

DISMEDI: UNA ARQUITECTURA MULTIUSUARIO, DISTRIBUIDA Y DE ALTAS PRESTACIONES PARA EL DIAGNÓSTICO POR IMAGEN

Carlos de Alfonso, Ignacio Blanquer*, Alejandro González*, Vicente Hernández**

** Univ. Politécnica de Valencia (UPV), Depto Sistemas Informáticos y Computación (DSIC), Grupo de Redes y Computación de Altas Prestaciones (GRyCAP) {calfonso, iblanque, agofau, vhernand}@dsic.upv.es*

RESUMEN

El objetivo del proyecto DISMEDI ha sido el desarrollo de un sistema servidor de cálculo para aplicaciones médicas distribuidas que pueda operar sobre una red DICOM. En este proyecto han participado dos hospitales de la Comunidad Valenciana (el Hospital de la Ribera, en Alzira y el Hospital Malvarrosa en Valencia) como expertos en radiología y como banco de trabajo para el sistema que se propone. Ambos hospitales constituyen dos ejemplos totalmente diferentes de modelo de gestión y volumen.

El sistema es accesible desde cualquier computador conectado a una intranet hospitalaria y permite el acceso a herramientas de altas prestaciones para el diagnóstico médico como la segmentación, reconstrucción 3D, reconstrucción multiplanar.

Palabras Clave: CORBA, Computación Paralela, Imágenes Médicas, Radiología.

INTRODUCCIÓN Y OBJETIVOS

El proyecto Europeo DISMEDI, (Distributed High Performance Processing of Medical Images. A New Component in Advanced PACS, <http://www.medicaltech.org/dismedi>) perteneciente al programa IST de la Comisión Europea (IST-1999-20778) tiene por objetivo el desarrollo de un sistema cliente/servidor para el diagnóstico por imagen, distribuido y de altas prestaciones. En este proyecto, además del Grupo de Redes y Computación de Altas Prestaciones (GRyCAP) y el Medical Image Laboratory (MedICLab) de la Universidad Politécnica de Valencia, han participado dos hospitales de la Comunidad Valenciana (el Hospital de la Ribera, en Alzira y el Hospital Malvarrosa en Valencia) como expertos en radiología y como banco de trabajo para el sistema que se propone. Ambos hospitales constituyen dos ejemplos totalmente diferentes. El hospital de la Ribera es un hospital de gestión privada, tecnológicamente maduro y dotado de un importante sistema de gestión de imagen médica digital. El hospital Malvarrosa es un buen ejemplo de hospital público modesto con un alto interés en soluciones de bajo coste.

Los objetivos del proyecto han sido:

- Desarrollar un dispositivo DICOM autónomo para proporcionar herramientas de altas prestaciones para el proceso de imágenes. El sistema es multiusuario, multiplataforma y paralelo.
- Desarrollar un ejemplo de programa cliente capaz de conectarse a este sistema y utilizar las herramientas que dispone.
- Fomentar el uso de la radiología digital en España.

El proyecto DISMEDI se integra en el cluster de proyectos de tecnologías de la información aplicadas a la medicina EUTIST-M (IST-1999-20226). Este cluster de proyectos, coordinado por el GRyCAP, ha apoyado las tareas de diseminación tecnológica y coordinación de los 11 proyectos asociados.

ARQUITECTURA DEL SISTEMA

El proyecto DISMEDI [Alf02, Alf01b] se ha construido sobre la arquitectura DISCIR¹, que permite la integración de sistemas de computación de altas prestaciones en un entorno distribuido.

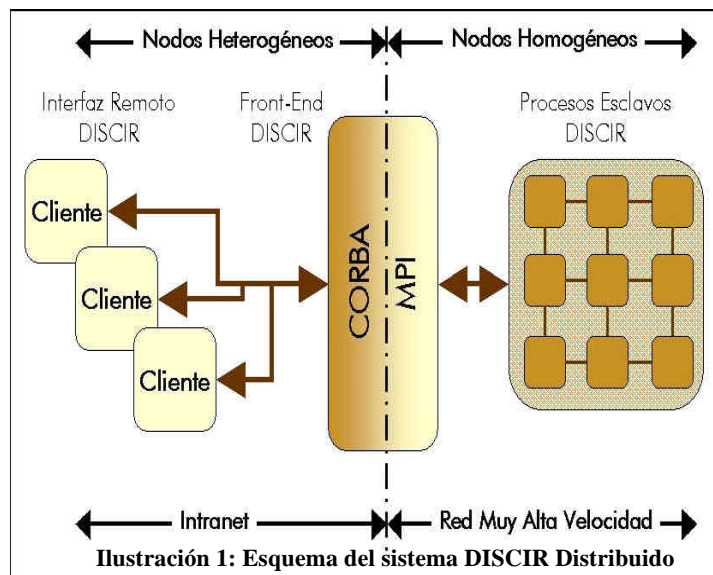
¹ El desarrollo de DISCIR fue parcialmente financiado por el MCyT, proy. CAPDIS, TIC2000-1165-C02.

Si bien la arquitectura de los sistemas distribuidos secuenciales se encuentra bien definida por estándares como CORBA, no ocurre lo mismo en el caso de componentes de computación paralela, donde la reutilización de las distribuciones de datos, el coste de transferencia de la información y el balance de carga juegan un papel muy importante. Existen diversas aproximaciones que intentan aportar a la metodología de CORBA una estructura que soporte estos componentes (COBRA, PACO, PARDIS, COVISE, TENT, CCA,...). La arquitectura DISCIR se orienta a permitir el acceso transparente a computadores de altas prestaciones de memoria distribuida desde computadores estándar.

La arquitectura DISCIR se compone de tres bloques principales.

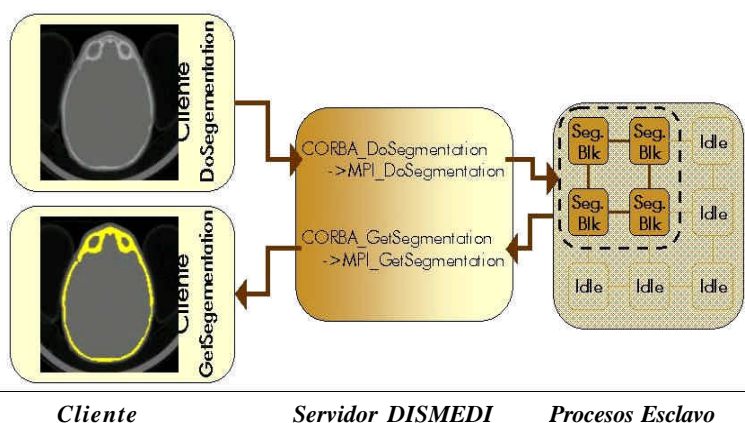
- **Servidor DISCIR.** Escrito en lenguaje “C” portable, constituye el componente maestro del sistema paralelo. Este módulo es el responsable de la conexión con los procesos del cluster de cálculo.
- **Proceso Esclavo PDISCIR.** Este módulo constituye el proceso esclavo del paralelismo simétrico y se ejecuta en cada uno de los nodos del cluster con excepción del ocupado por el servidor. Está codificado en lenguaje “C” portable y no depende de la arquitectura.
- **Interfaz Remoto (o Cliente) DISCIR.** Este módulo contiene el interfaz gráfico de DISCIR que permite el envío de peticiones de proceso al servidor del cluster.

La ilustración 1 muestra un esquema de la estructura del sistema DISCIR. La conexión entre el sistema servidor y el usuario se establece mediante el protocolo CORBA permitiendo la ejecución de objetos en servidores remotos con independencia del sistema operativo o arquitectura de los nodos origen y destino. El proceso servidor se encarga de implementar los objetos que atienden las peticiones remotas. Estos objetos servidores a su vez comunican con los procesos esclavos PDISCIR que se encuentran ocupando los diferentes procesadores del sistema de cálculo de altas prestaciones para llevar a cabo las operaciones en paralelo.



La comunicación entre el servidor y los procesos PDISCIR se realiza mediante MPI [Snir96], dado que el clúster constituye realmente una red de área local de altas prestaciones y totalmente homogénea. De esta forma se garantiza la eficiencia sin perder la portabilidad. Los clientes CORBA pueden encontrarse tanto en el propio computador donde se ejecuta el servidor como en computadores remotos conectados en red.

El sistema DISMEDI utiliza esta arquitectura para el proceso de imágenes médicas. La ilustración 2 muestra un ejemplo de la secuencia en la invocación de una operación de segmentación desde un cliente.



El cliente crea un objeto CORBA que incorpora un método ‘DoSegmentation’. Este método se ejecuta de forma remota en el servidor DISMEDI, manteniendo una imagen del objeto en el cliente. Al ejecutarse, la implementación del objeto CORBA ejecuta el método que contiene la implementación paralela del proceso (‘MPI_DoSegmentation’). Este método paralelo se ejecuta sobre el cluster produciendo unos resultados que son consultados por el cliente mediante un método CORBA (‘GetSegmentation’). Este método se traduce en el interno equivalente (‘MPI_GetSegmentation’).

El sistema DISMEDI incorpora la información sobre el paralelismo de forma intrínseca en la creación de los objetos remotos. El número de procesadores asociados a cada usuario se asigna en la creación del objeto que lo representa y esta configuración permanece constante durante la vida de dicho objeto. Todas las operaciones ejecutadas en paralelo (segmentación, proyección 3D) lo hacen sobre la distribución de datos existente y no requieren suministrar información adicional (número de procesadores, distribución de datos, tamaños de bloque,...). De esta forma se reutiliza la distribución de datos penalizando ligeramente la genericidad.

COMPONENTES DE DISMEDI

En esta sección se describen con más detalle los diferentes componentes del sistema DISMEDI.

Servidor DISMEDI

El proceso Servidor contiene tres funciones principales.

- ? **Recoger las peticiones que llegan de los clientes remotos.** Estas peticiones incluyen la selección y lectura de un estudio, la gestión de las acciones de segmentación, las tareas de preproceso de la imagen para la reconstrucción tridimensional y la proyección.
- ? **Acceder a la información almacenada en un servidor externo.** La información se almacena en un gran servidor de imágenes al que se accede a través de un protocolo DICOM [DCM01, DCM01b]. El uso de DICOM garantiza la portabilidad del sistema dentro del entorno hospitalario.
- ? **Gestionar los procesos esclavos.** La gestión de los procesos esclavos, incluyendo creación, distribución de datos, selección de la operación recogida de resultados, polling y destrucción.

Proceso esclavo PDISMEDI

El proceso PDISMEDI realiza todo el cálculo relativo a las operaciones computacionalmente costosas. En el presente artículo se describen los algoritmos paralelos de segmentación y proyección 3D.

El Componente Segmentación

La segmentación es la asociación de los diferentes vóxeles de la imagen a tejidos [Hara85, Lund97]. Existen métodos manuales, semiautomáticos y automáticos de carácter específico. De carácter general, los métodos semiautomáticos son los únicos factibles.

DISMEDI contiene un único objeto segmentación para cada volumen de datos. Cada objeto segmentación permite asociar hasta seis tejidos diferentes a cada píxel. Mediante máscaras es posible seleccionar un conjunto de segmentaciones como dato de entrada al componente de proyección.

El objeto de segmentación se encarga tanto de la gestión de los componentes relativos a la función de deshacer como de la gestión de las comunicaciones entre los procesos. La información distribuida de la asociación de tejidos a vóxeles se guarda en este objeto. La segmentación efectiva se realiza mediante otros objetos que interactúan con este objeto contenedor. Estos objetos (motores de segmentación) implementan de forma autónoma los diferentes métodos de segmentación y permiten una cierta transparencia en su uso.

Los métodos de segmentación que se han implementado son la segmentación por umbrales en 3D y la segmentación mediante crecimiento de regiones (*Region Growing*). La segmentación mediante umbrales determina el rango de intensidades que corresponde a cada tejido, mientras que la segmentación basada en crecimiento de regiones parte de un conjunto de semillas y expande este conjunto añadiendo al mismo aquellos puntos vecinos que además cumplen un cierto criterio de homogeneidad (similitud en los niveles de intensidad).

La implementación del algoritmo de crecimiento de regiones utiliza aceleraciones basadas en spans para aumentar la localidad de referencia y criterios de homogeneidad adaptativos [Bla00b] para reducir el efecto del ruido, que introduce huecos en las áreas segmentadas y produce sobresegmentación. La aproximación paralela distribuye el volumen de datos entre los diferentes procesadores e implementa un sistema de comunicaciones eficiente para intercambiar las semillas que trascienden del fragmento asociado a cada procesador.

El Componente Proyección

La proyección de imágenes médicas se centra bien en la extracción de modelos geométricos de los órganos de interés (surface rendering), bien en la proyección directa de los vóxeles de la imagen (volume rendering). Esta última aproximación produce imágenes de mayor relevancia para el diagnóstico [Dreb88, Fuc89, Lev90] y ha sido el objeto de estudio del proyecto.

La proyección 3D que incorpora DISMEDI tiene en cuenta las dos alternativas más aplicadas en la actualidad (*Splatting* [Dreb88] y *Ray Casting* [Bla99, Bla00]) y varias funciones de proyección. El componente proyección es el más complejo del sistema. Ante la necesidad de incluir diversos métodos para la proyección de la imagen, tipos de proyección, interpolación y perspectiva, se planteó el problema del mantenimiento del código. El objetivo es evitar que disponer de renderizado de superficies, estereoscopía y Máxima Intensidad, por ejemplo sobre *Ray Casting*, requiera replicar varias veces el código de *Ray Casting*. Siguiendo el planteamiento de DISMEDI se evita que cualquier optimización o cambio en el proceso de *Ray Casting* implique modificar todos los códigos.

Como solución a este problema se ha propuesto un sistema altamente desacoplado aprovechando las posibilidades que ofrece la herencia múltiple virtual. En el sistema DISMEDI existen diversos objetos proyección que implementan cada uno de los componentes diferenciados del proceso. Estos objetos son componentes derivados de la clase base que por sí solos no tienen autonomía (objetos virtuales). La autonomía se obtiene tras la fusión de un objeto de cada clase. De esta forma existen cuatro clases de objetos diferentes:

- ? Objetos Método de Proyección: Ray Casting, Splatting.
- ? Objetos Tipo de Proyección: Máxima Intensidad (MIP), Renderizado de Superficies (RS).
- ? Tipos de interpolación: Lineal y Trilineal.
- ? Tipos de Perspectiva: Proyección Ortogonal o Perspectiva Cónica.

Por tanto, un objeto capaz de realizar la proyección de la reconstrucción 3D de un volumen se obtendría derivando el objeto de cada una de las clases de forma conjunta.

Las aproximaciones paralelas permiten, bien distribuir el volumen de datos entre todos los procesos del sistema DISMEDI y realizar el proceso independiente de cada fragmento (aproximación en el espacio del volumen, método de *Splatting*) o bien distribuir el plano de la imagen y que cada proceso calcule un fragmento de la proyección (aproximación en el espacio del plano de proyección, método *Ray Casting*). El sistema además utiliza técnicas de compresión de datos [Alf01] para la codificación eficiente tanto de volumen de datos de la imagen como del plano de la imagen proyectada.

Interfaz Remoto DISMEDI

El servidor de cálculo DISMEDI actúa como motor para el procesamiento de las imágenes médicas, mientras que el acceso a las herramientas que ofrece se realiza de forma remota mediante un programa cliente. Este programa ofrece una interfaz de usuario, actualmente implementado sobre Visual C++ y operativo Windows que permite el acceso a las imágenes, la carga de las mismas, la segmentación de tejidos y la proyección 3D y multiplanar de dichas imágenes.

La estructura del sistema permite la implementación de clientes sobre cualquier plataforma al ofrecer una comunicación entre servidor y cliente basada en el protocolo CORBA.

Las operaciones de procesamiento 2D, como la adecuación del nivel y ventana (las imágenes médicas tienen una profundidad de 12-16 bits, mientras que los adaptadores gráficos y pantallas actuales admiten únicamente 8 bits) y brillo y contraste se realizan en los clientes. Todas las operaciones 3D se realizan de forma remota sobre el procesador paralelo. Esta distribución de tareas permite un uso más eficiente de los recursos de computación y comunicaciones. El interfaz básico se muestra en la ilustración 3.

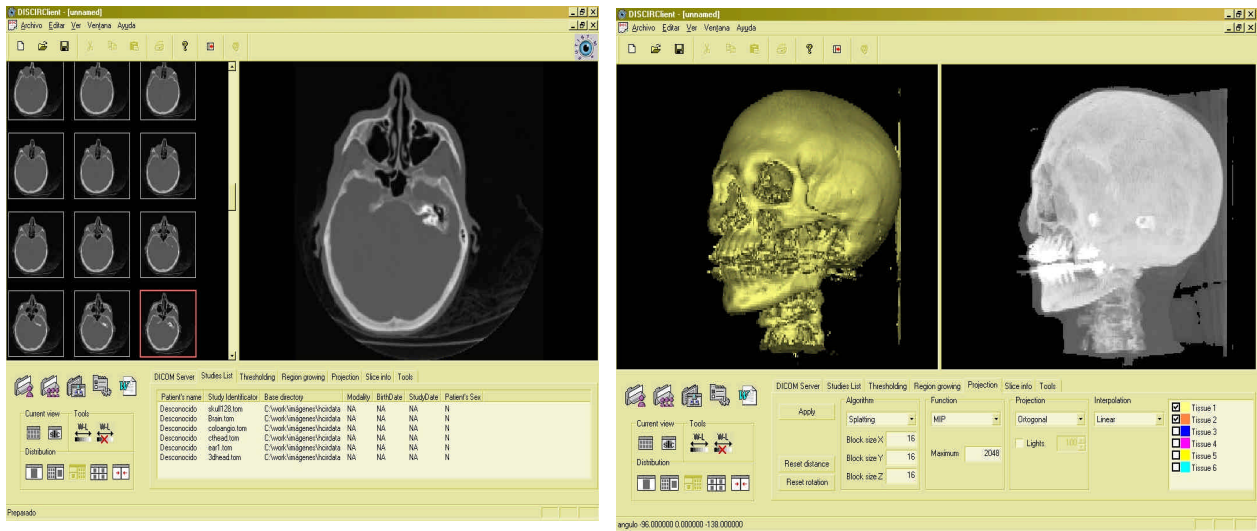


Ilustración 3: Dos vistas del interfaz cliente DISMEDI, multicorte (a) y dos proyecciones 3D (b).

El uso de un interfaz como un programa Windows autónomo presenta una serie de limitaciones:

- ? Dependencia de plataforma.
- ? Necesidad de instalar un software en cada cliente.
- ? Menor compacidad en la aplicación.

Sin embargo presenta numerosas ventajas:

- ? Mayor eficiencia y mejores prestaciones.
- ? Gran cantidad de recursos disponibles.
- ? Amplia disponibilidad del sistema operativo Windows.

RESULTADOS

En esta sección se muestran los resultados en tiempo de proceso de los algoritmos desarrollados que implementan la segmentación y la proyección 3D de imágenes médicas.

Prestaciones en la segmentación

Para la evaluación de las prestaciones del algoritmo de *Region Growing* se han creado varios conjuntos de semillas y se ha invocado directamente a los métodos con estos valores para permitir la reproducibilidad y la comparación entre diferentes configuraciones. Se han escogido 4 casos representativos de la batería de test y se ha calculado el tiempo de ejecución variando el número de nodos biprocesador desde 1 a 10. La ilustración 4 muestra dicha gráfica. Los casos que se han utilizado para la evaluación contienen entre 7 (CTC6) y 24 (MRC1) millones de vóxeles (puntos tridimensionales).

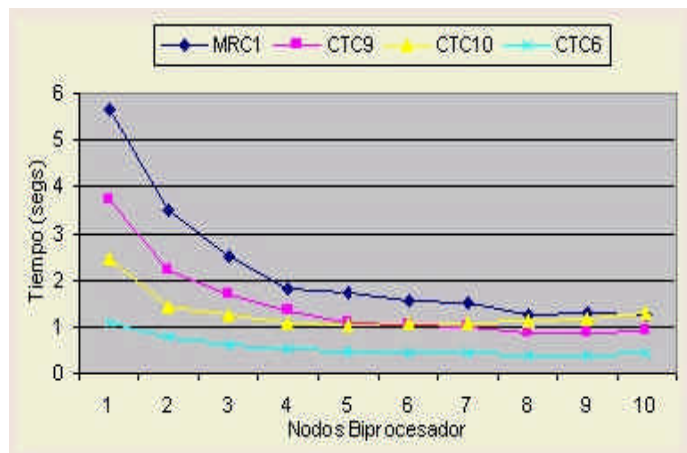


Ilustración 4: Tiempo de Ejecución del algoritmo paralelo de *Region Growing*

Prestaciones en la proyección 3D

Para la evaluación de los algoritmos de proyección se ha utilizado una segmentación del 70% de la imagen basada en umbrales y se han obtenido varias proyecciones con cada método utilizando diferentes puntos de vista. La media de estas proyecciones se muestra en las gráficas de la ilustración 6, variando el número de procesadores. Los casos utilizados coinciden con los del algoritmo de segmentación.

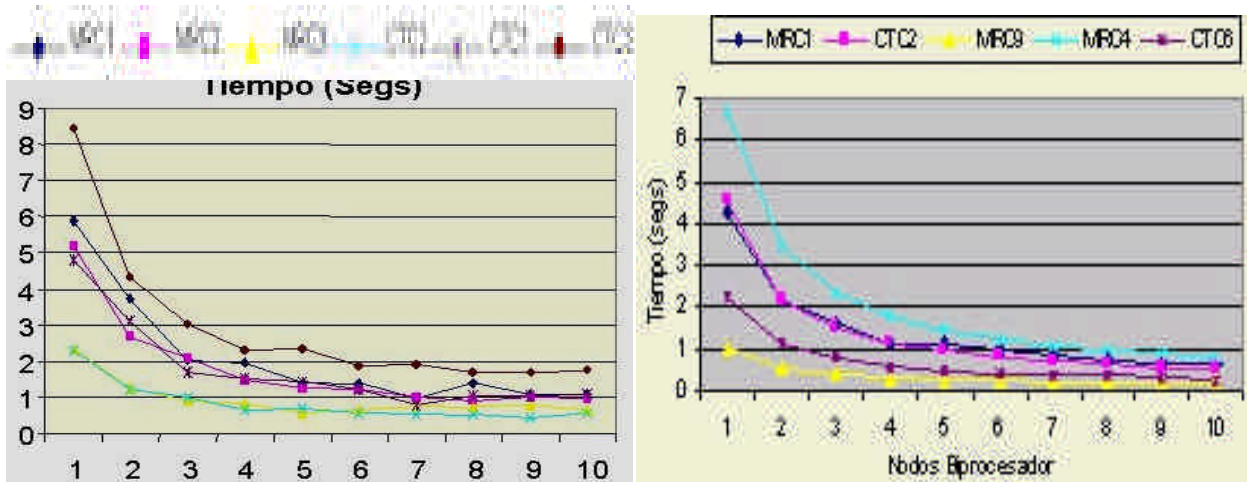


Ilustración 5: Tiempo consumido con *Splatting* (izda) y *Ray Casting* (dcha).

CONCLUSIONES

El presente artículo describe los resultados obtenidos en el proyecto DISMEDI. Los resultados combinan técnicas de imagen gráfica con computación de altas prestaciones, métodos numéricos y computación distribuida, constituyendo un trabajo multidisciplinar que supone además un interesante marco de trabajo para futuras acciones.

Durante el desarrollo de este proyecto se han implementado diversas versiones paralelas de algoritmos para la segmentación y reconstrucción 3D de imágenes médicas. Merece destacar del desarrollo del proyecto:

- ? El diseño de una arquitectura distribuida y orientada a componentes de software paralelo dirigida al diagnóstico por imagen pero con aplicación a otros muchas áreas.
- ? La implementación de dos algoritmos paralelos de proyección de imágenes mediante técnicas de volume rendering:
 - o Un algoritmo back-to-front que incorpora como novedades la orientación a bloques, la proyección de vóxeles comprimidos y la gestión de un raster comprimido.
 - o Un algoritmo front-to-back con terminación temprana del rayo y doble nivel de paralelismo (memoria distribuida / compartida).
- ? La implementación de un algoritmo paralelo de extracción de modelos geométricos basado en *Marching Cubes*.
- ? La implementación de un algoritmo de segmentación basado en crecimiento de regiones que incorpora expansión mediante spans 3D.

Los algoritmos han sido comprobados con un importante número de casos de test, que han permitido analizar el comportamiento de los mismos y la arquitectura en global, tanto en casos de práctica diaria como en casos de gran dimensión. Los algoritmos se han probado de forma extensiva en diversas plataformas, disponiendo de resultados para un número considerable de procesadores (hasta 24 agrupados en nodos biprocesador).

REFERENCIAS

- [Alf02] De Alfonso C, Blanquer I, Hernández V, “DISCIR: an architecture for high performance distributed and component-oriented image diagnosis applications”. Actas del Computer Assisted Radiology and Surgery International Congress and Conference, CARS 2002. Paris, Francia, Jun. 02.
- [Alf01] De Alfonso C, Blanquer I, Hernández V, “Reconstrucción 3D en Paralelo de Imágenes Médicas de Gran Dimensión Con Almacenamiento Comprimido”. Actas del Congreso Anual de la Sociedad Española de Ingeniería Biomédica, CASEIB 2001. Oct 01.
- [Alf01b] De Alfonso C, Blanquer I, Hernández V, “DISCIR: Una Arquitectura para la Implementación de Aplicaciones de Diagnóstico por Imagen Distribuidas y Paralelas Orientadas a Componentes”. Actas del Congreso Anual de la Sociedad Española de Ingeniería Biomédica, CASEIB 2001.
- [Bla00] Blanquer I, Hernández V, Ramírez J, “El Proyecto HIPERCIR: Procesamiento de Imágenes en Paralelo”, Vidal A., Actas del congreso “Congreso Anual de la Sociedad Española de Radiología”, SERAM 2000.
- [Bla00b] Blanquer I, Hernández V, Ramirez J, Vidal A., “Parallel Segmentation and Rendering Using Clusters of PCs”, Actas del congreso “Medicine Meets Virtual Reality, MMVR 2000”. (pags 33-35, ISBN: 1 58603 014 0).
- [Bla99] Blanquer I, Hernández V, J. Ramirez, Vidal A, “The HIPERCIR EU Project: fast MRI processing facilities at low-cost”, MAG*MA (Magnetic Resonance Materials in Physics, Biology and Medicine’, Septiembre 1999, pag 30, ISSN: 1352-8661/99.
- [DCM01] “Digital Imaging and Communications in Medicine (DICOM) Part 10: Media Storage and File Format for Media Interchange”. National Electrical Manufacturers Association, 1300 N. 17th Street, Rosslyn, Virginia 22209 USA.
- [DCM01b] “Digital Imaging and Communications in Medicine (DICOM) Part 9: Point to Point Communication Support for Message Exchange”. National Electrical Manufacturers Association, 1300 N. 17th Street, Rosslyn, Virginia 22209 USA.
- [Dreb88] Drebin R. A., Carpenter L, Hanrahan P., “Volume Rendering”, Computer Graphics, Vol. 22(4), August, 1988. 65-73
- [Fuc89] Fuchs, H., Levoy, M. y Pizer S.M. , “Interactive Visualization of 3-D Medical Data,” IEEE Computer, pp. 46-50, August, 1989.
- [Hara85] M. Haralick and L.G. Shapiro, “Image Segmentation Techniques,” Comput. Vision, Graphics, Image Processing, vol. 29, no. 1, pp. 100-132, Jan. 1985
- [Lev90] Levoy M. “Efficient Ray Tracing of Volume Data”, ACM Transactions on Graphics Vol 9. No.3 Juli 1990. 245-261
- [Lund97] Lundström C. “Segmentation of Medical Image Volumes”, Tesis doctoral por la Universidad de Linköping, Noviembre 1997.