

# DETERMINACIÓN AUTOMÁTICA DE COSTURAS PARA LA GENERACIÓN DE ORTOFOTOS

Roger Xandri, Vicenç Palà, Roman Arbiol

Unitat de Teledetecció  
Institut Cartogràfic de Catalunya  
Montjuïc s/n  
08038 Barcelona

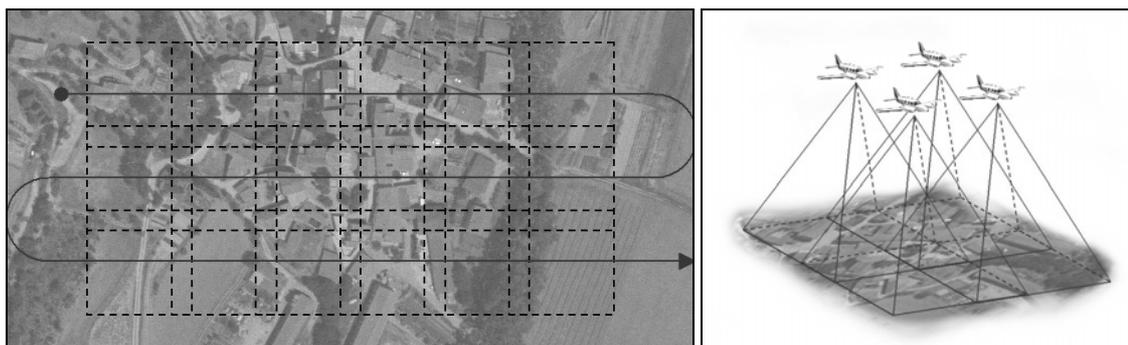
**Palabras clave:** Costuras, mosaico, ortofoto, estiradas.

## Resumen

De cara a la elaboración de una gran cantidad de productos cartográficos que pretenden cubrir una cierta extensión de territorio con un nivel de detalle elevado es imprescindible recurrir a la técnica del mosaico. Esto es fusionar todas las imágenes en un único producto. La calidad de este producto final depende de cómo su aspecto se asemeje a una sola imagen, disimulando los artefactos que el proceso de mosaico puede producir. Esta técnica tiene dos grandes problemas asociados: la igualación radiométrica y la forma de conexión entre cada una de las piezas. Para afrontar este segundo problema existen dos grandes grupos de métodos: los que proponen una zona de transición entre dos subescenas vecinas y los que a partir de una frontera deciden qué información aporta cada una. Este trabajo presenta un algoritmo de búsqueda completamente automática de estas fronteras, llamadas **costuras**. Este algoritmo más una óptima gestión de los recursos, es lo que permite trabajar con gran cantidad de datos (esto es, imágenes a gran resolución), garantizando la calidad a lo largo de cada tramo de las costuras y poder generar resultados con poca apariencia lineal que no se perciben incluso cuando las escenas vecinas difieren bastante.

## 1. Introducción

Cuando queremos cubrir un territorio de ciertas dimensiones no es posible, en general, extraer toda la información de un modo global, por ejemplo con una sola fotografía aérea. Es entonces cuando la solución pasa por una segmentación del territorio en pequeñas regiones, la captura aislada y su posterior unificación. Con el fin de garantizar la regularidad de la información al cambiar de región, éstas se acostumbran a definir de modo que haya zonas comunes entre ellas (**Figuras 1 y 2**). De este modo, hay puntos con más de una observación. Es necesario entonces determinar, para cada punto, la observación que se le va a asignar. Una buena aproximación consiste en trazar un enrejado de costuras –o contornos, o fronteras- sobre la totalidad del territorio observado que determine la parte de información que aporta cada adquisición y componer una imagen única con las aportaciones de cada región. Esta solución recibe el nombre de **mosaico**. Las líneas de esta malla han de transcurrir en su totalidad dentro de estas zonas comunes.



**Figuras 1 y 2:** Plan de vuelo de un territorio. Adquisición de fotogramas.

El objetivo de este trabajo consiste en mostrar un método que genera de forma automática las costuras que forman esta red. En él se describe cómo resolver el caso sencillo formado por dos imágenes y una sola zona de transición, sin explicitar cómo este proceso se encadena en la generación de las costuras de todo un bloque. Primero se realizará una breve descripción de la metodología ya establecida en el flujo de generación de ortofotos del ICC [1], la problemática que surge y seguidamente cómo ésta se resuelve.

## 2. Definiciones y conceptos

Partimos de dos observaciones,  $F_1$  y  $F_2$ , sobre dos regiones que tienen una zona común. Definimos en esta zona la imagen  $I$  formada por las diferencias absolutas de este par de observaciones. Para mayor comodidad, escalaremos estas diferencias entre 0 y 127, de modo que 0 indique que las dos observaciones sobre un mismo punto han coincidido, o que tienen un coste 0.

$$I(p) = |F_1(p) - F_2(p)| \quad (1)$$

### 2.1 Camino

Como es habitual, entenderemos por camino cualquier sucesión ordenada y conexa de puntos en una imagen. Por comodidad y sin pérdida de generalidad, de aquí en adelante, conectividad equivale a conectividad a 4. Es decir que cada punto tiene cuatro vecinos, o que dos puntos en diagonal necesitan de un tercero que los conecte. Denotaremos por  $C_{pq}$  al camino que une los puntos  $p$  y  $q$ , es decir aquella sucesión ordenada y conexa que empieza en  $p$  y acaba en  $q$ . Igualmente,  $C_{AB}$  es el camino que une cualquier punto del conjunto  $A$  con cualquier punto del conjunto  $B$ .

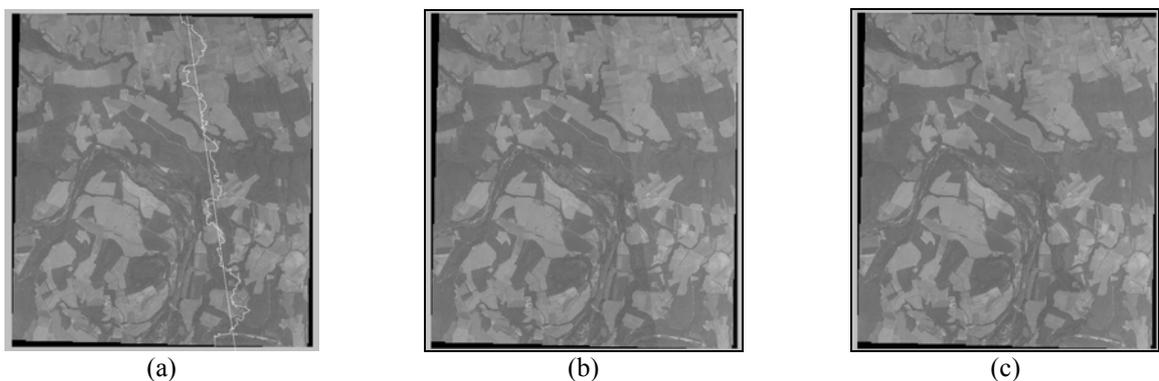
### 2.2 Coste

Es necesario determinar de qué forma asignamos un coste a todos y cada uno de los caminos que cruzan la imagen  $I$ . Esta asignación tiene que ser coherente con la idea que nos sugiere que caminos de bajo coste dan lugar a *buenas* costuras y caminos de coste elevado, costuras *malas*.

Quizás sea éste el momento de hacer una pequeña reflexión sobre qué entendemos por costura buena o mala. En nuestro caso, el mosaico de ortofotos aéreas, una buena costura es aquella que genera una transición entre dos fotogramas que no se detecta con la visión humana.

Supongamos por un momento que entre las adquisiciones de las observaciones de dos regiones vecinas hay un cierto espacio temporal. Es fácil imaginar que durante este espacio de tiempo, en la zona común de estas dos regiones se hayan producido ciertos cambios. Por ello, las dos observaciones tendrán información diferente. Pensemos durante un momento qué tipo de cambios pueden haber ocurrido. Por ejemplo, un campo sembrado puede haber cambiado su tonalidad; pero también puede haber aparecido una nueva construcción: una carretera, un edificio... Pensando un poco, nos convenceremos de que hay que penalizar pequeñas áreas con grandes cambios más que grandes áreas con pequeños cambios.

Buscando la definición más adecuada del coste que asignaremos a cada camino que cruza la imagen  $I$ , nos preguntamos si en esta definición ha de aparecer el concepto de longitud del camino, así como la idea de coste acumulado. Nuestra respuesta es que NO. Si trabajamos con coste acumulado, estamos premiando a caminos directos. Es sabido que la visión humana detecta de un modo más sencillo elementos lineales, formas simples o direcciones predominantes (**Figura 3**).



**Figura 3:** Detección visual de elementos lineales. La imagen (a) muestra dos costuras sobre una misma zona común: una más directa que la otra. La imagen (b) es el resultado de hacer el mosaico usando la costura lineal, mientras que la (c) utiliza la otra. La primera es mucho más fácil de detectar.

Así pues, se ha optado por una definición de coste bien sencilla: a cada camino le asignamos la diferencia más elevada que haya entre las dos observaciones de todos los puntos por los que pasa, esto es:

$$E(C) = \max_{p \in C} |F_1(p) - F_2(p)| = \max_{p \in C} I(p) \quad (2)$$

Para toda la descripción técnica detallada que viene a continuación, al referirnos al coste de un camino no tendremos en cuenta sus extremos. Con este convenio, un camino de sólo dos píxeles tiene siempre coste 0.

### 3. Primera aproximación

#### 3.1. Bisección

Lo primero que debemos hacer es determinar cuál es el mínimo coste necesario para cruzar una región. O, dicho de otro modo, queremos ir de **A** a **B** (donde **A** y **B** son los conjuntos de píxeles de salida y de llegada respectivamente) con el mínimo coste **n**. Eso quiere decir que hay caminos con costes de **n** a 127, pero no con costes de 0 a **n**-1. Para hacer esto, podríamos ir preguntando reiteradamente si hay caminos de coste 0, si hay de coste 1, de coste 2... hasta encontrar la primera respuesta afirmativa; o al revés, preguntar si hay caminos de coste 127, si hay de coste 126, de coste 125... hasta la primera respuesta negativa. Pero hay otra opción más rápida. Aplicamos un método de bisección al intervalo [0, 127]. Es decir, preguntamos si hay caminos de coste 63: si la respuesta es que sí, nos quedamos con el intervalo [0, 63]; si es que no, no quedamos con [63, 127]. En cualquier caso, iteramos hasta que la longitud del intervalo sea 1. Así, reducimos el número de consultas a  $\log_2 127 \approx 7$  (para más detalle ver [1]).

#### 3.2. Problemática

A primera vista, parece que con esto ya está todo listo: encontramos el coste mínimo y la solución será cualquier camino con este coste. ¡Pero no es tan sencillo! Seguidamente, intentaremos describir las problemáticas que aparecen. La que ha motivado este trabajo es la que se intenta describir a continuación [2].

Supongamos que estamos delante de una región que queremos cruzar de **A** a **B** y que ya hemos determinado que el mínimo coste posible es **n**. Seguramente habrá muchos caminos que van de **A** a **B** con este coste. ¿Cuál de ellos debemos tomar? De hecho, cualquiera de ellos podría ser válido, pero... ¿estamos controlando qué tiene que hacer el camino al pasar por las zonas con costes entre 0 y **n**?

Analicemos un ejemplo. Supongamos que esta región tiene una franja con un coste muy elevado que hay que cruzar. Supongamos también que el mínimo coste se logra precisamente al cruzar esta franja. En el resto de la región, el coste será inferior y por lo tanto el proceso de búsqueda de camino tiene absoluta libertad. Así, es fácil que forme costuras que crucen zonas de elevado coste (aunque sin alcanzar el coste de la franja sea suficientemente elevado para ser detectado por el ojo humano). Además si hay absoluta libertad en zonas con costes entre 0 y **n**, los métodos simples tendrán tendencia a dar soluciones en las que predominan las líneas rectas, y ya hemos mostrado en la **Figura 3** los problemas que esto causa.

### 4. Algoritmo de continuación

#### 4.1. Segmentación

Es necesario restringir la excesiva libertad que tiene el camino fuera de la zona de máximo coste. El algoritmo implementado, basado en la estrategia de *divide y vencerás*, es el siguiente:

#### ALGORITMO DE SEGMENTACIÓN

1. Encontramos el mínimo coste **n** para ir de **A** a **B**.
2. Tomamos un camino cualquiera con este coste.
3. Segmentamos el camino por los puntos donde se logra este coste **n**.
4. Repetimos 1. tomando como **A** y **B** los extremos de cada segmento.

Descrito en palabras:

Primero encontramos el mínimo coste **n** para ir de **A** a **B** y un camino cualquiera con este coste. Después cortamos el camino por los puntos donde se logra este coste (sabemos que hay uno o más) y para estos pequeños segmentos, buscamos el mínimo coste **n<sub>1</sub>** y un nuevo segmento con este coste que una sus extremos (sabemos que **n<sub>1</sub>** será menor que **n**, ya que los extremos no los consideramos). Podemos volver a aplicar este procedimiento de manera recursiva a cada pequeño segmento y para cada nuevo coste encontrado hasta que todos los segmentos tengan longitud 2 (y coste 0).

## EJEMPLO DEL ALGORITMO DE SEGMENTACIÓN

Primero hemos buscado con el método de la bisección el mínimo coste para ir de **A** a **B**. Ha resultado ser 17. Hemos encontrado un camino cualquiera con este coste (nos tenemos que convencer de que este coste se logra en uno o más píxeles; dos en este caso).

<b>A</b>	15	10	9	8	5	3	17	12	6	5	2	5	2	6	8	9	11	17	12	15	9	11	13	<b>B</b>
----------	----	----	---	---	---	---	----	----	---	---	---	---	---	---	---	---	----	----	----	----	---	----	----	----------

Los  $k$  píxeles con coste 17 nos dividen el camino en  $k + 1$  segmentos. Tomamos el último de estos, y llamamos  $A_1$  a su inicio:

...	<b>A<sub>1</sub></b>	12	15	9	11	13	<b>B</b>
-----	----------------------	----	----	---	----	----	----------

Este segmento ha sido encontrado con la única condición que no superara el máximo coste (17), pero este puede no ser el óptimo para ir de  $A_1$  a **B**. Por lo tanto, buscamos el mínimo coste para ir de  $A_1$  a **B** (también es necesario que nos convenzamos que será menor que 17). Seguidamente, tomamos un camino cualquiera con este coste (aunque este camino sea más largo que el anterior). Mostremos un posible resultado:

...	<b>A<sub>1</sub></b>	<b>10</b>	7	6	8	6	<b>10</b>	6	<b>B</b>
-----	----------------------	-----------	---	---	---	---	-----------	---	----------

Como antes, este segmento ha logrado su máximo coste (10) en dos puntos, que nos dividen este camino en tres segmentos. Tratemos el último de estos y llamemos  $A_2$  a su inicio:

...	<b>A<sub>2</sub></b>	6	<b>B</b>
-----	----------------------	---	----------

Busquemos el mínimo coste para ir de  $A_2$  a **B**. Esta vez será menor que 10. Supongamos que resulta ser 4 y que un camino con este coste es el siguiente:

...	<b>A<sub>2</sub></b>	3	3	4	<b>B</b>
-----	----------------------	---	---	---	----------

Dividamos este en segmentos por los píxeles en los que se logra el máximo coste (4). Obtenemos dos, y si tomamos el último de éstos, vemos que tiene **longitud 2**. Entonces damos el píxel de llegada como definitivo. Tomamos el siguiente segmento y llamamos  $B_1$  a su final:

...	<b>A<sub>2</sub></b>	3	3	<b>B<sub>1</sub></b>	<b>B</b>
-----	----------------------	---	---	----------------------	----------

Cuando calculamos el coste mínimo de  $A_2$  a  $B_1$  debe resultar menor que 4 (por ejemplo 3) y un camino con este coste puede ser el que ya tenemos. Este coste mínimo se logra en dos píxeles, que nos determinan tres segmentos. Pero, todos estos tienen **longitud 2**, y por lo tanto ya tenemos tres píxeles definitivos.

Si recapitulamos, tenemos:

...	<b>A<sub>1</sub></b>	<b>10</b>	7	6	8	6	<b>10</b>	<b>3</b>	<b>3</b>	4	<b>B</b>
-----	----------------------	-----------	---	---	---	---	-----------	----------	----------	---	----------

Ahora deberíamos continuar con el segmento delimitado por el coste 10 y así iterando hasta acabar.

## 5. Resultados

### 5.1. Tiempo

Esta variante del método inicial tiene un mayor coste de CPU (el método anterior es sólo el primer paso de éste). Además de CPU, y debido al etiquetaje y al uso de caché, el tiempo de ejecución depende de la velocidad de lectura-escritura a disco. Es difícil cuantificar este aumento, ya que hay muchos condicionantes y algunos muy difíciles de evaluar a priori. Uno de ellos es el tamaño de la imagen de diferencias **I**, claro está. Otros pueden ser el mínimo coste con el que podemos cruzar una imagen o su homogeneidad. Una de las partes del proceso que más tiempo de CPU consume es el etiquetaje, y cuanto mayor sea el mínimo coste, más capas deberemos etiquetar.

Estos aspectos provocan una gran variación en el tiempo de computación y en cambio no se pueden pronosticar. Las pruebas realizadas sobre regiones de 10000x4000 píxeles, con unos costes finales alrededor de 5 tardan cerca de 15 minutos en un PC de 300 MHz y cerca de 3 en uno de 2,4 GHz, ambos con 512 Mb de RAM. Las pruebas han mostrado la relevancia de la velocidad de transferencia a/de disco.

### 5.2. Histogramas

Con el fin de comprobar la calidad de los resultados y la magnitud de las mejoras, hemos hecho lo siguiente: hemos tomado cuatro imágenes distintas y hemos comparado los resultados actuales con los logrados con el método precedente. A cada imagen se le han calculado las dos costuras y se ha analizado el contenido de los píxeles que contenía cada una.

	Imagen 1				Imagen 2			
<b>0</b>	1316	<b>12.8%</b>	11165	<b>39.5%</b>	1405	<b>13.0%</b>	10235	<b>38.0%</b>
<b>1</b>	2547	<b>24.7%</b>	13965	<b>49.4%</b>	2601	<b>24.0%</b>	12913	<b>47.9%</b>
<b>2</b>	2436	<b>23.6%</b>	2562	<b>9.1%</b>	2707	<b>25.0%</b>	2847	<b>10.6%</b>
<b>3</b>	2268	<b>22.0%</b>	515	<b>1.8%</b>	2270	<b>21.0%</b>	797	<b>3.0%</b>
<b>4</b>	1744	<b>16.9%</b>	68	<b>0.2%</b>	1846	<b>17.0%</b>	164	<b>0.6%</b>

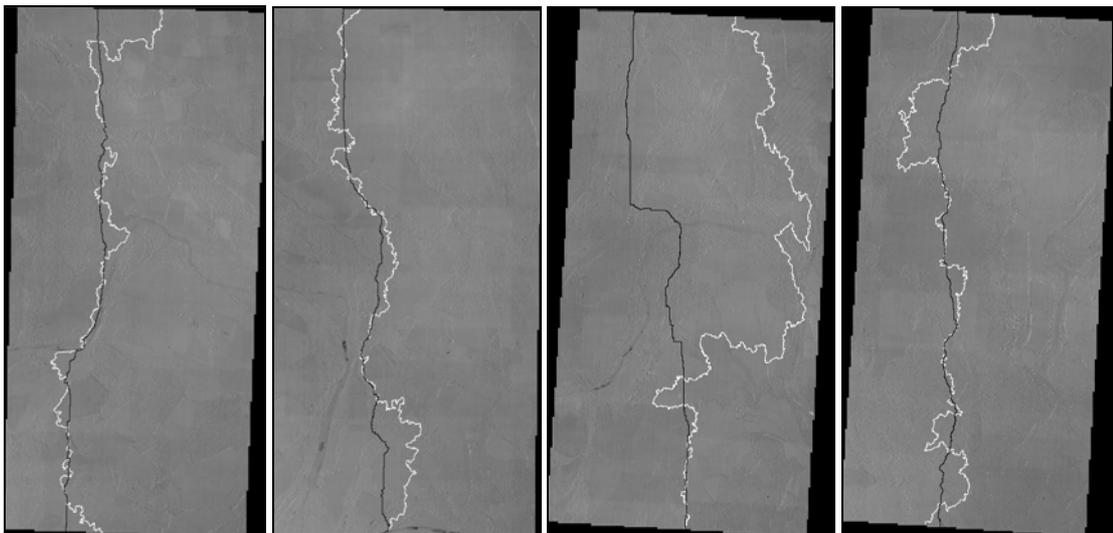
	Imagen 3				Imagen 4			
<b>0</b>	1090	<b>12.2%</b>	12895	<b>41.8%</b>	1118	<b>10.6%</b>	10470	<b>36.5%</b>
<b>1</b>	2255	<b>25.2%</b>	15330	<b>49.7%</b>	2384	<b>22.5%</b>	13581	<b>47.3%</b>
<b>2</b>	2116	<b>23.7%</b>	2343	<b>7.6%</b>	2491	<b>23.5%</b>	3567	<b>12.4%</b>
<b>3</b>	1891	<b>21.2%</b>	268	<b>0.9%</b>	2602	<b>24.6%</b>	988	<b>3.4%</b>
<b>4</b>	1579	<b>17.7%</b>	25	<b>0.1%</b>	1998	<b>18.9%</b>	104	<b>0.4%</b>

**Tabla 1:** Contenido radiométrico de las costuras detectadas con ambos métodos. En gris los resultados que se obtienen con el nuevo método.

Las cuatro imágenes han logrado costuras con coste máximo de 4 (el nuevo método no reduce el coste máximo). Para cada imagen y para cada uno de los dos métodos, se puede observar el número de píxeles que toma cada nivel digital de 0 a 4. Más interesante aún es el porcentaje que esto representa respecto al total de píxeles del camino.

### 5.3. Comparativa

La **Figura 4** muestra diferentes resultados con las costuras viejas (más oscuras) y las nuevas. Se puede observar la tendencia de las viejas a ir rectas, mientras que las nuevas buscan la mejor opción a lo largo de todo su recorrido.



**Figura 4:** Comparativa visual entre los dos métodos.

## 6. Conclusiones y vías abiertas

Después de utilizar esta nueva metodología para la generación de ortofoto 1:5000 del territorio de Catalunya los resultados obtenidos han sido muy satisfactorios. Se ha reducido casi en su totalidad la corrección manual de las costuras generadas de forma automática.

Cabe remarcar que la definición de coste establecida es incapaz de diferenciar situaciones, como la descrita en 6.1., en la que se atribuye el mismo coste a caminos claramente distintos desde el punto de vista de la percepción. Asimismo, en ocasiones, se determinan caminos de bajo coste claramente perceptibles bajo circunstancias como las que ilustra el apartado 6.2. Por otro lado en 6.3. se describen los resultados al definir las diferencias, y por tanto los costes, en un espacio de color (CIELuv) más próximo a la percepción humana. Para terminar, en el apartado 6.4., se describe cómo se puede utilizar esta metodología para reducir la incidencia de ciertos artefactos (aparición de nubes y estiradas) en el mosaico final.

### 6.1. Zonas mínimas

Presentaremos, mediante un ejemplo, una situación que el actual método no resuelve. Es la siguiente: en la imagen de costes que refleja la **Figura 5**, si pretendemos encontrar un camino que cruce de norte a sur, podemos observar que el coste mínimo es 8, y que una solución será cualquier camino que discurra dentro de la zona gris. Supongamos que el camino que se ha conseguido es el que está resaltado en negrita.

9	9	9	9	9	9	9	9
9	8	<b>8</b>	8	8	8	8	9
9	8	<b>8</b>	8	0	8	8	9
9	8	<b>8</b>	8	0	8	8	9
9	8	<b>8</b>	8	0	8	8	9
9	8	<b>8</b>	8	0	8	8	9
9	8	<b>8</b>	8	0	8	8	9
9	8	<b>8</b>	8	0	8	8	9
9	8	<b>8</b>	8	8	8	8	9
9	9	9	9	9	9	9	9

9	9	9	9	<b>9</b>	9	9	9
9	8	8	8	<b>8</b>	8	8	9
9	8	8	9	<b>0</b>	9	8	9
9	8	8	9	0	9	8	9
9	8	8	9	0	9	8	9
9	8	8	9	0	9	8	9
9	8	8	9	0	9	8	9
9	8	8	9	0	9	8	9
9	8	8	8	9	8	8	9
9	9	9	9	9	9	9	9

**Figuras 5 y 6:** Ejemplo de zona mínima no alcanzada.

Este camino ya es óptimo: no podemos unir ningún par de sus puntos con un “atajo” de coste menor que 8. Es evidente que hay un camino mejor: pasar por los ceros. El problema es que los píxeles con coste 0 no tocan el camino en negrita, y cuando buscan atajos, no los detectaran. Se puede pensar en la siguiente solución: una vez tenemos la mancha que nos posibilita cruzar la región de coste 8, analizamos los píxeles de menor coste y obligamos al camino a pasar por alguno de ellos. Pero, y si tenemos diferentes grupos de píxeles con este coste mínimo, ¿por cuál de ellos nos decantamos? ¿Y qué pasaría ante la situación de la **Figura 6**?

En gris vemos la mancha que nos permite la conexión de norte a sur con coste 8. Los píxeles de mínimo coste dentro de ella son los ceros. Pero en este caso, obligar al camino a pasar por 0 nos lleva a un callejón sin salida. Así pues, es necesario introducir criterios para decidir si podemos o no obligar al camino a alcanzar los píxeles de mínimo coste.

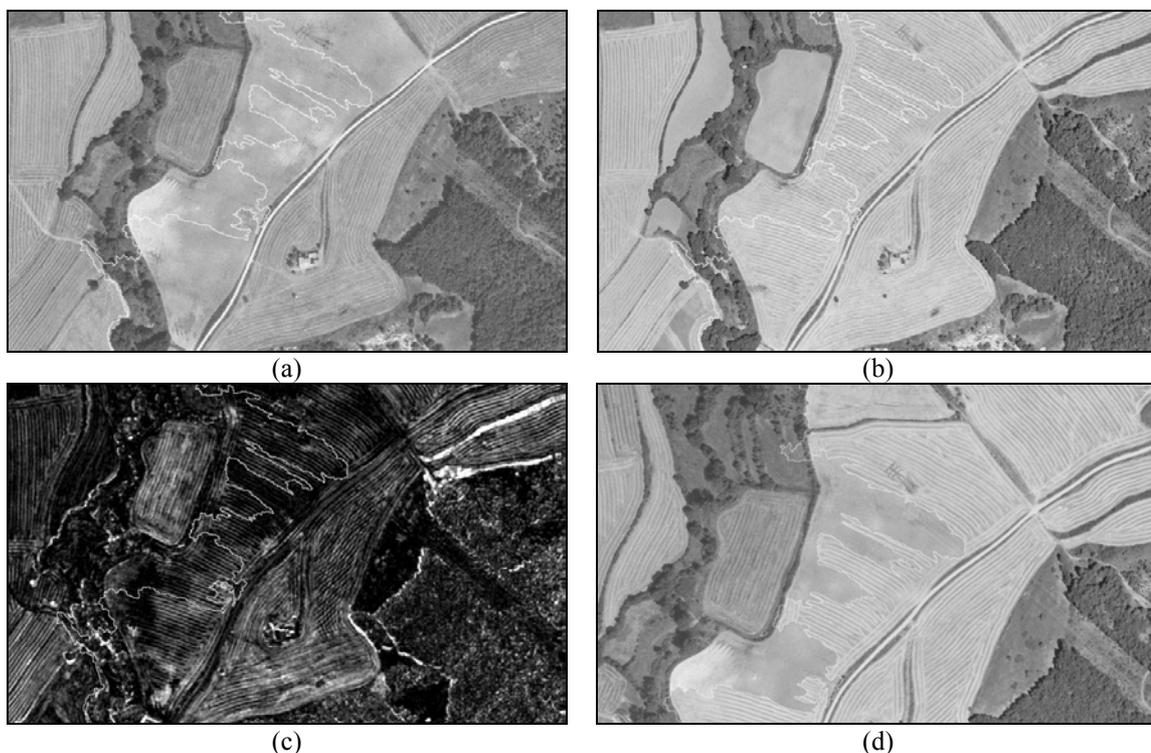
### 6.2. Espacio escala (reducción al máximo)

Querer evitar que las transiciones sean detectables por el sistema visual humano obliga, en determinadas ocasiones, a un filtrado de la imagen de diferencias **I**, con el fin de evitar que se dé por bueno un camino de bajo coste que discurre entre una zona de elevado coste.

30	30	30	1	30	30	30	20	10	8	8	8	8	8	8	8
30	30	30	1	30	30	30	20	10	8	8	8	8	8	8	8
30	30	30	1	30	30	30	20	10	8	8	8	8	8	8	8
30	30	30	1	1	1	30	20	10	8	8	8	8	8	8	8
30	30	30	30	30	1	30	20	10	8	8	8	8	8	8	8
30	30	30	30	30	1	30	20	10	8	8	8	8	8	8	8
30	30	1	1	1	1	30	20	10	8	8	8	8	8	8	8
30	30	1	30	30	30	30	20	10	8	8	8	8	8	8	8
30	30	1	30	30	30	30	20	10	8	8	8	8	8	8	8

**Figura 7:** Ilustra la necesidad de aplicar un filtro a la imagen de diferencias.

¿Cuál de los dos caminos de la **Figura 7** es mejor? Evidentemente, el primero tiene un coste menor, pero está claro que genera una frontera entre dos zonas con mucha diferencia. El ojo humano percibirá el cambio al cruzar de un fotograma al otro. Este cambio no se detecta por influencia de un solo píxel, sino de todo un conjunto. Analicémoslo en la **Figura 8**: tenemos dos fotogramas capturados con cierta diferencia temporal que contienen un mismo campo, labrado entre las dos adquisiciones. Observemos la correspondiente imagen de diferencias. El camino detectado busca zonas de mínima diferencia, pero con tanta precisión que consigue sortear las zonas de coste elevado. El resultado es claramente no deseable.



**Figura 8:** Las imágenes (a) y (b) muestran dos fotogramas adquiridos con cierto espacio temporal. La imagen (c) representa la imagen **I** de diferencias, y la imagen (d) la costura de bajo coste entre zonas de coste elevado.

Es acertado, entonces, aplicar un filtro a la imagen de diferencias. Tiene sentido y ventajas que este filtro sea un filtro del máximo (con una medida de máscara dependiendo de la resolución de la imagen). Sentido, porque evitará que hagamos mosaico unido por costuras de bajo coste entre zonas de coste elevado. Ventajas, porque utilizando el filtro del máximo podemos reducir el tamaño de la imagen de diferencia y, por consiguiente, los recursos de computación (con una ventana 5x5 pasamos de 64Mb de imagen **I** a 2,5Mb y de 15 minutos a 10 segundos).

Reduciendo la medida de la imagen de diferencias **I**, claro está, se reduce la precisión de la costura que se obtiene. Si para la precisión que se exige, la costura resultante no es aceptable, ésta se puede usar para definir una franja dentro de la que habrá que afinar la precisión.

### 6.3. Distancia y espacio CIELUV

Tal como hemos expuesto al inicio, lo que pretendemos es realizar un mosaico en el que las costuras sean lo más imperceptibles al ojo humano. Por eso se ha definido una distancia en el espacio de color a partir de la cual generábamos la imagen de diferencias **I**. Es sabido que en el espacio color RGB, ni la norma euclídea ni la norma del máximo son adecuadas para cuantificar cómo se asemejan (según el ojo humano) dos colores. Sería más preciso trasladarlo a algún otro espacio color: por ejemplo el CIELuv con la norma euclídea. La necesidad de esta modificación vendrá determinada, entre otros factores, por el rango de representación, la similitud de las imágenes en las zonas comunes, y la precisión que se quiera lograr. En nuestro caso, después de unas cuantas pruebas, no se han observado grandes mejoras tras esta transformación y se ha omitido.

### 6.4. Nubes y estiradas

Una diferencia, pequeña a priori, respecto a la metodología antigua es que el nuevo método no cruza una imagen de diferencias de Oeste a Este o de Norte a Sur, sino que una cualquier par de conjuntos de píxeles. Esto no parece muy relevante pero, pensando un poco más, veremos que nos permite abordar la detección de costuras de un modo “guiado”. Podemos añadir a la imagen de diferencias **I** otro tipo de información en ciertas zonas en las que además nos interesa determinar a qué lado de la costura queda cierta región de diferencias elevadas. Veamos unos ejemplos: supongamos que aparece una nube en la zona común de un par de fotogramas. Como que las diferencias radiométricas en la zona que corresponde a la nube son elevadas, la costura no la cruzará. ¿Pero nos importa a qué lado dejamos la nube?



**Figura 9:** Búsqueda “guiada” para evitar zonas no deseadas.

La tercera imagen, la de diferencias, muestra dos caminos: uno más claro que el otro. Se puede observar cómo la costura que pasa por la derecha de la nube generará un mosaico que obtiene la información de la zona de la nube del primer fotograma. Mientras que si tomáramos la costura de la izquierda, la nube aparecería en el mosaico final. Escoger la costura que nos deja la nube en el lado deseado es tan simple como penalizar con un coste muy elevado una franja desde la nube hasta el margen en la dirección del fotograma que no contiene la nube (el de la izquierda).

Está claro que para esta búsqueda “guiada” no hay suficiente con la información que nos aporta la imagen de diferencias. Necesitamos saber, además, de qué fotograma debemos coger la información de cada zona conflictiva.

Esta misma idea se puede aplicar para eliminar las estiradas y ocultaciones que se producen en los procesos de corrección geométrica de los fotogramas.

### Referencias

- [1]. E. FERNÁNDEZ, R. GARFINKEL, y R. ARBIOL. 1998. Mosaicking of Aerial Photographic Maps via Seams Defined by Bottleneck Shortest Paths. *Operations Research*, Vol 46, No 3, mayo-junio 1998, Universidad Politécnica de Cataluña.
- [2]. AURELI ALABERT y otros. Definición automática de la costura entre dos imágenes de un mosaico. Respuesta al problema presentado por R. Arbiol al Grupo de Estudio.