

# Aplicación de la pirámide irregular con nodos virtuales: segmentación de imagen usando información de textura y color

Antonio Bandera y Cristina Urdiales  
Dpto. Tecnología Electrónica, ETSI Telecomunicación,  
Universidad de Málaga, Campus de Teatinos, 29071

23rd October 2002

## RESUMEN

En este trabajo se presenta una estructura que permite segmentar una imagen en las regiones que la forman usando para ello información de textura y color. El proceso de segmentación se lleva a cabo usando una estructura piramidal irregular de enlace no adaptativo que se basa en una topología básica desarrollada por los autores y empleada con anterioridad para construir mapas topológico-métricos [1][22] o para segmentación de imagen usando información de nivel de gris [29]. Como principal novedad, la nueva estructura permite definir nodos virtuales, externos a la propia pirámide, cuya misión es enlazar nodos de un mismo nivel sin distorsionar la estructura 4-a-1 de la topología básica. De esta forma, la conectividad interna entre regiones se mantiene, evitándose los problemas que surgen cuando se emplean estructuras de enlace adaptativo y, sin embargo, se consigue que cada región de la imagen quede asociada a un único nodo de la estructura. Además de esta nueva estructura, este informe describe cómo se han mezclado las características de textura y color para conseguir segmentar una imagen, de manera que las regiones se crean inicialmente atendiendo a su color, caracterizado usando el espacio HSI [10], y sólo aquellas que no sean uniformes se caracterizan por textura, buscando regiones que sean uniformes en textura. La textura se caracteriza usando el método del LBP/C [18]. El algoritmo desarrollado no se basa, por tanto, en caracterizar cada pixel de la imagen con información de textura y color, o en llevar a cabo ambas segmentaciones, en color y textura, por separado, para luego fusionar de alguna forma los resultados obtenidos, sino que utiliza la propia estructura para decidir qué característica emplear en cada región. De esta manera, cada región se caracteriza sólo por su color o su textura, lo cual permite reducir considerablemente el tiempo empleado en segmentar la imagen (sobretudo, al no tener que caracterizar la textura asociada a toda ella).

## I. INTRODUCCIÓN

El proceso de segmentación de imágenes, usando para ello sus características de bajo nivel, es un paso intermedio vital para llevar a cabo posteriores tareas de más alto nivel que permitan, por ejemplo, el reconocimiento visual de los objetos presentes en la escena. Entre estas características de bajo nivel se encuentran el color, la forma o la textura. En particular, en este trabajo se usaran el color y la textura, características que son fundamentalmente utilizadas en conjunto para segmentar la imagen [14][2][25]. De manera general, existen dos opciones a la hora de usar ambas características: o bien se mezclan ambas características para formar el vector que caracteriza a cada pixel de la imagen [2][30], o bien se usan algoritmos de segmentación que trabajan con la textura o con el color por separado para, posteriormente, mezclar los resultados de ambas segmentaciones [14].

El uso de vectores de características que mezclen información de color y de textura es empleado en multitud de aplicaciones de análisis de imagen. Uno de los primeros algoritmos que emplea este tipo de vectores se debe a Rosenfeld *et al* [23], que caracteriza cada región usando las relaciones existentes entre los píxeles que están dentro de una determinada banda de color y también las relaciones entre píxeles de distintas bandas. Como descriptores de textura se usan estadísticos extraídos de las matrices de coocurrencia y de histogramas de diferencias. En esta misma línea de trabajo, Panjwani y Healey [19] usan un modelo de campos aleatorios de Markov que refleja estas relaciones entre píxeles. Jain y Healey [11] proponen emplear una representación multiescala que incluya características calculadas en cada banda espectral por separado, además de diferencias entre las distintas bandas, tratando así de reflejar el mecanismo que emplea el ojo humano para captar la información de color. En otras aproximaciones se usan sólo las relaciones espaciales dentro de cada banda. Así, por ejemplo, Caelli y Reye [5] proponen un método que extrae los parámetros de tres canales espectrales usando tres filtros multiescala. Paschos compara en [20] la efectividad de distintos espacios de color para describir texturas usando filtros de Gabor calculados por separado para cada banda. Mirmehdi y Petrou [16] han propuesto recientemente la segmentación de imagen en color usando relaciones entre el color y la distribución espacial de determinados patrones. En esta misma línea de trabajo, existen otro conjunto de algoritmos que usan gradientes de color [9].

Otra forma de mezclar color y textura en un único vector de características es dividir la señal de color en sus distintas componentes y procesarlas separadamente. En este grupo de métodos se encuadran los trabajos de Tan y Kittler [26], que extraen las características de textura de la componente de iluminación usando la transformada discreta del coseno y usan histogramas para reflejar la información de color, y los de Kyllönen y Pietikäinen [13] y Pietikäinen *et al*

[21], que usan la caracterización *Local Binary Patterns* (LBP) para la textura y parámetros extraídos de histogramas para el color. Estos métodos no procesan, sin embargo, vectores de color y de textura por separado, sino que emplean un vector que tiene información de ambas características.

Como se ha comentado, existe otra manera de enfocar la segmentación por color y textura, separando claramente ambos procesos y luego mezclando los resultados. En este sentido, Manduchi [14] propone usar un estimador bayesiano para mezclar los resultados obtenidos en los procesos de segmentación por color y por textura. La razón esgrimida para justificar la segmentación por separado es que las características de color y textura presentan comportamientos estadísticos muy distintos, por lo que es preferible emplear el algoritmo de segmentación que mejor se adapte a cada parámetro en particular. Además, también es frecuente que el tamaño de los vectores de características que mezclan ambas características sean excesivamente grandes, como ocurre en el algoritmo de Zöllner y Buhman [30], en el que se usa como vector asociado a cada región de la imagen un histograma de distribución de valores HSV para el color y un banco de filtros de Gabor con cuatro orientaciones equiespaciados en dos octavos para caracterizar su textura. Entre los métodos de segmentación que proponen trabajar independientemente con las características de bajo nivel, Dubuisson-Jolly y Gupta [8] proponen un método de segmentación de imagen aérea en el que los parecidos por color o textura son procesados por separado. La decisión final en segmentación se obtiene evaluando la certeza con que cada clasificador (textura y color, cada uno por separado) identifica una determinada región.

En este trabajo, el proceso de segmentación permite que se trabaje por separado con el color y con la textura. En este sentido, el método puede presentar cierta similitud con los trabajos de Dubuisson-Jolly y Gupta [8], aunque con notables diferencias, tanto en el método de caracterización utilizado como en la estructura y algoritmo de segmentación.

El resto del artículo se ha organizado como sigue. En el apartado II se introducen las estructuras jerárquicas piramidales de datos en las que se basa el método de representación propuesto. La estructura propuesta se describe con detalle en el apartado III. Finalmente, los apartados IV y V muestran los resultados y conclusiones obtenidas.

## II. ESTRUCTURAS PIRAMIDALES PARA SEGMENTACIÓN DE IMAGEN

La forma más eficiente de segmentar una imagen consistiría simplemente en agrupar los con-

juntos de píxeles conectados que tuvieran un color y textura similar en un único nodo, cuyas características serían la de este conjunto. Por lo tanto, el nodo resultante proporcionaría información sobre estas características pero, al contrario que el pixel, ese valor debería acompañarse por información sobre los píxeles asociados a él. En el caso normal de que la región de píxeles asociados al nodo no tenga una forma regular, su contorno o distribución deberá ser conocida. Toda esta información podría representarse mediante tablas o listas enlazadas [7], pero este tipo de estructuras son variables en el espacio y no resultan compatibles generalmente con la mayoría de las herramientas de procesamiento.

En este trabajo se emplea una estructura piramidal para representar la imagen. El concepto de pirámide fue inicialmente introducido por Tanimoto y Pavlidis en 1975 [27], y su desarrollo posterior puede seguirse revisando los libros de Rosenfeld [24], Tanimoto y Klinger [28] o Cantoni y Levaldi [6]. Básicamente, una pirámide consiste en un nivel inferior que contiene la imagen a analizar y un conjunto de niveles ubicados sobre éste, que son cada uno de ellos versiones reducidas del nivel inferior. Normalmente, la reducción es 4-a-1, teniendo cada nivel un cuarto de los nodos del inferior. Los niveles de la pirámide constituyen, por tanto, versiones a menor resolución de la imagen original. La complejidad que puede suponer el empleo de este tipo de estructuras se compensa por el hecho de que el procesado, en la mayoría de los casos, se puede acelerar si se usa la información contenida en los distintos niveles de la pirámide. Ésta será la principal justificación esgrimida para emplear este tipo de estructuras [12].

La Fig. 1 muestra la estructura de la pirámide clásica, utilizada comúnmente para la estimación de parámetros estadísticos como la media o la varianza [15]. En este tipo de estructuras, la detección de las distintas regiones que constituyen la imagen consiste en encontrar, en el nivel adecuado de la pirámide, un único nodo que represente a toda la región [12]. Dado que resulta prácticamente imposible que se cumpla esta premisa para toda región de la imagen (pues no siempre este tipo de descomposición 4-a-1 tiene que llegar a representar una región por un nodo), aparece en 1981 el concepto de pirámide de enlace adaptativo [3][4]. Así, si una pirámide básica es una secuencia de imágenes generadas a resolución progresivamente menor partiendo de la imagen inicial, en la pirámide construida usando el principio del Enlace Adaptativo (*pyramid linked*) las relaciones entre nodos de distintos niveles se redefinen iterativamente, cambiando los enlaces que se podrían establecer en la pirámide básica entre los cuatro nodos del nivel inferior -hijos- y el nodo que generan en el nivel superior -nodo padre-. Este proceso de reenlazado se lleva a cabo partiendo de los niveles más inferiores de la pirámide hacía los superiores, y se llevará a cabo de manera que los nodos de cada nivel se enlacen con los padres que presenten unas características más similares. Si se analiza el algoritmo que lleva a cabo esta redefinición,

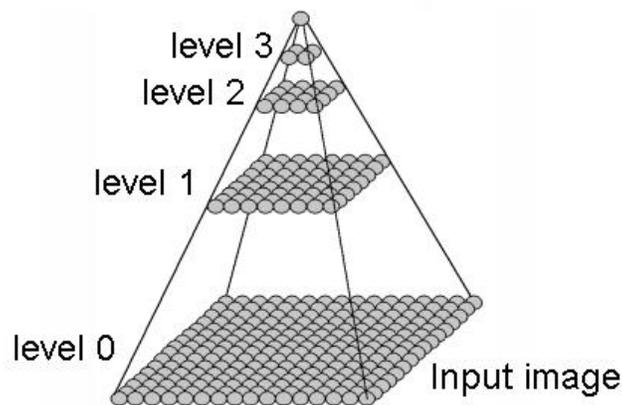


Figure 1: Estructura piramidal definida sobre una base cuadrada y con una reducción de un cuarto entre niveles consecutivos

el proceso consiste en permitir que todo nodo pueda elegir como padre el más parecido de entre los cuatro nodos más cercanos espacialmente del nivel superior. Este proceso se repite para cada nivel de la estructura hasta que no hay cambios de enlaces, y entonces pasa a ejecutarse en el nivel superior a éste. Después de que toda la pirámide ha sido estabilizada, los nodos de cualquier nivel estarán enlazados a una región irregular de celdas en la base de la estructura. Todas estas regiones tienen celdas con valores característicos similares, por lo que las distintas regiones que forman la imagen estarán agrupadas en un conjunto de nodos lo más reducido posible.

En la Fig. 2.a se muestran los niveles obtenidos empleando una estructura de enlaces 4-a-1, que implican que cada nodo -padre- del nivel  $l + 1$  tendrá un valor obtenido como la media de los 2x2 nodos inmediatamente inferiores en el nivel  $l$  -hijos-. Se puede observar como la estructura piramidal básica presenta un importante problema en su propia construcción, ya que, como se observa en la Fig. 3.a lo único que se hace para obtener cada nodo de una imagen de resolución menor es promediar valores característicos (niveles de gris) de los niveles inferiores, por lo que no se hace nada para que dicho nodo no tenga asociado, simultáneamente, celdas pertenecientes a distintas regiones. La Fig. 2.b muestra las distintas imágenes obtenidas tras el proceso de reenlazado, en las que se puede notar como los nodos de los distintos niveles conservan un valor característico definido, lo cual indica, como se observa en la Fig. 3.b, que ahora cada nodo está enlazado a una región de forma irregular en la base pero con celdas de valor característico homogéneo (*irregular partitioning*).

El principal problema del proceso de reenlazado adaptativo es que las regiones en la base, enlazadas a un único nodo en un nivel superior de la estructura, no tienen por qué estar conec-

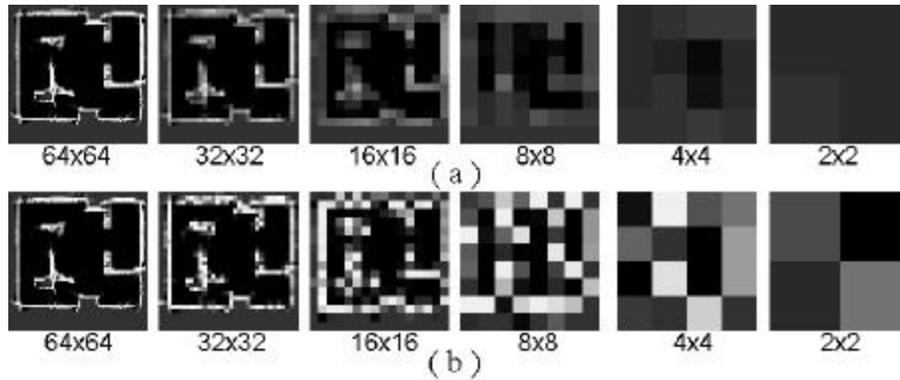


Figure 2: Distintos niveles disponibles en: a) una pirámide básica; y b) una pirámide adaptativamente estabilizada.

tadas [17]. La existencia de regiones no conectadas dificulta extraordinariamente el proceso de detección de objetos, objetivo fundamental para el que generalmente se lleva a cabo la segmentación de la imagen. Los niveles de la pirámide que se muestran en las Figs. 4.b y 4.c han sido obtenidos usando la estabilización adaptativa clásica. En la Fig. 4.d se puede observar cómo la región asociada al nodo marcado con una  $\mathbf{X}$  en la Fig. 4.b no está conectada internamente. Para evitar esta falta de conectividad interna, los cambios de enlaces que afecten a la conectividad de las regiones pueden ser penalizados, haciendo que el proceso de reenlazado no sólo tenga en cuenta información acerca de las características del nodo (color, ...), sino también información espacial. De esta forma, la medida de parecido entre probabilidades se pondera por un factor de distancia que depende de la proximidad entre nodos padre e hijo. La Fig. 5 muestra el conjunto de padres a los que se puede enlazar un determinado nodo hijo, siendo el factor  $d_1$  de valor  $\sqrt{2}/2$ , el factor  $d_2$  igual a  $\sqrt{10}/2$  y el factor  $d_3$  de valor  $3 \cdot \sqrt{2}/2$ . El nodo hijo se enlazará al padre más cercano, estando la distancia entre dos nodos definida por el producto entre la diferencia de sus valores característicos y el factor de ponderación espacial mencionado. Existen otras soluciones, que implican estudiar el conjunto de grafos generados por la representación multirresolución de la imagen [17].

### III. PIRÁMIDES IRREGULARES INTERNAMENTE CONECTADAS CON NODOS VIRTUALES

En este trabajo se propone un nuevo algoritmo para reenlazar los nodos entre niveles y así segmentar la imagen en las regiones que la integran. De esta forma se pretende obtener una segmentación multirresolución de la imagen, en la que las regiones no tengan una forma definida, pero en la que el proceso de reenlazado no tenga que ser interactivo, como se propone en la

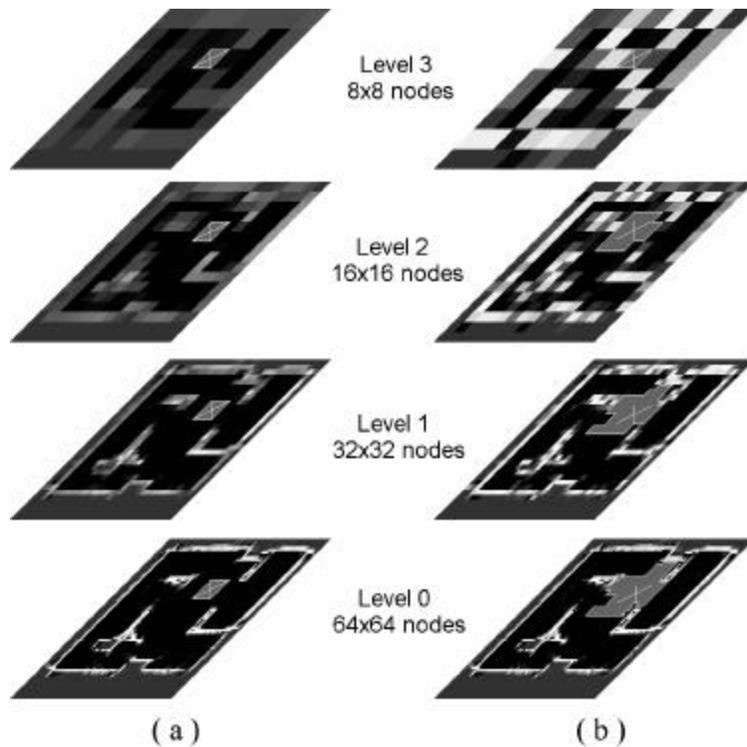


Figure 3: Nodos enlazados al nodo (2,4) del nivel 8x8 de: a) la pirámide básica; y b) la pirámide adaptativamente enlazada.

pirámide de enlace adaptativo. El algoritmo propuesto detecta un conjunto inicial de regiones homogéneas, usando un paso de inicialización que trabaja desde los niveles inferiores de la pirámide hacia los superiores. Una vez que la pirámide ha sido inicializada, se lleva a cabo un paso de mezclado que, trabajando desde los niveles superiores de la estructura hacia los inferiores, trata de crecer las regiones homogéneas detectadas o de mezclar aquellas que, estando en contacto, presentan características similares. Las principales ventajas del algoritmo son que los padres o nodos raíces, auténticas semillas del proceso de segmentación, no son elegidos al azar, y que la pirámide se estabiliza en un único paso, lo cual incrementa la velocidad del proceso de segmentación de manera significativa.

Los diferentes pasos del algoritmo de segmentación son:

1. Inicialización de la estructura. Se genera una pirámide 4-a-1 en la que la imagen HSI ocupa el nivel inferior de la estructura. Cada nivel  $l$  de la pirámide es una imagen reducida con un cuarto de las celdas del nivel inmediatamente inferior. Cada celda  $(x, y, l)$  de la pirámide presenta nueve campos:
  - Intensidad,  $I(x, y, l)$ . Si una celda es homogénea,  $I(x, y, l)$  es igual a la media de

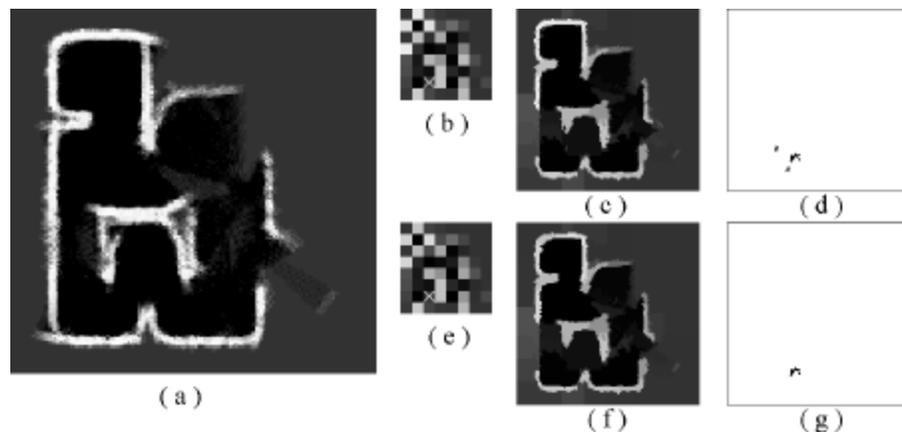


Figure 4: a) Grid uniforme de 256x256 nodos; b) nivel 8x8 de la pirámide estabilizada; c) regiones en la base de la pirámide asociadas a los nodos en b); d) región asociada al nodo marcado en b); e) nivel 8x8 de la pirámide estabilizada con peso por proximidad física; f) regiones en la base de la pirámide asociados a los nodos en e); y g) región asociada al nodo marcado en e).

los valores de intensidad de las cuatro celdas que están en el nivel inmediatamente inferior.

- *Hue*,  $H(x, y, l)$ . Si una celda es homogénea,  $H(x, y, l)$  es igual a la media de los valores de *hue* de las cuatro celdas que están en el nivel inmediatamente inferior.
- *Saturación*,  $S(x, y, l)$ . Si una celda es homogénea,  $S(x, y, l)$  es igual a la media de los valores de saturación de las cuatro celdas que están en el nivel inmediatamente inferior.
- *Área*,  $A(x, y, l)$ . En principio es igual a la suma de las áreas de las cuatro celdas del nivel inferior que generan este nodo. De cualquier forma, medirá siempre el área en píxeles que ocupan los nodos hijos asociados al nodo  $(x, y, l)$ .
- *Enlace al padre*,  $(X, Y)_{(x, y, l)}$ . Si la celda es homogénea, el valor de enlace al padre de

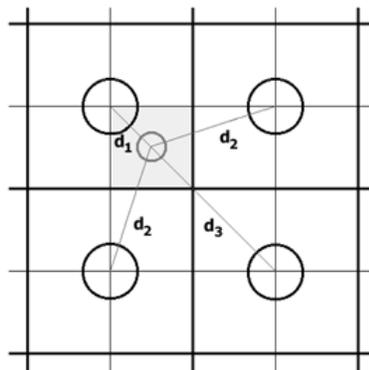


Figure 5: Relaciones entre un nodo hijo y sus cuatro posibles nodos padres.

las cuatro celdas del nivel inferior que dan forma al nodo  $(x, y, l)$  apunta a ese mismo nodo. Como se estudiará posteriormente, un nodo de la pirámide puede apuntar a un nodo virtual, en cuyo caso el campo  $X_{(x,y,l)}$  será igual a -1, y el valor de  $Y_{(x,y,l)}$  marca la posición, dentro de la lista de nodos virtuales, donde está el nodo virtual padre del nodo en cuestión.

- Homogeneidad en color,  $H(x, y, l)$ . Si los cuatro nodos que dan forma al nodo  $(x, y, l)$  tienen un color uniforme y son homogéneos en color, el valor de  $H(x, y, l)$  es igual a uno. En otro caso, es igual a cero. El valor que determina esa uniformidad en color,  $UMBHOM$ , es uno de los parámetros de ajuste del algoritmo.
- Homogeneidad en textura,  $HT(x, y, l)$ . Si los cuatro nodos que dan forma al nodo  $(x, y, l)$  tienen la misma textura y son homogéneos en textura, el valor de  $HT(x, y, l)$  es igual a uno. En otro caso es igual a cero. El valor de umbral de homogeneidad en textura es variable por nivel, siendo su expresión:

$$(UMBHOMTEX * TamNiveles[l + 1]/TAMBase), \quad (1)$$

donde  $TamNiveles[l + 1]$  es el tamaño del nivel  $l+1$  y  $TAMBase$  el tamaño de la base de la estructura piramidal. Este umbral determina la homogeneidad en textura y es uno de los parámetros de ajuste del algoritmo.

- Clase,  $C(x, y, l)$ . Una vez configurados los enlaces en la estructura, este campo permite determinar la región a la que pertenece el nodo en cuestión.

El paso de inicialización debe dividir la imagen en regiones de color o textura uniforme, pero atendiendo a una cuadrícula propia en algoritmos de *splitting*. Para ello, se obliga a los nodos de la base a ser homogéneos en color (de hecho lo son, aunque con un área igual a un pixel), y se generan los niveles superiores a la base, de manera que si los cuatro nodos que crean en la estructura 4-a-1 el nodo del nivel superior cumplen que: i) son homogéneos en color y ii) tienen un color parecido, entonces pasan a generar un nodo homogéneo, con cuatro veces más área y un color igual a la media de color de los cuatro nodos que lo generan. El color, por lo tanto, sube desde la base hasta el nivel superior de la pirámide. En la Fig. 6 se muestra una imagen original y las clases asociadas a los nodos de los distintos niveles de la pirámide. La segmentación de la imagen en el nivel de 256x256 nodos muestra el efecto de *splitting* comentado anteriormente.

En cuanto al análisis de la textura, y dado que no se puede caracterizar correctamente la textura de un bloque de píxeles de tamaño menor de 8x8 ó 16x16 [18], se ha optado por seleccionar un nivel de la pirámide en el que se lleva a cabo la caracterización en

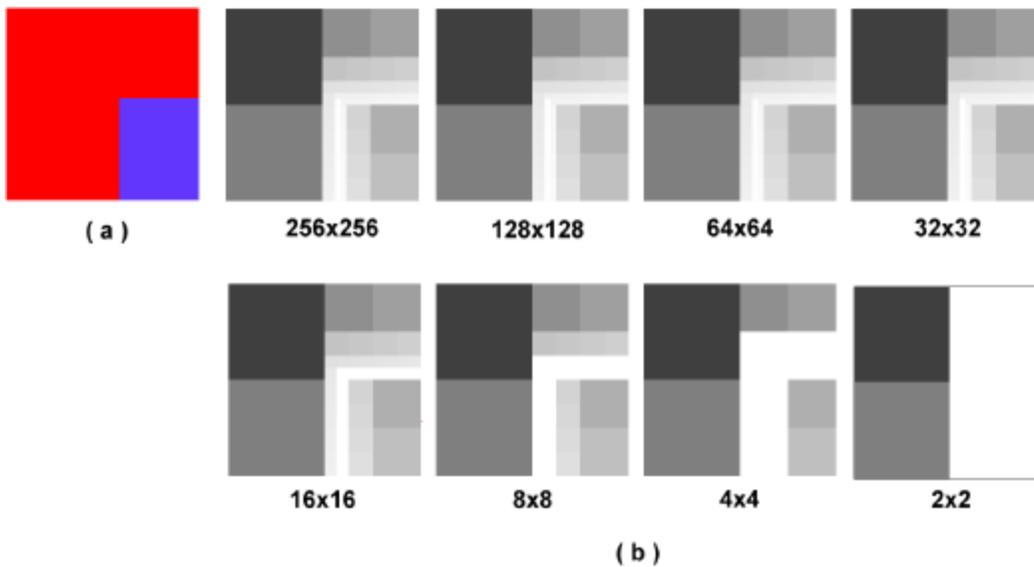


Figure 6: a) Imagen de 256x256; y b) campo *Clase* de los nodos de los niveles de la estructura.

textura,  $NivMarTex$ , generándose para cada nodo un histograma LBP/C [18]. En el nivel  $NivMarTex+1$  se estudiará si existen nodos homogéneos, que serán aquellos que cumplan que los cuatro nodos del nivel inferior relacionados con él tengan la misma textura. Para medir la diferencia en textura se ha empleado el estadístico  $G$ , tal y como se propone en los trabajos de Ojala y Pietikäinen [18]. Si un nodo del nivel  $NivMarTex+1$  resulta ser homogéneo, deberá de bajar dicha información a todos los nodos de los niveles inferiores que están relacionados con él, hasta llegar a la misma base de la estructura. Dado que en los niveles inferiores de la estructura no existe un estudio de homogeneidad en textura, la resolución que ofrece este primer paso del proceso de segmentación es bastante mala. En la Fig. 7 se muestra una imagen original y la segmentación, por niveles, en función de la textura.

Finalmente, se debe apreciar una característica importante del proceso de inicialización, y es que el proceso de segmentación por color se hereda desde los niveles inferiores de la pirámide, por lo que al llegar al nivel  $NivMarTex$  puede que existan muchos nodos que sean homogéneos en color y, por tanto, no será necesario estudiar su textura. Esto puede ahorrar al algoritmo el cálculo del valor de LBP/C de muchos píxeles de la imagen, con el consiguiente aumento en la velocidad de procesado.

2. Aumento de resolución en la segmentación por textura. Como se expuso en el punto anterior del algoritmo de segmentación, la resolución que ofrece la segmentación por textura es bastante pobre. Este paso del proceso de segmentación tratará de mejorar esta resolución, aumentando el tamaño de las regiones de textura uniforme. Para ello unirá, si sus tex-

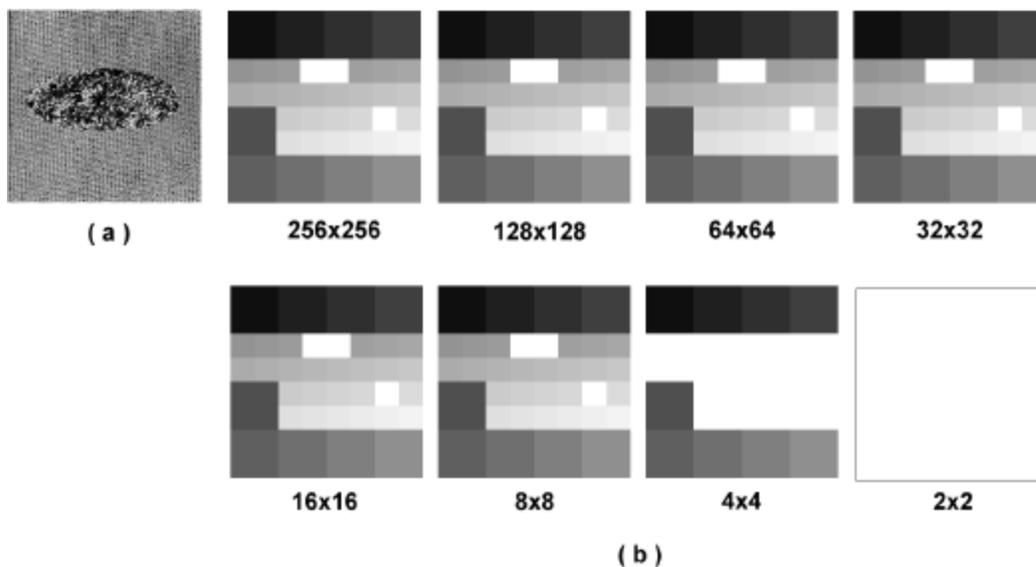


Figure 7: a) Imagen de 256x256; y b) campo *Clase* de los nodos de los niveles de la estructura.

turas son similares, nodos del nivel *NivMarTex* que no tienen padre con nodos vecinos del mismo nivel que sean homogéneos en textura (y, por tanto, tengan padre). El efecto de este aumento de resolución se puede apreciar en la Fig. 8.

3. Hermanamiento. Este paso lleva a cabo el mezclado de nodos que presentan el mismo color o textura y están ubicados en el mismo nivel de la estructura piramidal. Es un proceso que se lleva a cabo desde los niveles superiores hasta los inferiores, por lo que su implementación algorítmica se lleva a cabo en dos fases: i) desde el nivel 2x2 de la estructura piramidal hasta el nivel *NivMarTex*+1, en la cual se trabaja con color y

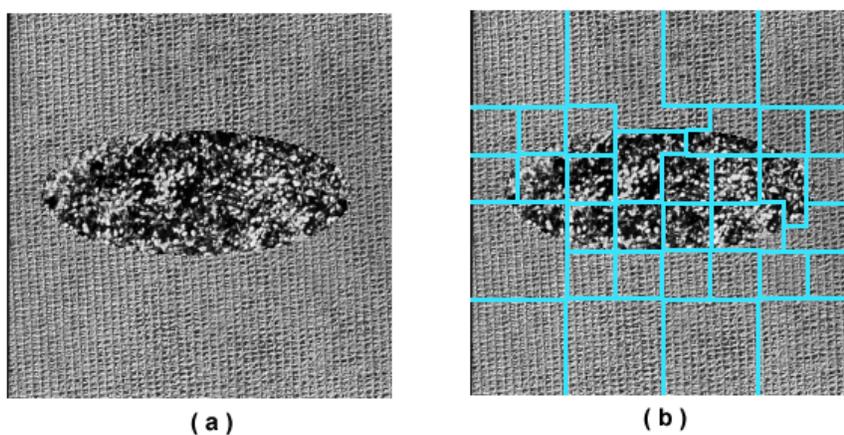


Figure 8: a) Imagen de 256x256; y b) división en regiones tras el paso de aumento de resolución en la segmentación por textura.

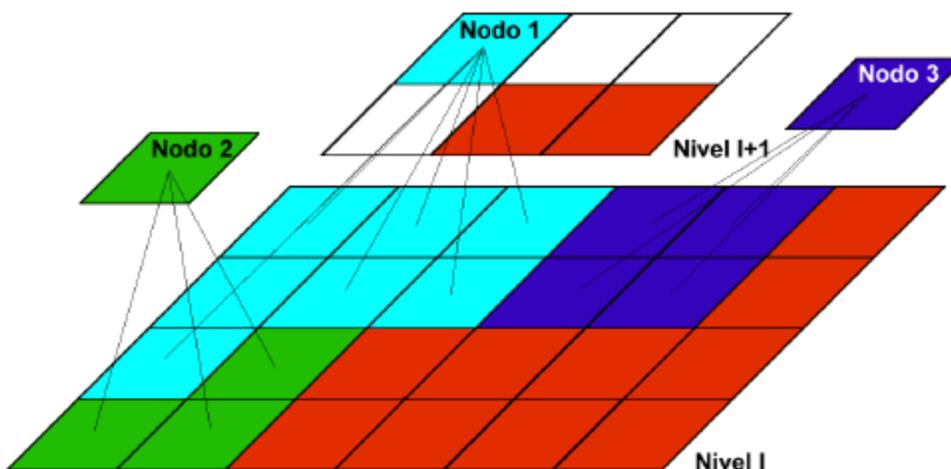


Figure 9: Ejemplo del paso de hermanamiento: el nodo 1 ha sido definido por cuatro nodos homogéneos, y su prole aumenta al enlazársele nodos huérfanos contiguos; el nodo 2 es un nodo virtual, externo a la estructura piramidal, que surge debido a que los tres nodos que se le enlazan no tienen padre, ni en el proceso de inicialización ni en el de hermanamiento; el nodo 3 es también un nodo virtual, que sirve de padre a cuatro nodos vecinos que no pueden tener padre en el proceso de inicialización debido simplemente a su posición en el espacio.

textura, y ii) desde el nivel *NivMarTex* hasta el nivel *NivAumRes*, en la cual se trabaja sólo con color. El nivel *NivAumRes* se ubica generalmente un nivel por encima de la base de la estructura piramidal, y en ellos no se lleva a cabo este proceso de hermanamiento porque la información de homogeneidad de un bloque de 2x2 es poco relevante, siendo el proceso de hermanamiento en estos dos últimos niveles de la pirámide distinto que en los niveles superiores.

Es en este paso del algoritmo de segmentación en el que aparecen los nodos virtuales. Con este tipo de nodos se trata de paliar la situación que aparece cuando existen dos o tres nodos con la misma textura situados debajo de un nodo que, por lógica, no será homogéneo, y que, por tanto, no puede ser el padre de este conjunto de nodos. El nodo virtual será el nodo padre o raíz al que quedará enlazado el conjunto de nodos. También será la solución para enlazar nodos que no compartirían nunca padre, pero que están en contacto físico y presentan similares características de color y textura. En la Fig. 9 se muestran estos dos ejemplos y la aparición de los nodos virtuales. El conjunto de nodos virtuales se almacenan en un vector de tamaño finito, que deberá estar inicializado a un valor suficiente grande. Dado que esta implementación podría llevar a fallos en el programa, se podría sustituir por una lista enlazada, lo que, sin embargo, ralentizaría la ejecución del algoritmo.

También se resuelve en esta fase el problema de unir regiones que no están en contacto en

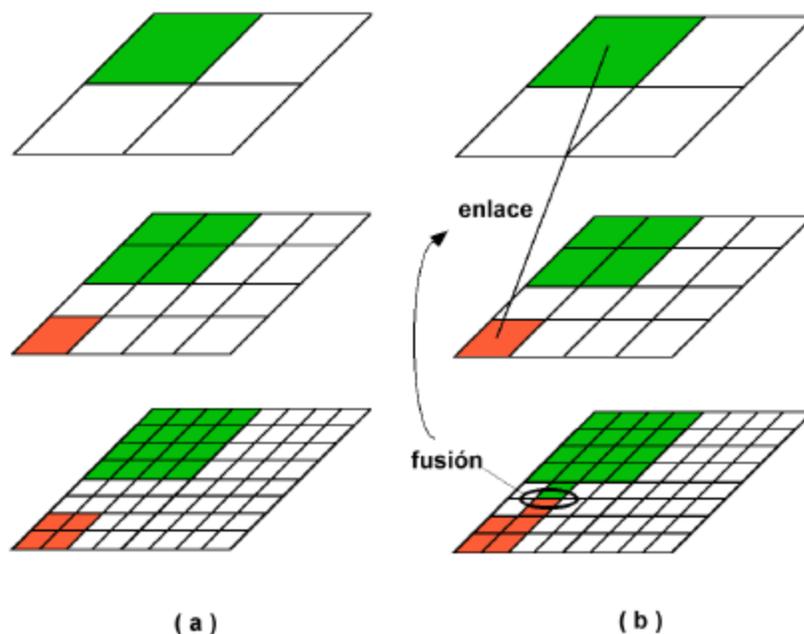


Figure 10: a) Estructura de enlaces tras el paso de inicialización; y b) la fusión marcada entre nodos origina la aparición de un nuevo enlace, que afecta a los niveles superiores.

niveles superiores de la estructura pero que si lo están en los niveles inferiores. Resolver este problema obliga a complicar el algoritmo que lleva a cabo esta fase, pero permite evitar posteriores procesos de fusión, que ralentizarían la segmentación. En la Fig. 10 se muestra esquemáticamente una situación en la que se debe llevar a cabo la fusión.

4. Clasificación. En la fase de clasificación se numeran los nodos raíces o padres, que darán lugar a las distintas regiones, para después propagar su valor hasta la base de la estructura. Serán nodos raíces todos los nodos virtuales y los nodos homogéneos no enlazados a otro nodo de la estructura piramidal. La Fig. 11 muestra el resultado de segmentar una imagen con regiones homogéneas en color y en textura. En dicha figura se aprecia cómo aparecen cinco clases (la región en blanco es zona no asignada a ninguna clase), entre ellas una de sólo cuatro píxeles (en negro, muy pequeña) y otra de mayor tamaño (también en negro) que no son realmente clases distintas presentes en la imagen. También se aprecia en la figura que la resolución no es máxima ni en las regiones uniformes en color ni en las uniformes en textura. Esto se debe a que el paso de hermanamiento no se llevo a cabo en los dos últimos niveles de la estructura piramidal (el nivel *NivAumRes* se fijo en dos). Para tratar de subsanar este problema se implementa un último paso en el algoritmo.

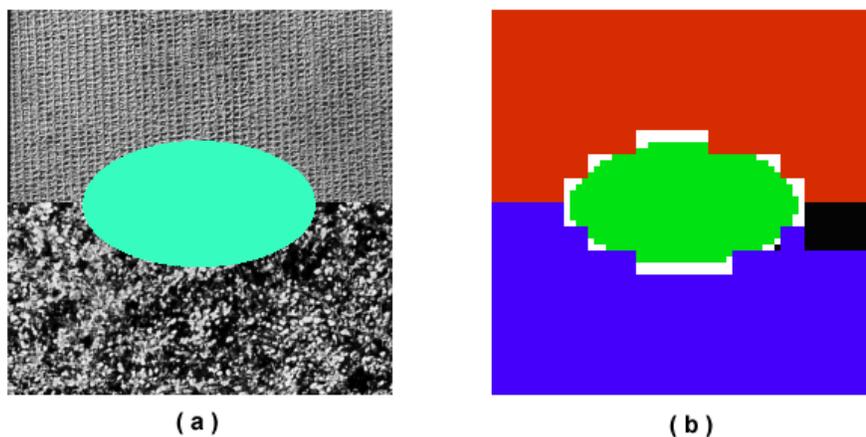


Figure 11: a) Imagen de 256x256 píxeles; y b) segmentación de a).

5. Aumento de resolución. Como se ha comentado, se puede aumentar la resolución de la segmentación obtenida tras aplicar los pasos anteriores. La justificación de no llevar el proceso de hermanamiento hasta la base (cosa que, en cualquier caso, se puede hacer simplemente haciendo  $NivAumRes$  igual a cero), se debe a que en estos últimos niveles aparecen muchas regiones nuevas de tamaño muy pequeño. Estas pequeñas regiones se asocian, generalmente, al borde entre regiones o a la presencia en la imagen de objetos con color y textura no definidos. Para evitar una excesiva descomposición de la imagen se opta por tratar de manera especial estas regiones.

El método de aumento de resolución trata de unir los nodos no clasificados de los niveles inferiores de la estructura al padre del nivel superior con un nivel de gris más parecido. Se usa el nivel de gris porque permite una clasificación rápida y, dado que estos puntos están en la frontera entre regiones y no tienen porqué pertenecer claramente a una determinada región, el algoritmo se decanta por la rapidez frente a la mejor calidad que podrían ofrecer otras características como el color o la textura. La Fig. 12.b muestra el resultado obtenido finalmente tras segmentar la Fig. 12.a.

Esta estructura presenta determinadas ventajas, aparte de las ya comentadas de selección de padres y no iteratividad del proceso de reenlazado. En primer lugar, no es necesario ningún preprocesado para forzar la conectividad de las regiones, lo cual permite reducir el tiempo de ejecución del algoritmo. Por otra parte, no es necesario seleccionar un nivel de la estructura para posteriormente llevar a cabo el proceso de segmentación. El hecho de obligar al usuario a seleccionar un determinado nivel de trabajo presupone la existencia de un conocimiento previo sobre el conjunto de regiones, e implica una cierta uniformidad en el tamaño de las regiones que

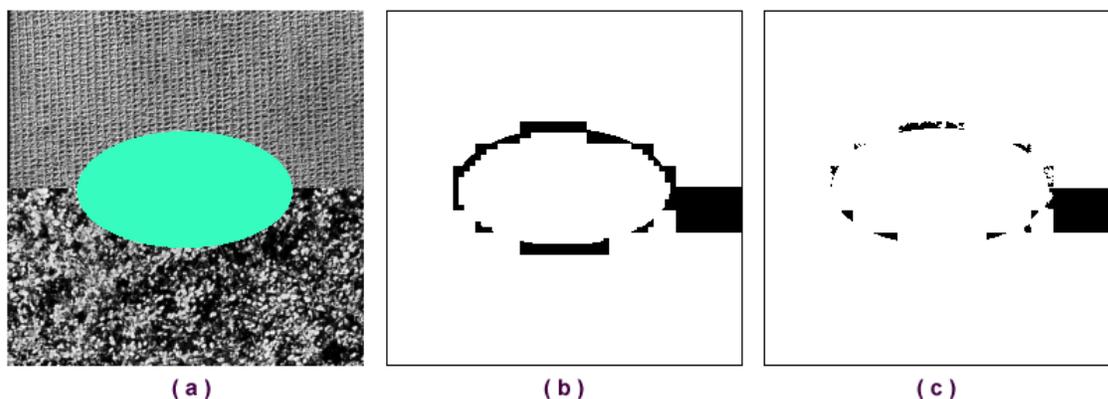


Figure 12: a) Imagen de 256x256 píxeles; b) error en la segmentación de a) antes del paso de aumento de resolución; y c) error final en la segmentación de a).

forman dicho espacio. En este caso, el algoritmo realiza, de manera automática, la búsqueda de los potenciales objetos en distintos niveles de la estructura tridimensional.

#### IV. RESULTADOS

En esta sección se presentan algunos resultados obtenidos de la aplicación del algoritmo, en todos los casos el procesador empleado para llevar a cabo las pruebas es un PC Pentium III a 933 MHz con 128 Mb de RAM.

En la Fig. 13 se muestra la imagen original que se va a utilizar para describir el algoritmo paso por paso. En esta descripción pormenorizada de funcionamiento se hará un seguimiento en la evolución de los principales campos de la estructura (homogeneidad en color y textura y el campo clase).

El primer paso del algoritmo es el de inicialización. En este paso se realiza un mezclado de regiones desde la base hasta el nivel superior de la pirámide cuando se trabaja en color, o desde el nivel  $NivMarTex$  hasta dicho nivel superior cuando se trabaja en textura. El campo homogeneidad en color,  $HC(x, y, l)$ , se actualiza, por tanto, desde la base, mientras que el campo homogeneidad en textura,  $HT(x, y, l)$ , se actualiza desde un nivel superior. Sin embargo, si se declara que un nodo es homogéneo en textura, todos los nodos de la estructura hasta la base, que deberían estar enlazados por su posición espacial a este nodo, son declarados como homogéneos en textura. Si se usa un valor de  $NivMarTex$  igual a 4 (nivel de 16x16 nodos), tras el paso de inicialización de la Fig. 13, los campos  $HC(x, y, l)$  y  $HT(x, y, l)$  de la estructura son los

mostrados en la Fig. 14.

En la Fig. 14 se puede observar que raramente existen regiones pequeñas homogéneas en color dispersas por los niveles inferiores de la estructura piramidal, apareciendo sólo estas pequeñas regiones alrededor de la región central, que es homogénea en color. Esto se debe a que al marcar como homogénea en textura una determinada región, los nodos unidos a ésta son homogéneos en textura y no homogéneos en color. Si no se hiciera este análisis de homogeneidad en textura, aparecería una mayor cantidad de nodos homogéneos en color, como muestra la Fig. 15.

Por tanto, el proceso de inicialización trabaja en color con toda la estructura y en textura sólo desde el nivel de  $8 \times 8$  nodos (dado que *NivMarTex* vale cuatro, es en ese nivel, de  $16 \times 16$  nodos, en el que se caracteriza la textura, y el siguiente nivel, de  $8 \times 8$  nodos, el primero en que se puede hablar de nodos homogéneos en textura). Si se baja el valor de *NivMarTex* a tres, el tamaño reservado en memoria es excesivamente grande, debido a tener que reservar espacio para  $32 \times 32 + 16 \times 16 + \dots + 1 \times 1$  histogramas LBP/C de textura. Esto impide el correcto funcionamiento del sistema en el equipo descrito.

El siguiente paso del algoritmo consiste en estudiar si es posible mezclar aquellos nodos del nivel *NivMarTex* que no tienen padre en textura, pero han sido caracterizados, con algún nodo de este nivel que sí presente padre en textura. Este paso es previo al posterior paso de mezclado desde los niveles superiores hasta la base, y se lleva a cabo de manera independiente porque cumple un requisito especial: los nodos que se van a mezclar no tienen porqué ser homogéneos en textura. Efectivamente, los nodos del nivel *NivMarTex* no pueden ser homogéneos en textura pues en este nivel la textura se puede caracterizar, pero son necesarios cuatro nodos caracterizados en textura para determinar si un nodo es homogéneo, por lo que la homogeneidad se fija en el nivel superior a éste. Sin embargo, si que se pueden comparar los histogramas

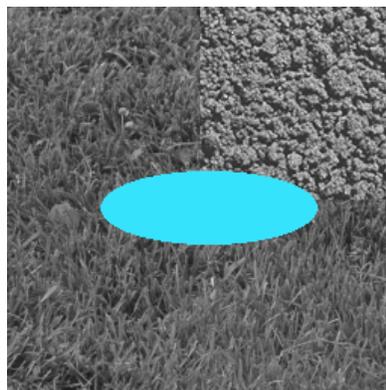


Figure 13: Imagen de  $256 \times 256$  píxeles.

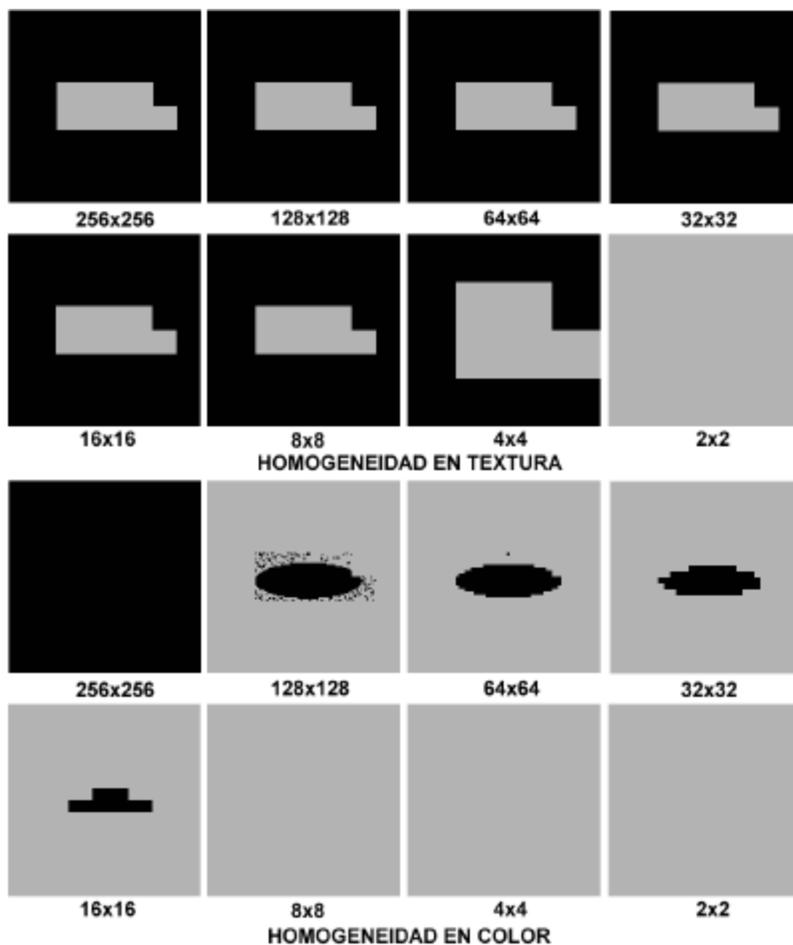


Figure 14: Campos homogeneidad en textura y en color tras aplicar el paso de inicialización a la Fig. 13

LBP/C de estos nodos. Para no complicar la función de mezclado posterior se implementó esta rutina. Su efecto ya se ha descrito al analizar el algoritmo y sólo se destacará en este apartado que, obviamente, los nodos de *NivMarTex* que encuentran padre en esta rutina pasan a ser homogéneos en textura y los nodos de niveles inferiores quedan correctamente enlazados a ellos. Al segmentar la imagen de la Fig. 13, el aumento en resolución define las regiones homogéneas en textura que se muestran en la Fig. 16.

El siguiente paso es el de mezclado, partiendo desde el nivel superior y hasta los inferiores. Este proceso cambia los enlaces entre nodos, tratando de unir toda región uniforme a un único nodo. Para ello se llevan a cabo tres procesos: i) el enlace de nodos homogéneos de un nivel a nodos homogéneos de un nivel superior con los que estén en contacto; ii) el enlace de nodos homogéneos del mismo nivel entre sí; y iii) el enlace de nodos padres entre sí al aparecer nodos homogéneos en contacto en un determinado nivel con las mismas características. Toda esta causística se

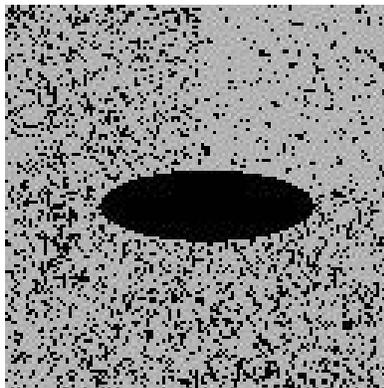


Figure 15: Campo homogeneidad en color del nivel 128x128 píxeles si no se lleva a cabo un estudio de homogeneidad en textura.

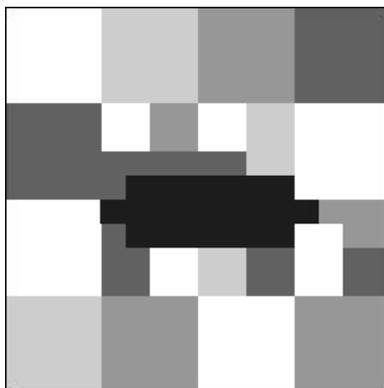


Figure 16: Regiones de textura homogénea tras el segundo paso del proceso de segmentación (el hecho de que dos regiones compartan brillo no quiere decir que sean la misma, pues las regiones detectadas están conectadas interiormente).

recoge en una misma función, que recorre los nodos homogéneos de cada nivel y estudia si tiene en contacto otros nodos homogéneos. Además, para hacer frente a situaciones de conexión que no puede reflejar la estructura piramidal por su fuerte jerarquía de enlaces 4-a-1, se introduce el concepto de nodo virtual, que se mantiene al margen de la pirámide. La Fig. 17 refleja el proceso de enlazado del nivel de 8x8 nodos. La Fig. 17.a muestra como se encuentra dicho nivel antes del proceso de hermanamiento, apreciándose el fenómeno de *splitting* y cómo se agrupan los nodos homogéneos (los cuadros del mismo color muestran una agrupación). La Fig. 17.b refleja como se han clasificado los nodos del nivel 4x4 tras el proceso de hermanamiento. Se aprecia como han aparecido dos clases, unidas a dos nodos virtuales (el (-1,0) y el (-1,1)). Si la vecindad-8 empleada se nombra como se muestra en la Fig. 17.c, los enlaces a padre de cada nodo van cambiando como refleja la Fig. 17.d. Se puede destacar como la estructura de enlaces se deteriora excesivamente, produciéndose una agrupación hacia los nodos (0,0) y (0,2), que quedan como representantes de las dos zonas de textura distinta en este nivel. Este deterioro

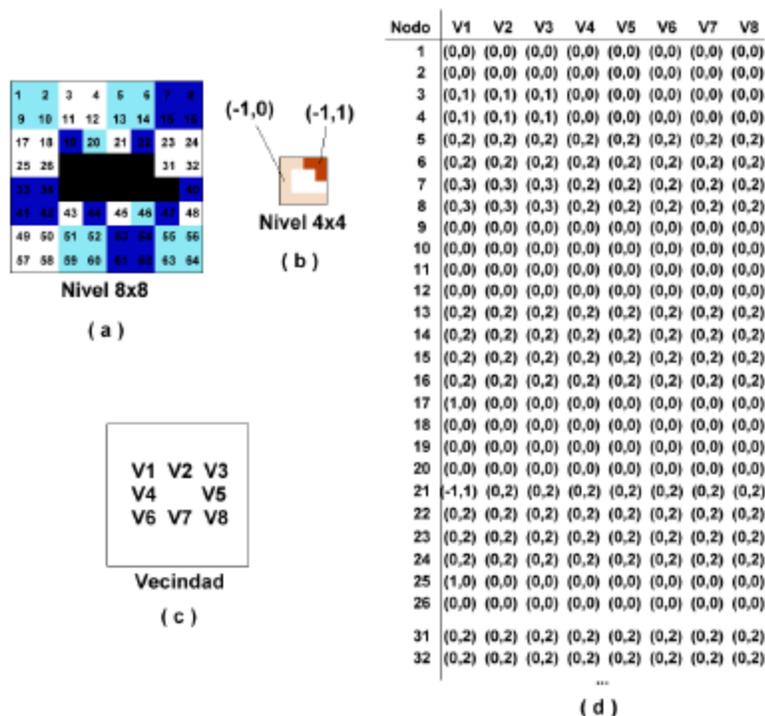


Figure 17: a) Nivel 8x8 antes del proceso de hermanamiento; b) nivel de 4x4 tras el paso de hermanamiento; c) vecindad-8 y d) enlace de los nodos del nivel 8x8 tras estudiar las distintas vecindades.

no afecta a la segmentación en sí, pues la región a la que se asocian los nodos es la misma, pero si que supone acabar prácticamente con toda la estructura de enlaces.

La Fig. 18.a muestra la división en regiones obtenida tras el paso de hermanamiento. Esta figura se ha obtenido tras llevar a cabo el proceso de clasificación, por lo que refleja la división final de la Fig. 13. Sólo restaría por ejecutar el aumento de resolución. La Fig. 18.b presenta el resultado final de segmentación obtenido tras llevar a cabo este último paso del proceso.

## V. CONCLUSIONES

Básicamente, el método de segmentación de imagen propuesto trata de compaginar la información de textura, extraída a una resolución necesariamente baja, con la de color, que se puede obtener a una resolución máxima de un píxel. Dado que el color puede usarse para segmentar desde los niveles inferiores de la estructura, sólo las regiones que no presentan un color no homogéneo serán estudiadas en textura. El tiempo de segmentación, para una imagen de 256x256 píxeles y usando el equipo descrito en el apartado de resultados, es de aproximadamente un

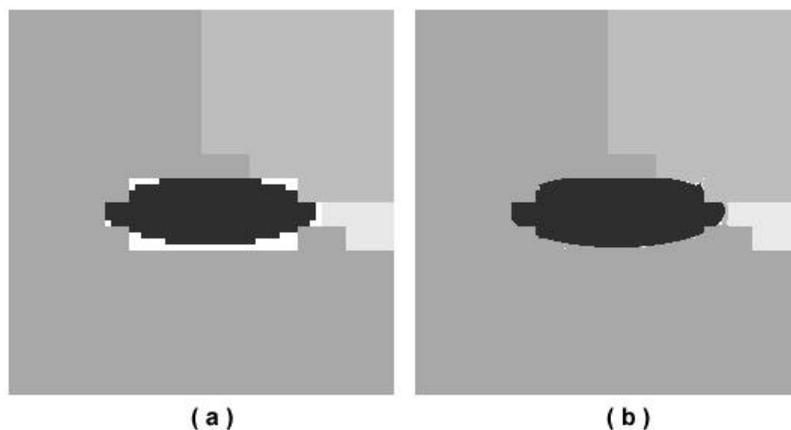


Figure 18: a) Imagen segmentada tras el paso de hermanamiento; y b) imagen segmentada tras el paso de aumento de resolución.

segundo. Quedan, sin embargo, muchas cuestiones por mejorar, tanto a la hora de mejorar la velocidad de ejecución, fácilmente mejorable simplemente cambiando opciones de compilación, como de la filosofía del proceso de hermanamiento que, como ya se ha descrito, tiende a fusionar los nodos de un nivel en un único nodo-padre del nivel superior, destrozando la estructura de enlaces creada.

## References

- [1] Bandera, A., C. Urdiales y F. Sandoval. A hierarchical approach to grid-based and topological map integration for autonomous indoor navigation. *IEEE Int. Conf. On Intelligent Robotics and Systems (IROS'2001)*, 2001
- [2] Belongie, S., C. Carson, H. Greenspan y J. Malik. Color- and texture-based image segmentation using EM and its application to content-based image retrieval. *Proc. of the 6<sup>th</sup> IEEE Int. Conf. on Computer Vision*, pp. 675-682, 1998
- [3] Burt, P., T. Hong y A. Rosenfeld. Image smoothing based on neighbor linking. *IEEE Trans. on Systems, Man and Cybernetics*, **11** (12), pp. 769-780, 1981
- [4] Burt, P., T. Hong y A. Rosenfeld. Segmentation and estimation on image region properties through cooperative hierarchical computation. *IEEE Trans. on Systems, Man and Cybernetics*, **11** (12), pp. 802-809, 1981
- [5] Caelli, T. y D. Reye. On the classification of image regions by colour, texture and shape. *Pattern Recognition*, **26** (4), pp. 461-470, 1993

- [6] Cantoni, V. y S. Levialdi, *Pyramidal Systems for Computer Vision*, NATO ASI, Series F, **25**, Springer-Verlag, Berlin, 1986
- [7] Chen, D., R. Szczerba y J. Uhran. A framed-quadtrees approach for determining euclidean shortest paths in a 2D environment. *IEEE Trans. on Robotics and Automation*, **13** (5), pp. 668-680, 1997
- [8] Dubuisson-Jolly, M.P. y A. Gupta. Color and texture fusion: application to aerial image segmentation and GIS updating. *Image and Vision Computing*, **18**, pp. 823-832, 2000
- [9] Gevers, T. y A. Smeulders. Color constant ratio gradients for image segmentation and similarity of texture objects. *Proc. IEEE Comp. Soc. Conf. on Computer Vision Pattern Recogn. (CVPR01)*, 2001
- [10] Hsieh, I., K. Fan y C. Lin. A statistic approach to the detection of human faces in color natural scene. *Pattern Recognition*, **35**, pp. 1538-1596, 2002
- [11] Jain, A. y G. Healey. A multiscale representation including opponent color features for texture recognition. *IEEE Trans. Image Processing*, **7** (1), pp. 124-128, 1998
- [12] Jolion, J. y A. Montanvert. The adaptive pyramid: a framework for 2D image analysis. *CVGIP: Image Understanding*, **55** (3), pp. 339-348, 1992
- [13] Kyllönen, K. y M. Pietikäinen. Visual inspection of parquet slabs by combining color and texture. *Proc. IAPR Workshop on Machine Vision Applications (MVA '00)*, Tokyo-Japón, pp. 187-192, 2000
- [14] Manduchi, R.. Bayesian fusion of color and texture segmentation. *Proc. of the 7<sup>th</sup> IEEE Int. Conf. on Computer Vision*, **2**, pp. 956-962, 1999
- [15] Meer, P., J. Jolion y A. Rosenfeld. Fast algorithm for blind noise variance estimation. *IEEE Trans. Pattern Anal. Mach. Intelligence*, **12** (2), pp. 216-223, 1990
- [16] Mirmehdi, M. y M. Petrou. Segmentation of color textures. *IEEE Trans. on Pattern Anal. Machine Intell.*, **22** (2), pp. 142-159, 2000
- [17] Nacken, P. Image Analysis Methods Based on Hierarchies of Graphs and Multi-Scale Mathematical Morphology. *PhD Thesis, Faculteit der Wiskunde en Informatica, Universidad de Amsterdam*, 1994

- [18] Ojala, T., K. Valkealahti, E. Oja y M. Pietikäinen. Texture discrimination with multidimensional distributions of signed gray-level differences. *Pattern Recognition*, **34** (3), pp. 727-739, 2001
- [19] Panjwani, D.K. y G. Healey. Unsupervised segmentation of textured color images. *IEEE Trans. Pattern Anal. Machine Intell.*, **17** (10), pp. 939-954, 1995
- [20] Paschos, G. Perceptually uniform color spaces for color texture analysis: an empirical evaluation. *IEEE Trans. Image Processing*, **10** (6), pp. 932-937, 2001
- [21] Pietikäinen, M., T. Mäenpää y J. Viertola. Color texture classification with color histograms and local binary patterns. *2nd Int. Workshop on Texture Analysis and Synthesis*, pp. 109-112, 2002
- [22] Poncela, A., E.J. Perez, A. Bandera, C. Urdiales y F. Sandoval. Efficient integration of metric and topological maps for directed exploration of unknown environments. *Robotics and Autonomous Systems*, **41** (1), pp. 21-39, 2002
- [23] Rosenfeld, A., C. Ye-Wang y A. Wu. Multispectral texture. *IEEE Trans. Systems, Man and Cybernetics*, **12** (1), pp. 79-84, 1982
- [24] Rosenfeld, A. *Multiresolution Image Processing and Analysis*, Springer-Verlag: Berlin, 1984
- [25] Rubner, Y., L. Guibas y C. Tomasi. The earth mover's distance, multi-dimensional scaling, and color-based image retrieval. *Proc. ARPA Image Understanding Workshop*, 1997
- [26] Tan, S. y J. Kittler. Colour texture classification using features from colour histograms. *Proc. 8th Scand. Conf. on Image Analysis*, pp. 807-813, 1993
- [27] Tanimoto, S. y T. Pavlidis. A hierarchical data structure for picture processing, *Comput. Graphics Image Process.*, **4** (2), pp. 104-119, 1975
- [28] Tanimoto, S. y A. Klinger. *Structured Computer Vision*, Academic Press: San Diego, 1980
- [29] Valencia, G., J.A. Rodríguez, C. Urdiales, A. Bandera y F. Sandoval. Fast hierarchical image segmentation through irregular pyramids. *2nd IASTED International Conference Visualization, Imaging and Image Processing (VIIP 2002)*, 2002
- [30] Zöllner, T. y J. Buhmann. Self-organized clustering of mixture models for combined color and texture segmentation. *2nd Int. Workshop on Texture Analysis and Synthesis*, pp. 163-167, 2002