

Procesamiento distribuido 2D y 3D de imágenes biomédicas mediante filtros de difusión no lineal

Carlos Platero, Gabriel Asensio

Escuela Universitaria de Ingeniería Técnica Industrial (UPM), c/ Ronda de Valencia, 3, 28012 Madrid,
cplatero@elai.upm.es, gasensio@ma.upm.es

Resumen

Este trabajo trata de mejorar los tiempos computacionales del filtrado de difusión no lineal CED (Coherence Enhancing Filtering). La optimización se hace en una doble vertiente: disminuyendo el esfuerzo computacional del cálculo del tensor de difusión y distribuyendo el procesado en un cluster de computadores. La primera opción se basa en sustituir los procedimientos numéricos por expresiones analíticas de los autovalores de la estructura del tensor. Se empleará un método novedoso en el cálculo del tensor 3D basados en invariantes de las matrices simétricas. La segunda vertiente emplea el escenario AOS (Additive Operator Splitting) y la extensión del kernel de LINUX OpenMosix para distribuir el procesamiento en un grupo de computadores.

Palabras Clave: Filtros de difusión no lineal. Tensor de difusión. AOS. Imágenes biomédicas.

1 INTRODUCCIÓN

En la mayoría de las aplicaciones donde se emplea el procesamiento de imágenes biomédicas, o bien se desea eliminar ruido o bien se pretende regularizar la imagen, intentando, en ambos casos, preservar la estructura de la imagen. Se podría decir que esta afirmación es extensible a otras muchas aplicaciones del procesamiento de imágenes. Durante mucho tiempo se ha estado empleando masivamente el filtrado lineal, con la ventaja del abaratamiento computacional. Sin embargo, tiene el gran inconveniente de aplicarse de forma isotrópica. No hay realimentación entre qué zona de la imagen se está procesando y el filtrado al que es sometido. Regiones homogéneas y contornos son procesados de igual manera. En esta última década ha habido un gran entusiasmo en aplicar PDEs (*Partial Differential Equations*) sobre el procesamiento de imágenes. En gran medida, por que resuelve el inconveniente anteriormente mencionado. Hay posibilidades de aplicar un filtrado no isotrópico. Un

ejemplo de esta forma de actuar está en el filtrado de difusión no lineal.

Atendiendo a la analogía entre el campo escalar de las temperaturas en un proceso de difusión y el campo escalar del nivel de gris de una imagen en un filtrado de suavizado, las dinámicas de ambos procesos responde a la expresión:

$$\partial_t u = \text{div}(\nabla u) \quad (1)$$

donde u es el potencial del campo escalar y la ecuación rige la dinámica de un proceso de difusión lineal. Koenderink[6] se dio cuenta de que la expresión era idéntica a una convolución de la imagen con una máscara de Gauss, cuya varianza está dada por:

$$\sigma = \sqrt{2t} \quad (2)$$

El resultado, a parte de corresponder con un filtro paso bajo, tiene una interpretación física. El nivel de gris, es en la analogía, un nivel de temperatura y tiende a homogenizarse en el tiempo, t . Por tanto, el suavizado de la imagen depende del tiempo transcurrido y desde el punto de vista de la implementación con el valor de la varianza de la máscara de Gauss.

Sin embargo, el gran salto se dará con el trabajo de Perona y Malik[8] que por primera vez se mostraba un filtrado de difusión no lineal. El objetivo es acentuar el proceso de difusión en las zonas homogéneas y preservar los bordes:

$$\partial_t u = \text{div}\left(g\left(|\nabla u|^2\right)\nabla u\right) \quad (3)$$

siendo $g(\cdot)$ una función de $g: \mathcal{R}^m \rightarrow \mathcal{R}$ (m es la dimensión del campo escalar u), decreciente con la magnitud del módulo del gradiente. Su valor es prácticamente nulo en los bordes y próximo a 1 en regiones regulares. A la función $g(\cdot)$ se le llama de la difusividad. Ha habido varias propuesta de esta función en los últimos años (ver un resumen en[4]).

Con todo, esta formulación lleva consigo que en algunos puntos se dé un proceso de difusión inversa, provocando el realce puntual del ruido[5][10]. Weickert[13] propuso un nuevo marco de difusión no lineal que no sólo aproveche la orientación local sino también las diferentes direcciones de suavizado. Para ello, ha sustituido la difusividad por el tensor de difusión:

$$\partial_i = \text{div}(D\nabla u) \quad (4)$$

donde D es una matriz $m \times m$ simétrica y definida semi-positiva. Estos tipos de filtros tienen un comportamiento bien establecido de no isotropía, garantizando los contornos y suavizando las áreas más homogéneas.

En aplicaciones donde se pretende recuperar la coherencia de la estructura de la imagen, perturbada generalmente por los sistemas de adquisición, Weickert[14] ha propuesto el uso de tensor de difusión de realce de la coherencia (*CED - Coherence Enhancing Filtering*). La estructura del tensor se basa en el producto del descriptor gradiente suavizado con su traspuesta y convolucionado con la máscara de Gauss K_ρ :

$$\begin{aligned} S &= J_\rho(\nabla u_\sigma) = K_\rho * (\nabla u_\sigma \nabla u_\sigma^T) \\ \nabla u_\sigma &= K_\sigma * \nabla u \\ K_\sigma &= \frac{1}{(2\pi\sigma)^{m/2}} \exp\left(-\frac{|u|^2}{2\sigma^2}\right) \end{aligned} \quad (5)$$

S o $J_\rho(\cdot)$ es una matriz que es invariante ante los cambios de signo de la dirección del gradiente, pero dependiente de su magnitud. El valor de la desviación típica σ depende de la escala del ruido y el valor de ρ depende de la orientación local. Con su convolución, se ha pasado de la dirección puntual del gradiente a la orientación prominente estimada de la estructura local. La matriz es simétrica y definida semipositiva. El tensor D estará determinado por los autovectores y autovalores de la estructura tensorial:

$$D = R^T C R \quad (6)$$

R es la matriz de rotación constituida por los autovectores de J_ρ y C es la matriz diagonal, cuyos elementos corresponden a los coeficientes de difusión en cada una de las componentes principales:

$$c_i = \alpha \quad i = 2, \dots, m \quad (7)$$

y

$$c_1 = \max\left(\alpha, 1 - \exp\left(\frac{-k}{T}\right)\right) \quad (8)$$

Habiendo ordenado las componentes de menor a mayor autovalor. Por tanto, la primera componente, estará indicando la dirección de máxima homogeneidad en el nivel de irradiación. Donde $T > 0$ es un parámetro umbral, α es el coeficiente de difusividad que está en el intervalo $[0,1]$ y k es una medida de la coherencia local:

$$k = \sum_{i=1}^m \sum_{i \neq j} (\mu_i - \mu_j)^2 \quad (9)$$

siendo μ_i el autovector de la componente principal i de $J_\rho(\cdot)$. El coeficiente de difusividad, α , es un valor pequeño positivo capaz de mantener el tensor definido positivo.

Esta ponencia se desenvuelve en este marco de trabajo, esto es, CED. Sus objetivos se centran en la mejora de su implementación mediante la sustitución del cálculo numérico por expresiones analíticas y en potenciar las capacidades intrínsecas del procesamiento distribuido.

La organización de esta ponencia está realizada como sigue. En la sección 2, se presenta la discretización de los filtros de difusión no lineal y se llega al escenario semimplícito AOS (*Additive Operator Splitting*). El tercer párrafo se destina a una mejora de los tiempos de computación a través de su reinterpretación matemática y a una implementación del algoritmo de forma distribuida. En la sección 4, los resultados experimentales sobre imágenes biomédicas son presentados y comentados, explorando, especialmente, los tiempos de computación. Por último, en el último apartado se reflejan las conclusiones de este trabajo.

2 ALGORITMIA DE LOS FILTROS DE DIFUSIÓN NO LINEAL

El desarrollo de este punto se va a realizar, en primer lugar, detallando la algoritmia para el caso más simple, esto es, mediante función de difusividad. Posteriormente, se extenderá al caso de tensor de difusión.

Considerando la ecuación de difusión no lineal con una función $g(\cdot)$, capaz de determinar para cada píxel el escalar de difusividad, la discretización explícita o de Euler de la ec. 3 quedará como:

$$\frac{u^{k+1} - u^k}{\tau} = \sum_{i=1}^m \partial_{x_i} (g(\cdot) \partial_{x_i} u) \quad (10)$$

En el entorno de vecindad 3X3 e independizado por cada eje del espacio, la discretización más simple del lado derecho de la expresión, en el caso unidimensional, será:

$$\partial_x (g(\cdot) \partial_x u) = \frac{1}{2} \sum_{j=-1}^1 (g_i + g_{i+j}) (u_{i+j} - u_i) \quad (11)$$

Expresión que puede ser colocada en notación matricial:

$$\frac{u^{k+1} - u^k}{\tau} = (A^k) u^k \quad (12)$$

donde $A(u^k)$ es una matriz tridiagonal, cuya diagonal y sus extremos vale:

$$a_{ij} = \begin{cases} -\frac{1}{2} \sum_{r=-1,1} g_i + g_{i+r} & i = j \\ \frac{1}{2} (g_i + g_j) & j = i + 1, i - 1 \end{cases} \quad (13)$$

El escenario explícito queda definido por el proceso iterativo:

$$u^{k+1} = (I + \tau A(u^k)) u^k \quad (14)$$

Sin embargo, este escenario es poco eficiente por qué según demuestra Weickert[12], el paso en cada iteración, τ , es muy pequeño para que el filtrado sea estable. Deduce que el tiempo de paso en cada iteración debe ser:

$$\tau < \frac{1}{2m} \quad (15)$$

Este efecto produce que el algoritmo evolucione lentamente en procesar volúmenes de información de tipo de imágenes 3D, ya que al ser muy pequeño el paso, se necesita aumentar el número de iteraciones.

Considerando una discretización más elaborada del PDE de difusión no lineal en su forma matricial:

$$\frac{u^{k+1} - u^k}{\tau} = A(u^k) u^{k+1} \quad (16)$$

el cual conduce a un escenario semimplícito lineal:

$$\begin{aligned} (-\tau A(u^k)) u^{k+1} &= u^k \\ u^{k+1} &= (-\tau A(u^k))^{-1} u^k \end{aligned} \quad (17)$$

Esta propuesta requiere resolver el sistema lineal de ecuaciones. El camino más efectivo para calcularlo es a través del algoritmo de Thomas. Éste es una extensión del método numérico de Gauss para matrices tridiagonales[3][9].

Aun así, el valor de paso de la algoritmia semimplícita todavía no sigue estando resuelta. Ha habido varias propuestas, pero una solución con posibilidad de procesamiento distribuido está en el escenario conocido con el acrónimo de AOS (*Additive Operator Splitting*) de Weickert[12]. Este algoritmo trata por igual a todas las direcciones axiales de difusión y permanece estable con cualquier tiempo de paso en el proceso iterativo:

$$u^{k+1} = \sum_{i=1}^m (I - m \tau A_i(u^k))^{-1} u^k \quad (18)$$

Este escenario, desarrollado para funciones de difusividad, es extensible al uso de tensores de difusión. En este caso, desde la ecuación continua dada en la ec. 4, la más simple discretización está dada por:

$$\frac{u^{k+1} - u^k}{\tau} = \sum_{i=1}^m \sum_{j=1}^m L_{ij} (u^k) u^k \quad (19)$$

L_{ij} es la matriz tridiagonal de la diferenciación central que aproxima al operador $\partial_{x_i} (d_{ij} \partial_{x_j})$. La conversión del escenario explícito con tensor de difusión a AOS está dado por:

$$u^{k+1} = \frac{1}{m} \sum_{l=1}^m (I - m \tau L_{ll}^k)^{-1} \left(I + \tau \sum_{i=1}^m \sum_{j \neq i} L_{ij}^k \right) u^k \quad (20)$$

3 IMPLEMENTACIÓN Y MEJORA DE LOS FILTROS DE DIFUSIÓN NO LINEAL

El caso más general de la algoritmia corresponde con el tensor de difusión. El empleo de la función de la difusividad es un caso particular del tensor. Por las características de las aplicaciones tratadas, se empleará el tensor CED. No obstante, teniendo en cuenta las peculiaridades propias de cada tensor, el esquema del flujograma que se va a exponer, es bastante similar para otras definiciones del tensor de difusión. Los pasos a seguir son:

1. Convolución de la imagen con una máscara de Gauss con desviación típica en función

de la escala del ruido, K_σ . Por las características propias de esta máscara puede ser separada en cada uno de los ejes axiales de difusión.

2. Determinación de la estructura del tensor, mediante el cálculo de las derivadas en cada dirección, sus productos y la convolución con la máscara de Gauss K_ρ .
3. Determinación de los autovalores y autovectores de la estructura del tensor. Está bien extendido el uso del cálculo numérico para su localización, como por ejemplo, el algoritmo cíclico de Jacobi[9]. Aquí se propone un novedoso método analítico, mediante el empleo de invariantes en la estructura del tensor.
4. Cálculo del tensor CED.
5. Determinación de la variable auxiliar:

$$k = \left(I + \tau \sum_{i=1}^m \sum_{j \neq i} L_{ij}^k \right) u^k \quad (21)$$

6. Para cada una de las componentes axiales de difusión, determinar la matriz inversa mediante el algoritmo de Thomas:

$$w_l^{k+1} = \left(I - m \tau L_{ll}^k \right)^{-1} v^k \quad (22)$$

7. Localizar el valor medio:

$$u^{k+1} = \frac{1}{m} \sum_{l=1}^m w_l^{k+1} \quad (23)$$

Un análisis más detallado de la construcción del tensor de difusión sugiere varias mejoras. Para el caso de aplicaciones 2D y 3D, resulta mucho más eficiente trabajar en las expresiones analíticas del tensor que recurrir a emplear métodos numéricos. Esta afirmación puede darse por un doble motivo: la disminución de operaciones a realizar y a que si los invariantes de la estructura del tensor muestra que los autovalores son similares no hay necesidad de calcular el tensor. Ciertamente, si los autovalores son prácticamente idénticos, el índice de coherencia local, k , será nulo y la matriz diagonal C se convertirá en una matriz identidad por el escalar α . Al ser multiplicado por la matriz de rotación, el tensor de difusión será una matriz diagonal $I\alpha$. Por tanto, cuando la estructura local no tenga componentes privilegiadas no hay necesidad de calcular el tensor CED. Esta conclusión sugiere emplear máscaras de procesamiento dependiendo de los valores de los autovalores de la estructura del tensor.

La forma de actuar será determinar el nivel de coherencia local, k , si esta medida se encuentra por debajo de un cierto nivel, no se calculará el tensor y se sustituirá por la matriz diagonal $I\alpha$. En caso contrario, se procederá a determinar el tensor. El valor de k depende de si se procesa imágenes 2D o 3D. Las expresiones analíticas de los autovalores exigen un estudio independiente de ambos casos. Se procederá a estudiar la estructura del tensor 2D y luego el tensor 3D.

Para el caso 2D, los autovalores de la estructura del tensor, S , valen:

$$\begin{aligned} \mu_1 &= \frac{1}{2}(s_{11} + s_{12} + \gamma) \\ \mu_2 &= \frac{1}{2}(s_{11} + s_{12} - \gamma) \\ \gamma &= \sqrt{(s_{11} - s_{22})^2 - 4s_{12}^2} \end{aligned} \quad (24)$$

Según la definición del índice de coherencia local, ec. 9, y la determinación del tensor CED, ecs. 6,7 y 8, para valores de γ que cumpla:

$$\gamma^2 \leq -T \ln(1 - \alpha)$$

El tensor CED se convierte en matriz unitaria por el coeficiente de difusividad, α . En caso contrario, el tensor CED está definido la matriz diagonal C y la estructura del tensor, S :

$$\begin{aligned} d_{11} &= \frac{1}{2} \left(c_1 + c_2 + \frac{(c_2 - c_1)(s_{11} - s_{22})}{\gamma} \right) \\ d_{22} &= \frac{1}{2} \left(c_1 + c_2 - \frac{(c_2 - c_1)(s_{11} - s_{22})}{\gamma} \right) \\ &= \frac{(c_1 - c_2)s_{12}}{\gamma} \end{aligned} \quad (25)$$

El cálculo de los autovalores de un tensor simétrico de orden 3 requiere introducir los siguiente invariantes de la estructura del tensor:

$$\begin{aligned} I_1 &= (S) \\ I_2 &= \frac{1}{2} \left((S^2) - (S)^2 \right) \\ I_3 &= \det(S) \end{aligned} \quad (26)$$

que sirven para calcular los invariantes de la componente desviadora de S , $\tilde{S} = S - \frac{1}{3} I_1 I$:

$$J_2 = \frac{1}{2} \left(\tilde{S}^2 - (\tilde{S})^2 \right) = \frac{1}{3} (I_1^2 + 3I_2) \quad (27)$$

$$J_3 = \det(\tilde{S}) = \frac{1}{27} (2I_1^3 + 9I_1I_2 + 27I_3)$$

que permite calcular el ángulo de Lode:

$$\cos(3\theta) = \frac{3\sqrt{3}}{2} \frac{J_3}{J_2^{3/2}} \quad 0 \leq \theta \leq \frac{\pi}{3} \quad (28)$$

se puede demostrar [1] que los autovalores del tensor S ordenados de menor a mayor son:

$$\begin{aligned} \mu_1 &= \frac{1}{3} I_1 + 2\sqrt{\frac{J_2}{3}} \cos\left(\theta + \frac{2\pi}{3}\right) \\ \mu_2 &= \frac{1}{3} I_1 + 2\sqrt{\frac{J_2}{3}} \cos\left(\theta - \frac{2\pi}{3}\right) \\ \mu_3 &= \frac{1}{3} I_1 + 2\sqrt{\frac{J_2}{3}} \cos(\theta) \end{aligned} \quad (29)$$

El índice de coherencia dependerá sólo del invariantes J_2 :

$$k = \sum_{i=1}^3 \sum_{i \neq j} (\mu_i - \mu_j)^2 = 6J_2 \quad (30)$$

Por tanto, para aquellos puntos cuyas componentes principales sean similares entre sí o autovalores muy pequeños, el tensor CED se comporta como una matriz unitaria escalada por α . No habiendo necesidad de calcularla. Esta condición queda refrendada por la desigualdad:

$$2 \leq -\frac{T}{6} \ln(1 - \alpha) \quad (31)$$

En aquellos píxeles que cumplan esta condición no será necesario determinar el tensor CED.

El cálculo de los elementos del tensor serán función de la estructura del tensor, de los coeficientes de difusividad en cada componentes y de los autovalores, $D = D(S, C, N)$.

La paralelización del filtrado se basa en la partición de la imagen. Esta estrategia tiene la dificultad del procesamiento en los bordes, como consecuencia de las operaciones en los entornos de vecindad. Hay varias soluciones [2], optándose por la más sencilla que no es la más eficiente. A cada procesador se le ha mandado un cubo particional, al que se le ha añadido la información de los píxeles exteriores. La vecindad de los píxeles fronteras está definido por el máximo, dado por la máscara de convolución de ρ , dependiente de la orientación local.

Realizadas las particiones, se manda a realizar un proceso por cada una de ellas. Dejándose a OpenMosix[7] la distribución de los procesos en el cluster.

4 APLICACIONES A IMÁGENES 2D Y 3D BIOMÉDICAS

Una de las aplicaciones 2D desarrolladas con procesamiento CED-AOS, fue la determinación de núcleos de células que no habían sido marcados biológicamente[11]. Estos núcleos aparecen con una tonalidad más oscura que aquellos que si han sido seleccionados. Pero debido al agrupamiento celular, entre núcleos marcados y no marcados, la información local aparece difuminada. Sin embargo, desde una perspectiva global se muestra la diferencia entre ambos tipos de núcleo. En la figura 1a se observa la imagen original, diferenciándose la células marcadas de aquellas que no lo están, por que las primeras aparecen con un color verdoso y las no seleccionadas con huecos de tonalidad marrón. La parte derecha de la figura, 1b, se han marcado los resultados del algoritmo. Se ha empleado la superposición del perímetro, en color rojo, para indicar la salida del algoritmo.

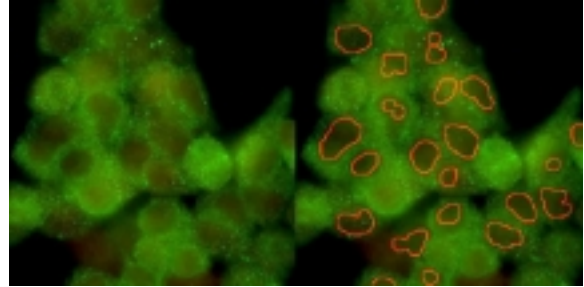


Figura 1 a) Imagen original (512x512) b) Imagen resultado

Para realzar la discrepancia entre marcadas y no marcadas, se realizó un preprocesado de la imagen, restando a la componente del canal verde del rojo. La imagen resultado puede ser visualizada en la figura 2a. Sobre esta imagen se procedió a la aplicación de CED con escenario AOS. Obsérvese la regularización de la imagen

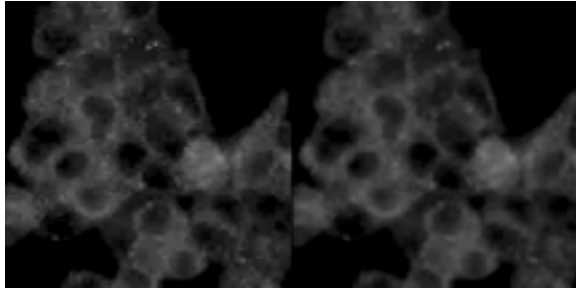


Figura 2 a) Imagen resultado del preprocesado, b) Imagen procesada con $\alpha=0.01$, $T=0.1$, $\sigma=1$, $\rho=5$, $\tau=25$ y 10 iteraciones

Hay que destacar que con valores de paso grandes, $\tau=25$, no hay pérdida de invarianzas a rotaciones. No muestra direcciones privilegiadas en el proceso de difusión. Evidentemente, este valor de paso en el algoritmo no se podría haber realizado con un escenario explícito.

Para recuperar la estructura de una célula madre neural, adquirida por un microscopio confocal, se ha procedido a la aplicación de difusión no lineal CED 3D.

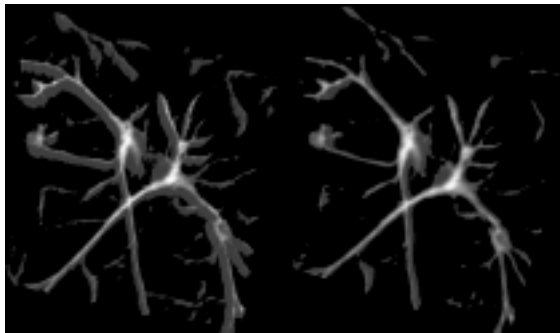


Figura 3 a) Célula madre neural adquirida por confocal (256x256x32) b) CED con $T=0.1$, $\sigma=1$, $\rho=6$, $\tau=4$ y 10 iteraciones

En la figura 3 se aprecia la eliminación del ruido de la adquisición y refleja la preservación de los contornos. En esta aplicación valores elevados del paso, $\tau>5$, se ha encontrado tendencia a perder la invarianza a rotaciones.

Las diferencias de tiempos de cómputo entre emplear métodos numéricos y las expresiones analíticas expuestas para el cálculo del tensor CED llegan casi a dos órdenes de magnitud en el caso 2D y un orden de magnitud en 3D:

| Tamaño | Numérico | Simbólico | Ratio |
|----------|------------|-----------|-------|
| 64x64 | 187,5ms | 3,1ms | 60,48 |
| 128x128 | 781,4ms | 13,7ms | 57,03 |
| 256x256 | 3.760,9ms | 57,6ms | 65,29 |
| 512x512 | 19.907,9ms | 206,3ms | 96,49 |
| 32x32x32 | 2.987,4 ms | 365.4ms | 8.71 |
| 64x64x64 | 25.370,6ms | 4.020,1ms | 6.31 |

Tabla 1. Tiempos de cómputo del tensor CED en un P-IV a 2.40GHz.

Para imágenes con una gran cantidad de píxeles con niveles de coherencia local muy bajo, se puede aprovechar que el tensor se convierte en una matriz unitaria escalada por la difusividad. Dependiendo del número de píxeles que cumple esta característica respecto del total se mejorará los tiempos. Esta reducción depende de la naturaleza de la imagen a procesar. En el caso 2D, por cada píxel de similares autovalores el ahorro es de 15 sumas, 3 multiplicaciones, 7 divisiones y 2 potencias. Más significativo es el caso de 3D, donde se ahorraría 232 sumas, 211 multiplicaciones, 68 potencias y 4 operaciones trigonométricas.

5 CONCLUSIONES

Se ha presentado el marco teórico y de implementación de los filtros de difusión no lineal CED. Las grandes posibilidades de eliminación del ruido o de regularización de las imágenes 2D y 3D con esta técnica exige de un estudio cuidadoso de su puesta en funcionamiento. El mayor inconveniente reside en su alto coste computacional. Dos vertientes se ha seguido para disminuir los tiempos de procesamiento: expresiones analíticas del tensor CED 2D-3D y procesamiento distribuido.

La determinación de expresiones analíticas de los tensores 2D y 3D en función de la estructura del tensor y de los coeficientes de difusión, alcanza reducciones del tiempo de computación casi en dos órdenes de magnitud respecto a los métodos numéricos. Además, la obtención del índice de coherencia local a partir de un enfoque novedoso, basado en invariantes de las matrices simétricas, facilita el tipo y cálculo del tensor CED.

La segunda vertiente viene dada por el empleo del escenario AOS. La extensión del kernel de LINUX OpenMosix, garantiza un procesamiento distribuido entre los nodos correspondiente al cluster de computadores.

Agradecimientos

Los autores dan las gracias a Ángeles Trillo del Servicio de Bioelectromagnetismo del Hospital Ramón y Cajal por las cesiones de las imágenes biomédicas 2D y 3D.

Este proyecto tiene financiación dentro de los proyectos precompetitivos de la Universidad Politécnica de Madrid.

Referencias

- [1] Asensio, G., (2003) , *Aportaciones orientadas a las deformaciones elastostáticas*, Tesis doctoral de la Universidad Politécnica de Madrid – ETSI Caminos, Canales y Puertos.
- [2] Bruhn, A., Jakob, T., Fischer, M., (2002) *Designing 3D nonlinear Directional Filters for Edge Orientation Estimation*, LNCS 2449, pp 290-297.
- [3] Fuente, J.L., (1998) *Técnicas de cálculo para sistemas de Ecuaciones, programación lineal programación entera*, Segunda edición, Editorial Reverté.
- [4] Guichard, F., Morel, J-M, (2001) *Image Analysis and PDE's*, versión 27/3/2001. Sin publicar. Disponible en la Red.
- [5] Kawohl, B., Kutev, N., (1997) *Maximum and comparison principles for anisotropic diffusion*, Mathematical Institute, Universidad de Colonia.
- [6] Koenderink, J.J., (1984) *The structure of edge*, Biological Cybernetics, 50, 363-370.
- [7] <http://openmosix.sourceforge.net>
- [8] Perona, P., Malik, J., (1990) *Scale space and edge detection using anisotropic diffusion*, IEEE Trans. Pattern Analysis and Machine Intelligence, 12(7) 629-639.
- [9] Press, W.H., Teukolsky, S.A., (2002) Vetterling, W.T., Flannery, B.P., *Numerical Recipes in C*, Cambridge University Press.
- [10] Tschumperlé, D., Deriche, R., (2002) *PDE's based regularization of multivalued Images and Applications*, Tesis doctoral de la Universidad de Nice-Sophia Antipolis.
- [11] Úbeda, A., (2003), *Bases biológicas para el diagnóstico por imagen*, Sociedad Española de Protección Radiológica.
- [12] Weickert, J., Haar Romeny, B.M., Viergever, M.A., (1998) *Efficient and reliable schemes for nonlinear diffusion filtering* IEEE Transactions on Image Processing, Vol. 7, 398-410.
- [13] Weickert, J., (1999) *Coherence-enhancing diffusion filtering*, International Journal of Computer Vision, Vol. 31, 111-127.
- [14] Weickert, J., Schnörr, C., (2000) *PDE-based Preprocessing of Medical Images*, Technical Report 8/2000, Computer Science Series, Universidad de Mannheim.