

Estudio Multiparamétrico para Reconstrucción de Imágenes de TAC Empleando el Método LSQR con Pocas Vistas

Mónica Chillarón, Iván Costa, Vicente Vidal, Gumersindo Verdú, Sergio Morato
Universitat Politècnica de València
Camino de Vera S/N. 46022 Valencia
vvidal@dsic.upv.es

Resumen – Para la reconstrucción de imágenes de TAC (Tomografía Axial Computarizada) existen diversos métodos a utilizar. El método objeto de este estudio es el LSQR utilizando pocas vistas, que combinado con un proceso de filtrado y aceleración permite obtener reconstrucciones de buena calidad exponiendo al paciente a menos radiación. En este trabajo se ha realizado un estudio multiparamétrico en un entorno Grid para determinar los parámetros óptimos de dicho método, y comparar la calidad de las imágenes obtenidas tras la reconstrucción con determinado número de vistas.

1. INTRODUCCIÓN.

En el campo de reconstrucción de imagen de TAC son muchos los métodos utilizados, sin embargo existe poca literatura sobre la eficacia de métodos para reconstrucción con pocas vistas. Estos métodos reconstruyen imágenes a partir de menos proyecciones que los TAC tradicionales, ya que en lugar de utilizar las 360 proyecciones típicas que se realizan para cubrir un corte de un objeto, aprovechan la redundancia que existe en el proceso de adquisición de datos para poder trabajar con muchas menos proyecciones, pudiendo utilizar únicamente 180 sin ninguna variación en la calidad de la imagen. El beneficio que aporta este proceso es obvio, ya que reduce la radiación recibida por el paciente en proceso de adquisición a al menos la mitad, lo cual puede resultar vital para aquellos que tengan que verse sometidos a esta prueba con más frecuencia de la deseada, puesto que la nocividad de los rayos X aumenta con la dosis absorbida.

Uno de los métodos más eficientes para la reconstrucción reduciendo el número de vistas o proyecciones es el algoritmo LSQR para resolución de sistemas de ecuaciones lineales. Dicho algoritmo, combinado con un filtro que elimina ruido de la imagen reconstruida, y un proceso de aceleración que ayuda a la convergencia del proceso, se constituye un método muy robusto y eficiente, propuesto en (E. Parcero).

Sin embargo, la efectividad del método y la calidad de la imagen obtenida es altamente dependiente de distintos parámetros que se especificarán a continuación, los cuales no han sido analizados y evaluados con profundidad en estudios anteriores.

El objetivo de este trabajo es realizar un estudio multiparamétrico utilizando una infraestructura Grid para determinar tanto los parámetros óptimos de los algoritmos del método aplicado como el número de iteraciones de cada uno. Se ha escogido realizar el proceso en un entorno Grid debido a la exhaustividad con la que se han estudiado los parámetros, lo cual ha originado 12000 ejecuciones del método para cada fantoma estudiado, resultando este entorno el óptimo para este tipo de estudio.

2. CONSIDERACIONES TEÓRICAS.

Para la reconstrucción de imágenes de TAC mediante métodos iterativos se parte de dos elementos:

- La matriz del sistema $A^*u=g$ definida como $A=a_{i,j} \in \mathbb{R}^{M \times N}$, siendo M el número de rayos trazados, N el tamaño en píxeles de la imagen a reconstruir. En este caso, la matriz se ha construido mediante el método Forward-Projection, en el cual $a_{i,j}$ representa la contribución del píxel j al rayo i .
- El vector de proyecciones g , $g=[g_1, g_2, \dots, g_M]^T \in \mathbb{R}^M$, que contiene los datos obtenidos al trazar los rayos X sobre el objeto de estudio y calculando la atenuación sufrida por el rayo.

2.1. Método LSQR.

A partir de los datos anteriores se determina que el problema a resolver viene definido por la $A^*u=g$. Se trata de la resolución de un sistema de ecuaciones lineales rectangular disperso cuya solución 'el vector u ($u=[u_1, u_2, \dots, u_N]^T \in \mathbb{R}^N$), representa la imagen 'aproximada' reconstruida. Dicho sistema será resuelto mediante el método Least Square QR (LSQR) descrito en (Paige y Saunders), el cual aproxima la solución iterativamente minimizando $\|g-Au\|^2$, (Maier) disminuyendo el residuo calculado como $r_k = g - Au_k$ hasta alcanzar una tolerancia dada.

2.2. Filtro STF.

El siguiente paso es la aplicación del filtro *Soft Thresholding Filter* (W. Yu), que aplicado sobre la aproximación de la imagen solución obtenida en el paso anterior elimina ruido de tipo Gaussiano conservando el gradiente no solo vertical-horizontal sino también el gradiente diagonal, provocando que la imagen sea más nítida sin perder su estructura, lo cual es de vital importancia en pruebas médicas como el TAC. Se describe en la **Figura 1**.

$$u_{i,j}^{(n+1)} = \frac{1}{4+4\alpha} (q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i+1,j}^{(n+1)}) + q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i,j+1}^{(n+1)}) + q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i,j-1}^{(n+1)}) + q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i-1,j}^{(n+1)}) + \alpha \cdot ((q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i+1,j+1}^{(n+1)}) + q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i+1,j-1}^{(n+1)}) + q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i-1,j-1}^{(n+1)}) + q(\omega, \tilde{u}_{i,j}^{(n+1)}, \tilde{u}_{i-1,j+1}^{(n+1)}))),$$

$$q(\omega, y, z) = \begin{cases} (y+z)/2, & \text{if } |y-z| < \omega \\ y - \omega/2, & \text{if } y-z \geq \omega \\ y + \omega/2, & \text{if } y-z \leq -\omega. \end{cases}$$

$$w = \max_i |r_i|, r = A * (g - Au)$$

Figura 1 Algoritmo filtro STF.

En este algoritmo el parámetro interés de estudio es el parámetro α , el cual pondera el peso que se le da al gradiente vertical-horizontal frente al diagonal.

2.3 Aceleración.

Por último se aplica la técnica de aceleración *Fast Iterative Shrinkage Thresholding Algorithm* (FISTA) definido en (A. Beck), la cual introduce nuevas direcciones de búsqueda sobre la aproximación de la solución, provocando de este modo que el método *LSQR* converja más rápidamente. El proceso se muestra en la **Figura 2**.

$$\begin{aligned} \mathbf{h} &= \mathbf{u}^{(n)}; \\ t_{n+1} &= \frac{1 + \sqrt{1 + 4t_n^2}}{2}; \\ \mathbf{u}^{(n+1)} &= \mathbf{h} + \left(\frac{t_n - 1}{t_{n+1}}\right)(\mathbf{h} - \mathbf{u}_{temp}); \\ \mathbf{u}_{temp} &= \mathbf{h}; \end{aligned}$$

Figura 2 Algoritmo FISTA.

3. ENTORNO DE EJECUCION GRID.

Para realizar el estudio de los parámetros óptimos se ha utilizado la infraestructura Grid denominada EGI (European Grid Infraestructure). Dicha infraestructura está financiada con fondos públicos para dar a los científicos acceso a más de 650.000 CPU lógicas, 500 PB de almacenamiento en disco y 21 proveedores de cloud federados para impulsar la investigación y la innovación en Europa. La infraestructura proporciona capacidades tanto de computación de alto rendimiento como de cálculo / almacenamiento en la nube.

Aunque originado en Europa, este Grid ya cuenta con recursos proporcionados por 325 centros de recursos que se distribuyen en 56 países en Europa, la región de Asia y el Pacífico, Canadá y América Latina.

Dentro de este Grid, se ha hecho uso de la Organización Virtual (VO) Biomed, dedicada exclusivamente a investigación en el sector de Ciencias de la Vida, como puede ser la Imagen Médica, Bioinformática o Farmacología.

4. EXPERIMENTO REALIZADO

Para realizar el experimento se han considerado los parámetros que se listan a continuación:

- Resolución de la imagen a reconstruir (32, 64, 128, 256, 512)
- Número de proyecciones a considerar (180, 120, 90, 60, 50, 45, 35, 30)
- Número de iteraciones de LSQR a aplicar cada iteración externa (desde 5 hasta 15)
- Valor alfa del filtro STF (0.5, 0.7, 0.9, 1.0, 1.1, 1.3, 1.5)
- Aplicación o no del filtro
- Aplicación o no de la aceleración

Esto supone 12320 ejecuciones del método para cada fantoma al que se vaya a aplicar. Por ello, es necesario automatizar la generación, lanzamiento, monitorización y recuperación de los trabajos a ejecutar, ya que es inviable gestionarlos manualmente. Para este fin se han desarrollado diversos scripts en lenguaje Bash que haciendo uso del middleware utilizado en EGI llamado GLite realicen todo el proceso de manera automática.

5. RESULTADOS.

5.1 Fantomas utilizados.

El estudio se ha realizado para un corte de dos fantomas diferentes, ambos parte de los desarrollados en el proyecto FORBILD. El primero se trata del fantoma que representa la cabeza humana. El corte escogido y sus elementos se observa en la **Figura 4**. El segundo fantoma utilizado representa un tórax humano, escogiendo el corte que se muestra en la **Figura 3**.

Cabe señalar que ambos fantomas han sido obtenidos utilizando el programa de imagen médica CONRAD (Maier).

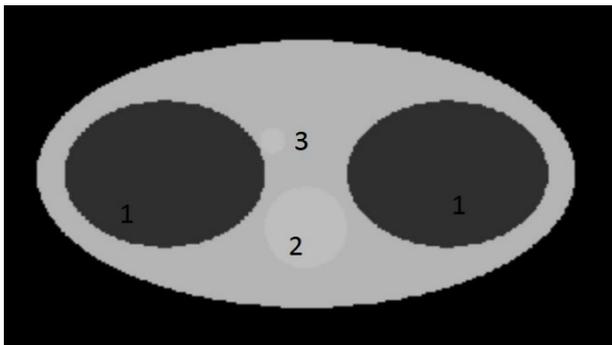


Figura 3 FORBILD Tórax, 1: pulmones, 2: corazón, 3: arteria aorta.

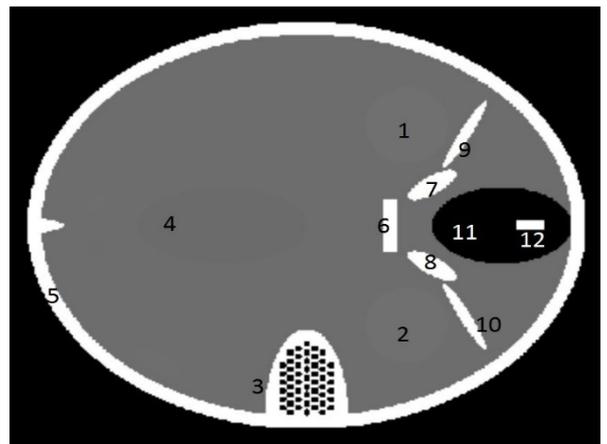


Figura 4 FORBILD Cabeza. 1, 2: ojos ; 3: oído derecho; 4: ventrículo ; 5: hueso craneal ; 6, 7, 8, 9, 10, 12: hueso ; 11: seno frontal.

5.2. Medidas de calidad.

Para poder comparar los resultados obtenidos y valorarlos en cuanto a calidad de la imagen reconstruida u y la original I_0 , se han considerado las siguientes métricas de calidad:

- **PSNR**

La métrica PSNR mide la relación entre la señal de la imagen y el ruido que contiene. Para calcularla se hace uso de otra métrica, la llamada Error Cuadrático Medio (MSE), que representa la media de los errores cuadráticos de la imagen.

Una vez calculado el MSE, se utiliza para calcular el PSNR según la ecuación siguiente en la cual MAX representa el máximo valor que puede tomar un píxel, siendo mejor resultado cuanto más alto:

$$PSNR = 10 * \log_{10} \frac{MAX I_0^2}{MSE}$$

- **SSIM**

Se trata de una medida que valora la estructura de la imagen, a diferencia de la anterior. Por ello, no valora los niveles de gris de los píxeles, sino las formas de la imagen reconstruida con respecto a la imagen de referencia, y por lo tanto mide aquello que es perceptible al ojo humano.

Se aplica mediante ventanas de tamaño fijo, y se calcula la diferencia entre dos ventanas x e y correspondientes a las dos imágenes a comparar, mediante la ecuación siguiente:

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{x,y} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

En dicha ecuación, μ_x y μ_y denotan el valor medio de la ventana x e y, σ_x^2 y σ_y^2 la varianza, σ_{xy} la covarianza entre las ventanas, y c_1 y c_2 son dos variables estabilizadoras dependientes del rango dinámico de la imagen.

5.3 Análisis de los resultados.

En este estudio se han valorado las configuraciones según sus valores de PSNR y SSIM, escogiendo aquellas con mejor resultado. Basándonos en estos parámetros se muestran a continuación los parámetros escogidos para cada resolución de imagen.

- Imagen reconstruida resolución 32x32

Proy	Iter. LSQR	Filtro	Acel .	Alfa STF	PSNR	SSIM	Iter. LSQR	Filtro	Acel .	Alfa STF	PSNR	SSIM
180	12	No	Si	-	111,47814	1	9	No	Si	-	116,66921	1
120	12	No	Si	-	110,92237	1	14	No	Si	-	114,53056	1
90	15	No	Si	-	110,29444	1	14	No	Si	-	115,41144	1
60	9	No	Si	-	105,63243	1	10	No	Si	-	109,63018	1
50	9	No	Si	-	104,61056	1	7	No	Si	-	110,55654	1
45	5	No	Si	-	101,5729	1	12	No	Si	-	107,88648	1
35	10	No	Si	-	98,92876	1	14	No	Si	-	103,85127	1
30	13	No	Si	-	91,79237	1	5	No	Si	-	101,53855	1

- Imagen reconstruida resolución 64x64

Proy	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM
180	14	No	Si	-	105,84979	1	6	No	Si	-	109,85228	1
120	6	No	Si	-	107,03783	1	7	No	Si	-	109,94079	1
90	13	No	Si	-	97,86274	1	8	No	Si	-	101,97794	1
60	12	No	Si	-	88,22246	1	10	No	Si	-	96,47998	1
50	10	No	Si	-	86,10104	1	14	No	Si	-	90,83038	1
45	5	No	Si	-	84,96388	1	10	No	Si	-	89,57176	1
35	5	No	Si	-	82,68266	1	5	No	Si	-	86,12875	1
30	12	No	Si	-	80,29179	1	5	No	Si	-	83,29618	1

- Imagen reconstruida resolución 128x128

Proy	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM
180	13	No	Si	-	94,04507	1	6	No	Si	-	96,84537	1
120	5	No	Si	-	86,08563	1	15	Si	Si	0.5	103,45988	1
90	5	No	Si	-	82,82579	1	14	Si	Si	0.9	98,95352	1
60	14	Si	Si	1.0	89,95326	1	10	Si	Si	0.5	97,23814	1
50	14	Si	Si	1.1	91,48527	1	9	Si	No	0.7	94,40738	1
45	13	Si	Si	0.9	94,93664	1	9	Si	Si	0.5	97,77963	1
35	12	Si	No	1.3	85,8938	1	8	Si	Si	0.5	94,24797	1
30	11	Si	Si	1.5	89,81934	1	8	Si	Si	1.5	91,79355	1

- Imagen reconstruida resolución 256x256

Proy	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM
180	12	Si	Si	1.0	104,90961	1	9	Si	Si	1.3	102,15789	1
120	9	Si	Si	0.9	97,15289	1	7	Si	Si	1.5	103,96321	1
90	9	Si	Si	1.3	96,56091	1	6	Si	Si	1.5	99,64562	1
60	7	Si	No	1.5	87,84987	1	5	Si	Si	1.3	97,89843	1
50	7	Si	No	1.5	86,92678	1	5	Si	No	1.5	89,9979	1
45	6	Si	Si	1.5	88,58674	1	5	Si	Si	1.5	92,67306	1
35	7	Si	No	1.3	83,99376	1	5	Si	No	1.5	83,83652	1
30	6	Si	Si	1.1	80,65471	1	5	Si	No	0.7	82,01356	1

- Imagen reconstruida resolución 512x512

Proy	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM	Iter. LSQR	Filtro	Acel.	Alfa STF	PSNR	SSIM
180	5	Si	Si	0.5	96,15906	1	6	Si	Si	1.1	104,26394	1
120	5	Si	Si	1.5	93,9963	1	6	Si	Si	1.5	99,29937	1
90	5	Si	Si	1.3	90,91528	1	5	Si	Si	1	96,3268	1
60	5	Si	Si	1.3	77,27016	0,9999	5	Si	Si	1.1	85,22683	1
50	5	Si	Si	1.3	54,96583	0,9985	5	Si	Si	1.5	62,43473	0,99919
45	5	Si	Si	1.5	48,51764	0,99004	5	Si	Si	1.5	50,37792	0.95545
35	5	Si	Si	1.3	25,22686	0,49201	5	Si	Si	1.5	31,22229	0.52200
30	5	Si	Si	1.5	24,78289	0,44427	11	Si	Si	1	19,44698	0.03499

Basándonos en estos resultados tan dispares, únicamente se puede determinar que:

- El filtro no es conveniente aplicarlo para resoluciones menores a 128x128 píxeles
- La aceleración por norma general siempre mejora el método
- La métrica SSIM únicamente baja de 1 para imágenes de resolución grande y con pocas vistas, lo cual quiere decir que a nivel estructural el método conserva toda la información en los demás casos
- Para mayor resolución baja el número de iteraciones internas LSQR a aplicar. Esto se traduce a mayor número de iteraciones externas y por tanto más aplicaciones de filtro y aceleración
- Para la resolución de 512 las imágenes reconstruidas para menos de 45 proyecciones no tienen una calidad aceptable

Sin embargo, no se ha podido verificar una configuración óptima para cada resolución y número de vistas.

6. CONCLUSIONES.

Tras concluir este trabajo se puede determinar que el estudio exhaustivo que se ha realizado en Grid se adapta muy bien al paradigma de programación a aplicar en este tipo de entornos, dividiendo las ejecuciones en pequeños trabajos y ejecutando todos ellos de manera independiente y distribuida en diversos nodos de trabajo.

En cuanto a la calidad del método y la búsqueda de parámetros óptimos, no se ha conseguido llegar a verificar una configuración clara para cada resolución y número de vistas que se comporte generalmente bien. Por lo tanto, se plantea como trabajo futuro encontrar dicha configuración considerando otros parámetros como puede ser el número de iteraciones totales del método y detectando las imágenes que se encuentren en un rango aceptable de calidad, no solo la mejor de ellas.

AGRADECIMIENTOS.

Agradecemos la colaboración del profesor Damián Segrelles por sus sugerencias y facilitar el uso del GRID. Este trabajo fue soportado por el proyecto ANITRAN PROMETEOII/2014/008 de la Generalitat Valenciana de España, por el Ministerio de Economía y Competitividad español con la subvención TIN2015-66972 - C5-4 - R cofinanciado por fondos FEDER y ENE2014-59442-P-AR.

REFERENCIAS

European Grid Infrastructure. s.f. Septiembre de 2016. <<http://www.egi.eu/>>.

A. Beck, M. Teboulle. «A fast iterative shrinkage-thresholding algorithm for linear inverse problems.» *SIAM journal on imaging sciences* 2.1 (2009): 183-202.

E. Parceros, L. Flores, M.G. Sánchez, V. Vidal, G. Verdú. «Impact of view reduction in CT on radiation dose for patients.» *Radiation Physics and Chemistry* (2016).

Maier, A., Hofmann, H., Berger, M., Fischer, P., Schwemmer, C., Wu, H., Müller, K., Hornegger, J., Choi, J., Riess, C., Keil, A., Fahrig, R. «CONRAD—A software framework for cone-beam imaging in radiology.» *Medical Physics* 40.10 (2013).

Paige, C. C. y M. A. Saunders. «LSQR: An algorithm for sparse linear equations and sparse least squares.» *ACM transactions on mathematical software* 8.1 (1982): 43-71.

W. Yu, L. Zeng. «A Novel Weighted Total Difference Based Image Reconstruction Algorithm for Few-View Computed.» *PLoS ONE* 9.10 (2014).