



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO
ESCOM

Trabajo Terminal

“Motor de búsqueda local mediante imágenes”

15-2 -006

Presenta

Martínez Rosas Omar

Directores

Dr. Juan Carlos Chimal Eguía

M. en C. Reyna Elia Melara Abarca



México, D.F., a 3 de Julio de 2015



INSTITUTO POLITÉCNICO NACIONAL

Escuela Superior de Cómputo



No. Registro: 15 – 2 – 006

Julio del 2015

Documento Técnico

Motor de búsqueda local mediante imágenes

Autores:

Martínez Rosas Omar¹

RESUMEN

En este reporte se muestra la documentación técnica del Trabajo Terminal Remedial 2015 – A006, que lleva por nombre «Motor de búsqueda local mediante imágenes», cuyo objetivo es crear un motor de búsqueda de información en el disco local de una computadora a través de la entrada de una imagen; la información puede consistir en más imágenes u otro tipo de documentos que contengan imágenes similares a la entrada, desplegando visualmente de los resultados obtenidos ordenados por afinidad de mayor a menor similitud.

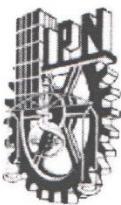
Palabras clave: Procesamiento y análisis de imágenes, Búsqueda, Archivos, Algoritmia, Consulta de imágenes mediante ejemplo (*CBIR*), Perceptual Hash, Hash, Block Mean.

Directores

Dr. Juan Carlos Chimal Eguía

M. en C. Reyna Elia Melara Abarca

¹*omack47@gmail.com*



ESCUELA SUPERIOR DE CÓMPUTO
SUBDIRECCIÓN ACADÉMICA

DEPARTAMENTO DE FORMACIÓN INTEGRAL
E INSTITUCIONAL

COMISIÓN ACADÉMICA DE TRABAJO TERMINAL



México, D.F. a 10 de Junio de 2015

DR. FLAVIO ARTURO SÁNCHEZ GARFIAS
PRESIDENTE DE LA COMISIÓN ACADÉMICA
DE TRABAJO TERMINAL
PRESENTE

Por medio del presente, se informa que la alumna **Omar Martínez Rosas**, que integra el **TRABAJO TERMINAL 15 - 2 - 006**, titulado "**Motor de búsqueda local mediante imagen**" concluyó satisfactoriamente su trabajo.

Los discos (DVDs) fueron revisados ampliamente por su servidor y corregidos, cubriendo el alcance y el objetivo planteados en el protocolo original y de acuerdo a los requisitos establecidos por la Comisión que Usted preside.

ATENTAMENTE

M. en C. Reyna Elia Melara Abarca

Advertencia

«Este documento contiene información desarrollada por la Escuela Superior de Cómputo del Instituto Politécnico Nacional, a partir de datos y documentos con derecho de propiedad y por lo tanto, su uso quedará restringido a las aplicaciones que explícitamente se convengan.»

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

Información adicional sobre este reporte técnico podrá obtenerse en:

La Subdirección Académica de la Escuela Superior de Cómputo del Instituto Politécnico Nacional, situada en Av. Juan de Dios Bátiz s/n Teléfono: 57296000 Extensión 52000.

Agradecimientos

*“A mi padre Ángel Martínez,
A mi madre María Evangelina Rosas,
A mi hermana Lorena Desireé,
A mi pareja Gaby Roque,
A mis amigos,
A mis mentores,
que con su tiempo y dedicación
han hecho posible cumplir mis metas una vez más.
Seguiremos este camino hasta el final, ya que esto
es solo el comienzo de una maravillosa historia...”*



Tabla de contenidos

CAPÍTULO 1 INTRODUCCIÓN	9
1.1 Preámbulo.....	9
1.2 Planteamiento del problema	9
1.3 Objetivos.....	10
1.3.1 Objetivo General.....	10
1.3.2 Objetivos Específicos	10
1.4 Justificación.....	10
1.5 Organización del documento	11
CAPÍTULO 2 ANTECEDENTES	12
2.1 MARCO TEÓRICO.....	12
2.1.1 Comparación Elástica	12
2.1.2 Perceptual Hashing	13
2.1.3 Funciones Hash.....	14
2.1.4 Tipos de funciones perceptual hash	17
2.1.5 Función Perceptual Hash TRANSFORMADA DISCRETA COSENO.....	18
2.1.6 Función Perceptual Hash BLOCK MEAN.....	20
2.1.7 Funciones de similitud para funciones perceptuales.....	21
2.2 ESTADO DEL ARTE	24
CAPÍTULO 3 ANÁLISIS	25
3.1 PROPUESTA DE SOLUCIÓN	25
3.2 CARACTERÍSTICAS DEL SISTEMA.....	25
3.3 ESTUDIO DE FACTIBILIDAD	26
3.3.1 Factibilidad técnica	26
3.4 HISTORIAS DE USUARIO.....	28
3.4.1 HU1: Primera ejecución del sistema.....	29
3.4.2 HU2: Generación de firmas de archivos indexados.....	30
3.4.3 HU3: Carga de imagen a buscar	31
3.4.4 HU4: Buscar imagen.....	32
3.4.5 HU5: Configurar búsqueda	33
3.5 GESTIÓN DE RIESGOS.....	34
3.5.1 Evaluación inicial de riesgos	34
3.5.2 Análisis de riesgos	35
3.5.3 Gestión de riesgos: Estrategias de control	36

3.6 ESTIMACIÓN DE COSTOS.....	37
CAPÍTULO 4 DISEÑO.....	38
4.1 Diagrama a bloques.....	38
4.2 Diagrama a bloques.....	39
4.3 Diagrama de clases.....	40
4.4 Diagramas de secuencia	43
4.5 Diseño de pantallas	47
4.5.1 Pantalla HU1: Configuración Primera Ejecución	47
4.5.2 Pantalla HU2: Cargar imagen	49
4.5.3 Pantalla HU3: Buscar imágenes similares	50
4.5.4 Pantalla HU4: Configuración de búsqueda.....	51
CAPÍTULO 5 DESARROLLO	52
5.1 PANTALLA DE INICIO	52
5.1.1 Indexación de archivos	53
5.1.2 Procesamiento de imágenes (generación firmas).....	54
5.1.3 Generación de una firma por medio de la función DCT Perceptual Hash.....	54
5.1.4 Generación de una firma por medio de la función Block Mean	56
5.2 PANTALLA PRINCIPAL DE BÚSQUEDA	58
5.2.1 Procesamiento de imágenes	58
5.2.2 Cálculo de similitud entre firmas.....	59
CAPÍTULO 6 PRUEBAS Y RESULTADOS.....	60
6.1 RESULTADOS OBTENIDOS	60
6.2 PRUEBAS DE RENDIMIENTO	72
CAPÍTULO 7 CONCLUSIONES	81
CAPÍTULO 8 TRABAJO A FUTURO	82
Índice Tablas	83
Índice Ilustraciones.....	84
Índice Graficas.....	85
GLOSARIO.....	86
BIBLIOGRAFÍA.....	87

CAPÍTULO 1

INTRODUCCIÓN

1.1 Preámbulo

Los sistemas de consulta de imágenes surgieron para solucionar los problemas de la indexación manual, que consisten en la creación de una estructura de almacenamiento manual a grandes conjuntos de imágenes especificando términos descriptivos para cada imagen. Estos problemas eran, y siguen siendo, una gran inversión en tiempo debido a la inconsistencia entre indicadores humanos, así como la dificultad de expresar mediante palabras las cualidades gráficas y las sensaciones estéticas que proporciona la percepción de una representación visual.

Los motores de búsqueda mediante imágenes dirigen su atención a las características intrínsecas de las imágenes como el color, texturas, metadatos (como nombre, características, descripción de imagen entre otros) para realizar búsquedas automatizadas las cuales despliegan en forma visible los resultados obtenidos; estos han sido y siguen siendo durante largo tiempo un área de investigación floreciente, donde cada año, aparecen nuevos documentos en foros científicos y técnicos que proponen nuevas soluciones a este problema.

1.2 Planteamiento del problema

Encontrar imágenes similares a una imagen específica es una tarea que si se realiza de forma manual en un conjunto de archivos que es muy grande o continuamente creciente, tiende a ser una labor imposible debido a que pueden existir factores humanos que lleven a limitaciones en tiempo donde una sola búsqueda podría tardar semanas a meses e incluso años si no se tiene una herramienta que facilite la búsqueda en archivos locales tomando las características visuales perceptibles de la imagen, independientes a las características descriptivas de los archivos de imágenes.

Es por ello que se propone una solución de un motor de búsqueda local mediante imágenes que facilite las búsquedas de imágenes, mostrando no solo imágenes que sean exactamente iguales si no también extendiendo la búsqueda a imágenes similares.

1.3 Objetivos

1.3.1 Objetivo General

- Implementar un sistema que busque imágenes similares a una imagen objetivo en un entorno local.

1.3.2 Objetivos Específicos

- Implementar un sistema de indexación de archivos.
- Implementar algoritmos de comparación de imágenes.
- Implementar algoritmo para búsqueda de imágenes similares.
- Extender la búsqueda de similitudes a imágenes contenidas en archivos de documentos.

1.4 Justificación

Los sistemas de Consulta de imágenes mediante ejemplo (*CBIR* por sus siglas en inglés) surgieron a comienzos de la década de los noventa para solucionar los problemas de indexación manual en imágenes, la cual consiste en la asociación de términos descriptivos a las imágenes.

Hoy en día existen herramientas que cumplen con la tarea de realizar búsquedas mediante imágenes, pero el problema principal de estas herramientas radica en el hecho de que están diseñadas para hacer una búsqueda en internet o en portales web específicos y no en archivos locales de un usuario, por ende los usuarios que trabajen imágenes no cuentan con una herramienta que apoye para indexar las grandes cantidades de imágenes que tengan almacenadas en sus archivos locales.

El desarrollar una herramienta *CBIR* local reducirá el tiempo de indexación de grandes cantidades de imágenes que están almacenadas de forma local contra los métodos de indexación manual, además de extender la búsqueda a imágenes contenidas en documentos, encargándose de realizar la indexación automática de los archivos para facilitar la búsqueda posterior de una imagen objetivo.

Originalidad del trabajo: Realiza la búsqueda de imágenes similares en un entorno local, extendiendo la búsqueda a imágenes contenidas en documentos.

Usuarios potenciales: Cualquier persona interesada en buscar información almacenada en archivos locales, a través de imágenes, por ejemplo: diseñadores gráficos, criminólogos, arquitectos, médicos entre otros.

1.5 Organización del documento

En el **Capítulo 1 Introducción**, se incluye una breve introducción acerca de la temática en general que se abordará a lo largo del presente documento, de igual manera se describen el planteamiento del problema a solucionar, la justificación, los antecedentes sobre el tema, el objetivo general y particulares de este trabajo de investigación.

En el **Capítulo 2 Antecedentes**, se presenta el estado del arte, donde se describen aquellos trabajos relacionados con sistemas que lleven a cabo consulta de imágenes mediante ejemplo. También se muestra el marco teórico que contiene una descripción básica de los temas que trata este trabajo, se incluyen también definiciones y la base teórica de los algoritmos y métodos de validación que se implementarán en el sistema.

En el **Capítulo 3 Análisis**, se describe el análisis sistema, siguiendo la metodología XP, elegida para este desarrollo.

En el **Capítulo 4 Diseño**, se muestra el diseño del sistema, las pantallas y la descripción de las mismas.

En el **Capítulo 5 Desarrollo**, se presentan los algoritmos utilizados en el sistema, especificando dónde y cómo se implementan.

En el **Capítulo 6 Pruebas y Resultados**, se pueden ver las pruebas realizadas al sistema y los resultados de las mismas.

En el **Capítulo 7 Conclusiones**, se detallan las conclusiones a las que se llegaron al realizar el desarrollo de este trabajo.

En el **Capítulo 8 Trabajo a Futuro**, se describe el trabajo a futuro y las posibles mejoras al sistema.

Para finalizar, se anexan el índice de tablas, el índice de ilustraciones, el índice de gráficas, el glosario y la bibliografía consultada para la realización de este trabajo

CAPÍTULO 2

ANTECEDENTES

2.1 MARCO TEÓRICO

Para entender el comportamiento del motor de búsqueda local mediante imágenes es necesario conocer algunos conceptos que se manejan en el presente documento, estos conceptos se explican a continuación.

2.1.1 Comparación Elástica

La comparación Elástica es una de las técnicas de reconocimiento de patrones en la informática, enfocado al análisis de la comparación de un acontecimiento, suceso o cualidad que pueda ser representado en un arreglo n-dimensional mayormente conocido como **fenómeno** contra sus similares, por medio de una comparación mediante una función dependiente de la **deformación** la cual puede describir si estos fenómenos comparados son similares o iguales.

Para el enfoque de este proyecto realizado se toma la definición de fenómeno aterrizado a una imagen la cual está definida como una función de intensidad bidimensional, que en la literatura se representa indistintamente dependiendo el contexto y nomenclatura como una función $f(x,y)$ proporcional a la intensidad de luminosidad en un punto dado, así pudiendo ser comparada contra otras imágenes.

La comparación elástica de dos imágenes es un problema NP-Completo (Keysersa, 2002), Los resultados de estudios anteriores en el tema como *Elastic image matching, 2D-warping* (Uchida, 1998) u otros tipos similares de métodos invariantes (Keysers, 2000) describen un problema intrínsecamente difícil, es decir, que para que el problema pueda ser resuelto en tiempo polinomial se proponen algunas técnicas de comparación elástica basado en algoritmos más rápidos, soluciones sub-óptimas como reconocimiento de patrones, restricciones a optimizaciones locales y aproximaciones heurísticas (*Simulated annealing, Genetic Algorithms*).



La comparación de dos imágenes es un problema NP-Completo, es decir que el problema no puede ser resuelto en tiempo polinomial.

No obstante paralelo a este tiempo de evolución en la digitalización de archivos multimedia, han surgido nuevos métodos de autenticación de contenido, los cuales buscan certificar la genuinidad de contenido en cada archivo.

Para ello comúnmente se utilizan elementos criptográficos llamados “**funciones hash**” las cuales son adecuadas para autenticar contenido haciendo una comparación rígida el cual compara bit por bit entre cada archivo, sin embargo en una imagen emergen diferentes representaciones que al presentar un cambio; esto debido a que tras un procesamiento de una imagen como compresión, recorte de imagen, cambio de dimensiones, o ecualización de histograma, para el humano la percepción de esta imagen es la misma, más sin embargo la representación binaria es diferente, y por lo tanto el uso de **funciones hash** basadas en la autenticación como un método de comparación entre imágenes falla.

2.1.2 Perceptual Hashing

Perceptual Hashing es una metodología enfocada a la comparación de similitud de imágenes basada en la simulación de la **percepción** del humano, es decir la capacidad de como el humano puede recibir, interpretar y comprender mediante sus sentidos la representación visual de cada imagen.

Esta metodología establece una “**similitud perceptual**” entre imágenes por una función dependiente de la deformación que compara **firmas** de imágenes generadas a través de un método de reducción de información de características a su mínima representación; de tal forma que en su firma se expresen las características visuales más importantes de una imagen.

Esta metodología sigue siendo un área de continua investigación y creciente interés en el ámbito científico, en la cual se han podido encontrar aún más aplicaciones de comparación para otros tipos de fenómenos como archivos de audio y video.



La metodología **Perceptual Hashing** propone establecer comparación similitud e igualdad entre imágenes, pudiéndose escalar la misma funcionalidad para diferentes tipos de fenómenos como archivos de audio y video.

Para poder realizar la comparación de similitud entre dos imágenes, **Perceptual Hashing** hace uso de un método de reducción de información conocido como **Función Perceptual Hash**, la cual genera una firma de una longitud reducida contra la representación original del archivo. Una vez calculada la firma del archivo esta puede ser comparada contra otra por medio de una función dependiente de la distorsión conocida como **Función de distancia**, el resultado obtenido determina la similitud entre estas imágenes.



Las funciones **perceptual hash** reducen la información de la imagen a una firma.

Las funciones de **distancia** determinan la similitud de dos imágenes basado en la distancia obtenida entre sus firmas.

La descripción formal para una **Función Perceptual Hash** única aún no han sido determinada debido a que se han encontrado múltiples factores a considerar durante la reducción de información, debido a que cada método resuelve un problema de representación de una imagen en concreto como la rotación, traslación cambio de color, compresión de datos entre otros. Es por ello que se consideran a las *funciones perceptual hash* un campo interdisciplinario de investigación, donde avances en la criptografía, procesamiento digital de señales , teoría de comunicaciones entre otros, forman parte del campo de desarrollo en esta área, derivando en la inexistencia de una nomenclatura o una definición aun totalmente aceptada de una *función perceptual hash*.

2.1.3 Funciones Hash

Partiendo de que no existe aun una definición totalmente aceptada de una *función perceptual hash*, se muestra la definición de una función hash, la cual en su definición base es transportada a su uso en la metodología perceptual hashing.

Una función Hash H proyecta un valor α de un conjunto con muchos (o incluso un número infinito) de miembros a un valor β de un conjunto con un número fijo de (menos) miembros, como limitante que las funciones hash no son reversibles, es decir no se puede obtener el valor original (α) a partir de β .

A diferencia de **Perceptual Hashing**, Las funciones hash en la criptografía se utilizan para determinar si dos objetos son iguales (posiblemente con un número medio fijo de errores).

En la Ilustración 1 se puede ver la reducción de elementos de una imagen tomada como el valor α con N miembros (mapa de bits) a una firma acotada β de L miembros.

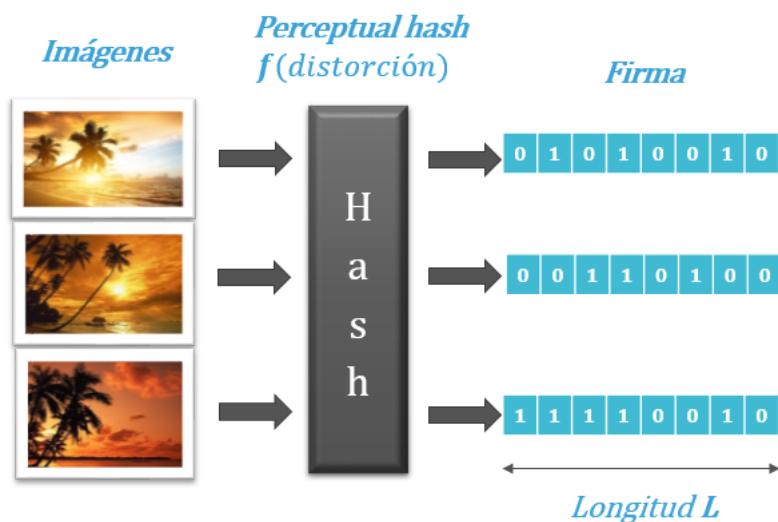


Ilustración 1 Función Perceptual Hash

2.1.3.1 Propiedades de una función perceptual hash

Una función perceptual hash utiliza los términos mostrados en Tabla 1 necesarios para definir sus propiedades.

Termino	Definición
Modificación	<i>Operación que no altera el contenido esencial de una imagen, después de una modificación, la imagen espera ser detectada perceptualmente como similar</i>
Manipulación	<i>Operación que altera el contenido esencial de una imagen, después de una modificación, la imagen espera ser detectada perceptualmente como no similar.</i>

Tabla 1 Conceptos función perceptual hash

De acuerdo con (Meixner, 2006) una función perceptual hash debe contener cuatro propiedades, las cuales son formuladas a partir de los siguientes párrafos:

“Sea P la denotación de probabilidad, sea H una función hash la cual toma como parámetro un archivo (una imagen como ejemplo), y produce una cadena binaria de longitud l , sea x otro archivo (imagen) particular donde \hat{x} denota una versión modificada del objeto original, el cual es perceptualmente similar a x .

Sea y un objeto “perceptualmente diferente” de x ; haciendo a α y β las cadenas binarias (hash) también conocidas como firmas resultantes de los objetos x y y sucesivamente representadas por $\{0/1\}^l$ donde l es la longitud del resultado hash”.

1. Igualdad de distribución (imprevisibilidad) de valores hash:

$$P(H(x) = \alpha) \approx \frac{1}{2^l}, \forall \alpha \in \{0/1\}^l \quad (1)$$

2. Independencia para objetos multimedia x y y “perceptualmente diferentes”:

$$P(H(x) = \alpha | H(y) = \beta) \approx P(H(x) = \alpha), \forall \alpha, \beta \in \{0/1\}^l \quad (2)$$

3. Invariancia de imágenes “perceptualmente iguales” x y \hat{x} :

$$P(H(x) = H(\hat{x})) \approx 1 \quad (3)$$

La mayoría de funciones hash perceptuales tratan de obtener las características de objetos multimedia más relevantes bajo modificaciones insignificantes a nivel global (ej. como compresión y recortes de la misma).

4. Distinción de objetos multimedia perceptualmente diferentes x y y :

$$P(H(x) = H(y)) \approx 0 \quad (4)$$

Dado un objeto multimedia x , es prácticamente imposible construir un objeto diferente y tal que sus resultados α, β sean iguales, ya que aunque las características utilizadas para la generación de una firma son públicas, estas son extremadamente difíciles de imitar.

Es muy importante tener en cuenta es que para cualquier objeto multimedia x sea prácticamente imposible tener un objeto multimedia \hat{x} perceptualmente similar tal que $H(x) \neq H(\hat{x})$, además de que una construcción de un objeto y perceptualmente diferente tal que su firma $H(x) \approx \alpha$ siendo $H(y) = \alpha$.

2.1.3.2 Limitaciones

Una limitación que surge al utilizar las funciones hash perceptuales, es que los objetos multimedia muy similares no pueden ser separados de manera precisa de los no similares;

Un ejemplo de este problema en imágenes es cuando las imágenes que contengan pequeños cambios que sean importantes para su significado semántico, sus resultados de comparación sean afectadas severamente con un cambio de compresión, rotación o color se encuentren en imágenes similares pero estas sean reconocidas como no similares.



Ilustración 2 Imágenes semánticamente detectadas como no similares.

En este caso las imágenes mostradas en la Ilustración 2 no se relacionan con la original dado que el resultado de similitud no es cercano debido a la perdida de información en la compresión utilizada por el formato JPEG (*Joint Photographic Experts Group*), cuestión que da pie a la definición mostrada en Tabla 2 de los valores propuestos por (Schneider, 1996).

Similitud	Definición	Medida Similitud (distorsión)
Exactamente similar	<i>Una imagen que es bit a bit idéntica a la imagen original.</i>	1.0
No similar	<i>Una imagen que no tiene nada en común con la imagen original</i>	0.0
Parcialmente similar	<i>Es un concepto y medición vagamente definido ya que relacionar la similitud es subjetivo, debido a que los cambios de su dominio varían según la aplicación.</i>	$[d(\alpha) > 0, d(\alpha) < 1]$ *donde $d(\alpha)$ es la función distancia definida más adelante.

Tabla 2 Similitud entre imágenes

En la Ilustración 3 de (Zauner, 2010) se puede observar la tolerancia de cambios en una imagen dentro un umbral donde un valor sobre T es considerado como similar, de otra forma la imagen baja su similitud y llega a ser considerada como completamente diferente, por ello existen diferentes tipos de *funciones perceptual hash* que atacan diferentes problemas conocidos como “*ataques*” o modificaciones en una imagen.

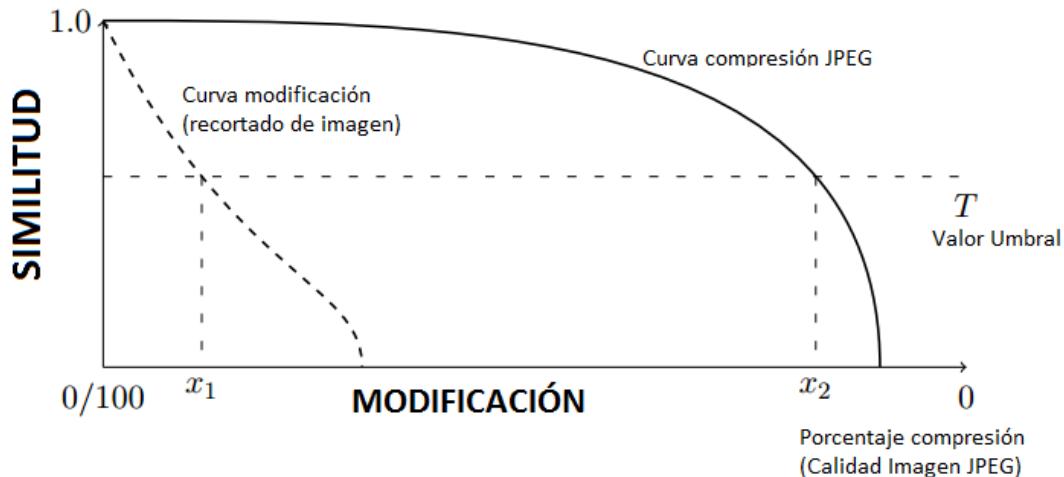


Ilustración 3 Representación de variación en similitud.

2.1.4 Tipos de funciones perceptual hash

Cada *función perceptual hash* se enfoca en la reducción de distintos tipos de información perceptual de una imagen, como se muestra en Tabla 3.

Función perceptual hash	Utilidad
Transformada discreta coseno (DCT)	Utilizada en comparación de imágenes con cambios de compresión, modificaciones pequeñas a la estructura semántica del objeto, variaciones de color.
Momentos Invariantes de Hú	Utilizada en comparación de imágenes con cambios de rotación, escala, modificaciones pequeñas en localización de regiones u objetos dentro de una imagen.
Marr-Hildreth	Basado en detección de bordes como extracción de características de una imagen dando paso a un análisis más detallado en estudio de similitud en objetos internos en una imagen.
Block Mean	En 2006 se propuso por (Yang, 2006) la división de una imagen en secciones definidas que promedian el valor de una imagen combinando diferentes técnicas para atacar la rotación y cambios perceptualmente similares en imágenes.

Tabla 3 Tipos de funciones perceptual hash.

Las *funciones perceptual hash* seleccionadas para el motor de búsqueda local mediante imágenes son la DCT y Block Mean debido a su rapidez de implementación y la eficacia en sus resultados seg\xfcreo (Zauner, 2010).

2.1.5 Función Perceptual Hash TRANSFORMADA DISCRETA COSENO

La transformada discreta del coseno (DCT), también denominada transformada del coseno; es la más ampliamente utilizada en compresión de imágenes, la DCT expresa una secuencia finita de varios puntos como resultado de la suma de distintas señales cosenoidales en frecuencias.

Esta transformación se usa para representar una función empleando los componentes espectrales más representativos de tal forma que la señal reconstruida aún tenga semejanza con la señal original, cabe mencionar que está se encuentra bastante relacionada con la Transformada discreta de Fourier o DFT, con la diferencia de que es una transformada real, debido a que los vectores base se componen exclusivamente de funciones coseno.

 *Esta transformación se usa para representar una función empleando las componentes espectrales más representativas de tal forma que la señal reconstruida aún tenga semejanza con la señal original..*

 *Está relacionada con la DFT con la diferencia que es una transformada real, es decir que solo contiene exclusivamente valores de funciones coseno.*

2.1.5.1 Definición formal

Formalmente, la transformada de coseno discreta es una función lineal invertible $f: \mathbb{R}^N \rightarrow \mathbb{R}^N$, (donde \mathbb{R} denota el conjunto de los números reales), o en forma equivalente a una matriz cuadrada de $N \times N$.

Las variantes más usadas son la DCT-I y la DCT-II, en este caso nos enfocamos en definir la Función DCT-II.

La DCT de dos dimensiones de una matriz A M-por-N se define como sigue.

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\left(\frac{\pi(2m+1)p}{2M}\right) \left(\frac{\pi(2n+1)q}{2N}\right), \quad \begin{matrix} 0 \leq p \leq M-1 \\ 0 \leq q \leq N-1 \end{matrix} \quad (5)$$

Donde:

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p = 0 \\ \sqrt{2/M}, & 1 \leq p \leq M-1 \end{cases} \quad \alpha_q = \begin{cases} 1/\sqrt{N}, & q = 0 \\ \sqrt{2/N}, & 1 \leq q \leq N-1 \end{cases} \quad (6)$$

Los valores B_{pq} se llaman los coeficientes DCT de la matriz A.

Los coeficientes DCT B_{pq} , pueden considerarse como los pesos de base aplicados a la función de la ecuación 5. Para matrices de 8 por 8, las funciones base son 64.

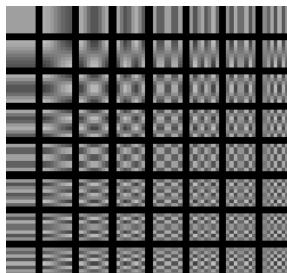


Ilustración 4 los 64 pesos base obtenidos de (5) de una matriz de 64 x 64

Las frecuencias horizontales aumentan de izquierda a derecha, y las frecuencias verticales incrementan de arriba hacia abajo, el valor calculado constante de la parte superior izquierda es comúnmente llamado “**Función base DC**”, y su correspondiente coeficiente en la función DCT en B_{00} es llamado “**coeficiente base DC**”; Los demás coeficientes B_{pq} son tomados como pesos aplicados a la función base, en este caso mostrado en una matriz de 8 x 8, los 64 pesos son ilustrados en la Ilustración 4.



El primer coeficiente en $B[0,0]$ es llamado Coeficiente Base DC

2.1.5.2 Uso perceptual DCT en una imagen

Como ya se ha explicado la función de la transformada DCT, surge el momento de identificar la razón de aplicación de la transformada en la similitud en una imagen, tomando como ejemplo en la Ilustración 5.



Image Courtesy of MIT

Ilustración 5 Imagen original y su compresión de una imagen por JPEG al 85%

Como se puede visualizar a simple vista el uso de compresión de una imagen al 85% no afecta drásticamente la percepción desde la perspectiva humana en los elementos semánticos de la imagen, por lo tanto la utilización de una función hash basada en la DCT muestra una tolerancia mayor a cambios de compresión y detalles pequeños en la estructura de una imagen.

2.1.6 Función Perceptual Hash BLOCK MEAN

En 2006, *Bian Yang, Fan Gu y Xiaomu Niu* propusieron la función Block Mean Perceptual Hash. Propusieron cuatro métodos ligeramente diferentes de esta función a continuación se presentan los métodos propuestos:

MÉTODO 1:

1. Convertir la imagen a escala de grises y normalizar la imagen original en un tamaño preestablecido.
2. Sea N la longitud de bits (por ejemplo, 256 bits) del valor final del hash. Divida los píxeles de la imagen en bloques I no superpuestos I_1, I_2, \dots, I_N .
3. Cifrar los índices de la secuencia de bloques $\{I_1, I_2, \dots, I_N\}$ usando una llave secreta K para obtener una secuencia de bloques un nuevo orden de barrido $\{I'_1, I'_2, \dots, I'_N\}$, la función criptográfica para llevar a cabo el cifrado puede ser cualquiera.
4. Calcular la media de los valores de píxeles de cada bloque. Es decir, calcular el valor medio de secuencia $\{M_1, M_2, \dots, M_N\}$ de la secuencia de bloques correspondiente $\{I'_1, I'_2, \dots, I'_N\}$, Finalmente obtener el valor medio M_d de la secuencia de valor medio.
5. Normalizar la secuencia de valor medio en una forma binaria y obtener el valor hash h como:

$$h(i) = \begin{cases} 0, & M_i < M_d \\ 1, & M_i \geq M_d \end{cases}$$

MÉTODO 2:

La única diferencia con el primer método es que los píxeles de la imagen se dividen en bloques superpuestos. El grado de la superposición se ajusta para que sea la mitad del tamaño de un bloque. Si se elige un tamaño preestablecido de 16x16 píxeles y un tamaño de bloque de 4x4 píxeles se utiliza, el primer método produciría un hash con un tamaño de 16 bits. Usando este método, los píxeles de la imagen se dividen en 49 bloques. Por lo tanto el hash tendría un tamaño de 49 bits.

MÉTODO 3:

El tercer método ofrece mayor robustez frente a los ataques de rotación. La diferencia es que los valores medios del arco de bloques de píxeles girar varias veces:

1. Girar por grados D la matriz M constituida por $\{M_1, M_2, \dots, M_N\}$, donde $D = \{0, 15, 30, \dots, 345\}$. Esto produce 24 matrices $(M_i, (i=1, 2, \dots, 24))$. Dividir cada una de las 24 matrices rotadas en N bloques. Obtener la secuencia de valor medio $\{M_{i1}, M_{i2}, \dots, M_{iN}\}$ de cada bloque y el valor medio M_{di} de esta secuencia, que forma 24 grupos de secuencias.
2. Normalizar la secuencia de valor medio en una forma binaria y obtener el valor hash h de los 24 grupos de secuencias de la siguiente manera:

$$h(i) = \begin{cases} 0, & M_i < M_d \\ 1, & M_i \geq M_d \end{cases}$$

MÉTODO 4:

El cuarto método es una combinación del segundo y tercer método. La imagen se divide en bloques traslapados como se describe en el segundo método. Además, el hash se calcula utilizando los valores medios rotados de los bloques como se describe en el tercer método.

2.1.7 Funciones de similitud para funciones perceptuales

Una vez definida la generación de firmas para un objeto multimedia como las imágenes, se procede a realizar la comparación de similitud mediante distancias, por lo que se procede a definir los siguientes conceptos.

2.1.7.1 Distancia

Un espacio de distancia (X, d) es un conjunto X (portador) compuesto por una Distancia d .

Una función $d: X \times X \rightarrow \mathbb{R}$ es definida como la **distancia** (o disimilaridad) en X si para todo $x, y \in X$, se mantiene que:

1. $d(x, y) \geq 0$ (Sin negatividad)
2. $d(x, y) = d(y, x)$ (Simetría)
3. $d(x, x) = 0$ (Reflexividad)

En la topología, una distancia con $d(x, y) = 0$ implica que $x = y$ y se le llama distancia simétrica.

Para cualquier distancia d , la función D_1 definida para $x \neq y$ por $D_1(x, y) = d(x, y) + c$, donde

$c = \max(d(x, y) - d(x, z) - d(y, z))_{x, y, z \in X}$ Y $D(x, x) = 0$, es una métrica.

También, $D_2(x, y) = d(x, y)^c$ es una métrica para un valor $c \geq 0$ suficientemente pequeño.

La función $D_3(x, y) = \inf \sum_i d(z_i, z_{i+1})$, donde el valor más bajo es tomado sobre toda las

Secuencias $x = z_0, \dots, z_{n+1} = y$, denominado como el **camino semimétrico** de un grafo completamente mapeado en X , donde para cada $x, y \in X$, el peso de la arista xy es $d(x, y)$.

2.1.7.2 Similitud

Sea X un conjunto, una función $s: X \times X \rightarrow \mathbb{R}$ es llamada **similitud** en X , si s es no negativa, simétrica y la desigualdad $s(x, y) \leq s(x, x)$

Se mantiene para todo $x, y \in X$, con la igualdad de si y solo si $x = y$.

Una de las principales transformaciones que se utiliza para obtener la distancia (disimilitud) d de una similitud s acotada por valores entre 0 y 1 es:

$$d = 1 - s$$

2.1.7.3 Semimétrico

Sea X un conjunto, una función $d: X \times X \rightarrow \mathbb{R}$ es llamada **semimétrica** en X si d es no negativa, simétrica y reflexiva es decir $(s(x, x) = 0 \text{ para } x \in X)$ sosteniendo que

$d(x, y) \leq d(x, z) + d(z, y), \forall x, y, z \in X$ (*Conocido como teorema de la desigualdad del triángulo*)

En topología es llamado pseudo-métrico (raramente semidistancia), mientras que el término semimétrico es algunas veces usado para simetría (una distancia $d(x, y)$ con $d(x, y) = 0$ solo si $x = y$).

2.1.7.4 Distancia y similitud en funciones perceptual hash

Para comparar dos objetos multimedia, la utilización de la firma creada por la función hash, es necesario utilizar las métricas necesarias para poder describir la similitud entre estas, uno de los métodos más simples y con mejores resultados son la tasa de error binario BER (*Bit Error Rate*), La distancia de haming y la correlación cruzada PCC (*Peak of cross correlation*); las primeras dos miden la distancia entre dos firmas, mientras la PCC mide la similitud de las dos firmas.

2.1.7.5 Tasa de error binario

Se define según (Yang, 2006) a p como el numero i de errores en bits de una firma normalizada de longitud k .

Por lo tanto $p = \frac{i}{k}$, donde $i \in \{0,1,\dots,k\}$ y $0 \leq p \leq 1$.

El número de errores en bits i es igual a la distancia de dos firmas, cuando se comparan diferentes objetos multimedia, por lo tanto la métrica de similitud es la siguiente:



Valores mayores a 0.5 se consideran perceptualmente diferentes.

Valores cercanos a 0 se consideran perceptualmente similares.

2.1.7.6 Distancia de Hamming

La métrica de hamming d_H (conocida como distancia **Dalal** en semántica) es una métrica en \mathbb{R}^n definida (Hamming, 1950) por:

$$|\{i: 1 \leq i \leq n, x_i \neq y_i\}| \quad (1)$$

En vectores binarios $x, y \in \{0,1\}^n$ la simétrica de hamming coincide con (Michel Deza, 2012) $|I(x) \Delta I(y)| = |I(x) \setminus I(y)| + |I(y) \setminus I(x)|$, donde $I(z) = \{1 \leq t \leq n: z_t = 1\}$.

Por lo tanto, $\max\{|I(x) \setminus I(y)|, |I(y) \setminus I(x)|\}$ es también una métrica.



La distancia hamming es la medida de distancia entre diferencia de dos cadenas.

En otras palabras la distancia hamming es la medida de distancia entre diferencia de dos cadenas, estas cadenas pueden ser números codificados en binario o algún otro sistema numérico u otros alfabetos, la Tabla 4 muestra ejemplos de la distancia de hamming en dos cadenas:

Cadena Binaria 1	Cadena Binaria 2	Distancia Hamming
00101	10101	1
12345	13344	2
well	wall	4

Tabla 4 Ejemplo de distancia hamming

2.1.7.7 Distancia Normalizada de Hamming

Para facilitar la comparación entre dos firmas la distancia de hamming presentada arriba puede ser normalizada con respecto a la longitud l de las cadenas.

Define (Venkatesan, 2000) la distancia normalizada como $\Delta n(x, y) = \frac{1}{n} \sum_{x \neq y} 1, i = 1, \dots, n$, si $x, y \in \{0,1\}^n$

Lo que sugiere que para calcular la distancia hamming en cadenas binarias la operación XOR puede ser usada como se indica en ecuación 7.

$$d(x, y) = \frac{1}{n} \sum_{i=1}^{n-1} (x \oplus y) \quad (7)$$

Una vez normalizada la distancia entre dos cadenas surge una representación diferente de la métrica distancia normalizada de hamming, definiendo los objetos perceptualmente iguales con un porcentaje alto ($EP \approx 100\%$), contrariamente para objetos perceptualmente diferentes con un valor bajo ($EP \approx 0\%$), donde $EP = 100 * \Delta n(x, y)$, conocido como **porcentaje de similitud** (o igualdad).

2.2 ESTADO DEL ARTE

Por medio de la búsqueda, se han podido identificar sistemas computacionales similares existentes los cuales no desarrollan la búsqueda de forma local. A continuación se listan estas herramientas en la Tabla 5.

NOMBRE DE LA HERRAMIENTA	TIPO	CARACTERÍSTICAS
Google Image Search	Privativo con licencia cerrada.	Sistema CBIR de Google, el cual no funciona en todas las imágenes, identifica imágenes similares en las páginas web indexadas por google.
eBay Image Search	Privativo con licencia cerrada.	Buscador diseñado para encontrar ropa a la venta en eBay mediante la imagen de una prenda deseada.
Incogna Image Search	Privativo con licencia cerrada.	Buscador web de imágenes desarrollado por la Universidad de Otawua, el cual no utiliza los metadatos de las imágenes para clasificarlas

Tabla 5 Sistemas similares

CAPÍTULO 3

ANÁLISIS

3.1 PROPUESTA DE SOLUCIÓN

Se propone realizar un motor de búsqueda en una aplicación de escritorio que permita el ingreso de una imagen para encontrar resultados afines a la misma, las imágenes desplegadas podrán ser también algunas incluidas en archivos. Se podrá especificar el porcentaje de similitud que desea obtenerse en los resultados.

Una vez ingresada la imagen a buscar, se elegirá el porcentaje de similitud que se desea alcanzar y al terminar la búsqueda se mostrarán los resultados obtenidos.

3.2 CARACTERÍSTICAS DEL SISTEMA

A continuación se describen las características del sistema a desarrollar en la Tabla 6.

NOMBRE DE LA HERRAMIENTA	TIPO	CARACTERÍSTICAS
Motor de búsqueda local mediante imágenes	Trabajo de investigación y desarrollo de software presentado ante la ESCOM, del I.P.N. para obtener la titulación curricular en la carrera de Ingeniería en Sistemas Computacionales por Omar Martínez Rosas, México, D.F., 2015	Motor de búsqueda implementado como una aplicación de escritorio desarrollada en .NET C# que permite la búsqueda local de imágenes mediante ejemplo y que utiliza la metodología <i>Perceptual Hashing</i> .

Tabla 6 Características del sistema

3.3 ESTUDIO DE FACTIBILIDAD

3.3.1 Factibilidad técnica

La factibilidad técnica es la evaluación del sistema donde se demuestra que puede ser desarrollarse. Con este objetivo, se ha llevado a cabo un análisis de los requerimientos de software, de hardware, de IDE de programación, de los lenguajes de programación y de los sistemas operativos. Los resultados se muestran a continuación:

Requerimientos de software

Se consideraron tres puntos para realizar la elección del software necesario para el desarrollo del sistema:

1. Lenguaje de Programación
2. Gestor de base de datos
3. IDE
4. Sistema Operativo

1.- Lenguaje de Programación: Los lenguajes de programación orientados a objetos, dominados, conocidos y útiles para el desarrollo del sistema que se consideraron ordenados por expertiz, definidos en la Tabla 7.

Lenguaje de programación	Software libre	Portabilidad	Desarrollo de interfaces de usuario
Microsoft Visual C# .NET	No	Únicamente en Sistemas Operativos Windows (Framework)	Se pueden desarrollar usando el estándar de aplicaciones Windows.
Java	Si	Si, aunque se requiere la Máquina Virtual instalada.	Se pueden desarrollar con diferentes IDEs que facilitan la tarea.
Python	Si	La mayoría de programas en Python se pueden ejecutar en la mayoría de plataformas.	Se pueden desarrollar con distintas herramientas.

Tabla 7 Lenguajes de programación

Tomando en cuenta las comparaciones entre los lenguajes de programación se pensó en utilizar Python, por la rapidez en que se desarrollan los scripts, por su característica de multiparadigma y por ser software libre, pero por nivel de *expertiz* que se tiene en los lenguajes analizados, se tomó la decisión de elegir Microsoft Visual C# .NET que, aunque represente realizar una inversión para comprar la licencia del IDE, permitirá que el tiempo ahorrado en aprender el lenguaje pueda ser utilizado en la investigación de los métodos de validación, de igual manera permite una mejor interacción con el sistema operativo más usado por usuarios el cual es Microsoft Windows .

2.- Gestor de base de datos: Los gestores de base de datos dominados y conocidos útiles para el desarrollo del sistema considerados para el desarrollo del sistema se enlistan en la Tabla 8.

Gestor de bases de datos	Características	Licencia
MS SQL	Sistema de manejo de bases de datos del modelo relacional, desarrollado por la empresa Microsoft.	Comercial - varios niveles de características basadas en versión libre Crippleware.
MySQL	Sistema de gestión de bases de datos relacional, multihilo y multiusuario	GPL Open Source, comercial.
PostgreSQL	Sistema de gestión de bases de datos relacional orientado a objetos y libre, publicado bajo la licencia BSD.	BSD Open Source

Tabla 8 Gestores de base de datos

Tomando en cuenta las comparaciones entre los gestores de bases de datos se consideró el uso de PostgreSQL por el alto nivel de madurez que tiene, pero por la conexión entre el lenguaje de programación en el que se va a desarrollar el sistema (Microsoft Visual C#) ya que la comunicación entre la aplicación y la base de datos será más estable.

3.- IDE (Integrated Development Environment): Se eligió usar Microsoft Visual Studio 2013: entorno de desarrollo integrado (IDE) de Visual C# que tiene un conjunto de herramientas de desarrollo expuestas a través de una interfaz de usuario común. Algunas de las herramientas se comparten con otros lenguajes de Visual Studio, y otras, como el compilador de C#, son exclusivas de Visual C#.

4.- Sistema Operativo: Debido a que el IDE elegido solo puede ser ejecutado en Sistemas Operativos Microsoft Windows, se eligió la versión 8 Professional.

3.4 HISTORIAS DE USUARIO

Las historias de usuario son utilizadas en la metodología XP (*Extreme Programming*), elegida para el desarrollo de este proyecto, para representar los requerimientos de software.

A continuación se describen las historias de usuario resultantes de la fase de exploración, acompañadas de la descripción de las pruebas de aceptación que permitirán verificar que el sistema de software producido cumple con las especificaciones y que logra su cometido.

Las tablas siguientes están compuestas por los siguientes elementos:

- ❖ **Número.**- Número de la iteración del sistema, en la que estará incluida la implementación de la historia de usuario.
- ❖ **Nombre.**- Nombre de la historia de usuario.
- ❖ **Usuario.**- Persona encargada de interactuar con el sistema en la historia de usuario.
- ❖ **Prioridad en el negocio.**- Se puede definir como Alta, Media o Baja. Toma en cuenta la importancia de implementar la historia de usuario para lograr el objetivo del sistema.
- ❖ **Responsable desarrollo.**- Persona encargada de desarrollar e implementar la historia de usuario.
- ❖ **Descripción.**- Enunciados que establecen lo que debe hacer la historia de usuario.
- ❖ **Observaciones.**- Notas adicionales sobre la historia de usuario.
- ❖ **Tabla de pruebas de aceptación.**- Lista de enunciados que establecen las condiciones que definirán si la implementación de la historia de usuario funciona.

3.4.1 HU1: Primera ejecución del sistema

Historia de Usuario 1

NOMBRE: Primera ejecución del sistema	NÚMERO: 1
USUARIO: Usuario final	ITERACIÓN: 1
PRIORIDAD EN NEGOCIO: Alta	
RESPONSABLE DESARROLLO: Omar Martínez Rosas	
DESCRIPCIÓN:	
Cuando el sistema se ejecuta por primera vez se requiere que el usuario seleccione un directorio para indexar los archivos contenidos en el mismo. En este directorio se realizarán las búsquedas posteriores.	

PRUEBAS:

Seleccionar un directorio para indexar y verificar que todos los archivos soportados que se encuentren dentro del directorio y subdirectorios se indexen en la base de datos.

Extensiones de imágenes soportadas: bmp, jpg, png, jpeg, tif, tiff, jif, jfif, jp2, jpx, j2k, j2c.

Extensiones de documentos que contienen imágenes soportadas: docx, odt, pptx.

Verificar que se muestra un mensaje que informa que el sistema está indexando los archivos.

3.4.2 HU2: Generación de firmas de archivos indexados

Historia de Usuario 2

NOMBRE: Generación de firmas de archivos indexados	NÚMERO: 2
USUARIO: Usuario final	ITERACIÓN: 1
PRIORIDAD EN NEGOCIO: Alta	
RESPONSABLE DESARROLLO: Omar Martínez Rosas	
DESCRIPCIÓN:	
Una vez indexados los archivos contenidos en el directorio especificado se genera automáticamente la firma para cada uno de dichos archivos con las 5 funciones especificadas para este propósito (Block Mean Método 1, Block Mean Método 2, Block Mean Método 3, Block Mean Método 4 y DCT Perceptual Hash).	

PRUEBAS:

Verificar que el sistema muestre un mensaje que indique que se encuentra generando las firmas de cada archivo.

3.4.3 HU3: Carga de imagen a buscar

Historia de Usuario 3

NOMBRE:	Carga de imagen a buscar	NÚMERO:	3
USUARIO:	Usuario final	ITERACIÓN:	1
PRIORIDAD EN NEGOCIO:	Alta		
RESPONSABLE DESARROLLO:	Omar Martínez Rosas		
DESCRIPCIÓN:	<p>Permitir seleccionar una imagen a buscar.</p>		

PRUEBAS:

Verificar que se pueda cargar una imagen a buscar por medio de un selector de archivos que se muestre al seleccionar un botón en la pantalla principal de búsqueda.

Verificar que pueda cargar una imagen a buscar a través de arrastrar la imagen a un contenedor en la pantalla principal de búsqueda.

3.4.4 HU4: Buscar imagen

Historia de Usuario 4

NOMBRE: Buscar imagen	NÚMERO: 4
USUARIO: Usuario final	ITERACIÓN: 1
PRIORIDAD EN NEGOCIO: Alta	
RESPONSABLE DESARROLLO: Omar Martínez Rosas	
DESCRIPCIÓN:	
Buscar imágenes similares una imagen seleccionada en el directorio especificado a través de la función de comparación especificada y con porcentaje de similitud especificado. (Predeterminados: Block Mean Método 4 y 85% de similitud)	

PRUEBAS:

Verificar que se muestren las imágenes similares a la imagen buscada con el porcentaje de similitud especificado.

Verificar que al dar clic sobre el enlace de ruta de un elemento resultado, se abra el archivo al que pertenezca esa ruta.

Verificar que al dar doble clic sobre un elemento resultado, se abra el archivo seleccionado.

3.4.5 HU5: Configurar búsqueda

Historia de Usuario 5

NOMBRE: Configurar búsqueda	NÚMERO: 5
USUARIO: Usuario final	ITERACIÓN: 1
PRIORIDAD EN NEGOCIO: Alta	
RESPONSABLE DESARROLLO: Omar Martínez Rosas	
DESCRIPCIÓN:	
Permitir seleccionar la función para realizar la comparación de imágenes, el porcentaje de similitud que se desea obtener y reiniciar la aplicación como si se ejecutara por primera vez.	

PRUEBAS:

Verificar que se permita seleccionar cualquier función de comparación, uno a la vez.

Verificar que se permita seleccionar el porcentaje de similitud a partir del 50%.

Verificar que se guarden los valores seleccionados.

Verificar que al seleccionar la opción de reiniciar el sistema devuelva los valores predeterminados para la función de comparación, el porcentaje de similitud, se borren los datos indexados y al abrir la aplicación de nuevo se requiera seleccionar un directorio de búsqueda.

3.5 GESTIÓN DE RIESGOS

El análisis de riesgos es de gran utilidad para identificar aquellas situaciones que no están bajo control en el proceso de desarrollo, pero que tienen impacto en el desarrollo del proyecto. Tener una gestión adecuada de los riesgos, permite poder prevenir la mayor cantidad posible y en caso de que ocurran, tener un plan de contingencia que minimice los daños y costos en tiempo y recursos.

3.5.1 Evaluación inicial de riesgos

En la siguiente Tabla 9 se muestran los riesgos que pueden presentarse durante el desarrollo del proyecto.

ID	Descripción	Tipo
R01	Poca experiencia de interacción con las herramientas seleccionadas para el desarrollo.	Tecnológico
R02	Poco conocimiento de los métodos a implementar y la teoría matemática necesaria para el tratado de imágenes.	Negocio
R03	Estimación errónea de costos, tiempos y Alcance.	Planeación
R04	El desarrollo del proyecto toma de mayor tiempo del previsto.	Planeación
R05	Falta de datos para prueba.	Contexto
R06	Los métodos utilizados no entregan los resultados esperados.	Producto

Tabla 9 Evaluación inicial de riesgos

3.5.2 Análisis de riesgos

A continuación se describe de forma detallada en la Tabla 10 cada uno de los riesgos presentados en la tabla anterior (Tabla 9).

ID Riesgo	ANÁLISIS DE RIESGO
R01	<p>Probabilidad: Media</p> <p>Descripción: Puede existir poca experiencia en las herramientas elegidas para el desarrollo del proyecto y para su documentación.</p> <p>Impacto: Pueden surgir retrasos en el cumplimiento de las tareas planeadas</p>
R02	<p>Probabilidad: Alta</p> <p>Descripción: Debido a la formación de Ingeniería en Sistemas se puede tener poco conocimiento de los métodos utilizados para el procesamiento y análisis de imágenes.</p> <p>Impacto: Puede no cumplirse el objetivo del proyecto si se desconoce cómo se utilizan los métodos de validación, si no se tiene clara la importancia de la validación, si se desconocen los términos usados en la descripción de la aplicación de los métodos, si no se tiene un marco de referencia sobre qué resultados esperados de las validaciones, entre otras situaciones que impliquen la falta de conocimiento del campo de estudio.</p>
R03	<p>Probabilidad: Baja</p> <p>Descripción: Puede ocurrir que los cálculos realizados no cubran los costos reales referentes al desarrollo del proyecto.</p> <p>Impacto: Puede provocar carga de trabajo mayor a la esperada, retrasos en la entrega y aumento de costos del producto final.</p>
R04	<p>Probabilidad: Media</p> <p>Descripción: El desarrollo del proyecto toma más tiempo del previsto.</p> <p>Impacto: Puede causar retraso en la entrega y que no se cumpla el objetivo del proyecto.</p>
R05	<p>Probabilidad: Media</p> <p>Descripción: Puede que no se cuente con suficientes datos para realizar pruebas y que permitan validar que el sistema cumple con los objetivos definidos.</p> <p>Impacto: La funcionalidad del proyecto no puede ser evaluada adecuadamente.</p>
R06	<p>Probabilidad: Media</p> <p>Descripción: Debido a la falta de experiencia en el tratado de imágenes, puede que los métodos no arrojen los resultados esperados.</p> <p>Impacto: No se cumplirán por completo los objetivos del desarrollo del proyecto.</p>

Tabla 10 Análisis de riesgos

3.5.3 Gestión de riesgos: Estrategias de control

En la siguiente Tabla 11 se describen las medidas de prevención contempladas para cada uno de los riesgos analizados anteriormente:

ID Riesgo	Prevención	Estrategia de minimización	Contingencia
R01	Realizar un análisis de las herramientas disponibles para el desarrollo y la documentación del sistema. Seleccionar en las que se tenga mayor dominio y que coadyuven en el cumplimiento de los objetivos del proyecto.		Investigar el uso de las herramientas seleccionadas y aumentar la expertiz en ellas.
R02	Realizar una investigación de los métodos de validación de los términos de los cuales no se tenga un conocimiento amplio.		
R03	Usar como apoyo un sistema de software de simulación de costos para verificar que el presupuesto sea correcto.	Reajuste de presupuesto.	
R04		Aumentar la carga y jornada de trabajo, hasta lograr un ajuste en los tiempos definidos.	
R05	Recopilar datos para pruebas a través de los expertos que asesoren el trabajo de investigación		Generar pruebas por modulo.
R06	Investigar a detalle cada uno de los métodos de validación elegidos para implementar. Reproducir su funcionamiento con varios ejemplos y escenarios para lograr el mayor dominio de los mismos.	Realizar correcciones en la implementación.	Cambiar de método de validación.

Tabla 11 Estrategias de control

3.6 ESTIMACIÓN DE COSTOS

El costo del desarrollo del sistema se especifica en la siguiente tabla:

Aspecto	Consideraciones	Costo
Equipo de computo	-2 equipos de cómputo con las siguientes características mínimas: Procesador Intel Core i5 4 Gigabytes en memoria RAM 250 Gigabytes en disco duro Sistema operativo Windows 8	\$30,000.00
Energía eléctrica	- 2 equipo de cómputo - Precio promedio del KWH: \$1.327 - 7 horas de trabajo por día	\$3,700.00
Transporte	- 5 días a la semana - Tarifa actual de boleto del STM: \$5.00	\$4,000.00
Desarrollo	- 2 Desarrolladores - Sueldo mensual: \$20,000.00	\$400,000.00
IDE	- Licencia Visual Studio Professional 2013	\$7,322.00
Total:		\$445,022.00

Tabla 12 Estimación costos

CAPÍTULO 4

DISEÑO

Se generará un sistema que implemente un motor de búsqueda de información a través de imágenes y cuente con despliegue visual de los resultados.

4.1 Diagrama a bloques

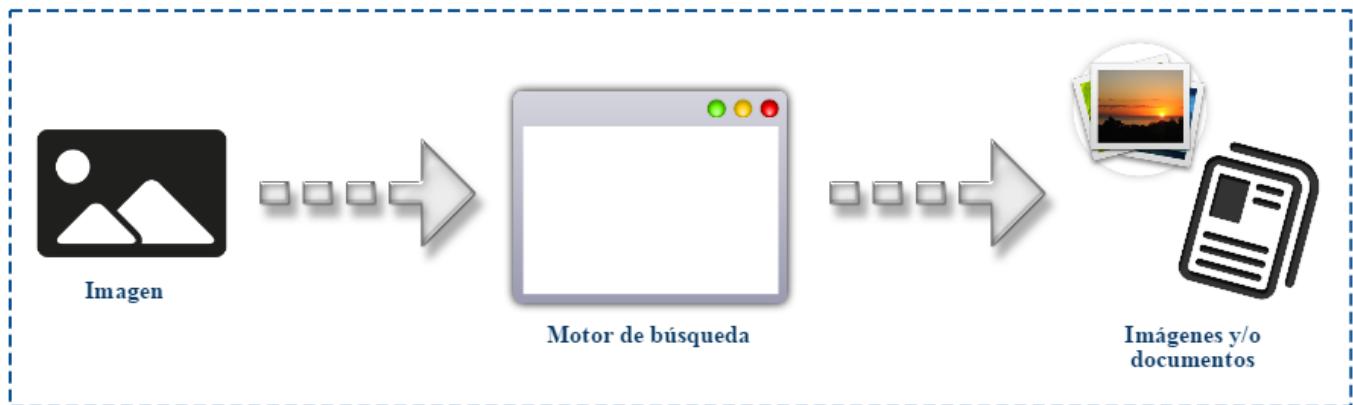


Ilustración 6 Diagrama a bloques del sistema

En la Ilustración 6 se muestra el diagrama a bloques del sistema a desarrollar. Se describen a continuación las etapas que conforman dicho diagrama, que corresponden al accionar del sistema:

- ❖ Imagen: Entrada del sistema. Imagen cuyas coincidencias se buscarán en el sistema de archivos.
- ❖ Motor de búsqueda: Conjunto de algoritmos que permitirá buscar las coincidencias de información con la entrada del sistema.
- ❖ Imágenes y/o documentos: Salida del sistema, se muestran las imágenes y/o documentos que coinciden con la entrada del sistema.

4.2 Diagrama a bloques

A continuación se presenta en la Ilustración 7 la descripción general del sistema del Motor de búsqueda local mediante imágenes.

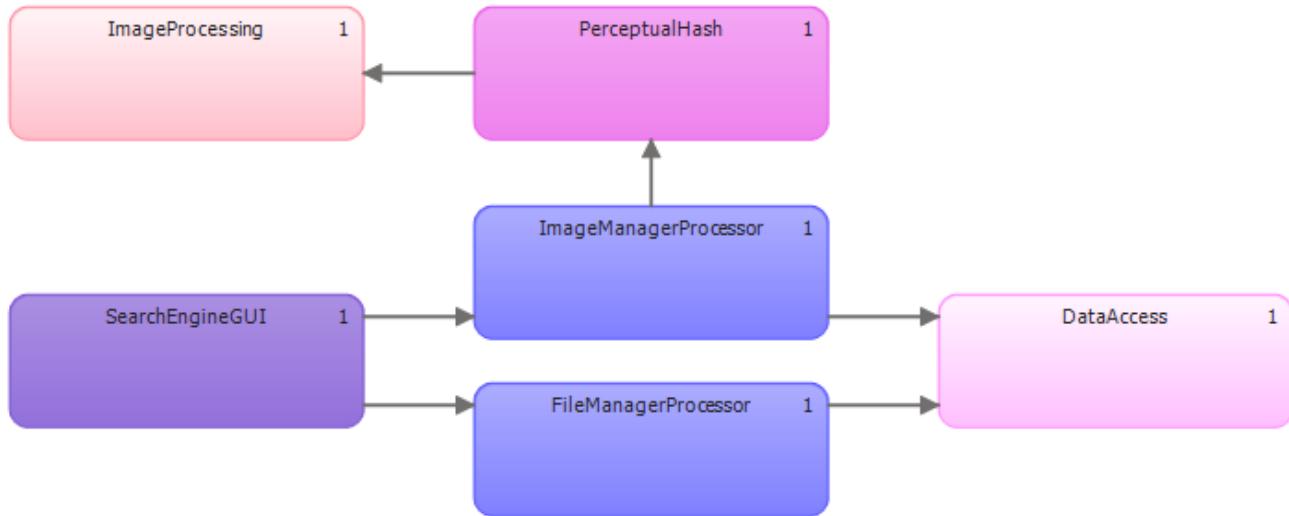


Ilustración 7 Diagrama Bloques

Los bloques que conforman la arquitectura del sistema son los siguientes:

- **SearchEngine GUI**
 - Representa la interfaz gráfica, que coordina y comunica todos los elementos del sistema
 - Mantiene flujo de datos con el bloque *ImageManagerProcessor* y *FileManagerProcessor*
- **ImageManagerProcessor**
 - Realiza todas las tareas de acceso a base de datos, enfocándose específicamente en los archivos de imágenes.
 - Realiza comunicación con el bloque *PerceptualHash* para la realización de generación de firmas de cada una de las imágenes indexadas.
- **FileManagerProcessor**
 - Realiza todas las tareas de acceso a base de datos, enfocándose específicamente en los archivos.
 - Realiza la indexación y actualización de los índices de datos en el directorio de trabajo.
 - Realiza la tarea de extracción de imágenes contenidas en documentos.
- **DataAccess**
 - Realiza todas las tareas de consulta de información en la base de datos, proporcionando información a los bloques que requieran información de la indexación de archivos e imágenes indexadas.
- **PerceptualHash**
 - Mantiene el control y lógica de generación de funciones hash, cálculo de distancias entre objetos multimedia, ofreciendo flexibilidad para diferentes tipos de objetos multimedia como audio, video, o imágenes.
- **ImageProcessing**
 - Realiza todas las tareas de procesamiento de imágenes necesarias en la generación de firmas.

4.3 Diagrama de clases

A continuación se presentan los diagramas de clase los cuales describen la estructura del sistema mostrando las clases, atributos y métodos además las relaciones entre los objetos. Ya que la metodología utilizada no requiere tanto detalle en el modelado y diseño del sistema se realizó la documentación de las clases en un documento *HTML* con *Doxygen* como referencia para mayor detalle del sistema completo.

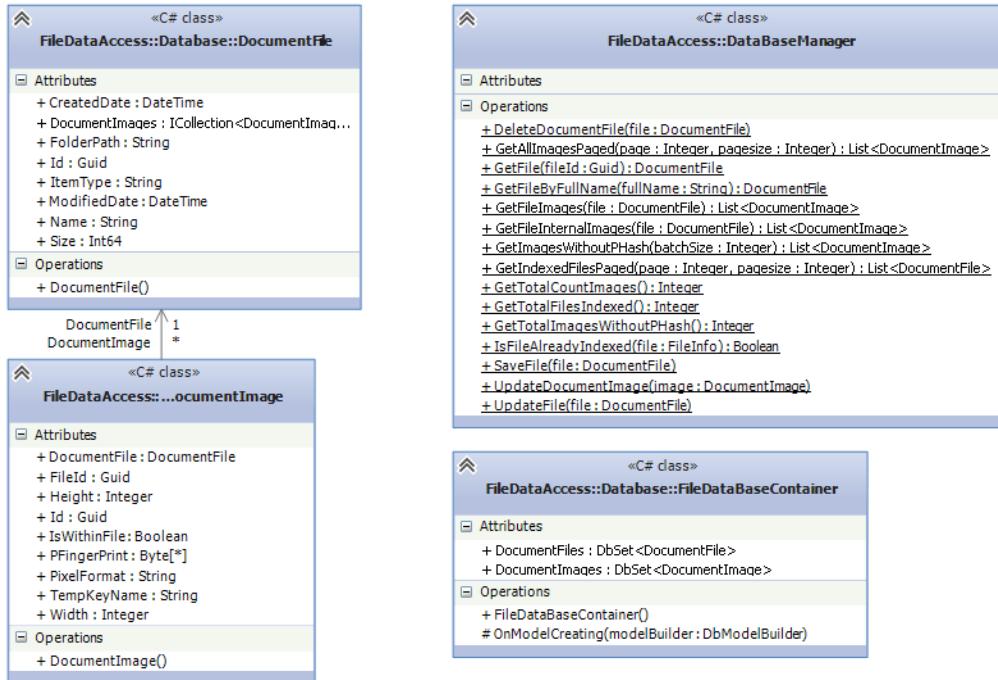


Ilustración 8 Diagrama de clases correspondientes al acceso de datos

La está compuesta de las clases que se encargan de toda lógica operaciones de bases de datos del Motor de búsqueda local mediante imágenes, en el cual la clase llamada **DataBaseManager** implementa todas las bases necesarias para obtener la información requerida en tareas como la indexación de archivos, búsqueda de imágenes y almacenamiento de firmas.

Las clases **DocumentFile** y **DocumentImage** representan los elementos en la base de datos que se obtienen por medio de un ORM (*Object-relational mapping*) como *EntityFramework* en este caso, el acceso de datos se mantiene por un proveedor de datos por medio de conexiones alojadas en la clase **FileDataBaseContainer** encargada de mantener el control de flujo de datos con las bases de datos.

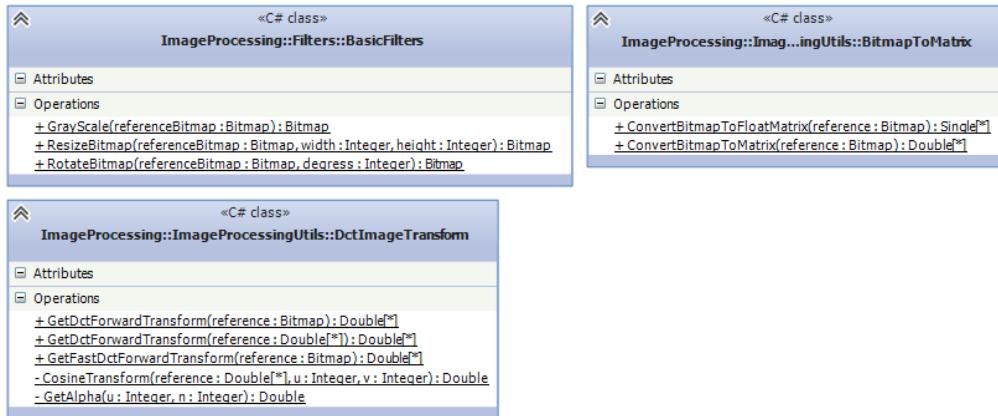


Ilustración 9Diagrama de clases Procesamiento de imágenes

La Ilustración 9 está compuesta de las clases que se encargan de realizar las operaciones a las imágenes, en ella se muestra la clase ***BasicFilters*** la cual realiza la aplicación de filtros básicos como Escala de grises, rotación de una imagen y redimensionamiento.

La clase **BitmapToMatrix** se encarga de realizar la transformación de una imagen a una secuencia de números flotantes o una matriz de dos dimensiones.

La clase **DCTImageTransform** se encarga de realizar las operaciones de transformación a la función DCT descrita con más detalle en el marco teórico de este documento.

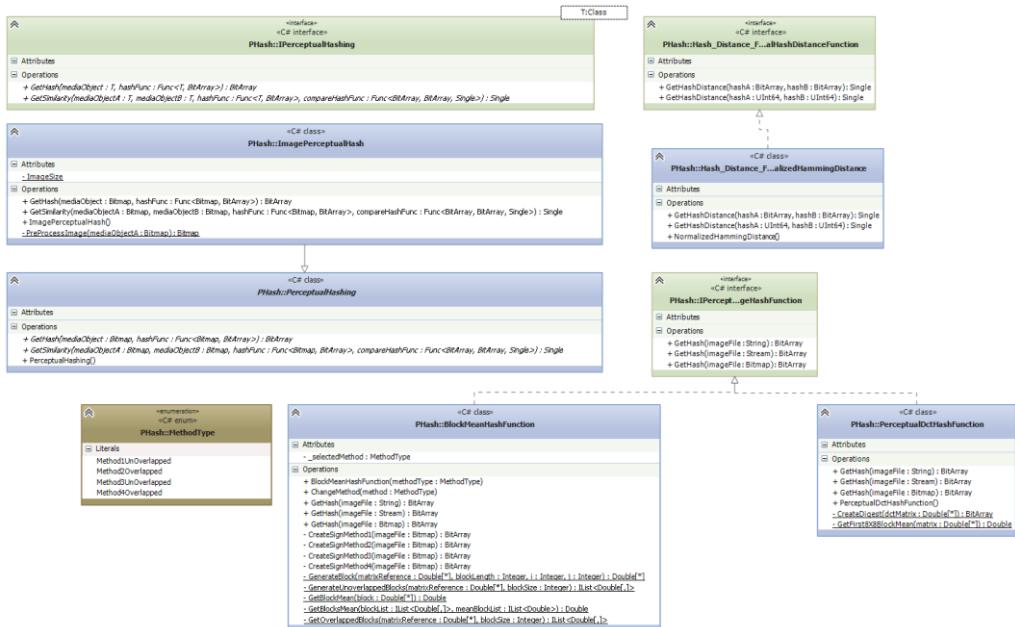


Ilustración 10 Diagrama de clases Perceptual Hash

La Ilustración 10 está compuesta de las clases que implementan la medida de similitud entre objetos multimedia mediante el cálculo y comparación de los hashes perceptuales.

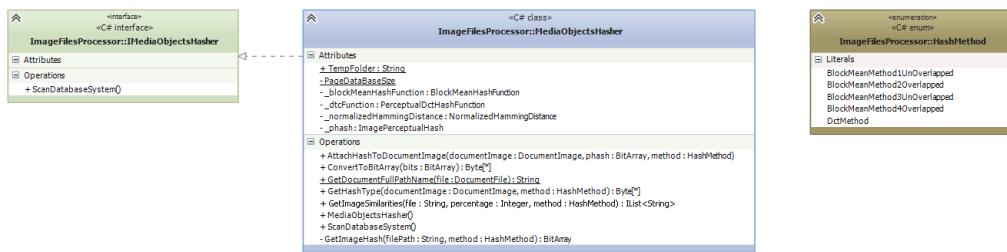


Ilustración 11 Diagrama de clases Media Object Handler

La Ilustración 11 está compuesta de las clases encargadas de realizar la generación de firmas de los archivos indexados, además de comparar una imagen externa contra los datos indexados.

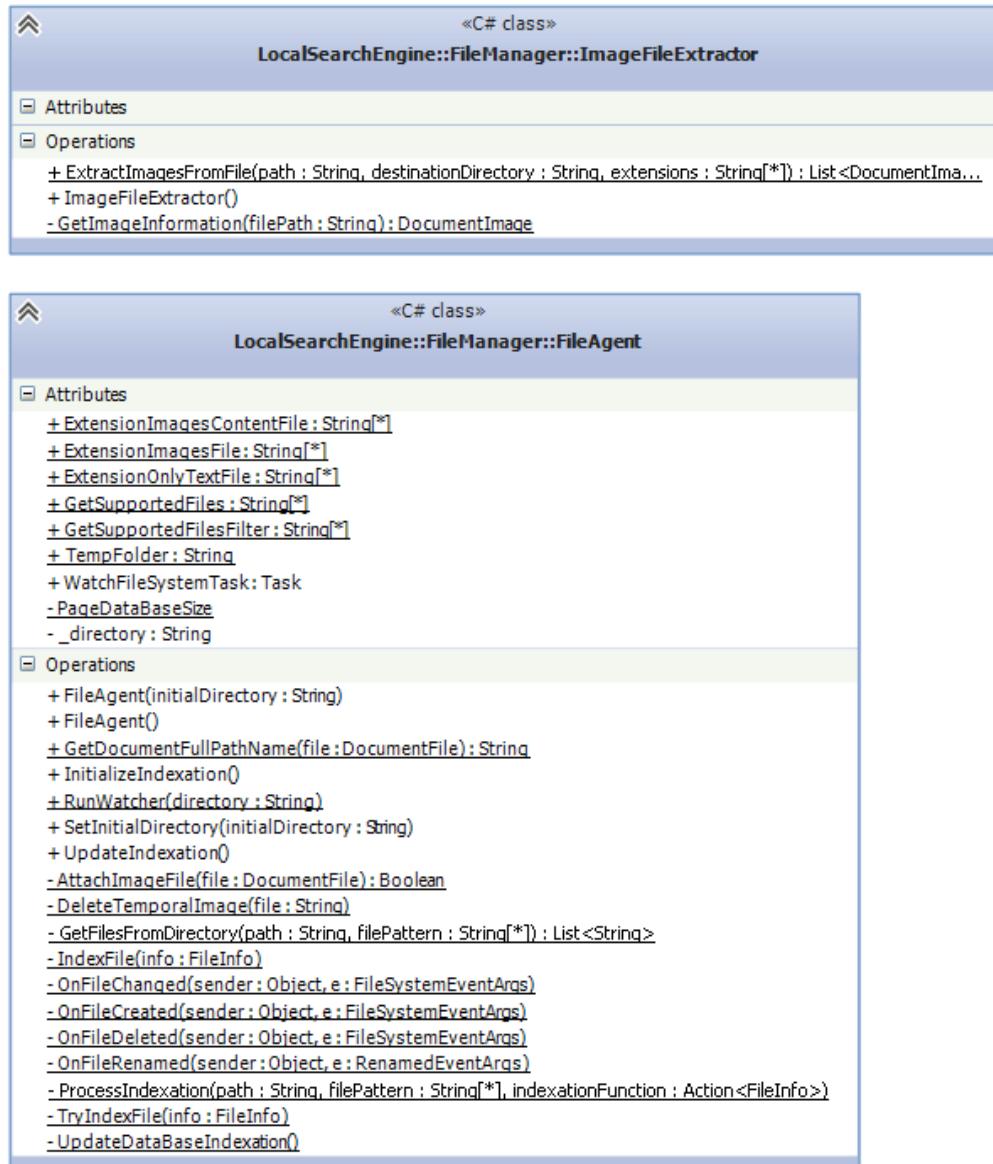


Ilustración 12 Diagrama de clases FileAgent

La Ilustración 12 está compuesta de las clases encargadas para formar un agente de archivos el cual es responsable de administrar el proceso de indexación, mantiene un registro de los cambios en el nivel de ejecución en la estructura de archivos dentro de la búsqueda local de directorio en el que se está llevando a cabo la búsqueda incluyendo subdirectorios dentro de este, de igual manera estas clases se encargan de la extracción de archivos sin necesidad de usar la definición interna de estos archivos.

4.4 Diagramas de secuencia

A continuación se presentan los diagramas de secuencia del Motor de búsqueda local mediante imágenes los cuales describen de forma gráfica los procesos que lleva acabo el sistema, estos son mejor descritos en la sección de desarrollo:

Generación de la firma de una imagen

En la Ilustración 13 se muestra el proceso secuencial de generación de la firma de una imagen, este proceso es mayormente detallado en la sección **5.1.2 Procesamiento de imágenes (generación firmas)**.

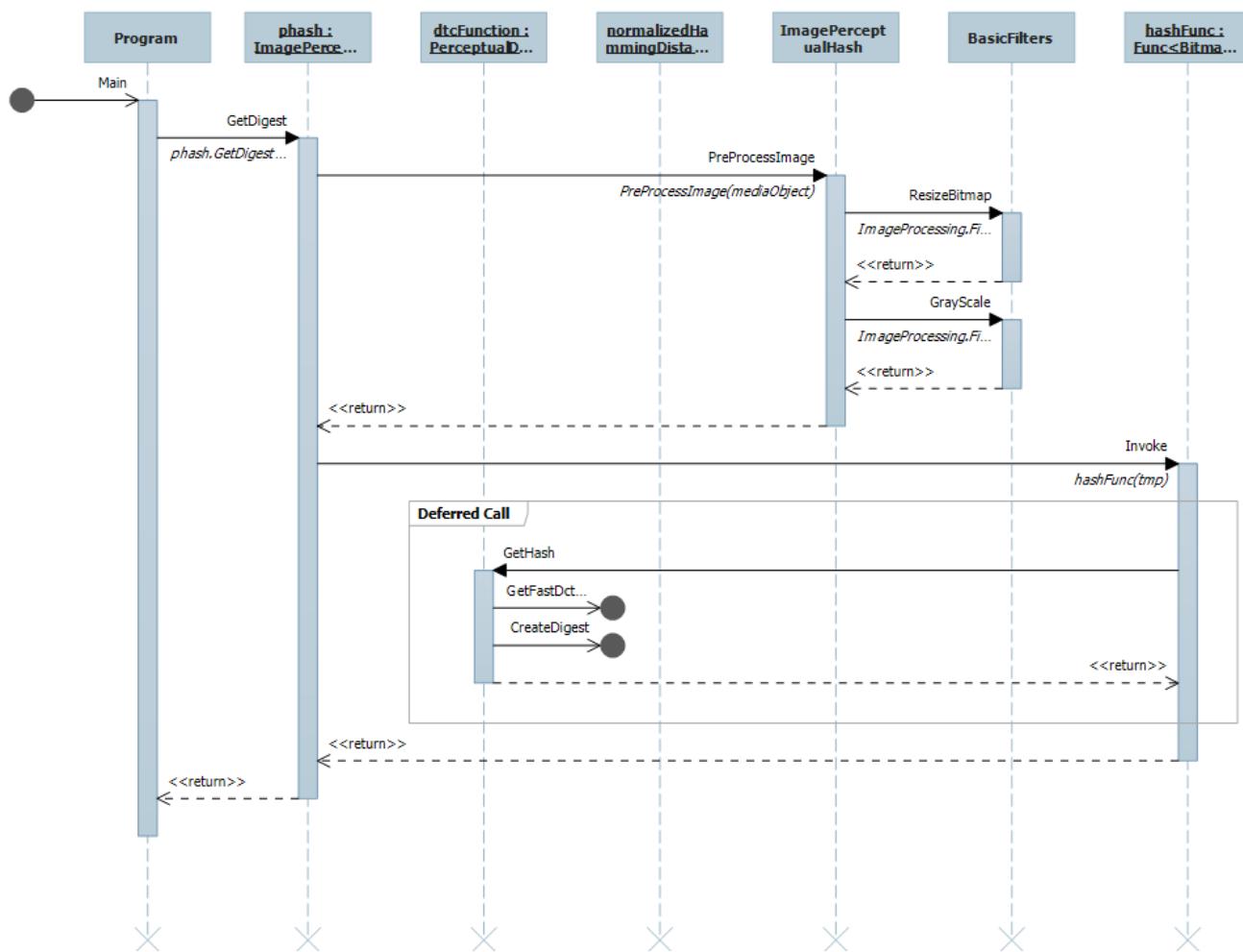


Ilustración 13 Diagrama secuencia Generación Firma

Generación de Firma e indexación

En la Ilustración 14 se muestra el proceso secuencial de generación de la firma de una imagen indexada en la base datos, este proceso es mayormente detallado de la sección 5.1.1 Indexación de archivos a la sección 5.1.2 Procesamiento de imágenes (generación firmas) realizando el almacenamiento de las firmas en la base de datos.

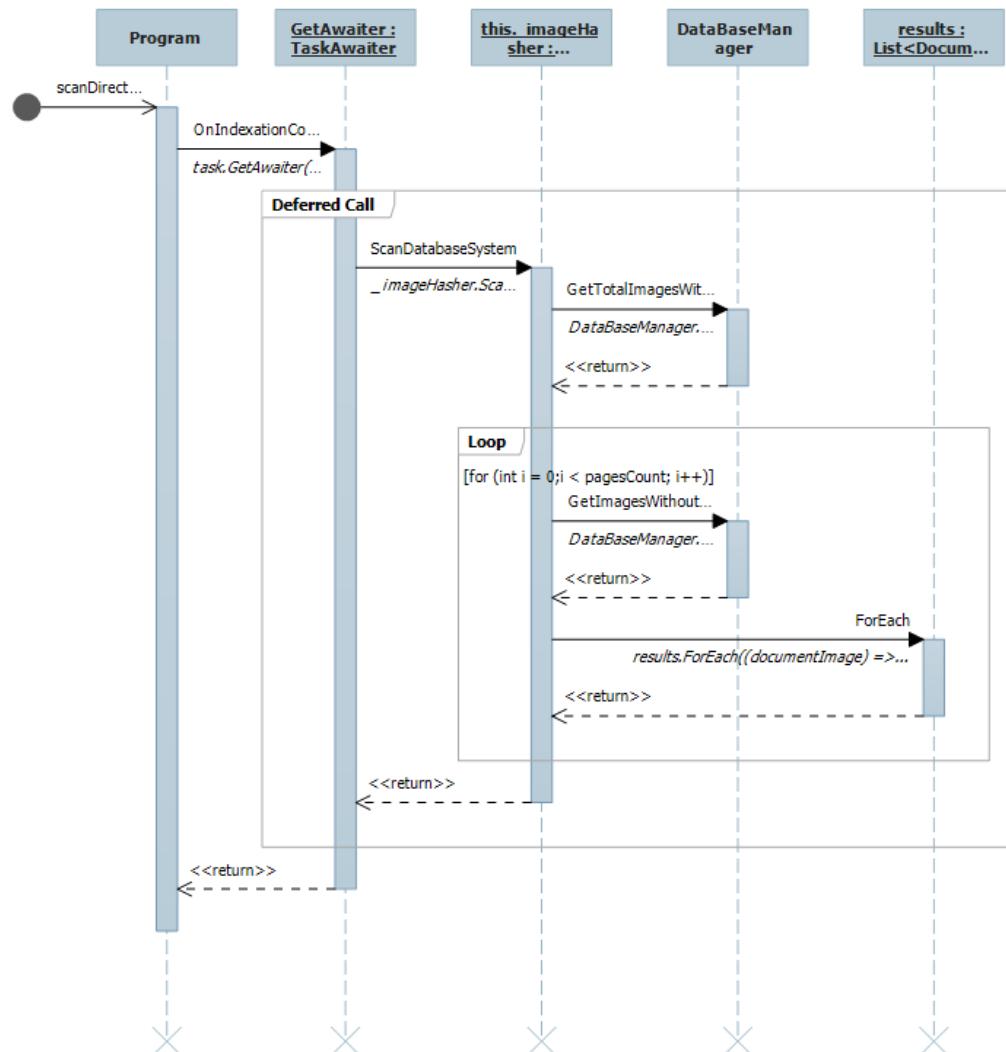


Ilustración 14 Diagrama secuencia Generación de firma desde archivos indexados

Extracción de imágenes en documentos

La extracción de imágenes en documentos es realizada por el sistema tomando en cuenta al archivo (documento) como un archivo comprimido, se procede a verificar todos sus elementos internos que cumplan con la característica de ser imagen, se extrae la imagen y se guarda en un directorio temporal para su posterior uso, este proceso es descrito gráficamente en la Ilustración 15.

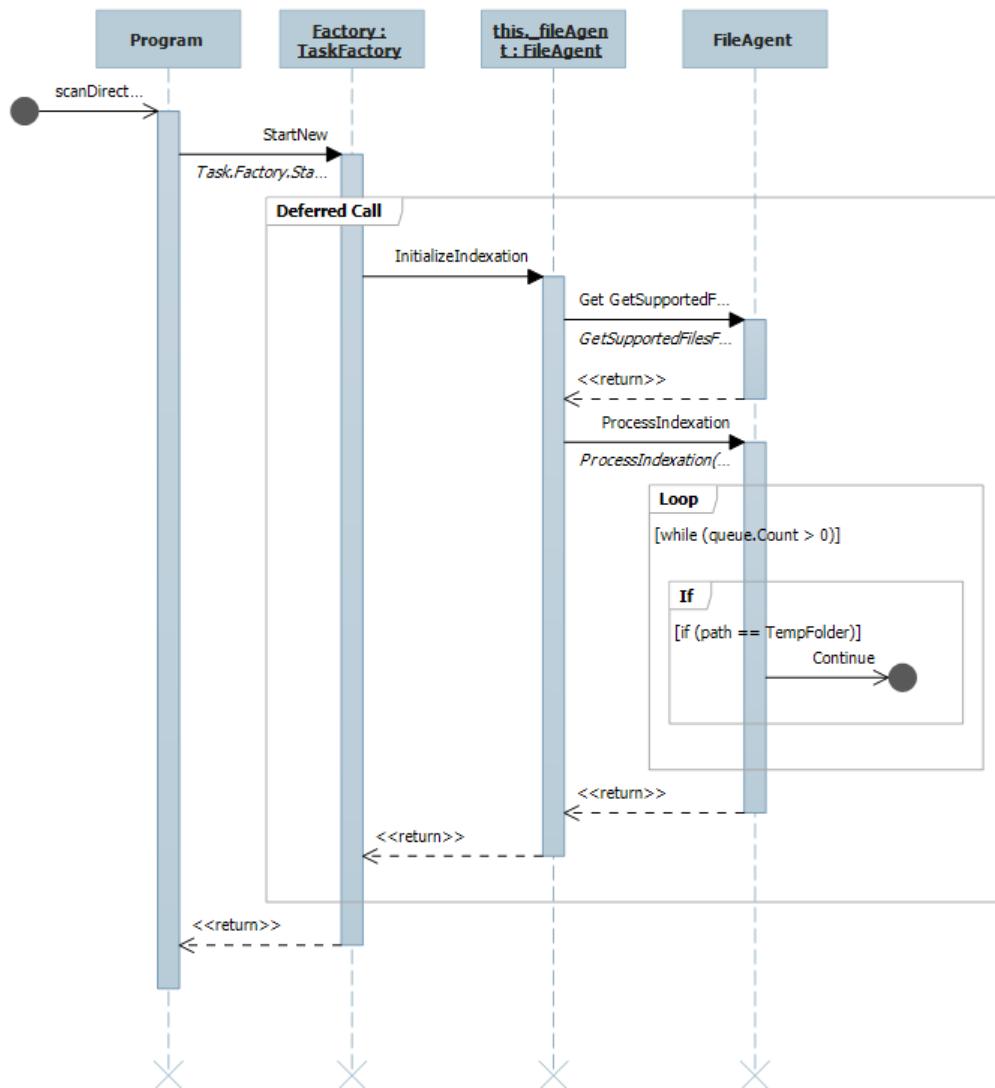


Ilustración 15 Diagrama de secuencia Extracción de imágenes en documentos

Búsqueda de imágenes similares

En la Ilustración 16 se muestra el proceso secuencial desde la generación de la firma de una imagen para su posterior búsqueda contra firmas indexadas en la base datos, este proceso es la unión de los procesos anteriormente descritos, concluyendo en el paso de comparación de firmas detallado de la sección 5.2.2 Cálculo de similitud entre firmas.

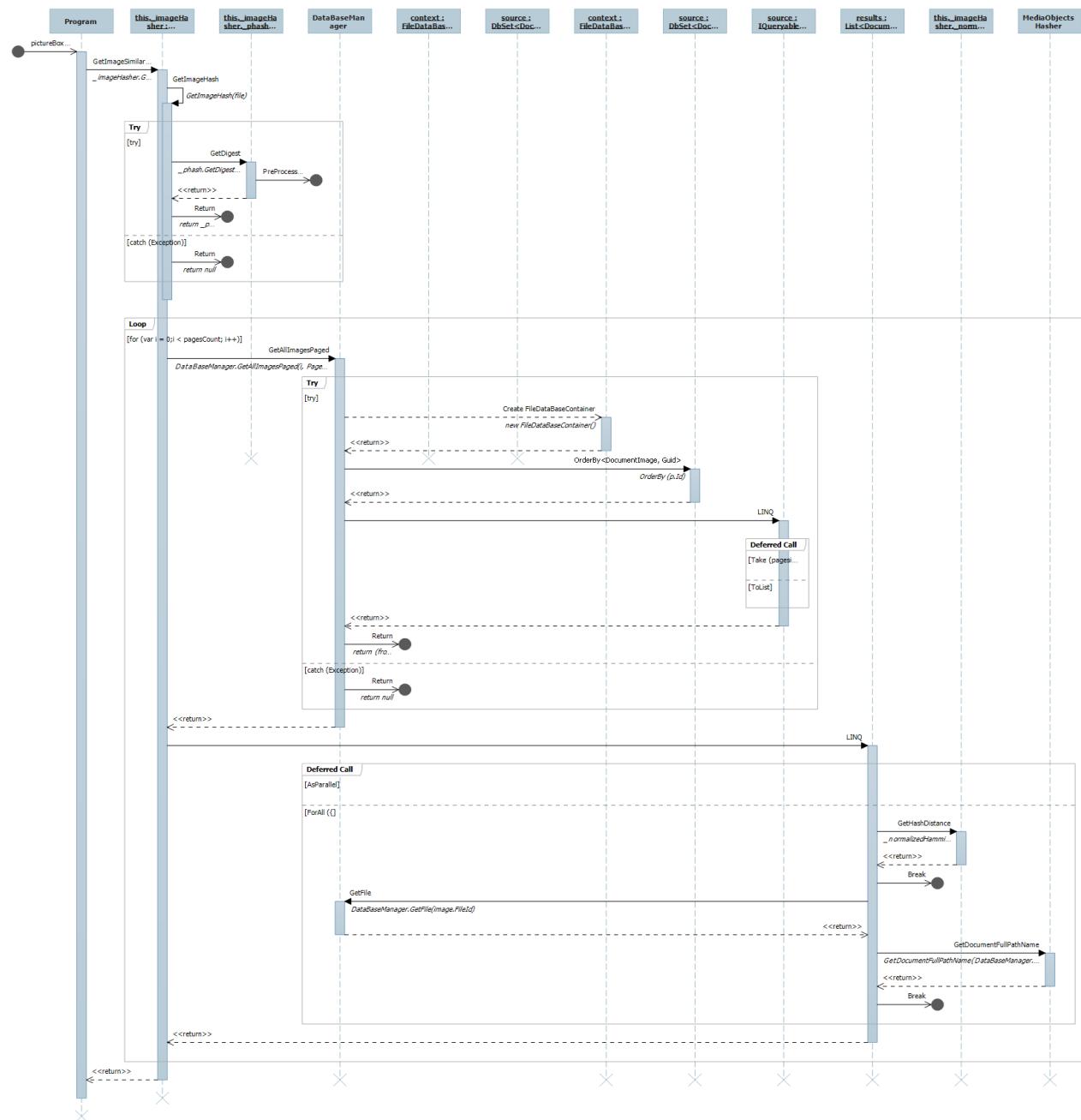


Ilustración 16 Búsqueda de imágenes

4.5 Diseño de pantallas

Se describe a continuación el diseño de las pantallas del sistema y su funcionalidad.

4.5.1 Pantalla HU1: Configuración Primera Ejecución

Objetivo

Permite seleccionar un directorio para indexación de archivos, cuando se ejecuta por primera vez la aplicación o después de que es reiniciada.

Diseño

La Ilustración 17 muestra la pantalla de configuración. Debajo del mensaje de información se encuentran los botones de:

- Seleccionar folder. Permite abrir un explorador de directorios para seleccionar una a indexar.
- Salir. Cierra la pantalla.

Esta pantalla se muestra cuando se ejecuta la primera vez la aplicación y después de que es reiniciada.

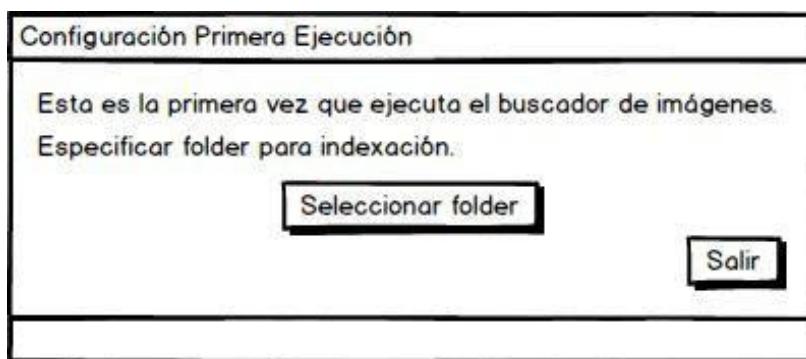


Ilustración 17 Primera Ejecución

Entradas

En esta pantalla el usuario selecciona la siguiente información:

- Directorio de indexación

Salidas

Se muestra automáticamente un mensaje informando que la indexación automática se está realizando. Como se muestra en la Ilustración 18

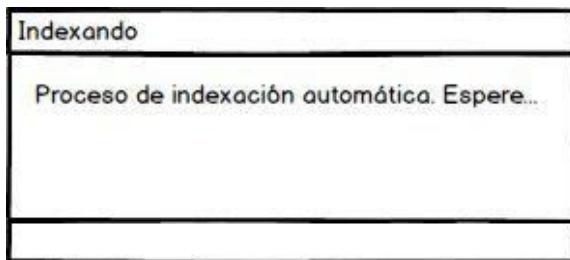


Ilustración 18 Indexado Automatico

Cuando se termina de hacer la indexación se muestra un mensaje informando que proceso de generación de firmas hash se está realizando. Como se muestra en la Ilustración 19

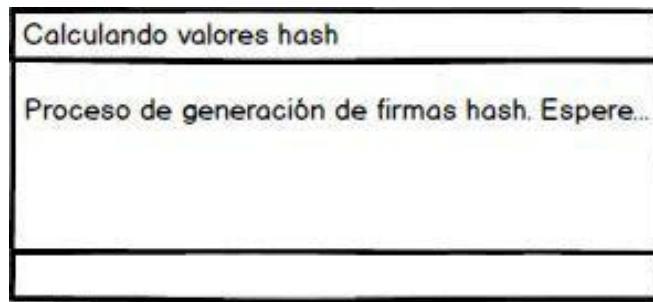


Ilustración 19 Generacion de firmas

Comandos

- **Seleccionar folder.** Permite abrir un explorador de directorios para seleccionar una a indexar.
- **Salir.** Cierra la pantalla.

4.5.2 Pantalla HU2: Cargar imagen

Objetivo

Permite cargar una imagen al sistema para poder hacer la búsqueda de imágenes similares.

Diseño

La Ilustración 20 muestra la pantalla principal donde se pueden cargar las imágenes para su posterior búsqueda. En ella se encuentran los botones de:

- **Cargar.** Abre un seleccionador de archivos para elegir la imagen a cargar.
- **Buscar.** Busca las imágenes similares a la imagen cargada.
- **Configuración de búsqueda.** Abre la pantalla Configuración de búsqueda de Ilustración 20

Para poder acceder a esta pantalla es necesario haber pasado por el proceso de indexación del directorio.

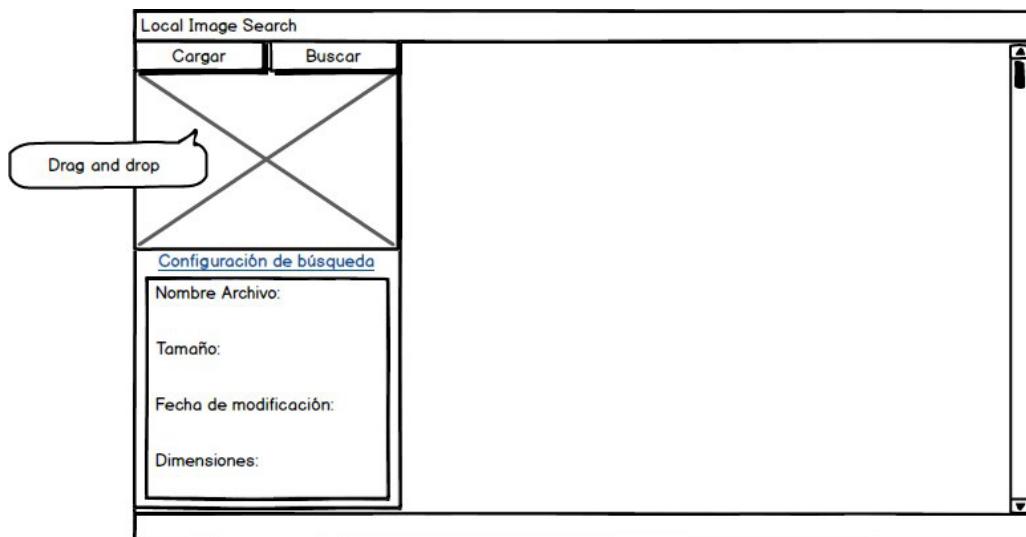


Ilustración 20 Pantalla principal

Entradas

- Control de carga de imágenes.

Salidas

Se mostrará la imagen seleccionada en el control de carga de imágenes.

Controles

- Control de carga de imágenes *Drag&Drop*. Permite arrastrar una imagen para cargarla al sistema y posteriormente realizar la búsqueda.

Comandos

- **Cargar.** Permite seleccionar una imagen de un directorio para cargarla al sistema.
- **Buscar.** Dispara la historia de usuario Buscar imagen en Ilustración 21
- **Configuración de búsqueda.** Abre la pantalla Configuración de búsqueda en Ilustración 22

4.5.3 Pantalla HU3: Buscar imágenes similares

Objetivo

Permite buscar las imágenes similares a la imagen cargada.

Diseño

La Ilustración 21 muestra la pantalla principal donde se pueden buscar las imágenes similares a la imagen cargada. En ella se encuentran los botones de:

- Cargar. Abre un seleccionador de archivos para elegir la imagen a cargar.
- Buscar. Busca las imágenes similares a la imagen cargada.
- Configuración de búsqueda. Abre la pantalla Configuración de búsqueda Ilustración 22

Para poder acceder a esta pantalla es necesario haber pasado por el proceso de indexación del directorio.

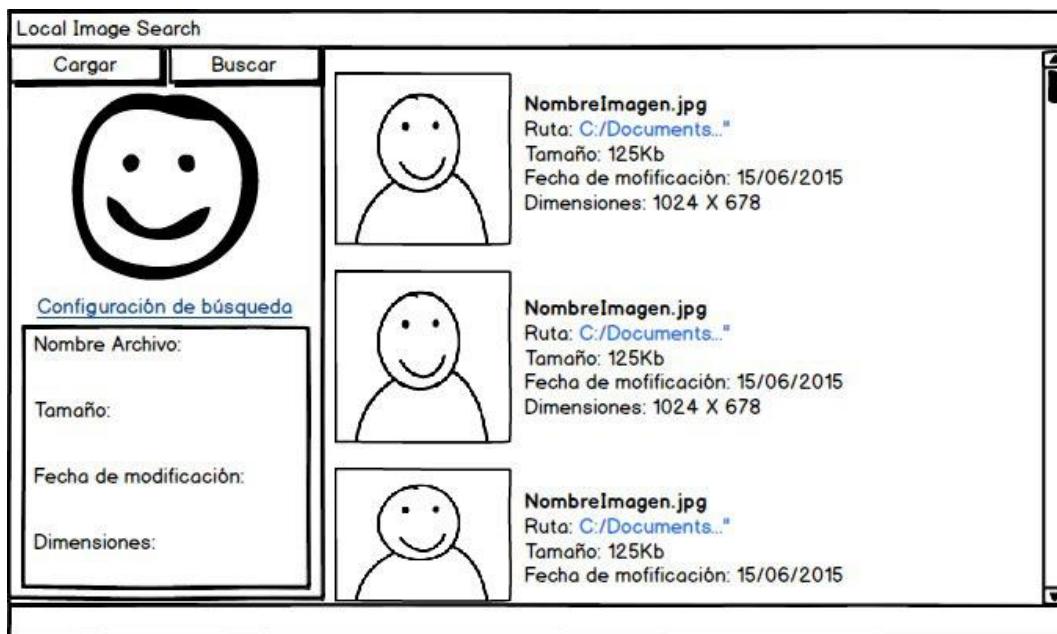


Ilustración 21 Realizacion de búsqueda

Entradas

- Control de carga de imágenes.

Salidas

Se mostrarán las imágenes similares encontradas con el método y porcentaje de similitud elegidos en la configuración de búsqueda.

Controles

- Control de carga de imágenes Drag&Drop. Permite arrastrar una imagen para cargarla al sistema y posteriormente realizar la búsqueda.

Comandos

- **Cargar.** Permite seleccionar una imagen de un directorio para cargarla al sistema.
- **Buscar.** Busca las imágenes similares a la imagen cargada con el método y porcentaje de similitud elegidas en la configuración de búsqueda.
- **Configuración de búsqueda.** Abre la pantalla Configuración de búsqueda Ilustración 22

4.5.4 Pantalla HU4: Configuración de búsqueda

Objetivo

Permite modificar la configuración de búsqueda.

Diseño

La Ilustración 22 muestra la pantalla donde se puede modificar la configuración de búsqueda. En ella se encuentran los botones de:

- Reiniciar Sistema. Reinicia el sistema completamente.
- Guardar. Guarda la configuración de búsqueda.
- Cancelar. Cierra la pantalla sin guardar los cambios.

Para poder acceder a esta pantalla es necesario seleccionar “Configuración de búsqueda” en la pantalla principal.

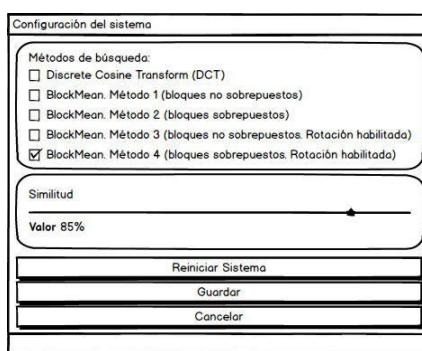


Ilustración 22 Configuración de sistema

Entradas

- Método de búsqueda.
- Porcentaje de similitud.

Salidas

Se guardará la configuración y se devolverá a la pantalla principal. Si se selecciona la opción de Reiniciar sistema, se mostrará un mensaje informando que se debe iniciar de nuevo la aplicación para que los cambios tengan efecto. Como se muestra en la Ilustración 23:

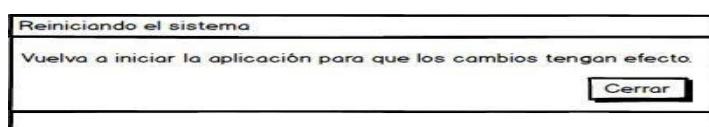


Ilustración 23 Re inicialización de sistema

Controles

- Grupo de cajas de selección “Métodos de búsqueda”. Permite seleccionar el método de búsqueda.
- Control de selección “Porcentaje de similitud”. Permite seleccionar el valor del porcentaje de similitud deseado en las búsquedas.

Comandos

- **Reiniciar sistema.** Cierra la aplicación y la reinicia para que se ejecute como si fuera la primera vez.
- **Guardar.** Almacena la configuración de búsqueda seleccionada.
- **Cancelar.** Cierra la pantalla sin almacenar los cambios.

CAPÍTULO 5

DESARROLLO

En este capítulo se muestra el desarrollo de software que se ha realizado para este trabajo terminal, dividido por historias de usuario, tal como se detallaron en el Capítulo 3.

Para el desarrollo de este sistema se utilizó el entorno de desarrollo Visual Studio 2012, que ayudó a codificar el mismo mediante el lenguaje de programación Visual C# y se pudo proveer al sistema de un interfaz gráfica de usuario, con el uso de la tecnología WPF (Windows Presentation Foundation).

5.1 PANTALLA DE INICIO

La pantalla de inicio del sistema cuando se ejecuta por primera vez contiene un selector de directorio para ingresar el directorio de búsqueda que se utilizará en el sistema.

En la Ilustración 24 Una vez seleccionado el directorio de búsqueda se indexan los archivos contenidos en el mismo y se muestra un mensaje como el siguiente

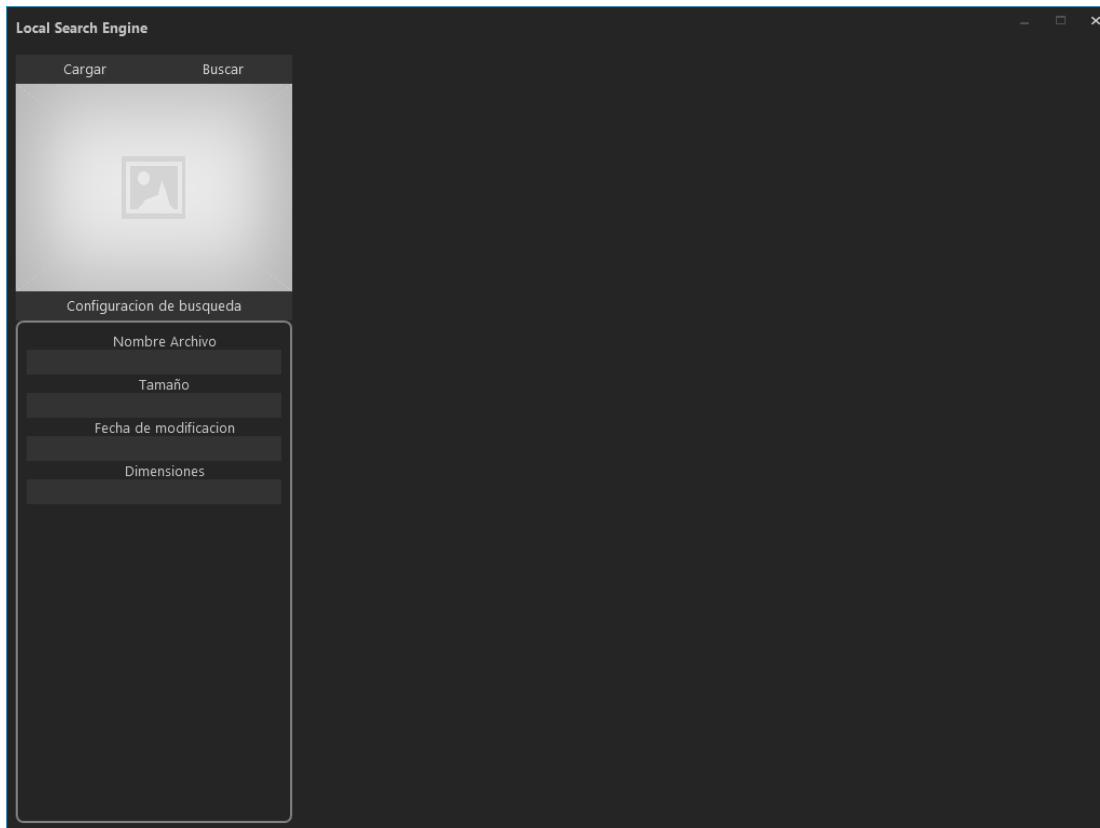


Ilustración 24 Pantalla Inicio Buscador

5.1.1 Indexación de archivos

El procedimiento de indexación definido en la Ilustración 25 de archivos se realiza de la siguiente manera:

1. Se escanea el directorio base buscando archivos compatibles dentro del mismo.
2. Se obtienen todos los subdirectorios dentro del directorio base y se agregan a una pila.
3. Se obtienen sus propiedades útiles de los archivos encontrados como nombre archivo, tamaño, fecha creación, etc.
4. Se almacenan en base de datos los archivos con su información.
5. Se saca un directorio de la pila y se vuelve al paso 1 como el directorio base, si no hay más archivos en pila se termina el proceso.

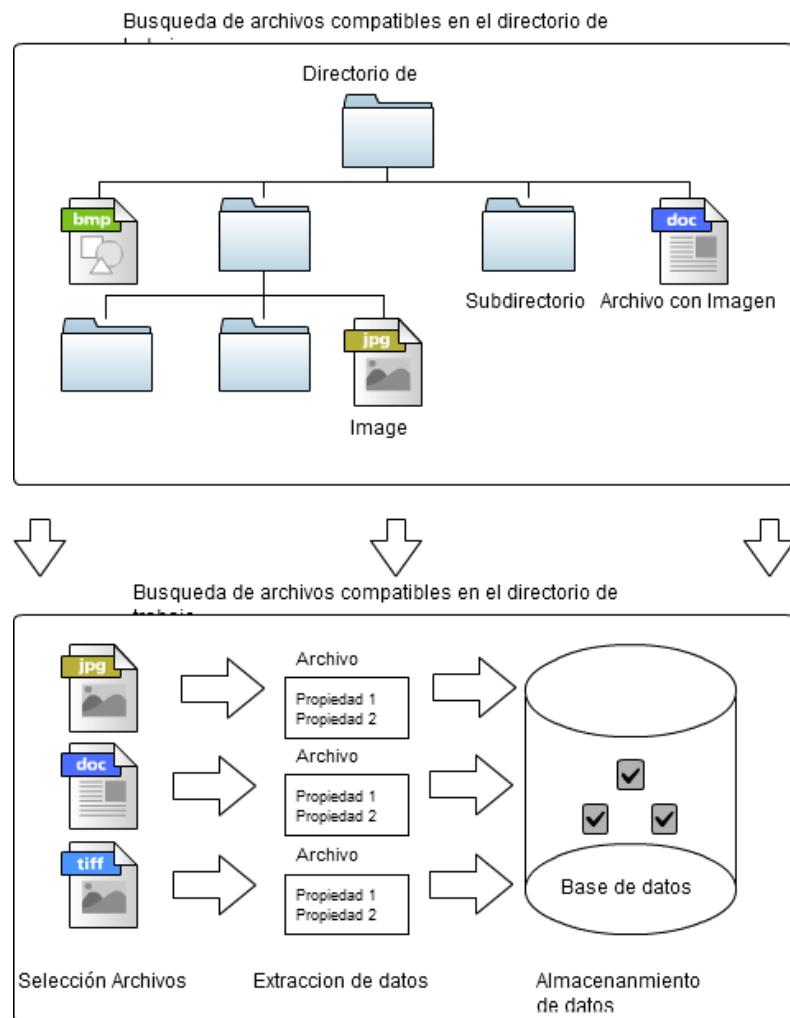


Ilustración 25 Indexación de archivos

5.1.2 Procesamiento de imágenes (generación firmas)

El procedimiento de generación de firmas para los archivos se realiza de la siguiente manera:

1. Se realiza una consulta a la base de datos con los elementos que no contienen información de firma
- 1.2. *Cada consulta a la base de datos se procesa en grupos de N elementos (1000 como ejemplo) donde un elemento x con su campo Firma es nulo (null).*
2. Por cada elemento en el grupo obtenido se realiza el cálculo de la firma.
3. Una vez obtenidas todas las firmas del grupo se almacena la información correspondiente en base de datos.
4. Se vuelve al paso 1 hasta que no exista ningún elemento más a procesar.

5.1.3 Generación de una firma por medio de la función DCT Perceptual Hash

La generación de una firma perceptual por medio de una función perceptual hash se muestra detalladamente en los siguientes pasos:

1. Obtener la imagen a procesar de tamaño $N \times M$
 - 1.1. Realizar un procesamiento de imagen a escala de grises
2. Realizar una reducción de imagen
 - 2.1. La reducción puede ser por medio del algoritmo [ResizeNearestNeighbor] el cual no procesa ninguna interpolación y reduce el tiempo de procesamiento.

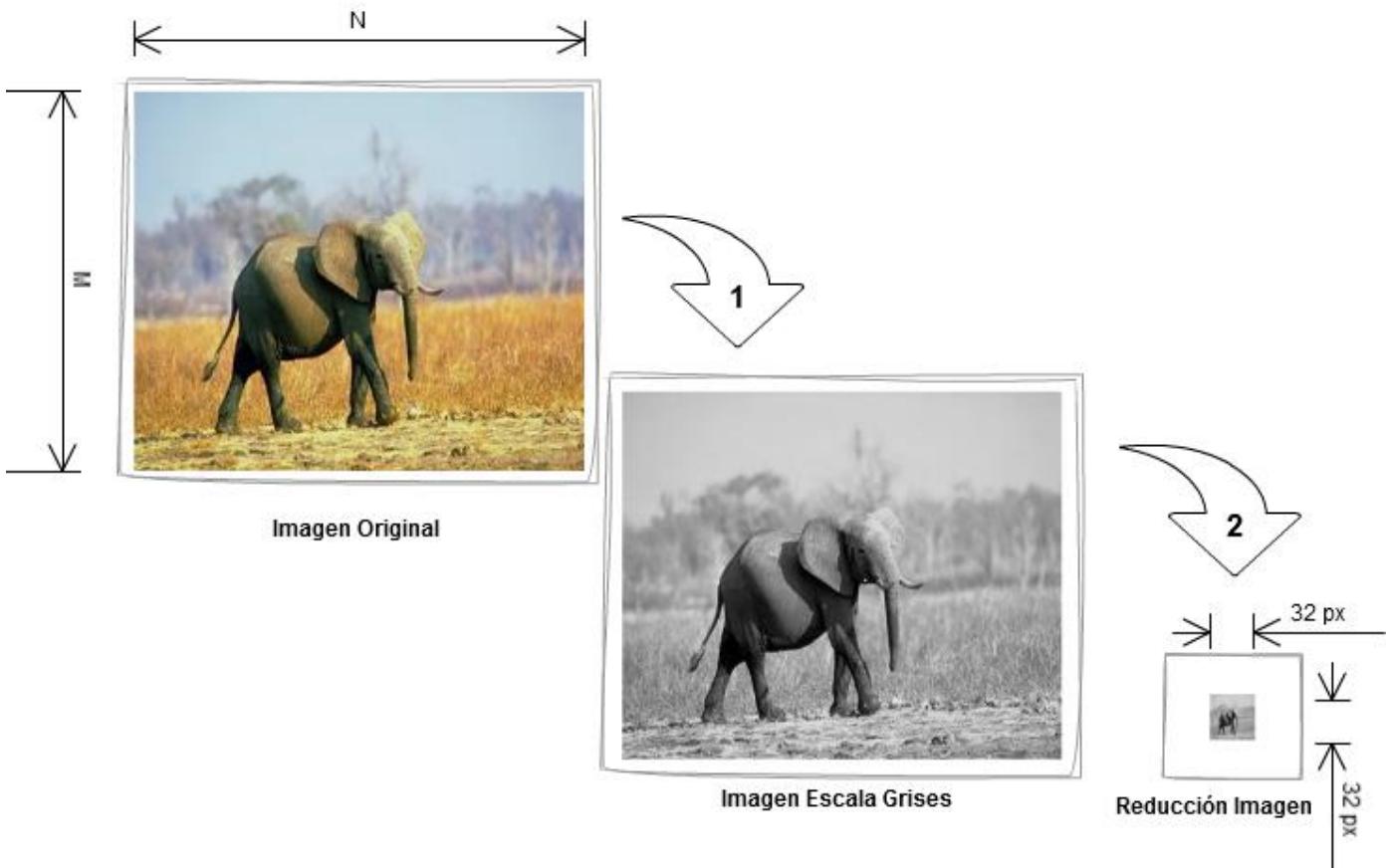


Ilustración 26 Pasos de pre procesamiento de imagen

3. Una vez reducida la imagen procedemos a aplicar la transformada DCT-II.
 - 3.1. *La imagen ya está reducida a dimensiones establecidas en 32x32.*

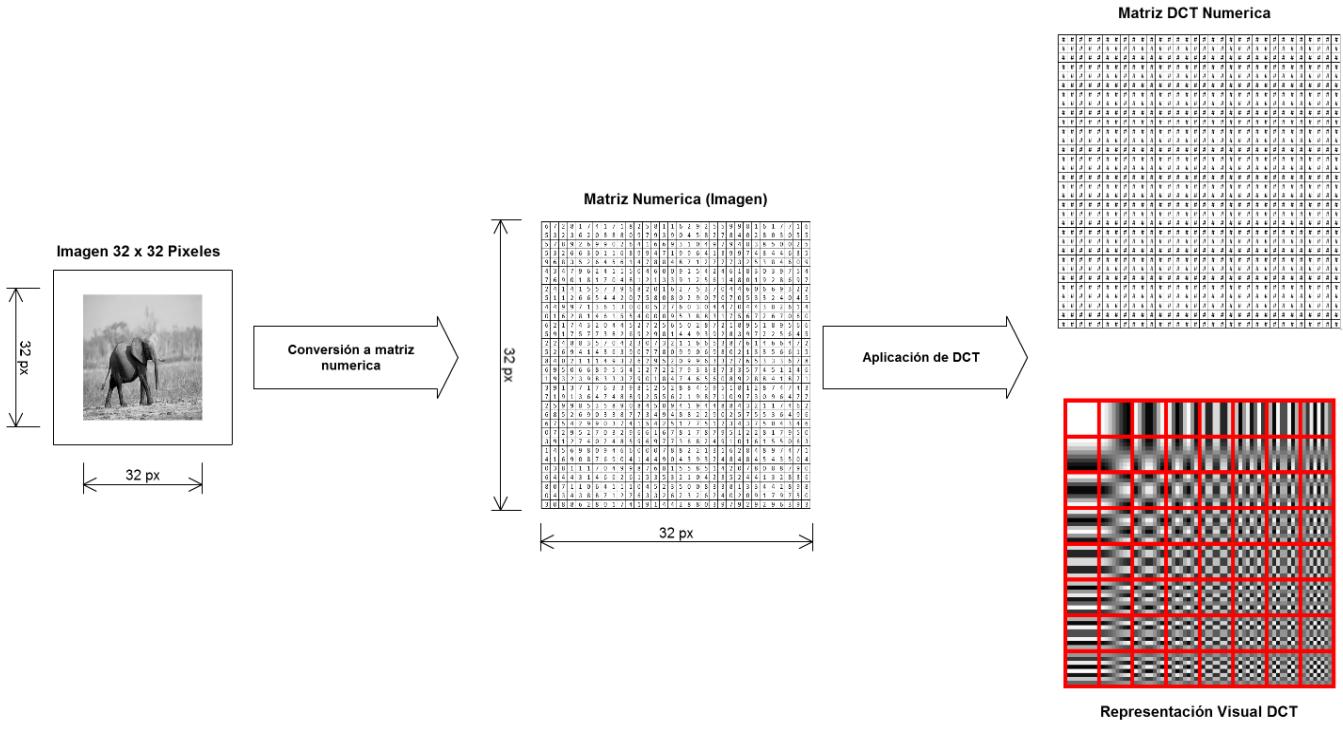


Ilustración 27 Generación de la DCT de una imagen

4. Se procede a calcular el promedio del primer cuadrante

$$4.1. \text{ El primer cuadrante } Q \text{ de } 8 \times 8 \text{ se define como } Q = \begin{bmatrix} B_{00} & \cdots & B_{70} \\ \vdots & \ddots & \vdots \\ B_{07} & \cdots & B_{77} \end{bmatrix}$$

4.2. Se ignora el valor Coeficiente Base DC

5. Realizar la generación de firma de la imagen

5.1. Considerando a T como el valor promedio, se procede a generar la firma de longitud 64, con los valores de Q.

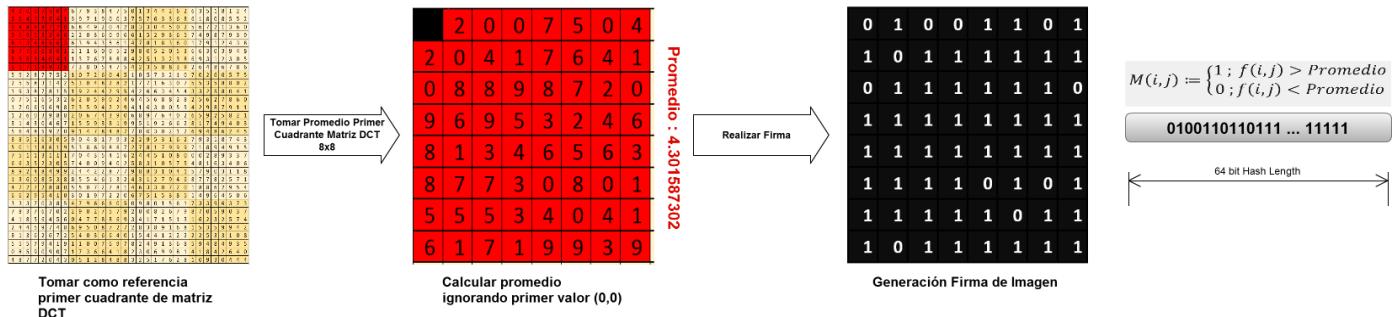


Ilustración 28 Generación de firma de la imagen

5.1.4 Generación de una firma por medio de la función Block Mean

La generación de firma por medio de Block mean se compone de 4 métodos los cuales se basan en el mismo principio mostrado aquí, estos están detallados en el apartado de marco teórico **Error! Reference source not found..**

1. Obtener la imagen a procesar de tamaño $N \times M$
 - 1.1. Realizar un procesamiento de imagen a escala de grises
2. Realizar una reducción de imagen
 - 2.1. La reducción puede ser por medio del algoritmo [ResizeNearestNeighbor] el cual no procesa ninguna interpolación y reduce el tiempo de procesamiento.

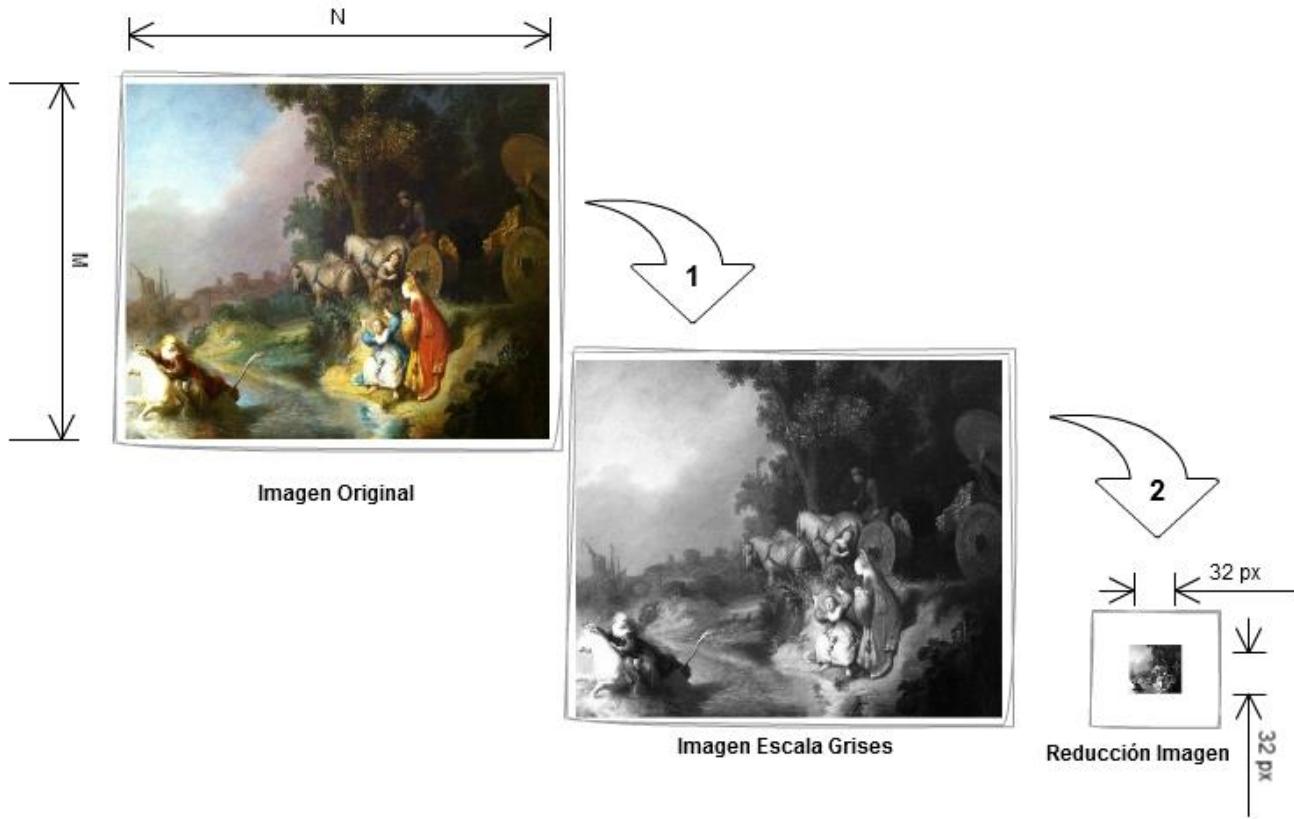


Ilustración 29 Pasos de reducción imagen

3. Se procede a dividir la imagen en bloques iguales de $m \times m$ por el método 1 (Block mean 1).

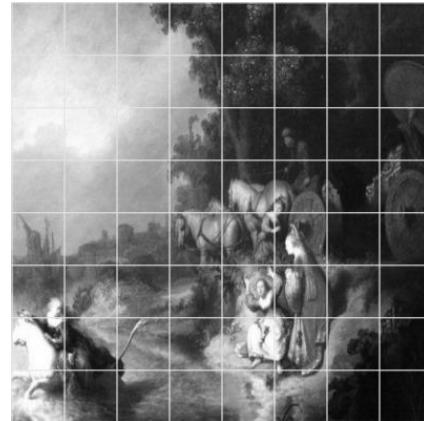


Ilustración 30 División por bloques

4. Se procede a calcular el valor promedio de cada bloque $M_{i,j}$ y el promedio de promedios M_{tot} como se indica.

$$M_{tot} = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} Mean(M_{i,j})}{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} M_{i,j}}$$

5. finalmente se genera la firma

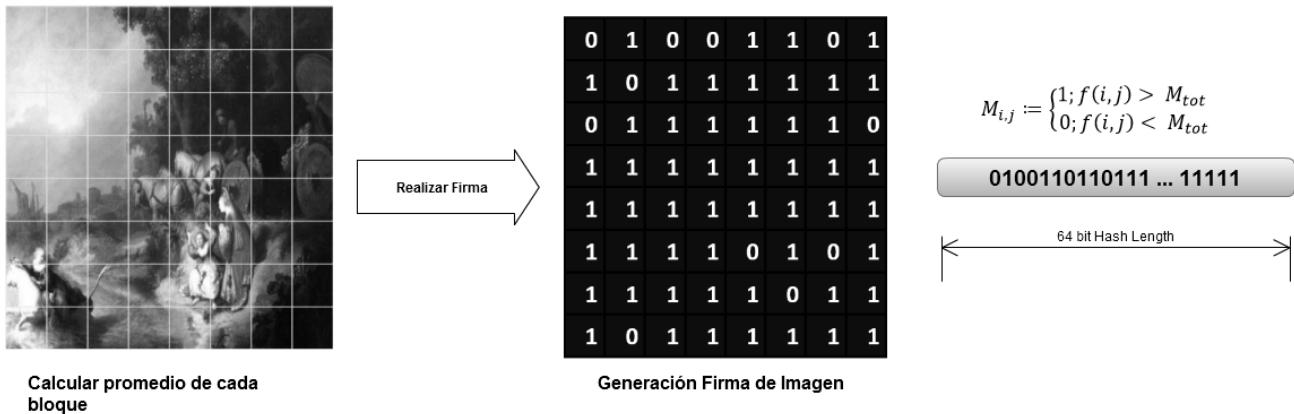


Ilustración 31 Generación de hash Block Mean

Mientras se realiza el cálculo de las firmas se muestra un mensaje como en la Ilustración 19 Generacion de firmas
Cuando ha finalizado el cálculo se muestra la pantalla principal de búsqueda.

5.2 PANTALLA PRINCIPAL DE BÚSQUEDA

La pantalla principal de búsqueda se puede observar de la Ilustración 24.

Se permite cargar una imagen a buscar a través del botón “Cargar” como se muestra en la Ilustración 24.

O arrastrando la imagen al contenedor de imagen de Ilustración 24.

Ya que la imagen está cargada se permite realizar la búsqueda de imágenes similares oprimiendo el botón “Buscar”, para ello se realiza el cálculo de similitud de firmas, descrito a continuación:

5.2.1 Procesamiento de imágenes

1. Se selecciona una imagen a buscar.
2. Se establece el valor de similitud S .
3. Se selecciona un método de búsqueda (selección de campo firma a comparar).
3.1. Basado en los tipos de funciones DCT, Block Mean métodos 1, 2, 3,4.
4. Se calcula la firma de la imagen seleccionada dando como resultado la firma α .
5. Se realiza una consulta a la base de datos con los elementos que contienen información de la firma especificada.
5.1. Cada consulta a la base de datos se procesa en grupos de N elementos (1000 como ejemplo) donde un cada elemento x contiene la firma β del tipo especificado en el punto 2.
6. Por cada elemento en el grupo obtenido se realiza el cálculo de distancia entre firma α y β donde el resultado es d .
6.1. $d = \text{distancia}(\alpha, \beta)$
7. Si d es mayor o igual al valor de similitud S especificado se agrega en lista de resultados.
8. Volver a paso 5 hasta que no haya más elementos por comparar.
9. Se regresa y muestra la lista de resultados.

5.2.2 Cálculo de similitud entre firmas

El procedimiento del cálculo de similitud o distancia entre firmas se describen en los siguientes pasos, necesarios para poder comparar los dos objetos multimedia en este caso imágenes:

1. Generar firma de imágenes A, B (descrito en *Generación de una firma por medio de la función DCT Perceptual Hash*).
2. Comparar distancia utilizando operación XOR.
3. Realizar la sumatoria en el resultado.
4. Realizar normalización $\frac{1}{n}$, donde n es la longitud de la cadena.
5. Acotar la similitud por $d = 1 - s$

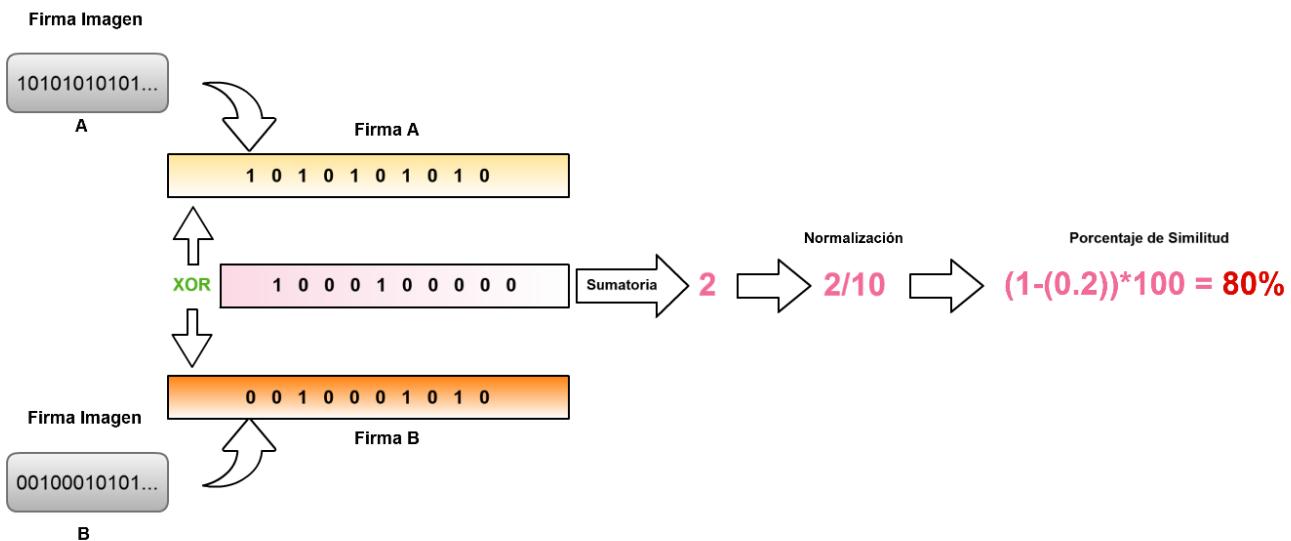


Ilustración 32 Generación de distancia entre firmas

Una vez realizada la comparación se muestran los resultados como se puede observar en la Ilustración 38 del apartado de pruebas.

CAPÍTULO 6

PRUEBAS Y RESULTADOS

6.1 RESULTADOS OBTENIDOS

Especificaciones de prueba

Las pruebas fueron realizadas en un entorno con un conjunto de imágenes indexadas en el sistema, a continuación se muestran los resultados obtenidos en pruebas ocupando las diferentes representaciones de firmas según la función perceptual utilizada.

Parámetros de pruebas

El único parámetro en estas pruebas es el valor porcentaje similitud en el sistema, puesto en un valor de 85%, los métodos de búsqueda y sus resultados quedan documentados con sus respectivos resultados.

Vista previa a una carpeta en el sistema de indexado en pruebas

En la Ilustración 33 mostramos específicamente el álbum Caltec999 con distintas imágenes que servirán para realizar distintas pruebas.

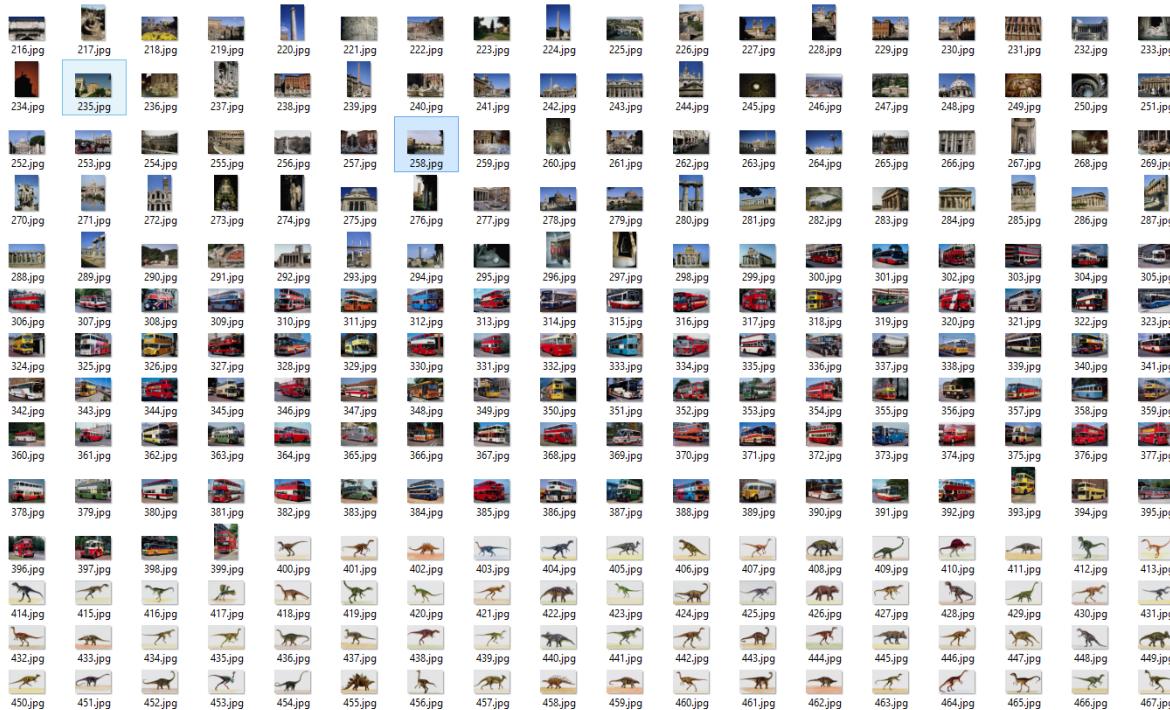


Ilustración 33 Vista a conjunto de biblioteca

Imágenes dinosaurios

En la Ilustración 34 se muestran distintas imágenes de dinosaurios, que se tomarán como marco de referencia para obtener resultados de la búsqueda de una imagen con un dinosaurio. Ilustración 34 Imágenes solo dinosaurios

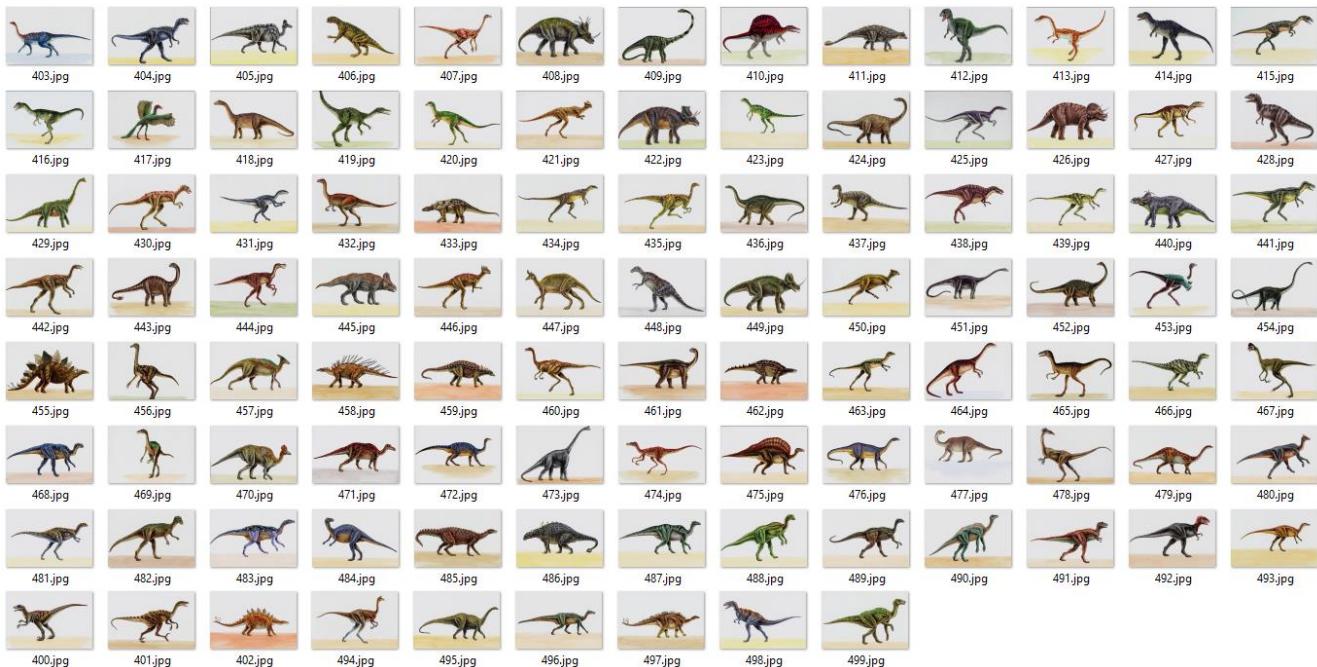


Ilustración 34 Imágenes solo dinosaurios

Imágenes fotografías de niña

En la Ilustración 35 se muestran distintas imágenes de una niña en distintas poses, que se tomarán como marco de referencia para obtener resultados de la búsqueda de una imagen con una niña. Ilustración 34 Imágenes solo dinosaurios

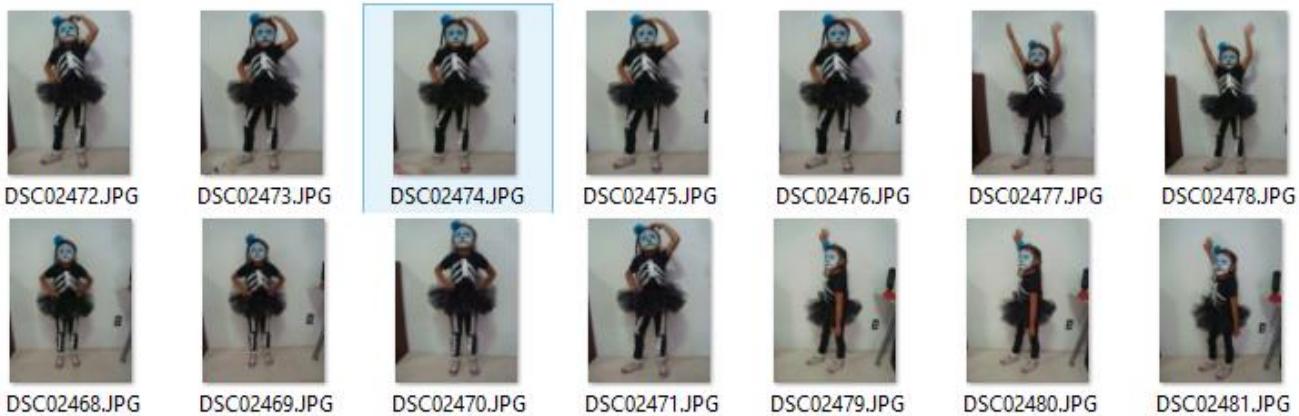


Ilustración 35 Fotos de niña

Imágenes fotografías de escenas

En la Ilustración 36 se muestran distintas imágenes de dos personas en una escena, que se tomarán como marco de referencia para obtener resultados de la búsqueda de una imagen con las personas en diferentes posiciones. Ilustración 34 Imágenes solo dinosaurios



Ilustración 36 Fotografía escena

Imágenes dentro de documentos

En la Ilustración 37 encontramos todos los documentos empleados en esta prueba, cada uno de ellos contiene las imágenes mostradas en el mismo directorio.

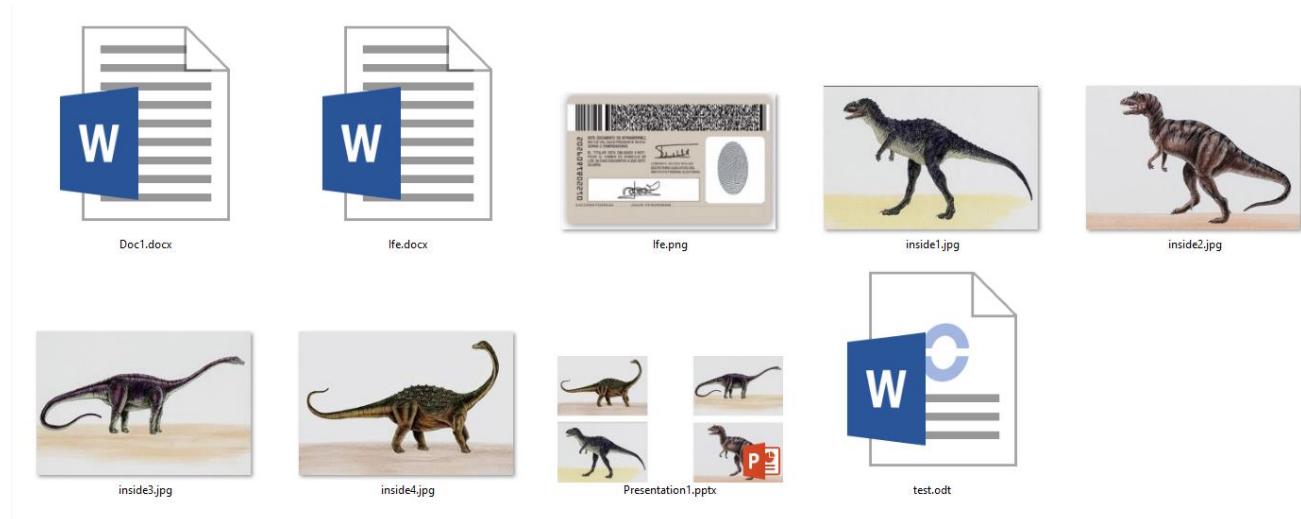


Ilustración 37 Imágenes dentro de documentos

PRUEBAS DINOSAURIOS

Para esta prueba se realiza la carga de una imagen de un dinosaurio.

Los resultados con la función DCT son los que se muestran en la Ilustración 38.

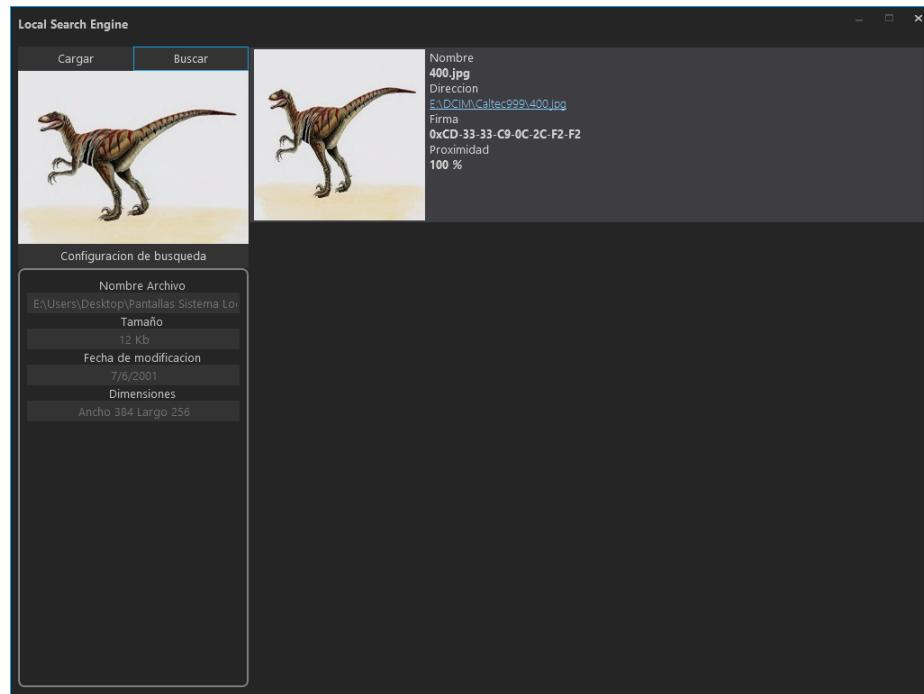


Ilustración 38 Dinosaurios DCT

Los resultados con la función Block Mean, método 1 son los que se muestran en la Ilustración 39.

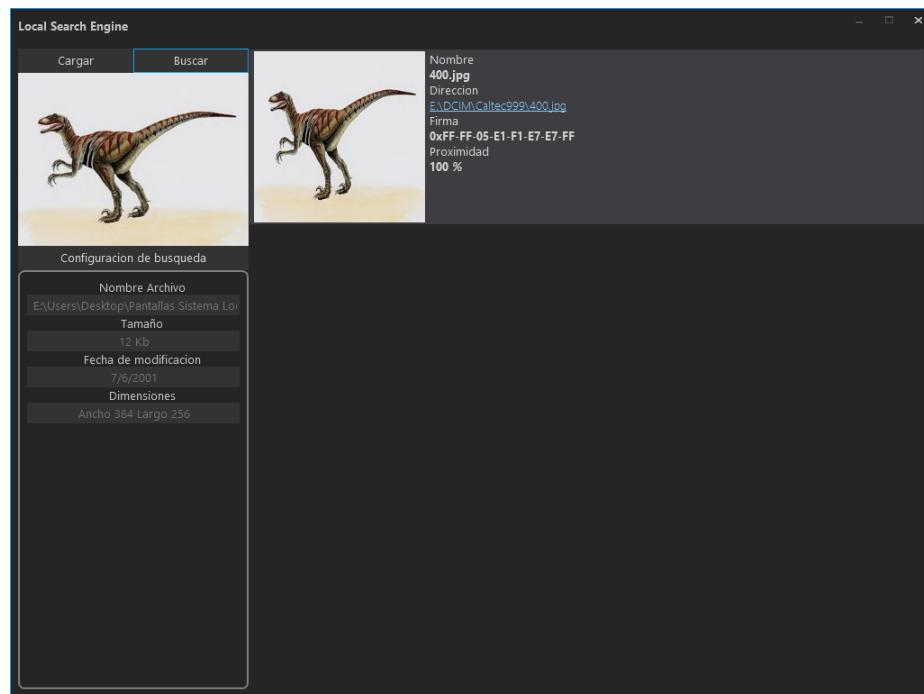


Ilustración 39 Dinosaurios Block Mean 1

Los resultados con la función Block Mean, método 2 son los que se muestran en la Ilustración 40, en este caso existen más resultados con su respectivo índice de similitud al 87%.

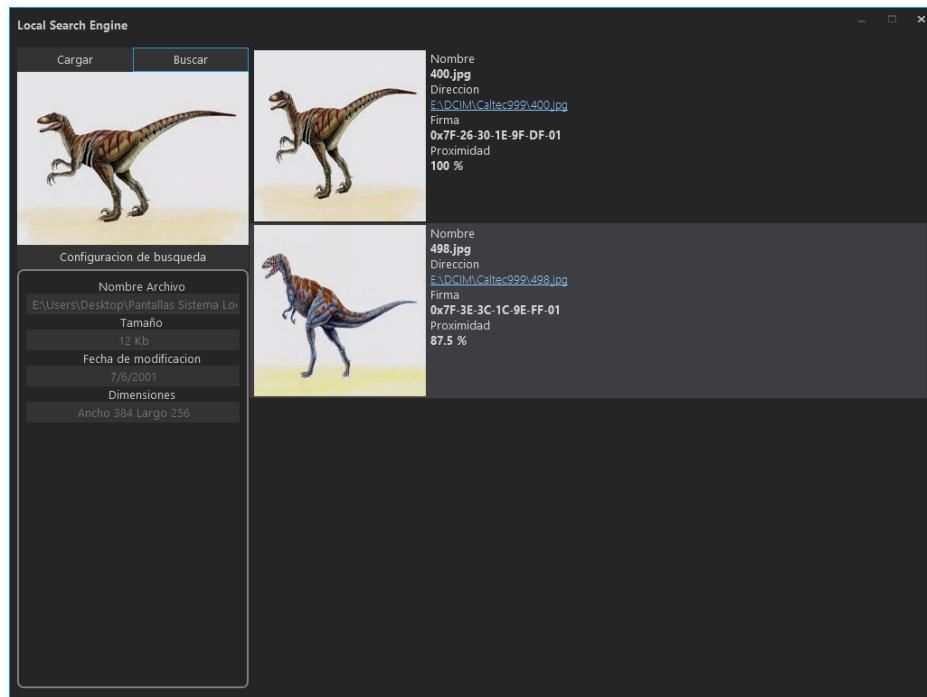


Ilustración 40 Dinosaurios Block Mean 2

Los resultados con la función Block Mean, método 3 son los que se muestran en la Ilustración 41, para esta prueba se encontraron más resultados a partir de 86%.

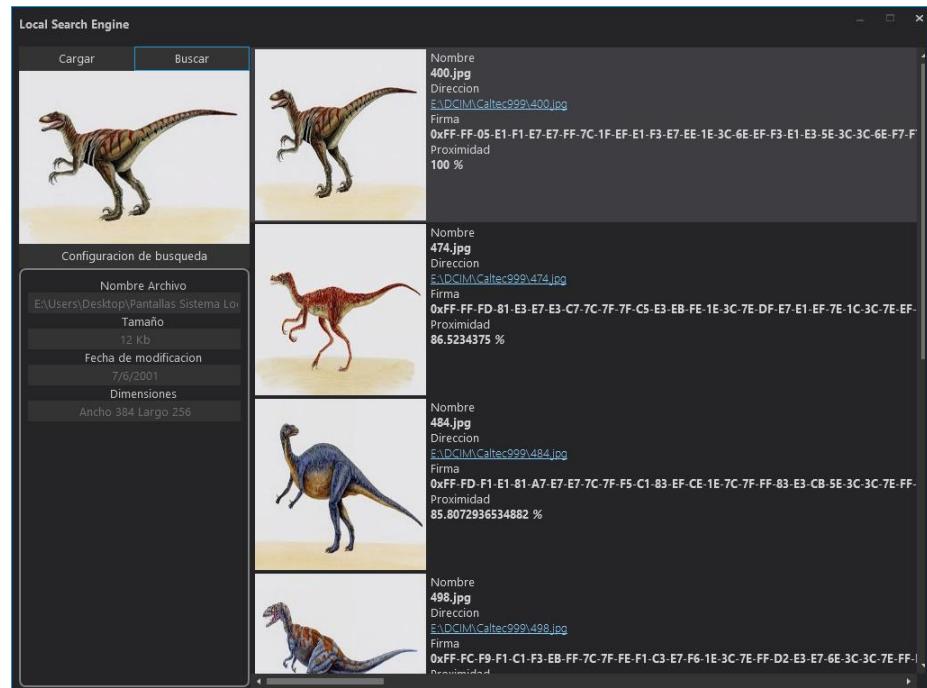


Ilustración 41 Dinosaurios Block Mean 3

Los resultados con la función Block Mean, método 4 son los que se muestran en la Ilustración 42, para esta prueba se encontraron más resultados a partir de 89%.

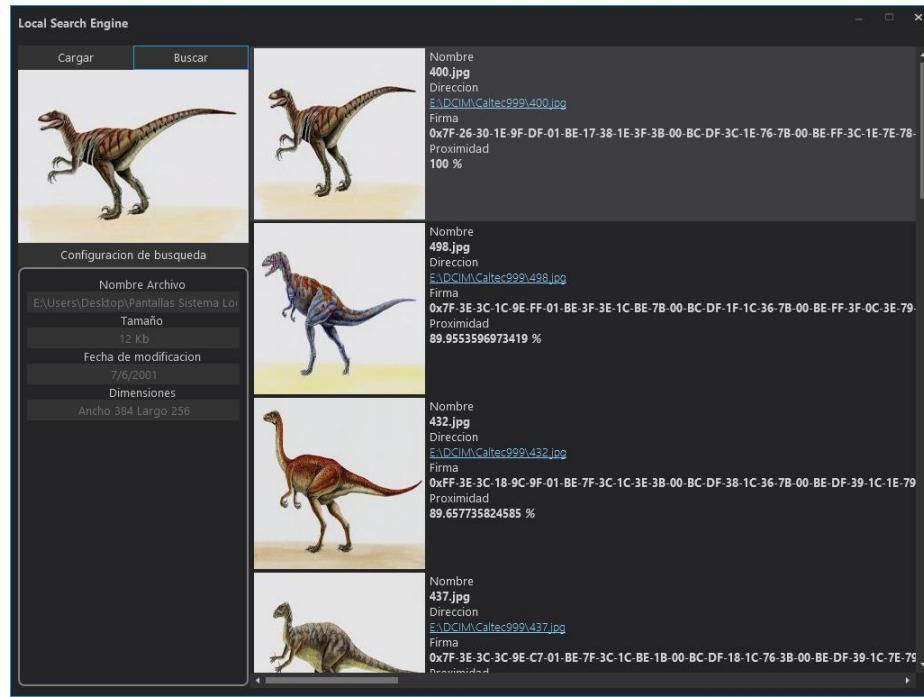


Ilustración 42 Dinosaurios Block Mean 4

PRUEBAS NIÑA

Para esta prueba se realiza la carga de una imagen de una fotografía con una niña.

Los resultados con la función DCT son los que se muestran en la Ilustración 43 donde se encuentra otra imagen en similitud de 85%.

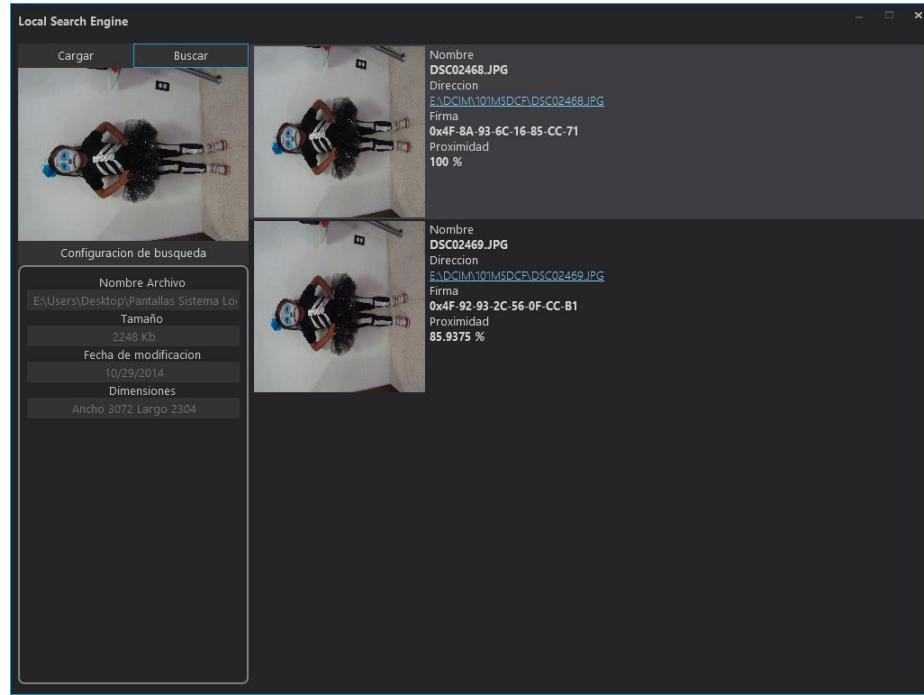


Ilustración 43 Niña DCT

Los resultados con la función Block Mean, método 1 son los que se muestran en la Ilustración 44, con resultados desde el 87%

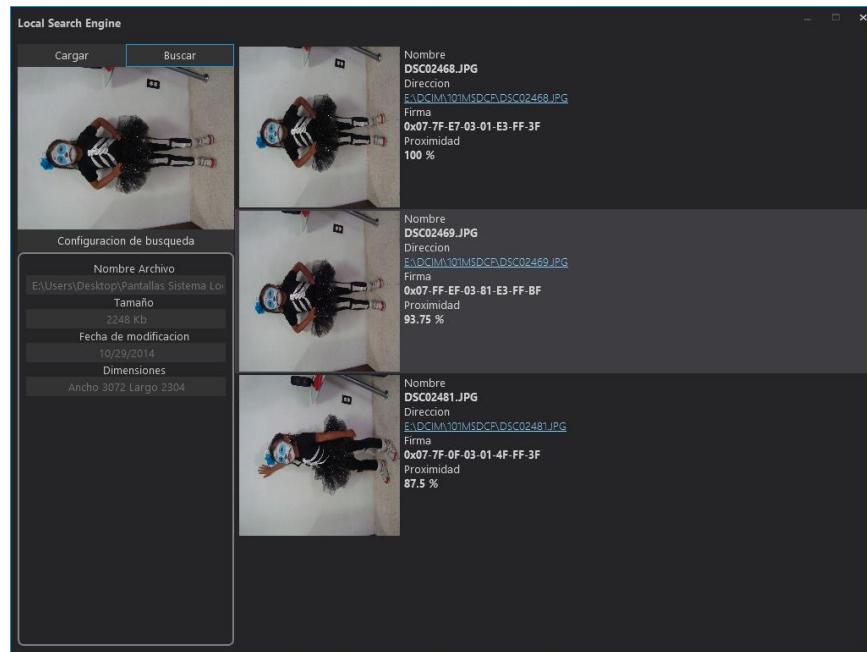


Ilustración 44 Niña Block Mean 1

Los resultados con la función Block Mean, método 2 son los que se muestran en la Ilustración 44, con resultados similares perceptualmente desde el 94%, obteniendo resultados visibles fuera de contexto a partir de 89%.

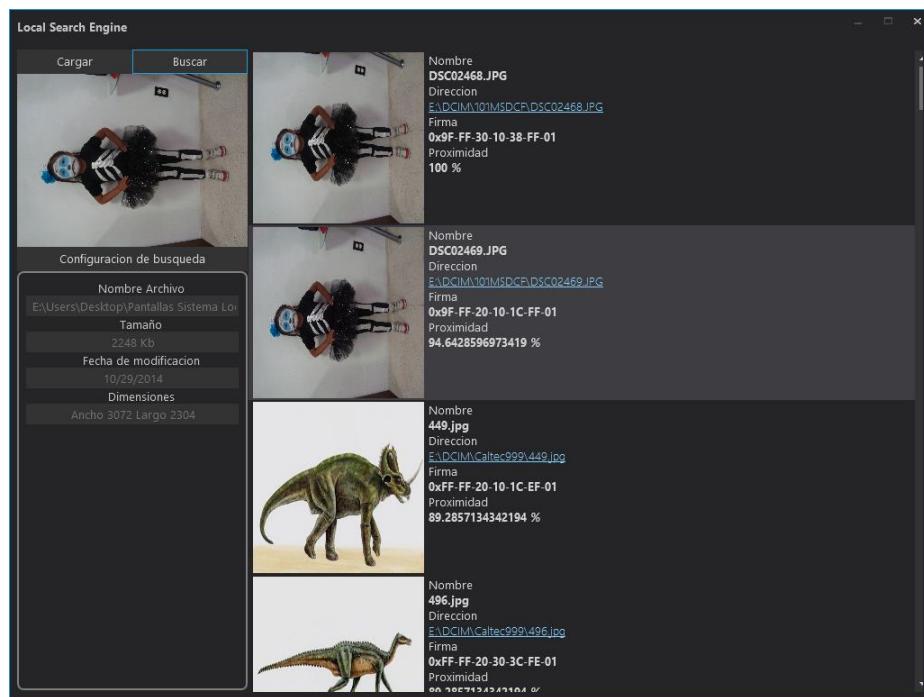


Ilustración 45 Niña Block Mean 2

Los resultados con la función Block Mean, método 3 son los que se muestran en la Ilustración 46, con resultados similares desde el 85%.

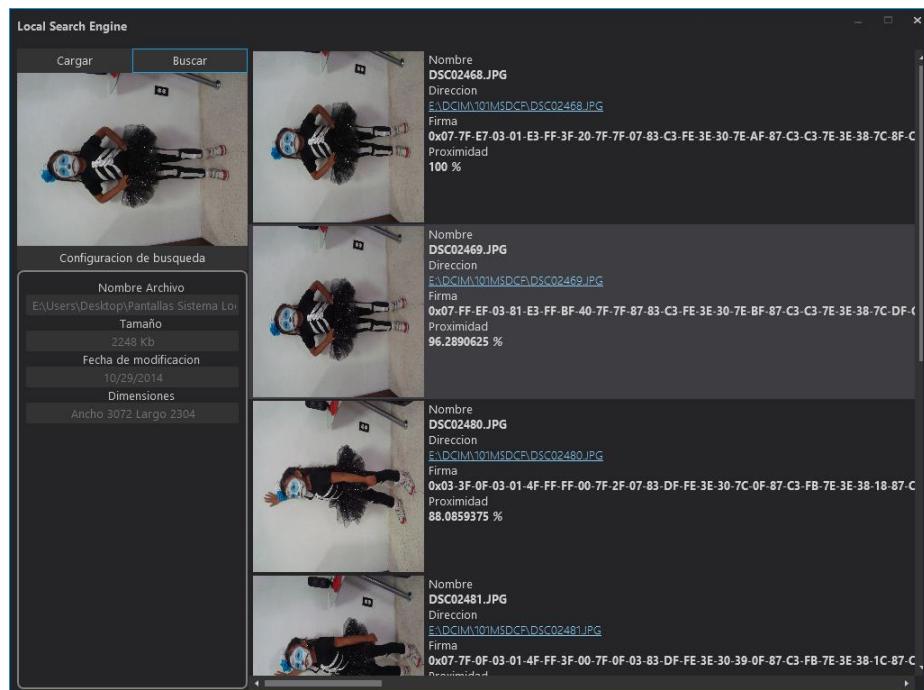


Ilustración 46 Niña Block Mean 3

Los resultados con la función Block Mean, método 4 son los que se muestran en la Ilustración 47, con resultados perceptualmente similares desde el 89%, y resultados fuera de contexto desde 87%.

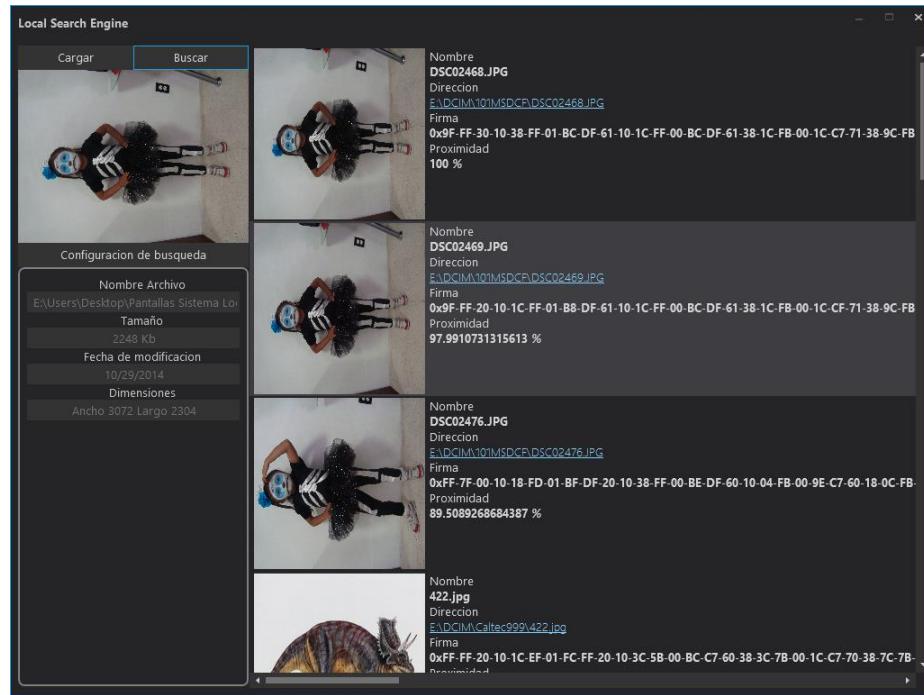


Ilustración 47 Niña Block Mean 4

PRUEBAS FOTOGRAFIA ESCENA

Para esta prueba se realiza la carga de una imagen de una fotografía con dos personas en una escena.

Los resultados con la función DCT son los que se muestran en la Ilustración 48 donde se encuentran imágenes perceptuales similares desde el 90%.

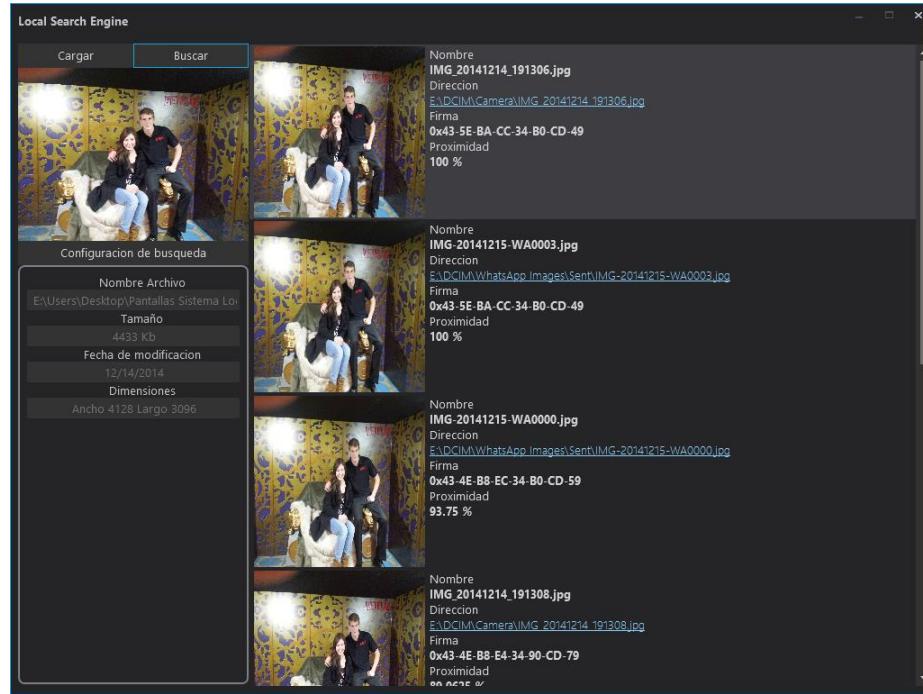


Ilustración 48 Escena DCT

Los resultados con la función Block Mean, método 1 son los que se muestran en la Ilustración 49, con resultados similares desde el 93%.

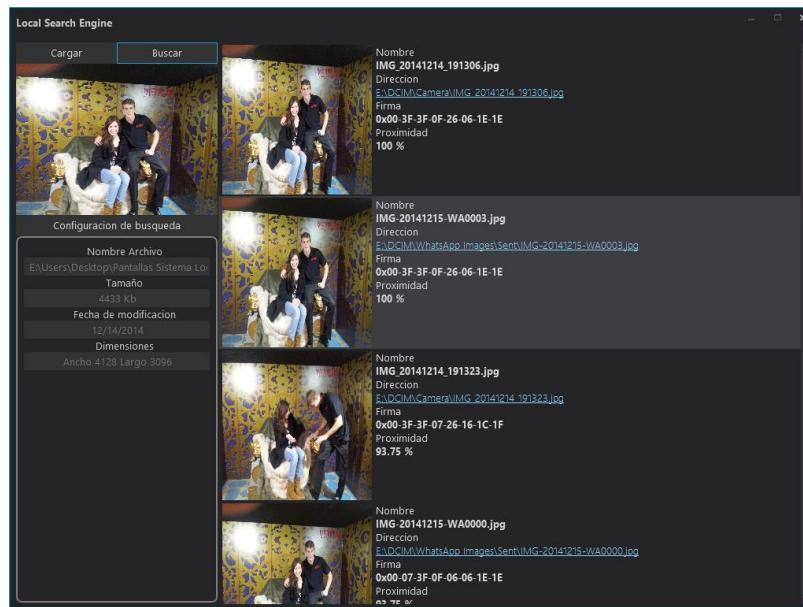


Ilustración 49 Escena Block Mean 1

Los resultados con la función Block Mean, método 2 son los que se muestran en la Ilustración 50, con resultados similares desde el 96%.

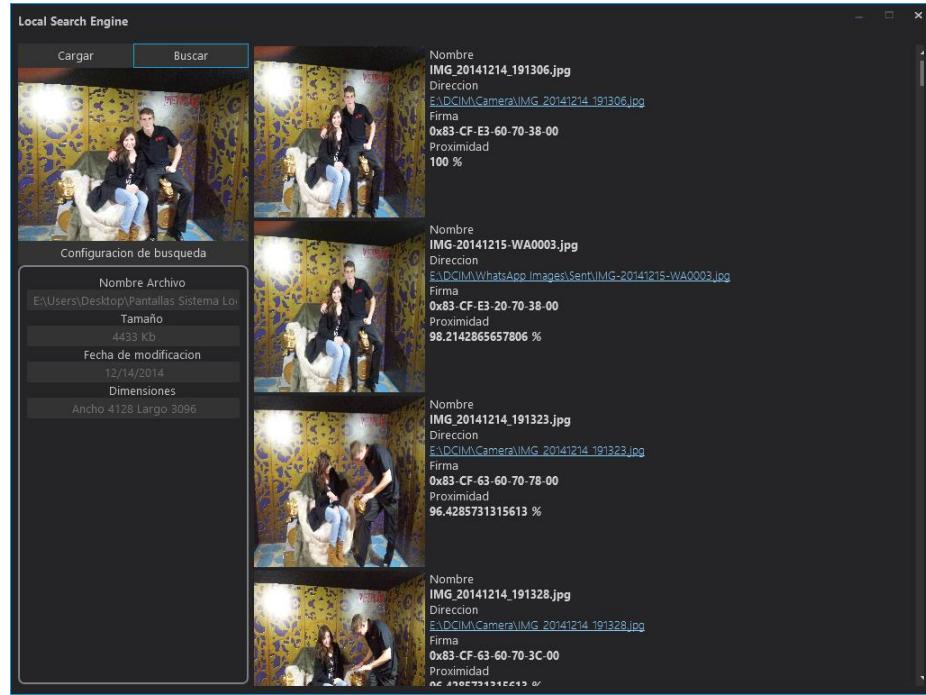


Ilustración 50 Escena Block Mean 2

Los resultados con la función Block Mean, método 3 son los que se muestran en la Ilustración 51, con resultados similares desde el 90%.

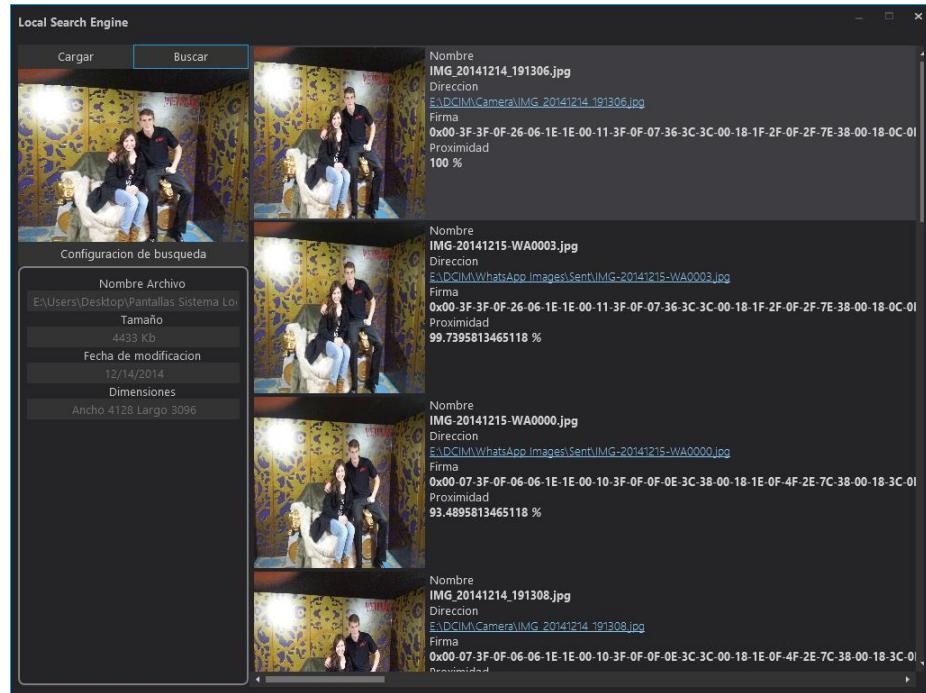


Ilustración 51 Escena Block Mean 3

Los resultados con la función Block Mean, método 4 son los que se muestran en la Ilustración 52, con resultados similares desde el 93%.

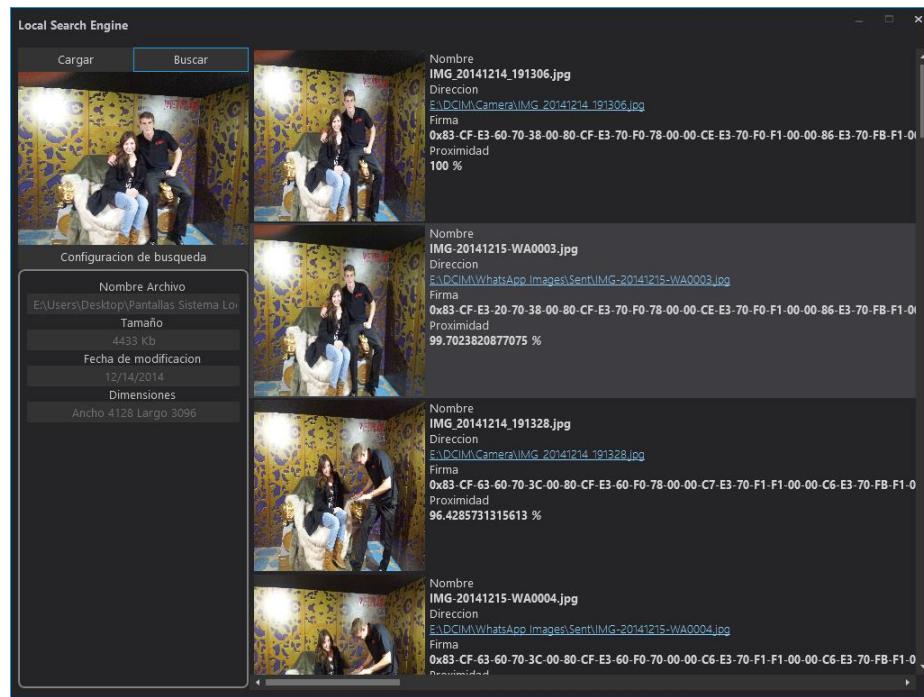


Ilustración 52 Escena Block Mean 4

Imágenes dentro de archivos

En la siguiente prueba se hizo una búsqueda de imagen, sabiendo que existen documentos que contienen estas imágenes, en este caso se muestra la similitud de la imagen contenida en los documentos Word® (.Docx) y Powerpoint® (.Pptx) del 100%.

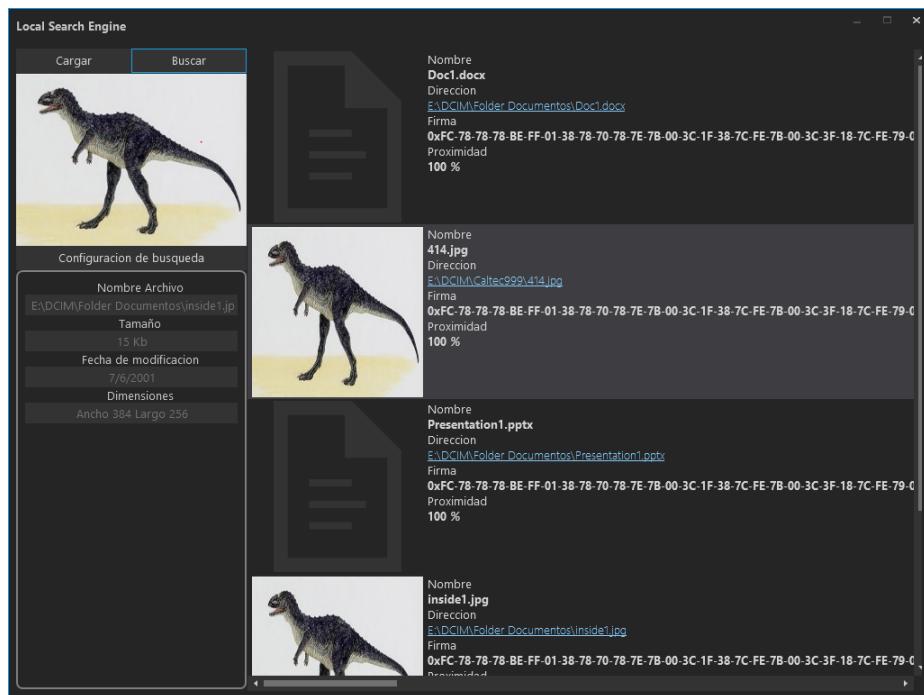


Tabla 13 Imágenes dentro de archivos

6.2 PRUEBAS DE RENDIMIENTO

Especificaciones de pruebas

En las siguientes pruebas de desempeño se realizaran en un equipo con las siguientes características mostradas en la Tabla 14

Elemento	Nombre	Características
Tarjeta Madre	<i>Elitegroup ECS A790GXM-A</i>	<i>Interfaz CPU 3.0 HyperTransport de alto rendimiento.</i> <i>Tasa máxima de transferencia de 5200 mbps</i>
Procesador	<i>Procesador AMD Phenom II X4 965</i>	<i>3.2GHz Frecuencia de procesador</i> <i>4 núcleos físicos</i> <i>6MB Tamaño Cache L3</i>
Memoria	<i>Memoria genérica Kingston DDR2</i>	<i>2x4Gb Slots</i> <i>8Gb Memoria</i>

HDD	<i>Western Digital</i>	<i>Interfaz SATA 6.0Gb/s</i>
	<i>WD10EZEX-22BN5A0</i>	<i>Capacidad 1TB</i>
		<i>RPM 7200 RPM</i>
		<i>Cache 64MB</i>

Tabla 14 Pruebas de rendimiento

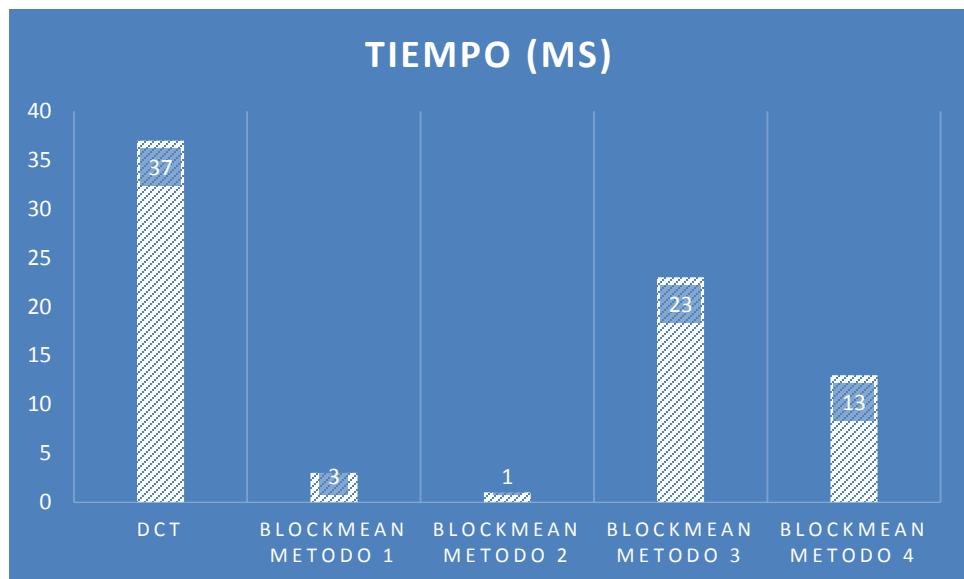
Las características de la imagen en la cual se extraen datos encontradas en la Tabla 15.

Propiedad	Valor
Tamaño	2.94 Mb
Formato	JPG
Dimensiones	3840 x 2160 pixeles
Profundidad de color (bit depth)	24 bits
Representación color	sRGB

Tabla 15 Especificaciones de imagen en pruebas de rendimiento

Análisis de rendimiento

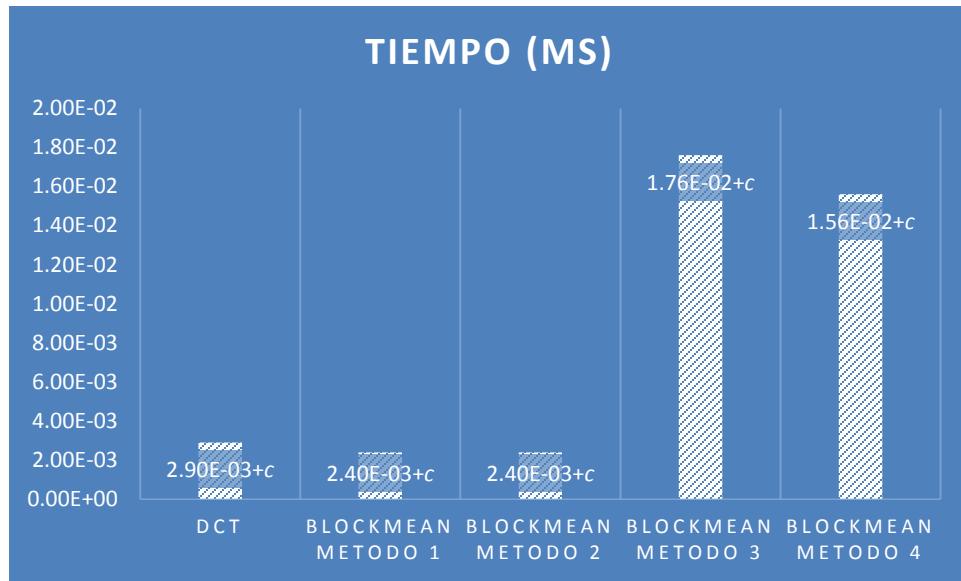
RESULTADOS DE GENERACIÓN DE FIRMA



Gráfica 1 Rendimiento generación de firma

Se puede observar en la Gráfica 1 que para la generación de las 5 firmas representativas de la imagen es la suma total de estos tiempos resultando un total de 77ms por cada imagen.

RESULTADOS COMPARACIÓN DE DOS FIRMAS ENTRE IMÁGENES



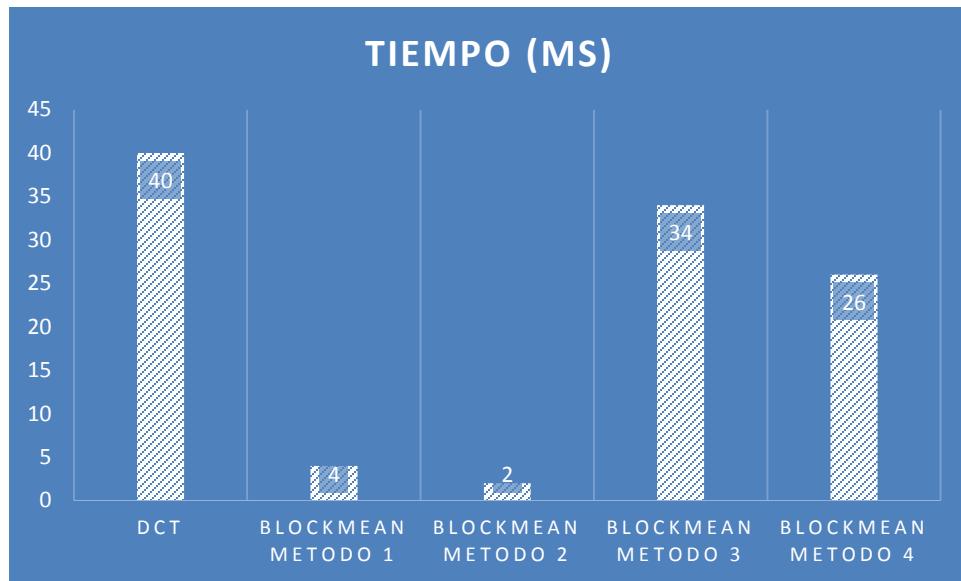
Gráfica 2 Rendimiento de comparacion de firmas de dos imágenes

Cabe destacar que los cálculos de distancias entre dos imágenes es realizado en un tiempo muy pequeño, por ello se propone como una constante de tiempo c la cual muestra el tiempo tomado de obtención de ambas firmas de las imágenes almacenadas en base de datos (imágenes indexadas).

Como se puede apreciar el tiempo de comparación aumenta debido a la longitud de las firmas de cada imagen.

RESULTADOS COMPARACIÓN DE DOS IMÁGENES (PROCESO SIN INDEXACIÓN)

Este resultado es conseguido al obtener la distancia de dos imágenes (*ambas con las mismas características mencionadas*) no indexadas; es decir pasando desde generación de firma hasta el cálculo de similitud.



Gráfica 3 Rendimiento de compracion de dos imágenes sin firma previa

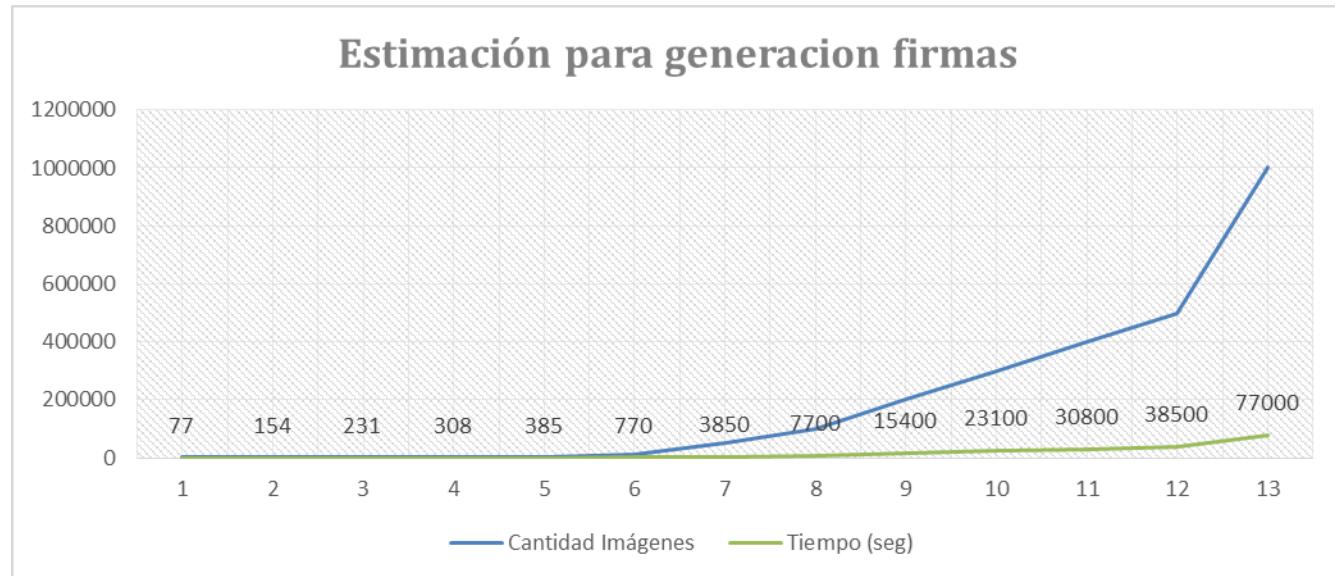
ESTIMACIÓN DE RENDIMIENTO EN GENERACIÓN DE FIRMAS

En la Tabla 16 se muestra la estimación de tiempo tomada para la generación de firmas desde diferentes escalas desde minutos, horas y segundos.

Cantidad Imágenes	Tiempo (seg)	Minutos	Horas
1000	77	1.2833333	0.021389
2000	154	2.5666667	0.042778
3000	231	3.85	0.064167
4000	308	5.1333333	0.085556
5000	385	6.4166667	0.106944
10000	770	12.833333	0.213889
50000	3850	64.166667	1.069444
100000	7700	128.33333	2.138889
200000	15400	256.66667	4.277778
300000	23100	385	6.416667
400000	30800	513.33333	8.555556
500000	38500	641.66667	10.69444
1000000	77000	1283.3333	21.38889

Tabla 16 Estimación de rendimiento en generacion de firmas

En la Gráfica 4 se muestra el tiempo requerido en segundos, para el procesamiento de firmas de imágenes en diferentes cantidades desde mil hasta un millón de imágenes.



Gráfica 4 Estimacion de generación de firmas

ESTIMACIÓN DE RENDIMIENTO EN BÚSQUEDA DE IMÁGENES SIMILARES (INDEXADAS)

En este proceso se analiza el tiempo estimado para la búsqueda de imágenes similares ya indexadas en el sistema, obtenido de la Gráfica 2, calculado el tiempo $c = 0.005 \text{ ms}$ representando la obtención de los datos indexados en el motor de base de datos para ello se muestra la comparativa de los 5 métodos.

Cantidad Imágenes	Tiempo (seg)	Minutos	Horas
1000	0.0079	0.0001317	2.19E-06
2000	0.0158	0.0002633	4.39E-06
3000	0.0237	0.000395	6.58E-06
4000	0.0316	0.0005267	8.78E-06
5000	0.0395	0.0006583	1.1E-05
10000	0.079	0.0013167	2.19E-05
50000	0.395	0.0065833	0.00011
100000	0.79	0.0131667	0.000219
200000	1.58	0.0263333	0.000439
300000	2.37	0.0395	0.000658
400000	3.16	0.0526667	0.000878
500000	3.95	0.0658333	0.001097
1000000	7.9	0.1316667	0.002194

Tabla 17 DCT

Cantidad Imágenes	Tiempo (seg)	Minutos	Horas
1000	0.0074	0.0001233	2.06E-06
2000	0.0148	0.0002467	4.11E-06
3000	0.0222	0.00037	6.17E-06
4000	0.0296	0.0004933	8.22E-06
5000	0.037	0.0006167	1.03E-05
10000	0.074	0.0012333	2.06E-05
50000	0.37	0.0061667	0.000103
100000	0.74	0.0123333	0.000206
200000	1.48	0.0246667	0.000411
300000	2.22	0.037	0.000617
400000	2.96	0.0493333	0.000822
500000	3.7	0.0616667	0.001028
1000000	7.4	0.1233333	0.002056

Tabla 18 Block Mean Método 1 y 2

Cantidad Imágenes	Tiempo (seg)	Minutos	Horas
1000	0.0226	0.0003767	6.27778E-06
2000	0.0452	0.0007533	1.25556E-05
3000	0.0678	0.00113	1.88333E-05
4000	0.0904	0.0015067	2.51111E-05
5000	0.113	0.0018833	3.13889E-05
10000	0.226	0.0037667	6.27778E-05
50000	1.13	0.0188333	0.000313889
100000	2.26	0.0376667	0.000627778
200000	4.52	0.0753333	0.001255556
300000	6.78	0.113	0.001883333
400000	9.04	0.1506667	0.002511111
500000	11.3	0.1883333	0.003138889
1000000	22.6	0.3766667	0.006277778

Tabla 19 Block Mean Método 3

Cantidad Imágenes	Tiempo (seg)	Minutos	Horas
1000	0.0206	0.0003433	5.72222E-06
2000	0.0412	0.0006867	1.14444E-05
3000	0.0618	0.00103	1.71667E-05
4000	0.0824	0.0013733	2.28889E-05
5000	0.103	0.0017167	2.86111E-05
10000	0.206	0.0034333	5.72222E-05
50000	1.03	0.0171667	0.000286111
100000	2.06	0.0343333	0.000572222
200000	4.12	0.0686667	0.001144444
300000	6.18	0.103	0.001716667
400000	8.24	0.1373333	0.002288889
500000	10.3	0.1716667	0.002861111
1000000	20.6	0.3433333	0.005722222

Tabla 20 Block Mean Método 4

En la siguiente grafica se muestra el comportamiento del tiempo con respecto a las imágenes

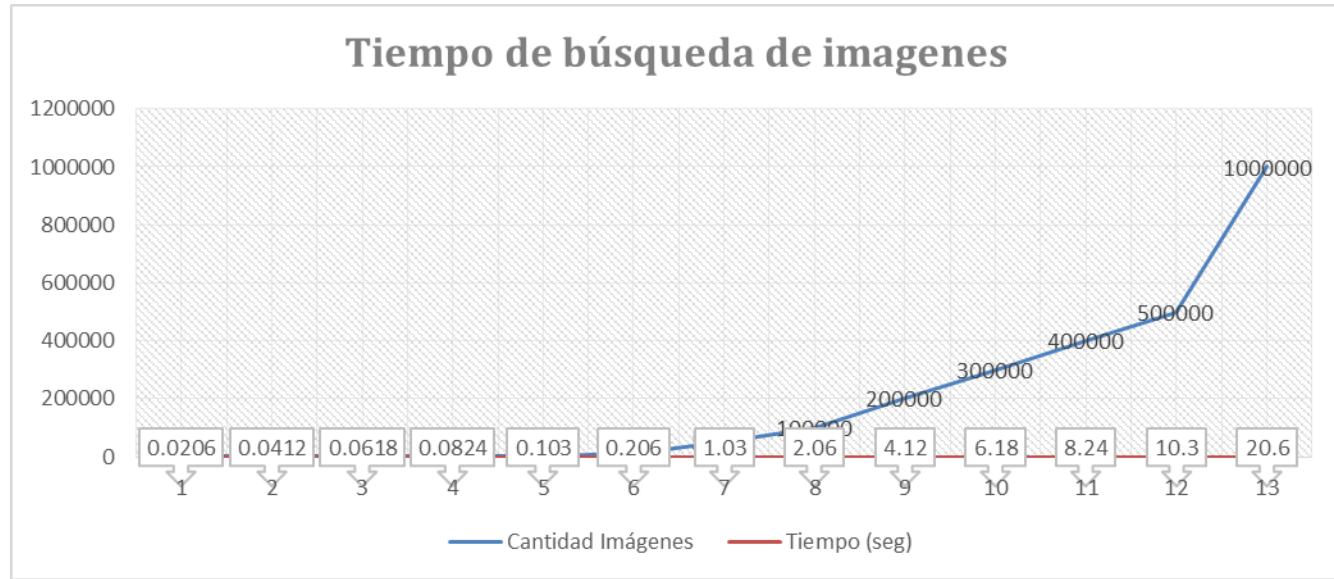


Tabla 21 Comportamiento de búsqueda con firmas Block Mean y DCT

Análisis de almacenamiento

En el siguiente análisis se hace un estudio de espacio ocupado indexado en la base de datos.

Definición de tamaño de firma en cada firma.

La siguiente Tabla 22 Tamaño de firmas en bytes muestra el tamaño determinado de firma para cada método de función perceptual

Función	Tamaño
DCT	8 bytes
Block Mean Método 1	7 bytes
Block Mean Método 2	7 bytes
Block Mean Método 3	48 bytes
Block Mean Método 4	147 bytes

Tabla 22 Tamaño de firmas en bytes

Análisis de espacio ocupado.

En Tabla 23 se muestra el espacio ocupado por las firmas una vez indexadas en el sistema.

Cantidad Imágenes	DCT Hash	Block Mean M1	Block Mean M2	Block Mean M3	Block Mean M4
1000	0.007629	0.006676	0.006676	0.045776	0.14019
2000	0.015259	0.013351	0.013351	0.091553	0.28038
3000	0.022888	0.020027	0.020027	0.137329	0.42057
4000	0.030518	0.026703	0.026703	0.183105	0.56076
5000	0.038147	0.033379	0.033379	0.228882	0.700951
10000	0.076294	0.066757	0.066757	0.457764	1.401901
50000	0.38147	0.333786	0.333786	2.288818	7.009506
100000	0.762939	0.667572	0.667572	4.577637	14.01901
200000	1.525879	1.335144	1.335144	9.155273	28.03802
300000	2.288818	2.002716	2.002716	13.73291	42.05704
400000	3.051758	2.670288	2.670288	18.31055	56.07605
500000	3.814697	3.33786	3.33786	22.88818	70.09506
1000000	7.629395	6.67572	6.67572	45.77637	140.1901

Tabla 23 Espacio almacenado en firmas (Mb)

CAPÍTULO 7

CONCLUSIONES

En este sistema se desarrolló la implementación de un CBIR (*Content based image retrieval*) incorporando nuevos métodos de aproximación a la solución del problema de *Elastic Image Matching*, dichos métodos nos permiten comparar con un margen de error pequeño las imágenes que son perceptualmente similares, incluyendo la búsqueda de imágenes contenidas en archivos.

Tomando en cuenta que no existe una herramienta de software similar para la búsqueda local, el sistema desarrollado demostró ser una herramienta ampliamente útil y de uso sencillo para realizar la búsqueda de una imagen, tarea que requiere gran esfuerzo especialmente cuando el conjunto de búsqueda es grande o continuamente está creciendo.

Se logró realizar la indexación de imágenes contenidas en documentos considerando a estos como un archivo comprimido, realizando la navegación de su estructura interna para obtener los datos correspondientes a las imágenes.

A lo largo del desarrollo se realizaban comprobaciones para que las firmas generadas por las *funciones perceptual hash* arrojaran la representación adecuada para su posterior utilización en la comparación de similitud.

Después de haber desarrollado el sistema “Motor de búsqueda local mediante imágenes”, se ha podido corroborar que de las *funciones perceptual hash* implementadas, la función que más resultados con mayor similitud perceptible para el humano arrojó fue la *transformada discreta coseno*, en la cual en una similitud mayor al 90%, los resultados estaban delimitados por sus características semánticas estéticas, no obstante este método no ofreció mayor holgura en sus resultados, impidiendo la inclusión de imágenes que representaran el mismo contexto estético como se puede observar al comparar la Ilustración 38 Dinosaurios DCT contra la Ilustración 42 Dinosaurios Block Mean 4, donde claramente el método block mean permitió ver resultados cercanos al contenido deseado. En toda búsqueda de imágenes realizada los resultados que estaban por encima del 97% de similitud mostraban la imagen perceptualmente igual a la buscada.

En la fase final de pruebas se realizó un análisis de rendimiento tomando en cuenta los tiempos medidos de ejecución en cada fase del sistema, desde la indexación de archivos hasta la búsqueda y comparación de firmas de imágenes dando resultados deseables de menos de 10 minutos de duración en la indexación de una biblioteca ligeramente grande de hasta veinte mil imágenes, pudiendo este resultado ser reducido si se usa un equipo con características más avanzadas e incluso ser paralelizado a nivel hardware con tecnologías crecientes como Nvidia (*Cuda*).

Uno de los mayores retos durante el desarrollo fue la falta de documentación sobre el funcionamiento de la metodología *perceptual hashing* ya que es un área reciente de investigación en la cual se combinan muchas disciplinas, haciendo difícil poder comprobar los resultados contra otros métodos propuestos debido a la poca información expuesta sobre esta metodología.

Resulta de gran interés la aplicación de nuevas opciones en el sistema como la generación de videos, la posible exportación a sistemas móviles los cuales tras los avances tecnológicos ya pueden realizar las operaciones requeridas para realizar la búsqueda de imágenes localmente o escalarlo a sistemas distribuidos para generar una indexación de tamaños realmente grandes, e incluso solo limitarlo a la similitud de imágenes explotando el potencial que tiene a la comparación y búsqueda de similitud entre fenómenos.

CAPÍTULO 8

TRABAJO A FUTURO

Para el trabajo que se pretende realizar a futuro se proponen dos puntos principales:

- Implementación de métodos de comparación de similitud en distintos fenómenos:
 - ✓ Descripción de similitud entre archivos de audio.
 - ✓ Descripción de similitud entre archivos de video.
 - ✓ Descripción de similitud entre fenómenos descritos en arreglos de N dimensiones.
- Perfeccionamiento y mejoras de búsquedas en el sistema de imágenes.
 - ✓ Utilización de metodologías u otras tecnologías combinadas para la obtención de mejores resultados
 - ✓ Mejora de algoritmo de búsqueda utilizando procedimientos de comparación hash optimizados a grandes cantidades de información.
 - ✓ Profundización de búsqueda orientada a elementos específicos dentro de imágenes.
- Escalabilidad en procesamiento
 - ✓ Utilización de tecnologías concurrentes como *Nvidia (cuda)*
 - ✓ Implementación de lógica en un entorno distribuido.

Índice Tablas

TABLA 1 CONCEPTOS FUNCIÓN PERCEPTUAL HASH.....	15
TABLA 2 SIMILITUD ENTRE IMÁGENES	16
TABLA 3 TIPOS DE FUNCIONES PERCEPTUAL HASH.....	17
TABLA 4 EJEMPLO DE DISTANCIA HAMMING	22
TABLA 5 SISTEMAS SIMILARES	24
TABLA 6 CARACTERÍSTICAS DEL SISTEMA.....	25
TABLA 7 LENGUAJES DE PROGRAMACIÓN.....	26
TABLA 8 GESTORES DE BASE DE DATOS.....	27
TABLA 9 EVALUACIÓN INICIAL DE RIESGOS	34
TABLA 10 ANÁLISIS DE RIESGOS.....	35
TABLA 11 ESTRATEGIAS DE CONTROL.....	36
TABLA 12 ESTIMACIÓN COSTOS	37
TABLA 13 IMÁGENES DENTRO DE ARCHIVOS.....	72
TABLA 14 PRUEBAS DE RENDIMIENTO	73
TABLA 15 ESPECIFICACIONES DE IMAGEN EN PRUEBAS DE RENDIMIENTO.....	73
TABLA 16 ESTIMACIÓN DE RENDIMIENTO EN GENERACION DE FIRMAS.....	75
TABLA 17 DCT	77
TABLA 18 BLOCK MEAN MÉTODO 1 Y 2	77
TABLA 19 BLOCK MEAN MÉTODO 3	78
TABLA 20 BLOCK MEAN MÉTODO 4	78
TABLA 21 COMPORTAMIENTO DE BÚSQUEDA CON FIRMAS BLOCK MEAN Y DCT	79
TABLA 22 TAMAÑO DE FIRMAS EN BYTES.....	80
TABLA 23 ESPACIO ALMACENADO EN FIRMAS (Mb).....	80

Índice Ilustraciones

ILUSTRACIÓN 1 FUNCIÓN PERCEPTUAL HASH	14
ILUSTRACIÓN 2 IMÁGENES SEMÁNTICAMENTE DETECTADAS COMO NO SIMILARES.....	16
ILUSTRACIÓN 3 REPRESENTACIÓN DE VARIACIÓN EN SIMILITUD.....	17
ILUSTRACIÓN 4 LOS 64 PESOS BASE OBTENIDOS DE (5) DE UNA MATRIZ DE 64 x 64	19
ILUSTRACIÓN 5 IMAGEN ORIGINAL Y SU COMPRESIÓN DE UNA IMAGEN POR JPEG AL 85%.....	19
ILUSTRACIÓN 6 DIAGRAMA A BLOQUES DEL SISTEMA.....	38
ILUSTRACIÓN 7 DIAGRAMA BLOQUES	39
ILUSTRACIÓN 8 DIAGRAMA DE CLASES CORRESPONDENTES AL ACCESO DE DATOS	40
ILUSTRACIÓN 9DIAGRAMA DE CLASES PROCESAMIENTO DE IMÁGENES	41
ILUSTRACIÓN 10 DIAGRAMA DE CLASES PERCEPTUAL HASH	41
ILUSTRACIÓN 11 DIAGRAMA DE CLASES MEDIA OBJECT HANDLER.....	42
ILUSTRACIÓN 12 DIAGRAMA DE CLASES FILEAGENT	42
ILUSTRACIÓN 13 DIAGRAMA SECUENCIA GENERACIÓN FIRMA	43
ILUSTRACIÓN 14 DIAGRAMA SECUENCIA GENERACIÓN DE FIRMA DESDE ARCHIVOS INDEXADOS	44
ILUSTRACIÓN 15 DIAGRAMA DE SECUENCIA EXTRACCIÓN DE IMÁGENES EN DOCUMENTOS	45
ILUSTRACIÓN 16 BUSQUEDA DE IMÁGENES	46
ILUSTRACIÓN 17 PRIMERA EJECUCIÓN	47
ILUSTRACIÓN 18 INDEXADO AUTOMATICO	47
ILUSTRACIÓN 19 GENERACION DE FIRMAS	48
ILUSTRACIÓN 20 PANTALLA PRINCIPAL	49
ILUSTRACIÓN 21 REALIZACION DE BUSQUEDA.....	50
ILUSTRACIÓN 22 CONFIGURACION DE SISTEMA.....	51
ILUSTRACIÓN 23 REINICIALIZACIÓN DE SISTEMA.....	51
ILUSTRACIÓN 24 PANTALLA ÍNICO BUSCADOR	52
ILUSTRACIÓN 25 INDEXACIÓN DE ARCHIVOS	53
ILUSTRACIÓN 26 PASOS DE PRE PROCESAMIENTO DE IMAGEN	54
ILUSTRACIÓN 27 GENERACIÓN DE LA DCT DE UNA IMAGEN	55
ILUSTRACIÓN 28 GENERACIÓN DE FIRMA DE LA IMAGEN	55
ILUSTRACIÓN 29 PASOS DE REDUCCIÓN IMAGEN	56
ILUSTRACIÓN 30 DIVISIÓN POR BLOQUES	57
ILUSTRACIÓN 31 GENERACIÓN DE HASH BLOCK MEAN.....	57
ILUSTRACIÓN 32 GENERACIÓN DE DISTANCIA ENTRE FIRMAS	59
ILUSTRACIÓN 33 VISTA A CONJUNTO DE BIBLIOTECA	60
ILUSTRACIÓN 34 IMÁGENES SOLO DINOSAURIOS	61
ILUSTRACIÓN 35 FOTOS NIÑA.....	61
ILUSTRACIÓN 36 IMÁGENES DENTRO DE DOCUMENTOS	62
ILUSTRACIÓN 37 DINOSAURIOS DCT	63
ILUSTRACIÓN 38 DINOSAURIOS BLOCK MEAN 1	63
ILUSTRACIÓN 39 DINOSAURIOS BLOCK MEAN 2	64
ILUSTRACIÓN 40 DINOSAURIOS BLOCK MEAN 3	64
ILUSTRACIÓN 41 DINOSAURIOS BLOCK MEAN 4	65
ILUSTRACIÓN 42 NIÑA DCT.....	66
ILUSTRACIÓN 43 NIÑA BLOCK MEAN 1	66
ILUSTRACIÓN 44 NIÑA BLOCK MEAN 2	67
ILUSTRACIÓN 45 NIÑA BLOCK MEAN 3	67
ILUSTRACIÓN 46 NIÑA BLOCK MEAN 4	68
ILUSTRACIÓN 47 ESCENA DCT	69

ILUSTRACIÓN 48 ESCENA BLOCK MEAN 1.....	69
ILUSTRACIÓN 49 ESCENA BLOCK MEAN 2.....	70
ILUSTRACIÓN 50 ESCENA BLOCK MEAN 3.....	70
ILUSTRACIÓN 51 ESCENA BLOCK MEAN 4.....	71

Índice Graficas

GRÁFICA 1 RENDIMIENTO GENERACION DE FIRMA	73
GRÁFICA 2 RENDIMIENTO DE COMPARACION DE FIRMAS DE DOS IMÁGENES	74
GRÁFICA 3 RENDIMIENTO DE COMPRAACION DE DOS IMÁGENES SIN FIRMA PREVIA.....	75
GRÁFICA 4 ESTIMACION DE GENERACIÓN DE FIRMAS.....	76

GLOSARIO

- **Funciones hash.**- algoritmos que consiguen crear a partir de una entrada una salida alfanumérica de longitud normalmente fija que representa un resumen de toda la información que se le ha dado.
- **Interpolación.**- Disponer de un cierto número de puntos obtenidos por muestreo o a partir de un experimento y pretender construir una función que los ajuste.
- **Joint Photographic Experts Group (JPEG).** - Grupo Conjunto de Expertos en Fotografía, es el nombre de un comité de expertos que creó un estándar de compresión y codificación de archivos e imágenes fijas.
- **NP.**- Acrónimo en inglés de nondeterministic polynomial time ("tiempo polinomial no determinista").
- **Problema NP.**- Es el conjunto de problemas que pueden ser resueltos en tiempo polinómico por una máquina de Turing no determinista.
- **Problema NP-Completo.**- Problemas que son completos en NP, es decir, los más difíciles de resolver en NP.
- **Transformada discreta coseno (DCT).**-es una transformada basada en la Transformada de Fourier discreta, pero utilizando únicamente números reales.
- **Drag & Drop.**-Acción de arrastrar y soltar un elemento dentro de un contenedor, mayormente utilizado en interfaces gráficas.

BIBLIOGRAFÍA

- Hamming, R. (1950). *Error Detecting and Error Correcting Codes*.
- Keysers. (2000). *Image Comparision Invariant methods*.
- Keysersa, D. (2002). *Elastic Image Matching is NP-Complete*.
- Meixner, A. a. (2006). *Robustness and security of a wavelet-based*.
- Michel Deza, E. D. (2012). *Encyclopedia of Distances*. Moscow State Pedagogical University.
- Romero, B., & Rojas, M. (2010, Julio). Pruebas PVT para Yacimientos de Gas Condensado. Barcelona, España.
- Schneider, M. a. (1996). *A robust content based digital signature*.
- Uchida, S. (1998). *Elastic image matching, two-dimensional (2D) warping*.
- Venkatesan, R. K. (2000). Robust image hashing. *Proceedings of the International Conference on Image* (pp. 664-666). ICIP.
- Yang, B. G. (2006). *Block mean value based image perceptual*. IEEE.
- Zauner, C. (2010). *Implementation and Benchmarking of Perceptual Image Hash Functions*. Hagenberg.