CLOUD COMPUTING EN SALUD: SISTEMA PARA ADMINISTRAR IMAGENES BIOMEDICAS

R. ALONSO-CALVO[1], J. CRESPO[1], V. MAOJO[1], A. MUÑOZ[2], M. GARCÍA ROJO[3], L. PÉREZ[1], J. AZPIAZU[1]

[1]*DLSIIS & DIA – Grupo de Informática Biomédica, Facultad de Informática, Universidad Politécnica de Madrid, Campus de Montegancedo, Avda Montepríncipe, S/N, 28660 Boadilla del Monte (España), e-mail: ralonso@infomed.dia.fi.upm.es*
[2]*Departamento de Radiología, Facultad de Medicina, Universidad Complutense de Madrid, Ciudad Universitaria, Pabellón II, 28040, Madrid, (España).*
[3]*Servicio de Anatomía Patológica, Hospital General de Ciudad Real C/ Tomelloso s/n. Polígono Larache, 13005 Ciudad Real (España).*

Resumen. En el campo de la biomedicina se genera una inmensa cantidad de imágenes diariamente. Para administrarlas es necesaria la creación de sistemas informáticos robustos y ágiles, que necesitan gran cantidad de recursos computacionales. El presente artículo presenta un servicio de cloud computing capaz de manejar grandes colecciones de imágenes biomédicas. Gracias a este servicio organizaciones y usuarios podrían administrar sus imágenes biomédicas sin necesidad de poseer grandes recursos informáticos. El servicio usa un sistema distribuido multi-agente donde las imágenes son procesadas y se extraen y almacenan en una estructura de datos las regiones que contiene junto con sus características. Una característica novedosa del sistema es que una misma imagen puede ser dividida, y las sub-imágenes resultantes pueden ser almacenadas por separado por distintos agentes. Esta característica ayuda a mejorar el rendimiento del sistema a la hora de buscar y recuperar las imágenes almacenadas.

1. Introducción

Una de tarea fundamental en la informática biomédica es el tratamiento de señales e imágenes biomédicas. En la labor diaria de los profesionales e investigadores se generan una gran cantidad de información multimedia, como imágenes de Micro-Arrays, Anatomía Patológica, CT, MRI, PET, Rayos-X, etc. Además, los continuos avances en los sensores y dispositivos de captura que generan estas imágenes, provoca que tengan cada vez una mayor resolución, y por lo tanto mayor información contenida y mayor tamaño. Esto hace que los requisitos para el almacenamiento y el procesamiento las imágenes vayan creciendo y evolucionando de forma continua.

Normalmente los sistemas de almacenamiento de imágenes biomédicas —Picture Archiving and Communication Systems (PACS) — se implementan con bases de datos, ampliadas con técnicas de recuperación de imágenes por contenido – Content-Based Image Retrieval (CBIR). Hasta la actualidad existen multitud de propuestas de sistemas con recuperación de imágenes por contenido. Estos sistemas usan las características de bajo nivel de las imágenes, como son el histograma de color y el análisis de texturas. Dentro de este grupo de sistemas se encuentran algunos famosos como QBIC [1], Virage [2], VisualSeek [3] y Blobworld [4].

Una desventaja de la creación de estos grandes sistemas PACS es que son complejos y costosos ya que necesitan habitualmente una gran infraestructura informática debido, como se ha comentado anteriormente, tanto al continuo crecimiento del número de imágenes como al crecimiento del tamaño de las propias imágenes.

El presente artículo presenta un prototipo de servicio de cloud computing implementado mediante un sistema multi-agente capaz de indexar y almacenar grandes colecciones de imágenes biomédicas. Este servicio de cloud computing permite a sus usuarios acceder a recursos virtualizados de almacenamiento y procesamiento. En particular, el servicio ofrecido permite a los usuarios almacenar, recuperar e incluso procesar imágenes sin poseer los recursos físicos computacionales necesarios [5]. El sistema está basado en trabajos previos de nuestro grupo de integración de bases de datos usando ontologías de fuentes de datos estructuradas [6] y fuentes de datos desestructuradas [7][8] y en procesamiento de imágenes [9][10].

La distribución de grandes bases de datos de imágenes es algo habitual para mejorar la escalabilidad y estabilidad. Un aspecto novedoso que distingue el sistema presentado en este articulo respecto a otros sistemas, como los mencionados anteriormente, es que la distribución de los datos no se limita a dividir las distintas imágenes en varias bases de datos; además, cuando un agente del sistema recibe una imagen muy grande, pueden dividirla en varias sub-imágenes, y cada sub-imagen es enviada y almacenada de forma remota por distintos agentes. Esta característica ayuda a mejorar el rendimiento del sistema.

El artículo está organizado de la siguiente manera. En la siguiente sección se describe la metodología usada para crear el servicio de cloud computing basado en agentes distribuidos para almacenar y procesar grandes colecciones de imágenes biomédicas. El siguiente sección presenta los resultados de las pruebas realizadas sobre el sistema a la hora de realizar consultas y de obtener imágenes almacenadas en el mismo. Por último, la sección con las conclusiones, que concluye el artículo.

2. Metodología

2.1. Servicio de Cloud Computing

El cloud computing, según la definición hecha por el NIST [11], es una herramienta para ofrecer una interfaz para usar recursos virtuales bajo demanda. Dichos recursos virtuales son controlados por el propietario del servicio. Uno de los objetivos de nuestro proyecto es la creación un servicio de cloud computing para almacenar, procesar y recuperar imágenes e información multimedia biomédica. Dicho servicio puede ser usado por profesionales e instituciones que no posean los suficientes recursos técnicos, capacidad de almacenamiento o de computación para crear aplicaciones basadas en dicho servicio. Podría llamarse un PACS Virtual basado en el cloud computing.

El servicio creado es privado, y sólo puede ser accedido por clientes con cuenta en el sistema y que posean autorización de acceso. Un cliente puede introducir y almacenar imágenes en el sistema. Dichas imágenes son privadas y sólo accesibles por el propietario de las mismas. Sin embargo, los usuarios tienen la posibilidad de hacer públicas sus imágenes si lo desean. Usando este servicio un usuario u organización podría crear su propio PACS privado de una manera sencilla y sin necesidad de costosos recursos informáticos.

La implementación de un servicio de cloud computing debe ser escalable y eficiente para reducir los costes de procesamiento de tareas complejas. El prototipo presentado en este artículo está implementado mediante un sistema multi-agente distribuido y balanceado, lo cual permite escalar el sistema simplemente añadiendo nuevos agentes al sistema.

2.2. Estructura de datos y algoritmo de división

El núcleo principal para desarrollar un sistema de almacenamiento, análisis y recuperación de imágenes es una estructura de datos. Dicha estructura de datos debe proporcionar una forma eficiente para el almacenamiento y posterior acceso a toda la información obtenida de las imágenes. El sistema desarrollado usa una estructura de datos tipo grafo utilizada para almacenar la información de las regiones contenidas en una imagen. Cada región se define por las siguientes características:

- Descriptores de color: media, moda, varianza, mínimo y máximo de cada banda de color de la imagen (Rojo, Verde y Azul).
- Descriptores de forma: área, tamaño y orientación de los ejes (mayor y menor), centroide, estimación del número de lados y distancia de los bordes al centroide.
- Relaciones con regiones: adyacente-a, es-parte-de, relacionado-con y disjunto-de.

Con el fin de que esta toda la información obtenida sea accesible, tanto para realizar análisis y sobre ella como para posteriormente hacer búsquedas, se ha diseñado un esquema de base de datos relacional que implementa la estructura de datos de tipo grafo. Algunos operadores básicos de la morfología matemática (erosión, dilatación, watershed, etc.) han sido implementados para poder ser aplicados a sobre las imágenes almacenadas en el sistema.

Una condición muy deseable al realizar el procesado de una imagen de forma distribuida es que el resultado del procesamiento de la imagen inicial sea el mismo que la unión de los resultados parciales resultantes de aplicar el procesamiento sobre las sub-imágenes que componen la imagen inicial por separado. Sin embargo, esto no es posible conseguirlo en todas las operaciones sobre imágenes.

Se ha creado un algoritmo de división que cumple ciertos requisitos para cumplir esta propiedad para el tratamiento distribuido; en concreto, está pensado específicamente para operadores de morfología matemática basados en regiones. Las regiones iniciales de una imagen serán las zonas planas [9] de dicha imagen. A la hora de dividir una imagen no podemos usar simplemente una división lineal, ya que se podría separar una misma zona plana inicial en varias sub-imágenes, creando así nuevas regiones, lo cual alteraría el resultado de las operaciones. En el algoritmo de división creado, la imagen es dividida por su lado más largo, cogiendo los píxeles de la línea media y obteniendo las zonas planas que contienen a dichos píxeles. De esta manera el algoritmo devuelve dos sub-imágenes sin tratar I_A y I_C, que son enviadas a otros agentes para ser tratadas, y una tercera sub-imagen I_B ya procesada y lista para ser almacenada en la estructura de datos. Tanto I_A como I_C necesitan información de vecindad contenida en I_B, y viceversa. Por este motivo, los agentes llevan a cabo un paso de sincronización que completa la información en las sub-imágenes almacenadas en el sistema al terminar de procesar la imagen por completo. Gracias a este algoritmo, la imagen es procesada recorriendo solamente una vez su matriz de píxeles.

2.3. Arquitectura del sistema de agentes distribuidos

El prototipo del sistema se ha implementado usando el paradigma de orientación a agentes y se pueden distinguir tres roles o tipos de agentes:

- El agente ***punto de acceso al servicio de cloud computing***. Este agente es la entrada para usuarios autorizados al sistema. Ofrece métodos para (a) almacenar imágenes en el sistema, (b) aplicar operaciones sobre las imágenes almacenadas, y (c) consultar las imágenes del sistema.

- El agente ***índice de recursos*** puede ser visto como el servicio de directorio del sistema. Este agente contiene la información de la localización, carga actual, estado del procesamiento de las imágenes y predicción de tiempos de ejecución de todos los agentes existentes en el sistema.
- Los ***agentes trabajadores*** son la parte principal del sistema. Estos agentes implementan todas las funcionalidades ofrecidas por el sistema. Cuando reciben una imagen, comprueban si hay que dividirla o procesarla, realizando la operación necesaria. Son capaces de manejar la base de datos que contiene la estructura de datos. Pudiendo almacenar, filtrar y consultar las imágenes en la base de datos. Además, poseen funcionalidades para comunicarse con otros agentes trabajadores para enviar sub-imágenes para ser procesadas y para actualizar la información de la vecindad de las regiones almacenadas en su base de datos.

Un punto clave en el procesamiento distribuido es la definición de la política de asignación de las tareas a ejecutar en los diferentes nodos del sistema. Los beneficios del balanceo de carga en sistemas multi-agente y sistemas Grid, para optimizar el tiempo de ejecución de los trabajos, es una cuestión ampliamente estudiada [12] [13] [14]. En el presente sistema se ha implementado un planificador de trabajos y balanceo de carga que es usado principalmente para el paso de división y almacenamiento de las imágenes [15] [16]. Los filtros y operaciones no requieren de balanceo de carga puesto que se ejecutan directamente en aquella base de datos donde la sub-imagen se encuentra almacenada.

3. Resultados

En este apartado mostraremos algunas pruebas realizadas usando un conjunto de imágenes almacenadas en el sistema. Como ejemplo que ilustre el comportamiento del sistema, se han seleccionado tres imágenes de diferentes tamaños: la Imagen 1 de 1000x1000 píxeles, la Imagen 2 de 3000x3000 píxeles y la tercera Imagen 3 de 5000x5000 píxeles. Las imágenes se han almacenado distribuyendo cada imagen en un número diferente de agentes (desde 1 a 3 agentes). Con lo que cuando hay sólo un agente es que la imagen no se divide, sino que esta almacenada entera en una sola base de datos. Cuando la imagen es distribuida, el tiempo mostrado (en segundos) es el de aquel agente que más tarda en realizar la consulta.

	1 Agente	*2 Agentes*	*3 Agentes*
Imagen 1	0,41	0,26	0,21
Imagen 2	3,36	1,86	1,11
Imagen 3	15,9	8,08	5,2

Tabla 1: *Tiempos de ejecución para una consulta multi-criterio sobre las regiones de una imagen.*

	1 Agente	*2 Agentes*	*3 Agentes*
Imagen 1	100,00%	36,39%	48,80%
Imagen 2	100,00%	44,65%	66,96%
Imagen 3	100,00%	49,28%	67,30%

Tabla 2: *Porcentaje de mejora para una consulta multi-criterio sobre las regiones de una imagen.*

En la Tabla 1, se pueden ver los tiempos de ejecución para una consulta compleja. En concreto, dicha consulta obtiene de las distintas imágenes todas aquellas regiones que tengan (a) un área menor que una constante, (b) un valor de etiqueta (color) mayor que un umbral dado, y (c) tanto el eje mayor de la región como el menor sean más pequeños que el 10% del tamaño de la imagen (Este tipo de consultas pueden usarse por ejemplo para eliminar regiones no significativas, o ruido de la imagen). La Tabla 2 muestra en porcentaje la mejora de tiempo de ejecución con respecto a la ejecución usando un sólo agente de la misma consulta.

	1 Agente	*2 Agentes*	*3 Agentes*
Imagen 1	0,44	0,21	0,18
Imagen 2	1,81	0,6	0,69
Imagen 3	5,98	3,12	2,51

Tabla 3: *Tiempos de ejecución para la obtención de los píxeles de un recuadro.*

	1 Agente	*2 Agentes*	*3 Agentes*
Imagen 1	100,00%	52,28%	60,00%
Imagen 2	100,00%	55,81%	61,88%
Imagen 3	100,00%	47,30%	58,03%

Tabla 4: *Porcentaje de mejora para la obtención de los píxeles de un recuadro.*

La Tabla 3 muestra como se reduce el tiempo al obtener los píxeles contenidos en un recuadro de 300x600 píxeles de las distintas imágenes (cada píxel es obtenido con la etiqueta de la región a la que pertenece) y la Tabla 4 presenta la mejora de rendimiento para esta segunda consulta.

Como se puede observar, el tiempo de ejecución se reduce en ambas consultas al incrementarse el número de agentes en el sistema. Cuando se usan dos agentes el tiempo ahorrado es, de media, un 48% del tiempo que utilizando un sólo agente para ejecutar la operación. Cuando de usan 3 agentes el porcentaje de ahorro de tiempo asciende al 61% de media. Demostrando así que la división en sub-imágenes, de las imágenes introducidas en el sistema, es beneficiosa y ayuda a reducir el tiempo de ejecución en consultas de recuperación.

4. Conclusiones

En el presente artículo se ha presentado un prototipo escalable de sistema multi-agente capaz de almacenar grandes colecciones de imágenes biomédicas. Dicho sistema ofrece métodos para almacenar, analizar y recuperar las imágenes a través de un servicio de cloud computing, permitiendo así a los clientes del servicio la posibilidad de administrar y procesar fácilmente y sin la necesidad de grandes servidores sus colecciones de imágenes biomédicas.

En la sección 3, se aportan resultados experimentales que muestran de forma cuantitativa el aumento de rendimiento del sistema en algunas operaciones que se realizan comúnmente sobre las imágenes cuando el número de agentes del sistema es incrementado. Por lo tanto la distribución de la colección de imágenes en diferentes bases de datos como hacen multitud de sistemas CBIR reduce el tiempo de ejecución de las consultas. Así mismo, nuestro sistema también es capaz de dividir una misma imagen en varias sub-imágenes y almacenarlas de forma distribuida, para poder reducir aún mas los tiempos de las consultas.

Agradecimientos

Este trabajo ha sido financiado en parte por el “Ministerio de Ciencia e Innovación” de España (Ref.: TIN2007-61768).

Referencias

[1] Flickner M. et al.: Query by Image and Video Content: The QBIC System, IEEE Computer 28(9),23-32 (1995

[2] Bach J. et al: Virage Image search engine: An open framework for image management. In SPIE Storage and Retrieval for Image and Video Databases 2670,76-87 (1996)

[3] Smith J. and Chang S.: Blobworld: VisualSeek: A fully Automated content-based image query system. In ACM International Conference on Multimedia 87-98 (1996)

[4] Carson C., Thomas M., Belongie S., Hellerstein J., and Mallik J.: Blobworld: A system for Region-Based Image Indexing and Retrieval. Third International Conference on Visual Information Systems 1614,509-516 (1999)
[5] Langmead B., Schatz M. C., Lin J., Pop M. and Salzberg S. L.:Searching for SNPs with cloud computing. Genome Biology (2009)
[6] Alonso-Calvo R., Maojo V., Billhardt H., Martín-Sánchez F., García-Remesal M., Pérez-Rey D.: An agent- and ontology-based system for integrating public gene, protein, and disease databases. Journal of Biomedical Informatics 40(1), 17-29 (Feb 2007)
[7] Pérez-Rey D., Maojo V., García-Remesal M., Alonso-Calvo R., Billhardt H., Martín-Sánchez F. and Sousa A.: Ontology-based integration of genomic and clinical databases. Comput. Biol. Med 36(7-8), 712–30 (2006)
[8] Maojo V., García-Remesal M., Billhardt H., Alonso-Calvo R., Pérez-Rey D., Martín-Sánchez F.: Designing new methodologies for integrating biomedical information in clinical trials. Methods of Information in Medicine 45(2), 180–5 (2006)
[9] Crespo, J., Serra, J., and Schafer, R. W.: Theoretical aspects of morphological filters by reconstruction. Signal Process. 47 (2), 201-225 (Nov. 1995)
[10] Crespo, J., Schafer, R. W., Serra, J., Gratin, C., and Meyer, F.: The flat zone approach: a general low-level region merging segmentation method. Signal Process. 62 (1), 37- 60 (Oct. 1997).
[11] NIST: definition of Cloud Computing (v. 15) 2010. NIST (2010) Available via web. http://csrc.nist.gov/groups/SNS/cloud-computing/. Cited Apr 2011
[12] Leinberger W., Karypis G., Kumar V., and Biswas R.: Blobworld: Load balancing across nearhomogeneous multi-resource servers. Heterogeneous Computing Workshop 60-71 (2000)
[13] Cao J., Spooner D. P., Jarvis S. A., and Nudd G. R.: Grid load balancing using intelligent agents. Future Generation Computer Systems 21(1), 135-149 (2005)
[14] Yagoubi B. and Slimani Y.: Task load balancing strategy for Grid computing. Journal of Computer Science 3(3), 186-194 (2007)
[15] Alonso-Calvo R., Crespo J., Maojo v., García-Remesal M., Anguita A. : On distributing load in cloud computing: A real application for very-large image datasets. International Conference on Computational Science 1(1), 2663-2671 (May 2010), doi: 10.1016/j.procs.2010.04.300
[16] Alonso-Calvo R., Crespo J., Maojo v., García-Remesal M., Anguita A. : Cloud Computing Service for Managing Large Medical Image Data-sets Using Balanced Collaborative Agents . Accepted in: International Conference on Practical Applications of Agents and Multi-Agent Systems (Abr 2011)