

Algoritmo evolutivo para la búsqueda de imágenes basada en el contenido

Miguel Arevalillo-Herráez, Francesc J. Ferri

Resumen— Los sistemas de recuperación de imágenes basados en el contenido intentan proporcionar un medio de búsqueda de imágenes en repositorios de gran tamaño sin utilizar ninguna otra información aparte de la contenida en la propia imagen, normalmente en forma de descriptores de bajo nivel. Dado que estos descriptores no representan con exactitud la semántica de la imagen, la evaluación de la similitud perceptual entre dos imágenes en base únicamente a ellos no es una tarea trivial. De hecho, la habilidad de un sistema para inducir conceptos semánticos de alto nivel a partir de los vectores de características de una imagen es uno de los aspectos que más afecta a su rendimiento. Este artículo describe un método efectivo para la recuperación de imágenes basado en técnicas de computación evolutiva. Los resultados se comparan con los obtenidos mediante el enfoque clásico del movimiento del punto de consulta y el re-escalado de los ejes y mediante una técnica basada en mapas autoorganizativos, evidenciando un rendimiento notablemente superior en los repositorios utilizados.

Palabras clave— recuperación de imágenes, IGA, algoritmo genético

I. INTRODUCCIÓN

La recuperación de la información trata el problema de la búsqueda de recursos digitales en bases de datos de gran tamaño. Los sistemas de recuperación de imágenes basados en el contenido son un tipo especial de sistemas de recuperación de información donde los recursos son imágenes. Comúnmente, estos sistemas representan una imagen mediante una serie de descriptores, relacionados con el color, la textura o la forma. Estos descriptores se utilizan en el proceso de recuperación para juzgar la similitud entre imágenes. Los sistemas más recientes siguen un modelo iterativo, y consideran al usuario parte activa del proceso. En cada iteración se le muestran un conjunto de imágenes, y el usuario marca aquellas que más se parecen al concepto que busca. El resultado del procesamiento de esta información es un nuevo conjunto de imágenes que vuelven a mostrarse, continuando con el proceso iterativo hasta que se encuentra la imagen buscada.

Por otro lado, la computación evolutiva trata la aplicación de la selección natural y los conceptos genéticos a modelos computacionales, abarcando una serie de técnicas basadas en la reproducción, competición, selección y existencia de mutaciones,

tales como los algoritmos genéticos [1] y las estrategias evolutivas [2]

La aplicación de técnicas de computación evolutiva en el campo de la recuperación de la información no es novedosa. En particular, se ha aplicado a aspectos específicos como la agrupación de resultados [3], la optimización de consultas [4] y la realimentación [5]. En el caso de la búsqueda de imágenes, se han utilizado estrategias evolutivas para medir la similitud entre dos imágenes combinando descriptores [6] y para construir sistemas de recuperación completos utilizando mecanismos interactivos [7], [8].

Este artículo presenta la utilización de un algoritmo genético interactivo como mecanismo de realimentación eficiente para la búsqueda de imágenes. El sistema ha sido evaluado y comparado con otros algoritmos existentes, obteniendo resultados prometedores.

II. LA REALIMENTACIÓN EN LA RECUPERACIÓN DE IMÁGENES

El propósito fundamental de un sistema de recuperación de imágenes es permitir al usuario realizar búsquedas en repositorios. Como en la mayoría de las ocasiones el interés del usuario puede únicamente expresarse en términos semánticos, el sistema debe ser capaz de traducir estos términos en representaciones de bajo nivel que puedan ser procesadas. La suposición de que la similitud semántica está relacionada con la similitud entre descriptores de bajo nivel queda implícita en este procedimiento. Dado que esta suposición no es completamente cierta, el objetivo de la mayoría de las técnicas es reducir el salto existente entre la semántica inducida a partir de los descriptores y semántica real de la imagen.

La propuesta de utilizar la realimentación como mecanismo básico de búsqueda ha sido heredada del campo de la recuperación de información textual [9]. Este mecanismo consiste en refinar la consulta original de forma progresiva e interactiva, utilizando la información proporcionada por el usuario. En el contexto de la recuperación de imágenes, la realimentación se apoya en alguna forma de aprendizaje supervisado, y permite reducir el “salto semántico”, convirtiendo la búsqueda en un proceso iterativo, dinámico e interactivo que ajusta gradualmente los resultados al interés del usuario.

La forma más común de realimentación consiste en solicitar al usuario que juzgue los resultados recuperados en cada iteración, ya sea marcándolos como

relevantes o no relevantes, o por medio de una barra de desplazamiento que permite asignar un grado de relevancia. Esta información se utiliza para refinar la consulta y producir un nuevo conjunto de resultados, repitiendo el proceso hasta obtener un resultado satisfactorio. Un sistema de realimentación debe intentar minimizar la interacción disminuyendo tanto el número de iteraciones como la cantidad de elementos que el usuario debe juzgar en cada iteración, con objeto de reducir el tiempo medio de búsqueda.

Los enfoques clásicos, heredados del campo de la recuperación de la información textual, asumen la existencia de un punto de consulta y de un conjunto de pesos en el espacio de características que, de encontrarse, proporcionan la respuesta. En estos métodos, el mecanismo de realimentación utiliza la selección del usuario para actualizar tanto el punto de consulta como los pesos, de forma que se enfatizan más aquellas características que le parezcan más relevantes [10], [6].

Otros enfoques incluyen el uso de técnicas Bayesianas [12], [13], Mapas Autoorganizativos (SOMs) [14] y otros métodos más recientes como la utilización de Máquinas de Vectores Soporte (SVM) [16], [17], modelos regresivos [18], o enfoques basados en conjuntos y lógica borrosa [19], [20].

III. TÉCNICAS EVOLUTIVAS

Los algoritmos evolutivos intentan solucionar un problema aplicando los principios básicos de la evolución de Darwin [21] sobre una población de soluciones potenciales a un problema, denominadas individuos. Los algoritmos genéticos son un tipo específico de algoritmo evolutivo que enfatiza la aplicación de los operadores genéticos, como las mutaciones y los cruzamientos. Un algoritmo genético utiliza un método de codificación para representar las soluciones potenciales, y una medida que permite una evaluación cuantitativa de cada candidato, llamada función de evaluación.

Comenzando con una población inicial, un algoritmo genético utiliza la función de evaluación para evaluar cada individuo. Los más prometedores se reproducen y determinan la próxima generación, de acuerdo a una serie de reglas de evolución pre-establecidas basadas en operaciones de selección, mutación y cruzamientos. Este proceso se repite hasta que se satisface una condición de terminación, como que un individuo sea una solución exacta al problema original, o que su función de evaluación alcance un umbral determinado. La representación de los individuos y la determinación de las reglas que gobiernan la evolución de la población son aspectos fundamentales en el desarrollo de un algoritmo genético. Normalmente, las soluciones potenciales se codifican como secuencias binarias, de caracteres, o como vectores de valores enteros o reales, donde cada elemento representa un aspecto particular de la solución. En términos genéticos, la solución potencial se denomina fe-

notipo, y su representación genotipo, genoma o cromosoma. La definición de las reglas que gobiernan la evolución incluye la determinación del método de selección de padres, y la forma de realizar las operaciones de mutación y cruzamiento para derivar los genotipos de los individuos que constituirán la próxima generación. Para preservar la diversidad en la población y evitar la convergencia prematura a una solución subóptima, la mayoría de los métodos de selección de padres son estocásticos. Las operaciones de cruzamiento emulan la meiosis y las mutaciones simulan los cambios en el ADN debidos a la transcripción de errores, radiación y los radicales libres que ocurren en la naturaleza.

En algunas ocasiones, el diseño de una función de evaluación apropiada se torna una tarea excesivamente complicada, o incluso imposible. Los algoritmos genéticos interactivos utilizan la evaluación de un ser humano para abordar el problema. En este tipo de algoritmos, el usuario directa o indirectamente evalúa a los individuos. Su utilización ha sido extensa en numerosos y variados campos de investigación (ver, por ejemplo, [22] para una revisión exhaustiva).

En el caso de la recuperación de imágenes, y debido al “salto semántico”, las únicas evaluaciones realmente fiables son aquellas que provienen directamente del usuario. Esto indica que la utilización de una función de evaluación basada en la interacción con el usuario podría constituir un punto de partida adecuado para la aplicación de la computación evolutiva al problema de la recuperación de imágenes.

IV. EL ALGORITMO

A. La búsqueda

El propósito de un sistema de búsqueda de imágenes es recuperar aquellas que mejor satisfacen una consulta semántica. Para realizar las búsquedas, se extraen una serie de descriptores de cada imagen y se almacenan en vectores de características. Estas características se agrupan en descriptores coherentes, como puede ser el histograma de color, coeficientes de wavelets o características en el dominio de Fourier. En términos genéticos, la representación de características de una imagen sería equivalente a su genotipo, mientras que la imagen en sí sería su fenotipo. Aunque es común tener algoritmos en los que existe una función que permite generar el fenotipo a partir de su genotipo, en este caso no es posible generar una imagen a partir de su representación en el espacio de características en general.

Desde una perspectiva genérica, podemos asumir que la consulta tiene un fenotipo y un genotipo, y que, aunque el usuario conoce el fenotipo, desconoce su genotipo. El sistema de recuperación no conoce ninguno de los dos, y su objetivo es encontrar el genotipo de la imagen buscada. Es razonable asumir que la relación entre el fenotipo y el genotipo es fuertemente causal [23], de forma que pequeñas variaciones en el espacio de los genotipos implican

pequeñas variaciones en el espacio de los fenotipos. Dado que la estructura de la vecindad bajo una relación fuertemente causal se mantiene, los vecinos del genotipo serán también vecinos en el espacio del fenotipo. Desde este punto de vista, podemos considerar la búsqueda como una tarea colaborativa en la que el usuario y el sistema cooperan para encontrar el genotipo de la consulta.

B. Representación

En principio, la técnica presentada en este artículo puede utilizarse con cualquier conjunto de descriptores, aunque el rendimiento del sistema resultante dependerá fuertemente del grado en el que los descriptores satisfagan la condición de fuerte causalidad en la relación genotipo-fenotipo.

C. Función de evaluación

En un algoritmo genético, la función de evaluación se utiliza para medir la bondad de una solución. En cada iteración del proceso, la información capturada por el usuario puede utilizarse para construir esta función. Si consideramos que la población actual se compone de la imágenes que se han mostrado al usuario, aquellas que se han calificado como positivas son las que tienen un fenotipo que más se ajusta a la consulta. Por lo tanto, estas son las imágenes que deberían reproducirse y se les asigna una puntuación máxima. El resto de imágenes mostradas son descartadas del proceso de cruzamiento. Podría argumentarse que sería un procedimiento más habitual asignar una bondad inferior a estas otras imágenes, asignándoles una probabilidad menor de participar en la reproducción. Sin embargo, un algoritmo genético diseñado de esta forma podría tardar demasiadas iteraciones en converger. Por otro lado, restringiendo el conjunto de individuos que se reproducirá hasta este punto, las nuevas generaciones pueden no tener una diversidad suficiente, y quedar atrapadas en las inmediateces de una solución subóptima. Este hecho está relacionado con la denominada presión de selección, que se define como la probabilidad relativa de que el mejor individuo de la población se elija como padre en relación a un individuo medio. Reduciendo la población que se reproduce a las selecciones positivas, la presión de selección es máxima, y es bien sabido que presiones de selección demasiado altas suelen conducir a una convergencia rápida hacia una solución subóptima. Para paliar este efecto, aplicamos una operación que denominamos enriquecimiento de la población y que ayuda a mantener la diversidad. El enriquecimiento consiste en introducir algunos nuevos individuos adicionales extraídos del propio repositorio: los más cercanos a las selecciones positivas. Esta decisión viene apoyada por la fuerte causalidad supuesta en la relación genotipo-fenotipo, que implica que, dado que los vecinos tienen un genotipo similar, su fenotipo también debe ser parecido.

Para calcular los vecinos puede utilizarse una medida de distancia compuesta que utilice las medidas de distancia entre cada uno de los descriptores individuales, como una combinación lineal de distancias normalizadas [24] (utilizada en nuestro caso por razones de simplicidad) o la combinación probabilística presentada en [25]. Dado que únicamente necesitamos una ordenación aproximada, cualquier medida consistente que ofrezca una precisión razonable puede ser válida. Mientras que una medida muy precisa podría acelerar la convergencia del método, una menos precisa incrementaría la diversidad en la población y protegería al algoritmo contra convergencias rápidas a soluciones subóptimas.

D. Selección de padres

Los padres se eligen utilizando el algoritmo de la ruleta rusa, asignando la misma probabilidad a cada una de las selecciones positivas y a sus vecinos, y valor cero a cualquier otra imagen en la población. En realidad, la aplicación del algoritmo de la ruleta rusa en este caso es equivalente a seleccionar individuos aleatoriamente de entre la población enriquecida.

E. Reglas de evolución

En un algoritmo genético standard, los individuos de una nueva generación se evalúan mediante la función de evaluación. En un algoritmo genético interactivo, la función de evaluación se reemplaza por el juicio del usuario, y éste únicamente es posible en el espacio de los fenotipos.

Dado que no tenemos una función de conversión entre los dos espacios, no es posible que el usuario juzgue los genotipos que se generan mediante operaciones estándar de mutación y cruzamiento, a menos que el dominio de soluciones potenciales al problema se restrinja a aquellos genotipos que tienen un fenotipo conocido. En el contexto de una sistema de búsqueda, las soluciones potenciales son las imágenes en el repositorio. De poco sirve generar un genotipo más adecuado si no podemos generar el fenotipo correspondiente para proporcionárselo al usuario. Esta restricción en el dominio tiene implicaciones significativas en las operaciones de cruzamiento, y condiciona el procedimiento utilizado para simular la evolución.

A continuación detallamos algunos de los problemas asociados a la aplicación directa de las operaciones estándar de cruzamiento sobre nuestra representación de genotipos:

- La aplicación de operaciones de cruzamiento puede resultar en genotipos inconsistentes. Los descriptores generalmente deben satisfacer algunas restricciones. A modo de ejemplo, los valores de un histograma de color normalizado deben sumar uno, o las funciones acumulativas deben ser monótonamente crecientes.
- Incluso aunque el genotipo producido sea consistente, su fenotipo debe mostrarse al usuario en la

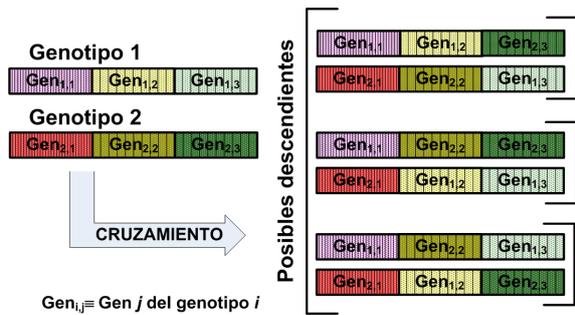


Fig. 1. Posibles resultados de una operación de cruzamiento

próxima iteración. Dado que no es posible generar fenotipos a partir de genotipos, si el genotipo no tiene una correspondencia directa en la base de datos, esta operación no sería posible.

- El número de selecciones positivas puede ser demasiado pequeño para permitir la construcción de una nueva población consistente que asegure la diversidad. Esto ocurre especialmente en la primera iteración, donde es común tener una o dos selecciones únicamente.

Para resolver el primer problema, consideramos cada descriptor como un gen independiente, y consideramos cada gen una unidad indivisible en las operaciones de cruzamiento. Como los genes en los genotipos de los padres son consistentes y no se alteran durante el proceso, permanecen consistentes en los hijos. En nuestra implementación realizamos el cruzamiento generando dos descendientes a partir de cada par de padres. El primer descendiente elige sus genes al azar de entre los dos padres, y el otro toma los que el primer descendiente no ha escogido. El concepto de un gen y sus implicaciones en las operaciones de cruzamiento se ilustran en la figura 1, en la que se muestran los posibles resultados de una operación de cruzamiento sobre dos genotipos con tres genes cada uno.

El segundo problema se soluciona buscando el genotipo más próximo existente en la base de datos, de acuerdo a la misma función compuesta de distancia utilizada para el cálculo de los vecinos más próximos. Esta operación puede considerarse como la aplicación de pequeñas mutaciones en el genotipo original para producir un genotipo existente en la base de datos y que, por lo tanto, tiene un fenotipo conocido. En realidad, cada uno de los hijos proporciona una ordenación de las imágenes en la base de datos, en la que el primer elemento se corresponde con la imagen más cercana. Para producir una ordenación global de todas las imágenes en la base de datos, cada imagen toma como valor su mejor posición entre todas las ordenaciones, y la ordenación final se calcula en base a estos valores.

La solución al tercer inconveniente ya se ha descrito en el contexto de la función de evaluación y

consiste en enriquecer la población con los vecinos más cercanos a las selecciones positivas.

Cada una de las iteraciones del algoritmo propuesto se puede describir de la siguiente forma:

1. Repetir k veces
 - a) Seleccionar un par de genotipos al azar.
 - b) Aplicar el operador de cruzamiento para producir un par de descendientes.
 - c) Ordenar todas las imágenes del repositorio por distancia a cada descendiente
 - d) Actualizar las puntuaciones de cada imagen en el repositorio

Dado que la realimentación es un proceso interactivo, el número de veces k puede definirse de forma que el algoritmo se ejecute en un tiempo aceptable. En nuestra implementación, hemos elegido un valor $k = 80$, de forma que el tiempo de procesamiento se mantenga por debajo de 2 segundos. Nótese que no hemos hecho uso de ningún operador de mutación en el proceso.

Como paso final, las selecciones positivas se desplazan a las primeras posiciones de la ordenación. Este paso es una forma de fuerte elitismo, y causa que los mejores individuos de la última generación sobrevivan en la próxima. Como mecanismo de apoyo, también se mantiene una historia de la búsqueda actual. Aquellas imágenes que han sido previamente mostradas pero que el usuario no ha seleccionado se marcan y no se muestran en iteraciones posteriores dentro de la misma búsqueda (se desplazan a las últimas posiciones de la ordenación).

F. Una perspectiva global

El algoritmo puede resumirse en los siguientes pasos:

1. Primeramente, se muestran una serie de imágenes en la interfaz gráfica, en el orden determinado por la ordenación establecida en la iteración anterior. Se solicita entonces que el usuario marque aquellas que son relevantes para la búsqueda.
2. Como parte de la operación de enriquecimiento de la operación, se utiliza una función compuesta de distancia para obtener los genotipos más cercanos a las selecciones positivas.
3. Se aplica el operador de cruzamiento sobre padres escogidos aleatoriamente de entre los individuos que componen la población enriquecida. Los genotipos resultantes son sintéticos, en el sentido de que pueden no corresponderse con una imagen existente en la base de datos. Para cada uno de estos genotipos, se produce una nueva ordenación, mediante la misma función de distancia que fue utilizada para encontrar los vecinos.
4. Se asigna a cada imagen una puntuación inversamente proporcional a su mejor posición en las ordenaciones.

V. EVALUACIÓN

En el campo de la recuperación de imágenes se han realizado importantes esfuerzos para establecer

métodos comunes de evaluación. Como buenos ejemplos podemos citar el mecanismo de evaluación automático propuesto en [26] y el evento Benchathlon [27] -una propuesta para constituir el equivalente a TREC (Text Retrieval Conference) para los sistemas de recuperación de documentos-. A pesar de ello, actualmente no existen sistemas de evaluación estándar para medir el rendimiento de este tipo de sistemas.

Para poder comparar el rendimiento de sistemas de recuperación de imágenes, es necesario utilizar un repositorio común. Aunque la colección denominada Corel Photo CD ha sido empleada por varios autores para evaluar sus algoritmos, cada uno lo ha hecho sobre un subconjunto de imágenes distinto, la colección está sujeta a derechos de autor, no es gratuita y ya no puede obtenerse a través de Corel. Además, la clasificación que proporciona no es adecuada para propósitos de prueba dado que, tal como se reporta en [28], muchas imágenes que representan conceptos similares se encuentran en directorios diferentes y algunos directorios contienen imágenes con un contenido semántico bien distinto.

Por ello, para poder evaluar nuestro algoritmo, hemos creado manualmente un conjunto de pruebas utilizando los dos repositorios que describimos a continuación:

- Una base de datos de pequeño tamaño compuesta por 1 508 imágenes, extraídas de la Web y manualmente clasificadas como pertenecientes a 28 conceptos semánticos distintos, como flores, caballos, cuadros, nubes, árboles, etc. El número de imágenes bajo cada categoría varía de 24 a 300. Esta base de datos y su clasificación han sido utilizadas en [13] y [18], donde se proporciona una explicación más detallada. Los descriptores utilizados en este caso son el histograma de color HS (con 10 x 3 componentes), y las granulometrías horizontal y vertical [29], calculadas en la forma especificada en [30]. Para estimar similitudes entre los descriptores de color de dos imágenes hemos utilizado la intersección del histograma [31]. Para el resto de los descriptores, y en general en el resto del artículo a menos que se especifique lo contrario, hemos utilizado la distancia euclídea.

- Un repositorio compuesto de 5 746 imágenes, extraídas manualmente de una colección de 102 894 imágenes de libre distribución denominada “Art Explosion” adquirida a través de la empresa Nova Development (<http://www.novadevelopment.com>). Las imágenes han sido cuidadosamente seleccionadas y clasificadas manualmente en 62 categorías, de forma que las imágenes que pertenecen a una misma categoría representan conceptos similares. Los descriptores calculados para estas imágenes son el histograma de color HS (con 10 x 3 componentes), y seis descriptores de textura: las energías de la convolución de Gabor [32], la matriz de coocurrencias en niveles de grises [33], campos aleatorios de Markov gaussianos [34], los coeficientes de ajustar la distribución granulométrica con una base de B-splines [35]

y dos versiones de la distribución de tamaño espacial [36], una utilizando un segmento horizontal y otra un segmento vertical.

Hemos comparado el rendimiento de nuestro algoritmo con otros dos enfoques de recuperación de imágenes representativos. En particular, comparamos nuestros resultados con los obtenidos utilizando un algoritmo clásico de reposicionamiento del punto de consulta y reescalado de ejes [37], [38] y mediante un motor que utiliza principios similares a los presentados en el sistema PicSOM [39], usando SOMs de tamaño 16x16 y 32x32 para los dos repositorios descritos anteriormente, y sin implementar el SOM jerárquico debido al relativamente pequeño tamaño de los repositorios. Para permitir la evaluación, todos los algoritmos se han adaptado para trabajar con los mismos descriptores.

Para realizar los experimentos hemos diseñado un sistema similar a los descritos en [40] y [26], simulando los juicios del usuario en base a la clasificación disponible. Hemos seleccionado un cierto número de imágenes pertenecientes a cada categoría (de forma aleatoria y sin repetir ninguna), y se han proporcionado al sistema a modo de consulta, representando el concepto a buscar. En base a este concepto, y emulando la actuación de usuario, el sistema emite juicios automáticos sobre las 50 primeras imágenes de la ordenación que devuelve el algoritmo, marcando como positivas aquellas que se encuentran en la misma categoría que la consulta.

Para medir el rendimiento utilizamos el número de imágenes relevantes entre las primeras k imágenes de la ordenación producida por cada algoritmo (precisión en el valor de corte k). Presentamos los resultados para $k = 50$.

Bajo esta configuración en particular, las selecciones positivas no dependen del contexto y por lo tanto la decisión de si una imagen es o no relevante no cambia conforme progresa la búsqueda. Dado que la precisión medida en un cierto valor de corte k es proporcional al número de imágenes que se encuentran entre las k primeras imágenes de la ordenación, esta medida puede mejorarse significativamente si las selecciones positivas se posicionan siempre en este conjunto. El fuerte elitismo propuesto en nuestra implementación garantiza este tratamiento. Sin embargo, los enfoques con los que nos comparamos, tal y como fueron propuestos en los artículos originales, no lo utilizan, proporcionándonos una ventaja significativa con respecto a ellos. Por esta razón, hemos modificado los algoritmos originales con los que nos comparamos para que también implementen esta característica.

Para obtener datos representativos, cada técnica se ha evaluado sobre más de 1 000 búsquedas, promediando los resultados. Todos los algoritmos han sido sometidos a exactamente la mismas búsquedas y se ha forzado artificialmente a que siempre haya una imagen relevante entre las 50 primeras de la ordenación inicial. En las tablas I y II se muestra la

precisión obtenida en cada iteración para cada uno de los algoritmos comparados, y en las figuras 2 y 3 se muestran estos mismos resultados de forma gráfica.

TABLA I
PRECISIÓN EN EL VALOR DE CORTE 50 OBTENIDA PARA CADA UNO DE LOS ALGORITMOS PRESENTADOS, EN EL REPOSITORIO DE 1 508 IMÁGENES.

Iteración	Evolutivo	SOM	Movimiento del punto de consulta
1	0.347	0.327	0.338
2	0.472	0.455	0.443
3	0.543	0.521	0.500
4	0.590	0.564	0.534
5	0.622	0.597	0.566
6	0.647	0.622	0.589
7	0.667	0.645	0.613
8	0.683	0.663	0.630
9	0.695	0.678	0.648
10	0.706	0.692	0.662

TABLA II
PRECISIÓN EN EL VALOR DE CORTE 50 OBTENIDA PARA CADA UNO DE LOS ALGORITMOS PRESENTADOS, EN EL REPOSITORIO DE 5 476 IMÁGENES.

Iteración	Evolutivo	SOM	Movimiento del punto de consulta
1	0.253	0.201	0.249
2	0.384	0.335	0.357
3	0.470	0.407	0.419
4	0.527	0.459	0.457
5	0.568	0.495	0.494
6	0.602	0.526	0.521
7	0.627	0.551	0.548
8	0.648	0.574	0.570
9	0.664	0.594	0.590
10	0.678	0.611	0.608

Como puede apreciarse, el algoritmo genético interactivo presentado en este artículo proporciona mejores resultados sobre ambos repositorios.

VI. DISCUSSION

El rendimiento del algoritmo evolutivo que hemos presentado en este artículo descansa en la suposición de fuerte causalidad entre el fenotipo y el genotipo, lo que ha hecho posible trabajar con vecindades para enriquecer la población y basar la ordenación de las imágenes en la base de datos de acuerdo a su cercanía a los descendientes generados.

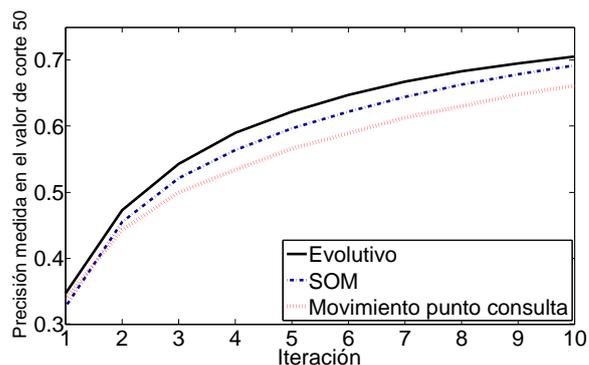


Fig. 2. Precisión en el valor de corte 50 obtenida con cada uno de los algoritmos considerados, en el repositorio de 1 508 imágenes

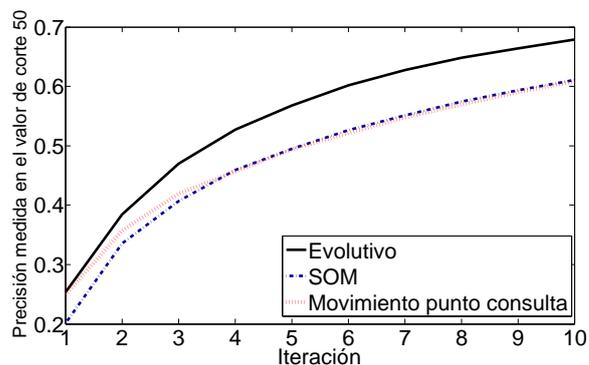


Fig. 3. Precisión en el valor de corte 50 obtenida con cada uno de los algoritmos considerados, en el repositorio de 5 476 imágenes

Esta técnica puede asociarse con un razonamiento intuitivo sencillo. El sistema asume que una selección positiva se basa en una fuerte similitud entre alguna característica de la imagen seleccionada y de la buscada. Si el algoritmo trabaja con descriptores de color, textura y forma, el sistema intentará averiguar cuál de estas características motivó la selección. Para ello, mostrará otra serie de imágenes, algunas con el mismo color que la seleccionada, y otras con la misma forma o la misma textura. Las operaciones de cruzamiento provocan que las características que son comunes a varias selecciones sean predominantes en la nueva población.

El enfoque que hemos presentado en este artículo no es únicamente válido para sistemas basados en selecciones binarias (relevante/no relevante). Por el contrario, puede ser fácilmente adaptado para funcionar con otro tipo de paradigmas. A modo de ejemplo, algunos sistemas de búsqueda permiten al usuario asignar un grado de similitud a cada imagen, mediante barras de desplazamiento. En este caso, el grado de similitud puede ser utilizado como función de evaluación, alterando las probabilidades de cada imagen de forma que aquellas con un mayor grado de similitud tengan mayor probabilidad de reproducirse.

Como trabajo futuro, sería interesante estudiar el efecto de utilizar distintas representaciones y técnicas de cruzamiento, y comparar el rendimiento obtenido con cada una de ellas en el contexto del problema.

REFERENCIAS

- [1] J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, MA, 1992.
- [2] H.-P. Schwefel, *Numerical Optimization of Computer Models*, Wiley, Chichester, 1981.
- [3] Gordon M., "User-based document clustering by re-describing subject descriptions with a genetic algorithm," *J Amer Soc Information Sci*, vol. 42, no. 5, pp. 311–322, 1991.
- [4] D.H. Kraft, F.E. Petry, B.P. Buckles, and T. Sadasivan, *Genetic algorithms for query optimization in information retrieval: relevance feedback*, chapter Genetic Algorithms and Fuzzy Logic Systems, p. 155173, World Scientific, 1997.
- [5] O. Cordón, F. Moya, and C. Zarco, "A new evolutionary algorithm combining simulated annealing and genetic programming for relevance feedback in fuzzy information retrieval systems," *Soft Computing*, vol. 6, pp. 308–319, 2002.
- [6] R. da S. Torres, A. X. Falcão, M. A. Goncalves, B. Zhang, W. Fan, E. A. Fox, and P. Calado, "A new framework to combine descriptors for content-based image retrieval," in *Fourteenth Conference on Information and Knowledge Management*, Bremen, Germany, 2005, pp. 335–336.
- [7] Sung-Bae Cho and Joo-Young Lee, "A human-oriented image retrieval system using interactive genetic algorithms," *IEEE Transactions on Systems, Man, and Cybernetics. Part A: Systems and humans*, vol. 32, no. 3, pp. 452–458, May 2002.
- [8] Shang-Fei Wang, Xu-Fa Wang, and Jia Xue, "An improved interactive genetic algorithm incorporating relevant feedback," in *Proceedings of the 4th International Conference on Machine Learning and Cybernetics*, Guangzhou, August 2005, pp. 2996–3001.
- [9] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*, McGraw Hill, 1983.
- [10] Yoshiharu Ishikawa, Ravishankar Subramanya, and Christos Faloutsos, "Mindreader: Querying databases through multiple examples," in *Proc. 24th Int. Conf. Very Large Data Bases, VLDB*, New York, USA, 1998, p. 433438.
- [11] I.J. Cox, M.L. Miller, T.P. Minka, T. V. Papatomas, and P.N. Yianilos, "The bayesian image retrieval system, pichunter: Theory, implementation and psychophysical experiments," *IEEE Transaction on image processing*, vol. 9, no. 1, pp. 20–37, 2000.
- [12] E. de Ves, J. Domingo, G. Ayala, and P. Zuccarello, "A novel bayesian framework for relevance feedback in image content-based retrieval systems," *Pattern Recognition*, vol. 39, pp. 1622–1632, 2006.
- [13] Markus Koskela, Jorma Laaksonen, and Erkki Oja, "Use of image subset features in image retrieval with self-organizing maps," in *Image and Video Retrieval: Third International Conference*, Dublin, Ireland, July 2004, pp. 508–516.
- [14] Xiang Sean Zhou and Thomas S. Huang, "Small sample learning during multimedia retrieval using biasmap," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001, pp. 11–17.
- [15] Dacheng Tao, Xiaou Tang, Xuelong Li, and Xindong Wu, "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1088–1099, 2006.
- [16] T. León, P. Zuccarello, G. Ayala, E. de Ves, and J. Domingo, "Applying logistic regression to relevance feedback in image retrieval systems," *Pattern Recogn.*, vol. 40, no. 10, pp. 2621–2632, 2007.
- [17] Wang Xiaoling and Xie Kanglin, "Application of the fuzzy logic in content based image retrieval," *Journal of Computer Science and Technology*, vol. 5, no. 1, pp. 19–24, 2005.
- [18] M. Arevalillo Herraes, M. Zacaes, X. Benavent, and E. de Ves, "A relevance feedback cbir algorithm based on fuzzy sets," *Signal Processing: Image Communication*, vol. 23, no. 7, pp. 490–504, August 2008.
- [19] C. Darwin, *The Origin of Species*, Dent Gordon, London, 1973.
- [20] H. Takagi, "Interactive evolutionary computation: fusion of the capabilities of ec optimization and human evaluation," *Proceedings of the IEEE*, vol. 89, no. 9, pp. 1275–1296, 2001.
- [21] Bernhard Sendhoff, Martin Kreutz, and Werner von See-len, "A condition for the genotype-phenotype mapping: causality," in *Proceedings of the 7th International Conference on Genetic Algorithms*, Bäck, Ed., San Francisco, 1997, pp. 2996–3001, Morgan Kauffman.
- [22] Qasim Iqbal and J.K. Aggarwal, "Combining structure, color and texture for image retrieval: A performance evaluation," in *16th International Conference on Pattern Recognition (ICPR)*, Quebec City, QC, Canada, August 2002, pp. 438–443.
- [23] M. Arevalillo Herraes, J. Domingo, and F. Ferri, "Combining similarity measures in content-based image retrieval," *Pattern Recognition Letters*, vol. In press, 2008.
- [24] Henning Müller, Wolfgang Müller, David McG. Squire, Stéphane Marchand-Maillet, and Thierry Pun, "Performance evaluation in content-based image retrieval: overview and proposals," *Pattern Recognition Letters*, vol. 22, no. 5, pp. 593–601, 2001.
- [25] N.J. Gunther and G. Beretta, "A benchmark for image retrieval using distributed systems over the internet: Birds-I," HPL technical report HPL-2000-162, HP Labs, Palo Alto, San Jose, USA, 2001.
- [26] Dacheng Tao, Xuelong Li, and Stephen Maybank, "Negative samples analysis in relevance feedback," *IEEE transactions on Knowledge and Data Engineering*, vol. 19, no. 4, pp. 568–580, 2007.
- [27] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag, Berlin, 2003.
- [28] E. de Ves, X. Benavent, G. Ayala, and J. Domingo, "Selecting the structuring element for morphological texture classification," *Pattern Analysis and Applications*, vol. 9, pp. 48–57, April 2006.
- [29] Michael J. Swain and Dana H. Ballard, "Color indexing," *Int. J. Comput. Vision*, vol. 7, no. 1, pp. 11–32, 1991.
- [30] G. Smith and I. Burns, "Measuring texture classification algorithms," *Pattern Recognition Letters*, vol. 18, no. 14, pp. 1495–1501, December 1997.
- [31] R. W. Connors, M. M. Trivedi, and C. A. Harlow, "Segmentation of a high-resolution urban scene using texture operators," *Computer Vision, Graphics, and Image Processing*, vol. 25, no. 3, pp. 273–310, March 1984.
- [32] R. Chellappa and S. Chatterjee, "Classification of textures using gaussian markov random fields," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. 33, pp. 959–963, 1985.
- [33] Y. Chen and E.R. Dougherty, "Gray-scale morphological granulometric texture classification," *Optical Engineering*, vol. 33, no. 8, pp. 2713–2722, August 1994.
- [34] G. Ayala and J. Domingo, "Spatial size distributions. applications to shape and texture analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 12, pp. 1430–1442, December 2001.
- [35] G. Ciocca and R. Schettini, "A relevance feedback mechanism for content-based image retrieval," *Information processing and management*, vol. 35, no. 1, pp. 605–632, 1999.
- [36] G. Ciocca and R. Schettini, "Content-based similarity retrieval of trademarks using relevance feedback," *Pattern Recognition*, vol. 34, no. 8, pp. 1639–1655, August 2001.
- [37] Jorma Laaksonen, Markus Koskela, and Erkki Oja, "Picson: Self-organizing image retrieval with MPEG-7 content descriptors," *IEEE Transactions on Neural Networks*, vol. 13, no. 4, pp. 841–853, 2002.
- [38] G. Giacinto, "A nearest-neighbor approach to relevance feedback in content based image retrieval," in *Proceedings of the 6th ACM international conference on Image and video retrieval (CIVR'07)*, Amsterdam, The Netherlands, 1993, pp. 456–463, ACM Press.