador, careciendo por ejemplo de algoritmos de clasificación y algoritmo de detección de contornos (Pons Calvo, 2008).

Para el desarrollo de esta investigación se utiliza la librería de OpenCV 2.2, que posee más de 500 funciones desarrolladas en C y C++ las cuales son utilizadas para desarrollar tanto productos comerciales como no comerciales. La librería actúa bajo la licencia BSD, por lo que es libre y de código abierto, siendo uno de los requisitos primordiales para desarrollar el componente que dará solución al problema descrito en el capítulo anterior, además puede ser utilizada tanto en Windows como Linux. Además esta versión agrega el modulo features2d, que implementa detectores de características y provee herramientas de alto nivel para el macheo de imagen y detección de objetos y texturas.

La librería está formada por:

CxCore: Contiene funciones para estructuras de datos, algebra de matrices, transformación de datos, persistencia de objetos, manejo de memoria, manejo de errores, carga dinámica de código, dibujo, texto y matemática básica.

CvReference: Contiene funciones para procesamiento de imágenes, análisis de estructura de imágenes, captura de movimiento, reconocimiento de patrones y calibración de cámaras.

CvAux: Contiene interfaces de usuario y funciones de imagen/video y memoria.

Highgui: Posee funciones que permiten, además del control de entrada y salida de video, rutinas de interfaces gráfica para usuarios y manejo de grabación y lectura de archivos de imagen y video.

Machine Learning (ml): Contiene muchas funciones de clustering, clasificación y de análisis de datos.

CvCam: Es un módulo de procesamiento de video para cámaras digitales el cual esta implementado como una librería de vínculos dinámicos (DLL) para Windows y una librería de objetos compartidos (SO) para Linux. Esta sección es muy parecida a la sección Highgui. Tiene las mismas funcionalidades definidas de otra forma. Además permite acceder fácilmente a las propiedades de la cámara tan solo utilizando las funciones del módulo, evitando engorrosas configuraciones de controladores y la modificación de cuadros por segundo (Cia Ulacia, y otros, 2010)(Martínez Mejia, 2005)

Estos módulos convierten a OpenCV en la librería de visión por computador más utilizada, debido al sin número de funciones que puede realizar, en el procesado de imágenes, trabajo con videos, detección de objetos y contornos que la hacen imprescindible para la búsqueda de imágenes por contenido. Sin contar que fue diseñada para la eficiencia computacional y con un fuerte enfoque a aplicaciones en tiempo real.

2.3 Lenguajes

2.3.1 C++

La utilización de la librería OpenCV para el tratamiento de imágenes, fuerza a utilizar como lenguaje base C++, lo que permite la compatibilidad y fusión de las sentencias netamente de OpenCV y el código propio de la aplicación.

El lenguaje C++ es unos de los más utilizados, siendo una versión mejorada del lenguaje C. Incorpora la programación orientada a objetos y da la posibilidad de redefinir los operadores, es decir, permite la sobrecarga de operadores, y de poder crear nuevos tipos que se comporten de manera diferente. Además añade al tratamiento de excepciones. Como complemento C++ permite trabajar tanto a alto como a bajo nivel (Bustamante, y otros, 2004)(Stroustrup, 1985).

Su uso hace más de 20 años demuestra su consistencia y utilidad, y por ende se ha estandarizado pudiéndose ejecutar en cualquier plataforma. Una de las características que lo hace el mejor candidato para el desarrollo del componente es su eficiencia, siendo uno de los lenguajes más rápidos, debido a que es un lenguaje compilado donde el proceso de traducción del código se realiza una vez, lo que provoca resultados muchos más compactos y menor utilización de la memoria. Es además un lenguaje de propósito general, lo que indica que con él se pueden desarrollar sistemas operativos, compiladores, aplicaciones de base de datos, juegos, entre otros programas con funciones diversas. Otras de las características que lo hace potente para el desarrollo del componente es su capacidad de estructurar de manera racional programas grandes.

2.3.2 UML

Como lenguaje de modelado se propone el uso del Lenguaje de Modelado Unificado (UML), creado por la OMG (Grupo de Gestión de objetos o por sus siglas en inglés Object Management Group) con el propósito de definir los sistemas de software, detallar sus artefactos y documentarlos, siendo un proceso útil para la construcción posterior.

Una de las ventajas de UML es que es de propósito general y un mecanismo estándar para el modelado orientado a objetos que asegura llevar a un lenguaje común todo el diseño y análisis de un software. Además con la versión utilizada la 2.0, se definen una serie de relaciones y diagramas adicionales que permiten la producción automática de programas basados en la especificación del software. Con el mismo se pueden modelar todas las fases de un proyecto, tanto el análisis, con los diagramas de caso de usos, el diseño del diagrama de clase que se aplica al componente, así como la implementación con el diagrama de componentes y despliegues. Lo que permite la validación y posterior verificación, y adjuntado a esta, una documentación consistente que sirve para futuras versiones. Estas características hacen que prácticamente todas las herramientas CASE (Ingeniería de Software asistida por ordenador o por sus siglas en inglés Computer-Aided Software Engineering) la hayan adaptado como lenguaje de modelado.

2.4 Herramienta CASE

Las herramientas CASE son programas informáticos que dan asistencia a los analistas en el proceso de desarrollo de software, permitiéndole de manera automatizada generar diagramas, documentación,

hasta código fuente de programas, lo que conlleva a la mejor planificación y distribución del tiempo, sin contar con la facilidad de corrección de errores que otorga.

Debido a su importancia se han creado un sin número de herramientas CASE destacándose entre ellas el Visual Paradigm que además de existir en una versión comercial, también puede ser utilizado bajo licencia gratuita conocido en estos casos como el producto Community Edition.

2.5 Visual Paradigm

El Visual Paradigm 8.0 es una herramienta CASE que permite diseñar diagramas UML. Puede ser usado bajo licencia gratuita y es multiplataforma, lo que es ventajoso para el desarrollo del componente en el sistema operativo libre Linux. Unas de sus características fundamentales son la facilidad de operación y generación de diagramas entre ellos los de caso de uso, actividades, secuencia, colaboración entre otros, de diseño de clases y su facilidad de instalación y actualización, agregado a su compatibilidad con otras ediciones y su capacidad de ingeniería directa e inversa.

2.6 Entorno de desarrollo (IDE)

La utilización de OpenCV como librería de visión por computador y de C++ como el lenguaje base del componente estimula a la utilización de Qt Creator como IDE de desarrollo. Este IDE posee un avanzado editor de código C++ , lenguaje que utiliza de forma nativa, y capacidad de vincularse fácilmente con la librería OpenCV, permitiendo aprovechar la potencia de esta para el procesamiento de imágenes y video y la facilidad del IDE para el desarrollo rápido de aplicaciones interactivas en entornos de ventanas. Ambas son multiplataforma, una de las exigencias para desarrollar el componente de búsqueda y recuperación de imágenes. Su uso es abierto y gratuito, otorgándole libertad al programador de ver el código fuente y poder modificarlo, además de utilizarlo sin restricciones de licencia.

El framework Qt está destinada fundamentalmente para el desarrollo de interfaces, sin embargo facilita ciertas tareas de programación como el trabajo con socket, la programación multihilos, soporte de red y el trabajo con archivos donde será almacenado vectores, matrices y otras estructuras de datos, que se utilicen en la aplicación propuesta (Martínez Muñoz, 2011). Qt permite además el desarrollo ágil y óptimo del componente ya que con él se puede construir herramientas de gestión, dar soporte para el control de versiones, crear librerías y la implementación de Señales y Slots permitiendo al programador

tomar control sobre los eventos que se disparan asociando a un comportamiento indicado, dando cumplimento a determinadas funcionalidades

2.7 Conclusiones Parciales

En el presente capítulo se hizo referencia a las herramientas y tecnologías para el buen desarrollo del componente, aportándole características de usabilidad y consistencia. La utilización de tecnologías libres y multiplataforma como la librería OpenCV, Qt Creator y Visual Paradigm, aseguran que el componente pueda en un futuro ser usado sin necesidad de pagar licencia, ni poseer tiempo límite para actualización o condiciones reducidas de aplicación. Además es una ventaja, para la campaña en la cual está involucrado el país en la migración a software libre. Con tales tecnologías se cuenta con elementos robustos para el procesamiento de imágenes, y la realización de los algoritmos que intervienen en la solución de la problemática descrita, siendo C++ un lenguaje orientado a objeto que posibilita mayor rapidez en cuanto a compilación se refiere y Qt Creator un IDE de desarrollo capaz de soportar este lenguaje y posibilitar el diseño potente de interfaces gráficas, en la conformación de un sistema que sea agradable al usuario.

CAPÍTULO 3: PROPUESTA DE LA SOLUCIÓN

3.1 Propuesta de Solución:

Propuesta del Algoritmo

En el algoritmo de recuperación de imágenes propuesto se definen dos procesos fundamentales, el de entrenamiento y el de recuperación, requiriendo este último del resultado del proceso anterior. El proceso de entrenamiento, conlleva a adiestrar el sistema a partir de un conjunto de imágenes de entrada, que constituyen el repositorio de imágenes a recuperar. El mismo podría desglosarse en las etapas: extracción de características, conformación del vocabulario y construcción de los vectores prototipos o representativos de cada imagen. El segundo proceso tiene como entrada una imagen a la cual se le aplican 3 etapas: extracción de características, construcción de su prototipo, y la determinación de las imágenes semejantes. Estas etapas serán descritas con profundidad en el presente capítulo.

Definición Formal del Algoritmo

Dado $I = \{i_1, i_2, ...i_n\}$ el conjunto de n imágenes de entrenamiento, se tiene un vector prototipo $P_j \ \forall \ i_j$ tal que P_j representa a la imagen i_j en el conjunto de imágenes I. Una vez obtenido el prototipo P_y de una imagen de consulta y se le asocia el conjunto M de imágenes, $M \subset I$ tal que para todo $i_j \in M$ se cumple que la distancia $d(P_i, P_y) = \min(d(P_i, P_y))$.

3.1.1 Proceso de Entrenamiento

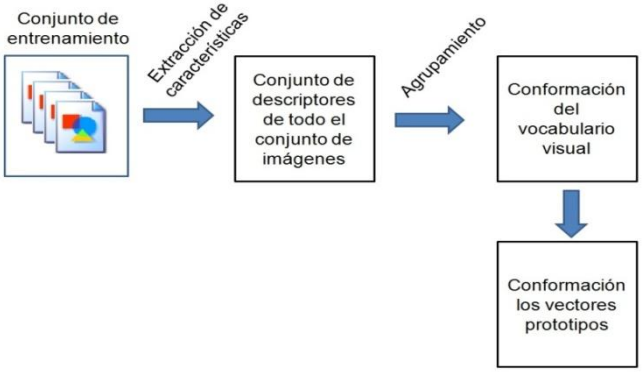


Figura 1: Proceso de Entrenamiento.

Primera Etapa

Caracterización de los datos

Una imagen puede ser distinguida por un conjunto de características que se basan en su información visual. La detección de las características es un proceso donde se extraen los puntos singulares que especifican la escala y orientación de una ubicación en la imagen. Estos codifican información proveniente de los pixeles en su vecindario, y son estables a transformaciones afines y fotométricas.

Para la extracción de las características el mejor descriptor a utilizar es el descriptor SIFT, debido a que retorna como resultado características locales que son invariantes a traslación, rotación, reducción de escala, amplificación, cambios de brillo, oclusión y ruido, además pueden ser estables a cambios de visión y transformación afín, hasta cierto punto. Este es un factor importante, ya que muchas imágenes suelen representar el mismo contexto u objetos, pero vistos desde diferentes puntos de vista, y no bastaría identificar puntos que son solo invariantes a transformaciones fotométricas, pues se pudieran detectar dos imágenes diferentes cuando en realidad, representan el mismo contenido pero desde otra perspectiva. Una mayor fundamentación de este descriptor se puede ver en la sección 1.3.4.

El SIFT implementa el operador de puntos de interés DoG visto en la sección 1.3.5, que permite rapidez en la búsqueda de los puntos, pues el algoritmo se basa en una simple resta de escalas posteriores en cada octava. Además en (Mikolajczyk, 1999) se afirma que los máximos y mínimos encontrados produce las características de imagen más estables que otras funciones como el Gradiente, el Hessiano o el Harris Corner Detector y devuelve mayor cantidad de puntos que otros operadores, siendo clave en la realización de aplicaciones de recuperación de imagen, pues al tener mayor información, se puede establecer con mayor certeza las comparaciones, pues existe un mayor criterio de decisión (Boullosa, 2011).

Debido a las limitaciones de las máquinas que actualmente se utilizan para realizar las pruebas, y la cantidad de puntos de interés que retorna el descriptor SIFT, siendo imposible generar estos datos por la insuficiencia de la memoria RAM y la demora del procesamiento, se empleó el descriptor SURF por su rapidez computacional ya que los filtros se realizan sobre la imagen original, con la utilización de filtros de tipo caja e imágenes integrales, y la dimensión de los descriptores contienen la mitad de las características del descriptor SIFT. (Ver la sección 1.3.4). Para la detección de los puntos de interés utiliza el operador Hessian-Laplace que aunque no es invariante a transformaciones afines, puede detectar regiones estables a cambios de rotación y escala. La selección de este método asegura que la demora del procesamiento de las imágenes se reduzca.

Para ello se emplea el descriptor SURF que provee la librería OpenCV con la función cvExtractSURF el cual determina los puntos de interés de una imagen y sus respectivos descriptores de 64 características. Como resultado de este proceso se obtiene un archivo txt por cada imagen, que contiene los descriptores de cada punto singular.

Segunda Etapa

Conformación del vocabulario

El vocabulario (este tema ha sido descrito en la sección 1.3.3) es la representación matemática del concepto de saco de características, que representa a las imágenes como un conjunto poco ordenado de características locales. A partir del agrupamiento de las mismas, cada grupo o clúster puede representar una palabra visual o una colección de características similares, que pasaría a formar parte del vocabulario. De esta manera cada grupo es reducido en casi su totalidad, pues solamente se recoge aquel punto cuyas características puedan representar a todo su conjunto.

Después de la extracción de las características de las imágenes de entrenamiento visto en la etapa anterior, se procede a construir el vocabulario, a partir de un sin número de característica que no guardan correspondencia con la imagen de la cual fue extraída, intentando encontrar similitudes entre ellas, para poder conformar las palabras visuales. De tal manera que los grupos no representan a una imagen en específico, sino al contenido de muchas de ellas, que son semejantes y muestran objetos o contextos similares.

Método de Clúster Jerárquico

Los métodos de agrupamiento, se basan en la congregación de características, actuando sobre un conjunto no ordenado de valores, de esta manera particionan un grupo de objetos, para conformar grupos de atributos homogéneos, tales que los patrones de cada grupo sean similares. Existen dos subgrupos de algoritmos para la conformación de clúster, en el primero de ellos el número de grupos está predefinido, situación que no sucede en el segundo grupo, los mismo comprenden a los algoritmos particionales y a los algoritmos jerárquicos respectivamente.

La predeterminación de una cantidad de clúster, para los algoritmos particionales, puede retornar resultados pocos favorables, pues depende de las conjeturas del usuario, y no de los propios datos. Si este valor falla en sus inicio, el algoritmo estaría iterando sobre un error irreparable, y conformaría asociaciones que no se corresponden con la realidad. Es por esta limitante que se decide utilizar en la investigación los algoritmos jerárquicos (Ver sección 1.3.7), que establecen una jerarquía en los datos,

para luego a través de particiones iterativas conformar los grupos mejores asociados. Además en los enfoques de recuperación de imágenes suele aplicarse el método k-means que utiliza la distancia euclidea por definición, la cual no sería apropiada pues las características que fueron extraídas no están representadas esféricamente.

Para conformar los grupos o palabras visuales, se construye un árbol con los puntos de interés de todas las imágenes, extraídos en el paso anterior. El proceso de construcción del árbol jerárquico depende de la distancia que exista entre cada par de puntos. Mientras que cada elemento del par esté conformado por un solo punto, se utiliza como distancia la correlación de Pearson, de no ser así se aplica una medida de Average-Linkage, que puede establecer distancias entre grupos de puntos. De esta manera la conformación del árbol jerárquico ascendente parte de las hojas que están representadas por cada punto de interés del conjunto de imágenes de entrenamiento, que son analizadas como clústeres independientes y se combinan iterativamente usando una función de similitud, ya sea Pearson o Average-Linkage entre pares de clústeres, hasta conformar un clúster único. Es importante destacar que en la medida que se van uniendo los pares en un nuevo nodo dentro de la jerarquía del árbol estos distan de las hojas a una distancia mayor por tanto ninguno está al mismo nivel.

Definición Formal

Sea P(A, B, C,…) el conjunto de puntos de las imágenes de entrenamiento, y A y B dos grupos con n_A, y n_B elementos. Si d(A, B) = min entonces AB es un nuevo grupo. Si la distancia de AB a otro grupo C de n_C elementos es d(C; AB)= min entonces se conforma un nuevo grupo ACB. Este proceso es iterativo, hasta aglomerar a todos los puntos de P.

Para realizar tal procedimiento se necesita almacenar en una matriz, la distancia inicial entre los puntos, a medida que los grupos se van uniendo, se actualiza la matriz, con el nuevo par conformado y su respectivo valor de distancia hacia los demás puntos dentro de la matriz.

La matriz de distancia es una estructura la cual tiene k fila, siendo k el total de puntos de interés del conjunto de imágenes de entrenamiento donde para cada fila i se reserva el espacio en memoria para i valores double, garantizando así que sea una matriz triangular inferior.

Una vez que en el proceso de conformación del árbol se determine el par de puntos a unir en esta matriz se eliminaría tanto las filas como las columnas correspondientes a este par de puntos y se

agregaría una nueva fila que contiene las distancias del nuevo nodo a el resto de los nodos de la matriz.

Correlación de Pesaron

La correlación de Pearson es una media muestral entre dos vectores x e y donde $\bar{\mathbf{x}}$, $\bar{\mathbf{y}}$ son las medias de cada vector, y $\boldsymbol{\sigma_x}$,$\boldsymbol{\sigma_y}$ constituyen las desviaciones estándar de la respectiva muestra x e y. Esta función de distancia indica una medida de que también una línea recta puede ser instalada en un gráfico de dispersión x e y. Su objetivo es medir la relación lineal entre dos variables aleatorias cuantitativas, es decir en variables que expresan distintas cualidades o características, indicando el grado de dependencia que se establece entre ellas. La correlación de Pearson se define como sigue:

$$\boldsymbol{d_p = 1 - r} \qquad \textbf{(1)}$$

$$\boldsymbol{r = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - \bar{x}}{\sigma_x}\right)\left(\frac{y_i - \bar{y}}{\sigma_y}\right)} \qquad \textbf{(1.1)}$$

En el presente trabajo se emplea la función de distancia de Pearson descentrada, la cual es la misma que la definición ordinaria solo que toma la media de cada vector con valor fijo, cero. Esta medida es escogida debido a que retorna valores positivos de distancias, mientras que la correlación de Pearson original da resultados entre [-1,1]. La distancia de Pearson descentrada se define como sigue:

$$\boldsymbol{d_u = 1 - r_u} \qquad \textbf{(2)}$$

$$\boldsymbol{r_u = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i}{\sigma_x^{(0)}}\right)\left(\frac{y_i}{\sigma_y^{(0)}}\right)} \qquad \textbf{(2.1)}$$

$$\boldsymbol{\sigma_x^{(0)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\,,\ \sigma_y^{(0)} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} y_i^2}} \qquad \textbf{(2.2)}$$

Función de similitud entre pares de clústeres(González Linares)

El cálculo de la distancia entre dos nuevos nodos formados del árbol se determina haciendo uso de la función de similitud average-linkage, la cual determina el promedio de todas las distancias entre los pares de puntos de interés de los dos subnodos. La cual combina los grupos de tal forma que la distancia promedio entre todos los clúster resultante sea la más pequeña. Esta función de similitud se define como:

$$d(C_q, C_s) = \frac{1}{2}\left(d(C_i, C_s) + d(C_j, C_s)\right) \quad \textbf{(3)}$$

Siendo: C_q el clúster resultado de unir los clústeres C_i y C_j, y C_s un clúster formado con anterioridad.

Ejemplo de Seudocódigo

```
Buscar_menor_par(distmatrix_copia)
inicio
    menor_dist <- MAX
    par <- (0,0)
    para i desde 0 hasta elementos_distmatrix_copia hacer
        para j desde 0 hasta i hacer
            si menor dist < distmatrix_copia[i,j]
               menor_dist <- distmatrix_copia[i,j]
                par <- (i,j)
            fin_si
        fin_para
    fin_para
    devolver par
fin

Calcular_tree(distmatrix, cant_elementos)
inicio
    distmatrix_copia <- distmatrix
    tree <- vacio
    para i desde 0 hasta cant_elementos -1 hacer
            par <- Buscar_menor_par(distmatrix_copia)
            tree[i] <- nuevo Node(par)
            actualizar(distmatrix_copia,par)
    fin_para
    devolver tree
fin
```

Descripción de las variables:

menor_dist: es la menor distancia que existe entre los todos los pares de puntos.

par: es el par de puntos que tiene la menor distancia.

distmatrix_copia: es la matriz en la que se actualizarán los valores según se vayan uniendo los pares.

Método de validación de clúster

Para conocer que tan bueno es un grupo u otro se utiliza un CVI o índice de validación de clúster, que se basan por lo general en el análisis de la cohesión y la separación de sus agrupaciones. En algunos casos la mejor partición se selecciona mientras que el algoritmo está haciendo la construcción de la jerarquía, conociéndose el CVI como criterio de parada. Este enfoque no es válido en muchos casos debido a que los cortes horizontales que genera pueden ignorar la mejor partición. Por lo tanto se define un espacio de búsqueda nuevo que contiene todas las particiones del árbol, con un nuevo CVI, que se basa en evaluar las particiones por las ramas, y no por un nivel en específico, debido que en algunos casos la partición correcta no está explícitamente en la jerarquía, pero puede ser implícitamente descrito por ella.

Una partición no es más que un conjunto de grupos disjuntos, del conjunto de la base de datos, representados por una serie de puntos, las mismas pueden ser parciales cuando están hechas sobre un subconjunto de grupos o clúster de los datos de entrenamiento y total si se realiza con respecto al conjunto de datos generales.

En el presente trabajo se hace usó del COP como Índice de Validación de Clúster.

Propiedades de los COP

- Se basan en la relación intra e interclúster en el árbol.
- Dado que este método trata de minimizar la relación intra clúster y maximizar la relación inter clúster su valor está en un rango de [0,1].
- Se considera que a menor valor de COP mejor conformado se encuentra clúster.
    - Formulación Matemática

$$\boldsymbol{COP(P^Y, X) = \frac{1}{|Y|} \sum_{C \in P^Y} |C| \frac{intraCOP(C)}{interCOP(C)}} \qquad \textbf{(2)}$$

$$\boldsymbol{intraCOP(C) = \frac{1}{|C|} \sum_{x \in C} d(x, mean(C))} \qquad \textbf{(3.1)}$$

$$\boldsymbol{interCOP(C) = min_{x_i \notin C}\, max_{x_j \in C} d(x_i, x_j)} \qquad \textbf{(3.2)}$$

DondeP^Y es la partición parcial de todo el conjunto de datos X.

$d(x, mean(C))$: es la función de distancia del punto x al centroide del clúster C.

En la investigación se ignora el factor C/Y, donde C es la cantidad de características que conforman a un clúster, y Y la cantidad de clúster dentro de la partición. Esta fracción conlleva a que la ecuación de COP no solo dependa de la relación intra e inter clúster, sino también del total de puntos agrupados. Por lo que a mayor valor de C peor será el resultado y viceversa. Esto implica que el valor final dependa de una peso extra, que no es objetivo, pues lo que se busca no es cuantía sino que tan compacto es un clúster y cuán separado puede estar de los demás clúster en su partición, lo que identifica la mejor agrupación, sin importar la cantidad que la conforme. La función de otorgar importancia a los grupos se le otorga al TF-IDF, que se explica posteriormente en el capítulo.

Con el COP no se puede evaluar el nodo raíz y los nodos hojas, por lo que en estos casos se establece su valor como 1. En caso contrario el COP de un nodo puede ser calculado mediante la combinación de los COP de sus hijos. La función básica de este CVI es analizar una partición parcial Y de un conjunto de datos X, por lo que ignora todos los puntos que se encuentra en las particiones X-Y.

Algoritmo SEP-COP

La implementación del algoritmo SEP-COP está propuesta en (Gurrutxaga, y otros, 2010), este algoritmo es el encargado de determinar en el árbol de agrupamiento jerárquico los clústeres de mejor conformación. Para cumplir tal propósito se basa en una búsqueda con heurística, utilizando el CVI COP como indicador heurístico. El SEP recorre el árbol de arriba hacia abajo calculando el COP para cada uno de los nodos y de abajo hacia arriba determinado la mejor partición en cada rama. Como resultado se obtiene la mejor partición del árbol, o sea, la mejor cantidad k de clústeres en las que se puede separar ese conjunto de datos.

La determinación de una partición se realiza en un subárbol teniendo en cuenta la altura de cada uno de los hijos del nodo raíz, garantizando que existan como mínimo dos clústeres, lo que permite de esta manera calcular el COP para cada nodo. El algoritmo procede particionando a la altura del nodo izquierdo generando los clústeres correspondientes, el cálculo del COP para ese conjunto de clústeres representa el valor de CVI en este nodo, lo que sucede de igual manera para el nodo derecho. Este proceso es recursivo, repitiéndose el mismo procedimiento para cada hijo del nodo que se esté analizando, de esta manera todos los nodos serán pesados por una heurística COP.

Una vez determinado el COP para los hijos del nodo raíz actual, los dos conjuntos izquierdo y derecho son agrupados en un único conjunto y se determina para ellos el valor del CVI. Este valor es comparado con el peso del padre. La decisión se basa en escoger la unión de las mejores particiones de los nodos hijos u optar por un único grupo con todos los hijos del nodo raíz en cuestión. El algoritmo dividirá siempre el conjunto de datos en más de un clúster debido a que el CVI del nodo raíz del árbol es 1.

Construcción del Vocabulario

El vocabulario visual como anteriormente se había visto, se construye a partir del agrupamiento de características de un conjunto de imágenes entrenamiento, como un conjunto de palabras en un diccionario representan a un documento. Básicamente el vocabulario almacena una característica representativa por cada clúster o grupo al cual pertenece.

El procedimiento se basa en determinar el centroide para cada grupo o clúster, y almacenar el punto de interés más cercano a su centroide, en una matriz de k filas y m columnas, donde k es el número total de clústeres encontrado en el árbol y m representa el número de características del descriptor.

Tercera Etapa

Construcción de los vectores prototipos

El uso del vocabulario evita hacer comparaciones a nivel de imágenes, donde el procesado pixel a pixel causa una irreparable demora computacional, limitándose en cambio a manejar solo las características sobresalientes de las imágenes, descartando aquellas que aportan poca o ninguna información. De esta manera la comparación se reduce a chequear la similitud entre los prototipos de dos imágenes, la de consulta y la de entrenamiento, que no son más que vectores numéricos que registran la cantidad de apariciones de una palabra del vocabulario en la imagen, los cuales son conformados a partir de un método de asignación de peso, que se utiliza en la construcción de un vector término que indica la importancia de una pablara visual en el conjunto de imágenes de entrenamiento.

Métodos de asignación de pesos

La cuantificación de un conjunto de características presentan problemas cuando los descriptores están distribuidos en tal manera que los mecanismos simples de agrupamiento sobre-representa algunos

descriptores y sub-representan otros, lo que quiere decir que se puede ignorar características relevantes y reconocer aquellas que realmente no lo son. Una estrategia de mitigación a estos problemas, es asignar pesos a las características de los puntos de interés del vocabulario.

Con los pesos se puede penalizar a los términos encontrados a ser demasiados comunes para ser discriminativos y hacer hincapié en aquellos que son más singulares o únicos. En el presente trabajo se utiliza el método de asignación de peso TF-IDF.

Formulación Matemática del TF-IDF

$$TF - IDF = t\,f_i \log\left(\frac{N}{N_i}\right) \qquad (4)$$

Donde: $t\,f_i$ es un término de frecuencia de la i-ésima palabra del vocabulario.

N: es el número total de imágenes en el conjunto de entrenamiento.

N_i: Número de imágenes en el conjunto de entrenamiento que contienen la i-ésima palabra.

El TF-IDF se le determina a todas las palabras del vocabulario visual, conformando por cada una un vector término. Es un factor de peso que da la medida de que tan representativa es una palabra dentro del vocabulario. A mayor valor de TF-IDF menos representativa es la palabra.

Conformación de los Prototipos

Después de obtenidos el vocabulario y su respectivo vector término, también conocido como vector de pesos, el cual recoge el grado de relevancia de cada una de las palabras visuales que lo componen, se conforman los prototipos de cada imagen de entrenamiento. Este proceso consiste en analizar las características de cada imagen, para determinar que palabras están presentes en ella y así conformar su vector prototipo. Los prototipos constituyen vectores numéricos, de dimensión n, donde n es el número de palabras del vocabulario. Cada elemento del vector almacena la cantidad de ocurrencia de una palabra en la imagen por su correspondiente peso.

Tanto los vectores términos, como los prototipos cuando se utilizan vocabularios grandes son extremadamente diseminados o sparcer, lo que indica que el número de términos antes de cualquier normalización son en su mayoría 0.

3.1.2 Proceso de Recuperación

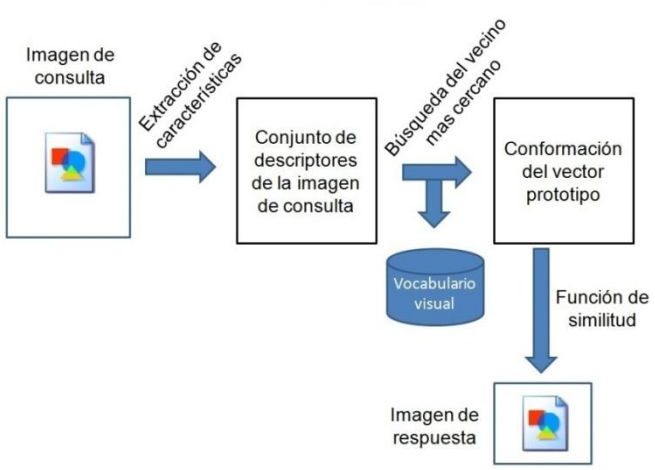


Figura 2: Proceso de Recuperación

En el proceso de recuperación se realiza el mismo procedimiento que indica la etapa 1 y 3 del entrenamiento. La primera etapa para obtener una imagen a partir de otra de consulta, consiste en aplicar un descriptor, obteniéndose un conjunto de características descriptivas. Una vez obtenido los descriptores de los puntos de interés se determina al igual que la etapa 3 del proceso de entrenamiento el prototipo de esta imagen, el cual es comparado con los prototipos de las imágenes que se utilizaron en el adiestramiento del sistema, mediante una función de disimilaridad. Este proceso es el que se muestra en la figura 2.

Función de Disimilaridad

La función de disimilaridad escogida es la correlación de Pearson, que establece una comparación entre vectores de datos numéricos. En este caso la imagen que se retorna como resultado de la recuperación, es la que cuyo vector prototipo conjunto al identificador de la imagen de consulta, después de aplicado la función de disimilaridad de más próximo a 1.

3.2 Diagrama de clases del diseño

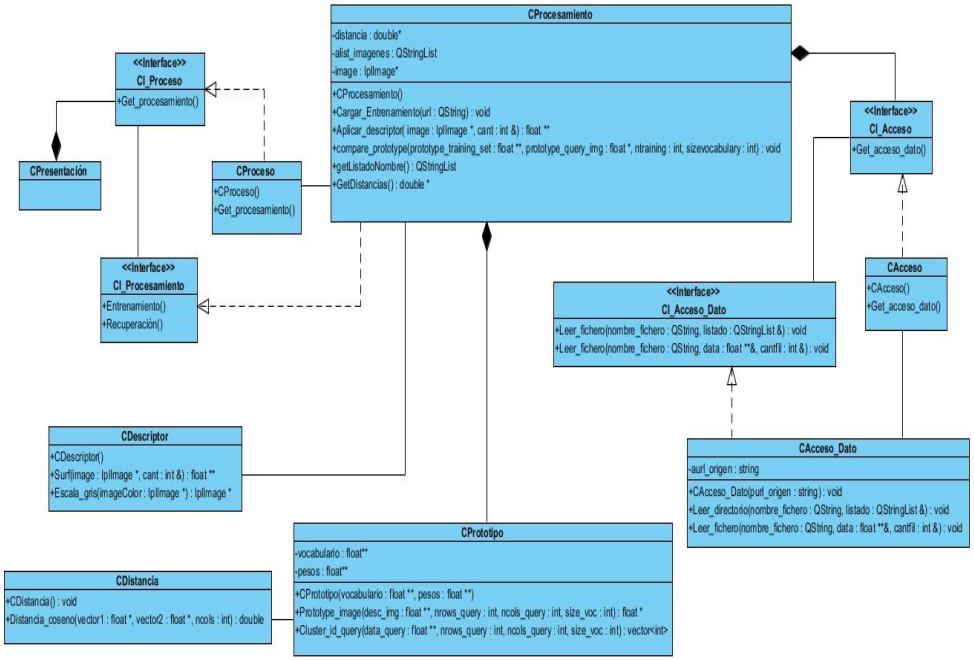


Figura 3: Diagrama de Clase del Proceso de Recuperación

La recuperación de imágenes está conformada por dos procesos, el de entrenamiento, donde se construye el vocabulario y los prototipos de las imágenes de entrenamiento y el de recuperación donde se obtiene aquella imagen de la base de entrenamiento cuyo prototipo sea el más semejante al prototipo de la imagen de consulta, como se había visto con anterioridad en el capítulo.

El proceso de entrenamiento es un proceso lento, que depende de la cantidad de imágenes de la base de datos, por ende el tiempo computacional es mucho más costoso cuando más imágenes se procesen. Debido a ello el mismo requiere ser un proceso offline, cuyo resultado debe estar listo antes que el usuario acceda a la aplicación, para garantizar eficiencia y rapidez. Por lo que necesita ser supervisado por un agente externo, que lo ejecute teniendo en cuenta las imágenes representativas del contexto del negocio, donde sea aplicado el sistema.

Por las razones anteriormente planteadas el diseño de clase esta modelado representando solo el proceso de recuperación, dado que fue el único que se integró al componente en aras de mostrar los resultados finales de la investigación de una manera más interactiva para el usuario. No se hizo así con el proceso de entrenamiento dado que no es urgente en estos momentos para el proyecto contar con esta aplicación.

El modelo que se propone es un modelo extensible, con el propósito de poder incluir nuevos descriptores y métodos de distancia según se requieran en un futuro. Esto posibilita darle continuidad al software de una manera sencilla. Las clases que lo conforman están descritas a continuación:

CProcesamiento:

La clase CProcesamiento conoce que clases son responsables de una determinada acción y delega funciones a los objetos apropiados. Ella se encarga de dirigir el proceso de recuperación donde se involucran a su vez otros procesos, como el de descripción y construcción de prototipos.

CDescriptor

La clase CDescriptor se encarga de procesar una imagen, en este caso la de consulta, determinándole los puntos de interés y las características asociados a ellos. Su función es incluir métodos estáticos, que implemente los diferentes descriptores que se deseen utilizar, pudiera ser SIFT o SURF. En la investigación se propone el segundo de ellos como se había visto con anterioridad en el capítulo.

CPrototipo:

La clase CPrototipo se encarga de construir el vector numérico representativo de la imagen de consulta. El cual se empleará en la comparación con los prototipos de las imágenes de entrenamiento. Se ha diseñado con el objetivo de que esta clase pueda ser aplicada para el proceso de entrenamiento.

CDistancia

La clase CDistancia almacena las posibles distancias que pudieran utilizarse, como la distancia euclidea y ucorrelation, métodos estáticos para calcular la similitud entre los prototipos y definir a que palabra del vocabulario pertenece las características de la imagen de consulta. Su objetivo es separar la implementación de estas funciones, ya que suelen ser utilizadas tanto en la clase CProcesamiento como en la clase CPrototipo, lo que evitaría tener que implementarlas más de una vez.

CAcceso_Dato:

La clase de CAcceso_Dato, accede a los archivos de textos producido por el entrenamiento, donde se almacena los pesos de las palabras del vocabulario, el vocabulario en sí y los prototipos de las imágenes de entrenamiento, para poder construir el prototipo de la imagen de consulta y realizar la recuperación.

3.3 Patrones de Diseño

Para el diseño de clase se utilizó el conjunto de patrones de diseño GRASP y GOF, que intervienen en la asignación de responsabilidades, estructura y comportamiento respectivamente. De los patrones **GRASP** se evidencia el **patrón experto** que permite asignarles las responsabilidades necesarias a desarrollar a cuyas clases contengan la información para realizarlas. En este caso cada clase está diseñada según su función en el proceso, la clase CDescriptor, describe a las imágenes según sus vectores características, la clase CPrototipo construye el prototipo de la imagen de consulta, la clase CDistancia posee diferentes funciones que permite calcular la distancia entre dos vectores, la clase CAcceso_Dato interfiere en la captura de los datos de los archivos .txt.

También se utiliza el **patrón creador** que indica cuando una clase puede crear instancia de otra, cuando la contiene o agrega, cuando posee datos necesarios para su inicialización, cuando la registra o cuando se relaciona estrechamente con sus objetos. En el diseño se evidencia que CProcesamiento, es la que tiene la facultad de crear los demás objetos, pues dirige y ejecuta las funciones de las clases que lo componen, por ende manipula los datos de las mismas y los relaciona.

Otro patrón que se manifiesta es el **patrón de bajo acoplamiento** que trata de establecer la menor dependencia posible entre las clases, de modo que sea más adaptable a cambios futuros. En el diagrama solo la clase CProcesamiento puede instanciar a la clase de CAcceso_Dato, teniendo en cuenta que aunque ella sea necesaria para guardar las características de la imagen de consulta y los prototipos de la misma, no será CDescriptor y CPrototipo quienes la instancie, pues de ser así la dependencia aumenta, así como la dificultad de nuevas modificaciones, viéndose involucrada más de una clase en el proceso. Tener un bajo acoplamiento asegura **alta cohesión** donde no toda la responsabilidad recae en una sola clase.

Para el **patrón controlador** se asigna la tarea de controlar a una clase que no pertenezca a la interfaz, y que por sí sola no realice ninguna funcionalidad sino que ordene a otras clases, a operar según se desee. Este patrón se evidencia en la clase CProcesamiento, que se encarga solo de delegar responsabilidades a las clases que posean los atributos y funciones para realizarlas, según las acciones que haya realizado el usuario en la interfaz.

De los patrones GOF el patrón fachada permite la existencia de una clase que conoce a quienes tiene que otorgar la responsabilidad de una determinada petición y delega esas peticiones de los clientes a los objetos asociados a estas clases. La ventaja fundamental de este patrón es que los cambios repercutirán mayormente en la clase fachada, asegurando un bajo acoplamiento. En el diseño de clase

este patrón se manifiesta en la clase CProcesamiento, que actúa como una portada, interactuando con la capa presentación, y solicitando a las demás clases funciones que respondan a las acciones del usuario.

Otro de los patrones **GOF** que se utiliza es el **patrón builder o constructor** que separa la construcción de un objeto complejo de su representación, lo que se evidencia en la comunicación entre las capas creándose entre las mismas un conjuntos de clases como CIAcceso, CAcceso, CIProceso y CProceso con las que se garantiza una dependencia débil entre las capas. De tal forma que la clase CProcesamiento no crea un objeto directo de la clase CAcceso_Dato, ni la clase CPresentación de CProcesamiento, garantizándose un bajo acoplamiento.

3.4 Conclusiones Parciales

En el presente capítulo se pudo apreciar la solución propuesta para confeccionar el componente de tal forma que genere respuestas aceptadas. La utilización del agrupamiento jerárquico junto con el índice de validación de clúster COP incorporado al método de búsqueda SEP, garantiza estructurar los datos según el acercamiento existente entre ellos, y generar agrupaciones de características que puedan denotar a un objeto específico. Además constituyen elementos renovadores, al igual que el paradigma de saco de características que revolucionan temas tanto de recuperación y clasificación en el mundo del procesamiento de imágenes. Aunque resulta ser el entrenamiento un proceso lento, no se puede obviar la facilidad que brinda este componente de software para los usuarios de Captura y Catalogación de Media, en cuanto a poseer otra opción de búsqueda mucho más automática, que no depende de un previo etiquetado donde las etiquetas introducidas por un usuario denotan a las imágenes, sino de valores propios que son extraídos de forma computacional.

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN

En el presente capítulo se documentan las pruebas, con el objetivo de demostrar la eficacia del algoritmo propuesto para la recuperación. Los resultados obtenidos se demostrarán mediante diferentes tablas y representaciones gráficas.

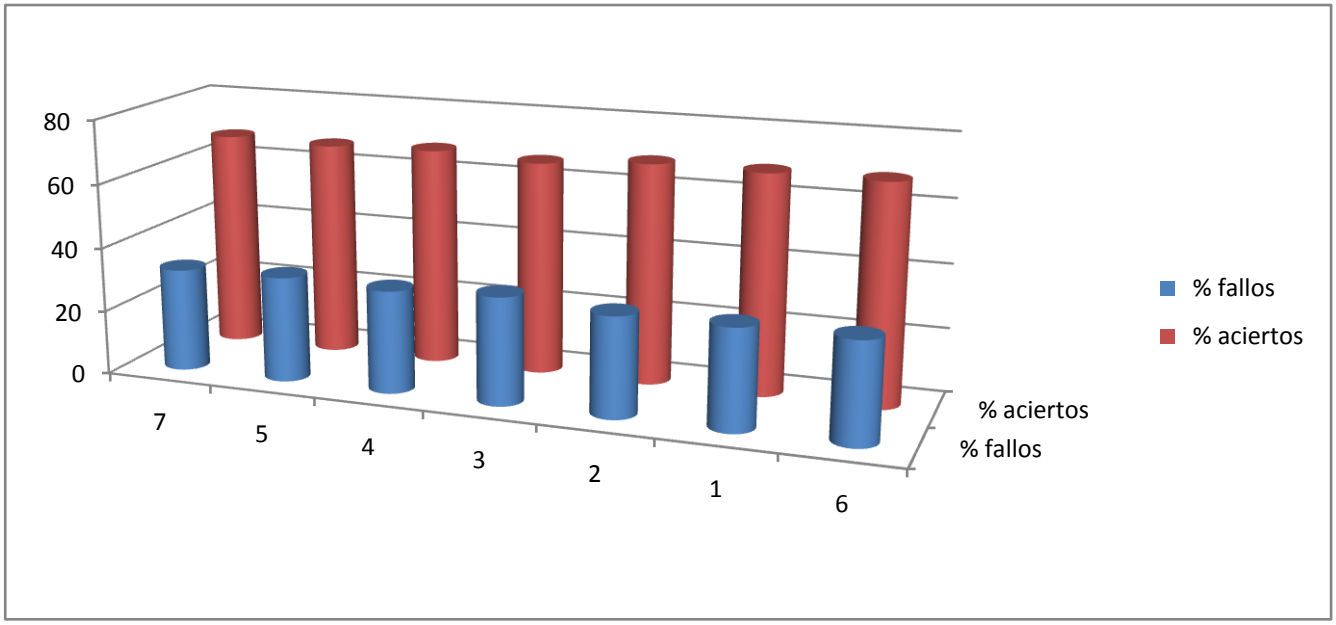
En la realización de las pruebas se utilizó la base de datos de imágenes ETH80, que contiene 3280 imágenes, cuya esencia es representar a un solo objeto en su contenido, intentando eliminar fondos heterogéneos y demasiado diversos, para poder analizar con mayor precisión la capacidad de la solución en detectar puntos invariantes en objetos reales y visibles. Esta base de dato es una de las más sencillas, que contiene 8 clases diferentes de objetos, para cada uno se tomaron 41 vistas de diferentes ángulos. El procedimiento experimental analiza 7 grupos de imágenes escogidas al azar de la base de datos, cada grupo posee 320 imágenes, y serán las que representen al conjunto de entrenamiento, con el propósito de observar el comportamiento de la solución para imágenes diferentes, analizándose si se comportó de manera semejante con diferentes juegos de datos, y poder establecer un porciento global de fallos y aciertos. La recuperación se basó en las 3280 imágenes de la base de datos.

En la siguiente tabla se muestra la cantidad de aciertos y fallos para cada uno de los 7 conjuntos de imágenes escogidas aleatoriamente. Se puede observar, que los resultados fueron semejantes, donde el porciento de fallos oscila entre 31% y 32 % y los aciertos son más del doble de este valor, comprendido entre 67% y 68%. Concluyendo que la cantidad de aciertos para todos los grupos doblan en cantidad a los fallos, demostrándose la eficacia de la solución hasta cierto punto y la estabilidad e independencia de los datos de entrada del algoritmo.

Corrida	Total de aciertos	Total de Fallos	% de Fallos	% de Aciertos
7	2227	1053	32.1	67.9
5	2202	1078	32.86	67.14
4	2231	1049	31.98	68.02
3	2181	1099	33.5	66.5
2	2255	1025	31.25	68.75
1	2248	1032	31.46	68.54
6	2248	1032	31.46	68.54

Tabla 1: Prueba para los siete conjuntos de imágenes aleatorias.

Para una mejor visibilidad de los resultados, la siguiente gráfica trata recoger los resultados de la tabla anterior, mostrando en este caso, solo la cantidad de aciertos contra fallos.



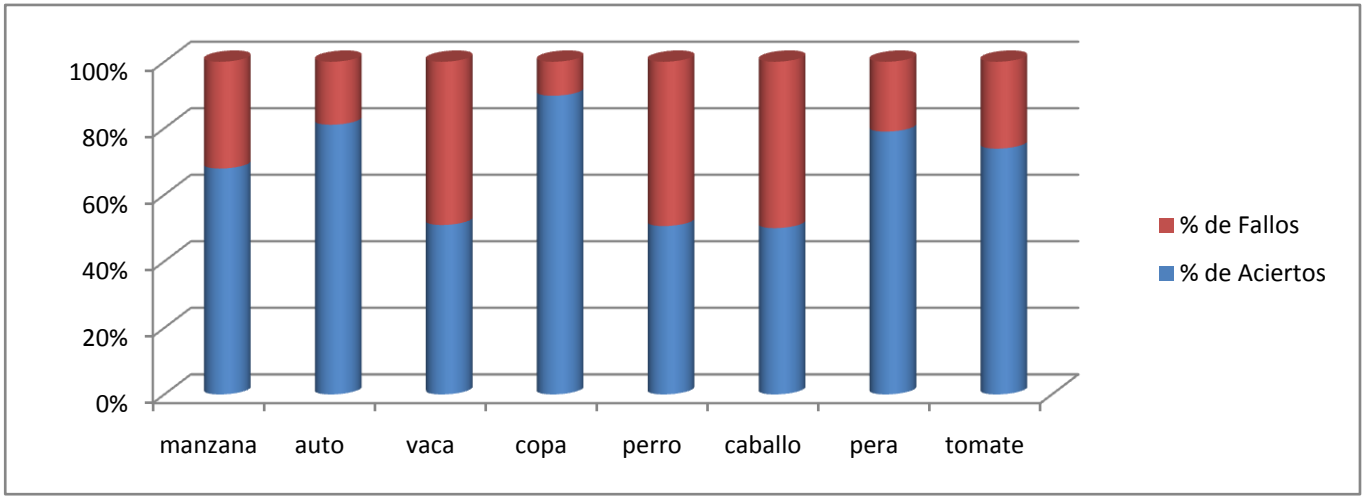
Gráfica 1: Aciertos contra Fallos de los siete conjuntos de imágenes aleatorias.

Analizando con detenimiento los resultados por clases, teniendo en cuenta un promedio total sobre los 7 grupos de imágenes escogidos aleatoriamente, se puede concluir que la clase que mejores resultados arroja es la copa, puesto que sus características son distintivas dentro del conjunto de imágenes de entrenamiento, lo que la hace realzar su diferencia en comparación con las demás clases del conjunto, permitiendo detectarla con mejor facilidad. Lo mismo ocurre con los autos, que aunque denotan resultados menores que las copas, poseen mucho más aciertos que de fallos, posicionándose en segundo lugar en la respuesta del experimento. En cambio no sucede lo mismo con los perros, las vacas y caballos, ya que semánticamente tienen rasgos semejantes que tienden a confundirse, mostrando en muchos casos un animal en lugar de otro, lo que hace aumentar la cantidad de fallos para la recuperación de algunas de ellas. Con las manzanas, las peras y los tomates, los resultados suelen semejarse a los resultados devueltos por la consulta de algún animal, aunque con menor cantidad de fallos, pues sus estructuras son similares y tienden a entremezclarse en las respuestas de una consulta determinada, que toma como referente a estas clases en particular.

En la siguiente tabla se ilustra los resultados anteriormente descritos, cuyos datos están representados en la Gráfica 1, donde se contraponen los aciertos y fallos por clases.

Clases	Promedio de Aciertos	Promedio de Fallos
Manzanas	278.43	131.57
Autos	332.43	77.57
Vaca	208.86	201.14
Copa	368.14	41.86
Perro	207.57	202.43
Caballo	205.00	205.00
Pera	324.00	86.00
Tomate	303.00	107.00

Tabla 2: Promedio de aciertos contra fallos por cada clase del conjunto de entrenamiento.



Gráfica 2: Porciento de aciertos contra fallos por cada clase del conjunto de entrenamiento.

Como conclusión de estos resultados, se evidencia que la recuperación se basa en el contenido de las imágenes, en su estructura semántica, por lo que suele ser impredecible para imágenes que denotan objetos similares, lo que resulta ser una situación propia para una recuperación basada en su semántica. Aunque no son un 100% eficaz, son más precisas que la recuperación que toma como referente a las características visuales como color, iluminación y sombra, que utiliza a los histogramas de estos valores como estructura identificativa y definitoria, a la hora de tomar la decisión de devolver una imagen similar a una de consulta, pues valores similares de estas características pueden identificar objetos muy distintos, lo que demuestra que los descriptores de alto nivel aportan mayor información sobre los objetos y acontecimientos de la escena.

En la siguiente gráfica finalmente se muestra la cantidad de fallos y aciertos totales que se devolvió como resultado las pruebas realizadas. Donde la cantidad de fallos es de un 32% y la cantidad de aciertos de un 68%, asegurando que el algoritmo de recuperación, donde interviene el SEP-COP encargado de buscar el mejor agrupamiento dentro de las características y separar aquellas que pertenecen a clases diferentes, denote eficacia.

Gráfica 3: Porciento total de aciertos contra fallos.

4.1 Conclusiones Parciales

Tras las pruebas realizadas se puede asegurar que la recuperación retornará resultados favorables, que ayuden a los usuarios a buscar contenido específico dado una imagen de consulta. Teniendo en cuenta el artículo (Teynor, y otros, 2008), los resultados del algoritmo propuesto en la investigación haciendo uso del descriptor SURF, no está muy lejos de los resultados allí representados con el descriptor SIFT, que como se había dicho en el capítulo 3 retorna mayor cantidad de puntos de interés invariantes por imágenes, lo que lo hace más robusto y apropiado para temas de recuperación. Obteniéndose como porciento de aciertos un 74.6, que en comparación con el resultado obtenido en la investigación de un 68%, es mucho mejor, aunque se evidencia una buena aproximación a este valor, pudiéndose verificar la eficacia del algoritmo de agrupamiento, del cual depende el resultado de las consultas deseadas.

CONCLUSIONES GENERALES

La realización de la presente investigación permitió desarrollar un componente de software que facilita la búsqueda y recuperación de imágenes, posibilitando ampliar las opciones de búsqueda de los usuarios que utilizan soluciones como las que propone el proyecto, dando cumplimiento al objetivo general planteado. De esta manera se puede arribar a las siguientes conclusiones:

- El estudio de los descriptores desarrollado como parte de la investigación, evidenció que los de alto nivel, por su capacidad de extracción de características invariantes son más afines a una interpretación semántica del contenido de las imágenes.
- La selección de las tecnologías y herramientas para el desarrollo del componente satisfacen las necesidades de selección de tecnologías libres y multiplataforma, teniendo en cuenta las políticas que acogen la universidad y el país.
- La utilización del paradigma de Saco de Características condujo el proceso de desarrollo hacia la implementación de etapas bien definidas que aseguran la recuperación basada en un conjunto reducido de datos.
- La validación de la solución propuesta mostró que el algoritmo SEP-COP implementado retorna resultados aceptables acorde a respuestas actuales, haciendo uso de la misma base de datos de prueba con los métodos de agrupación utilizados comúnmente.

RECOMENDACIONES

Se tiene como recomendaciones para darle continuidad a la investigación:

- Proponer el uso de técnicas de clúster basadas en densidad, combinadas con los métodos jerárquico propuestos para reducir el tamaño del conjunto de datos iníciales a la entrada del método de agrupamiento.
- Proponer la evaluación del uso de asignaciones múltiples en el proceso de recuperación manejando factores de probabilidades de asignación.

BIBLIOGRAFÍA

Alvarez, Sara. 2003. Biblioteca Complutense. *Análisis de usabilidad de sistemas CBIR.* [En línea] 6 de 2 de 2003. [Citado el: 13 de 11 de 2011.] http://www.ucm.es/BUCM/revistas/inf/02104210/articulos/DCIN0303110313A.PDF.

Arias, Rafael. 2009. E-Prints Complutense. *El video en el ciberespacio: uso y lenguaje.* [En línea] 1 de 10 de 2009. [Citado el: 4 de 11 de 2011.] http://eprints.ucm.es/9492/1/OriginalComunicar.pdf.

Boullosa, Oscar. 2011.*Estudio comparativo de descriptores visuales para la detección de escenas cuasi-duplicadas.* Madrid : s.n., 2011.

Bustamante, Paul, y otros. 2004.*Aprenda C++ Básico como si estuviera en primero.* 2004.

Chang, Leandro y Hernández, José.*DETECCIÓN DE PUNTOS DE INTERÉS DEL SIFT USANDO HARDWARE RECONFIGURABLE.* Cuba : s.n.

Cia Ulacia, Ioritz y Pagola Barrio, Miguel. 2010.*IMPLEMENTACIÓN DE UN SISTEMA AUTOMÁTICO DE LOCALIZACIÓN DE REGIONES FACIALES EN VÍDEO POR WEBCAM.* 2010.

Cruz, Angel, y otros.*Sistema para la Recuperación por Contenido en un banco de imágenes médicas.* Bogotá : s.n.

Dominguez, Juan Manuel. 2009.*ESTIMACION DE LA DISTANCIA RECORRIDA POR UN ROBOT MOVIL MEDIANTE LA DESCRIPCION DE DESCRIPTORES SURF.* [En línea] 12 de 2009. [Citado el: 15 de 11 de 2011.] http://e-archivo.uc3m.es/bitstream/10016/8048/2/PFC_JuanMAnuel_Peraza_Dominguez.pdf.

Dorigo, Marco. 1992.*Optimization, learning and natural algorithms.* Italia : s.n., 1992.

G. Lowe, David. 2004.*Distinctive Image Features from Scale-Invariant Keypoints.* Canada : s.n., 2004.

García, Cristina y Gómez, Irene. 2008.*ALGORITMOS DE APRENDIZAJE: KNN & KMEANS.* [En línea] 2008. [Citado el: 19 de 11 de 2011.] http://www.it.uc3m.es/jvillena/irc/practicas/08-09/06.pdf.

García, Guillermo. 2010. Universidad de Valencia. *La eclosión del vídeo como mecanismo de comunicación política en Internet.* [En línea] 2010. [Citado el: 4 de 11 de 2011.] http://www.uv.es/guilopez/aeic/texto.pdf.

Grauman, Kristen y Leibe, Bastian. 2010.*Local Features: Detection and Description.* 2010.

—. 2008.*Local Features: Detection and Description.* 2008.

Gurrutxaga, Ibai, y otros. 2010.*SEP/COP: An efficient method to find the best partition in hierarchical clustering based on a new cluster validity index.* 2010.

Han, Zenjun, Ye, Qixiang y Jiao, Jianbin. 2011.*Combined feature evaluation for adaptive visual object tracking.* China : ELSEVIER, 2011.

Huang, Jing, y otros.*Image Indexing Using Color Correlograms.* New York : s.n. 14853.

Ingelmo, Alejandro Mateo. 2009.*Estudio de Técnicas para la caracterización de la Figura Humana, para su Posible Aplicación a Problemas de Reconocimientos de Género.* [En línea] 25 de 5 de 2009. [Citado el: 14 de 11 de 2011.] http://www.vision.uji.es/~montoliu/docs/pfc/AlejandroMateoIngelmo.pdf.

La Serna Palomino, Nora, Contreras, Walter y Ruiz, María Elena. 2010.*Procesamiento Digital de textura: Técnicas utilizadas en aplicaciones actuales de CBIR.* 2010.

La Serna, Nora y Alvarado, Luis. 2010.*Recuperación de Imagenes basados en Contenidos (CBIR): Técnicas de Representación visual actuales y Aplicaciones.* 2010.

Marcos Recio, Juan Carlos. 2007.*La fotografía en la publicidad: archivos, bancos de imágenes y centros de documentación en el siglo XXIi.* 2007.

Martínez Mejia, David Alberto. 2005.*INTERCEPTOR DE TRAYECTORIAS BASADO EN VISIÓN ARTIFICIAL.* 2005.

Martínez Muñoz, León Alberto. 2011.*Implementación de un editor gráfico de circuitos eléctricos con Qt.* 2011.

Mikolajczyk, K. 2005.*A Comparison of Affine Region Detectors.* 2005.

—. 1999.*Filtering for texture classification: a comparative study.* 1999.

Moreno, Jose Guillermo. 2008.*Recuperación de Imágenes: Estado del Arte.* Colombia : s.n., 2008.

Muños, Natalia. 2010.*Sistema de clasificación automática de imágenes médicas basado en contenido.* Madrid : s.n., 2010.

Ordoñez Santiago, Cristian Andrés. 2005.*Formatos de Imagen Digital.* 2005.

Pérez, J., y otros. 2007.*Mejora al algoritmo de agrupamiento K-means mediante un nuevo criterio de Convergencia y su Aplicación a Base de Datos Poblacionales de Cáncer.* [En línea] 10 de 2007. [Citado el: 15 de 11 de 2011.] http://www.tlaio.org.mx/DOCS/T2_5_A27iJPO.pdf.

Pons Calvo, Carlos. 2008.*GRUPO VISIÓN POR COMPUTADOR.* 2008.

Robles Sánchez, Oscar David. 2004.*Técnicas de Recuperación por Contenido para Imagen y Video en Arquitecturas Paralelas.* Madrid : s.n., 2004.

Rodriguez, Jose Luis. 2007. dZoom. [En línea] 14 de 11 de 2007. [Citado el: 03 de 11 de 2011.] http://www.dzoom.org.es/noticia-1708.html.

Romero, A. M. y Cazorla, M. 2009.*Comparativa de detectores de características visuales y su aplicación SLAM.* 2009.

Rui, Yong y S. Huang, Thomas. 1999.*Image Retrieval: Current Techniques, Promising Directions, and Open Issues.* Urbana : s.n., 1999.

Stroustrup, Bejarne. 1985.*El lenguaje de programación C++.* 1985.

T. Freeman, Willian y Roth, Michal. 1994.*Orientation Histograms for Hand Gesture.* [En línea] 12 de 1994. [Citado el: 14 de 11 de 2011.] http://www.google.com.cu/url?sa=t&rct=j&q=%2BOrientation%2Bhistograms%2Bfor%2Bhand%2Bgesture%2Brecognition&source=web&cd=1&ved=0CCMQFjAA&url=http%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2Fdownload%3Fdoi%3D10.1.1.165.1575%26rep%3Drep1%26type%3Dpdf&ei=uHvBT.

Teynor, Alexandra, y otros. 2008.*Properties of Patch Based Approaches for the Recognition of Visual Object Classes.* 2008.

Triggs, Navneet Dalal and Bill. 2005.*Histograms of Oriented Gradients for Human Detection.* [En línea] 2005. [Citado el: 14 de 11 de 2011.] http://www.acemedia.org/aceMedia/files/document/wp7/2005/cvpr05-inria.pdf.

Van de Sande, Koen, Gevers, Theo y G. M. Snoek, Cees. 2010.*Evaluating Color Descriptors for Objct and Scene Recognition.* [En línea] 9 de 9 de 2010. [Citado el: 29 de 11 de 2011.] http://staff.science.uva.nl/~gevers/pub/GeversPAMI10.pdf.

Vapnik, V. 2000.*The Nature of Statistical Learning Theory.* New York : s.n., 2000.

Via2 Platform. 2004.*Las necesidades públicas del software.* [En línea] 3 de 2004. [Citado el: 13 de 11 de 2011.] http://www.socinfo.info/contenidos/pdf2/software2.PDF.

Villamizar, Michael, y otros. 2009.*Combining Color-Based Invariant Gradient Detector with HoG Descriptors For Robust Image Detection in Scenes under Cast Shadows.* [En línea] 12 de 5 de 2009. [Citado el: 28 de 11 de 2011.] http://digital.csic.es/bitstream/10261/30099/1/doc1.pdf.

Visual Century. 2003.*Via2 Platform: La nueva manera de gestionar archivos multimedia.* [En línea] 2003. [Citado el: 13 de 11 de 2011.] http://www.pdi.es/documents/Via2Plattform.pdf.

Yoo, Hun-Woo, y otros. 2002.*Visual Information Retrieval System via Content-based Approach.* 2002.

Yuste Cortés, Silvia. 2009.*Interfaz Gráfica de Usuario para la Búsqueda de Imágenes basada en Imágenes.* [En línea] 6 de 2009. [Citado el: 13 de 11 de 2011.] http://upcommons.upc.edu/pfc/bitstream/2099.1/8588/1/memoria.pdf.

Zhao, Ying y Karypis, George. 2002.*Evaluation of Hierarchical Clustering Algorithms for Document Datasets.* [En línea] 3 de 6 de 2002. [Citado el: 20 de 11 de 2011.] http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA439551&Location=U2&doc=GetTRDoc.pdf.

ANEXOS

Anexo 1

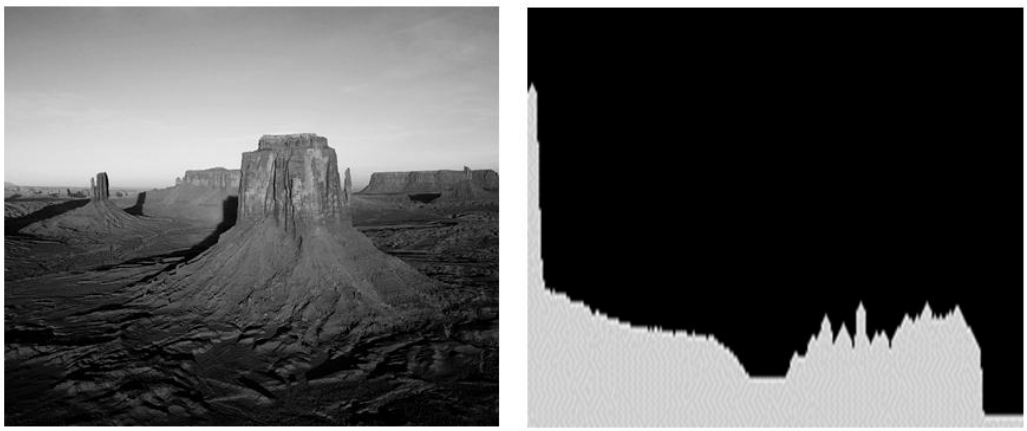


Figura 4: a) Imagen, b) Histograma de la imagen.

Anexo 2

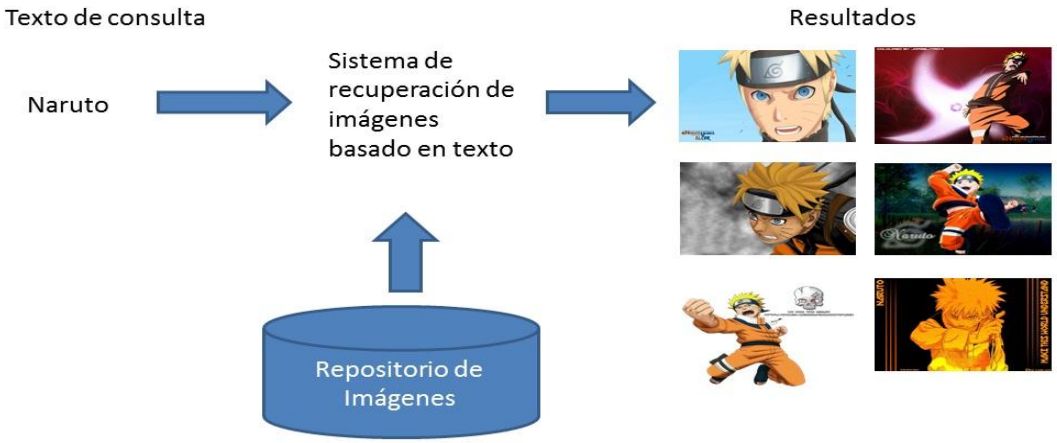


Figura 5: Sistema basado en texto.

Anexo 3

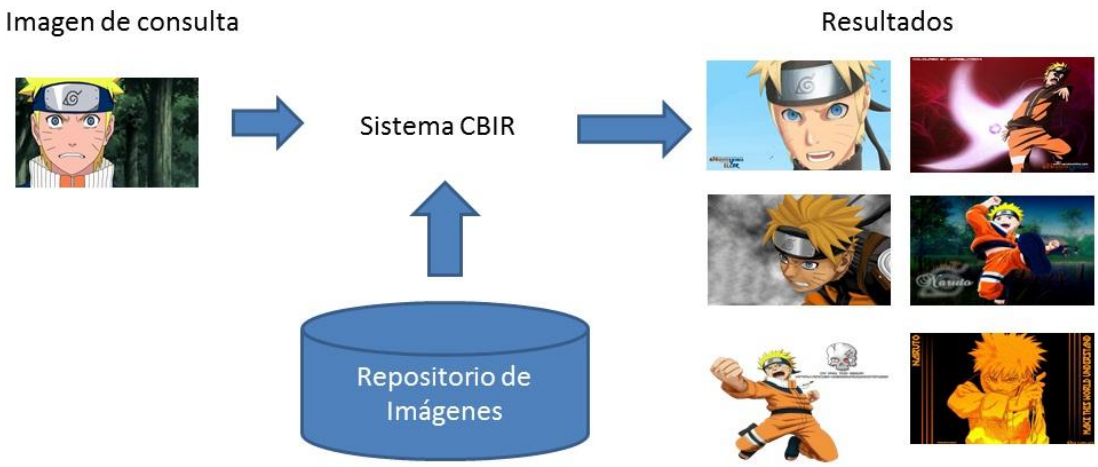


Figura 6: Sistema basado en contenido.

Anexo 4

Figura 7: Sistema basado en búsqueda semántica.

Anexo 5

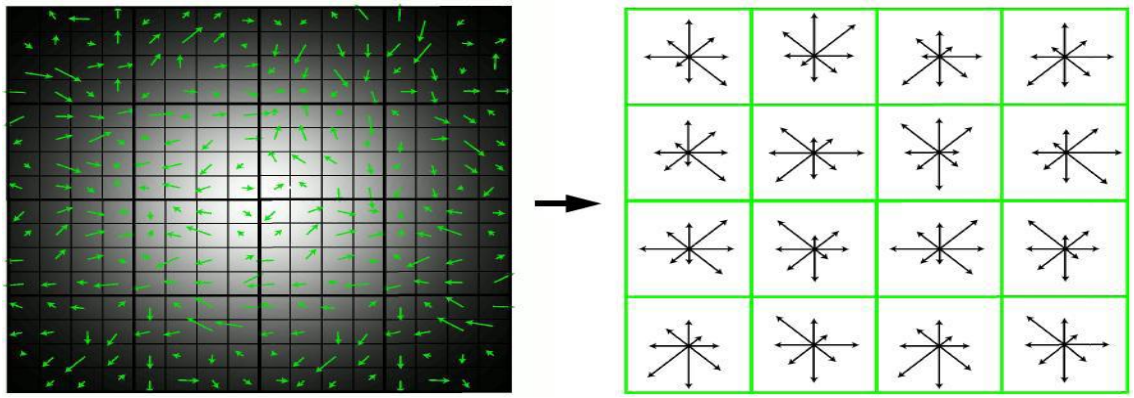


Figura 8: Puntos de Interés

Anexo 6

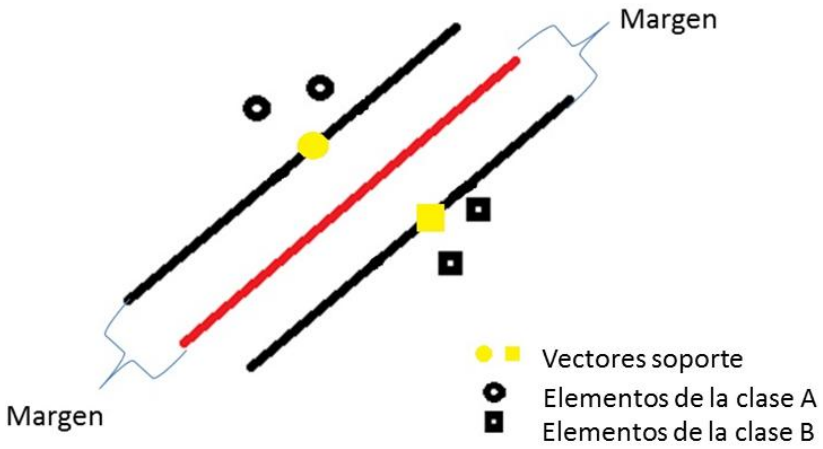


Figura 9: Funcionamiento de la Máquina Soporte Vectorial.