

Proyecto Final de Carrera

Interfaz Gráfica de Usuario para la Búsqueda de Imágenes basada en Imágenes



GOS- Graphic Object Searcher

Alumna: Silvia Cortés Yuste

Tutor: Xavier Giró i Nieto

*Ingeniería Técnica de Telecomunicaciones, especialidad
Sonido e Imagen (EUETIT –UPC)*

Junio 2009

Datos de contacto:

Silvia Cortés Yuste

Licenciada en Comunicación Audiovisual

Estudiante de Ingeniería Técnica de Telecomunicaciones, Sonido e Imagen.

silcoryus@hotmail.com

Xavier Giró i Nieto

Profesor de Ingeniería Técnica de Telecomunicaciones.

Universitat Politècnica de Catalunya (UPC)

xavier.giro@upc.edu // <http://gps-tsc.upc.es/imatge/Xgiro/start.html>

"Las imágenes televisivas utilizadas en este proyecto son propiedad de TVC, Televisió de Catalunya, SA, y contienen copyright.

Las imágenes televisivas de TVC que aparecen en este trabajo han sido proporcionadas por TVC con finalidad exclusiva de investigación para el Proyecto i3media"

Índice

Tabla de Figuras	6
1. Introducción	8
2. Estado del arte	13
2.1 Importancia de la HCI (<i>Human-Computer Interaction</i>).....	13
2.2 Requisitos para GUI de sistemas CBIR.....	14
2.3 Tecnología utilizada en la implementación de las GUI	19
2.4 Tipología de GUI para sistemas CBIR.....	22
2.4.1 Interfaces de sistemas CBIR comerciales	22
<input type="checkbox"/> QBIC (Query By Image Content)	23
<input type="checkbox"/> Excalibur Visual RetrievalWare.....	27
<input type="checkbox"/> GazoPa Similar Image Searcher	29
<input type="checkbox"/> Similar Images (Google Labs)	32
<input type="checkbox"/> Picollator.....	33
<input type="checkbox"/> Piximilar, PixID, TinEye	36
<input type="checkbox"/> ImageFinder	41
<input type="checkbox"/> Like.com (Visual Shopping)	44
2.4.2 Interfaces de sistemas CBIR académicos	47
<input type="checkbox"/> Photobook	48
<input type="checkbox"/> AMORE (Advanced Multimedia Oriented Retrieval Engine).....	51
<input type="checkbox"/> MARS (Multimedia Analysis and Retrieval Systems)	53
<input type="checkbox"/> CIREs: Content Based Image REtrieval System	58
<input type="checkbox"/> CIRCUS (Content-based Image Retrieval and Consultation User System)	65
<input type="checkbox"/> Caliph & Emir	68
<input type="checkbox"/> Buscador de videos mediante descriptores MPEG-7.....	74
3. Requisitos	79
3.1 Llamada a B_RANKER	80
3.2 Formato MPEG-7/XML para datos de entrada y salida	81
3.2.1 Datos de entrada.....	81
<input type="checkbox"/> Ejemplo de archivo XML con datos de entrada.....	83
3.2.2 Datos de salida	85
<input type="checkbox"/> Ejemplo de archivo XML con datos de salida	86
4. Entorno de Desarrollo	89
4.1 Entorno de ejecución	89
4.2 Entorno de trabajo.....	91
4.2.1 Java	93
4.2.2 Eclipse	95
4.2.3 Subversion (SVN).....	95
4.2.4 NoMachine	96
5. Diseño de la Interfaz Gráfica.....	98
5.1 Estilo de la interfaz.....	98
5.1.1 Diseño del logo	99
5.1.2 Look&Feel.....	100
5.1.3 Iconos	101
5.2 Elementos básicos de la GUI	102
5.3 Organización de los elementos	105
5.3.1 Patrones de lectura de pantalla	106
5.3.2 Presentación de la información	108

5.3.2.1	Ventana principal.....	108
5.3.2.2	Menú general	109
5.3.2.3	Barra de herramientas.....	110
5.3.2.4	Área de consulta.....	112
5.3.2.5	Área de resultados	115
5.4	Funcionalidades de la interfaz.....	118
5.4.1	Nueva búsqueda	118
5.4.2	Navegación por los resultados	118
5.4.3	Selección de resultados	119
6.	Futuro del GOS.....	121
7.	Conclusiones	123
8.	Bibliografía.....	126

ANEXO I. Manual de Usuario

1.	Introducción.....	132
1.1	Recursos online.....	132
1.2	Requisitos	133
1.2.1.	Motor de búsqueda.....	133
1.2.2.	Estructura de directorios.....	133
1.2.3.	JRE de Java	134
2.	Interfaz de usuario.....	135
2.1	Qué puede hacer el GOS.....	135
2.2	Áreas de trabajo.....	136
2.2.1	Menú principal.....	136
2.2.2	Barra de herramientas.....	138
2.2.3	Área de consulta	139
2.2.3.1	Panel de imagen de consulta	140
2.2.3.2	Panel de criterios de fusión	141
2.2.3.3	Panel de espacio de búsqueda.....	143
2.2.4	Área de resultados	144
2.2.4.1	Imagen destacada (con foco)	144
2.2.4.2	Parrilla de resultados	146
3.	Cómo utilizar el GOS	148
3.1	Cómo buscar.....	148
	Paso 1. Cargar la imagen de consulta.....	148
	Paso 2. Configurar el motor de búsqueda	149
	Paso 3. Configurar el ámbito de búsqueda.....	151
	Paso 4. Ejecutar la búsqueda	152
3.2	Cómo navegar por los resultados	152
3.2.1	Convertir una imagen en imagen destacada.....	153
3.2.2	Desplazarse con los botones de navegación	154
3.2.3	Desplazarse con la rueda central del ratón	154
3.3	Cómo seleccionar y guardar resultados	154
3.3.1	Selección individual.....	154
3.3.2	Selección en grupo	155
3.3.3	Selección combinada.....	156
3.3.4	Deseleccionar resultados.....	157
3.3.5	Guardar resultados.....	157
3.4	Cómo afinar la búsqueda	158

3.5	Cómo limpiar la pantalla	158
3.6	Cómo consultar la ayuda.....	159

ANEXO II. Archivos de Ejemplo

<i>Fichero de datos de entrada</i>	161
<i>Fichero de datos de salida</i>	162
<i>Fichero de directorio del espacio de búsqueda</i>	167
<i>Fichero de descripción de un imagen</i>	168

ANEXO III. Recursos Online

<i>Web del GOS</i>	171
<i>Vídeo Demo</i>	172

Tabla de Figuras

Figura 1. Consorcio de empresas i3media.....	9
Figura 2. Esquema de la GUI del GOS.....	11
Figura 3. HCI- Human Computer Interaction	13
Figura 4. Esquema de la perspectiva del usuario.....	15
Figura 5. Esquema de la perspectiva del sistema.....	16
Figura 6. Ejemplo de GUI para el sistema VIPER-GIFT.....	17
Figura 7. Interfaz web-like de Similar Images de Google Labs.....	19
Figura 8. Ventana de la interfaz de ImageFinder.....	20
Figura 9. Formulario de petición de búsqueda de CIRES en la web.	21
Figura 10. Applet de la interfaz de MuSIQUE.....	21
Figura 11. Interfaz de búsqueda de QBIC.....	24
Figura 12. QBIC integrado en la web del State Hermitage Museum.....	26
Figura 13. Interfaz de ejemplo del kit de desarrollo de Excalibur.....	28
Figura 14. Interfaz de GazoPa.....	29
Figura 15. Herramienta de Trimming de GazoPa.....	30
Figura 16. Resultados en modo lista en GazoPa.....	31
Figura 17. Similar Images de Google Labs.....	33
Figura 18. Picollator, buscador de caras.....	34
Figura 19. Imagen desactivada y Buscador en el campo de texto en Picollator.....	35
Figura 20. Imagen de licencia libre identificada por PixID en la portada de un libro.	36
Figura 21. TinEye, visualización de resultados en lista.	37
Figura 22. TinEye, visualización de resultados en mapa.....	38
Figura 23. Pop-up con la herramienta de comparación de TinEye.....	38
Figura 24. Interfaz de Piximilar para búsqueda por imagen y/o texto en IdéeLabs.....	39
Figura 25. Interfaz de Piximilar para búsqueda por color en IdéeLabs.....	39
Figura 26. SimSearch en Masterfile.....	40
Figura 27. Interfaz antigua de ImageFinder.....	41
Figura 28. Interfaz actual de ImageFinder.....	42
Figura 29. Ventana de selección de búsqueda de ImageFinder.....	43
Figura 30. Portal de Like.com.....	45
Figura 31. Visualización de resultados en el portal Like.com.....	47
Figura 32. Interfaz de Photobook.....	48
Figura 33. Interfaz de FourEyes anotando con etiquetas “Water”, “Sky” y “Grass”.....	50
Figura 34. Interfaz del sistema AMORE.....	51
Figura 35. Interfaz del sistema MARS.....	54
Figura 36. Ejemplo ventana de configuración de MARS.....	56
Figura 37. Interfaz del sistema ImageGrouper.....	58
Figura 38. Página de selección de categoría del sistema CIRES.....	59
Figura 39. Página de formulación de la consulta del sistema CIRES.....	60
Figura 40. Nueva interfaz del sistema CIRES.....	61
Figura 41. Página de los tags más populares del sistema CIRES.....	62
Figura 42. Página de resultados actual del sistema CIRES.....	63
Figura 43. Antigua página de resultados con relevance feedback del sistema CIRES.....	64
Figura 44. Interfaz de CIRCUS.....	66
Figura 45. Resultados de dos consultas combinadas en CIRCUS.....	68
Figura 46. Interfaz de Caliph.....	69
Figura 47. Consultas por texto (izquierda) y por gráfico (derecha) de EMIR.....	71
Figura 48. Ventana de consulta de EMIR.....	72
Figura 49. Ventana de resultados de EMIR.....	73
Figura 50. Visualización de resultados en 2D de EMIR.....	74

Figura 51. Página principal del Buscador de vídeos mediante MPEG-7.....	75
Figura 52. Página inicial del Buscador de vídeos mediante MPEG-7.....	77
Figura 53. Página de resultados del Buscador de vídeos mediante MPEG-7	78
Figura 54. Esquema diálogo	79
Figura 55. Diálogo para introducir la imagen de consulta.....	80
Figura 56. Archivo de configuración del espacio de búsqueda.....	90
Figura 57. Ejecución remota de GOS	91
Figura 58. Plataforma JAVA.....	94
Figura 59. Repositorio en SVN de Eclipse	96
Figura 60. Interfaz del GAT.....	99
Figura 61. Logos del GAT y el GOS.....	100
Figura 62. Interfaz del GOS	101
Figura 63. Iconos comunes del GAT y el GOS para “Guardar”, “Borrar” y “Ayuda” ...	102
Figura 64. Iconos específicos del GOS para “Buscar” y “Seleccionar”	102
Figura 65. Los tres ángulos y los tres roles del diseño de GUI.....	105
Figura 66. Zonas de importancia en la pantalla.....	106
Figura 67. Patrón F de lectura típico de las páginas web.....	107
Figura 68. Organización de elementos en pantalla del GOS.....	107
Figura 69. Ventana principal del GOS.....	108
Figura 70. Menú Archivo en la barra de menú del GOS.....	110
Figura 71. Barra de herramientas del GOS.....	111
Figura 72. Área de consulta del GOS	112
Figura 73. Panel de criterio de fusión del GOS.....	113
Figura 74. Panel de espacio de búsqueda del GOS.....	114
Figura 75. Panel de resultados del GOS.....	115
Figura 76. Área de texto informativa de la imagen destacada del GOS.	116
Figura 77. Diapositiva con foco, seleccionada y sin foco respectivamente.....	118
Figura 78. Botones de navegación al lado de la imagen con foco.....	119
Figura 79. Selección de un grupo de resultados	120
Figura 80. El proyecto i3media en el contexto de las empresas miembros del consorcio y los sectores industriales implicados.....	125

1. Introducción

Dicen que “*una imagen vale más que mil palabras*”. La imagen le ha ganado terreno a la palabra en la sociedad actual, donde los documentos gráficos y audiovisuales son los formatos de información preferidos por los usuarios, tanto en el ámbito profesional como doméstico. La generación de contenido audiovisual se ha incrementado de forma vertiginosa en los últimos años gracias a la digitalización, dificultando el acceso a una información que no da tiempo a ordenar ni catalogar, y por lo tanto, no es fácil de encontrar. Cada vez es más evidente la necesidad de la aparición de nuevas técnicas y herramientas de gestión de todos estos contenidos.

Dentro de esta coyuntura, este Proyecto de Final de Carrera (PFC) responde a esa necesidad de creación de herramientas de acceso a contenido multimedia, nuevas herramientas que faciliten la recuperación de toda esa información audiovisual almacenada. El **Graphic Object Searcher (GOS)** es una interfaz gráfica para realizar búsquedas de imágenes alojadas en grandes bases de datos a partir de una imagen ejemplo y de unos criterios de búsqueda establecidos por el usuario.

El GOS es un PFC enmarcado dentro de la iniciativa **i3media** [1], un macroproyecto estatal dedicado a la investigación y desarrollo de tecnologías para la creación y la gestión automatizada de contenidos audiovisuales inteligentes. Las principales empresas del sector media se han unido en el consorcio *i3media* con el objetivo de impulsar diversas áreas de investigación asociadas al contenido audiovisual, con la colaboración de grupos de investigadores expertos de universidades y centros tecnológicos que llevan años trabajando con éxito en este campo. El proyecto *i3media*, alineado con las prioridades de los programas de investigación de la Comisión Europea, aspira a propiciar un papel de liderazgo de la industria audiovisual española en el mercado global.

El *i3media* incluye la investigación en las áreas de procesamiento de imágenes, visión artificial por ordenador, computación gráfica, semántica y ontologías audiovisuales, tecnologías de audio, síntesis del habla y lingüística computacional, así como en recuperación de imágenes adecuada al contexto. Para abarcar todas estas grandes áreas, el proyecto comprende diez objetivos científicos, que se estructuran en

diez actividades principales (A1-A10), que a su vez se componen de diversas subtareas.



Figura 1. Consorcio de empresas i3media

El GOS forma parte de la tarea A2.T3 de las actividades definidas por el i3media. La actividad A2 está dedicada al **análisis y modificación de imágenes**, con problemáticas como el reconocimiento, identificación y sustitución de personajes y objetos en escena. La tarea 3 de esta actividad trata la **anotación, indexado y resumen automático de vídeo**, cuyo objetivo es proporcionar tecnologías para mejorar la segmentación de imágenes en clases semánticas¹, para su posterior reconocimiento con múltiples funcionalidades: extracción de objetos y personajes en imágenes, su sustitución por otros objetos en la propia imagen o su recuperación en otras imágenes. El GOS es una aplicación de soporte para estas tecnologías desarrolladas en esta actividad, ya que proporciona un entorno gráfico amigable para la utilización de estas herramientas de indexación y recuperación de contenido audiovisual.

Empresas como **Mediapro**, líder del proyecto y dedicada a la producción de contenidos y provisión de servicios media, y la **Corporació Catalana de Mitjans Audiovisuals (CCMA)**, empresa de desarrollo de tecnologías de gestión de

¹ Una clase semántica es una unidad de significado representada a través de descriptores (en el caso de imágenes, descriptores visuales), utilizada para la interpretación y comprensión de la información de un contenido audiovisual por una máquina.

contenidos, son los principales destinatarios de esta herramienta. El GOS permite realizar búsquedas con imágenes de ejemplo en sus archivos de vídeo para crear nuevos contenidos, agilizando el proceso de documentación y recuperación de contenido, al visualizar rápidamente los resultados de la búsqueda.

ACTIVIDADES Y TAREAS	EMPRESAS	CENTROS (GRUPOS) DE INVESTIGACIÓN
A2. ANÁLISIS Y MODIFICACIÓN DE IMÁGENES	Líder MPG	
A2.T1 Reconocimiento y extracción de objetos y personajes a partir de imágenes	MP	BMCI(GPI)
A2.T2 Sustitución de objetos y personajes en imágenes	MP, MPG	BMCI (GPI)
A2.T3 Anotación, indexado y resumen automático de vídeo	MP, CCMA	UPC (GPI)
A2.T4 Relleno automático de escenas con objetos y personajes	MP, MPG	BMCI(LMEX), BMCI (GTI)

Empresas: MP Media Pro, MPG Media Planning Group, CCMA Corporació Catalana de Mitjans Audiovisuals.

Centros de Investigación: BMCI Barcelona Media Centro de Innovación, UPC Universidad Politècnica de Catalunya.

Grupos de Investigación: GPI Procesamiento de Imágenes, LMEX Laboratorio de Medios Experimentales, GTI Procesamiento de Imágenes.

Tabla 1. Relación de Actividades y Tareas con las empresas y grupos de investigación que los llevan a cabo. [3]

El GOS es una interfaz gráfica de usuario (GUI – *Graphic User Interface*) desarrollada por el Grupo de Procesado de Imagen (GPI) de la Universidad Politècnica de Catalunya (UPC), pensada para facilitar la utilización de métodos de búsqueda inteligente de imágenes desarrollados por el grupo. El GOS pretende ser una pieza más dentro de un conjunto de aplicaciones gráficas y herramientas que automatizan procesos de gran complejidad como la indexación, selección, búsqueda y recuperación de imágenes y vídeos a través de repositorios de datos distribuidos de gran tamaño.

Una GUI, por definición, es el área de comunicación entre el hombre y la máquina [2]. Los usuarios pueden interactuar con procesos ejecutados por un

ordenador a través de los elementos visuales presentados en pantalla, enviar y recibir información e incluso generar nuevos contenidos. El GOS se encarga de realizar esta tarea de intermediario entre el usuario y el sistema de búsqueda de imágenes. Por un lado, la interfaz recoge los datos de entrada necesarios para realizar la consulta y se pone en contacto con el motor de búsqueda para que la ejecute. Por otro, una vez realizada la búsqueda, la interfaz presenta al usuario los resultados obtenidos.

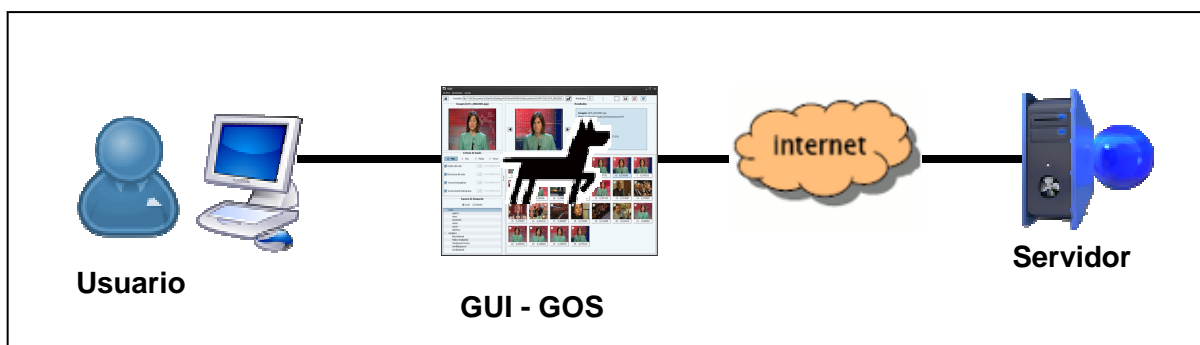


Figura 2. Funcionamiento del GOS

La interfaz del GOS utiliza un sistema de búsqueda inteligente para recuperar imágenes similares a la utilizada en la consulta. Este tipo de sistemas de búsqueda son conocidos como **Content-Based Image Retrieval (CBIR)**, ya que la recuperación de la información se realiza a través del propio contenido de la imagen indexada, sin necesidad de tener asociada ningún tipo de anotación manual. La mayoría de estos sistemas utilizan la técnica de **consulta mediante ejemplo (QbE - Query by Example)**, que consiste en realizar la búsqueda a partir de una imagen de consulta y una serie de criterios establecidos por el usuario para encontrar imágenes parecidas en grandes bases de datos de imágenes. El GOS gestiona los parámetros de búsqueda especificados por el usuario para lanzar la petición de búsqueda contra la base de datos a través de un motor de búsqueda con ejemplo. Este motor de búsqueda nos devuelve una lista de resultados ordenados según la similitud con la imagen ejemplo consultada.

El GOS también gestiona la presentación de los resultados obtenidos en la búsqueda y nos ofrece una serie de funcionalidades para la visualización de imágenes resultado, la selección y almacenamiento de resultados satisfactorios, y la realización automática de nuevas búsquedas.

La realización de esta aplicación como PFC permite trabajar en las dos áreas tecnológicas con más auge actualmente: el sector multimedia (gestión de contenido audiovisual) y las tecnologías de la información (TIC) (informática al servicio de la comunicación). Estas dos áreas tienden a aunar esfuerzos en una sociedad abocada al uso y consumo de contenido audiovisual a través de múltiples plataformas y dispositivos en cualquier sector económico y social (ocio, formación, servicios, etc.). Cualquier profesional del sector audiovisual ha de adquirir conocimiento y experiencia en ambas áreas para cimentar su carrera.

A nivel personal, el proyecto también me brinda la oportunidad de aplicar los conocimientos adquiridos en mi anterior titulación, la licenciatura de Comunicación Audiovisual, relacionados con el diseño y presentación de contenido audiovisual y su consumo.

2. Estado del arte

2.1 Importancia de la HCI (*Human-Computer Interaction*)

Hoy en día, personas y ordenadores están condenados a “entenderse”. Las interfaces de usuario son las herramientas clave para establecer esa comunicación entre el hombre y la máquina. El usuario debe transmitir a la máquina lo que desea conseguir, la máquina debe entender la orden y ejecutarla, y finalmente responder al usuario con el resultado del proceso o acción ejecutada.



Todo este diálogo se realiza a través de la interacción que se establece entre ambos actores y se gestiona a través de un elemento intermedio, la **interfaz de usuario**. En cualquier sistema de recuperación de información o contenido audiovisual, este diálogo es imprescindible, y de la precisión del “traductor” depende el éxito de los resultados.

Figura 3. HCI- Human Computer Interaction

La disciplina **human-computer interaction (HCI)**, enmarcada dentro de las ciencias documentales, nace con la voluntad de ayudar a mejorar la comunicación entre los usuarios y los sistemas de documentación (informatizados). La HCI se ocupa del análisis y diseño de interfaces entre el hombre y la máquina, conocidas como interfaces de usuario, estudiando la creación de productos informáticos que ayuden en la realización de tareas a sus usuarios atendiendo a la facilidad de uso, al tiempo de ejecución, a la evitación de los posibles errores y, en consecuencia, a su satisfacción [5]. Destaca el carácter interdisciplinar de la HCI, que abarca aspectos humanos, tecnológicos y la comunicación entre ambos.

La HCI afirma que la forma de presentar la información al usuario en un sistema de recuperación de contenido hace variar su manera de interactuar [5], y por lo tanto influirá en la utilidad de la herramienta y el grado de satisfacción final que se genere. En este sentido, una aplicación de búsqueda de imágenes como la que nos ocupa, debe trabajar con minuciosidad la presentación de la consulta a realizar y la presentación de los resultados obtenidos.

El diseñador de la interfaz se enfrenta a dos preguntas clave: **¿cómo introducirá la información el usuario?** y **¿cómo se presentará al usuario la información obtenida por el motor de búsqueda?** [4]

La presentación al usuario de la información de entrada y salida al sistema será nuestro punto de partida, debido a su importancia en este tipo de interfaces. A continuación estableceremos los criterios que nos servirán para analizar las diferentes propuestas existentes actualmente y la evolución que se aprecia desde las primeras interfaces hasta las más contemporáneas.

2.2 Requisitos para GUI de sistemas CBIR

Los sistemas CBIR surgieron a comienzos de la década de los '90, pero no es hasta finales de esa misma década que se produce un auge en los trabajos de investigación dedicados a este ámbito. Sin embargo, la mayoría trata el tema desde el punto de vista técnico y son pocos los trabajos dedicados a temas como la evaluación de su eficacia, el diseño de interfaces o estudio de los usuarios [8].

“While the underlying technology of CBIR systems is being advanced both system developers and researchers have generally overlooked the importance of the human-computer interaction (HCI) and the crucial role of the user interface.” [10]

Este PFC se centra en el diseño de una GUI para un sistema CBIR, para ampliar ese pequeño conjunto de trabajos dedicados a este tema. Es evidente, que el método de diseño ha de centrarse en el usuario², puesto que el objetivo final de todo sistema de recuperación de imágenes es permitir que el usuario acceda al contenido que busca.

Para caracterizar la interacción entre el usuario y el sistema de búsqueda que se llevará a cabo en la interfaz, se han de tener en cuenta ambos puntos de vista [11]. La GUI será la encargada de establecer la conexión entre los interlocutores en los tres aspectos fundamentales que conforman la búsqueda:

² **User-centered design (UCD)** is an approach to design that grounds the process in information about the people who will use the product. UCD processes focus on users through the planning, design and development of a product.
(http://www.upassoc.org/usability_resources/about_usability/what_is_ucd.html)

1. **Presentación de la información** (para la elaboración de la consulta por parte del usuario y la visualización de los resultados entregados por el sistema).
2. **Ámbito de la búsqueda** (local, bases de datos especializadas, la web, etc.).
3. **Modalidad de la consulta** (palabras clave, texto libre, imagen, gráficos o una combinación de algunas de las anteriores).

Veamos cómo interpreta cada interlocutor estos 3 puntos:

A. PERSPECTIVA DEL USUARIO: a la hora de realizar una búsqueda, el usuario deberá tomar una serie de decisiones que determinarán el resultado:

1. **Qué busca:** el usuario debe definir su consulta, estableciendo los parámetros de entrada al sistema.
2. **Dónde busca:** el ámbito de la búsqueda.
3. **Cómo lo busca:** el tipo de consulta utilizada en la búsqueda.

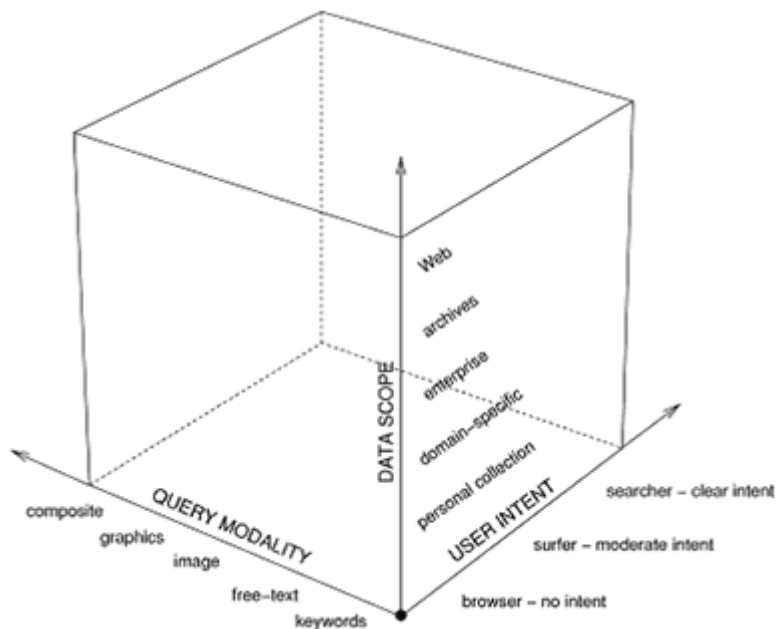


Figura 4. Esquema de la perspectiva del usuario

B. PERSPECTIVA DEL SISTEMA: a la hora de ejecutar la búsqueda, el sistema procederá según los parámetros establecidos en la consulta, que condicionarán los resultados:

1. **Cómo presenta los resultados:** el sistema debe saber qué parámetros de salida ha de entregar, desde el número de resultados, su orden de presentación, la información extra asociada, etc.
2. **Dónde busca:** el ámbito de la búsqueda.
3. **Qué tipo de consulta ejecuta:** qué tipo de búsqueda emplea para encontrar los resultados.

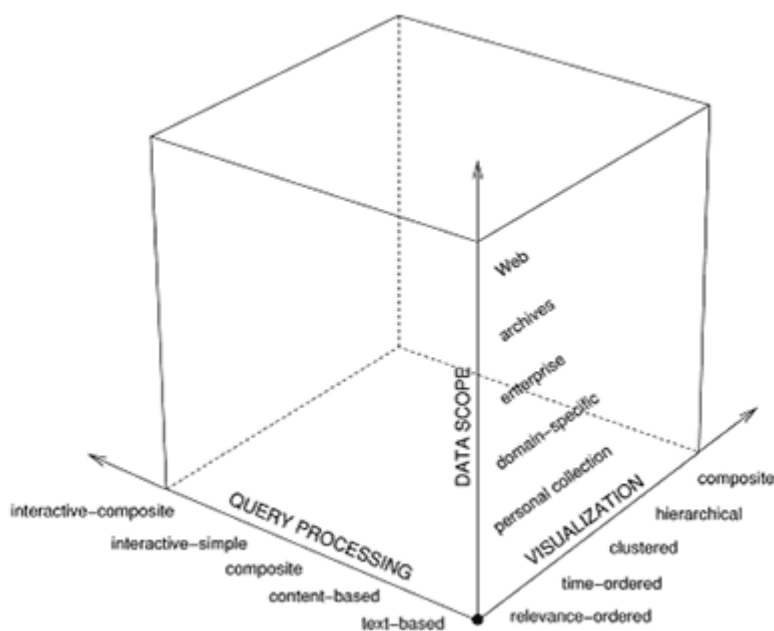


Figura 5

Figura 5. Esquema de la perspectiva del sistema

Con el fin de crear una interfaz que satisfaga las necesidades de ambos puntos de vista, conviene establecer una serie de requisitos previos que toda GUI de soporte a sistemas CBIR debe plantearse para definir su diseño:

1. **Cómo presentar las áreas de consulta y de resultados:** como premisa, el área de selección de consulta siempre debe estar visible, y accesible de manera rápida al usuario. El área de resultados debe hacerse visible tras la consulta.

- 2. Qué formatos de imágenes soporta:** quizás lo ideal sería tratar la máxima variedad posible de formatos de imágenes (diferentes tipos de archivo de imágenes, dibujos como imagen de entrada, etc.), pero la decisión final dependerá del uso que se hará de la herramienta.
- 3. Cómo configurar el motor de búsqueda:** estableciendo un tipo de búsqueda automática o permitiendo la configuración manual por parte del usuario de los algoritmos que realizan la búsqueda (otorgan pesos a los criterios de búsqueda, combinarlos, etc.).
- 4. Qué tipos de formulación de consulta implementa:** existen diferentes modalidades de búsqueda en sistemas CBIR, los más empleados son el método basado en texto (*tags* o palabras clave), las consultas con ejemplo o una combinación de ambas.
- 5. Qué métodos para refinar las búsquedas utiliza:** generalmente las imágenes resultado pueden ser seleccionadas como punto de entrada en una nueva consulta. Si el tipo de interfaz lo permite, se pueden implementar sistemas de feedback, conocidos como *Relevance Feedback*.
- 6. A qué tipo de usuario va destinada:** genérico y/o experto.



Figura 6. Ejemplo de GUI para el sistema VIPER-GIFT³

³ GIFT: GNU Image Finding Tool

Paralelamente a estos requisitos, el resultado final debería cumplir unos criterios mínimos de usabilidad. Estudios previos han establecido una serie de principios o criterios de usabilidad básicos para determinar el grado de utilidad de este tipo de herramientas [8] [9]:

CRITERIOS DE USABILIDAD	ELEMENTOS DE LA GUI
1. Visibilidad del estado del sistema: el sistema debe mantener informado al usuario.	<ul style="list-style-type: none"> - Búsquedas sin resultados - Mensajes de error - Numeración de los resultados - Ponderación de los resultados
2. Relación entre el sistema y el mundo real: la información debe aparecer en un orden natural y lógico.	<ul style="list-style-type: none"> - Información sobre las imágenes - Orden de instrucciones lógico - Lenguaje orientado al usuario
3. Control y libertad del usuario:	<ul style="list-style-type: none"> - Opciones de hacer y deshacer - Cancelación de búsquedas - Barra de navegación propia
4. Consistencia y estándares: el usuario debe conocer el significado de palabras, situaciones o acciones.	<ul style="list-style-type: none"> - Plataforma de convenciones aceptada - Coherencia interna
5. Prevención de errores:	<ul style="list-style-type: none"> - Detección de problemas
6. Reconocer mejor que recordar:	<ul style="list-style-type: none"> - Visibilidad de los objetos, acciones y opciones - Instrucciones apropiadas y fácilmente recuperables.
7. Flexibilidad y eficacia de uso:	<ul style="list-style-type: none"> - Indicación de la relevancia - Distinción entre usuarios expertos e inexpertos - Navegación por similitud - Recuperación por palabras clave introducidas por el usuario - Recuperación de imágenes similares a partir de otra imagen - Recuperación de atributos visuales - Opciones de ponderación de los atributos de recuperación
8. Diseño estético y minimalista:	<ul style="list-style-type: none"> - Diseño atractivo - Sin elementos irrelevantes o innecesarios
9. Ayuda a los usuarios:	<ul style="list-style-type: none"> - Mensajes de error (indican problema y sugieren solución)
10. Documentación de ayuda:	<ul style="list-style-type: none"> - Documentación didáctica - Fácil acceso a la documentación

Tabla 2. Criterios empleados en análisis de usabilidad de interfaces de sistemas CBIR

De estos diez criterios, nos centraremos en los cuatro resaltados en la tabla anterior para describir algunas GUI relevantes de sistemas CBIR. A partir del análisis de estos puntos, veremos las virtudes y las carencias de las interfaces desarrolladas hasta el momento.

2.3 Tecnología utilizada en la implementación de las GUI

Antes de repasar con más detenimiento las GUI, es necesario prestar atención a los entornos donde se implementan las interfaces, y en consecuencia la tecnología empleada para su desarrollo, que determina en gran medida las posibilidades que ofrecerá la herramienta. A este nivel, podemos definir dos categorías de interfaz diferenciadas:

1. **INTERFACES COMO WEBS (WEB-LIKE INTERFACE):** se construyen estrictamente para un entorno web. Propuestas como Blobworld, MARS, GazoPa y Similar Images de Google Labs siguen este prototipo de interfaz.



Figura 7. Interfaz web-like de Similar Images de Google Labs

2. **INTERFACES COMO APLICACIÓN DE VENTANAS (*WINDOW-LIKE INTERFACE*):** su diseño reproduce la clásica aplicación con ventanas. Aplicaciones como ImageFinder, Emir o NeTRA siguen esta tendencia.



Figura 8. Ventana de la interfaz de ImageFinder

La mayoría de sistemas CBIR soportan una interfaz *web-like*, ya que los sistemas más populares surgidos actualmente se dedican a la búsqueda de imágenes en la web. Incluso los productos comerciales suelen poner a disposición de sus potenciales clientes demos on-line para probar el producto, como hace Attrasoft con su producto estrella ImageFinder y su versión demo AttraSeek⁴ disponible en la web.

Las interfaces de usuario en la web se basan en el soporte proporcionado por HTML (el lenguaje de marcado de descripción de documentos utilizado por las páginas web). Las más sencillas utilizan un formulario para los datos de entrada y una lista o tabla de resultados con las imágenes obtenidas como datos de salida.

⁴ Demo de AttraSeek: <http://www.imagequery.net> o <http://attraseek.com/>



Figura 9. Formulario de petición de búsqueda de CIRES en la web.

Las más complejas aumentan las funcionalidades de la aplicación con lenguajes de programación como Java que permiten asociar programas con los componentes de una página a través de los applets⁵.

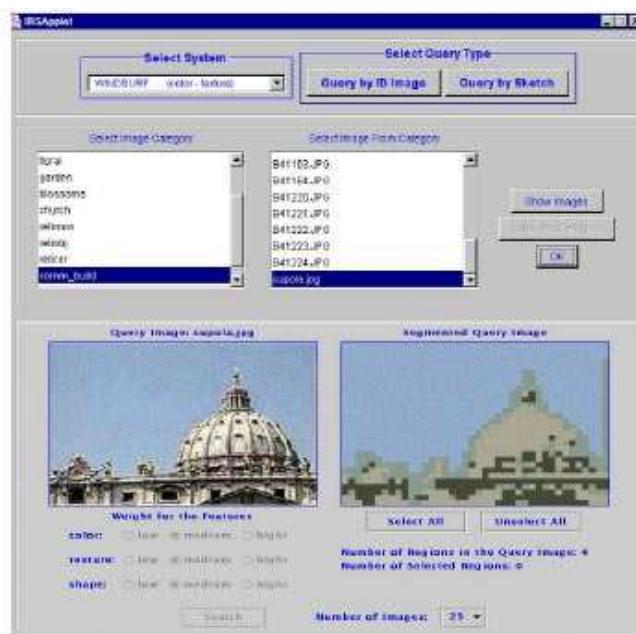


Figura 10. Applet de la interfaz de MuSIQUE⁶

⁵ Un *applet* es un componente de una aplicación escrita en Java que se ejecuta en el contexto de otro programa, por ejemplo un navegador web. (<http://es.wikipedia.org/wiki/Applet>)

⁶ MuSIQUE (*MULTI-System Image Querying User interfacE*), desarrollado en Laboratory of Advanced Research on Computer Science at the DEIS Dept. of University of Bologna.

2.4 Tipología de GUI para sistemas CBIR

La historia de las GUI va unida indiscutiblemente a la evolución de la tecnología. Antes de la explosión tecnológica que se dio en los años 70, los ordenadores eran manejados por personas muy especializadas. El hombre se adaptaba en su totalidad a la máquina, y no al revés (la antítesis de la usabilidad). A partir de los años 80, la proliferación del ordenador personal dirigido a un usuario final no experto en informática (para uso comercial, administrativo y empresarial), revoluciona esta área. Surge la necesidad de crear herramientas que faciliten el trabajo con el ordenador: **“el usuario no quiere utilizar una aplicación, quiere hacer su trabajo de la forma más sencilla y rápida posible”** [7].

Estos dos tipos de usuario tan diferenciados marcan las dos tendencias existentes en tipos de GUI de sistemas CBIR desarrollados hasta el momento. Por un lado, encontramos las interfaces dirigidas a los académicos e investigadores de este campo, que utilizan interfaces más pobres en diseño y usabilidad, pero más complejas en su manejo y en sus funcionalidades. Por otro, encontramos aplicaciones destinadas a un público más amplio, productos más comerciales enfocados a un usuario no experto, con diseños más trabajados y atractivos, en general más usables y a su vez más sencillos.

Esta diferencia de enfoque en el diseño según el usuario tipo de la aplicación nos servirá para establecer una clasificación de las interfaces que presentaremos a continuación. Debido a la coetaneidad de la mayoría de aplicaciones existentes (todas se desarrollaron en los últimos 15 años), esta línea de análisis es más interesante que el clásico análisis cronológico.

2.4.1 Interfaces de sistemas CBIR comerciales

De manera genérica, vamos a aplicar el calificativo de **interfaces comerciales** a aquellas aplicaciones que están desarrolladas por empresas y destinadas a su comercialización. La mayoría de estas empresas inicialmente se preocupan de implementar los sistemas de búsqueda que posteriormente los clientes adaptarán a sus productos.

Actualmente una gran parte de estas aplicaciones proliferan en la web, presentadas como valor añadido a los tradicionales buscadores de imágenes que utilizan como métodos de búsqueda la recuperación basada en texto. Este método anota las imágenes a través del texto HTML del documento que las contiene, basándose en la suposición de que una imagen en una página web está semánticamente relacionada con el texto que la rodea.

El negocio en la web viene de la mano de compañías especializadas en proveer de contenido visual a determinados consumidores (generalmente profesionales del sector audiovisual). Estas empresas disponen de grandes colecciones de imágenes, generalmente catalogadas manualmente, y utilizan sistemas propietarios para indexar y recuperar estas imágenes a través de palabras clave (las colecciones suelen ser actualizadas periódicamente). Getty Images⁷, Corbis⁸ o Masterfile⁹ son algunos ejemplos, y algunas de ellas como Masterfile, ya incorporan en sus buscadores el método de consulta por imagen ejemplo.

Pero este tipo de empresas es solo una pequeña muestra de las posibilidades de negocio de estos sistemas de búsqueda. Tiendas *on-line* como Like.com [16] son propuestas innovadoras, que apuestan por el uso de este tipo de tecnología orientado a un público masivo, el consumidor particular.

Veamos algunos ejemplos de interfaces comerciales y sus aplicaciones.

- **QBIC (Query By Image Content)**

Desarrollador: IBM Almaden Research Center, San Jose, CA.

URL: <http://www.qbic.almaden.ibm.com/>.

Año: 1995

⁷ <http://www.gettyimages.com>

⁸ <http://pro.corbis.com/>

⁹ <http://www.masterfile.com/info/products/simsearch.html>

El sistema QBIC desarrollado por IBM fue uno de los sistemas CBIR pioneros junto con PhotoBook de MIT¹⁰. El objetivo de IBM era vender simplemente el sistema de búsqueda, por lo que no se preocupó demasiado en diseñar una buena GUI para complementarlo. En la figura 10 vemos la interfaz de búsqueda que utilizó QBIC durante años.

- **PRESENTACIÓN DE LA INFORMACIÓN**

Es una interfaz *web-like*, con el formato de formulario que posiciona la área de consulta en la parte superior de la pantalla y la área de resultados en la parte inferior.



Figura 11. Interfaz de búsqueda de QBIC

¹⁰ Véase apartado 2.4.2 Interfaces de sistemas CBIR académicas.

▪ **TIPOS DE CONSULTA**

QBIC permite consultas a través de imagen, dibujos realizados por el usuario y/o características de color o patrones de textura. En los dos últimos casos, el usuario determina los colores o las texturas desde un muestrario. El porcentaje de color deseado es ajustable con una barra deslizante.

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

Los resultados con mejores puntuaciones son presentados en orden decreciente según su similitud con la consulta y, opcionalmente, presenta la puntuación obtenida al lado de la imagen. Incluye una lista superior de utilidades disponibles (a modo de menú de opciones) para ejecutar sobre cualquiera de las imágenes resultado. La navegación sobre los resultados se limita a los botones “Previous” y “Next” de la esquina superior derecha del área de resultados.

▪ **RELEVANCE FEEDBACK**

Cualquier imagen resultado puede ser utilizada como punto de partida en una nueva consulta para refinar la búsqueda.

▪ **VALORACIÓN DE LA GUI**

A nivel estético, la interfaz no destaca por un diseño atractivo ni intuitivo, el usuario debe emplear cierto tiempo en familiarizarse con la disposición de los elementos en la pantalla y su significado. Los iconos de las funcionalidades se presentan en diferente orientación según las dimensiones de la imagen resultado, desconcertando al usuario sobre su función y utilidad. La información sobre la consulta realizada aparece a modo de recordatorio en la parte inferior de la pantalla, quedando separada de la zona de formulación de la consulta (esta información debería permanecer en la parte superior).

A nivel funcional, la aplicación es bastante flexible y eficaz, permite al usuario definir diferentes modalidades de consulta, combinarlas y refinar el resultado con nuevas búsquedas a partir de los primeros resultados. El punto fuerte de QBIC es su potente motor de búsqueda, que actualmente se utiliza en múltiples aplicaciones.



Figura 12. QBIC integrado en la web del State Hermitage Museum

▪ EJEMPLO DE APLICACIÓN

El sistema QBIC está integrado en la web del State Hermitage Museum¹¹ de St. Petersburg (Russia), a través de applets de Java que permiten realizar búsquedas sobre la colección digital del museo a través del espectro de color (*QBIC Colour Search*) o de formas geométricas (*QBIC Layout Search*) de las obras. La aplicación combina consultas de características visuales y texto. El objetivo es que el usuario utilice, en una primera instancia, las clasificaciones temáticas y vaya afinando la búsqueda con las técnicas de recuperación visuales. Los resultados son listados según la similitud con los criterios de búsqueda, y cada imagen se presenta acompañada de su posición, nombre de la obra y una breve descripción.

¹¹ <http://www.hermitagemuseum.org/fcgi-bin/db2www/qbicSearch.mac/qbic?selLang=English>

Clicando sobre una imagen resultado permite visualizar diferentes tamaños de la imagen, realizar zooms sobre los detalles y realizar nuevas búsquedas a partir de esa imagen.

- **Excalibur Visual RetrievalWare**

Desarrollador: Excalibur Technologies Corporation.

Año: 1997-1999

Excalibur Corp. desarrolló software para crear aplicaciones de manipulación de imágenes digitales y su contenido visual, extracción de características, indexación y recuperación basada en contenido. Al igual que IBM, Excalibur priorizó las técnicas de búsqueda en detrimento del diseño de una GUI de soporte para su tecnología. Veamos brevemente, un ejemplo de interfaz de su sistema de búsqueda.

- **PRESENTACIÓN DE LA INFORMACIÓN**

La distribución de los elementos no es ordenada. La área de resultados se encuentra en la izquierda de la pantalla y la definición de la consulta se realiza a través del pequeño formulario de la derecha. Para ejecutar una búsqueda, el usuario debe realizar los siguientes pasos: en primer lugar ha de especificar los parámetros y su relevancia en la parte derecha de la interfaz, y a continuación debe seleccionar una de las imágenes presentadas en la parte izquierda de la pantalla para lanzar la consulta, invirtiendo el orden lógico de las zonas de atención del usuario, que por inercia, focaliza su atención primero en la parte izquierda de la pantalla y luego en la derecha.

- **TIPOS DE CONSULTA**

El método de búsqueda empleado es una imagen ejemplo. La consulta se realiza especificando la importancia de los atributos visuales en la búsqueda (color, forma, textura, brillantez, estructura de color, etc.). Permite ajustar el peso de estos atributos a través de un combo desplegable con los valores aceptados por el sistema.



Figura 13. Interfaz de ejemplo del kit de desarrollo de Excalibur

- **VISUALIZACIÓN DE LOS RESULTADOS**

En esta interfaz, los resultados son mostrados sin ningún orden establecido, por lo que el usuario no conoce cuál es el grado de similitud de cada imagen visualizada con la consultada. Además, no se presenta ningún tipo de información sobre las imágenes.

- **RELEVANCE FEEDBACK**

Cualquier imagen resultado puede ser utilizada como punto de partida en una nueva consulta para refinar la búsqueda.

- **VALORACIÓN DE LA GUI**

La interfaz no posee un diseño atractivo, ni usable, ni excesivamente funcional. No existe ningún elemento diferenciador de las áreas de consulta y resultados, el usuario debe ir de derecha a izquierda continuamente para trabajar con la aplicación y no se aporta información alguna sobre los resultados.

- **GazoPa Similar Image Searcher**

Desarrollador: Hitachi.

URL: <http://www.gazopa.com>

Año: 2008

GazoPa es de los proyectos más interesantes realizados recientemente. Desarrollado por la empresa Hitachi, es otra de las aplicaciones que utiliza QBIC de IBM como sistema de búsqueda inteligente. Se encuentra en fase beta, pero permite descubrir el potencial de este tipo de buscadores en el mercado y la clara tendencia actual a proporcionar herramientas cada vez más elaboradas y orientadas al consumidor de contenidos de la web (un público masivo).



Figura 14. Interfaz de GazoPa

- **PRESENTACIÓN DE LA INFORMACIÓN**

Es una interfaz muy pensada para la web, que utiliza el formato de formulario para realizar la búsqueda en la parte superior de la pantalla, claramente diferenciada del área de resultados en la parte inferior. La

organización de la información está realizada de forma lógica y clara para el usuario, que tiene accesible el área de consulta en todo momento.

▪ **TIPOS DE CONSULTA**

GazoPa permite consultas a través de imagen (proporcionada por el usuario o a partir de su URL), dibujos realizados por el usuario y/o palabras clave. Permite combinar los métodos de consulta y establecer opciones de búsqueda (filtros) con los parámetros disponibles en el recuadro situado bajo el panel de búsqueda superior resaltado con el fondo gris.

Además de imágenes, también puede analizar imágenes congeladas de un vídeo y buscar en la web otros vídeos que sean similares (a partir de imágenes clave de vídeo (*keyframes*)). El buscador incluye su propia herramienta de dibujo, basada en Flash (fig. 15), que permite realizar un boceto rápido o seleccionar una parte de una imagen para realizar la consulta.



Figura 15. Herramienta de Trimming¹² de GazoPa

Como aspecto negativo (aunque quizás sería más oportuno decir “no tan bueno”), GazoPa no permite ajustar los criterios de búsqueda numéricamente o a través de paletas de color, y el usuario no puede llegar a apreciar las diferencias entre las opciones de búsqueda llamadas “Moderate1” y “Moderate2” (la puntuación de los resultados se obtiene a través de una media, pero *moderate2* da prioridad a la composición).

¹² Herramienta de recorte para imágenes.

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

GazoPa permite dos tipos de visualización de los resultados, en modo parrilla (por defecto) y en modo lista. El modo parrilla muestra la información asociada a la imagen resultado de una forma más reducida: los datos que incluye son el grado de similitud con la consulta a través de una barra de progreso en la parte inferior de la imagen, la URL donde se encuentra y los iconos de funcionalidades disponibles (nueva búsqueda, búsqueda por cara, guardar el resultado o visualizar la imagen o vídeo original). La vista en modo lista muestra más información, como un texto descriptivo y el tamaño.



Figura 16. Resultados en modo lista en GazoPa

La navegación se realiza a través de la paginación inferior de la página web, como en cualquier buscador de la web.

Como funciones extra, ofrece unas pestañas en la parte superior del área de resultados para elegir entre tres tipos de resultados: la pestaña “Search Result” que muestra todos los resultados encontrados, la pestaña “Video” que filtra por los resultados de tipo vídeo y la pestaña “Amazon” que permite visualizar productos a la venta en esta tienda on-line con imágenes similares a la consulta.

▪ **RELEVANCE FEEDBACK**

Cualquier imagen resultado puede ser utilizada como punto de partida en una nueva consulta para refinar la búsqueda, pero no solo para encontrar nuevas imágenes similares, sino que GazoPa incorpora un buscador de caras, disponible para algunas imágenes.

▪ **VALORACIÓN DE LA GUI**

A nivel estético, GazoPa es un producto elaborado y atractivo. El logo recuerda ligeramente al de Google, el buscador más famoso de la web, pero su diseño es muy superior al de cualquiera de las herramientas de Google. Tanto la distribución de los elementos como el diseño de las diferentes áreas e iconos son intuitivos (por ejemplo, los vídeos son diferenciados de las imágenes con un marco de rollo de película, ver figura 16) y facilitan la utilización de la aplicación.

A nivel funcional, la inclusión de la herramienta de dibujo y recorte/retoque de imágenes en Flash, es otro atractivo que sorprende al usuario y le proporciona un valor añadido. Además, dispone de un *plugin*¹³ para Firefox y de un *bookmarklet*¹⁴. También supone una gran novedad la posibilidad de buscar dentro del portal Amazon, para unir búsqueda y compra.

▪ **Similar Images (Google Labs)**

Desarrollador: Google.

URL: <http://similar-images.googlelabs.com/>

Año: 2009

Recientemente Google ha lanzado **Similar Images**, como funcionalidad extra a su clásico buscador de imágenes en la web. La gran ventaja de esta herramienta es que el usuario ya está familiarizado con la estética y el entorno de buscador creados por Google. Su atractivo no radica en el diseño de la aplicación sino en su fácil manejo (dominado por todos sus usuarios) y la garantía que aporta la marca Google.

¹³ Los *plugins* permiten al navegador realizar funciones específicas, a partir de aplicaciones que se instalan de forma modular al programa principal. (<https://addons.mozilla.org/es-ES/firefox/browse/type:7>)

¹⁴ Un *bookmarklet* es un marcador que, en lugar de apuntar a una dirección URL, hace referencia a una pequeña porción de código JavaScript para ejecutar ciertas tareas automáticamente, como por ejemplo abrir una consulta directamente en un motor de búsqueda. (<http://es.wikipedia.org/wiki/Bookmarklet>)

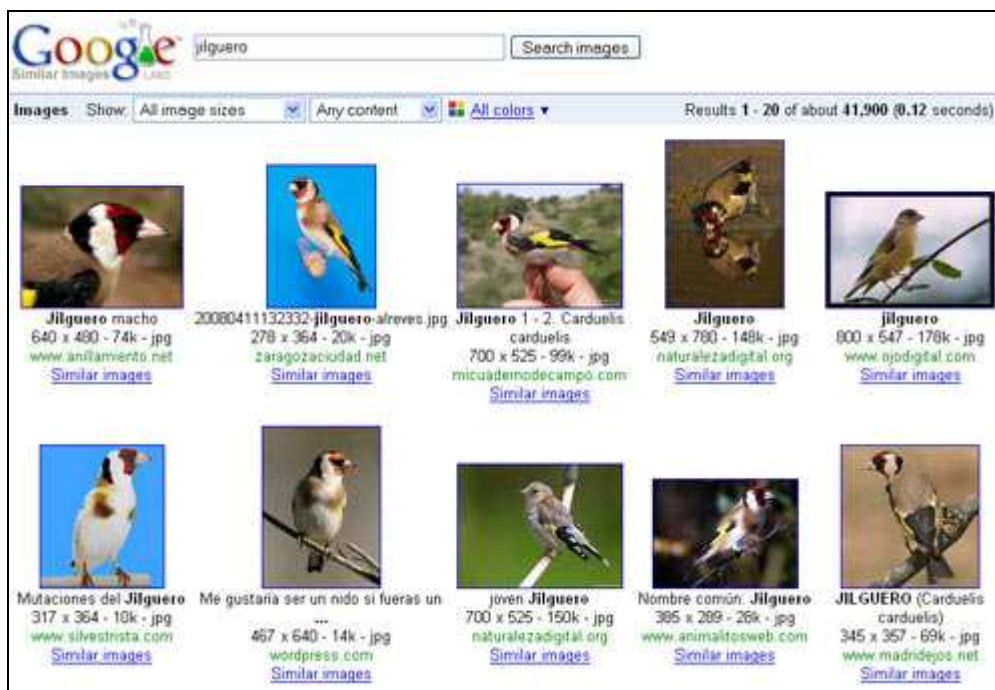


Figura 17. Similar Images de Google Labs

No obstante, después de analizar GazoPa, la interfaz del buscador de imágenes de Google no supone ningún referente a tener en cuenta en este tipo de GUI, pero es necesaria su mención debido a la repercusión que su aparición supone en el ámbito de la inclusión de sistemas CBIR en buscadores de imágenes convencionales. La aplicación *Similar Images* es muy limitada, ya que aprovecha el diseño del buscador de imágenes basado en texto (punto de entrada de la búsqueda, no permite partir de una imagen ejemplo) y añade la funcionalidad de consultas por imagen a los resultados (aunque no disponible para todas las imágenes resultado). La visualización de los resultados no obedece a ningún criterio o orden establecido.

A su favor, encontramos una serie de filtros para refinar la búsqueda y obtener resultados más satisfactorios, seleccionando el tamaño, el tipo de contenido o el color predominante en la imagen.

- **Picollator**

Desarrollador: Recogmission, LLC.

URL: <http://www.picollator.com/>

Año: 2008

Picollator es un buscador de imágenes especializado en el reconocimiento facial desarrollado por la empresa de origen ruso *Recogmission LLC*¹⁵. Este producto también se encuentra en una versión beta en la web y merece especial atención por su diseño sencillo pero muy cuidado y sus funcionalidades bien definidas.

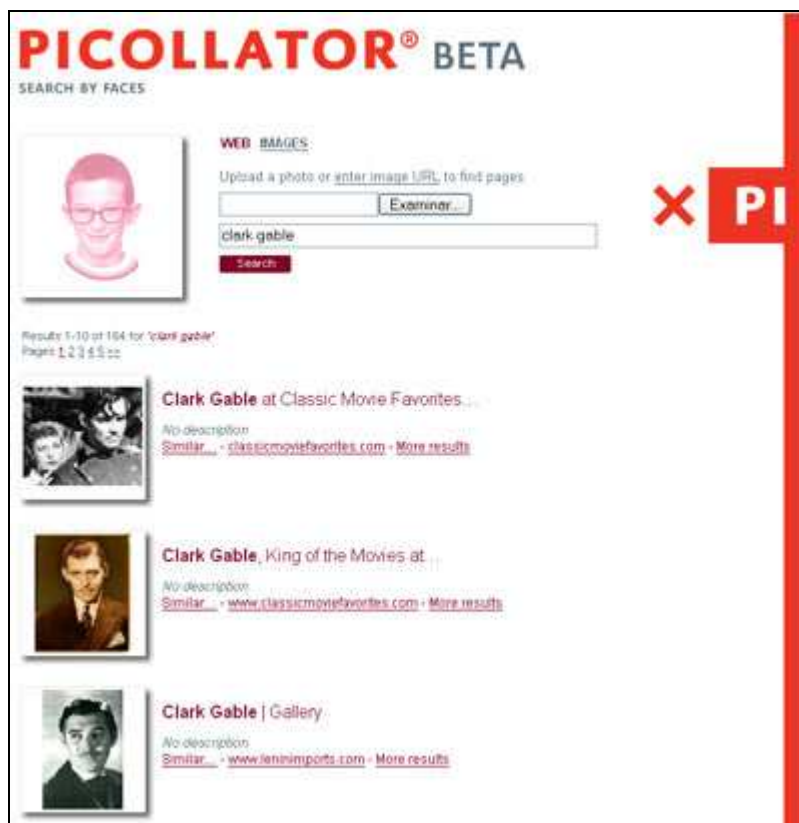


Figura 18. Picollator, buscador de caras.

▪ **PRESENTACIÓN DE LA INFORMACIÓN**

Como todo producto web, esta interfaz dispone su área de consulta en la parte superior de la pantalla, a modo de formulario, y el área de resultados en la parte inferior.

▪ **TIPOS DE CONSULTA**

En *Picollator* destaca la posibilidad de configurar la consulta según el ámbito de búsqueda (la web o su catálogo de imágenes) y formular la búsqueda a través de texto, una imagen (particular o su URL) o con ambos métodos a la vez.

¹⁵ <http://www.recogmission.com/?lang=1>



Figura 19. Imagen desactivada y Buscador en el campo de texto en Picollator

Picollator dispone además dos funcionalidades de ayuda a la hora de confeccionar la búsqueda, ya que incluye un buscador en el campo de texto para introducir las palabras clave y las imágenes de consulta pueden ser activadas y desactivadas en función de las necesidades del usuario (con un icono de un tomate en rojo (imagen activa) o en gris (inactiva) en la esquina superior derecha de la imagen).

- **VISUALIZACIÓN DE LOS RESULTADOS**

Picollator puede presentar los resultados en forma de lista, colocando la información asociada al resultado en el margen derecho de la imagen (título, descripción, URL donde se encuentra, etc.), o en forma de parrilla, limitando esta información colocada bajo la imagen al grado de similitud y su URL.

Como extra, Picollator incluye la opción “More results” que muestra todas las imágenes indexadas por Picollator del mismo sitio web de la imagen resultado.

La navegación se realiza a través de la paginación colocada en la parte superior del área de resultados, en vez de en la parte inferior de la página.

- **RELEVANCE FEEDBACK**

Cualquier imagen resultado puede ser utilizada como punto de partida en una nueva consulta para refinar la búsqueda.

- **VALORACIÓN DE LA GUI**

Los puntos fuertes de esta interfaz son su enfoque de especialización en el reconocimiento facial y su diseño, atractivo y sofisticado, con un acertado uso de los colores y tratamiento de las imágenes resultado, y orientado a presentar la información al usuario de una forma ordenada, clara y agradable a la vista a partir de un estilo homogéneo en todos los elementos de la interfaz, y con detalles interesantes como el icono del tomate para activación/desactivación de imagen. También dispone de *plugin* para Firefox.

- *Piximilar, PixID, TinEye*

Desarrollador: Idée Inc.

URL: <http://labs.ideeinc.com/>

Año: 2008

Piximilar, PixID y *TinEye* son productos creados por la compañía Idée Inc., empresa especializada en software de reconocimiento avanzado de imágenes y búsqueda visual. Idée es otra muestra del creciente interés que los sistemas CBIR están adquiriendo en el mercado. Utilizan un sistema de búsqueda propietario llamado Visual Search.

PixID es un servicio de monitorización de imagen para identificar las imágenes que se utilizan en las publicaciones impresas y en Internet, especialmente dedicado al control del uso de imágenes (cumplimiento de licencias, malos usos, reclamar pagos por usos no autorizados, etc.).



Figura 20. Imagen de licencia libre identificada por PixID en la portada de un libro.

TinEye es una herramienta de búsqueda de imágenes en la web que es capaz de detectar dónde y cómo una imagen es utilizada (incluso si ha sido modificada). Existe una versión beta en la web (<http://tineye.com/>). Como podemos apreciar en la figura 21, *TinEye* reproduce el tipo de diseño para web analizado en herramientas presentadas anteriormente, por lo que no repetiremos sus características.

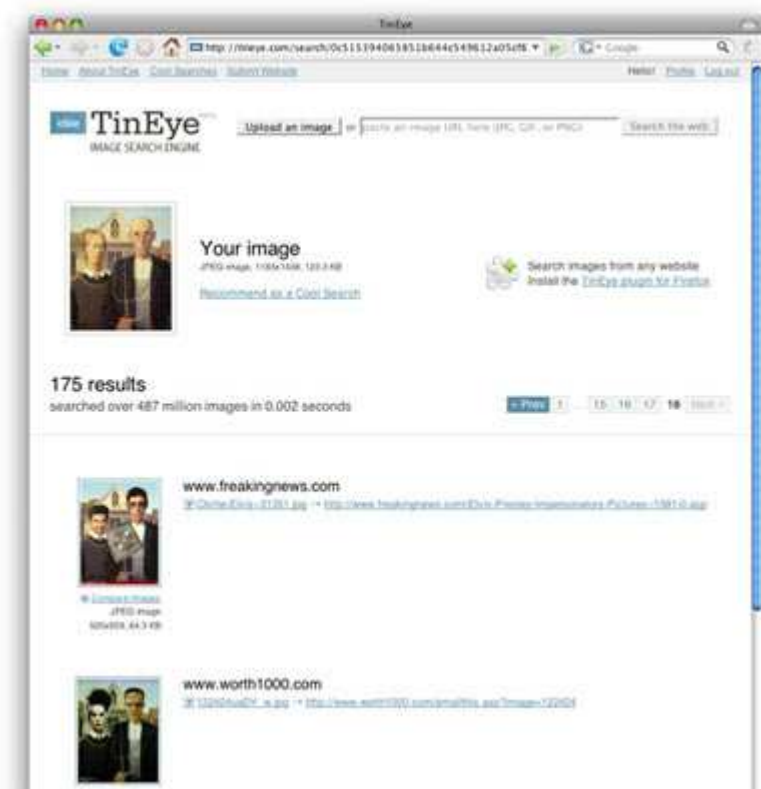


Figura 21. TinEye, visualización de resultados en lista.

Como novedad, sí que debemos destacar la posibilidad de visualizar los resultados no solo listados como en la mayoría de los buscadores, sino de forma gráfica en un mapa (figura 22, no disponible en la versión beta).

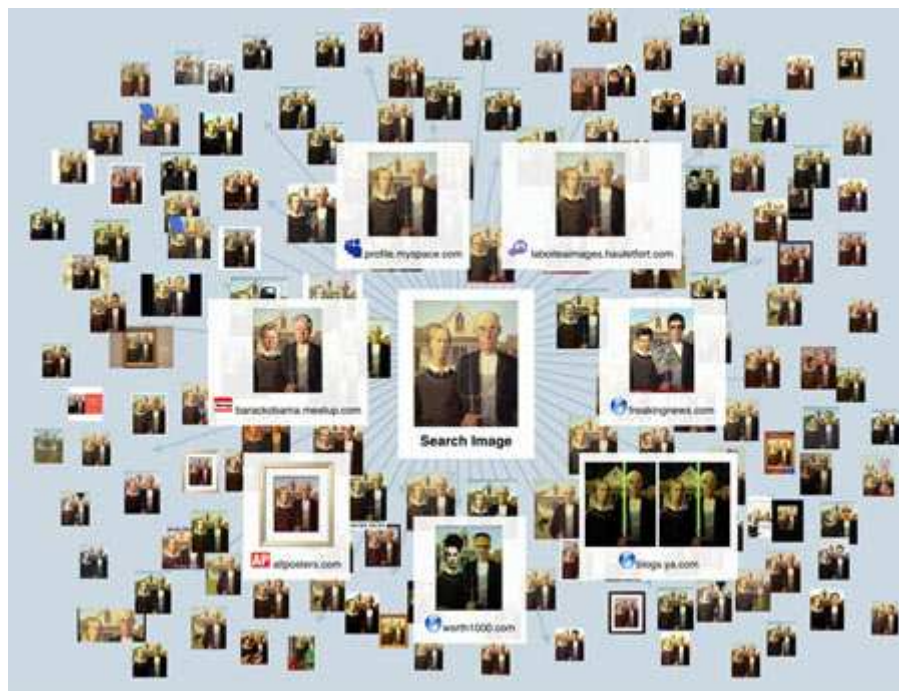


Figura 22. TinEye, visualización de resultados en mapa.

Además TinEye incluye una herramienta de comparación para ver cuánto se parecen los resultados encontrados con la imagen de consulta.



Figura 23. Pop-up con la herramienta de comparación de TinEye

Piximilar es una herramienta especializada en búsqueda de imágenes pertenecientes a una colección. Trabaja con el método de búsqueda por imagen similar o por selección de varios colores.

La diferencia destacable de las GUI de Piximilar con el resto de interfaces de buscadores de imágenes es la distribución horizontal de las áreas de consulta y resultados en vez de la típica en vertical. Además, altera el orden lógico de la ejecución de la consulta, ya que posiciona la parrilla de resultados a la izquierda y la área de resultados a la derecha, cuando por inercia, un usuario focaliza su atención primero en la parte izquierda de la pantalla y luego en la derecha.

Un ejemplo de aplicación actual de la tecnología Visual Search es el buscador *SimSearch* desarrollado por Idée para Masterfile¹⁶, proveedor de contenido visual en la red, que se dedica a vender imágenes bajo licencia para su uso comercial en los medios de comunicación, desde prensa escrita hasta sitios web.

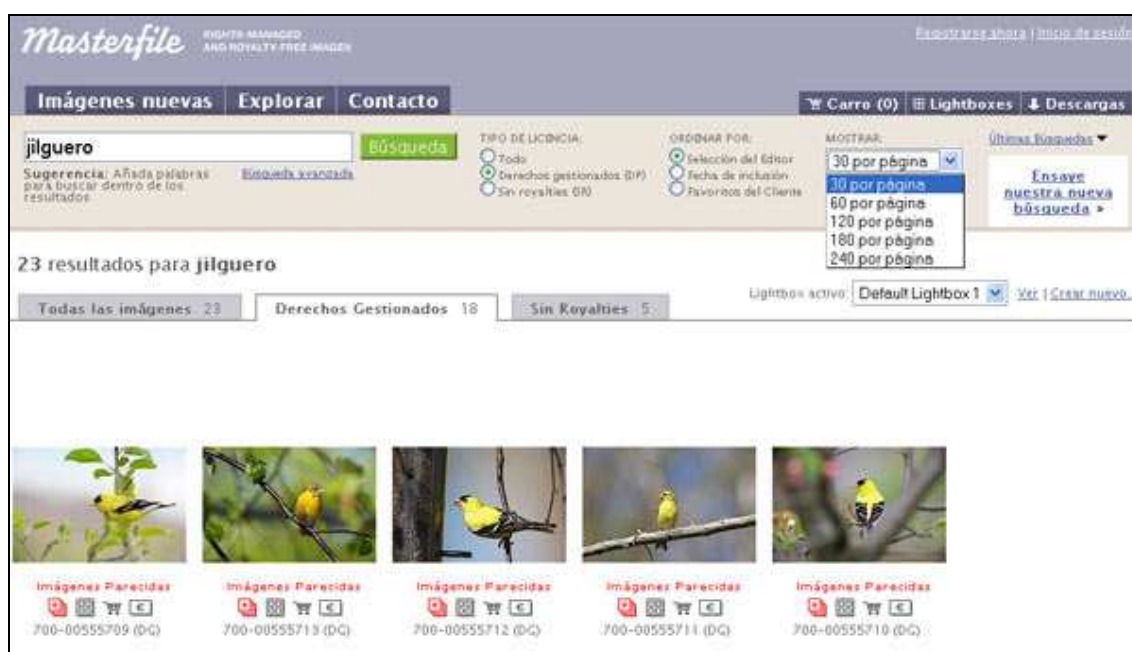


Figura 26. SimSearch en Masterfile

La interfaz del buscador de Masterfile vuelve a reproducir el esquema de los buscadores de imágenes para web descritos con anterioridad. Destaca la utilización de elementos gráficos orientados a la compra para mostrar la información de los resultados (como el icono del carrito o el de la moneda), un lenguaje visual fácilmente reconocible por el usuario y que agiliza el uso de la aplicación.

¹⁶ <http://www.masterfile.com/>

- **ImageFinder**

Desarrollador: Attrasoft Inc.

URL: <http://www.attrasoft.com/imagefinder70/>

Año: 1998-2006

ImageFinder es un sistema CBIR para Windows desarrollado por la compañía Attrasoft. Este sistema de recuperación de imágenes se comercializó en tres productos diferentes (*ImageFinder*, *Internet ImageFinder* y *ImageHunt*), que utilizaban la misma tecnología, y su única diferencia era el diseño de sus interfaces. Actualmente solo sobrevive ImageFinder.

La primera versión de ImageFinder data de 1998, y hoy se comercializa la versión 7.0. La evolución del producto es evidente tanto a nivel de diseño como de funcionalidades, aunque la estética de la aplicación se ha mantenido fiel a la primera versión, quedándose bastante anticuada hoy en día. ImageFinder es una herramienta compleja, no utilizada solamente para realizar búsquedas, sino que también permite el tratamiento y el procesado de imágenes, por lo que el destinatario final es un usuario experto, familiarizado con estas técnicas. Pero para su evaluación, nos centraremos sólo en aquellos aspectos relacionados con la búsqueda de imágenes.

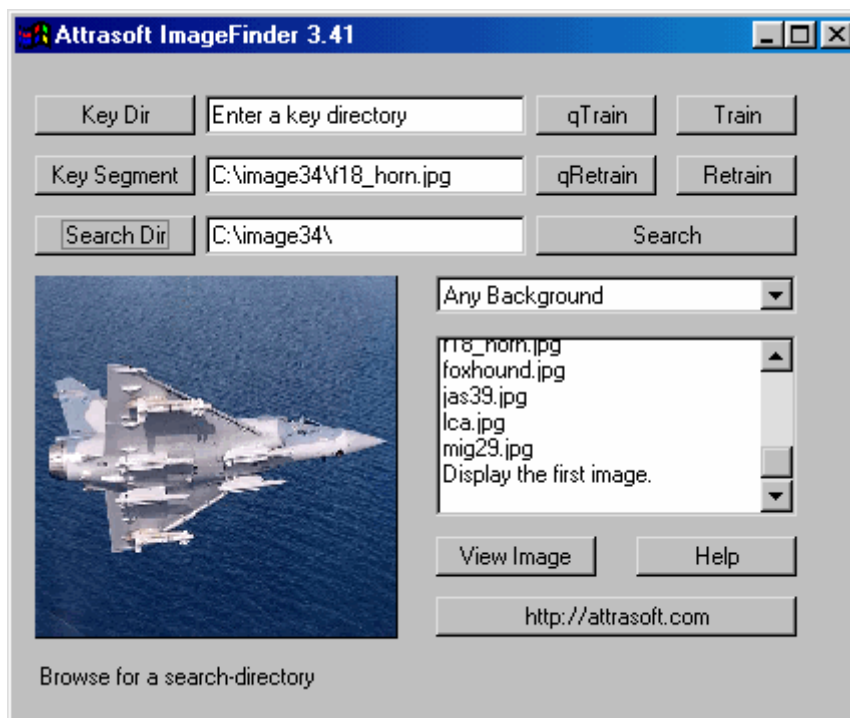


Figura 27. Interfaz antigua de ImageFinder

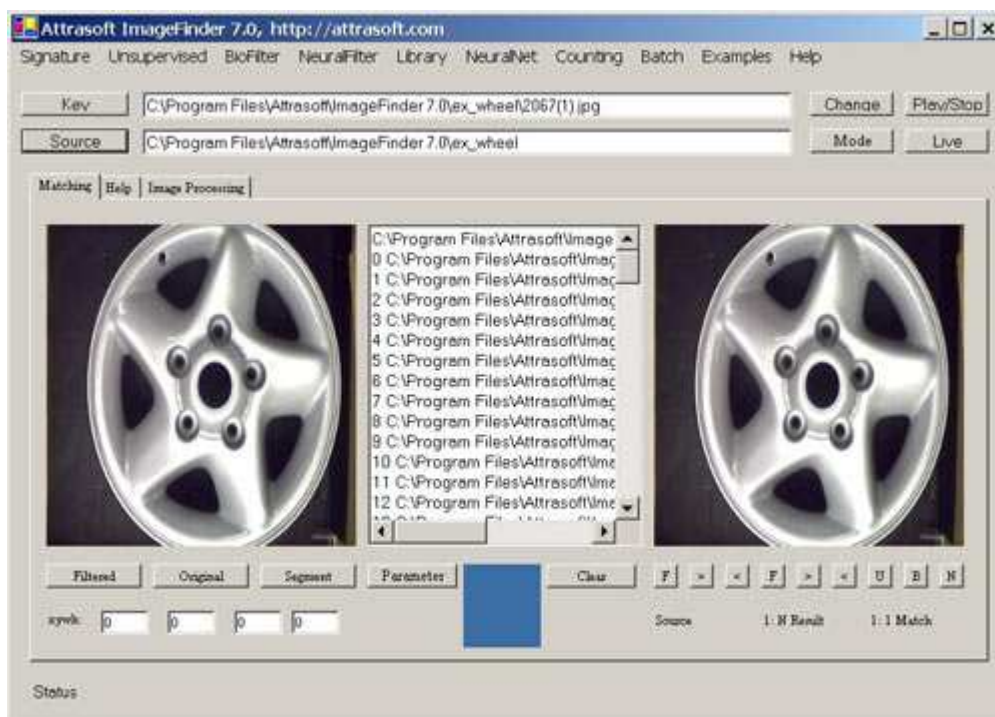


Figura 28. Interfaz actual de ImageFinder

- **PRESENTACIÓN DE LA INFORMACIÓN**

Esta aplicación adopta el diseño *window-like*, tan característico de las aplicaciones para Windows. No obstante, no aprovecha las facilidades que este tipo de aplicaciones brinda. La consulta y la visualización de los resultados se realiza en la mismo área, el panel de la pestaña “Matching”, no llegando a diferenciar correctamente estos dos pasos y dificultando una rápida ejecución de la búsqueda y visualización de los resultados.

Los elementos pertenecientes a la consulta se distribuyen entre la parte superior de la pantalla, donde el usuario selecciona el directorio de búsqueda y la imagen a buscar, y la parte izquierda del panel “Matching”, donde se visualiza la imagen seleccionada y se presentan una serie de botones con las opciones disponibles (configuración de filtros, parámetros, creación de segmentos, etc.). Cada botón abre una ventana que despliega las posibilidades de cada opción.

El cuadro de texto central tiene la función de ir informando al usuario sobre los pasos del proceso, y lista los resultados que se pueden ir visualizando en la parte derecha.

Al tratarse de una herramienta compleja, pensada para usuarios expertos, es necesario familiarizarse previamente con la aplicación y sus posibilidades antes de realizar un uso correcto de la misma.

▪ **TIPOS DE CONSULTA**

ImageFinder utiliza sólo el método de consulta por imagen ejemplo, pero el usuario puede escoger el ámbito de búsqueda y la fuente (un directorio, una base de datos, etc).

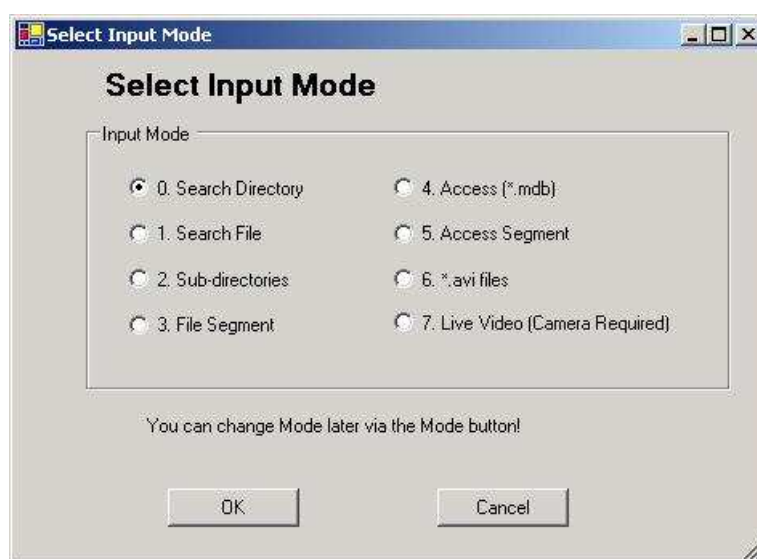


Figura 29. Ventana de selección de búsqueda de ImageFinder

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

Como hemos visto anteriormente, el área de visualización de resultados no ocupa un espacio diferenciado en la pantalla, y dificulta su fácil reconocimiento a simple vista. Sólo se muestra un resultado cada vez, es necesario utilizar las flechas de navegación inferiores para ir pasando de un resultado a otro, sin la posibilidad de tener una vista general completa o parcial de un conjunto de resultados. No obstante, permite comparar la imagen consultada (a la izquierda) y la imagen resultado (a la derecha), a un gran tamaño (el mismo para ambas).

Una opción interesante que nos ofrece es la posibilidad de guardar los resultados en un archivo de texto (.txt) o en formato web (en HTML).

- **RELEVANCE FEEDBACK**

Es necesario volver a seleccionar la imagen de consulta a través del botón “key” en la parte superior de la aplicación para realizar una nueva consulta. No dispone de ningún elemento para refinar las búsquedas.

- **VALORACIÓN DE LA GUI**

La interfaz presenta un diseño muy básico, típico de aplicaciones de ventanas antiguas. Es obvio, que prima su funcionalidad por encima de su diseño, y que no se ha tenido en cuenta ningún criterio de usabilidad para facilitar al usuario el trabajo con esta herramienta. ImageFinder integra diversas funcionalidades dentro de la misma aplicación, y la función de buscador de imágenes no parece tener excesiva relevancia dentro del conjunto, ya que no está muy trabajada.

- ***Like.com (Visual Shopping)***

Desarrollador: Riya, Inc.

URL: <http://www.like.com/>

Año: 2005-2009

Like.com es una tienda on-line que ha acuñado el término de *Visual Shopping* (compra visual). El portal se describe como “*the first true visual search engine, where the contents of photos are used to search and retrieve similar items*” [16]. Esta web es un buen ejemplo del futuro comercial de los sistemas de búsqueda de imágenes. La interfaz está completamente orientada al usuario, a facilitar su uso, que en definitiva es sinónimo de facilitar la compra.

A nivel estético, está cuidado hasta el último detalle, todos los elementos de la interfaz son atractivos y muy gráficos, con el claro objetivo de destacar los productos puestos a la venta. Destaca la inclusión de fotografías de personajes famosos, donde aparecen estrellas (icono del portal) sobre la ropa que visten y que permiten con un solo clic, iniciar la búsqueda de la prenda.

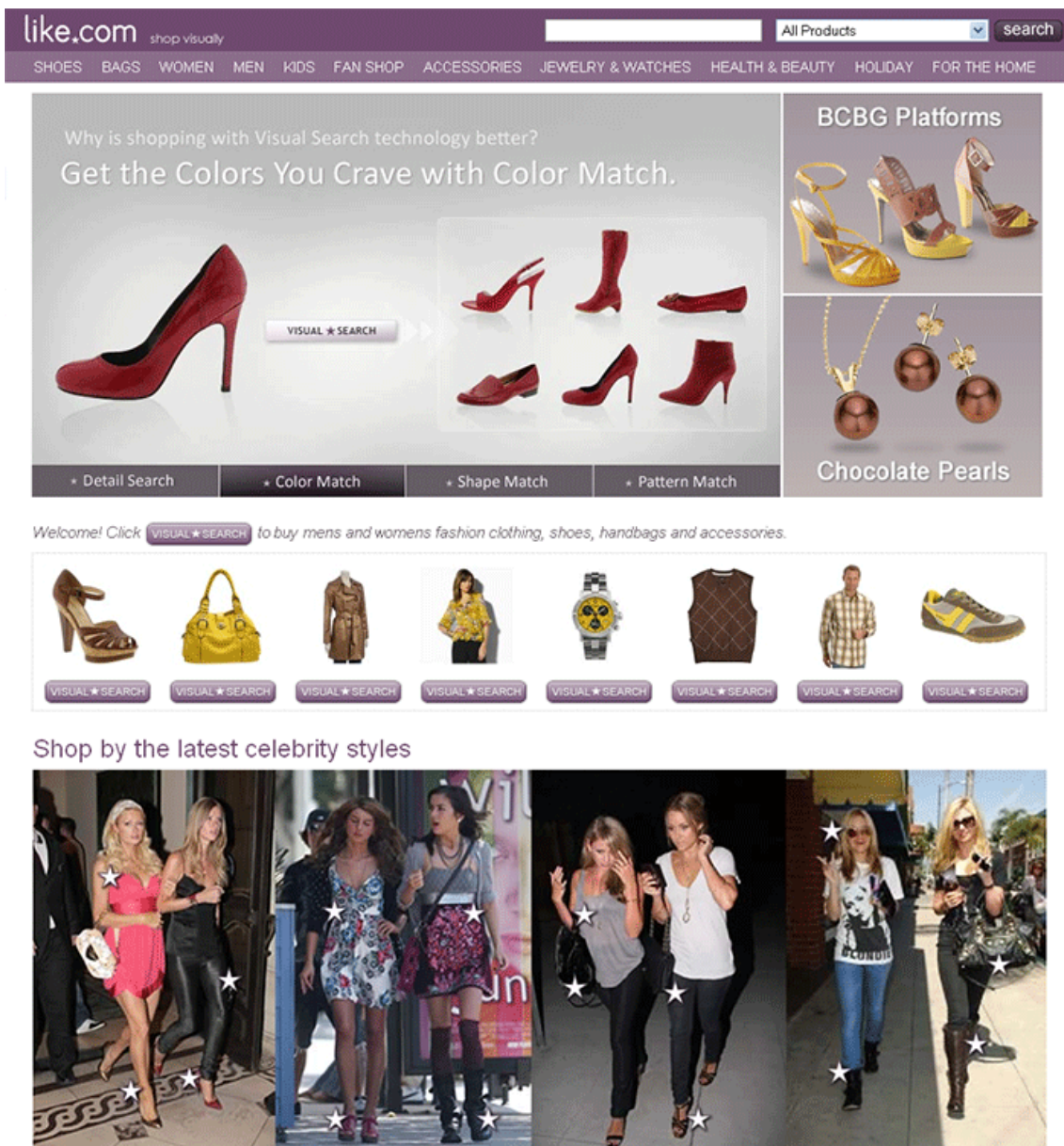


Figura 30. Portal de Like.com

▪ **PRESENTACIÓN DE LA INFORMACIÓN**

Esta interfaz web no presenta un área específica para formular la consulta, ya que la mayoría de elementos en pantalla permiten iniciar una búsqueda (botones, imágenes, menú superior con categorías, etc.). En la parte superior de la pantalla se incluye el típico buscador para web visible en todas las páginas del portal, que permite seleccionar una categoría y/o introducir texto para empezar a buscar.

▪ **TIPOS DE CONSULTA**

Like.com soporta la búsqueda de imágenes a través de texto e imágenes ejemplo de productos incluidos en el portal o de fotografías cargadas por el propio cliente. El usuario también puede especificar varias características a buscar, como el color, la forma, un patrón o textura e incluso algún detalle del artículo (por ejemplo una hebilla, cremalleras, etc.).

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

Los resultados pueden ser mostrados en forma de lista o parrilla, ordenados según el grado de similitud con la consulta. El usuario puede configurar el número de resultados por página que desea visualizar, y navegar por los resultados a través de la paginación incluida en el borde superior derecho de los resultados. En el margen izquierdo de la página, se ofrece la posibilidad de refinar la búsqueda seleccionando un color, estilo, textura, marca, material o algún detalle específico del artículo (ver figura 31).

▪ **RELEVANCE FEEDBACK**

Cualquier imagen resultado puede ser utilizada como punto de partida en una nueva consulta, y el usuario dispone de varias herramientas para refinar su búsqueda.

▪ **VALORACIÓN DE LA GUI**

La interfaz de Like.com es muy comercial, los protagonistas son los productos en venta, y el buscador es una mera herramienta, muy atractiva, para facilitar su compra (incita al usuario a jugar y experimentar). La información visual es básica en esta aplicación, que pone al servicio del cliente todo el potencial del sistema de búsqueda, cada opción y funcionalidad tienen como objetivo enganchar al usuario.

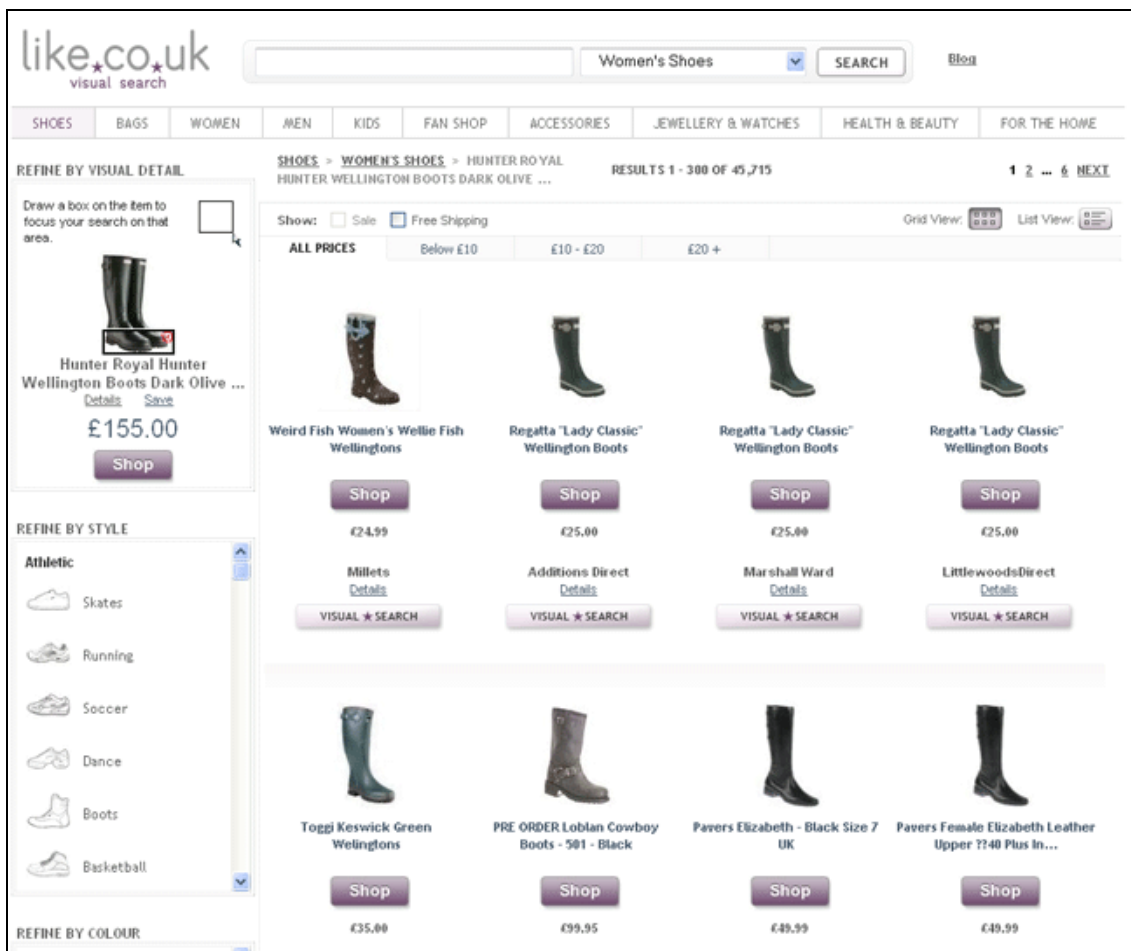


Figura 31. Visualización de resultados en el portal Like.com

2.4.2 Interfaces de sistemas CBIR académicos

En contraposición de las interfaces comerciales, otorgaremos el calificativo de **interfaces académicas** a aquellas aplicaciones desarrolladas por universidades y centros de investigación, que en principio, no tienen ningún tipo de interés en su comercialización. Por lo tanto, la interfaz de estos sistemas no serán especialmente atractivas en diseño, nos interesarán más a nivel funcional.

Para este tipo de sistemas, la web sigue siendo el banco de pruebas preferido, ya que permite una mayor difusión del trabajo realizado. Además, la mayoría de sistemas CBIR académicos elaboran proyectos completos, que abarcan desde herramientas para la anotación e indexación de las imágenes, hasta las técnicas de búsqueda.

Veamos algunas de las interfaces más destacadas en esta categoría.

- **Photobook**

Desarrollador: Vision and Modeling Group, MIT Media Laboratory, Cambridge, MA..

URL: <http://vismod.media.mit.edu/vismod/demos/photobook/index.html>

Año: 1994

El sistema Photobook se desarrolló como un conjunto de herramientas interactivas para la búsqueda y navegación de imágenes y de secuencias de imágenes. Es el primer sistema que subraya la importancia de la interactividad entre el motor de búsqueda y el usuario para conseguir resultados satisfactorios.



Figura 32. Interfaz de Photobook

- **PRESENTACIÓN DE LA INFORMACIÓN**

La interfaz de Photobook es del tipo *window-like*, aunque dispone de una versión web para demos on-line (actualmente no disponible). Se pensó en

la tecnología Motif¹⁷ para crear una interfaz realmente eficaz en términos de interactividad con el usuario [12]. El resultado es una interfaz aparentemente sencilla, pero que permite múltiples funciones que dificultan ligeramente su utilización.

Las funciones se reparten de forma coherente entre el margen izquierdo y derecho de la pantalla, dejando la parte central (más amplia), para la visualización de las imágenes, tanto de consulta como los resultados. En el margen izquierdo encontramos las opciones de configuración de la aplicación y de la consulta, como la selección del ámbito de búsqueda (la base de datos), el modo de visualización de las imágenes y el tipo de algoritmo (métrica) utilizado en la consulta. En el margen derecho, se sitúan en la parte superior las opciones principales de la aplicación, como inicializar el sistema, lanzar la búsqueda o guardar los resultados. En la parte inferior encontramos las opciones de navegación por los resultados, a través de páginas o ítems.

▪ **TIPOS DE CONSULTA**

El sistema permite formular la consulta a partir de imágenes (una o varias) y/o texto (realiza la función de filtro). El procedimiento a seguir suele ser la selección de una categoría por parte del usuario para delimitar el ámbito de búsqueda, que se puede complementar con búsqueda por texto. Según la configuración previa del sistema, mostrará unos primeros resultados, a partir de los cuales el usuario reiterará la consulta para afinar la búsqueda.

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

Los resultados son ordenados en formato parrilla en la parte central de la pantalla según el grado de similitud con la consulta realizada. El usuario puede navegar por las imágenes a través de los controles situados en el margen inferior derecho.

¹⁷ The Motif Interface is a windows-based graphical user interface that uses the X Windows system. Items on the screen appear three-dimensional and can be accessed with the keyboard or by clicking with the mouse. You can also use the mouse to resize or move application windows.

▪ **RELEVANCE FEEDBACK**

Una vez configurado el sistema, la tecnología Motif posibilita una rápida interacción con el usuario, que puede realizar búsquedas reiterativas a través de la selección de una o varias imágenes como nueva consulta. Con el botón izquierdo del ratón se selecciona la imagen, con el botón central se lanza la consulta y con el derecho se obtiene información de la imagen. El usuario puede elegir entre un gran repertorio de métricas, incluyendo un algoritmo que incorpora la retroalimentación a los demás algoritmos y les permite "aprender" del usuario el concepto de similitud de imagen.

▪ **VALORACIÓN DE LA GUI**

La interfaz de Photobook no presta ninguna atención a la estética de la aplicación, pero en cambio es una interfaz pensada para facilitar el trabajo del usuario. A nivel funcional, es una aplicación pionera, que destaca por la buena distribución de sus elementos, su usabilidad y la rápida ejecución de sus funciones.

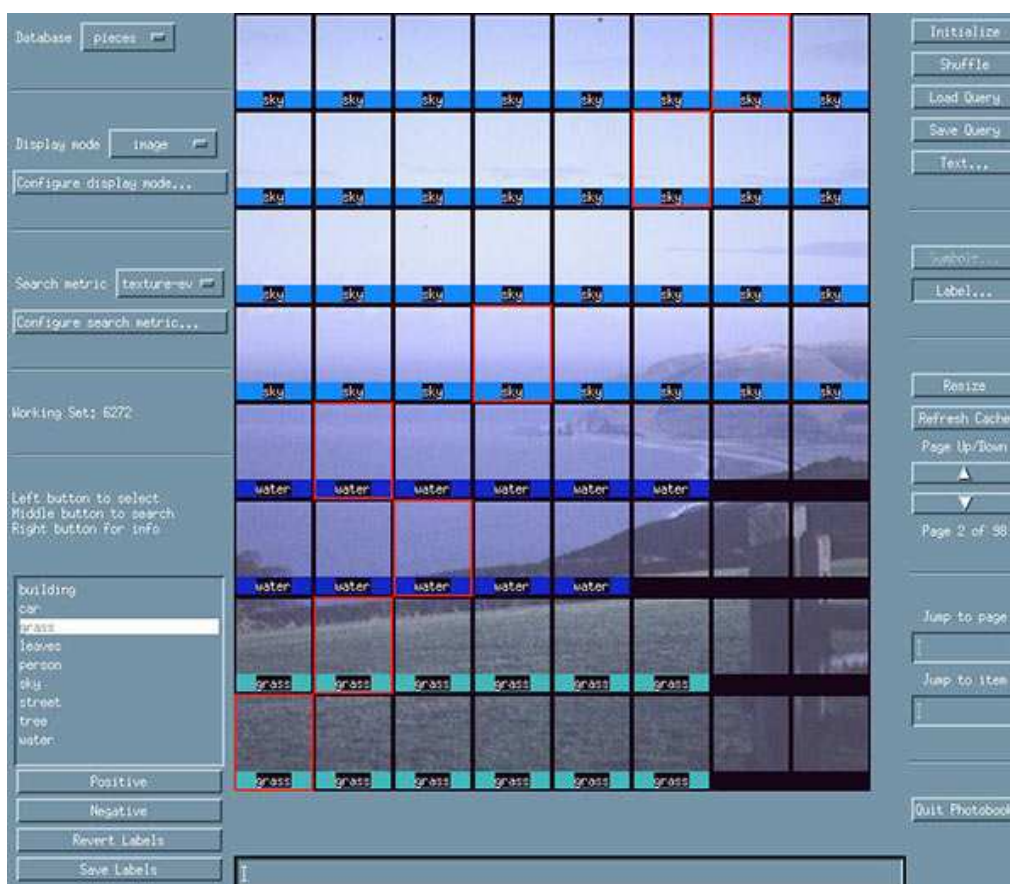


Figura 33. Interfaz de FourEyes anotando con etiquetas “Water”, “Sky” y “Grass”

El sistema de búsqueda de Photobook se complementa con la herramienta de anotación FourEyes, que utiliza el mismo tipo de interfaz (ver figura 33).

- **AMORE (Advanced Multimedia Oriented Retrieval Engine)**

Desarrollador: C & C Research Laboratories NEC USA, Inc.

Año: 1999

El sistema AMORE es uno de los primeros buscadores de imágenes para web que se preocupó por implementar un método de visualización de resultados más cuidado, tendencia que se extendería a partir del año 2000, cuando los sistemas de búsqueda de imágenes empiezan a mostrar más interés por el uso más eficiente del espacio de la pantalla [13].

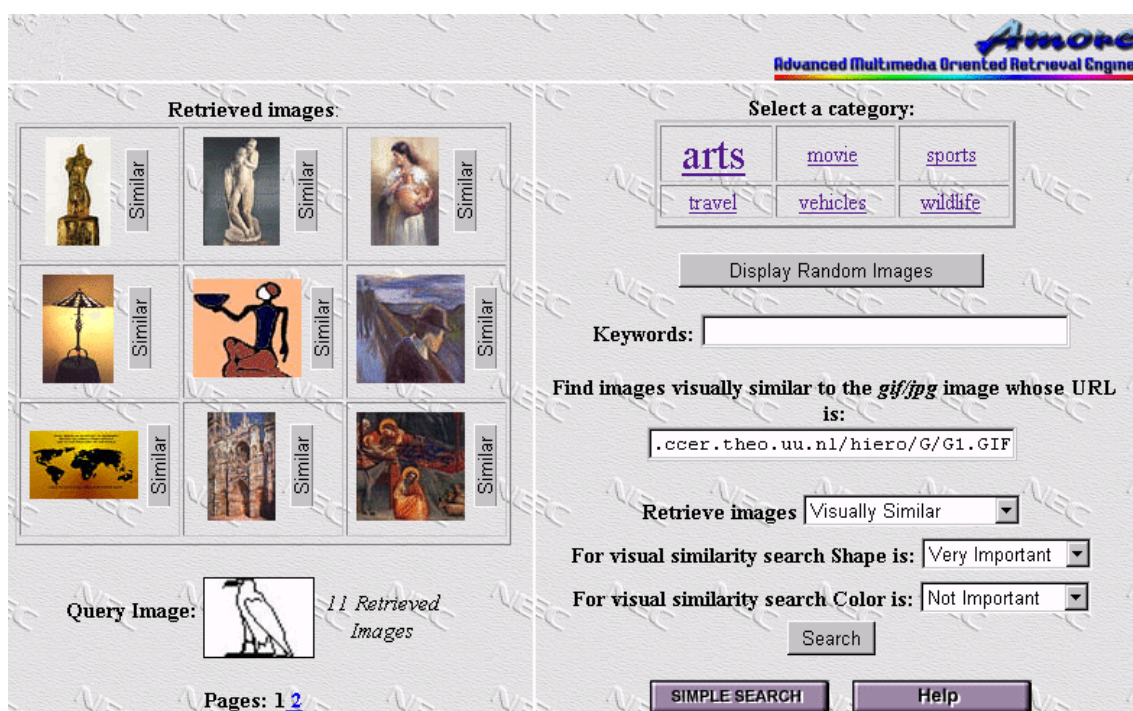


Figura 34. Interfaz del sistema AMORE

- **PRESENTACIÓN DE LA INFORMACIÓN**

AMORE es una interfaz *web-like*, que separa la área de consulta situada en la parte derecha de la pantalla, de la área de resultados de la parte izquierda a través de un marco blanco. Volvemos a encontrarnos con esta

distribución izquierda-derecha, menos usable por la tendencia del usuario a focalizar su atención primero en la zona izquierda de la página, donde debería comenzar el proceso de consulta, en vez de estar colocado en la parte derecha. Esto obliga al usuario a moverse de derecha a izquierda para realizar sus búsquedas. La consulta utiliza el formato de formulario, bien organizado verticalmente, para ir configurando los diferentes elementos que compondrán la petición de búsqueda a través del orden lógico que emplea el sistema. Primero se determina el ámbito de búsqueda a partir de una categoría, luego se escoge el método de consulta y por último se define el algoritmo y los criterios a utilizar.

▪ **TIPOS DE CONSULTA**

La interfaz soporta la recuperación de imágenes a través de palabras clave (texto), a partir de una imagen especificada con su URL o una combinación de ambos métodos. El usuario puede elegir la categoría de imágenes en donde realizar la búsqueda y ajustar la relevancia en la búsqueda del color y las formas de la imagen a través de un combo de opciones desplegables.

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

Los resultados son mostrados en una tabla como miniaturas, para aprovechar mejor el espacio disponible en pantalla, pero no se aporta ningún tipo de información sobre las imágenes. Éstas se visualizan sin un orden explícito. En la parte inferior del área de resultados se destina un pequeño espacio para recordar la imagen de consulta, importante para que el usuario determine su satisfacción con la búsqueda. La navegación se realiza a través de la paginación inferior en la página web, como en la mayoría de buscadores de la web.

▪ **RELEVANCE FEEDBACK**

Cualquier imagen resultado puede ser utilizada como punto de partida en una nueva consulta para refinar la búsqueda a través del botón “Similar” que acompaña a cada imagen.

- **VALORACIÓN DE LA GUI**

A nivel estético, esta interfaz no es nada atractiva aunque se aprecia la voluntad de crear una aplicación armoniosa, con dos áreas equilibradas y diferenciadas, y otorgando a sus elementos un tamaño apropiado.

A nivel funcional, hemos visto como se invierte la lógica natural de los pasos a realizar en la búsqueda con una orientación derecha-izquierda, aunque la organización vertical de la consulta es muy correcta.

- **MARS (Multimedia Analysis and Retrieval Systems)**

Desarrollador: Department of Computer Science, University of Illinois at Urbana-Champaign, further developed at Department of Information and Computer Science, University of California at Irvine, CA.

URL: <http://www.ifp.illinois.edu/~qitian/MARS.html>

Año: 1997-2003

Los objetivos del proyecto MARS son diseñar y desarrollar un sistema integrado de gestión de bases de datos y recuperación de información multimedia. El sistema CBIR de MARS destaca por ser el primero en introducir técnicas de *relevance feedback* en este tipo de aplicaciones, para intentar emular las necesidades del usuario [13].

- **PRESENTACIÓN DE LA INFORMACIÓN**

La GUI de MARS adopta el diseño *window-like* y está programada en el lenguaje de Microsoft Visual C++. El sistema fue implementado parcialmente en Java para su versión on-line (actualmente no disponible).

La interfaz que ofrece MARS supone una evolución en el tratamiento visual de la información de la pantalla. En un primer nivel de distribución vertical, la zona de resultados se muestra en la parte superior de la aplicación, y la zona de configuración y formulación de la consulta en la parte inferior.

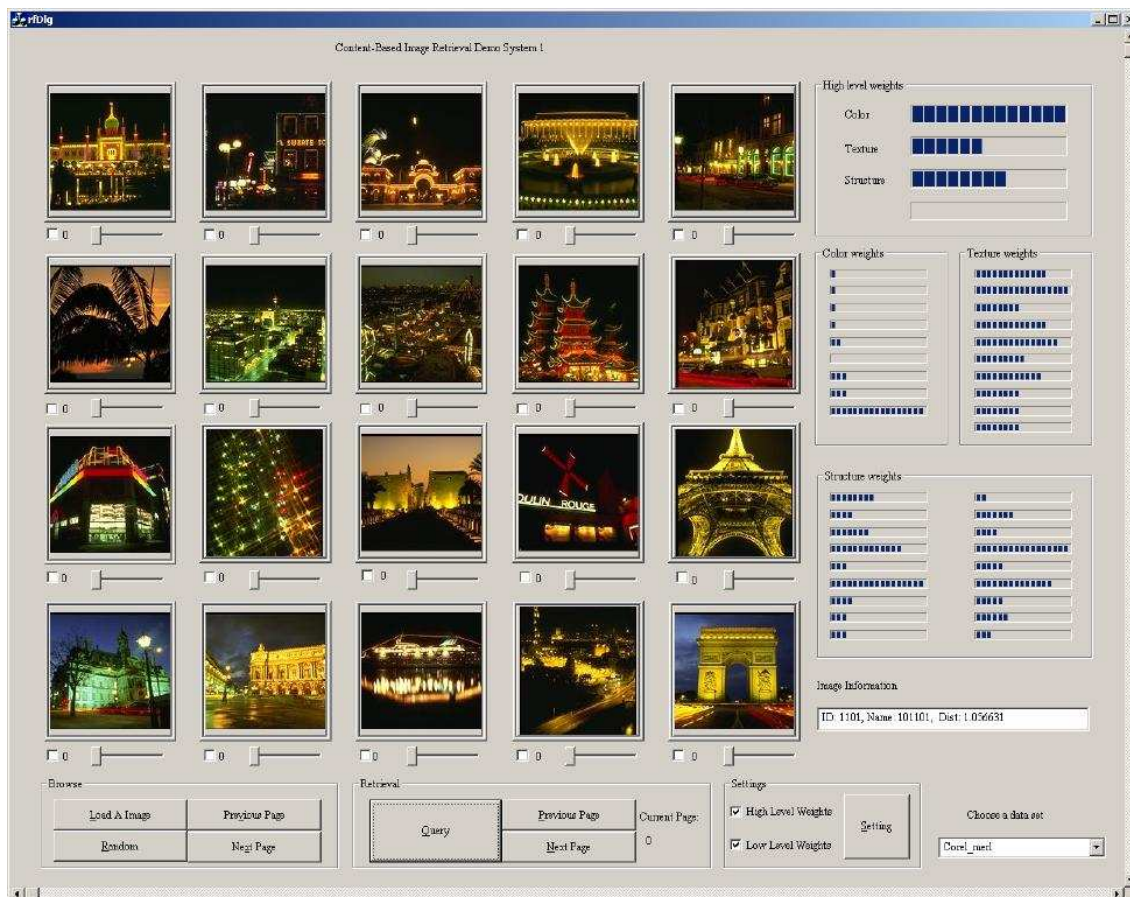


Figura 35. Interfaz del sistema MARS

Dentro de cada zona y en un segundo nivel de distribución horizontal, la zona de resultados está dividida en dos partes:

- 1. Visualización de imágenes:** en la parte izquierda a modo de parrilla se encuentran las imágenes (cargadas para realizar la consulta a partir de alguna de ellas, o las obtenidas en la búsqueda). Cada imagen dispone de un *checkbox* y una barra deslizante con valores de 0 a 1 implementar el sistema *relevance feedback* de la aplicación.
- 2. Información sobre la imagen:** a la derecha se sitúa la información extraída de cada imagen resultado. Un primer panel gráfico nos muestra, a través de barras de progresión, el grado de similitud del resultado con la consulta, según criterios de alto nivel (los pesos genéricos de color, textura y estructura). Los siguientes paneles gráficos corresponden a criterios de bajo nivel (características específicas configurables por el usuario de cada peso genérico). Los

elementos gráficos ya informan de manera visual de esta diferencia de nivel de los pesos, ya que los pesos genéricos se muestran en la parte superior y sus barras de progresión son más grandes, mientras que los pesos de bajo nivel se muestran más abajo en la pantalla y sus barras de progresión son más pequeñas. Por último, se incluye un campo de texto con información sobre la imagen (identificador, nombre, puntuación obtenida en la búsqueda).

En la zona de configuración y consulta, se encuentran 4 elementos diferenciados:

- 1. Panel de navegación de la consulta:** para cargar las imágenes de consulta y navegar por las imágenes disponibles (va visualizando imágenes sin ningún orden (paginación) establecida).
- 2. Panel de recuperación:** para lanzar la búsqueda y navegar por los resultados (informa de la página actual, para poder moverse con más facilidad por los resultados).
- 3. Panel de configuración:** MARS permite al usuario elegir qué tipo de pesos quiere utilizar en la búsqueda, seleccionando a través de checkboxes los de alto nivel, los de bajo o ambos. Además estos pesos son configurables a través de ventanas de diálogo, para seleccionar qué pesos aplicar y qué características específicas de bajo nivel utilizar para cada uno (ver figura 36).
- 4. Ámbito de la búsqueda:** a través de un combo, el usuario puede escoger en qué conjunto de imágenes buscar.

▪ **TIPOS DE CONSULTA**

MARS soporta la formulación de consultas complejas a través de operadores booleanos. Los criterios de búsqueda pueden ser especificados por el usuario a través de una imagen ejemplo (cargada por el usuario o seleccionada del conjunto de datos del ámbito de búsqueda) o a partir de paletas de colores o patrones de texturas.

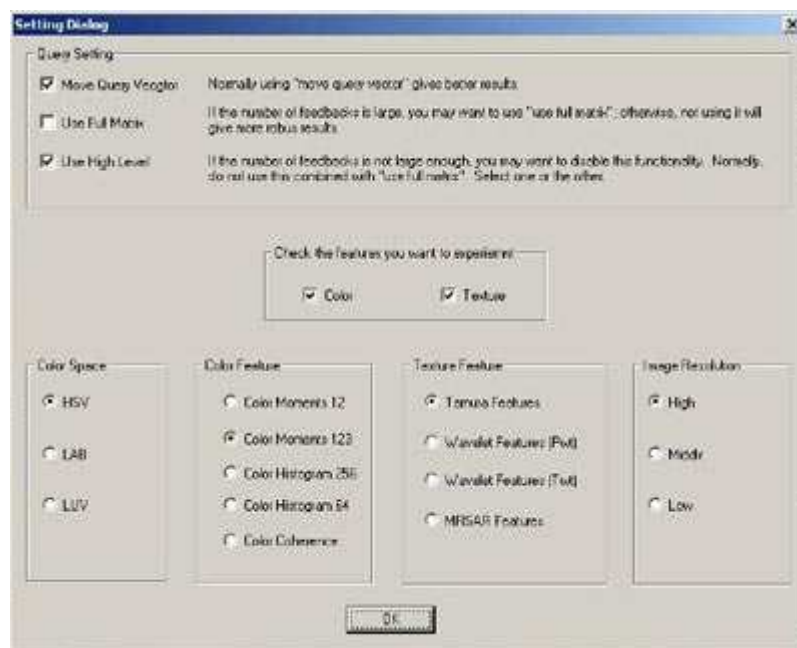


Figura 36. Ejemplo ventana de configuración de MARS

▪ VISUALIZACIÓN DE LOS RESULTADOS

Los resultados son mostrados en la parrilla de imágenes de la pantalla, ordenados según el grado de similitud con la consulta.

▪ RELEVANCE FEEDBACK

Cualquier imagen resultado puede ser utilizada como punto de partida en una nueva consulta para refinar la búsqueda. El sistema incluye un método explícito de retroalimentación para mejorar la búsqueda. El usuario puede otorgar la relevancia de la imagen obtenida según los criterios de búsqueda empleados en la consulta. A través de la barra deslizante que acompaña a cada imagen, puede establecer un valor entre el 0 y el 1, dónde 0 significa ninguna relevancia y 1 significa muy relevante. A continuación, puede marcar aquellas imágenes que se aproximen más al resultado buscado a través del *checkbox* situado al lado de la barra deslizante, para que el sistema refine la búsqueda teniendo en cuenta esta información.

▪ VALORACIÓN DE LA GUI

Aunque hemos visto como MARS es una interfaz que se preocupa presentación de la información en la pantalla, a nivel estético, nos volvemos a encontrar con una GUI poco atractiva, sin ningún tipo de diseño para los elementos gráficos de la aplicación, tanto botones, paneles, tipografía, etc.,

que utilizan los estilos básicos que por defecto traen estos componentes. La interfaz no es muy amigable para el usuario, aunque destaca la voluntad de dar relevancia visual a ciertos elementos en la pantalla, para facilitar la tarea de búsqueda: el nivel de los pesos se corresponde con el tamaño de sus barras de progreso (que indican su relevancia en la búsqueda), y los botones de “*Query*” y “*Settings*” son de mayor tamaño que el resto (rápidamente localizables a golpe de vista).

A nivel funcional, MARS supone un avance importante en dos aspectos: la organización de la información en pantalla y el sistema de *relevance feedback*. La interfaz agrupa de forma lógica y diferenciada los elementos que corresponden a funciones de un mismo ámbito, con paneles específicos para las imágenes de consulta y su navegación, para lanzar la consulta y navegar por los resultados, para la configuración de los criterios y sus pesos, para la visualización de los resultados y la información de las imágenes. Además incorpora un nuevo sistema de comunicación entre el usuario y el motor de búsqueda, para que este último tenga en cuenta su valoración de los resultados para las próximas búsquedas y mejorar su rendimiento.

▪ **NUEVO SISTEMA DE MARS: IMAGEGROUPE**

Los desarrolladores de MARS también son los responsables de *ImageGrouper*, un sistema CBIR basado en consultas de grupos de imágenes, implementado como un Applet de Java2 con Swing API (ver figura 37).

La interfaz de ImageGrouper se integra en cualquier navegador, y adopta el estilo de la mayoría de GUI para web. Dispone un pequeño formulario de búsqueda en la parte superior de la pantalla y deja el resto de la pantalla disponible para la visualización de las imágenes, área que divide en dos zonas: la zona de resultados (vista a modo de parrilla) y una zona de grupos. La novedad de esta aplicación radica en su forma de realizar las búsquedas, con *Query-by-Groups* [13], ya que el usuario puede arrastrar las imágenes de la zona de resultados hasta la zona de grupos, y organizarlas según si cumplen sus expectativas (grupo positivo enmarcado

en azul), si no las cumplen (grupo negativo enmarcado en rojo) o son neutras (grupo neutro enmarcado en blanco).

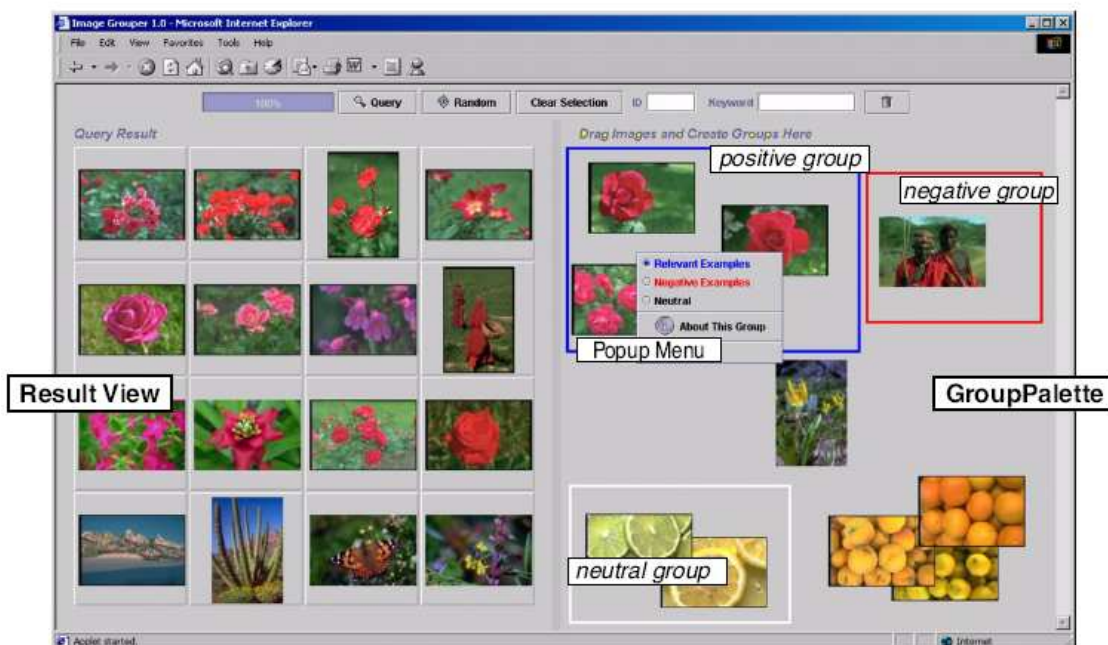


Figura 37. Interfaz del sistema ImageGrouper

El sistema de *relevance feedback* se implementa a través del método *Trial-and-Error-Query* [13]. La aplicación permite que las imágenes pasen de un grupo a otro simplemente arrastrándolas con el ratón. El usuario puede experimentar con diferentes combinaciones de forma rápida y sencilla. Las imágenes que no se utilizan para la consulta quedan fuera de los grupos, pero se mantienen en la pantalla para poder ser reutilizadas más tarde.

- **CIRES: Content Based Image REtrieval System**

Desarrollador: Computer & Vision Research Center, The University of Texas

URL: <http://cires.matthewriley.com/>

Año: 2002-2007

CIRES fue junto con MARS, uno de los sistemas CBIR de referencia por su implementación de métodos *relevance feedback* en sus interfaces. Pero a diferencia de MARS, que evoluciona sus propuestas entorno a la mejora de estos métodos con

su *ImageGrouper*, CIRES ha sorprendido por un cambio de la interfaz hacia un diseño más elaborado estéticamente pero más simple en funcionalidades. La nueva GUI de CIRES (fig.40), actualizada en el 2007, ha eliminado la funcionalidad de *relevance feedback* en su aplicación y ha reducido las opciones de configuración de la consulta por parte del usuario.

▪ **PRESENTACIÓN DE LA INFORMACIÓN**

CIRES es una interfaz *web-like*, que reproduce nuevamente el sencillo formato de formulario para web. Como diferencia a otras interfaces web, CIRES separa la parte de definición de la consulta y la visualización de los resultados en páginas diferentes, no ubica las dos áreas en la misma página como casi todos los buscadores de imágenes web.

En su primera interfaz, la definición de la consulta se planteaba a partir de dos páginas: el usuario debía seleccionar primero la categoría de imágenes en la que deseaba buscar a través de una tabla (fig. 38) y a continuación definir la consulta (fig. 39).



Figura 38. Página de selección de categoría del sistema CIRES



Figura 39. Página de formulación de la consulta del sistema CIRES

Podemos observar en la figura 38 que la página de selección de categoría a través de una tabla no era un sistema demasiado práctico ni optimizaba el espacio, ya que el usuario debía utilizar el scroll vertical del navegador para visualizar las opciones, incluso las más generales. En la primera columna encontrábamos la categoría más genérica, y en las sucesivas columnas, las correspondientes subcategorías asociadas. Esta distribución de las jerarquías de forma horizontal limitaba la ampliación de las categorías disponibles enormemente por la gran cantidad de espacio que se necesitaba.

Una vez escogida la categoría, el usuario debía completar el formulario de la consulta, eligiendo la imagen ejemplo y ponderando los tres pesos disponibles en la aplicación (color, textura y agrupación perceptual). También permitía configurar la textura, seleccionando los tres canales (L=luminancia, A y B = crominancia) o simplemente el canal de la luminancia.

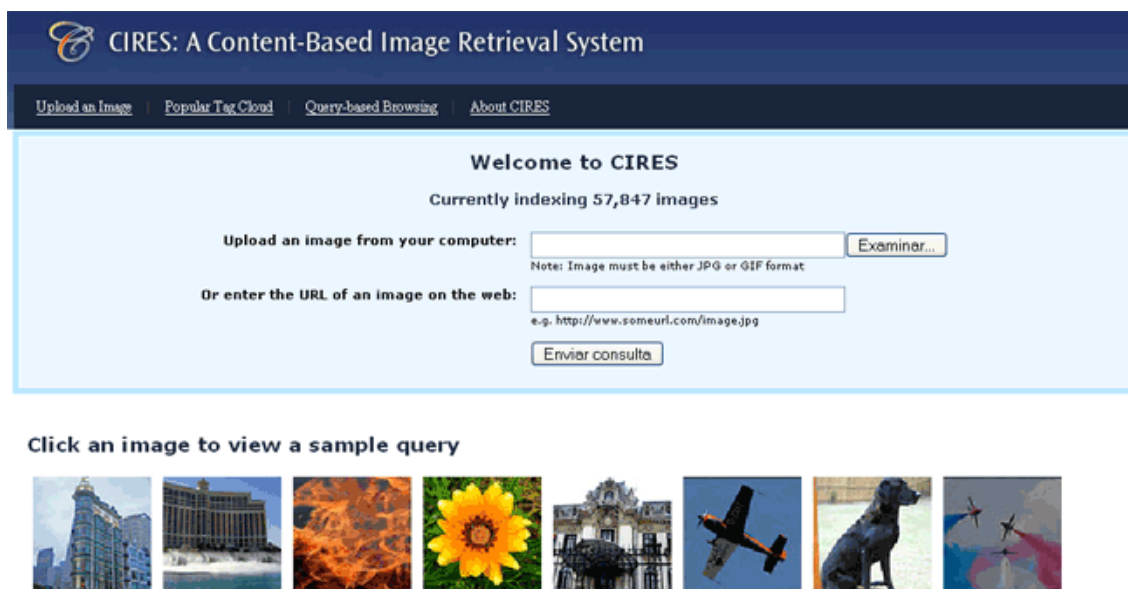


Figura 40. Nueva interfaz del sistema CIRES

La nueva interfaz de CIRES simplifica la definición de la consulta, ya que el usuario puede iniciar la búsqueda a partir de cualquier imagen, propia, de la web o una de las que proporciona el sistema (figura 40), o si lo prefiere, puede seleccionar un *tag* (una etiqueta) que le proporciona el sistema para conseguir un primer conjunto de imágenes resultado (fig.41).

Como podemos apreciar en la figura 41, la presentación de los tags no mejora mucho en comparación con la anterior tabla de categorías, ya que también requiere de bastante espacio (volvemos a necesitar el scroll vertical) y el tamaño de la letra en función de su popularidad acaba dificultando su lectura.

La nueva interfaz ha eliminado en el formulario de entrada de datos de la primera consulta la posibilidad de que el usuario configure los pesos para realizar la búsqueda.

▪ TIPOS DE CONSULTA

CIRES combina los métodos de consulta basada en imagen ejemplo y texto, aunque no permite que la búsqueda se realice a partir de una entrada de texto libre por parte del usuario.



Figura 41. Página de los tags más populares del sistema CIRES

- **VISUALIZACIÓN DE LOS RESULTADOS**

Actualmente, CIRES presenta la siguiente página (fig. 42) con los primeros resultados de la consulta. En la parte superior de la pantalla encontramos un área destacada (fondo azul) con la imagen de consulta, tres pesos a ponderar para redefinir nuestra consulta (color, textura y estructura), y una serie de tags relacionados con la imagen consultada para ejecutar una nueva búsqueda. A continuación se muestran los resultados en forma de parrilla de miniaturas.

La aplicación no ofrece ningún tipo de información sobre las imágenes resultado y ya no permite tampoco la retroalimentación del sistema con la opinión del usuario.

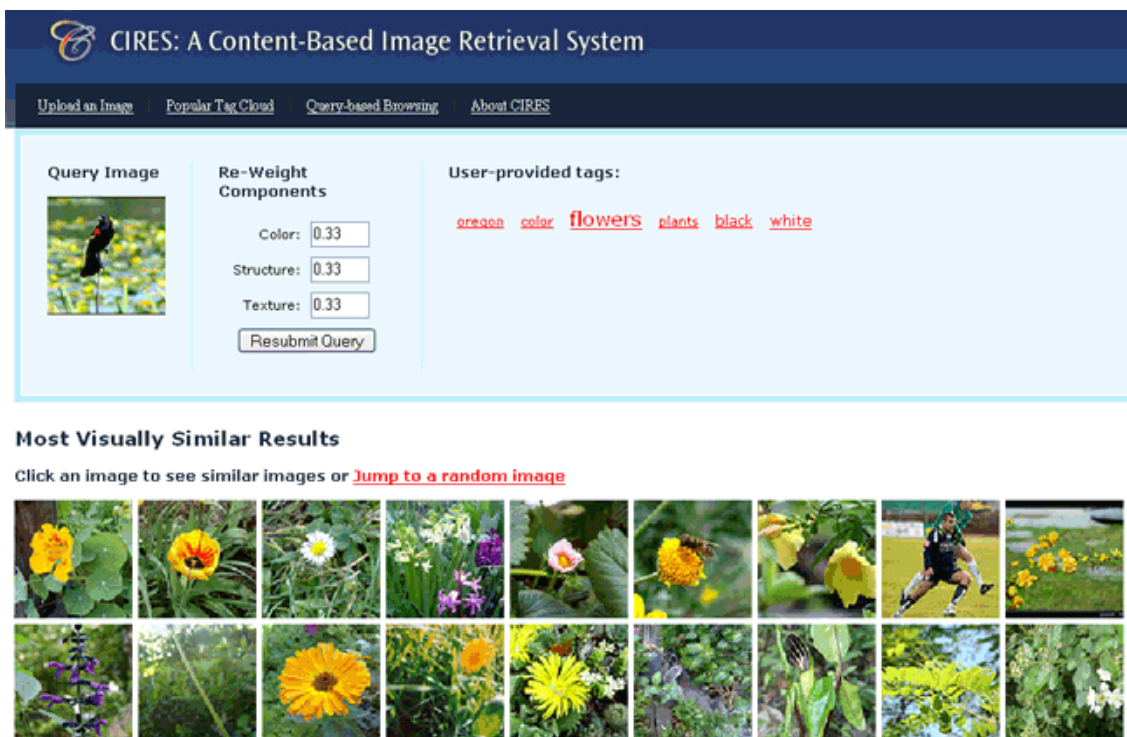



Figura 42. Página de resultados actual del sistema CIREs

- **RELEVANCE FEEDBACK**











La actual interfaz solo permite que cualquier imagen resultado pueda ser utilizada como punto de partida en una nueva consulta para refinar la búsqueda, clicando sobre ella. La anterior interfaz (fig. 43) permitía al usuario mejorar los resultados obtenidos a través de los tres *checkboxes* que aparecían bajo las imágenes. Si el usuario seleccionaba “Yes”, comunicaba al sistema que consideraba la imagen similar al contenido de la imagen ejemplo. Seleccionaba “No” si no era considerada similar y “NS” (*not sure*) si el usuario no estaba seguro. Por defecto estos *checkboxes* eran inicializados con “No”. Además CIREs ofrecía dos mecanismos diferentes de *relevance feedback*, entre los que el usuario podía escoger.

Query Image



Relevance feedback type: Cluster. Weight: Perceptual Grouping = 0.33, Color = 0.33, Texture = 0.33, L, A, B channels.
For relevance feedback, please select the check boxes below each image, and then select the feedback type. Note that NS = "Not Sure".

Retrieved Images

				
<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No
				
<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No	<input type="radio"/> Yes <input type="radio"/> NS <input checked="" type="radio"/> No

Relevance feedback type:

- Cluster: Only "Yes" images used as a multiple image query set.
- Class: "Yes", "Not Sure" and "No" images used as training samples in a multi-class classification.

Note that, for multi-class feedback, running a feedback query with the default value of "No" for all images would return results that **do not** agree with class(es) represented by the current result set.

Weights (should sum to 1):

Perceptual grouping: Color: Texture:

Texture:

- L, A and B channels
- L channel only (~Grayscale texture)

Figura 43. Antigua página de resultados con relevance feedback del sistema CIRES

o **Valoración de la GUI**

Comparando ambas interfaces, vemos que no ha habido una evolución paralela a nivel estético y funcional. La nueva interfaz ha ganado en diseño, y usabilidad, el usuario realiza las búsquedas de forma más rápida y simple, pero esta simplicidad ha ido en detrimento de las funcionalidades, ya que se ha perdido flexibilidad en la consulta, ahora mucho menos configurable, y el valor añadido al sistema que proporcionaban sus métodos de *relevance feedback*.

- **CIRCUS (Content-based Image Retrieval and Consultation User System)**

Desarrollador: Colaboración entre Laboratoire de communications audiovisuelles (LCAV), Ecole Polytechnique Fédérale de Lausanne (EPFL), y el Computer Vision Group University of Geneva.

URL: http://viper.unige.ch/doku.php/demos#content-based_image_retrieval

Año: 2002

CIRCUS es un proyecto dedicado al desarrollo de un sistema para la recuperación de imágenes de colecciones distribuidas, heterogéneas y anotadas [6]. Destaca de CIRCUS su elaborada arquitectura cliente/servidor a través del protocolo abierto de comunicación MRML¹⁸, disponible bajo licencia pública GNU. Esta arquitectura permite que varias interfaces de usuario se conecten al conjunto de servidores que implementan diferentes métodos de recuperación de contenido y operan sobre diferentes colecciones.

El trabajo realizado en CIRCUS tiene su continuidad en VIPER, un proyecto para la recuperación de información multimedia gestionado por el *Computer Vision and Multimedia Lab (CVML)* de la universidad de Ginebra¹⁹. El CVML ha desarrollado el GIFT²⁰, un paquete de código abierto para implementar sistemas CBIR utilizando el método QBE (*Query-by-Example*).

Para diseñar la GUI del sistema, se realizó un análisis de las tareas que necesitaba implementar un sistema de recuperación de información multimedia, y se llegó a la conclusión que este tipo de herramientas debían potenciar la especificación de la consulta y la visualización interactiva de los resultados [6].

¹⁸ Multimedia Retrieval Markup Language (MRML). This XML-based markup language is the basis for an open communication protocol for content-based image retrieval systems (CBIRs).[6]

¹⁹ VIPER: Multimedia Information Retrieval (<http://viper.unige.ch/doku.php/research>)

²⁰ Véase la figura 4, interfaz web en php para VIPER-GIFT.

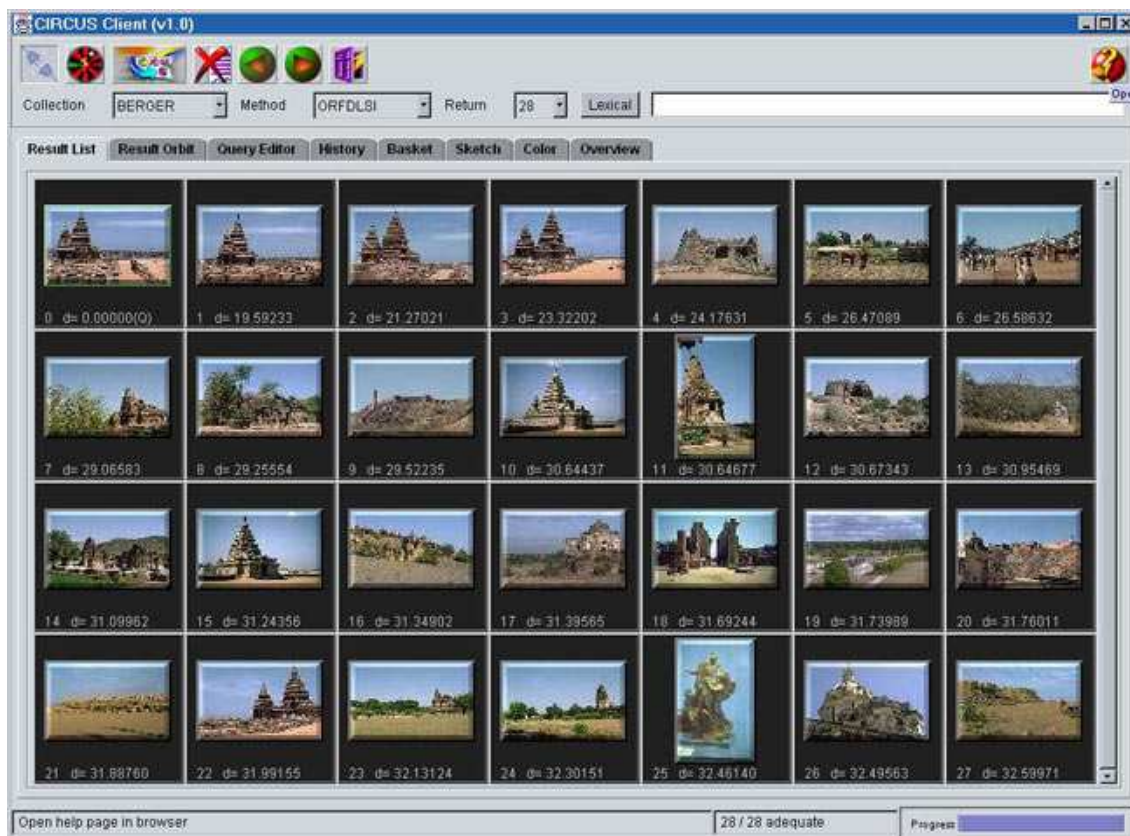


Figura 44. Interfaz de CIRCUS

▪ PRESENTACIÓN DE LA INFORMACIÓN

La interfaz de CIRCUS es del tipo *window-like*, implementada con Java, y aprovecha muy bien el sistema de ventanas para organizar las diferentes funcionalidades de la aplicación. En la parte superior de la pantalla sitúa una barra general de herramientas que son accesibles en todo momento por el usuario. Bajo esta barra de herramientas encontramos una serie de elementos genéricos configurables por el usuario: tres combos desplegables para seleccionar la colección de imágenes dónde buscar, el algoritmo de búsqueda y el número de resultados a mostrar, y un campo de texto para realizar búsquedas a través de texto libre.

El resto de pantalla se utiliza para visualizar las diferentes ventanas según la operación que el usuario esté realizando. Cada ventana tiene una pestaña superior con el título de su funcionalidad. Las dos primeras son ventanas de visualización de resultados, una en forma de lista y otra en forma de mapa ("*Result Orbit*"). La ventana "*Query Editor*" permite configurar la consulta y otras como "*Color*" o "*Sketch*" permiten especificar

algunas características de búsqueda a través de la selección de colores a partir de paletas del sistema o de formas de una imagen ejemplo.

▪ **TIPOS DE CONSULTA**

El sistema permite formular la consulta a partir de imágenes y/o texto. El usuario puede generar la búsqueda combinando los diferentes métodos disponibles y estableciendo la relevancia de las características visuales empleadas, por ejemplo especificando el porcentaje de cada color deseado en los resultados.

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

La visualización de los resultados es un punto fuerte en esta interfaz, que ofrece dos posibilidades: la vista en forma de lista, con las imágenes ordenadas según el grado de similitud con la consulta (se muestra la puntuación obtenida bajo cada resultado) o la vista en forma de gráfico (órbita). La ventana con los resultados en órbita muestra la imagen consultada en la esquina inferior izquierda del área gráfica y los resultados se posicionan según la similitud con la consulta, cuanto más cerca de la imagen consulta, más similitud. Es posible navegar por los resultados a través de los botones de la barra superior de herramientas.

CIRCUS permite realizar varias consultas y combinar los resultados en diferentes ejes del espacio: la figura 45 muestra el resultado de dos consultas, la primera utiliza el eje horizontal y la segunda el eje vertical.

▪ **RELEVANCE FEEDBACK**

Los usuarios pueden especificar en cada búsqueda los resultados positivos (al seleccionarlos quedan enmarcados con un borde verde) y los negativos (quedan enmarcados con un borde rojo), y lanzar nuevas consultas que realizan la intersección entre las características de los ejemplos positivos menos las características de las imágenes consideradas negativas.

▪ **VALORACIÓN DE LA GUI**

La interfaz de CIRCUS es un buen ejemplo de organización y distribución de los elementos relevantes en un sistema CBIR. Como hemos podido comprobar, la separación de las funcionalidades por ventanas ayuda al

usuario a utilizar sin problemas la aplicación, y las acciones más genéricas se sitúan en la parte superior de la pantalla para facilitar su accesibilidad. Así, la usabilidad ha primado una vez más sobre la estética, que no es muy acertada en el estilo de los iconos, excesivamente grandes y visualmente confusos (el usuario puede tardar en reconocer su función).

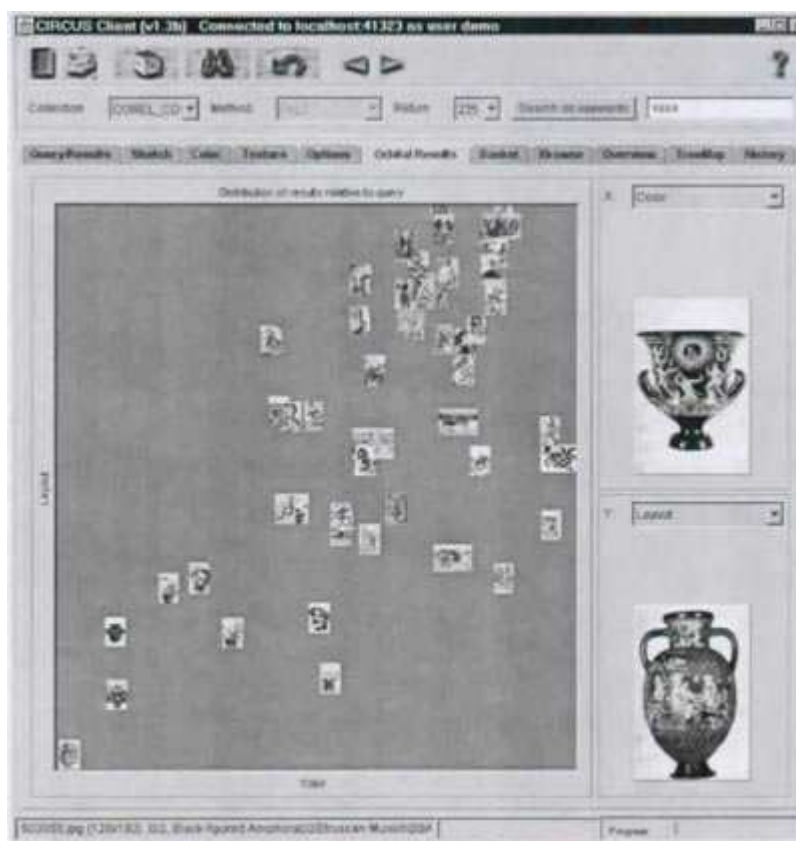


Figura 45. Resultados de dos consultas combinadas en CIRCUS

- **Caliph & Emir**

Desarrollador: SourceForge.net (SourceForge Inc.)

URL: <http://sourceforge.net/projects/caliph-emir/> , <http://caliph-emir.sf.net>

Año: 2004-2007

SourceForge.net es un sitio web para la gestión de proyectos dedicados al desarrollo de software de código abierto. Bajo su auspicio, se ha desarrollado **Caliph & Emir**, un kit de herramientas en Java para la anotación y recuperación de imágenes y fotografías digitales basándose en los descriptores visuales definidos en el estándar

MPEG-7. Aunque SourceForge.net es propiedad de la empresa SourceForge Inc, los proyectos que se gestionan en este sitio web no pueden ser considerados comerciales, ya que contribuyen a la difusión de nueva tecnología.

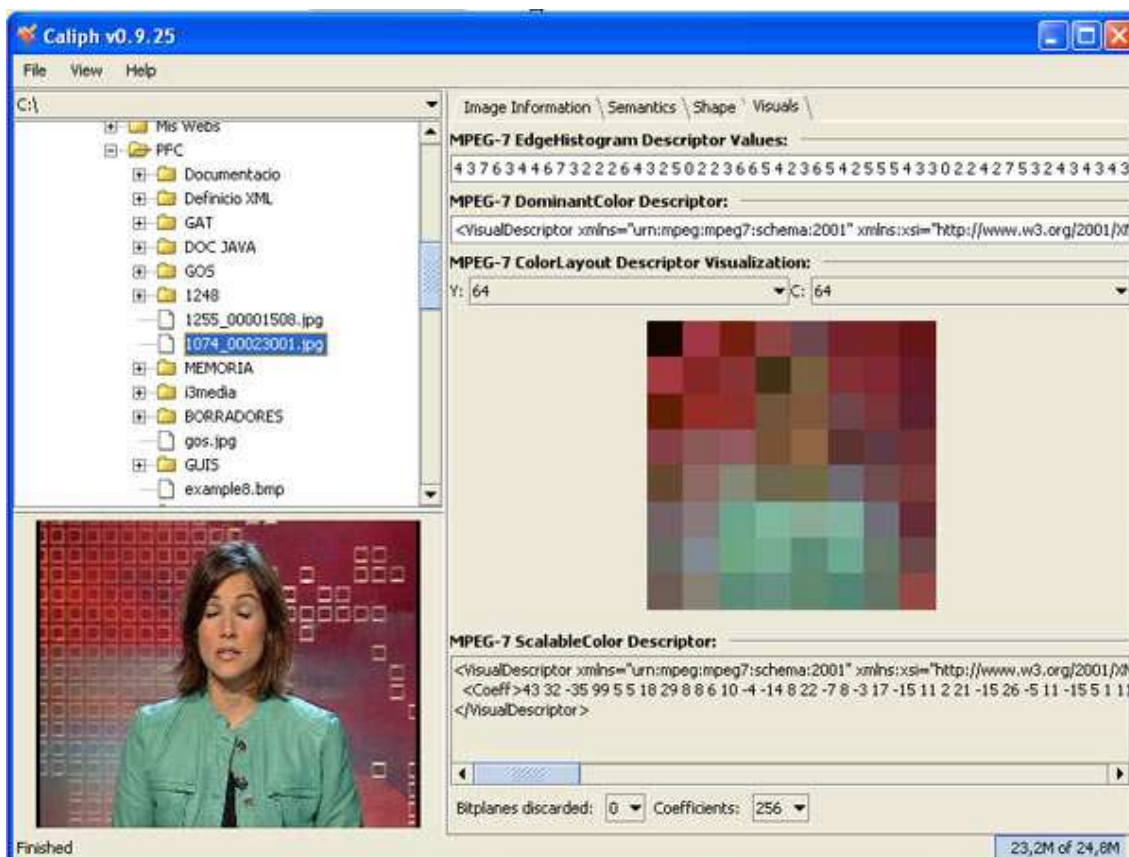


Figura 46. Interfaz de Caliph

Dentro de SourceForge.net, también se ha desarrollado la librería **LIRE** (Lucene²¹ Image REtrieval) para implementar aplicaciones CBIR. LIRE proporciona un sistema para crear índices de imágenes, realizar búsquedas, navegar por estos índices y crear mosaicos de imágenes para su visualización, que es utilizado por Caliph & Emir.

Muchos de los proyectos CBIR de ámbito académico se preocupan por todos los procesos necesarios en la recuperación de imágenes, desde la anotación e indexación del contenido hasta su búsqueda y posterior visualización. Caliph & Emir se compone de dos aplicaciones complementarias, por un lado tenemos **Caliph**

²¹ Apache Lucene es un proyecto para desarrollar software de búsqueda (<http://lucene.apache.org/>)

(Common And Light weight PHoto) como herramienta de anotación de imágenes, y por otro **Emir** (Experimental Metadata based Image Retrieval) como herramienta de búsqueda y visualización. Ambas aplicaciones comparten el mismo estilo de GUI, así el usuario puede familiarizarse rápidamente con los entornos de trabajo. Nosotros analizaremos la interfaz de Emir dedicada a la búsqueda de imágenes.

▪ PRESENTACIÓN DE LA INFORMACIÓN

La interfaz de Emir es del tipo *window-like*. La aplicación dispone de un menú superior con las tareas más importantes de la herramienta, que siempre está accesible al usuario, y utiliza una serie de ventanas para presentar la información. Cada ventana tiene una pestaña con el título de su función, y de esta manera se separan las funciones de búsqueda y visualización de resultados perfectamente. Emir permite realizar búsquedas de imágenes previamente anotadas por Caliph, por lo que necesita que el usuario configure el directorio donde se encuentren las imágenes anotadas (proceso automático de indexación) antes de lanzar cualquier consulta. La primera vez que el usuario realice este paso previo requerido por la aplicación necesitara consultar la documentación porque no es nada intuitivo (existe un *wizard*²² para realizar esta configuración).

Al iniciar la aplicación nos aparecen tres pestañas, correspondientes a los tres tipos de búsqueda que soporta: “Index”, “Graph” y “Image”. Estas tres ventanas se componen de los elementos necesarios para formular la consulta. La ventana de “Index” contiene un campo de texto en la parte superior para realizar la consulta a partir de palabras clave introducidas por el usuario, y el resto de pantalla no tiene ninguna funcionalidad (se disimula incluyendo iconos de ayuda) (ver figura 47). La ventana “Graph” permite crear esquemas gráficos con nodos semánticos y relacionarlos para lanzar una búsqueda. En la parte superior de la ventana se crean los nodos y se utiliza el resto de pantalla para dibujar el gráfico (ver figura 47).

²² A **wizard** is a user interface element where the user is presented with a sequence of dialog boxes. Through these dialog boxes, the user is led through a series of steps, performing tasks in a specific sequence. Sometimes it may be easier to perform tasks using a wizard, especially for complex or infrequently performed tasks where the user is unfamiliar with the steps involved. ([http://en.wikipedia.org/wiki/Wizard_\(software\)](http://en.wikipedia.org/wiki/Wizard_(software)))

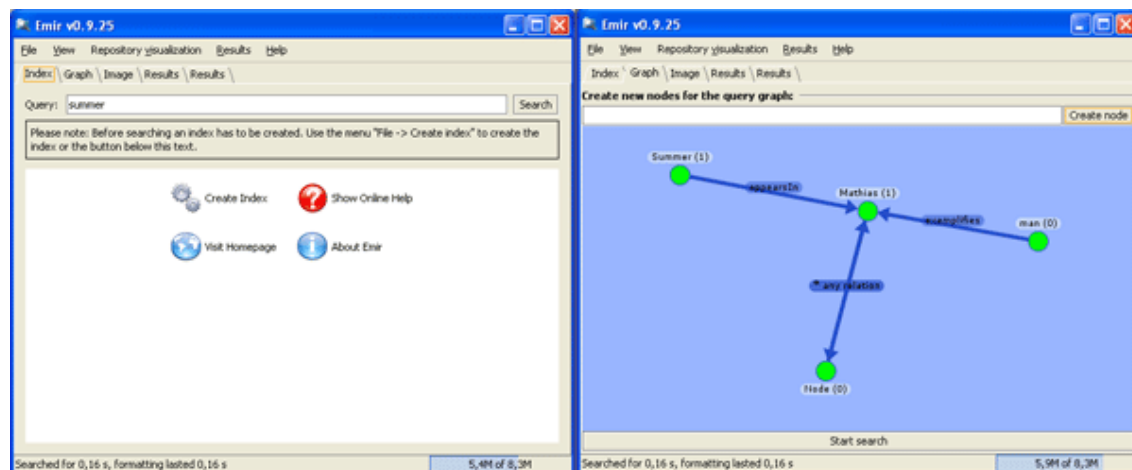


Figura 47. Consultas por texto (izquierda) y por gráfico (derecha) de EMIR

La ventana “Image” permite formular la consulta a través de una imagen (ver figura 48). La distribución de la pantalla organiza la información de manera que la imagen consulta y su mapa de color aparezcan a la izquierda y el pequeño formulario de consulta a la derecha. El usuario puede escoger qué características visuales utilizar en la búsqueda, combinándolas como mejor le interese. La presentación de las opciones no es acertada, ya que al utilizar un grupo de *checkboxes* de selección única, se necesita crear una opción para cada combinación posible. De momento se utilizan tres características, que suponen siete opciones, pero la inclusión de sólo una nueva característica de búsqueda duplicaría el número de opciones, creando un problema de espacio (vertical y horizontal).

▪ TIPOS DE CONSULTA

El sistema permite formular la consulta de tres formas diferentes, texto, gráfico semántico o imagen, pero cada método se ejecuta por separado, no soporta la combinación entre ellos. En la consulta a partir de una imagen el usuario escoge el método de búsqueda de entre tres descriptores visuales tomados del estándar MPEG-7: *ScalableColor*, *ColorLayout* y *EdgeHistogram*, aunque no permite ponderarlos. Recientemente LIRE ha implementado un cuarto descriptor, *AutoColorCorrelogram*, que podría ser incorporado en futuras versiones de Caliph & Emir.

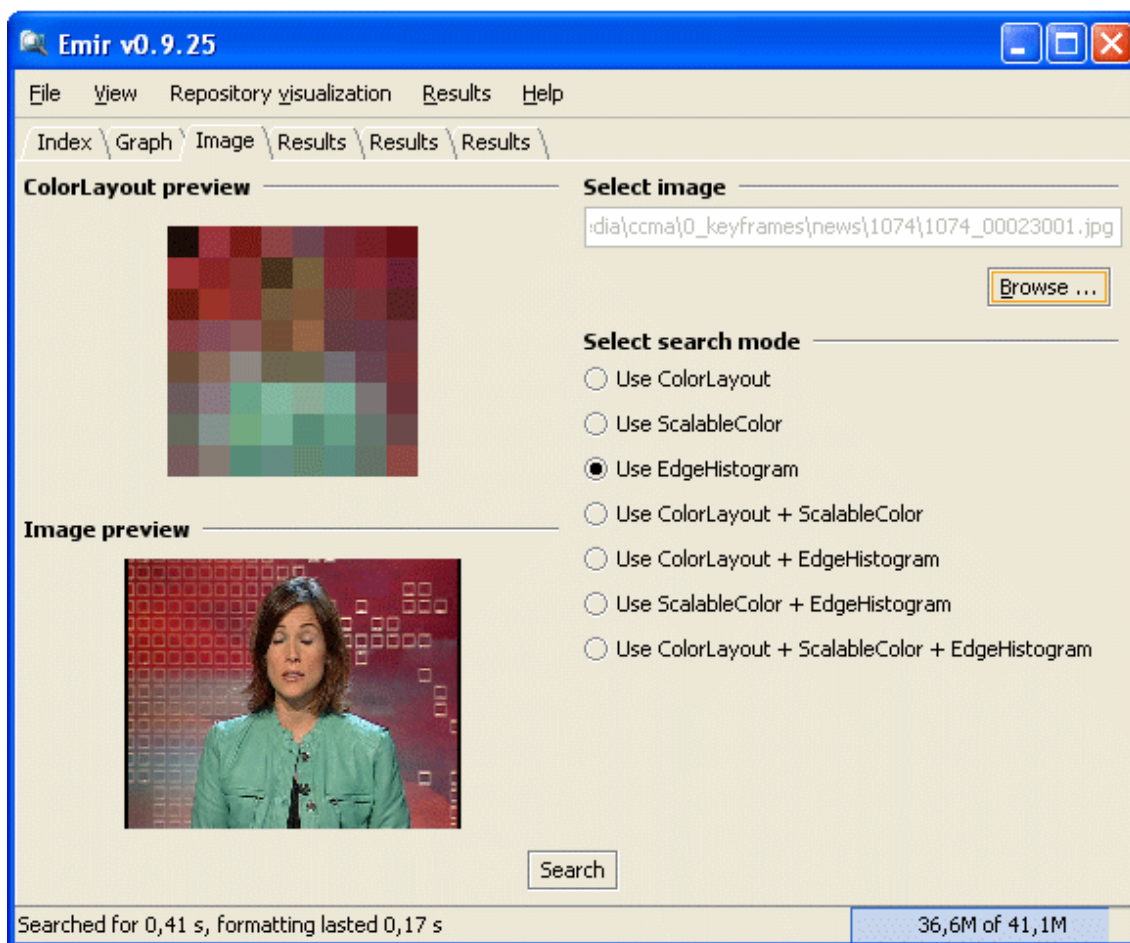


Figura 48. Ventana de consulta de EMIR

▪ VISUALIZACIÓN DE LOS RESULTADOS

Una vez ejecutada la búsqueda, los resultados aparecen en una nueva ventana “Results” ordenados según su relevancia en una lista (ver figura 49). Cada imagen aparece acompañada a su derecha de información relacionada: su puntuación, nombre y tamaño del archivo, creador de la imagen, fecha y una breve descripción. El usuario puede realizar varias búsquedas, e ir las acumulando en la aplicación, ya que cada consulta crea su propia ventana de resultados, no se pierden los anteriores. Clicando sobre un resultado se abre una ventana pop-up para visualizar la imagen a tamaño original.

Como aspecto negativo de la aplicación, encontramos que no permite definir el número de resultados que se desean obtener, y si éstos son muy numerosos, la única forma de recorrerlos es con el scroll vertical de la ventana, no incluye ningún sistema de navegación alternativo.

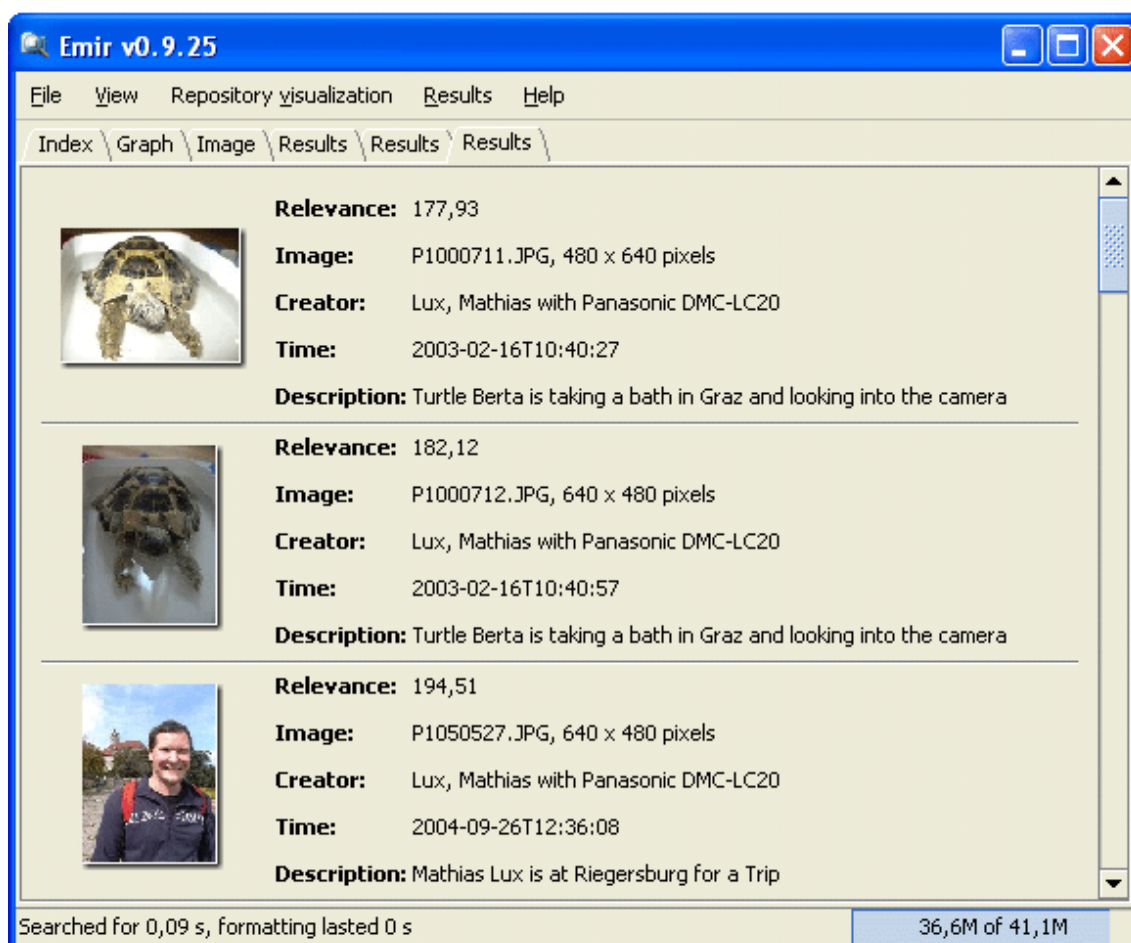


Figura 49. Ventana de resultados de EMIR

Los resultados también pueden ser visualizados en un gráfico 2D (ver figura 50), pero tampoco es una función muy útil debido al tamaño en que se presentan las imágenes y su baja resolución incluso utilizando el zoom.

- **RELEVANCE FEEDBACK**

Emir no incluye ningún sistema de feedback, ni siquiera permite lanzar nuevas consultas a partir de una imagen resultado.

- **VALORACIÓN DE LA GUI**

En general, Emir es una buena interfaz de búsqueda de imágenes, ya que a nivel funcional organiza la información correctamente en ventanas separadas para consultas y visualización de resultados. Esta distribución es práctica, pero en algunos casos se desaprovecha el espacio en pantalla. Las consultas de "Index" e "Image" se podrían combinar y utilizar una única ventana para formular la consulta. Las ventanas de visualización de

resultados también pueden llegar a ser más usables, incluyendo más herramientas de navegación que faciliten la presentación de las imágenes (en la forma de lista, la información ocupa mucho espacio, y en la forma de gráfico 2D, las imágenes no se ven bien).

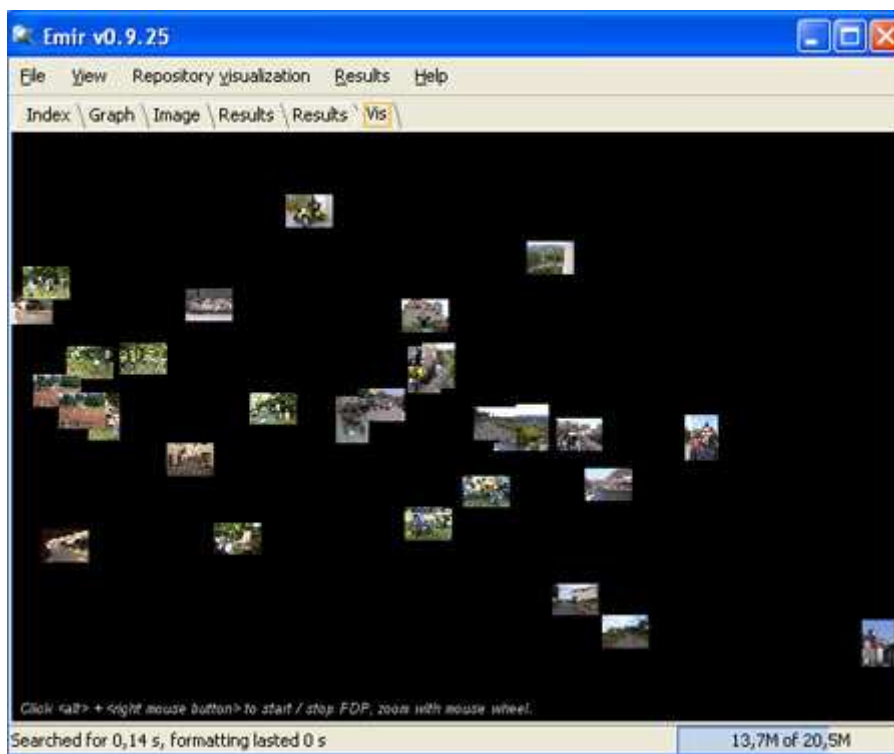


Figura 50. Visualización de resultados en 2D de EMIR

A nivel estético, Emir utiliza el estilo por defecto de los elementos de aplicaciones de ventanas, no crea un diseño propio, que haga la interfaz más atractiva al usuario.

- **Buscador de videos mediante descriptores MPEG-7**

Desarrollador: Grupo de Procesado de Imagen y Vídeo del Departamento de Teoría de la Señal y Comunicaciones (TSC) de la Universidad Politécnica de Cataluña (UPC).

Año: 2006

Aunque no es un buscador de imágenes CBIR, incluimos en nuestro análisis este buscador web de secuencias de vídeo mediante descriptores MPEG-7 por tratarse del precedente inmediato de la interfaz del GOS. Esta aplicación web es la

primera GUI creada por el GPI de la UPC [14] que utiliza un sistema de búsqueda de imágenes a partir de descriptores MPEG-7. Estos descriptores, desarrollados también por el GPI [15], son los mismos utilizados por el GOS.

NOMBRE	FRAME	FORMATO	COMENTARIO	LONGITUD	TAMAÑO_X	TAMAÑO_Y	Fps	UBICACIÓN
Akiyo		YUV	Akiyo	176	144	300	30	/opt/lampp/html
Coastguard		YUV	Coastguard	176	144	300	30	/opt/lampp/html
Concrete		YUV	Concrete	352	288	574	30	/opt/lampp/html
Destruction		YUV	Destruction	176	128	300	30	/opt/lampp/html

Figura 51. Página principal del Buscador de vídeos mediante MPEG-7

▪ PRESENTACIÓN DE LA INFORMACIÓN

Esta herramienta de búsqueda para vídeos se ideó como un portal web, compuesto por diversas páginas que permiten organizar las diferentes funcionalidades requeridas en el sistema de búsqueda. La página principal del portal (fig. 51) se estructura a base de marcos (*frames*)²³, formato que posibilita la creación de contenido estático en zonas específicas de la pantalla. Así, en la parte superior encontramos dos marcos, uno con el logo de la UPC y otro con el nombre de la aplicación, y a la izquierda, dentro de otro marco, se sitúa el menú principal, con enlaces a las páginas disponibles en el portal. El contenido de estos tres marcos es siempre el mismo, no varía. Un último marco central, que ocupa el resto de pantalla, alberga las diferentes páginas a las que el usuario tiene acceso a través de menú izquierdo. En este marco central se va mostrando diferente información en función de la página seleccionada. Esta distribución de contenidos en el portal es muy práctica y cómoda para el usuario, que

²³ El lenguaje HTML dispone de etiquetas (o marcas) <FRAME> y <FRAMESET> para crear páginas con marcos, divisiones dentro de una misma página.

puede acceder a cualquier contenido en todo momento a través del menú izquierdo, siempre visible en pantalla.

A través de la página principal del portal, el usuario tiene acceso a cuatro secciones:

1. *Buscador*, que analizaremos a continuación.
2. *Vídeos*: página que gestiona la inserción y la eliminación de vídeos por parte del usuario.
3. *Descriptores*: página que gestiona la inserción y la eliminación de los descriptores disponibles para realizar las búsquedas.
4. *Formatos*: página que gestiona la inserción y la eliminación de los formatos disponibles para cada vídeo.

El buscador consta de dos páginas, una para formular la consulta y otra para visualizar los resultados. Veamos como se han diseñado.

▪ **FORMULARIO DE CONSULTA**

La consulta se realiza a través de un formulario web muy sencillo. En la página inicial del buscador (ver figura 52) se confecciona la consulta seleccionando un vídeo y un descriptor a través de sendos combos desplegables con las opciones disponibles. La zona destinada al formulario es la parte inferior de la pantalla, mientras que en la parte superior, ocupando casi toda la pantalla, se presentan dos tablas con los datos disponibles, introducidos previamente por el usuario. Esta organización no es acertada, ya que la ubicación de los elementos denota su nivel de importancia dentro de la interfaz. En un buscador, la formulación de la consulta es la parte más importante de la aplicación, porque es el primer paso para iniciar la búsqueda. El área de consulta debe estar en un lugar destacado, generalmente en la parte superior de la pantalla, y en esta interfaz, la consulta queda relegada a un segundo plano. Esta mala distribución se convertirá en un problema cuando las tablas de vídeos y descriptores aumenten de tamaño, ya que irán desplazando el formulario de consulta hacia abajo, quedando oculto para el usuario. A nivel de usabilidad, no es aceptable que el área de formulación de consulta no esté visible, obligando al usuario a buscar dónde puede lanzar una búsqueda.

TABLA DE VIDEOS

NOMBRE	FRAME	FORMATO	COMENTARIO	LONGITUD	TAMAÑO_X	TAMAÑO_Y	Fps	UBICACIÓN
Akiyo		YUV	Akiyo	176	144	300	30	/opt/lampp/htdocs/pr
Coastguard		YUV	Coastguard	176	144	300	30	/opt/lampp/htdocs/pr
Coastguard_cif		YUV	Coastguard_cif	352	288	300	30	/opt/lampp/htdocs/pr
Concrete		YUV	Concrete	352	288	574	30	/opt/lampp/htdocs/pr

TABLA DE DESCRIPTORES

NOMBRE	BINARIO DE EXTRACCIÓN	BINARIO DE COMPARACIÓN
ColorLayout	/opt/lampp/htdocs/proyecto/binarios/B_COLORLAYOUT_EXTRACT	/opt/lampp/htdocs/proyecto/binarios/B_CO
Motion Activzty	/opt/lampp/htdocs/proyecto/binarios/B_MOTIONACTIVITY_EXTRACT	/opt/lampp/htdocs/proyecto/binarios/B_MO

Selecciona un video:

Selecciona un descriptor:

Figura 52. Página inicial del Buscador de vídeos mediante MPEG-7

El uso del espacio en pantalla tampoco es eficiente. Los vídeos y descriptores mostrados en las tablas son los mismos que aparecen en los combos desplegados del formulario, presentado dos veces la misma información en una misma página. La información extra que contienen las tablas es consultable en las secciones del portal destinadas a vídeos y descriptores, donde se muestran exactamente las mismas tablas. Es por tanto innecesaria esta duplicación de la información y el consumo de espacio que supone.

▪ **VISUALIZACIÓN DE LOS RESULTADOS**

Los resultados se muestran en una nueva página, listados en una tabla por orden descendiente según la puntuación obtenida en la búsqueda. Cada fila de la tabla corresponde a un resultado, presentando el nombre del vídeo, un *frame* y la puntuación obtenida durante el proceso de búsqueda. No dispone de ningún sistema de navegación de resultados.

**LA LISTA DE VIDEOS ES LA SIGUIENTE
HABIENDO SELECCIONADO COMO VIDEO: Akiyo
Y COMO DESCRIPTOR: ColorLayout**

Video	FRAME	Valor de Comparación
Akiyo		1.000000
Coastguard		0.968235
Destruction		0.967935
Concrete		0.956557

Figura 53. Página de resultados del Buscador de vídeos mediante MPEG-7

▪ **VALORACIÓN DE LA GUI**

A nivel de estructura, la elección de un portal web para implementar las funcionalidades del sistema de búsqueda es acertada, el usuario accede a todas las páginas disponibles en cualquier momento y con un solo clic. En cambio a nivel estético y funcional es una interfaz muy precaria, el diseño no ha tenido en cuenta las necesidades del usuario, especialmente en la parte del buscador. La organización de los elementos en pantalla, así como sus dimensiones (los títulos son excesivamente grandes, y de hasta tres líneas en la página de resultados del buscador), no son adecuados para obtener una interfaz práctica, usable y amigable.

3. Requisitos

El grupo GPI de la UPC ha desarrollado una plataforma de herramientas para el procesamiento de imágenes llamada **SoftImage**. Esta plataforma cuenta con el programa **B_RANKER**, un motor de búsqueda de imágenes que implementa el método de consulta por ejemplos. El B_RANKER es un programa desarrollado en lenguaje de programación ANSI C, el cual podemos ejecutar a partir de un archivo binario (un ejecutable). La versión actual del ejecutable sólo está disponible para sistemas operativos GNU/Linux.

El objetivo de este proyecto es diseñar e implementar una interfaz gráfica de usuario (GUI-*Graphical User Interface*) para el programa B_RANKER. Para ello, debemos tener en cuenta los requisitos que B_RANKER, como motor de búsqueda, nos impone. Recordemos que una interfaz para un buscador CBIR debe prestar atención tanto a las necesidades del usuario como del sistema de búsqueda. Desde la perspectiva del sistema de búsqueda, la interfaz traducirá las instrucciones del usuario al sistema, que recibirá unos determinados parámetros de entrada imprescindibles para iniciar una búsqueda. Una vez ejecutada la consulta, la propia interfaz traducirá al usuario los resultados obtenidos por el sistema, devueltos como parámetros de salida.

La clave de esta comunicación es el “idioma” utilizado. El lenguaje que emplean usuario e interfaz no es el mismo que el utilizado por la interfaz y el sistema. El usuario se comunica con la interfaz interactuando con los elementos gráficos de la aplicación, a través del teclado y el ratón. El motor de búsqueda B_RANKER utiliza parámetros y ficheros MPEG-7/XML para recibir y enviar los datos de entrada y salida.

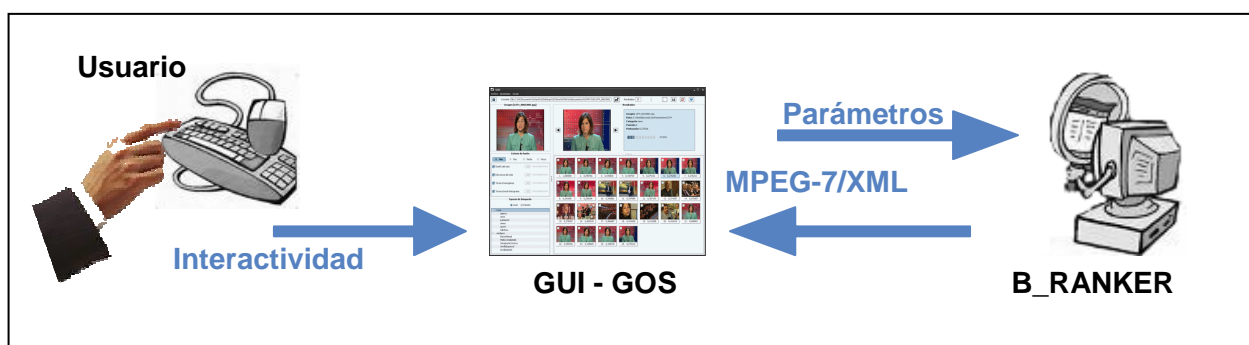


Figura 54. Esquema diálogo

3.1 Llamada a B_RANKER

El programa B_RANKER necesita recibir tres parámetros para poder ejecutar una búsqueda. La primera tarea de la interfaz es confeccionar la llamada a B_RANKER con estos tres parámetros y toda la información que necesita el sistema para funcionar correctamente. El formato de la llamada es el siguiente:

```
B_RANKER -query [localización en disco de la imagen de consulta]  
-conf [localización en disco del fichero de configuración]  
-results [directorio de disco donde se dejan los resultados]
```

- **PARÁMETRO *-query***

El sistema de búsqueda debe saber qué busca, y con el parámetro *-query* recibe la ruta de la imagen de consulta. B_RANKER admite imágenes en formato .jpg, .gif, .png, .bmp, .jpeg (se implementará un filtro para controlar el formato introducido por el usuario).

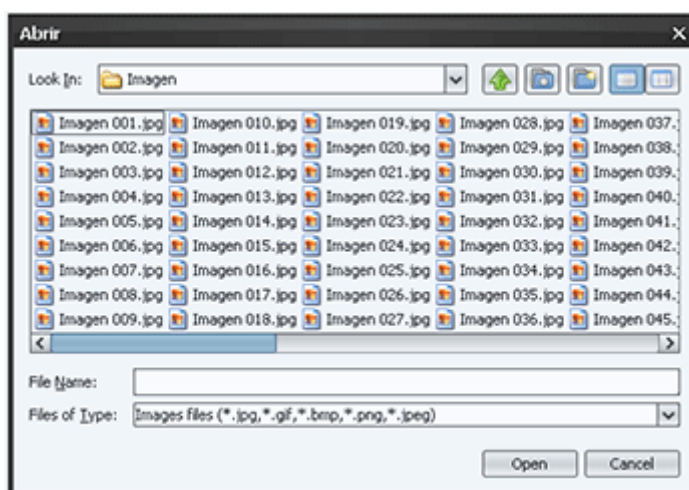


Figura 55. Diálogo para introducir la imagen de consulta

- **PARÁMETRO *-conf***

B_RANKER también necesita saber cómo y dónde buscar, información proporcionada en un archivo de configuración. Este parámetro le indica al programa dónde encontrar las instrucciones de búsqueda, que la interfaz

recoge en un fichero MPEG-7/XML, a partir de los datos introducidos por el usuario (véase 3.2.1).

- **PARÁMETRO –results**

Una vez realizada la búsqueda, B_RANKER obtendrá una lista con el conjunto de resultados, que depositará en un directorio para que la interfaz los pueda mostrar al usuario. Esta lista también se presenta en un fichero MPEG-7/XML (véase 3.2.2).

3.2 Formato MPEG-7/XML para datos de entrada y salida

El *Moving Picture Coding Experts Group* (MPEG) es un grupo de la organización ISO/IEC a cargo del desarrollo de estándares internacionales para la compresión, decompresión, procesamiento y representación codificada de imágenes en movimiento (vídeo), audio y combinación de ambas. Suyo es el estándar *MPEG-7*, denominado “interfaz de descripción de contenidos multimedia”, que se creó en el año 2001 para estandarizar la extracción de características basadas en el contenido de los diferentes tipos de información multimedia.

MPEG-7 se basa en el lenguaje de metadatos XML (*eXtensible Markup Language*) para favorecer la interoperabilidad y la creación de aplicaciones que manipulan contenido multimedia. En el caso de nuestra interfaz, se requiere que sea capaz de crear un archivo XML con formato MPEG-7 con los datos introducidos por el usuario y hacerlo llegar al programa B_RANKER. Gracias a este formato, B_RANKER entiende las instrucciones del usuario (la petición de una consulta), se ejecuta, y retorna los resultados también en un archivo MPEG-7/XML, que la interfaz interpretará para mostrar la información obtenida en pantalla.

¿Cómo deben ser estos archivos MPEG-7/XML?

3.2.1 Datos de entrada

Para que B_RANKER comprenda cómo el usuario quiere realizar una búsqueda, necesita recibir unos datos de entrada: cómo ha de buscar y dónde ha de buscar. La

interfaz recoge esta información proporcionada por el usuario, y construye el archivo MPEG-7/XML de entrada con los siguientes datos:

1. **CRITERIO DE FUSIÓN:** el usuario puede elegir entre cuatro tipos de criterio de fusión y los descriptores visuales a tener en cuenta en la búsqueda.

- a. **Tipo de criterio:**

- i. *Máximo (OR):* realiza la búsqueda tomando como distancia de ordenación la máxima obtenida.
 - ii. *Mínimo (AND):* realiza la búsqueda tomando como distancia de ordenación la mínima obtenida.
 - iii. *Media:* realiza la búsqueda promediando las distancias obtenidas para cada descriptor.
 - iv. *Pesos:* realiza la búsqueda combinando linealmente las distancias obtenidas para cada descriptor y ponderándolas con el peso indicado.

- b. **Descriptores visuales:** El estándar MPEG-7 define una serie de descriptores que permiten analizar y caracterizar el contenido audiovisual para su posterior indexación, búsqueda o comparación. B_RANKER implementa la búsqueda de imágenes a partir de cuatro de estos descriptores visuales [15], que el usuario puede escoger y combinar como desee:

- i. *Diseño del color:* este descriptor permite representar la distribución espacial del color dentro de las imágenes.
 - ii. *Estructura del color:* este descriptor caracteriza la distribución de los colores en una imagen, construyendo una especie de histograma de color en el que tendrán mayor importancia los colores que más se repartan por la imagen. Permite distinguir entre dos imágenes que tengan la misma cantidad de píxeles de un color pero con distinta distribución.

- iii. *Textura homogénea*: este descriptor utiliza un banco de 30 filtros que permite obtener una afinada descripción de las distintas texturas de la imagen para poder compararlas con las texturas de otras.
 - iv. *Histograma de bordes de textura*: este descriptor informa sobre el tipo de contornos o bordes que aparecen en la imagen. Trabaja dividiendo la imagen en 16 sub-imágenes y es capaz de analizar los tipos de borde existentes con el uso de distintos filtros que diferencian si se trata de bordes horizontales, verticales, oblicuos o aleatorios.
- c. Ponderación de los pesos**: si el criterio de fusión seleccionado es “pesos”, el usuario puede establecer un valor normalizado entre 0 y 1, en pasos de 0.1, a cada descriptor empleado en la búsqueda.
- 2. ÁMBITO DE BÚSQUEDA**: nuestro sistema de búsqueda ofrece la posibilidad al usuario de seleccionar el espacio de búsqueda para la consulta. Actualmente existen dos corpus de imágenes disponibles, correspondientes al CCMA y a Mediapro, con un gran número de imágenes clasificadas por categorías.
- 3. NÚMERO DE RESULTADOS** a devolver.

La interfaz genera un archivo XML con el formato propuesto por el estándar MPEG-7 a partir de estos datos de entrada, que guarda en disco.

- **Ejemplo de archivo XML con datos de entrada**

En este ejemplo de archivo de entrada XML para el programa B_RANKER, vemos como se configuran algunos de los parámetros requeridos para realizar una consulta a través del tag *Settings*, estableciendo el nombre del parámetro con el atributo *name* y el valor asignado con el atributo *value*:

- **ResAmount** → Número de resultados a devolver.
- **FusionCriterion** → Tipo de criterio de fusión.
- **ColorLayoutType** → Descriptor visual de diseño del color con un peso de 0.5.
- **ColorStructureType** → Descriptor visual de estructura del color con un peso de 0.5.
- **HomogeneousTextureType** → Descriptor visual de textura homogénea con un peso de 0.5.
- **TextureEdgeHistogramType** → Descriptor visual de histograma de bordes de textura.

También se determina el ámbito de búsqueda a través del tag *RelatedMaterial* con el identificador *SearchSpace*.

```
<?xml version="1.0" encoding="UTF-8"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Description xsi:type="urn:ContentEntityType" xmlns:urn="urn:mpeg:mpeg7:schema:2001">
    <MultimediaContent xsi:type="urn:MultimediaCollectionType">
      <Collection xsi:type="urn:ContentCollectionType">
        <CreationInformation>
          <Creation>
            <CreationTool>
              <Setting name="ResAmount" value="25"/>
              <Setting name="FusionCriterion" value="max"/>
              <Setting name="ColorLayoutType" value="0.5"/>
              <Setting name="ColorStructureType" value="0.5"/>
              <Setting name="HomogeneousTextureType" value="0.5"/>
              <Setting name="TextureEdgeHistogramType" value="0.5"/>
              <Setting name="UseScale" value="image"/>
            </CreationTool>
          </Creation>
          <RelatedMaterial id="SearchSpace">
            <MediaLocator>
              <MediaUri>/home/xavi/i3media//2_semantic/queryByExample/1_image/2_databases
/ccma.txt</MediaUri>
            </MediaLocator>
          </RelatedMaterial>
        </CreationInformation>
      </Collection>
    </MultimediaContent>
  </Description>
</Mpeg7>
```

En el diseño de la GUI, se debe tener en cuenta cómo recoger estos datos de configuración de la búsqueda a la hora de confeccionar la zona de formulación de la consulta, poniendo a disposición del usuario una serie de elementos que faciliten la introducción de los datos. En el capítulo dedicado al diseño de la interfaz analizaremos estos elementos (véase 5).

3.2.2 Datos de salida

Después de ejecutar la consulta, el motor de búsqueda retorna los resultados encontrados, que la interfaz recibe, interpreta y presenta al usuario. Los datos de salida también utilizan el formato XML del estándar MPEG-7, pero la interfaz se encarga de mostrarlos de una forma amigable y ofrece al usuario distintas herramientas de navegación y gestión de resultados.

La GUI recibe del motor de búsqueda la siguiente información:

1. **INFORMACIÓN SOBRE LA CONSULTA:** los parámetros básicos de entrada utilizados en la búsqueda (imagen consulta, número de resultados a devolver y criterio de fusión utilizado).
2. **COLECCIÓN DE RESULTADOS:** colección con las imágenes obtenidas como resultado con la siguiente información asociada:
 - a. **URI (*Uniform Resource Identifier*)²⁴ de la imagen:** la dirección URI de la imagen nos proporciona su ruta, el nombre del fichero y su categoría, entre otros datos.
 - b. **Posición:** los resultados son ordenados según el grado de similitud con la imagen consulta (en orden decreciente).
 - c. **Puntuación:** la distancia obtenida por la imagen, que determina su posición en la lista ordenada de resultados. Es un valor normalizado entre 0 y 1, donde 0 significa mínima similitud con la imagen consulta y 1 máxima similitud.

²⁴ Un URI es una cadena corta de caracteres que identifica inequívocamente un recurso (servicio, página, documento, dirección de correo electrónico, enciclopedia, etc.). Normalmente estos recursos son accesibles en una red o sistema.

- **Ejemplo de archivo XML con datos de salida**

En este ejemplo de archivo de salida XML del programa B_RANKER, observamos como aparecen estos datos de salida en dos tags diferenciados:

- **CreationInformation** → contiene la información sobre la consulta realizada, con una estructura muy similar al formato de entrada. Además, entre los datos que retorna, se encuentra el nombre de la herramienta que lo ejecutó (en este caso "GOS beta"), el modo de ejecución (en el ejemplo "localhost") y el momento de la consulta (con el tag *TimePoint*).
- **ContentCollection** → contiene los resultados de la búsqueda, contenidos agrupados en tags *Content* del tipo imagen (con el identificador *ImageType*). Cada imagen resultado viene acompañada por información sobre su distancia (puntuación obtenida en la búsqueda indicada por el atributo *strenght*), su posición en la lista ordenada según el grado de similitud con la consulta (indicador *id*) y su dirección URI (tag *MediaUri*).

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Description xsi:type="ContentEntityType">
    <MultimediaContent xsi:type="MultimediaCollectionType">
      <Collection xsi:type="ContentCollectionType">
        <CreationInformation>
          <Creation>
            <Creator>
              <Role>
                <Name>i3server</Name>
              </Role>
              <Agent id="localhost"/>
            </Creator>
            <CreationCoordinates>
              <Date>
                <TimePoint>Sun May 24 19:00:44 2009</TimePoint>
              </Date>
            </CreationCoordinates>
            <CreationTool>
              <Tool>
                <Name>B_RANKER alfa</Name>
              </Tool>
              <Setting MinScore="0.000000"/>
              <Setting ColorLayoutType="0.500000"/>
            </CreationTool>
          </CreationInformation>
        </Collection>
      </MultimediaContent>
    </Description>
  </Mpeg7>
```

```
<Setting ColorStructureType="0.500000"/>
<Setting TextureEdgeHistogramType="0.500000"/>
<Setting HomogeneousTextureType="0.500000"/>
<Setting numOfResults="25"/>
<Setting FusionCriterion="min"/>
</CreationTool>
</Creation>
<RelatedMaterial id="SearchSpace">
  <MediaLocator>
    <MediaUri>file:/home/xavi/i3media//2_semantic/queryByExample/1_image/2_databases/
ccma.txt</MediaUri>
  </MediaLocator>
</RelatedMaterial>
<Collection xsi:type="ContentCollectionType">
  <Creation>
    <Creator>
      <Role>
        <Name>i3user</Name>
      </Role>
      <Agent id="username"/>
    </Creator>
    <CreationTool>
      <Tool>
        <Name>GOS beta</Name>
      </Tool>
    </CreationTool>
  </Creation>
  <ContentCollection name="QueryXXX">
    <Content xsi:type="RegionsType">
      <MediaLocator>
        <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/series/1764/1764_001
00020.jpg</MediaUri>
      </MediaLocator>
    </Content>
  </ContentCollection>
</Collection>
</CreationInformation>
<ContentCollection>
  <Content xsi:type="ImageType">
    <Image>
      <Relation type="score" id="1" strength="0.823438"/>
      <MediaLocator xsi:type="ImageLocatorType">
        <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1034/1034_00000
000.jpg</MediaUri>
      </MediaLocator>
    </Image>
  </Content>
  <Content xsi:type="ImageType">
    <Image>
      <Relation type="score" id="2" strength="0.814062"/>
```

```
<MediaLocator xsi:type="ImageLocatorType">
  <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1004/1004_00001
724.jpg</MediaUri>
</MediaLocator>
</Image>
</Content>
.
.
.
<Content xsi:type="ImageType">
  <Image>
    <Relation type="score" id="25" strength="0.750000"/>
    <MediaLocator xsi:type="ImageLocatorType">
      <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1003/1003_00010
921.jpg</MediaUri>
    </MediaLocator>
  </Image>
</Content>
</ContentCollection>
</Collection>
</MultimediaContent>
</Description>
</Mpeg7>
```

La forma en que la GUI presenta los resultados en pantalla debe permitir al usuario acceder al máximo de información disponible, utilizando los elementos gráficos más adecuados según el tipo de datos recibidos. Analizaremos estos elementos en el capítulo dedicado al diseño de la interfaz (véase 5).

4. Entorno de Desarrollo

El GOS forma parte de un conjunto de aplicaciones gráficas y herramientas que automatizan procesos de gran complejidad como son la indexación, selección, búsqueda y recuperación de imágenes y vídeos a través de grandes repositorios de datos distribuidos. Todas las piezas de ese conjunto deben encajar para conseguir que todo funcione a la perfección.

4.1 Entorno de ejecución

En el capítulo anterior hemos visto los requisitos que el programa B_RANKER necesita para ejecutarse. La interfaz se encarga de realizar la llamada a B_RANKER cumpliendo estos requisitos, pero también establece sus propias condiciones para que el proceso de búsqueda y visualización de resultados se realice correctamente.

Para el correcto funcionamiento del GOS, el motor de búsqueda B_RANKER debe encontrarse en la siguiente ubicación:

```
$HOME/SoftImage/bin/release/
```

donde la variable `$HOME` corresponde al directorio de trabajo del usuario.

Actualmente, el GOS también utiliza una estructura de ficheros *hard-coded*²⁵ para gestionar la información relacionada con el espacio de búsqueda. Las ubicaciones de los ficheros del espacio de búsqueda y el directorio de resultados deben ser las siguientes:

- **Para el directorio de los ficheros del espacio de búsqueda:**

```
$HOME/2_semantic/queryByExample/1_image/2_databases
```

- **Para el directorio de resultados:**

```
$HOME/2_semantic/queryByExample/1_image/3_results
```

²⁵ Término del mundo de la informática, también conocido como grabado a fuego. Esta práctica consiste en incrustar datos directamente en el código fuente del programa, en lugar de obtener esos datos de una fuente externa. (http://es.wikipedia.org/wiki/Hard_code)

En el directorio del espacio de búsqueda se encuentran una serie de ficheros de texto (.txt) donde se listan las direcciones de los archivos descriptores (.xml) de las imágenes pertenecientes a cada espacio de búsqueda (bases de datos). Es requisito indispensable que estos ficheros se hayan generado antes de realizar una búsqueda. En el anexo II encontramos ejemplos de estos ficheros.

El GOS dispone de un archivo de configuración (*configuration.propiedades*) para definir las categorías de los espacios de búsqueda que se mostrarán en la interfaz, y que el usuario podrá seleccionar a la hora de realizar sus consultas. Estas categorías deben corresponderse con los ficheros de texto (.txt) disponibles en el directorio de ficheros del espacio de búsqueda. Así, en este directorio se encuentran los ficheros *ccma.txt*, *agency.txt*, *news.txt*, *sports.txt*, etc.

```
#-----  
#PROPERTIES TO CONFIGURATE SEARCHSPACES  
#-----  
  
#Categories to searchspaces  
searchspace.categories=ccma,mediapro  
searchspace.categories.ccma=agency,news,parlament,series,sports,talkshow  
searchspace.categories.mediapro=BarsaVilareal,MallorcaValladolid,SaragossaCoru  
nya,SevillaEspanyol,SevillaGetafe
```

Figura 56. Archivo de configuración del espacio de búsqueda

La definición de la estructura de ficheros directamente en el código fuente condiciona el entorno de ejecución. El GOS implementa dos modos de ejecución, en local y en remoto. El usuario puede decidir qué modo de ejecución desea a través de la interfaz:

1. **Local:** si se dispone del ejecutable del B_RANKER en la máquina del usuario. Además, el GOS también condiciona al usuario a tener la misma estructura de directorios en local para poder realizar búsquedas.
2. **Remoto:** B_RANKER se puede ejecutar en modo remoto a través de un servicio web sobre SOAP²⁶. El servidor *upseek* (fig. 57) del GPI dispone este servicio.

²⁶ **SOAP** (siglas de *Simple Object Access Protocol*) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos

Cualquier modificación de la estructura de directorios implica una modificación del código fuente, sino se producirían errores de ejecución. Esta estructura de directorios *hard-coded* es provisional, ya que en un futuro, los espacios de búsqueda serán leídos directamente de disco (en el caso de ejecución en local) o consultados al servidor (en caso de ejecución remota).

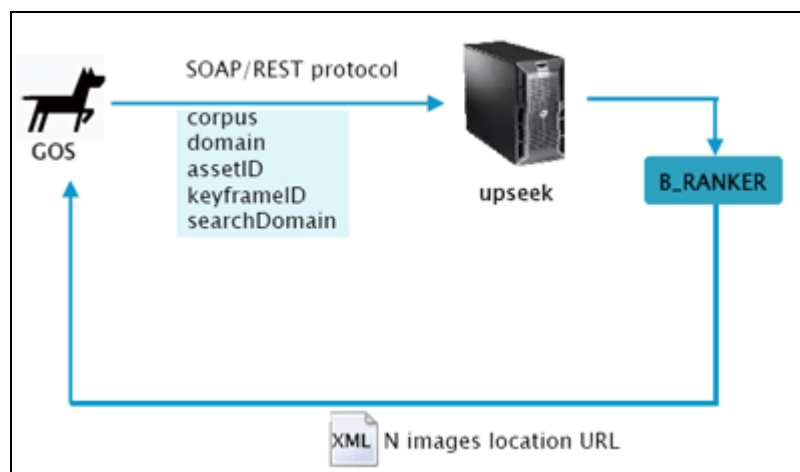


Figura 57. Ejecución remota de GOS

4.2 Entorno de trabajo

El grupo de imagen (GPI) de la UPC dispone de una buena infraestructura para el desarrollo de proyectos relacionados con el departamento. Todos los proyectistas e investigadores utilizan como base el mismo entorno de trabajo, que facilita la gestión de recursos, la organización de las tareas y la colaboración entre miembros del grupo.

Esta infraestructura base se compone de los siguientes elementos:

- **REPOSITORIO:** el GPI utiliza un repositorio central para albergar el código generado por todos los desarrolladores. Un repositorio es un lugar para almacenar información (por ejemplo un conjunto de máquinas dónde se encuentran localizadas varias bases de datos o grandes grupos de ficheros). Los usuarios del repositorio suelen acceder a esta información a través de la red.

XML. Está actualmente bajo el auspicio de la W3C y es uno de los protocolos más utilizados en los servicios web. (<http://es.wikipedia.org/wiki/SOAP>)

- **SERVIDOR CENTRAL CON ACCESO REMOTO:** d5lnx10.upc.es (147.83.50.71), donde se encuentra instalado el software necesario para desarrollar.
- **IMATGEWIKI (WIKI OF THE IMAGE AND VIDEO PROCESSING GROUP):** Wiki²⁷ con información, recursos y documentación útil.
- **CURSO I3MEDIA DEL MOODLE**²⁸ del Departamento de *Teoria del Senyal i Comunicacions* (TSC) de Terrassa.
- **LABORATORIO:** sala equipada con los medios necesarios para realizar pruebas y demos.

Para realizar el desarrollo de la interfaz del GOS utilizaremos esta infraestructura como base.

Cuando nos planteamos el desarrollo de una GUI, es necesario tomar una serie de decisiones relacionadas con el entorno de trabajo, que debe facilitarnos al máximo la tarea, y la tecnología empleada en la implementación, que debe ser la adecuada para establecer la comunicación necesaria entre el usuario y el sistema. Como hemos visto anteriormente, existen dos tipos de interfaz utilizadas en aplicaciones de búsquedas de imágenes: interfaces *web-like*, que utilizan tecnologías para aplicaciones en la web (lenguaje HTML, PHP, Applets de Java, Flash, etc.), e interfaces *window-like*, que utilizan tecnologías que permiten crear programas más complejos (como Java, C++, Perl, etc.). La tendencia de todos los sistemas es a integrarse en redes, y especialmente, en la web, por lo que la mejor opción para desarrollar una herramienta flexible y versátil es **Java**.

²⁷Un **wiki**, o una **wiki**, es un sitio web cuyas páginas pueden ser editadas por múltiples voluntarios a través del navegador web. Los usuarios pueden crear, modificar o borrar la información que comparten.

ImatgeWiki: <https://147.83.50.70/ImatgeWiki/> (es necesario tener una cuenta de usuario para acceder)

²⁸ **Moodle** es un sistema de gestión de cursos, de distribución libre, que ayuda a los educadores a crear comunidades de aprendizaje en línea. Este tipo de plataformas tecnológicas también se conocen como LMS (Learning Management System).

Moodle del TSC: <http://terrassatasc.upc.edu/> (es necesario tener una cuenta de usuario para acceder a los cursos)

4.2.1 Java

Java es un lenguaje de programación creado por Sun Microsystems²⁹, ideal para desarrollar aplicaciones de todo tipo. Nos interesa especialmente por las siguientes características:

- **Crea programas interplataforma:** Java debe su gran éxito a ser un lenguaje independiente de la plataforma donde se ejecuta, que evita generar versiones distintas de las aplicaciones según el sistema operativo que se utilice.
- **Crea interfaces gráficas de usuario:** Java proporciona una serie de librerías que implementan elementos gráficos para crear interfaces gráficas de forma sencilla (como AWT y SWING).
- **Dispone de gran variedad de librerías:** a parte de las librerías para GUI, Java dispone de múltiples librerías útiles para realizar todo tipo de funcionalidades (lectura/escritura de archivos, parseo de XML, conexión con bases de datos, etc.)
- **Permite trabajar en grupo:** la filosofía de Java posibilita la colaboración entre los desarrolladores, que bajo la misma tecnología implementan sistemas y arquitecturas complejas.
- **Crea aplicaciones que se comunican a través de la web:** Java dispone de mecanismos para integrar fácilmente sus aplicaciones en cualquier red, especialmente en la web, a través de *webservices*.

¿Cómo funciona el desarrollo con Java? Todos los expertos coinciden en alabar la robustez y la reducción de la complejidad a la hora de programar que ofrece el lenguaje Java [17]. No es objeto de este documento describir con detalle la plataforma de Java, veamos simplemente sus componentes básicos.

La plataforma de Java se compone de tres partes básicas:

1. **API (*Application Programming Interface*) de Java**, que contiene las clases básicas que utiliza el lenguaje Java. Cada clase define un conjunto de tipos de objetos y los métodos necesarios para trabajar con ellos. Estas

²⁹ <http://java.sun.com/>

clases se organizan en paquetes (librerías), normalmente según su funcionalidad. El desarrollador puede crear nuevas clases específicas para su aplicación, que también organizará en paquetes, a partir de esta API.

2. **JRE (Java Runtime Environment)**, el entorno de ejecución de Java que se utiliza para ejecutar las aplicaciones. El JRE proporciona la API y la JVM (Java Virtual Machine)³⁰.
3. **JDK (Java Development Kit)**³¹, que incluye un conjunto de herramientas para desarrollar aplicaciones, como el compilador de Java a código de bytes, un generador de documentación, el depurador de programas (*debugger*), etc. También incluye el JRE.

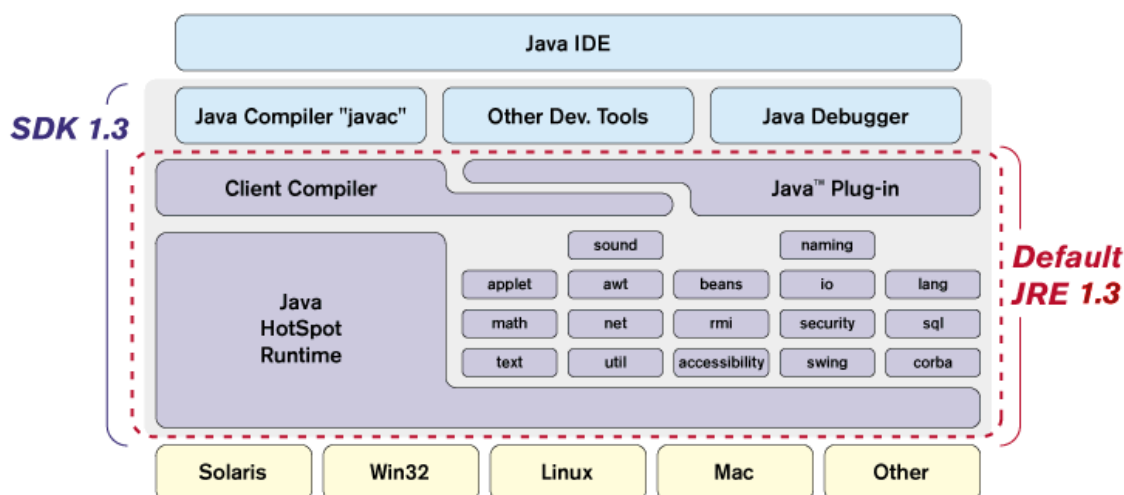


Figura 58. Plataforma JAVA

Sun Microsystems distribuye gratuitamente desde su página web diferentes entornos de desarrollo para la plataforma Java (con la API, el JRE y el JDK correspondientes). Actualmente, todos los programadores de Java utilizan entornos de

³⁰ Una **Máquina virtual Java (JVM)** es un programa nativo (ejecutable en una plataforma específica), capaz de interpretar y ejecutar instrucciones expresadas en un código binario especial (el Java bytecode), el cual es generado por el compilador del lenguaje Java. (http://es.wikipedia.org/wiki/M%C3%A1quina_virtual_Java)

³¹ SDK (Java Software Development Kit) used to be called the Java Development Kit (JDK) before the marketing department at Sun got crazy with the "tm" and terminology. For political reasons & for sanity, they call the meaningful names (jdk) & versions (1.2 / 1.3 / 1.4 1.5 / 1.6) "engineering" terms. The marketing terms are "Java2 platform" (aka jdk 1.2 thru 1.4) or Java5 (aka jdk 1.5) or Java6 (aka jdk1.6).

desarrollo integrados (*Integrated Development Enviroment* – IDE), programas compuestos por un conjunto de herramientas útiles a la hora de crear aplicaciones. El GOS ha sido implementado en **Eclipse**, uno de los IDE de Java más utilizados y completos que existen actualmente.

4.2.2 Eclipse

Eclipse es un IDE de código libre, muy utilizado gracias a su extensibilidad. El usuario dispone de las herramientas básicas para desarrollar incluidas en el software básico de Eclipse, pero además dispone de múltiples plug-ins y módulos para instalar y ampliar el programa. Podemos descargar Eclipse desde su página web³², donde se encuentran disponibles versiones para Windows, MAC OS y Linux (32bit y 64bit).

Para desarrollar nuestra interfaz hemos utilizado la versión de Eclipse *Ganymede* para Windows³³ (Eclipse Platform, v.3.3.2), y le hemos añadido un sistema de control de versiones llamado **Subversion (SVN)** para poder compartir el código fuente con otros desarrolladores del proyecto i3media.

4.2.3 Subversion (SVN)

Cualquier programador que participe en un proyecto en equipo necesita utilizar un sistema de control de versiones. Estos sistemas permiten administrar y guardar cualquier documento que se modifique con frecuencia, sin miedo a que se produzcan conflictos entre los cambios realizados por diferentes usuarios del sistema.

Los miembros del proyecto i3media del GPI utilizan un repositorio central para almacenar el código. Este repositorio se gestiona a través de *Subversión (SVN)*, un sistema de control de versiones de código abierto y de distribución libre, disponible como *plug-in* para el Eclipse. El programa cliente SVN para Eclipse es **Subclipse**³⁴, y se instala fácilmente a través de la herramienta de actualización de software de Eclipse.

³² <http://www.eclipse.org/downloads/>

³³ <http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/ganymede/SR2/eclipse-je-ganymede-SR2-win32.zip>

³⁴ <http://subclipse.tigris.org/update>

Una vez instalado el Subclipse, al acceder al repositorio encontraremos un árbol de directorios y ficheros que podremos descargar en nuestra máquina. Cada desarrollador sincroniza su copia local con el repositorio cada vez que realiza cambios importantes en la aplicación o necesita actualizar su copia con los cambios realizados por otros desarrolladores. El repositorio realiza las funciones de un servidor de ficheros ordinario, pero además recuerda todos cambios realizados, permitiendo recuperar versiones antiguas o examinar el historial de cambios si se producen conflictos entre versiones.

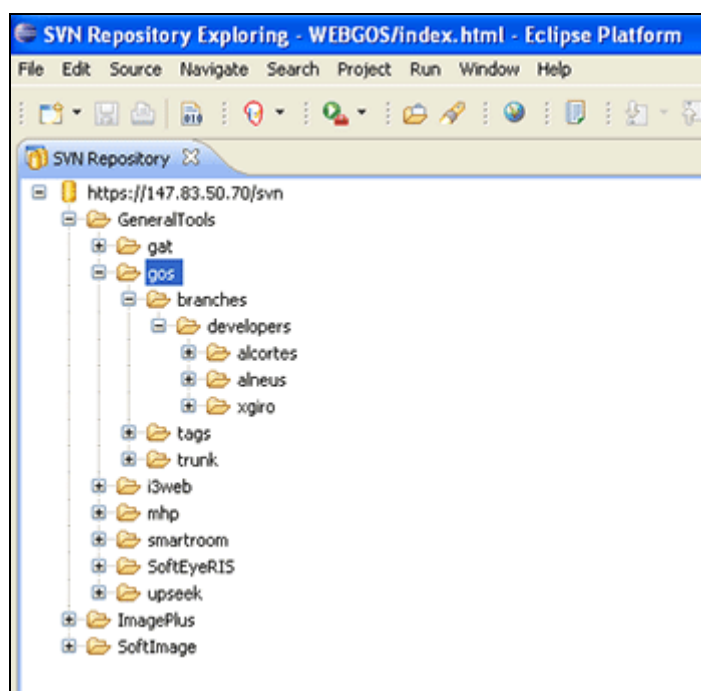


Figura 59. Repositorio en SVN de Eclipse

4.2.4 NoMachine

El GPI dispone de un servidor de alta capacidad al cual se puede acceder de forma remota: d5lnx10.upc.es (147.83.50.71). El grupo utiliza la aplicación **NoMachine NX Server**³⁵ para realizar conexiones remotas a esta máquina central donde se encuentran las herramientas desarrolladas en el proyecto i3media. *NoMachine* es una

³⁵ NoMachine NX está basado en el protocolo de comunicación "X Windows" y ofrece acceso a aplicaciones hospedadas en cualquiera de los sistemas operativos conocidos. (<http://www.nomachine.com/>)

solución de arquitectura Cliente / Servidor de software que permite implementar un acceso centralizado y seguro a las aplicaciones albergadas en esta máquina remota.

Para desarrollar el GOS necesitaremos realizar conexiones periódicas al d5lnx10.upc.es, donde se encuentran el motor de búsqueda y el corpus de datos con las imágenes cedidas por Mediapro y el CCMA para el proyecto i3media.

5. Diseño de la Interfaz Gráfica

El usuario final del GOS probablemente no será un experto en informática, pero sí será una persona habituada a trabajar con interfaces gráficas y estará familiarizado con los elementos comunes que estas aplicaciones ofrecen al usuario para realizar tareas concretas de forma rápida y eficaz. Nuestro reto es diseñar una GUI que responda a estos tres objetivos:

- 1. Ser una interfaz fácil de aprender y utilizar:** los usuarios menos familiarizados con sistemas de recuperación de imágenes mediante QbE habrán de ser capaces de manejar sin problemas la aplicación después de una breve sesión de demostración.
- 2. Cumplir las necesidades de interacción entre usuario y sistema de búsqueda:** la interfaz es la pieza clave entre el diálogo usuario – máquina, y por tanto, deberá ser capaz de transmitir la información necesaria para que ambos interlocutores entiendan los datos que se intercambian.
- 3. Ofrecer los elementos apropiados para realizar las tareas requeridas:** la presentación de la información, tanto a la hora de recoger los datos de entrada como de mostrar los datos de salida, debe ser adecuada y facilitar el trabajo en todo momento al usuario.

La consecución de estos tres objetivos ya nos garantiza una buena herramienta en términos de usabilidad, pero además, queremos que nuestra interfaz sea estéticamente atractiva, que agrade al usuario y que posea un estilo que la identifique y la diferencie de otros productos similares.

5.1 Estilo de la interfaz

Hemos visto anteriormente como muchos sistemas CBIR desarrollan herramientas de anotación de imágenes que complementan los buscadores. Así, Photobook tiene la aplicación de anotación FourEyes o el buscador Emir se distribuye junto a Caliph. Estas interfaces de aplicaciones complementarias suelen utilizar el mismo estilo en su diseño como marca de imagen corporativa.

Dentro del proyecto i3media existe una herramienta de anotación de imágenes, el GAT (*Graphic Annotation Tool*)³⁶ desarrollada con anterioridad al GOS. Para dotar a ambas herramientas de una apariencia similar, que las identifique como partes de un mismo conjunto, el estilo gráfico del GAT ha sido aplicado al GOS.

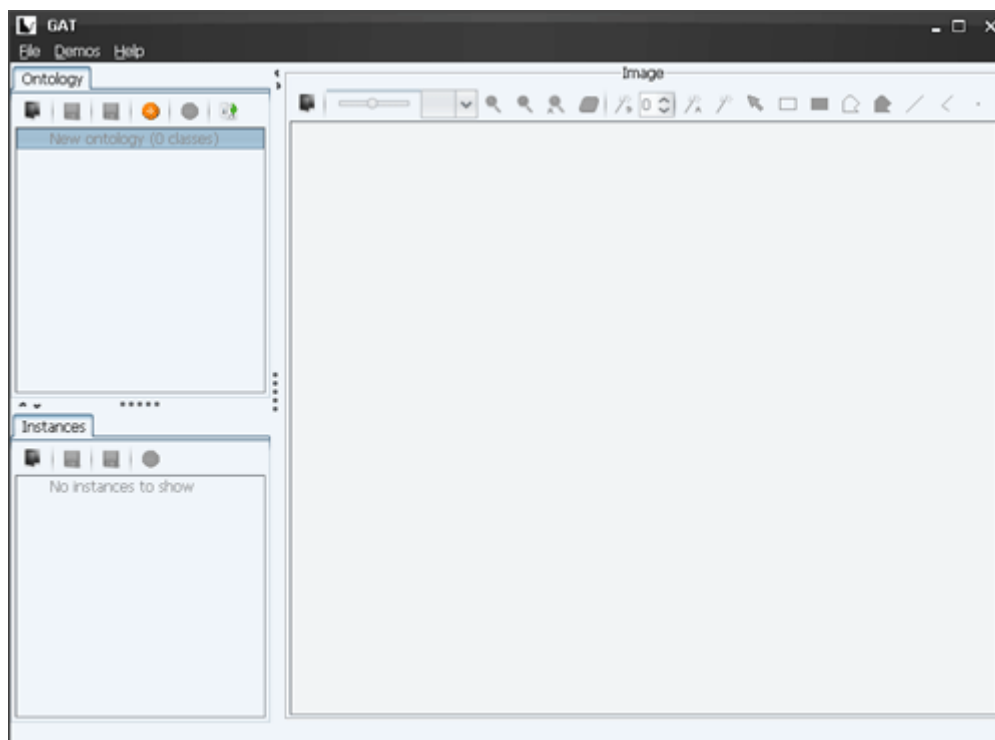


Figura 60. Interfaz del GAT

El nombre de las herramientas también juega un papel importante en este sentimiento de pertenencia a un conjunto. El GAT, “gato” en catalán, se complementa con el GOS, “perro” en catalán. El *pack* del “GAT & GOS” suena atractivo y divertido para los usuarios, al estilo del juego de palabras que SourceForge realiza con su Caliph & Emir.

5.1.1 Diseño del logo

El logo del GOS simboliza un perro de muestra (en inglés *pointing dog*), un tipo de perro utilizado para la caza, apreciado por su habilidad rastreadora. Este tipo de

³⁶ Web del GAT: http://gps-tsc.upc.es/imatge/i3media/gat/index_es.html

perros adoptan la famosa postura de estatua³⁷, apuntando el lugar donde se encuentra la presa sin lanzarse sobre ella y mostrándole a su dueño dónde se esconde. El significado visual de esta postura define muy bien el talante del GOS: por un lado es una herramienta de ayuda para realizar una búsqueda, al igual que el perro de caza es un ayudante imprescindible para el cazador, y por otro guía al usuario para encontrar el resultado que busca, al igual que el perro rastreador indica a su dueño dónde está la presa que quiere cazar.

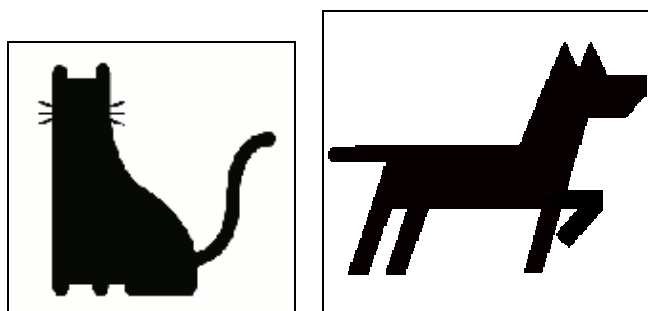


Figura 61. Logos del GAT y el GOS

El diseñador del logo del GAT, Carles Ballesteros³⁸, también ha ayudado a la creación de un logo para el GOS, siguiendo el mismo estilo, una silueta estilizada en blanco y negro, y en 2D.

5.1.2 Look&Feel³⁹

La apariencia general de la interfaz del GOS, lo que se conoce como *Look&Feel*, hereda sus características básicas del GAT. Ambas aplicaciones utilizan uno de los *skins* desarrollados en el proyecto *Substance* de Java.net⁴⁰, dedicado a la creación de librerías para dotar a las aplicaciones de Swing (una biblioteca gráfica de Java) de una apariencia visualmente atractiva y coherente [18]. Un *skin* (traducido

³⁷ <http://www.perrosycaza.com/razas/57-pointer>

³⁸ <http://carlesballesteros.blogspot.com/>

³⁹ **Look and feel** is a term used in descriptions of products and fields such as product design, marketing, branding and trademarking, to describe the main features of its appearance. (http://en.wikipedia.org/wiki/Look_and_feel)

⁴⁰ java.net is the realization of a vision of a diverse group of engineers, researchers, technologists, and evangelists at Sun Microsystems, Inc. to provide a common area for interesting conversations and innovative development projects related to Java™ technology. (<http://www.java.net/about.csp>)

como piel) es el conjunto de elementos gráficos que, al aplicarse sobre un determinado software, modifican su apariencia externa.

El *skin* seleccionado para el GAT y el GOS es el denominado *SubstanceBusinessBlackSteelLookAndFeel*⁴¹, que dota a las interfaces de una apariencia moderna y elegante. La utilización de cualquiera de los *skin* de Substance requiere de Java 6.0 o superior para funcionar.

Este *skin* utiliza el negro y el azul como colores básicos para todos los elementos de la interfaz. Como veremos a continuación, el diseño del GOS ha respetado esta base y ha implementado el resto de elementos gráficos con estos colores, para transmitir un entorno gráfico equilibrado y homogéneo.

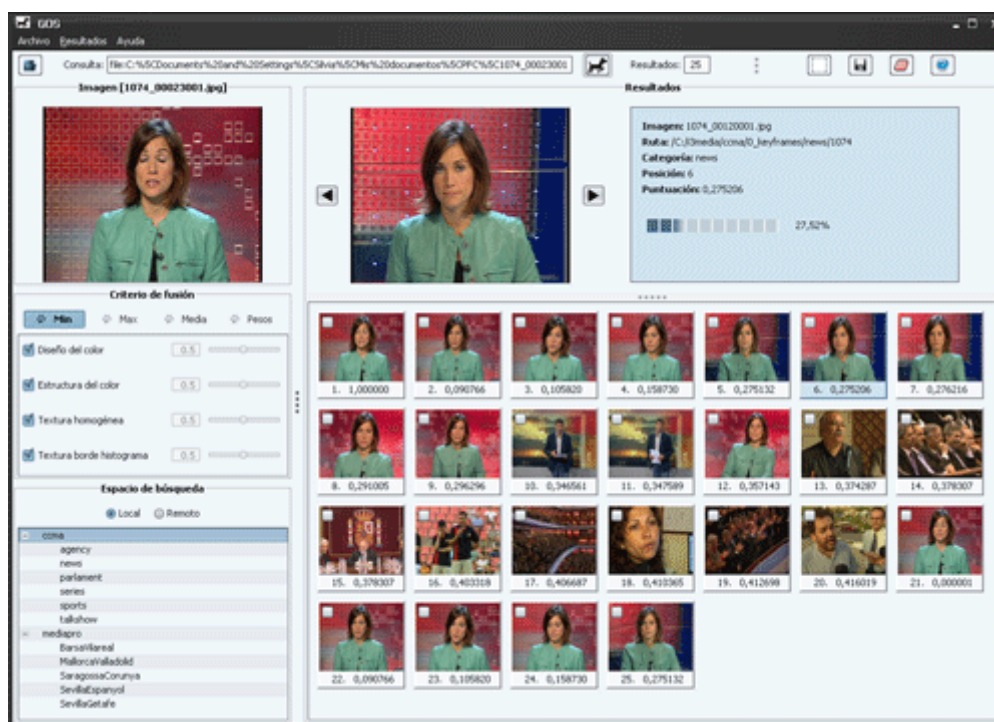


Figura 62. Interfaz del GOS

5.1.3 Iconos

Un icono es la representación gráfica de acciones que podemos desarrollar en la interfaz. Normalmente ocupan muy poco espacio, por lo que son muy utilizados en barras de herramientas y menús. Algunos iconos son ya universales, comprendidos

⁴¹ Librería: org.jvnet.substance.skin.SubstanceBusinessBlackSteelLookAndFeel

por gran cantidad de usuarios, debido a su amplia difusión y utilización tanto en programas como en la web.

El GAT y el GOS también comparten algunos iconos, para aquellas tareas genéricas fácilmente reconocibles para cualquier usuario. El objetivo es el mismo, transmitir que forman parte del mismo grupo de herramientas. Con todo, se han diseñado nuevos iconos específicos para el GOS y algunas funcionalidades concretas.



Figura 63. Iconos comunes del GAT y el GOS para “Guardar”, “Borrar” y “Ayuda”



Figura 64. Iconos específicos del GOS para “Buscar” y “Seleccionar”

5.2 Elementos básicos de la GUI

Para familiarizarnos con los componentes básicos de una GUI, vamos a enumerar los utilizados en la interfaz del GOS. Los listaremos alfabéticamente junto a su componente correspondiente en Java y describiremos qué función desempeñan dentro de una aplicación:

- **ADMINISTRADOR DE DISEÑO (LAYOUT)**

Utilizado para organizar los componentes situados dentro de un contenedor, posicionándolos y fijando su tamaño dentro del área disponible del contenedor según un esquema de distribución determinado.

Disponemos de:

- **FlowLayout:** dispone los elementos de izquierda a derecha y de arriba abajo.
- **BoxLayout:** distribuye los componentes en una sola fila o columna.

- **GridLayout.** despliega los componentes en filas y columnas espaciadas regularmente.
- **BorderLayout.** distribuye los objetos en cinco puntos geográficos del contenedor (norte, sur, este, oeste y centro).
- **GridBagLayout.** organizador más flexible que permite posicionar los componentes con relación a sí mismos utilizando restricciones.

- **ÁRBOL (JTREE)**

Componente utilizado para representar información jerarquizada (el ejemplo de árbol que todos conocemos es la representación de los archivos y directorios de una unidad de disco).

- **BARRA DE HERRAMIENTAS (JTOOLBAR)**

Componente para crear barras de herramientas. Es un contenedor que agrupa varios elementos, normalmente botones con iconos, para visualizarlos en fila o columna.

- **BARRA DESLIZANTE (JSLIDER)**

Componente utilizado para facilitar al usuario la entrada de un valor numérico limitado por un valor mínimo y máximo.

- **BORDE (BORDERFACTORY)**

Componente que permite crear cualquier tipo de borde decorativo aplicable a cualquier elemento gráfico.

- **BOTÓN (JBUTTON)**

Componente que se puede presionar para ejecutar una acción.

- **BOTONES DE OPCIÓN (JRADIOBUTTON)**

Elemento que puede ser seleccionado o deseleccionado, y que suele formar parte de un grupo (*ButtonGroup*) en el que sólo un botón puede estar seleccionado al mismo tiempo.

- **CAMPO DE TEXTO (JTEXTFIELD)**

Componente para introducir y mostrar cadenas de texto.

- **CASILLAS DE VERIFICACIÓN (JCHECKBOX)**

Elemento que puede ser seleccionado o deseleccionado. Al contrario que los botones de opción, si forman parte de un grupo, pueden estar seleccionados varios al mismo tiempo.

- **CUADRO DE TEXTO (JTEXTAREA)**

Componente que muestra múltiples líneas de texto.

- **DIÁLOGO (JDIALOG)**

Componente para crear subventanas al margen de la aplicación principal y mostrar mensajes temporales al usuario (errores, avisos, etc.).

- **ETIQUETA (JLABEL)**

Componente que muestra cadenas de texto.

- **IMAGEN (IMAGE)**

Elemento empleado para representar imágenes gráficas, que posteriormente podremos visualizar dentro de la interfaz.

- **MENÚ (JMENU)**

Componente para crear menús desplegables, que generalmente se agrupan en una barra de menú (JMenuBar).

- **PANEL (JPANEL)**

Contenedor genérico para albergar todo tipo de componentes.

- **VENTANA (JFRAME)**

Contenedor de alto nivel, componente que no puede ser contenido por otro, normalmente la ventana principal de la aplicación.

Veamos a continuación cómo se usan estos componentes en nuestra interfaz.

5.3 Organización de los elementos

Al enfrentarnos al diseño de una GUI no debemos descuidar tres aspectos fundamentales que darán forma a la aplicación:

- **Diseño visual:** el componente visual y estético de las interfaces da forma a la aplicación, la dota de identidad y refuerza su funcionalidad.
- **Usabilidad:** un diseño que no es funcional y no facilita la utilización de la aplicación, no es un buen diseño.
- **Tecnología:** se pone al servicio de una interfaz usable y atractiva.

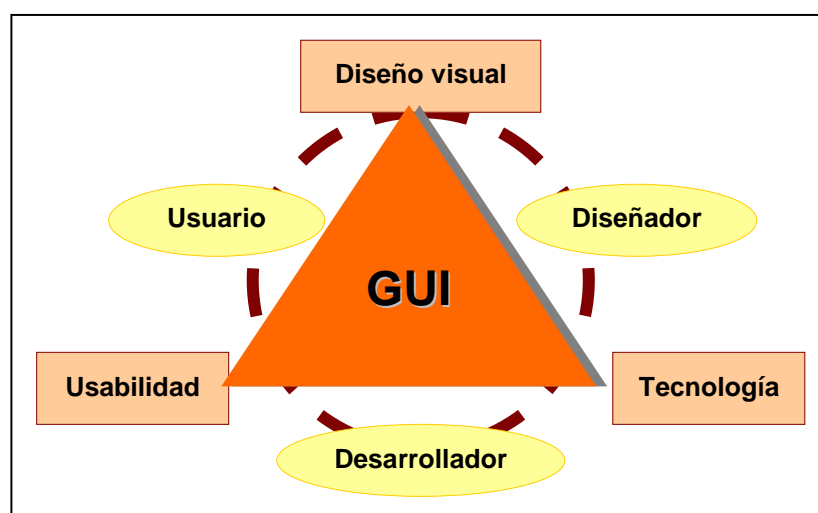


Figura 65. Los tres ángulos y los tres roles del diseño de GUI

Estas tres perspectivas se relacionan a través de los diferentes roles que intervienen en el proceso de implementación de la interfaz. El desarrollador se encarga de programar una herramienta que realiza determinadas funciones, su prioridad es utilizar la tecnología disponible para crear una interfaz funcional, usable. El usuario utiliza esta interfaz, interactúa con sus elementos gráficos que deben facilitarle el trabajo. El diseñador se preocupa que estos elementos gráficos representen la información de la mejor manera posible, aunando estética y tecnología.

La primera decisión importante que debemos tomar es la distribución de los elementos en la interfaz. Sabemos que un buscador de imágenes mediante consulta por ejemplos debe diferenciar dos zonas en pantalla, la zona destinada a la formulación de la consulta y la zona destinada a la visualización de los resultados. Pero ¿dónde debemos colocar estas áreas para optimizar la utilización de la interfaz?

5.3.1 Patrones de lectura de pantalla

Existe un patrón genérico de lectura adquirido por el hombre, determinado por su cultura, y que es válido tanto para la lectura en papel como en pantalla. La dirección de lectura horizontal, de izquierda a derecha y de arriba abajo, característica de la cultura occidental, se aplica también al diseño de GUI. Dentro de la disciplina de la HCI se han realizado numerosos estudios de usabilidad a través de técnicas de *eyetracking*⁴² para establecer los hábitos del usuario frente a la información presentada en pantalla [19] [20]. En el gráfico de la figura 66 vemos como se distribuyen los puntos con mayor atención visual por parte del usuario:



Figura 66. Zonas de importancia en la pantalla.

Estas zonas ayudan al diseñador a distribuir los elementos gráficos en función de su importancia. En general, las áreas de máximo interés deben situarse en la parte superior izquierda, generalmente destinadas al logo y nombre de la aplicación (elementos clave sobre todo en términos comerciales) y al menú principal con las funciones básicas más importantes. Todos los programas que conocemos y utilizamos, presentan esta estructura. La organización del resto de elementos por la pantalla responderá a las necesidades de la aplicación.

Como referente, un reciente estudio de usabilidad realizado por Jacob Nielsen, conocido como el “gurú” de la usabilidad, ha establecido el patrón F de lectura digital (en inglés denominado “*F-Pattern for reading*”) [21], aplicado especialmente a la

⁴² El *eyetracking* es una tecnología que permite seguir los movimientos oculares de una persona para inferir qué mira y qué ve, en concreto en qué áreas fija su atención, durante cuánto tiempo y qué orden sigue en su exploración visual.

lectura de páginas web. Este patrón determina que los ojos de los usuarios se mueven siguiendo una forma de F, primero realizando un movimiento horizontal por la parte superior del área de contenido, luego realizando un segundo desplazamiento horizontal muy cerca del área del primer desplazamiento, y por último, escanean el lado izquierdo de la pantalla con un movimiento vertical.



Figura 67. Patrón F de lectura típico de las páginas web.

La lógica del patrón F, en consonancia con la dirección de lectura asumida por nuestra cultura, nos lleva a establecer una primera distribución básica de elementos en la interfaz del GOS (fig. 68).

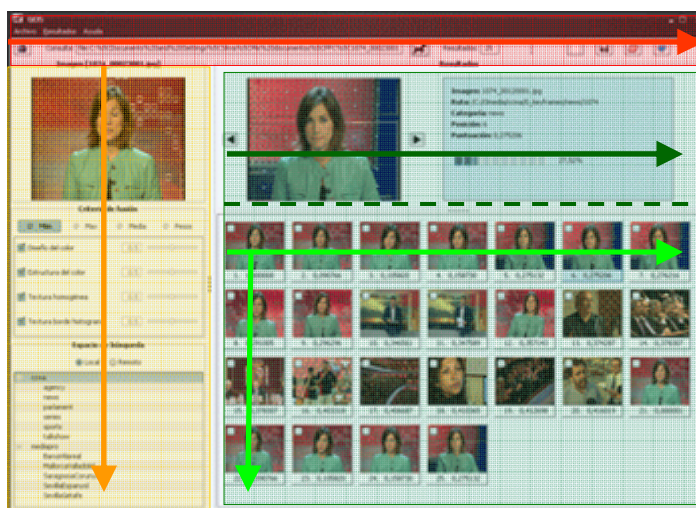


Figura 68. Organización de elementos en pantalla del GOS.

La zona superior se destina al menú principal y la barra de herramientas, con una organización horizontal de izquierda a derecha (zona roja de la figura). En la zona izquierda ubicamos el área de consulta, con una organización vertical de arriba a abajo (zona amarilla de la figura). El resto de pantalla se utiliza para la visualización de los resultados, que vuelve a reproducir el esquema anterior: en la parte superior del área

de resultados se coloca una zona destinada a la presentación de información destacada (una imagen resultado junto a un área de texto con la información asociada a esta imagen), y en la parte inferior se visualizan todos los resultados de la búsqueda en forma de parrilla de imágenes, ordenados en filas (zona verde de la figura).

5.3.2 Presentación de la información

Aprovecharemos la arquitectura jerárquica de Java para analizar los diferentes componentes que van dando forma a la aplicación.

5.3.2.1 Ventana principal

La ventana principal de la aplicación es el contenedor de todos los elementos gráficos de la interfaz. El objeto *JFrame* que nos proporciona Java incluye los controles habituales de interfaces *window-like* para cambiar de tamaño y desplazar la ventana, y los botones para minimizar, maximizar y cerrar la aplicación.

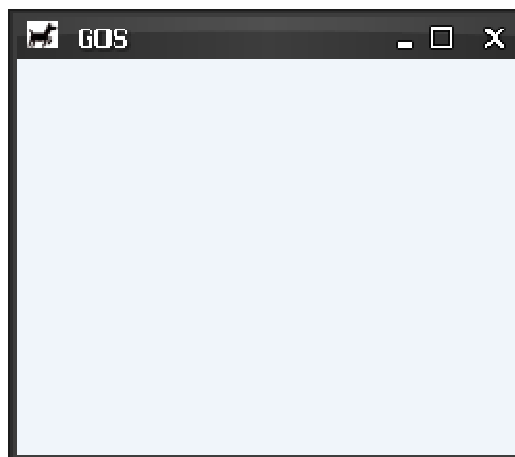


Figura 69. Ventana principal del GOS.

La ventana muestra el logo y el nombre de la aplicación en la zona superior izquierda del marco. Dentro de esta zona superior, los objetos *JFrame* ofrecen un elemento *JMenuBar* para configurar el menú principal de la aplicación (véase 5.3.2.2).

Los objetos *JFrame* disponen también de un elemento contenedor llamado *ContentPane* para albergar el siguiente nivel de componentes. Este contenedor utiliza

un administrador de diseño *BorderLayout*, que nos permite ubicar en la zona norte la barra de herramientas general (véase 5.3.2.3) y destinar el espacio restante a organizar las dos áreas de la aplicación. Para ello Java nos proporciona un panel especial para visualizar dos componentes simultáneamente, el *JSplitPane*, horizontal o verticalmente. Para nuestra interfaz configuraremos el panel separador en modo horizontal y colocaremos a la izquierda un panel contenedor con los elementos del área de consulta (véase 5.3.2.4) y a la derecha un segundo panel contenedor con los elementos del área de visualización de resultados (véase 5.3.2.5).

5.3.2.2 Menú general

El menú principal de una aplicación ocupa un lugar destacado en la pantalla. Hemos visto como el objeto *JFrame* permite la inclusión de una barra de menú (*JMenuBar*) en la ventana de la interfaz, situando el menú dentro del marco superior de la aplicación, inmediatamente después del logo y el nombre. El *skin* de nuestra interfaz resalta el marco de la ventana en color negro, y el menú principal también queda resaltado del resto de componentes al formar parte del marco.

Utilizamos esta barra de menú para configurar los tres menús generales que presentan al usuario las acciones más destacadas de la aplicación, disponibles también a través de otros elementos de la interfaz, pero a las que el usuario puede acceder de manera rápida y sencilla a través de este menú superior.

Cada menú, objeto *JMenu*, contiene una serie de ítems asociados a la categoría del menú. El GOS ofrece estos tres menús:

1. Menú Archivo, que contiene las acciones:

- *Nueva Imagen*: cargar una imagen de consulta.
- *Buscar*: lanzar una consulta.
- *Salir*: cerrar la aplicación.

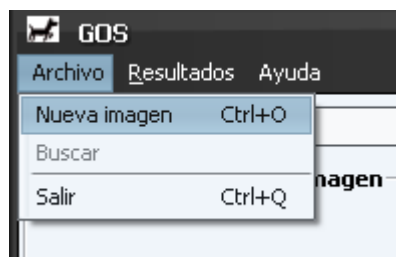


Figura 70. Menú Archivo en la barra de menú del GOS.

2. Menú Resultados: las opciones del menú resultado aparecen desactivadas al iniciar la aplicación, y se encuentran disponibles después de realizar una búsqueda.

- *Guardar seleccionados:* guarda los resultados seleccionados en la parrilla de imágenes en un archivo XML con formato MPEG-7 (véase en 5.4.3).
- *Limpiar:* borra los resultados y la imagen consulta para iniciar nuevas búsquedas desde cero.

3. Menú Ayuda, con información de soporte:

- *Ayuda de GOS:* documentación de ayuda al usuario.
- *Acerca de GOS:* información sobre la aplicación.

Algunos ítems de menú llevan asociados unos comandos de teclado para ejecutar la acción a través de ciertas teclas. Por ejemplo, la ayuda se ejecuta al pulsar la tecla F1.

5.3.2.3 Barra de herramientas

La barra de herramientas de la interfaz se encuentra en la parte superior de la pantalla, justo debajo del menú principal. Contiene toda una serie de elementos gráficos básicos en la aplicación, distribuidos de forma horizontal y agrupados según sus funciones. Un separador visual nos diferencia los componentes relacionados con la consulta, colocados a la izquierda de la barra, y los componentes relacionados con



los resultados, situados en la parte derecha. En último lugar se incluye un botón de ayuda.



Figura 71. Barra de herramientas del GOS.




▪ ELEMENTOS DE LA CONSULTA

En esta sección de la barra de herramientas encontramos:

-  Botón para cargar una imagen de consulta, abre archivos en formato BMP/GIF/JPG/JPEG/PNG (acción disponible también a través del menú principal).
- Campo de texto con la URL de la imagen de consulta.
-  Botón de buscar (con el logo del GOS) para lanzar una búsqueda (acción disponible en el menú principal).
- Campo de texto con el número de resultados a obtener.

▪ ELEMENTOS DE LOS RESULTADOS

Después del separador visual encontramos los siguientes elementos:

-  Botón de selección rectangular para seleccionar resultados en grupo.
-  Botón de guardar para crear un archivo XML con los resultados seleccionados y guardarlo en disco búsqueda (acción disponible en el menú principal).
-  Botón de Limpiar para borrar las imágenes visualizadas en la interfaz, tanto en el área de resultados como en la consulta búsqueda (acción disponible en el menú principal).

▪ AYUDA



Botón de ayuda para abrir la documentación de ayuda al usuario búsqueda (acción disponible en el menú principal).

5.3.2.4 Área de consulta

La área de consulta se encuentra ubicada en el lateral izquierdo de la aplicación. Un panel contenedor alberga en su interior tres paneles distribuidos de forma vertical, correspondientes a los pasos lógicos que el usuario debe realizar a la hora de confeccionar una consulta:



1. Cargar la imagen de consulta

2. Configurar el algoritmo de búsqueda:

- a. Seleccionando el criterio de fusión
- b. Seleccionando los descriptores a utilizar en la búsqueda.
- c. Si el criterio de fusión es "Pesos", determinando el peso de cada descriptor en la búsqueda.

3. Configurar el ámbito de búsqueda:

- a. Seleccionado el modo de ejecución del motor de búsqueda.
- b. Seleccionando la categoría dónde buscar.

Figura 72. Área de consulta del GOS

Cada uno de estos tres paneles se presenta enmarcado en un fino borde decorativo que permite diferenciar las tres fases de formulación de la consulta. En el marco, se incluye el título del panel, destacado en negrita, para informar y guiar al usuario sobre el contenido mostrado en cada panel.

El GOS implementa estos paneles específicos en clases propias:

- **IMAGEPANEL: PANEL DE IMAGEN DE CONSULTA**

Este panel muestra la imagen seleccionada por el usuario como ejemplo en la consulta. La imagen se visualiza a tamaño reducido de 250x200 píxeles, guardando la relación de aspecto 1.25:1 de la imagen original (de 720x576 píxeles). Al cargar la imagen, en el título del panel aparece el nombre del archivo seleccionado.

- **WEIGHTSPANEL: PANEL DE CRITERIOS DE FUSIÓN**

El panel de criterio de fusión está compuesto por un conjunto de cuatro botones en la parte superior del panel, correspondientes a los cuatro criterios de fusión que soporta el sistema de búsqueda. Por defecto, el método de fusión seleccionado es el “Mínimo” (Min), pero el usuario puede cambiar el criterio simplemente clicando sobre cualquiera de los cuatro botones. La opción seleccionada se destaca sobre el resto cambiando el aspecto del botón, aplicando un fondo azul, poniendo el texto en negrita y acentuando el icono de la huella situada al lado del texto. Así, el usuario puede ver a simple vista qué opción está utilizando en su consulta.

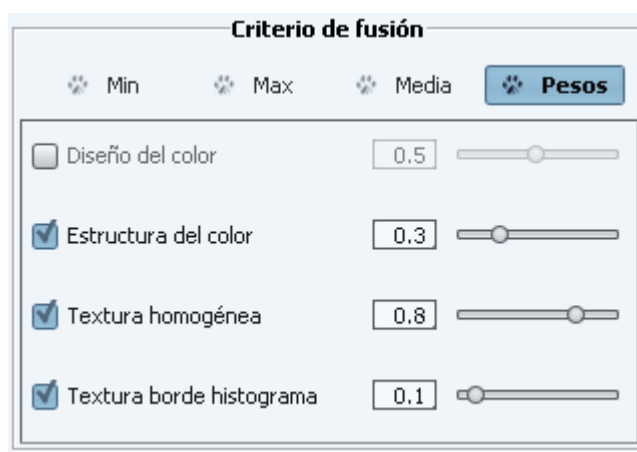


Figura 73. Panel de criterio de fusión del GOS.

Bajo los botones de los criterios de fusión, se encuentra un panel contenedor con la colección de descriptores visuales disponibles en el sistema de búsqueda. Estos descriptores son configurables, ya que el usuario puede seleccionarlos, ponderarlos y combinarlos como desee. Se gestionan a través de una lista de objetos *Weight* propios de la interfaz.

Cada objeto *Weight* dispone de una casilla de selección (*checkbox*) para seleccionar o deshabilitar el descriptor en la búsqueda. Con la utilización de este tipo de componente optimizamos el espacio dedicado a los descriptores, ya que permite realizar cualquier combinación (varios checkboxes seleccionados al mismo tiempo) sin necesidad de mostrar un ítem por combinación posible (cosa que incrementaría enormemente el tamaño de este panel).

Si el criterio de fusión seleccionado es "Pesos", el usuario tiene la opción de otorgar a cada descriptor un factor de ponderación a través de una barra deslizante con rango de 0.0 a 1.0. El uso de este componente garantiza que el usuario no introduce valores no aceptados por el sistema. Junto a esta barra deslizante, encontramos un cuadro de texto para visualizar en todo momento el peso otorgado al descriptor. Por defecto el peso de cada descriptor se establece en 0.5.

▪ **SEARCHSPACEPANEL: PANEL DE ESPACIO DE BÚSQUEDA**

El panel de espacio de búsqueda consta de dos partes: la selección del modo de ejecución del sistema de búsqueda y la selección del ámbito de búsqueda a través de las categorías disponibles.



Figura 74. Panel de espacio de búsqueda del GOS.

En la parte superior del panel, se sitúan dos botones de opción única, ya que el usuario debe escoger una de las dos posibilidades (son excluyentes). Por defecto, el modo de ejecución es “Local”.

Bajos los botones de modo de ejecución, se muestran las categorías de búsqueda en forma de árbol. Actualmente se encuentran disponibles dos espacios de búsqueda, *CCMA* y *Mediapro*. Cada espacio de búsqueda corresponde al corpus de imágenes facilitadas por estas empresas y posee sus propias subcategorías (clasificación de las imágenes en la base de datos). Por ahora, aplicación permite la selección de un único ítem. Por defecto, si el usuario no selecciona ninguna opción, el ámbito de búsqueda es *CCMA*.

5.3.2.5 Área de resultados

La área de resultados ocupa el resto de pantalla disponible en la interfaz. Se ha implementado una clase propia, *ResultsPanel*, para albergar todos los componentes necesarios para visualizar los resultados. Este panel también se presenta enmarcado con un borde decorativo que incluye su título.

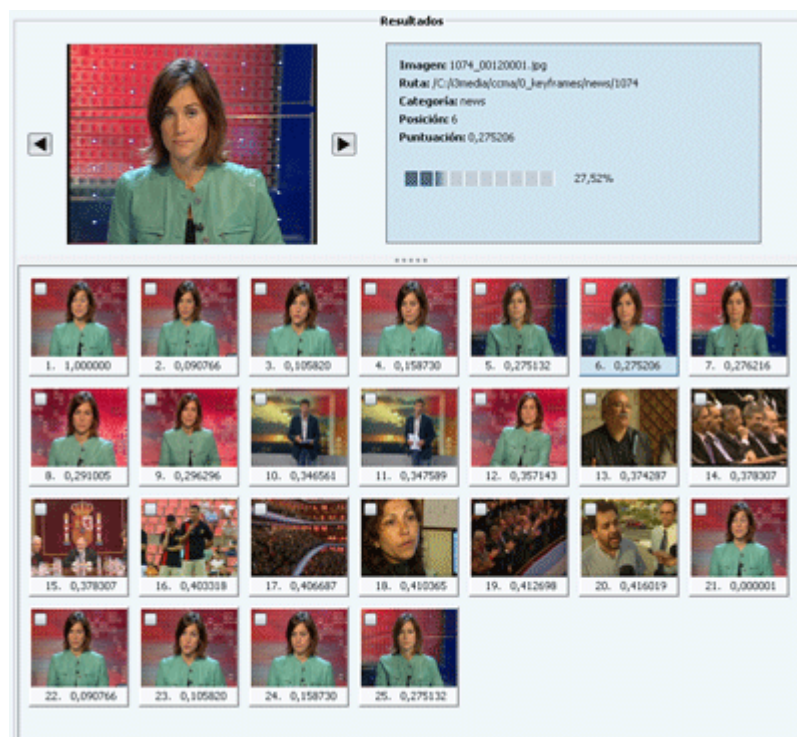


Figura 75. Panel de resultados del GOS.

La zona de resultados se estructura en dos áreas diferenciadas, por un lado la parte superior del panel presenta una imagen resultado destacada junto a su información asociada y por otro, en la parte inferior se visualizan todos los resultados obtenidos en la búsqueda ordenados por grado de similitud con la consulta (orden decreciente de mayor a menor similitud).

▪ ***FOCUSIMAGEPANEL: PANEL DE LA IMAGEN DESTACADA (CON FOCO)***

Este panel se sitúa en la parte superior de la zona de resultados, en un lugar privilegiado, que permite destacar la información que contiene. Consta de los siguientes elementos:

- ***Imagen destacada (con foco)***: se muestra la imagen en un formato mayor al resto de resultados, con un tamaño de 250x200 píxeles. Su colocación en la pantalla, a la misma altura que la imagen consulta, así como su mismo tamaño, permite al usuario comparar fácilmente ambas imágenes, el ejemplo con el resultado obtenido.
- ***Botones de navegación***: a los lados de la imagen destacada se encuentran dos botones de navegación, representados con las flechas genéricas de anterior y siguiente, que permiten desplazarse a través de los resultados con un simple clic.
- ***Área de texto informativa***: junto a la imagen, se presenta un cuadro de texto con una serie de información asociada que incluye el nombre del archivo imagen, su ruta, su categoría, su posición en la lista de resultados ordenados y la puntuación obtenida en la consulta.

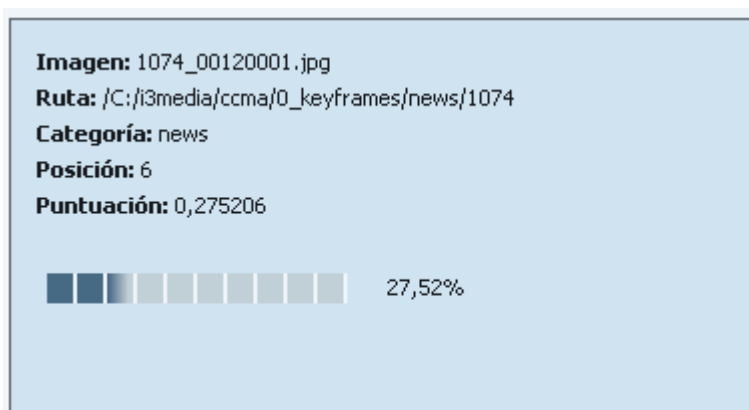


Figura 76. Área de texto informativa de la imagen destacada del GOS.

Como refuerzo visual, esta puntuación se muestra de forma gráfica a través de una barra de progresión junto a una cadena de texto con la puntuación en porcentaje..

▪ **GRIDTHUMBNAILSPANEL: PANEL DE PARRILLA DE RESULTADOS**

Todos los resultados obtenidos en la búsqueda se visualizan en este panel, que utiliza el administrador de diseño *GridLayout*, para mostrar la lista ordenada de imágenes en forma de parrilla. Cada resultado adopta la apariencia de una pequeña diapositiva, que se genera a través de la clase propia *ThumbnailPanel*.

Un *ThumbnailPanel* se compone de tres elementos:

- **Imagen miniatura (*thumbnail*⁴³):** imagen en formato reducido, con tamaño de 100x80 píxeles.
- **Checkbox:** casilla de verificación en el margen superior izquierdo para seleccionar la imagen.
- **Texto informativo:** indica la posición de la imagen en la lista de resultados y la puntuación obtenida en la consulta.

Estas diapositivas se visualizan de forma diferente en función de su estado (fig. 77). Si la diapositiva tiene el foco, es la imagen destacada en el panel superior de la zona de resultados, y se indica visualmente aplicando un fondo azul y un leve borde azul oscuro al panel de la diapositiva. Para dar foco a una imagen, basta con hacer clic sobre su diapositiva. Si la imagen ha sido seleccionada por el usuario clicando sobre la casilla de selección, se indica a través del *checkbox*, que aparece seleccionado, y de un borde más grueso en un color azul intenso. Por el contrario, si la imagen no es objeto de atención del usuario, se visualiza con un fondo blanco y un leve borde gris. De esta manera tan visual, el usuario puede manipular los resultados a su antojo.

⁴³ Los *thumbnails* son versiones reducidas de imágenes, usadas para ayudar a su organización y reconocimiento, especialmente en motores de búsqueda visuales y los programas destinados a organizar imágenes. (<http://es.wikipedia.org/wiki/Thumbnail>)



Figura 77. Diapositiva con foco, seleccionada y sin foco respectivamente.

5.4 Funcionalidades de la interfaz

El GOS es un buscador de imágenes QbE, y como tal, permite formular una consulta, ejecuta la búsqueda y visualiza los resultados. Estas son las funciones básicas que todo buscador debe desarrollar. Pero nuestra interfaz ofrece otras funcionalidades que le otorgan un valor añadido a la herramienta, con el objetivo de facilitar el trabajo de búsqueda y recuperación de imágenes al usuario. Veamos cómo se han implementado estas funcionalidades.

5.4.1 Nueva búsqueda

La aplicación dispone de un método para refinar las búsquedas, ya que cualquier imagen resultado puede ser utilizada como punto de partida para una nueva consulta. Este método es utilizado por la mayoría de interfaces de sistemas CBIR como técnica de *Relevance Feedback*.

La interfaz del GOS permite lanzar una nueva búsqueda a partir de un resultado haciendo doble clic directamente sobre la imagen resultado que se desee configurar como nueva imagen ejemplo.

5.4.2 Navegación por los resultados

La parrilla de imágenes miniatura nos muestra todos los resultados obtenidos en la búsqueda, pero la información que nos ofrece de cada resultado es muy reducida. Para obtener más información sobre una imagen resultado, el usuario puede darle foco haciendo clic sobre la diapositiva. Cuando una diapositiva obtiene el foco, su fondo cambia de blanco a azul y su borde de gris a un azul más oscuro, y

automáticamente pasa a ocupar el lugar de la imagen destacada del panel superior. Esta forma de visualizar una imagen con foco y su información asociada es muy sencilla, pero algo limitada si el usuario quiere ver con detalle y de forma rápida todo el conjunto de resultados, o analizar un grupo de imágenes determinado.



Figura 78. Botones de navegación al lado de la imagen con foco.

La interfaz dispone de dos métodos alternativos para navegar por los resultados y visualizarlos como imágenes destacadas. Por un lado, encontramos dos típicos botones de navegación a la izquierda y derecha de la imagen destacada, para recorrer los resultados (anterior y posterior respectivamente) con un solo clic, y sin necesidad de desplazarnos a la zona de la parrilla para clicar sobre la imagen deseada. Por otro, a través de la rueda central del ratón, el usuario puede desplazarse por los resultados de forma rápida y sin necesidad de ir realizando clics.

5.4.3 Selección de resultados

El GOS permite al usuario seleccionar aquellos resultados que sean de su interés y guardarlos en un archivo XML. Esta selección se puede realizar de dos formas:

1. **Selección individual:** haciendo clic sobre el checkbox de la diapositiva.
2. **Selección de un grupo:** a través de la herramienta de selección rectangular, el usuario puede seleccionar de una sola vez un grupo de resultado.

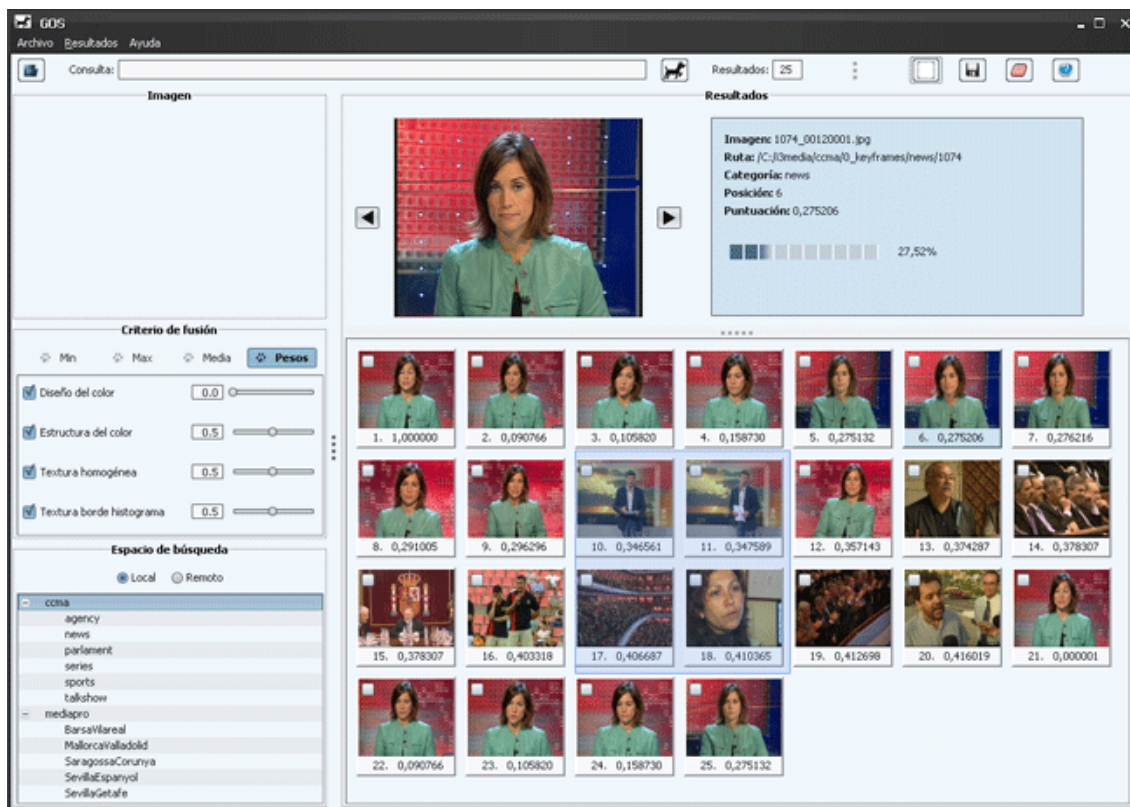


Figura 79. Selección de un grupo de resultados.

Para más información sobre las funcionalidades, véase el anexo I (manual de usuario).

6. Futuro del GOS

El diseño de una buena interfaz gráfica de usuario es una tarea compleja. Este proyecto ha abordado los primeros pasos del desarrollo de una herramienta pensada para facilitar la utilización del sistema de búsqueda de imágenes mediante imagen ejemplo implementado en el programa B_RANKER. Pero esto solo es el principio. La aplicación que se presenta en esta memoria es una primera versión de la interfaz, que cubre las funcionalidades básicas de este tipo de buscadores, pero está abierta a mejoras, a nuevas ideas y a la ampliación de sus funcionalidades, siguiendo el proceso de evolución de las técnicas de recuperación de contenido audiovisual.

El siguiente paso que se plantea el GOS es la ampliación de los métodos de búsqueda que soporta la interfaz. Esta ampliación deberá realizarse en dos fases: la primera, y más inmediata, es la implementación de la búsqueda de imágenes basada en regiones, y la segunda, dar la posibilidad de utilizar y combinar más de un tipo de consulta (basada en texto, imágenes y regiones).

Para mejorar la aplicación también es necesario eliminar sus restricciones de uso, condicionadas por la inclusión de algunos elementos de configuración *hard-coded* como son el número de resultados a devolver, la definición de los espacios de búsqueda y la estructura de ficheros. En un futuro, el usuario debe poder establecer el número de resultados que desea obtener y debe poder escoger el directorio de trabajo de la aplicación.

La herramienta también será más útil y dinámica si ofrece la posibilidad de generar automáticamente los descriptores visuales de cualquier imagen que introduzca el usuario. Actualmente, las consultas se realizan sobre imágenes cuyos descriptores visuales han sido calculados previamente.

Otra limitación del GOS viene dada por la única versión existente del ejecutable del B_RANKER, sólo disponible para sistemas operativos GNU/Linux. Esto imposibilita la ejecución de la aplicación en local bajo Windows, el sistema operativo más utilizado (especialmente en entornos profesionales). Cuando esté disponible el binario para Windows, el GOS habrá de ser capaz de ejecutarse en ambos entornos.

Con el tiempo, también irá surgiendo la necesidad de ir ampliando las funcionalidades de la interfaz, creando nuevas herramientas para la visualización y gestión de resultados. Por ejemplo, sería interesante mejorar la selección en grupo de resultados, ofreciendo la posibilidad al usuario de combinar varias herramientas, la actual selección rectangular y habilitar la selección a través de la tecla CONTROL (para añadir resultados a una selección previa), o permitir la selección a partir de la tecla SHIFT y las flechas de navegación del teclado.

Hemos visto, que en este tipo de proyectos no hay que descuidar el punto de vista del usuario, por lo que también será necesario realizar un test de usabilidad con usuarios finales (en esta primera fase de diseño no ha sido posible realizarlo por falta de tiempo). Es importante recoger las impresiones del usuario que trabajará con la herramienta, que aportará sugerencias y comentarios de gran valor para mejorar la aplicación.

7. Conclusiones

El proyecto del diseño e implementación de la interfaz gráfica de usuario (GUI) del GOS ha requerido un proceso de desarrollo elaborado. Desde el principio nos planteamos el reto de crear una herramienta completa, práctica y eficiente desde el punto de vista técnico, atractiva y moderna el punto de vista estético, sencilla y usable desde el punto de vista funcional.

¿Cómo lo hacemos? El primer paso es decidir qué tipo de interfaz necesita el sistema de búsqueda (B_RANKER). La decisión fácil es optar por una interfaz *web-like* e implementarla con tecnología HTML, PHP o JSP (*Java Server Pages*, una tecnología Java que permite generar contenido dinámico para web, generalmente en forma de documentos HTML). Podemos crear un buscador con un simple formulario y mostrar los resultados en una tabla o lista. Pero el GOS, quiere ir más allá, está destinado a un uso profesional y es el punto de partida para la conseguir un buscador de imágenes basado en imágenes ejemplo ahora, y en regiones en un futuro inmediato. Desarrollar la aplicación en Java permite crear una herramienta más versátil, con más posibilidades de evolución e integración con el resto de aplicaciones y herramientas, del GPI y del proyecto i3media.

Programar en Java no es tarea fácil, especialmente una interfaz gráfica con Swing. Aunque se conozca la dinámica de Java, a base de clases y de una programación orientada a objetos, la metodología a seguir para desarrollar un proyecto con parte gráfica, enteramente en Java, requiere algo de experiencia. La mayor dificultad es aunar funcionalidad y estética, dotar a los elementos gráficos de una “personalidad” propia, que sea característica de la aplicación, pero sin perder de vista las necesidades reales del usuario, y paralelamente ir sorteando las limitaciones de la tecnología. Y esta dificultad es a la vez una de las partes más interesantes del proyecto, ya que requiere de capacidad de resolución e ingenio.

El resultado final es un pequeño triunfo. La interfaz es fácil de utilizar, bien estructurada, pensada para simplificar el trabajo de recuperación de imágenes al usuario, y a la vez atractiva, con una imagen corporativa propia. Como valor añadido, la aplicación permite su ejecución en entorno remoto y local, aunque el objetivo final es acabar ofreciendo el servicio de búsqueda de manera remota, siguiendo la tendencia

de la gran mayoría de proveedores de contenido media. A partir de servicios web (como los que ofrecerá *upseek*⁴⁴), cualquier usuario en red puede tener acceso a múltiples aplicaciones, de forma cómoda y sencilla. El enorme volumen de información disponible, especialmente en la industria audiovisual, precisa de herramientas que se adapten a este tipo de arquitecturas, modulares e integrables.

Desde un punto de vista más personal, la realización del PFC debe cubrir tres objetivos básicos: motivación personal, aportación valiosa y perspectivas de futuro.

Primero, es indispensable trabajar en un tema que despierte el interés del proyectista, que le planteen nuevos retos y a la vez fomenten su capacidad de aprendizaje. Este proyecto me ha brindado la oportunidad de poder explotar mi capacidad creativa y de análisis, de aplicar conocimientos en comunicación audiovisual (adquiridos en mi anterior titulación), y a la vez descubrir nuevas áreas de desarrollo e investigación en el sector audiovisual de ámbito profesional. El diseño e implementación de una GUI implica la inmersión del desarrollador en diversas disciplinas, desde el diseño gráfico y la programación hasta la usabilidad y la HCI. El abanico de áreas tratadas ha sido muy enriquecedor.

En segundo lugar, es importante dedicar esfuerzos a proyectos que aporten un granito de arena al mundo real, que no se limiten en el marco académico. La iniciativa *i3media*, a la que pertenece el proyecto, aglutina universidades y empresas muy relevantes del sector audiovisual. Participar en iniciativas de esta envergadura es una oportunidad única para conocer las líneas de futuro del sector, sus actividades en innovación y desarrollo (I+D), y sus necesidades en materia de nuevos profesionales. El GOS es una herramienta esperada por empresas como *Mediapro* y la *CCMA*, y será utilizada para realizar tareas de documentación y recuperación de información, tan importantes en empresas de gestión de contenidos y servicios media.

Por último, el siguiente paso es la inmersión en el mercado laboral. Un proyecto permite desarrollar aptitudes de cara al mundo profesional (investigación, planificación, metodología de trabajo, etc.) y puede ser una buena toma de contacto con empresas y abrir puertas. El proyecto *i3media* abarca una gran diversidad de sectores (fig.80), siendo un buen ejemplo del futuro que un profesional formado en materia audiovisual y

⁴⁴ Véase capítulo 4. Entorno de Desarrollo

telecomunicaciones tiene por delante. Un perfil multidisciplinar es una buena baza para abrirse camino, esperemos que con éxito.



Figura 80. El proyecto i3media en el contexto de las empresas miembros del consorcio y los sectores industriales implicados

Los objetivos están cumplidos.

8. Bibliografía

- [1] Web i3media: <http://www.i3media.org>
- [2] Royo, Javier. Diseño Digital. Ediciones Paidós Ibérica, S. A. Barcelona, 2006. ISBN: 84-493-1550-6
- [3] i3media: Tecnologías para la creación y gestión automatizada de contenidos audiovisuales inteligentes. Memoria Técnica. Programa CENIT 2006.
- [4] Sommerville, Ian. Ingeniería de software. Cap. 15: *Diseño de la interfaz de usuario*. Pearson Educación de México, S. A. México, 2002. ISBN: 970-26-0206-8
- [5] Marcos, Mari-Carmen. "HCI (Human computer interaction): concepto y desarrollo". En: El profesional de la información, 2001, junio, v. 10, n. 6, pp. 4-16. (<http://www.elprofesionalde lainformacion.com/contenidos/2001/junio/1.pdf>)
- [6] Pecenovic, Z. Integrating visual and semantic descriptions for effective, flexible and user-friendly image retrieval, *Ph.D. Thesis*, 2002. (<http://library.epfl.ch/theses/?nr=2679>)
- [7] Roe, Benjamin. Diseño de interfaces de usuario usables. Publicado en Mundo Geek. Diciembre 2004 (<http://mundogeek.net/traducciones/interfaces-usuario-usables/gui.html>)
- [8] Pérez Álvarez, Sara. Análisis de usabilidad de sistemas CBIR | User friendliness of CBIR systems analysis -. Abril 2008 (<http://biblioteca.universia.net/ficha.do?id=34103694>)
- [9] Colin C. Venters and Dr. Matthew Cooper, A Review of Content-Based Image Retrieval Systems. Manchester Visualization Centre and the Institute for Image Data Research, Manchester Computing, University of Manchester. Junio 2000.

- [10] Colin C. Venters. User Interface Design & Evaluation for a Content-Based Image Retrieval System. Department of Information & Library Management, University of Northumbria at Newcastle, England. (<http://www.bcs-hci.org.uk/hci1998/C51/>)
- [11] Datta, Ritendra, Joshi Dhiraj, Li, Jia and Wang, James Z. Image Retrieval: Ideas, Influences and Trends of New Age. April, 2008. The Pennsylvanic State University (<http://infolab.stanford.edu/~wangz/project/imsearch/review/JOUR/datta.pdf>)
- [12] Pentland, R.W.; Picard, S.; Sclaroff, S. Photobook: Content-Based Manipulation of Image Databases. The Media Laboratory, Massachusetts Institute of Technology. June 1995
- [13] Nakazato, Munehiro and Manola, Ljubomir and Huang, Thomas S. (2003) ImageGrouper: a group-oriented user interface for content-based image retrieval and digital image arrangement. Journal of Visual Languages & Computing, 14 (4). pp. 363-386. ISSN 1045926X
- [14] PFC: Luna Baeza, Mireia. Creación de un sistema WEB para la indexación, búsqueda y adquisición de videos mediante descriptores MPEG-7. EUETIT-UPC, Junio-2006. (<http://upcommons.upc.edu/pfc/handle/2099.1/3865>)
- [15] PFC: Delcor Ballesteros, Jordi; Pérez Noriega, Verónica. Descripción, indexación, búsqueda y adquisición de secuencias de vídeo mediante descriptores MPEG-7". EUETIT-UPC, Julio-2006. (<http://upcommons.upc.edu/pfc/handle/2099.1/3855>)
- [16] <http://www.like.com/aboutus.py>
- [17] Eckel, Bruce. Piensa en Java. Pearson Educación S.A. Madrid, 2007. ISBN: 978-84-8966-034-2
- [18] Proyecto Substance de Java.net: <https://substance.dev.java.net/>
- [19] La aportación del eyetracking en el sector de la usabilidad . Febrero 2007. <http://www.usolab.com/articulos/eyetracking-usabilidad-comunicacion.php>
- [20] Eye-Tracking en Interacción Persona-Ordenador (HCI) <http://www.nosolousabilidad.com/articulos/eye-tracking.htm>
- [21] El patrón F. Septiembre 2008. <http://leerenpantalla.wordpress.com/2008/09/19/el-patron-f/>

ANEXO I. MANUAL DE USUARIO

GOS – Graphic Object Searcher

MANUAL DE USUARIO



Versión Beta

Página web: <http://gps-tsc.upc.es/imatge/i3media/gos/>

- ✿ Silvia Cortés Yuste
- ✿ Xavier Giró i Nieto

Grupo de Procesado de Imagen (GPI) – Universitat Politècnica de Catalunya (UPC)



CONTENIDOS

1	<i>Introducción</i>	132
1.1	Recursos online	132
1.2	Requisitos	133
1.2.1.	Motor de búsqueda	133
1.2.2.	Estructura de directorios	133
1.2.3.	JRE de Java	134
2.	<i>Interfaz de usuario</i>	135
2.1.	Qué puede hacer el GOS	135
2.2.	Áreas de trabajo	136
2.2.1.	Menú principal	136
2.2.2.	Barra de herramientas	138
2.2.3.	Área de consulta	139
2.2.3.1.	<i>Panel de imagen de consulta</i>	140
2.2.3.2.	<i>Panel de criterios de fusión</i>	141
2.2.3.3.	<i>Panel de espacio de búsqueda</i>	143
2.2.4.	Área de resultados	144
2.2.4.1.	<i>Imagen destacada (con foco)</i>	144
2.2.4.2.	<i>Parrilla de resultados</i>	146
3.	<i>Cómo utilizar el GOS</i>	148
3.1.	Cómo buscar	148
Paso 1.	Cargar la imagen de consulta	148
Paso 2.	Configurar el motor de búsqueda	149
Paso 3.	Configurar el ámbito de búsqueda	151
Paso 4.	Ejecutar la búsqueda	152
3.2	Cómo navegar por los resultados	152
3.2.1	Convertir una imagen en imagen destacada	153
3.2.2	Desplazarse con los botones de navegación	154
3.2.3	Desplazarse con la rueda central del ratón	154



3.3	Cómo seleccionar y guardar resultados	154
3.3.1.	Selección individual	154
3.3.2.	Selección en grupo	155
3.3.3.	Selección combinada.....	156
3.3.4.	Deseleccionar resultados.....	157
3.3.5.	Guardar resultados	157
3.4	Cómo afinar la búsqueda	158
3.5	Cómo limpiar la pantalla	158
3.6	Cómo consultar la ayuda	159



1. Introducción

Este manual describe cómo utilizar el buscador de imágenes mediante imagen ejemplo GOS (*Graphic Object Searcher*). El objetivo de este documento es enseñar al usuario a trabajar con la aplicación, comprendiendo sus funcionalidades y experimentando con todas sus posibilidades. Después de su lectura, el usuario será capaz de realizar búsquedas de imágenes alojadas en grandes bases de datos a partir de otra imagen y de unas determinadas características visuales, de una forma rápida y eficiente.

La herramienta es un sistema de búsqueda para recuperar imágenes basándose en su contenido y permite realizar consultas a través de una imagen ejemplo seleccionada de un directorio local y configurar los parámetros de búsqueda para obtener unos resultados satisfactorios. El usuario puede elegir el espacio de búsqueda, los descriptores visuales que desea aplicar en la búsqueda y el criterio de fusión de éstos.

1.1 Recursos online

- **Página web del GOS:** http://gps-tsc.upc.es/imatge/i3media/gos/index_es.html
- **Vídeo Demo:** [Demostración de consultas basadas en imágenes y diversos descriptores](#)
(Disponible en: <http://gps-tsc.upc.es/imatge/i3media/gos/videos/gos.avi>)



1.2 Requisitos

1.2.1. Motor de búsqueda

Para su funcionamiento, el GOS necesita el motor de búsqueda B_RANKER, programa incluido en la plataforma *SoftImage*. La versión actual de su ejecutable sólo está disponible para sistemas operativos GNU/Linux. B_RANKER debe encontrarse en la ubicación:

- `$HOME/SoftImage/bin/release/`

donde la variable `$HOME` corresponde al directorio de trabajo del usuario.

1.2.2. Estructura de directorios

La ubicación de los ficheros del espacio de búsqueda y el directorio de resultados deben ser las siguientes:

- **Para el directorio de los ficheros del espacio de búsqueda:**
`$HOME/2_semantic/queryByExample/1_image/2_databases`
- **Para el directorio de resultados:**
`$HOME/2_semantic/queryByExample/1_image/3_results`

donde la variable `$HOME` corresponde al directorio de trabajo del usuario.

En el directorio del espacio de búsqueda se encuentran los ficheros de texto (.txt) donde se listan las direcciones de los archivos descriptores (.xml) de las imágenes pertenecientes a cada espacio de búsqueda (bases de datos). Es requisito indispensable que estos ficheros se hayan generado antes de realizar una búsqueda.



1.2.3. JRE de Java

GOS es una aplicación implementada en Java y su ejecución solo está condicionada a la instalación del Java en el sistema operativo, que podrá ser MacOSX, Microsoft Windows o GNU/Linux. Se requiere de Java 6.0 o superior para funcionar.

NOTA: Actualmente el GOS no permite ser ejecutado en Windows, ya que la estructura de directorios utilizada está codificada sólo para formato GNU/LINUX.



2. Interfaz de usuario

2.1 Qué puede hacer el GOS

La interfaz de usuario de GOS permite realizar todas las funcionalidades básicas de un buscador de imágenes mediante ejemplo:

- Seleccionar una imagen de consulta.
- Seleccionar el criterio de fusión para la búsqueda.
- Seleccionar los descriptores visuales a utilizar en la búsqueda.
- Configurar el peso otorgado a cada descriptor visual (si el criterio de fusión lo permite).
- Seleccionar el modo de ejecución (local o remoto).
- Seleccionar el espacio de búsqueda.
- Lanzar la consulta.
- Visualizar los resultados.
- Navegar por las imágenes resultado.
- Seleccionar y guardar imágenes resultado.
- Limpiar la pantalla para crear nuevas consultas.
- Afinar la búsqueda a partir de nuevas consultas con una imagen resultado.

Las acciones que realiza el GOS son accesibles a través del menú y/o la barra de herramientas. Algunas de estas acciones también están disponibles a través de combinación de teclas.

INFO: La combinación de teclas disponible para las acciones de menú se muestra junto a la opción correspondiente cuando se despliega el menú.



2.2 Áreas de trabajo

La ventana principal de GOS se compone de 3 áreas:

- **Menú principal y barra de herramientas.**
- **Área de consulta:** destinada a la formulación de la consulta.
- **Área de resultados:** destinada a la visualización y gestión de las imágenes resultado.

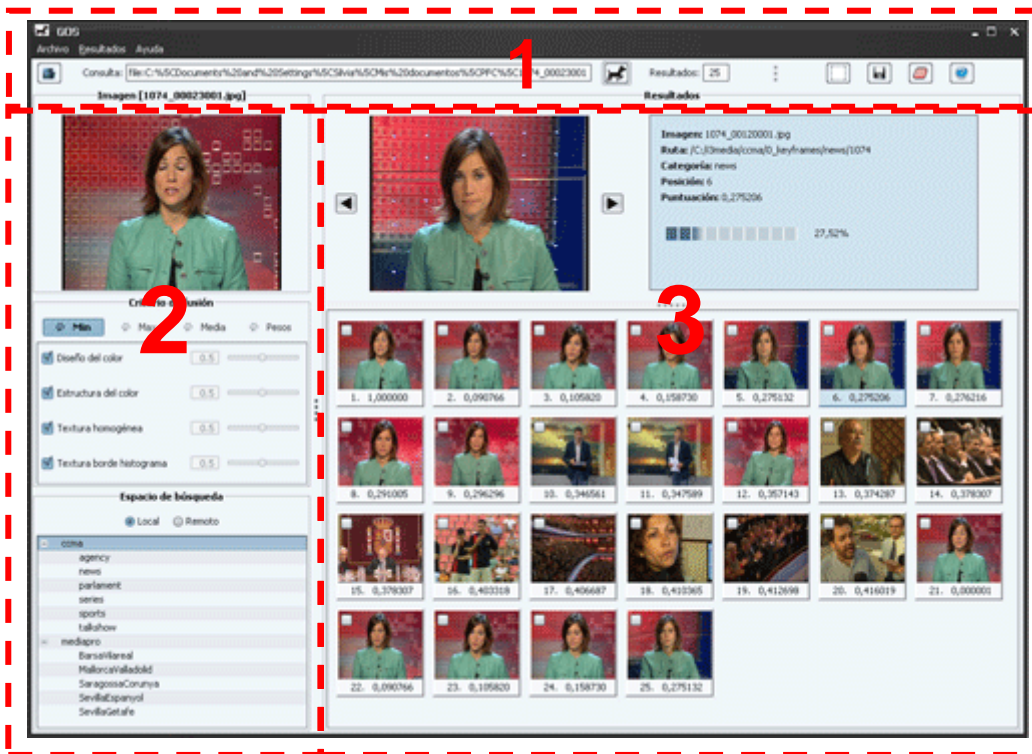


Figura 81. Interfaz del GOS

2.2.1 Menú principal

El GOS ofrece tres menús: Archivo, Resultados y Ayuda.

1. **Menú Archivo**, que contiene las acciones:
 - **Nueva Imagen:** cargar una imagen de consulta.
 - **Buscar:** lanzar una consulta.
 - **Salir:** cerrar la aplicación.

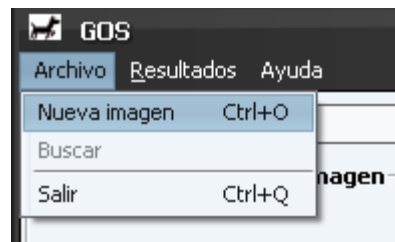


Figura 82. Menú Archivo en la barra de menú del GOS.

2. **Menú Resultados:** las opciones del menú resultado aparecen desactivadas al iniciar la aplicación, y se encuentran disponibles después de realizar una búsqueda.

- **Guardar seleccionados:** guardar los resultados seleccionados en la parrilla de imágenes en un archivo XML con formato MPEG-7.
- **Limpiar:** borra los resultados y la imagen consulta para iniciar nuevas búsquedas desde cero.

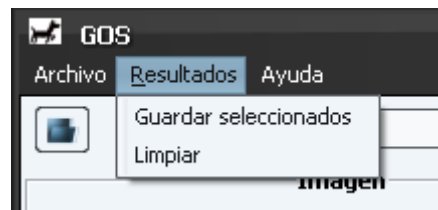


Figura 83. Menú Resultados en la barra de menú del GOS.

3. **Menú Ayuda,** con información de soporte:

- **Ayuda de GOS:** documentación de ayuda al usuario.
- **Acerca de GOS:** información sobre la aplicación.

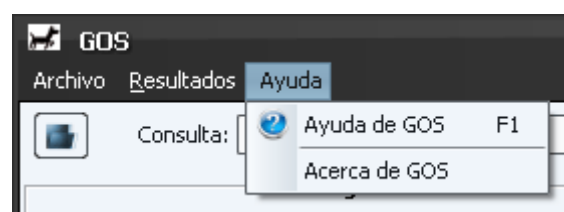


Figura 84. Menú Ayuda en la barra de menú del GOS.



2.2.2 Barra de herramientas

En la parte superior, justo debajo del menú principal, se encuentra la barra de herramientas. Contiene los comandos básicos para trabajar con la aplicación, agrupados por áreas: elementos relacionados con la consulta, elementos relacionados con los resultados y un botón de ayuda.

Elementos relacionados con la consulta



Figura 85. Barra de herramientas del GOS.

▪ ELEMENTOS DE LA CONSULTA



Nueva Imagen

Botón para cargar una imagen de consulta, abre archivos en formato BMP/GIF/JPG/JPEG/PNG (acción disponible también a través del menú principal y el comando Ctrl+O).



Consulta

Campo de texto con la URL de la imagen de consulta.



Buscar

Botón de buscar (con el logo del GOS) para lanzar una búsqueda (acción disponible en el menú principal).



Resultados

Campo de texto con el número de resultados a obtener.



- **ELEMENTOS DE LOS RESULTADOS**



Selección rectangular

Botón de selección rectangular para seleccionar resultados en grupo.



Guardar

Botón de guardar para crear un archivo XML con los resultados seleccionados y guardarlo en disco búsqueda (acción disponible en el menú principal).



Limpiar

Botón de Limpiar para borrar las imágenes visualizadas en la interfaz, tanto en el área de resultados como en la consulta búsqueda (acción disponible en el menú principal).

- **AYUDA**

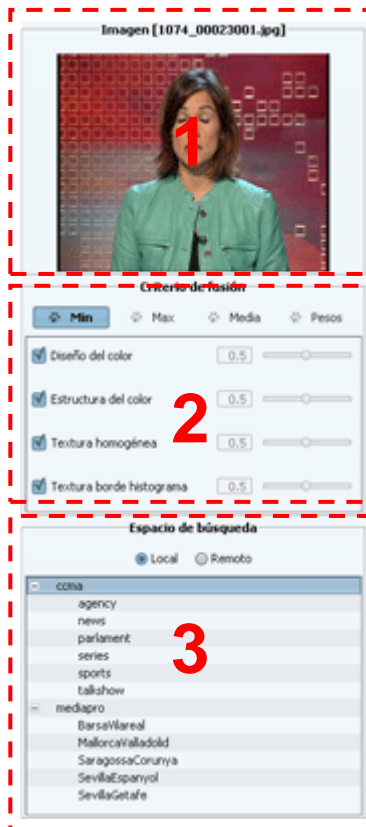


Ayuda

Botón de ayuda para abrir la documentación de ayuda al usuario búsqueda (acción disponible en el menú principal).

2.2.3 Área de consulta

El área de consulta se encuentra ubicada en el lateral izquierdo de la aplicación. Consta de 3 paneles para formular la petición de búsqueda:



1. Panel de imagen de consulta

2. Panel de criterios de fusión

3. Panel de espacio de búsqueda

Figura 86. Área de consulta.

2.2.3.1 *Panel de imagen de consulta*

Este panel muestra la imagen seleccionada por el usuario como ejemplo en la consulta. La imagen se visualiza a un tamaño reducido de 250x200 píxeles, guardando la relación de aspecto 1.25:1 de la imagen original (de 720x576 píxeles). Al cargar la imagen, en el título del panel aparece el nombre del archivo seleccionado.

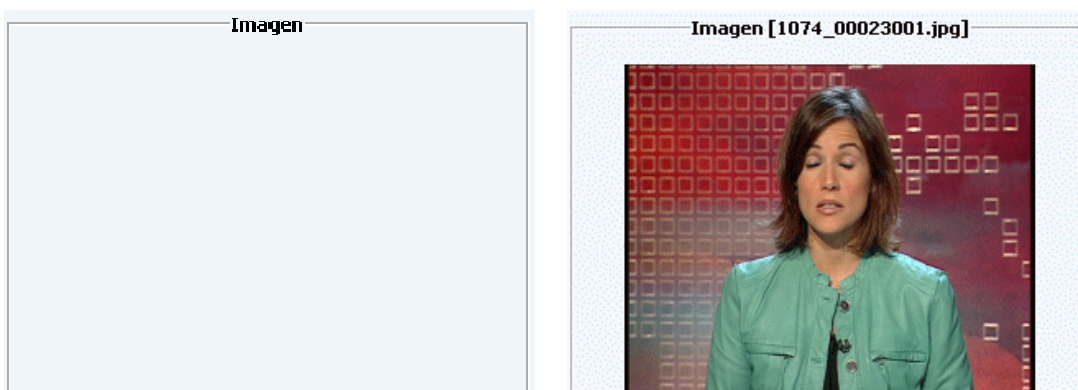


Figura 87. Panel de imagen consulta sin imagen y con imagen.



2.2.3.2 Panel de criterios de fusión

Panel compuesto por un conjunto de cuatro botones en la parte superior, correspondientes a los cuatro criterios de fusión que soporta el sistema de búsqueda. Bajo los botones de los criterios de fusión, se encuentra la colección de descriptores visuales disponibles en el sistema de búsqueda. Estos descriptores son configurables, el usuario puede seleccionarlos, ponderarlos y combinarlos como desee.

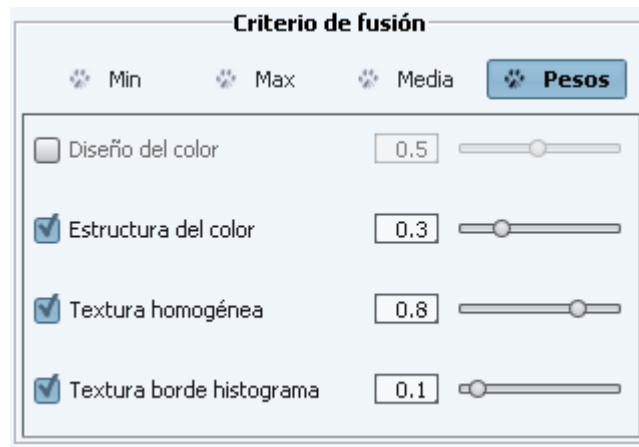


Figura 88. Panel de criterio de fusión

▪ TIPOS DE CRITERIOS DE FUSIÓN

- **Máximo (OR):** realiza la búsqueda tomando como distancia de ordenación la máxima obtenida.
- **Mínimo (AND):** realiza la búsqueda tomando como distancia de ordenación la mínima obtenida.
- **Media:** realiza la búsqueda promediando las distancias obtenidas para cada descriptor.
- **Pesos:** realiza la búsqueda combinando linealmente las distancias obtenidas para cada descriptor y ponderándolas con el peso indicado.

INFO: Por defecto, el método de fusión seleccionado es el "Mínimo" (Min).



▪ **TIPOS DE DESCRIPTORES VISUALES**

- ❖ **Diseño del color:** este descriptor permite representar la distribución espacial del color dentro de las imágenes.
- ❖ **Estructura del color:** este descriptor caracteriza la distribución de los colores en una imagen, construyendo una especie de histograma de color en el que tendrán mayor importancia los colores que más se repartan por la imagen. Permite distinguir entre dos imágenes que tengan la misma cantidad de píxeles de un color pero con distinta distribución.
- ❖ **Textura homogénea:** este descriptor utiliza un banco de 30 filtros que permite obtener una afinada descripción de las distintas texturas de la imagen para poder compararlas con las texturas de otras.
- ❖ **Histograma de bordes de textura:** este descriptor informa sobre el tipo de contornos o bordes que aparecen en la imagen. Trabaja dividiendo la imagen en 16 sub-imágenes y es capaz de analizar los tipos de borde existentes con el uso de distintos filtros que diferencian si se trata de bordes horizontales, verticales, oblicuos o aleatorios.

▪ **PONDERACIÓN DE LOS PESOS**

Si el criterio de fusión seleccionado es “Pesos”, el usuario puede establecer un valor normalizado entre 0 y 1, en pasos de 0.1, a cada descriptor empleado en la búsqueda a través de la barra deslizante.

INFO: Por defecto, aparecen seleccionados todos los descriptores visuales con un peso de 0.5.



2.2.3.3 Panel de espacio de búsqueda

El panel de espacio de búsqueda consta de dos partes: la selección del modo de ejecución del sistema de búsqueda y la selección del ámbito de búsqueda a través de las categorías disponibles.



Figura 89. Panel de espacio de búsqueda.

▪ MODO DE EJECUCIÓN

En la parte superior del panel, se sitúan dos botones de opción única para que el usuario escoja ejecutar el GOS en local o remoto.

INFO: Por defecto, el modo de ejecución es “Local”.

▪ CATEGORÍAS DE BÚSQUEDA

Bajos los botones de modo de ejecución, se muestran las categorías de búsqueda en forma de árbol. Actualmente se encuentran disponibles dos espacios de búsqueda, *CCMA* y *Mediapro*. Cada espacio de búsqueda corresponde al corpus de imágenes facilitadas por estas empresas y posee sus propias subcategorías (clasificación de las imágenes en la base de datos). La aplicación permite la selección de un único ítem.



INFO: Por defecto, si el usuario no selecciona ninguna opción, el ámbito de búsqueda es CCMA.

2.2.4 Área de resultados

La área de resultados ocupa la parte central de la aplicación. Se estructura en dos áreas diferenciadas:

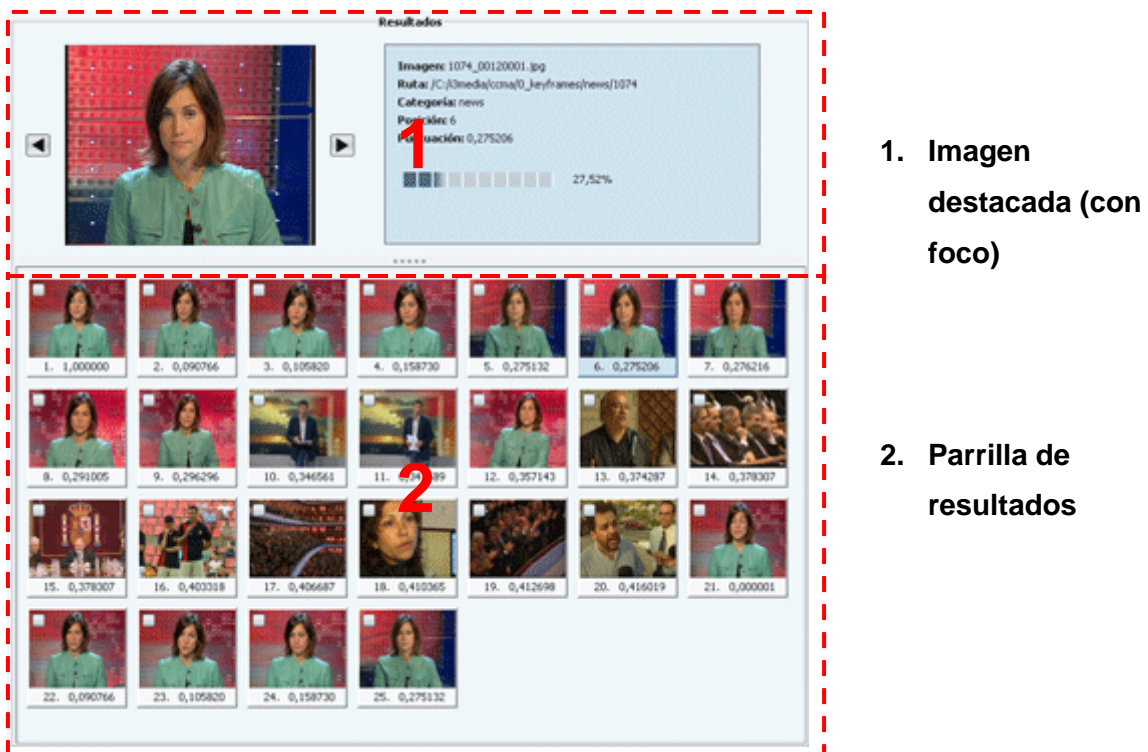


Figura 90. Panel de resultados

2.2.4.1 Imagen destacada (con foco)

Este panel situado en la parte superior de la zona de resultados, presenta una imagen destacada junto a su información asociada. Consta de los siguientes elementos:



- ❖ **Imagen destacada (con foco):** se muestra la imagen en un formato mayor al resto de resultados, con un tamaño de 250x200 píxeles.
- ❖ **Botones de navegación:** a los lados de la imagen destacada se encuentran dos botones de navegación, anterior y siguiente, que permiten desplazarse a través de los resultados con un simple clic.



Figura 91. Imagen destacada con los botones de navegación.

- ❖ **Área de texto informativa:** junto a la imagen, se presenta un cuadro de texto con una serie de información asociada que incluye el nombre del archivo imagen, su ruta, su categoría, su posición en la lista de resultados ordenados y la puntuación obtenida en la consulta. Como refuerzo visual, esta puntuación se muestra de forma gráfica a través de una barra de progresión junto a una cadena de texto con la puntuación en porcentaje.

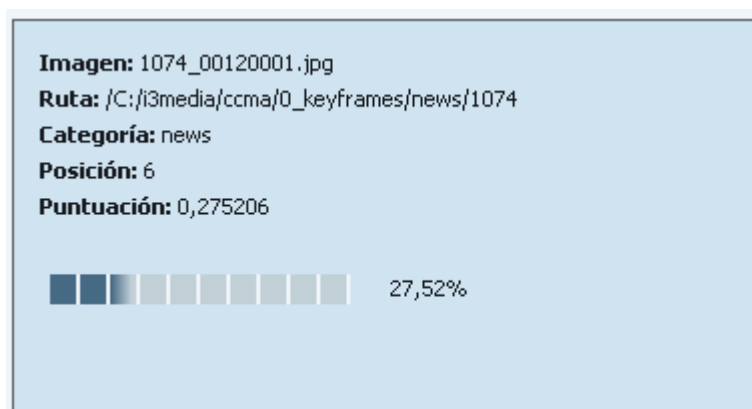


Figura 92. Área de texto informativa de la imagen destacada.



2.2.4.2 Parrilla de resultados

Todos los resultados obtenidos en la búsqueda se visualizan en este panel. Las imágenes son ordenadas por grado de similitud con la consulta (orden decreciente de mayor a menor similitud) presentadas en una parrilla. Cada resultado adopta la apariencia de una pequeña diapositiva.



Figura 93. Parrilla de imágenes

- **DIAPOSITIVA**

Cada diapositiva consta de tres elementos:

- ❖ **Imagen miniatura (thumbnail):** imagen en formato reducido, con tamaño de 100x80 píxeles.
- ❖ **Checkbox:** casilla de verificación en el margen superior izquierdo para seleccionar la imagen.
- ❖ **Texto informativo:** indica la posición de la imagen en la lista de resultados y la puntuación obtenida en la consulta.

Estas diapositivas se visualizan de forma diferente en función de su estado:



Si la diapositiva tiene el foco, es la imagen destacada en el panel superior de la zona de resultados, y se indica visualmente aplicando un fondo azul y un leve borde azul oscuro al panel de la diapositiva. Para dar foco a una imagen, basta con hacer clic sobre su diapositiva.



Si la imagen ha sido seleccionada por el usuario clicando sobre la casilla de selección, se indica a través del *checkbox*, que aparece seleccionado, y de un borde más grueso en un color azul intenso.



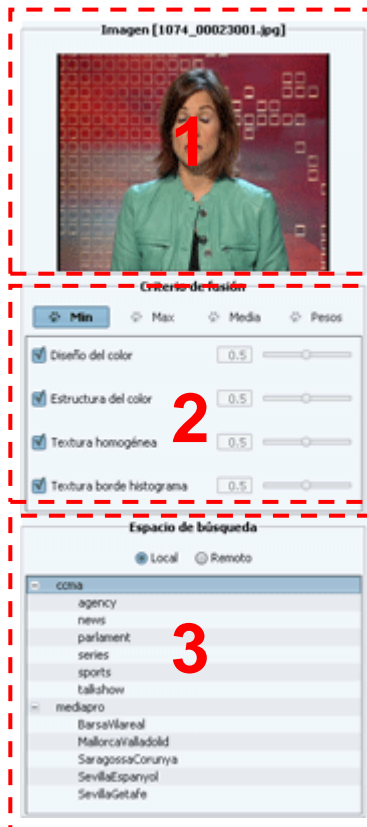
Si la imagen no es objeto de atención del usuario, se visualiza con un fondo blanco y un leve borde gris.



3. Cómo utilizar el GOS

3.1 Cómo buscar

Para realizar una búsqueda, el usuario debe seguir los siguientes pasos:



1. **Cargar la imagen de consulta**
2. **Configurar el motor de búsqueda:**
 - a. Seleccionando el criterio de fusión.
 - b. Seleccionando los descriptores a utilizar en la búsqueda.
 - c. Si el criterio de fusión es “Pesos”, determinando el peso de cada descriptor en la búsqueda.
3. **Configurar el ámbito de búsqueda:**
 - d. Seleccionado el modo de ejecución del motor de búsqueda.
 - e. Seleccionando la categoría dónde buscar.

Figura 94. Pasos para formular una búsqueda

Una vez formulada la consulta, el último paso es ejecutar la búsqueda.

Paso 1. Cargar la imagen de consulta

Podemos cargar la imagen de consulta a través de:

🐾 **Opción *Nueva Imagen* del menú *Archivo* (ver Figura 82. Menú Archivo en la barra de menú del GOS.)**

🐾 **Botón *Nueva Imagen* de la barra de herramientas:**





Se abrirá la siguiente ventana, que permite abrir archivos en formato BMP/GIF/JPG/JPEG/PNG.

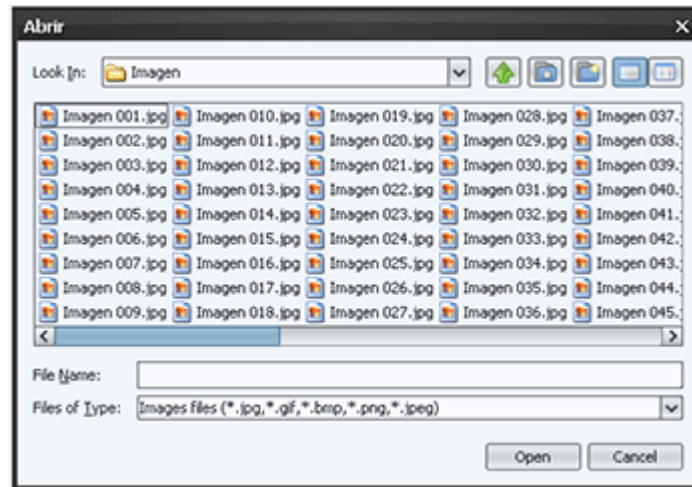


Figura 95. Ventana para abrir archivos imagen

Al cargar la imagen, en el título del panel de imagen de consulta aparece el nombre del archivo seleccionado.



Figura 96. Panel de imagen consulta con imagen cargada

Paso 2. Configurar el motor de búsqueda

❖ Seleccione el criterio de fusión

Por defecto, el método de fusión seleccionado es el “Mínimo” (Min), pero el usuario puede cambiar el criterio simplemente clicando sobre cualquiera de los cuatro botones. La opción seleccionada se destaca sobre el resto cambiando el



aspecto del botón, aplicando un fondo azul, poniendo el texto en negrita y acentuando el icono de la huella situada al lado del texto.

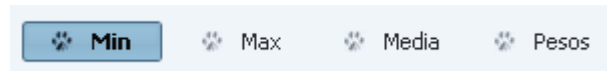


Figura 97. Tipos de criterio de fusión

🐾 Seleccione los descriptores visuales

Cada descriptor dispone de una casilla de selección (*checkbox*) para ser seleccionado o deshabilitado en la búsqueda. El usuario puede escoger los que desee, ya que el sistema permite realizar cualquier combinación (varios checkboxes seleccionados al mismo tiempo). Los descriptores no seleccionados aparecerán deshabilitados.

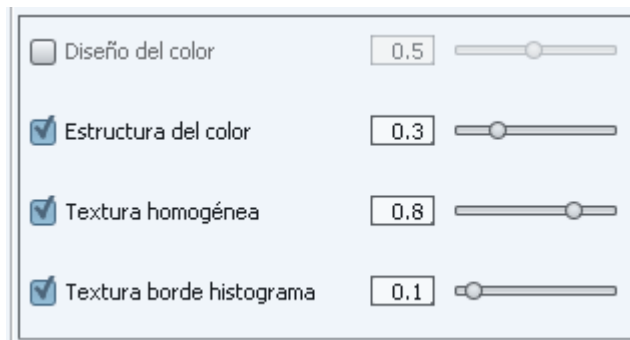


Figura 98. Descriptores visuales

🐾 Establezca los pesos de los descriptores (para “Pesos”)

Si el criterio de fusión seleccionado es “Pesos”, el usuario tiene la opción de otorgar a cada descriptor un factor de ponderación a través de su barra deslizable con rango de 0.0 a 1.0. Por defecto, el valor es 0.5.

INFO: Más información sobre los tipos de criterio de fusión y los descriptores visuales que utiliza el sistema en el apartado 0



Paso 3. Configurar el ámbito de búsqueda

❗ Seleccione el modo de ejecución

Dispone de dos modos de ejecución:

- **Local:** si se dispone del ejecutable del B_RANKER en la máquina (se debe tener la estructura de directorios en local para poder realizar búsquedas).
- **Remoto:** B_RANKER se puede ejecutar en modo remoto a través de un servicio web sobre SOAP. El GOS utiliza el servidor *upseek* que dispone de este servicio.



Figura 99. Modos de ejecución

Por defecto, el modo de ejecución es local.

❗ Seleccione la categoría de búsqueda

Puede elegir la categoría dónde realizar la búsqueda. La aplicación permite la selección de un único ítem. Por defecto, si el usuario no selecciona ninguna opción, el ámbito de búsqueda es *CCMA*.

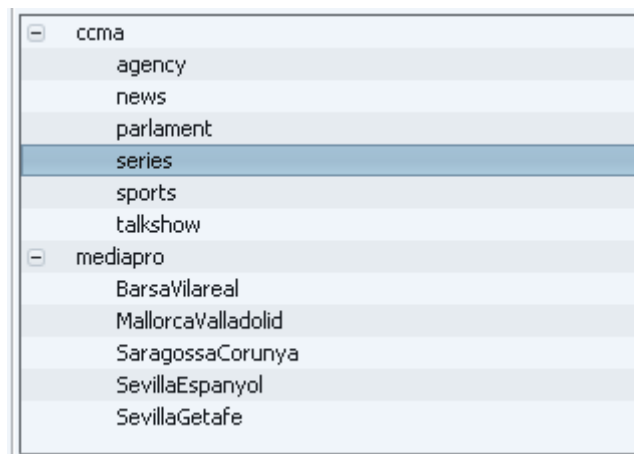



Figura 100. Categorías de búsqueda



Paso 4. Ejecutar la búsqueda

Podemos lanzar la búsqueda:

- ❗ **Opción *Buscar* del menú *Archivo*** (ver Figura 82. Menú *Archivo* en la barra de menú del GOS.)
- ❗ **Botón *Buscar* de la barra de herramientas:** 

Los resultados se visualizan automáticamente en la pantalla al finalizar la búsqueda (ver Figura 90. Panel de resultados).

Si se produce cualquier error durante la búsqueda, se muestra en pantalla el siguiente mensaje:

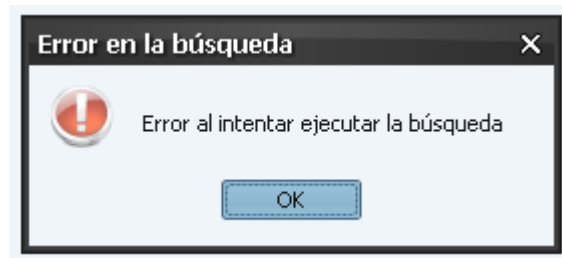


Figura 101. Mensaje que informa sobre errores durante la búsqueda

3.2 Cómo navegar por los resultados

La parrilla de imágenes miniatura nos muestra todos los resultados obtenidos en la búsqueda, pero la información sobre cada resultado es muy reducida. Para conocer más información sobre una imagen resultado, debe obtener el foco para ocupar el lugar de imagen destacada (ver 0).



3.2.1 Convertir una imagen en imagen destacada

Puede darle foco haciendo un sólo clic sobre la diapositiva. Cuando una imagen obtiene el foco, su diapositiva cambia el fondo de blanco a azul y su borde de gris a un azul más oscuro, y automáticamente pasa a ocupar el lugar de la imagen destacada del panel superior.

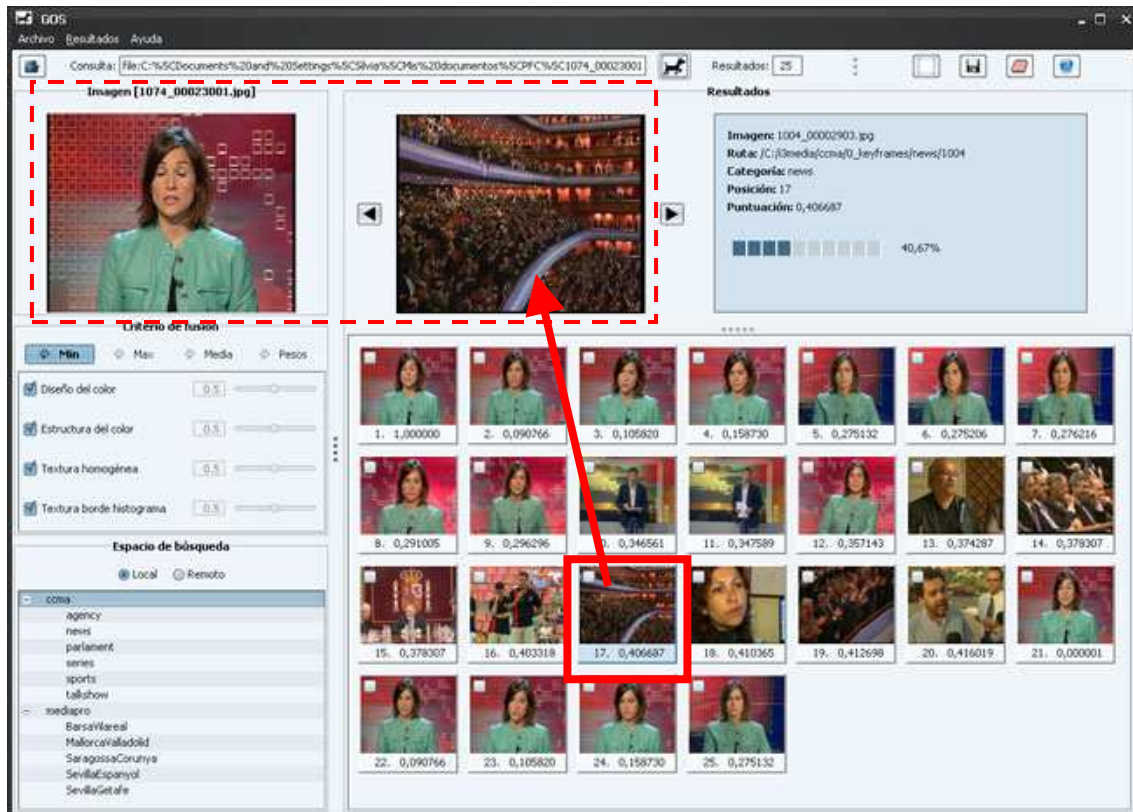


Figura 102. Convertir una imagen en imagen destacada para compararla con el ejemplo

La imagen destacada tiene un tamaño mayor que el resto de resultados: 250x200 píxeles. Su colocación en la pantalla, a la misma altura que la imagen consulta, así como su mismo tamaño, permite al usuario comparar fácilmente ambas imágenes, el ejemplo con el resultado obtenido.

Si se prefiere ver con detalle y de forma rápida todo el conjunto de resultados, de forma correlativa, puede navegar de dos formas: con los botones de navegación (ver 3.2.2) o con la rueda central del ratón (ver 3.2.3).



3.2.2 Desplazarse con los botones de navegación

Los botones de navegación “Anterior” y “Siguiente”, situados a los lados izquierdo y derecho respectivamente, permiten recorrer la lista de resultados con simples clics, sin necesidad de desplazarnos a la zona de la parrilla para clicar sobre la imagen deseada. Con cada clic, la imagen destacada cambia.



Figura 103. Botones de navegación al lado de la imagen destacada.

3.2.3 Desplazarse con la rueda central del ratón

Utilizando la rueda central del ratón, el usuario puede desplazarse por la lista de resultados, arriba y abajo, sin necesidad de ir realizando clics. Con cada movimiento de la rueda, la imagen destacada cambia.

3.3 Cómo seleccionar y guardar resultados

El GOS permite al usuario seleccionar aquellos resultados que sean de su interés y guardarlos en un archivo XML. La selección puede realizarse de forma individual o en grupo.

3.3.1 Selección individual

Haciendo clic sobre la casilla de selección (*checkbox*) de la diapositiva.

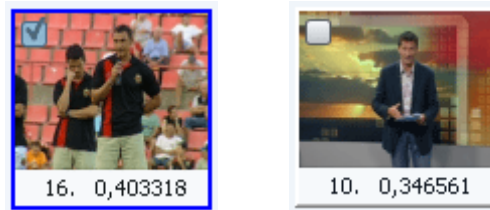



Figura 104. Diapositiva seleccionada y deseleccionada respectivamente.

Puede ir añadiendo resultados a su selección con cada clic (se van acumulando). Para deseleccionar una imagen, haga clic de nuevo en la casilla de verificación.

3.3.2 Selección en grupo

Puede seleccionar de una sola vez un grupo de resultados. Al activar la selección rectangular el cursor aparecerá en forma de cruz, y no volverá a aparecer la forma de flecha hasta que no se desactive esta función.

Haga clic sobre el botón de *Selección Rectangular*  que se encuentra en la barra de herramientas para activar este tipo de selección.

A continuación haga clic sobre la parrilla de imágenes, en el punto donde quiera empezar la selección y arrastre el botón hasta que el rectángulo azul que se dibuja incluya todos los resultados que desee seleccionar. Al soltar el ratón, todas las imágenes que se encuentren bajo el rectángulo, total o parcialmente, serán seleccionadas (ver Figura 105).

Una vez hecha la selección, puede deshacer la selección haciendo clic en cualquier punto de la parrilla.

INFO: La selección rectangular solo funciona en el área de resultados destinada a la parrilla de imágenes. Si hace clic fuera de esta área, se desactivará la herramienta.

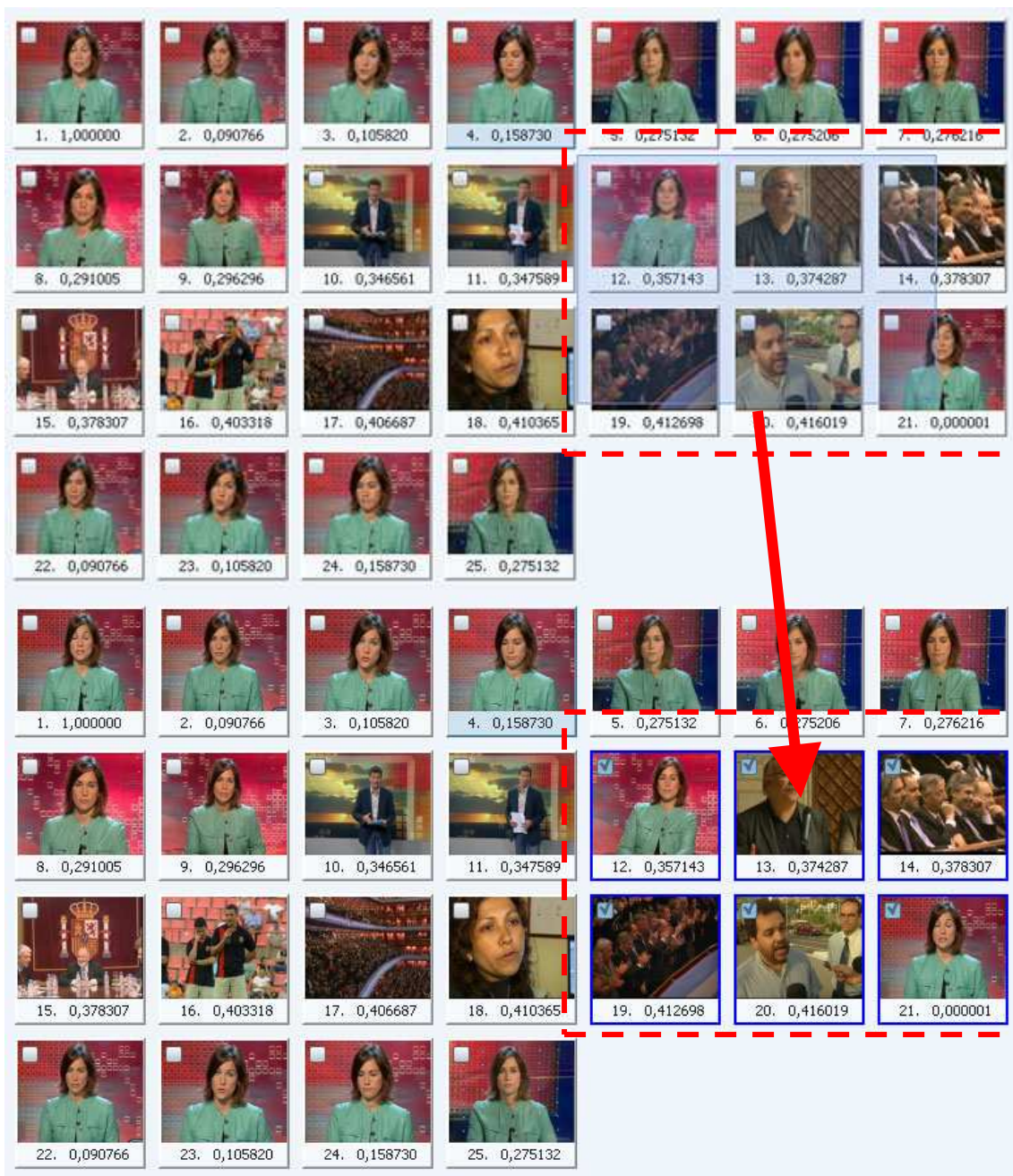


Figura 105. Selección en grupo con la herramienta de selección rectangular

3.3.3 Selección combinada

Puede combinar la selección individual y en grupo. Para ello, deberá primero utilizar la selección rectangular para realizar una selección en grupo, desactivar la herramienta y añadir selecciones individuales a la selección anterior haciendo clic sobre la casilla de verificación de cada diapositiva. Este procedimiento no funciona si



se realiza a la inversa, ya que si se activa la herramienta de selección rectangular después de haber realizado selecciones individuales, al hacer clic sobre la parrilla de imágenes se desharía la selección previa.

3.3.4 Deseleccionar resultados

Para deseleccionar una imagen, haga clic en su casilla de verificación activada. Si desea deseleccionar todos los resultados a la vez, active la función de selección rectangular y haga clic sobre cualquier punto de la parrilla de imágenes.

3.3.5 Guardar resultados

Podemos guardar los resultados seleccionados a través de:

❖ **Opción *Guardar seleccionados* del menú *Resultados* (ver Figura 83. Menú Resultados en la barra de menú del GOS.)**

❖ **Botón *Guardar* de la barra de herramientas:**



Se abrirá la siguiente ventana, que permite guardar un archivo XML en disco:

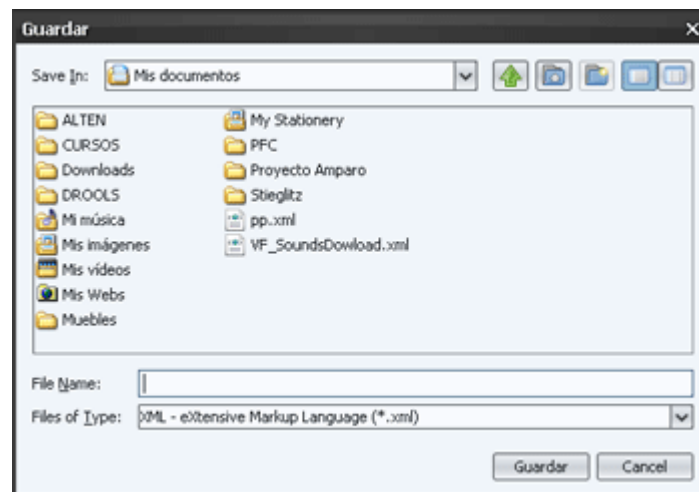


Figura 106. Ventana para guardar resultados



3.4 Cómo afinar la búsqueda

Cualquier imagen resultado puede ser utilizada como punto de partida para una nueva consulta, haciendo doble clic directamente sobre la imagen resultado que se desee configurar como nueva imagen ejemplo.

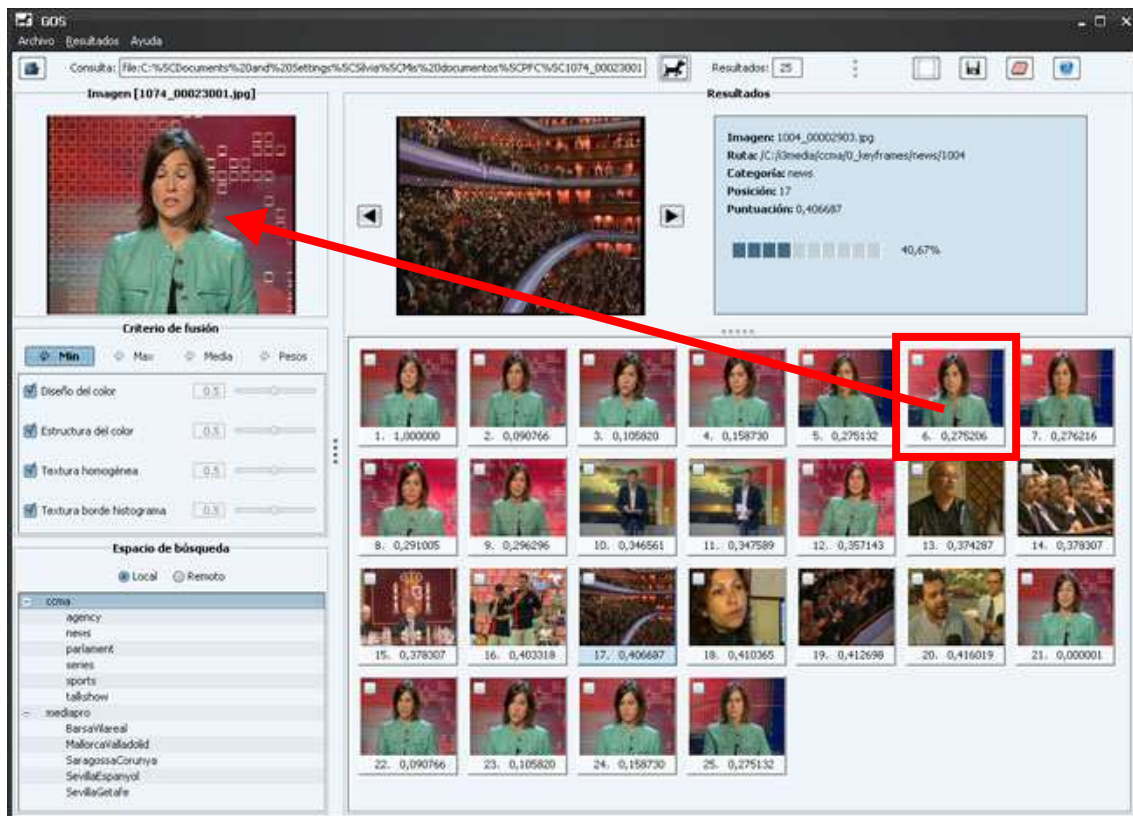



Figura 107. Usar un resultado como imagen consulta en una nueva búsqueda

3.5 Cómo limpiar la pantalla


Podemos limpiar la pantalla para realizar búsquedas desde cero a través de:

- 🐾 Opción **Limpiar** del menú **Resultados** (ver Figura 83. Menú Resultados en la barra de menú del GOS.)
- 🐾 Botón **Limpiar** de la barra de herramientas: 



3.6 Cómo consultar la ayuda

Podemos consultar la ayuda de GOS a través de:

- ❖ **Pulsando F1.**
- ❖ **Opción *Ayuda de GOS* del menú *Resultados* (ver Figura 84. Menú *Ayuda* en la barra de menú del GOS.)**
- ❖ **Botón *Ayuda* de la barra de herramientas:** 

ANEXO II. ARCHIVOS DE EJEMPLO

Fichero de datos de entrada

Archivo de configuración con los datos de entrada para el motor de búsqueda (programa B_RANKER). Disponible en el CD de documentación en:

Archivos de ejemplo\Entrada_1764_00100020.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Description xsi:type="urn:ContentEntityType" xmlns:urn="urn:mpeg:mpeg7:schema:2001">
    <MultimediaContent xsi:type="urn:MultimediaCollectionType">
      <Collection xsi:type="urn:ContentCollectionType">
        <CreationInformation>
          <Creation>
            <CreationTool>
              <Setting name="ResAmount" value="21"/>
              <Setting name="FusionCriterion" value="min"/>
              <Setting name="ColorLayoutType" value="0.5"/>
              <Setting name="ColorStructureType" value="0.5"/>
              <Setting name="HomogeneousTextureType" value="0.5"/>
              <Setting name="TextureEdgeHistogramType" value="0.5"/>
              <Setting name="UseScale" value="image"/>
            </CreationTool>
          </Creation>
          <RelatedMaterial id="SearchSpace">
            <MediaLocator>

<MediaUri>/home/xavi/i3media//2_semantic/queryByExample/1_image/2_databases/ccma.txt</Me
diaUri>
          </MediaLocator>
        </RelatedMaterial>
      </CreationInformation>
    </Collection>
  </MultimediaContent>
</Description>
</Mpeg7>
```

Fichero de datos de salida

Archivo de salida con los resultados de la búsqueda (programa B_RANKER).
Disponible en el CD de documentación en:

Archivos de ejemplo\Salida_1764_00100020.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Description xsi:type="ContentEntityType">
    <MultimediaContent xsi:type="MultimediaCollectionType">
      <Collection xsi:type="ContentCollectionType">
        <CreationInformation>
          <Creation>
            <Creator>
              <Role>
                <Name>i3server</Name>
              </Role>
              <Agent id="localhost" />
            </Creator>
            <CreationCoordinates>
              <Date>
                <TimePoint>Sun May 24
19:00:44 2009
</TimePoint>
              </Date>
            </CreationCoordinates>
            <CreationTool>
              <Tool>
                <Name>B_RANKER alfa</Name>
              </Tool>
              <Setting MinScore="0.000000" />
              <Setting
ColorLayoutType="0.500000" />
              <Setting
ColorStructureType="0.500000" />
              <Setting
TextureEdgeHistogramType="0.500000" />
              <Setting
HomogeneousTextureType="0.500000" />
              <Setting numOfResults="25" />
              <Setting FusionCriterion="min" />
            </CreationTool>
          </Creation>
          <RelatedMaterial id="SearchSpace">
            <MediaLocator>
              <MediaUri>file:/home/xavi/i3media//2_semantic/queryByExample/1_image/2_databases/
ccma.txt</MediaUri>
            </MediaLocator>
          </RelatedMaterial>
        </Collection xsi:type="ContentCollectionType">
          <Creation>
            <Creator>
              <Role>
                <Name>i3user</Name>
              </Role>
              <Agent id="username" />
            </Creator>
            <CreationTool>
              <Tool>
                <Name>GOS beta</Name>
              </Tool>
            </CreationTool>
          </Creation>
          <ContentCollection name="QueryXXX">
            <Content xsi:type="RegionsType">
              <MediaLocator>
```



```

    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1035/1035_00010
923.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
    </Content>
    <Content xsi:type="ImageType">
    <Image>
                                <Relation type="score" id="7"
strength="0.771875"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1004/1004_00004
200.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
    </Content>
    <Content xsi:type="ImageType">
    <Image>
                                <Relation type="score" id="8"
strength="0.768750"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1003/1003_00005
809.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
    </Content>
    <Content xsi:type="ImageType">
    <Image>
                                <Relation type="score" id="9"
strength="0.765625"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1003/1003_00005
805.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
    </Content>
    <Content xsi:type="ImageType">
    <Image>
                                <Relation type="score" id="10"
strength="0.765625"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1034/1034_00003
420.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
    </Content>
    <Content xsi:type="ImageType">
    <Image>
                                <Relation type="score" id="11"
strength="0.765420"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1004/1004_00003
408.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
    </Content>
    <Content xsi:type="ImageType">
    <Image>
                                <Relation type="score" id="12"
strength="0.764063"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1004/1004_00004
416.jpg</MediaUri>
                                </MediaLocator>
                                </Image>

```

```

        </Content>
        <Content xsi:type="ImageType">
            <Image>
                <Relation type="score" id="13"
strength="0.762500"/>
                <MediaLocator
xsi:type="ImageLocatorType">
                    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1003/1003_00003
712.jpg</MediaUri>
                </MediaLocator>
            </Image>
        </Content>
        <Content xsi:type="ImageType">
            <Image>
                <Relation type="score" id="14"
strength="0.760938"/>
                <MediaLocator
xsi:type="ImageLocatorType">
                    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1004/1004_00004
308.jpg</MediaUri>
                </MediaLocator>
            </Image>
        </Content>
        <Content xsi:type="ImageType">
            <Image>
                <Relation type="score" id="15"
strength="0.759375"/>
                <MediaLocator
xsi:type="ImageLocatorType">
                    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1034/1034_00000
507.jpg</MediaUri>
                </MediaLocator>
            </Image>
        </Content>
        <Content xsi:type="ImageType">
            <Image>
                <Relation type="score" id="16"
strength="0.759375"/>
                <MediaLocator
xsi:type="ImageLocatorType">
                    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1036/1036_00003
602.jpg</MediaUri>
                </MediaLocator>
            </Image>
        </Content>
        <Content xsi:type="ImageType">
            <Image>
                <Relation type="score" id="17"
strength="0.757812"/>
                <MediaLocator
xsi:type="ImageLocatorType">
                    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1037/1037_00001
720.jpg</MediaUri>
                </MediaLocator>
            </Image>
        </Content>
        <Content xsi:type="ImageType">
            <Image>
                <Relation type="score" id="18"
strength="0.753440"/>
                <MediaLocator
xsi:type="ImageLocatorType">
                    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1004/1004_00004
120.jpg</MediaUri>
                </MediaLocator>
            </Image>
        </Content>
        <Content xsi:type="ImageType">
            <Image>
                <Relation type="score" id="19"
strength="0.753125"/>

```

```

                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1036/1036_00002
800.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
                                </Content>
                                <Content xsi:type="ImageType">
                                <Image>
                                <Relation type="score" id="20"
strength="0.751562"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1004/1004_00003
810.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
                                </Content>
                                <Content xsi:type="ImageType">
                                <Image>
                                <Relation type="score" id="21"
strength="0.750000"/>
                                <MediaLocator
xsi:type="ImageLocatorType">
    <MediaUri>file:/home/xavi/i3media/1_perceptual/1_images/ccma/news/1003/1003_00010
921.jpg</MediaUri>
                                </MediaLocator>
                                </Image>
                                </Content>
                                </ContentCollection>
                                </Collection>
                                </MultimediaContent>
                                </Description>
</Mpeg7>

```

Fichero de directorio del espacio de búsqueda

Archivo .txt donde se listan las direcciones de los archivos descriptores (.xml) de las imágenes pertenecientes a cada espacio de búsqueda (bases de datos).

Disponible en el CD de documentación en:

Archivos de ejemplo\EspacioBusqueda_news.txt

```
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000000-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000007-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000021-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000103-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000111-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000117-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000124-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000207-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000410-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000420-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000501-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000610-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000614-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000619-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000624-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000703-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000805-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000809-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000811-  
image.xml  
/imatge/xgiro/i3media/1_perceptual/5_features-image/ccma/news/1003/1003_00000813-  
image.xml
```

.
.br/>.

Fichero de descripción de un imagen

Archivo con la descripción de una imagen (descriptores visuales).

Disponible en el CD de documentación en:

Archivos de ejemplo\1074_00023001-image.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <Description xmlns:urn="urn:mpeg:mpeg7:schema:2001"
xsi:type="urn:ContentEntityType">
    <MultimediaContent xsi:type="urn:ImageType">
      <Image>
        <CreationInformation>
          <Creation>
            <Title>MPEG-7 description of an
image</Title>
            </Creation>
            <RelatedMaterial id="ImageFile">
              <MediaLocator>
                <MediaUri>/imatge/xgiro/i3media/l_perceptual/l_images/ccma/news/1074/1074_0002300
1.jpg</MediaUri>
              </MediaLocator>
            </RelatedMaterial>
          </CreationInformation>
          <SpatialDecomposition size="1" id="image" gap="false"
overlap="true">
            <StillRegion id="Region1">
              <Collection
xsi:type="DescriptorCollectionType">
                <VisualDescriptor
xsi:type="ColorLayoutType" numOfYCoeff="64" numOfCCoeff="64">
                  <YCoeff>
                    <YDCCoeff>16</YDCCoeff>
                    <YACCCoeff>19 8 16 16
11 17 19 16 19 15 17 17 17 14 16 18 17 15 17 17 17 17 18 17 17 16 16 16 17 16 15 16
16 16 16 15 16 16 17 15 18 16 17 16 17 17 16 17 17 16 16 16 16 16 16 16 16 16 16
16 </YACCCoeff>
                    </YCoeff>
                    <CbCoeff>
                      <CbDCCoeff>23</CbDCCoeff>
                      <CbACCCoeff>14 13 17 15
17 16 17 17 18 16 17 15 16 15 17 17 17 16 16 17 16 16 16 17 17 16 16 16 16 17 17
16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16
16 </CbACCCoeff>
                      </CbCoeff>
                      <CrCoeff>
                        <CrDCCoeff>47</CrDCCoeff>
                        <CrACCCoeff>16 29 14 18
24 14 11 15 12 18 13 17 16 16 14 10 15 16 14 15 16 16 14 15 15 17 15 16 17 16 16 15 14
16 17 16 17 15 17 16 17 16 15 15 16 16 15 16 16 16 16 16 16 16 16 16 16 16 16 16
16 </CrACCCoeff>
                        </CrCoeff>
                      </VisualDescriptor>
                    <VisualDescriptor
xsi:type="ColorStructureType" numValues="256">
                      <Values>21 6 6 6 7 3 6 4 3 3
4 7 10 10 10 11 10 7 5 3 2 1 1 1 1 0 0 0 0 0 0 25 33 27 36 38 35 29 12 3 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 39 85 84 56 15 10 14 63 86 59 11 8 4 3 11 62 65 24 2 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3
82 78 54 6 2 1 15 35 22 6 5 2 2 4 73 117 28 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 </Values>
                </VisualDescriptor>
              </Collection>
            </StillRegion>
          </SpatialDecomposition>
        </Image>
      </MultimediaContent>
    </Description>
  </Mpeg7>
</instance>
```



```

xsi:type="ShapeAreaType">
    <Occupation>100.000000</Occupation>
    <VisualDescriptor>
    <VisualDescriptor
        <NumPixels>414720</NumPixels>
    </VisualDescriptor>
    </VisualDescriptor>
xsi:type="ShapeBoxType">
    height="576"/>
    <Occupation>100.000000</Occupation>
    <VisualDescriptor>
    <VisualDescriptor
        <Size width="720"
    </VisualDescriptor>
xsi:type="TextureEdgeHistogramType">
    <BinCounts>4 4 7 8 8 5 5 8 8
    8 3 4 7 7 6 2 2 6 6 4 4 4 7 7 7 4 3 7 7 7 4 3 7 7 7 2 2 6 6 4 1 3 5 5 5 3 3 6 5 6 4 3 6
    6 6 2 1 4 4 3 1 2 4 4 4 3 2 6 5 5 4 2 7 7 6 2 2 5 5 4 </BinCounts>
    </VisualDescriptor>
    <VisualDescriptor
xsi:type="HomogeneousTextureType">
    <Energy>10.88698 11.45966
    10.94075 6.04525 0.00010 0.00000 3.41465 2.23398 0.00500 0.00000 0.00000 0.00000 0.00000
    0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
    0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 </Energy>
    <EnergyDeviation>13.64664
    14.21933 13.70042 8.80491 0.05594 0.00000 6.17414 4.99111 0.88391 0.00000 0.00000
    0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000
    0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 0.00000 </EnergyDeviation>
    </VisualDescriptor>
    </Collection>
    </StillRegion>
    </SpatialDecomposition>
    </Image>
    </MultimediaContent>
    </Description>
</Mpeg7>

```

ANEXO III. RECURSOS ONLINE

Web del GOS

Disponible online en: <http://gps-tsc.upc.es/imatge/i3media/gos/>

Wednesday, 3 June 2009 Eng | Cat | Esp



GOS - Graphic Object Searcher

[What is it?](#) | [Contact](#)

What is it?

GOS is a graphic user interface for a Content-Based Image Retrieval (CBIR) system that supports Query-by-Example method of search. It's developed by the [Image and Video Processing Group](#) of the [Signal Theory and Communications Department](#) of the [Technical University of Catalonia](#).



Play with it

Available on [Sun's Java Web Start](#) technology.

[Launch](#)

Watch it work

Video: [Demo of image-based query with different descriptors](#)

Publications

Xavier Giró and Ferran Marqués, "System Architecture for Indexing Regions in Keyframes", Poster and Demo Proceedings of the [3rd International Conference on Semantic and Digital Media Technologies \(SAMT\)](#), Koblenz, Germany, 3rd-5th December 2008. [\[Short paper\]](#) [\[Poster\]](#)

Who are the authors

Researchers

Silvia Cortés, [Xavier Giró](#) and [Ferran Marqués](#).

Technical & Designers

Technical suport by Albert Gil.
Logo by [Carles Ballesteros](#) and Silvia Cortés. Web design by Silvia Cortés.

Partners



Generalitat de Catalunya
Corporació Catalana
de Mitjans Audiovisuals



MEDIAPRO



i3media



UPC

Grup de processament d'imatge | Universitat Politècnica de Catalunya | Campus Nord UPC | Jordi Girona 1-3 08034 Barcelona

Vídeo Demo

Demostración de consultas basadas en imágenes y diversos descriptores .

Disponible en: <http://gps-tsc.upc.es/imatge/i3media/gos/videos/gos.avi>

