Revista Digital Universitaria
1 de agosto 2011 • Volumen 12 Número 8 • ISSN: 1067-6079

ARTICULO

MINERÍA DE DATOS VISUAL

Laura Patricia Ramírez Rivera

Revista Digital Universitaria

1 de agosto 2011 • Volumen 12 Número 8 • ISSN: 1067-6079

Minería de datos visual

Resumen

Nunca antes en la historia se han generado tal cantidad de datos como en estos días. Explorar y analizar todos estos datos empieza a ser increíblemente difícil. La visualización y la minería visual pueden ayudar a tratar este enorme flujo de datos. La ventaja de la exploración visual es que el usuario es envuelto en el proceso de minería, lo cual permite una manera más intuitiva en el descubrimiento de la información.

Ese problema implica buscar la manera de aprovechar el uso de técnicas de minería visual, las cuales permiten el apoyo de la tecnología en pro de un buen entendimiento de los datos. Además es posible aprovechar las ventajas que ofrece una pared de video. Podemos usar una pared de video para desplegar múltiples gráficas de los mismos datos y así permitir su visualización en diferentes planos.

El objetivo del proyecto es aprovechar la tecnología para permitir el descubrimiento del conocimiento, a través del uso de una minería visual, con el apoyo de una pared de video, es decir se trata de aplicar un enfoque matemático-computacional, enlazando una correlación de Pearson con el manejo de una pared de video.

Palabras Clave: minería visual, pared de video, cluster.

Data visual mining

Abstract

This paper explains how it was implemented a method of visual data mining deals to maximize the advantages provided by the use of distributed object over a video wall, to facilitate the assessment of large amounts of data. It is seeking an alternative to the growing problem of having information and wanting to learn something from it so that they can minimize the costs in time and covers transportation, processing and visualization. Thats why we have selected the fundamental techniques in the process of mining as the coefficient of Pearson, and part of the management of distributed object has opted to use the advantages of the Mac platform. The main objective of this project is to create a system that allows for a friendly display large amounts of data.

Keywords: visualization, distributed object, visual data mining.

Introducción

La exploración de información en espacios heterogéneos requiere métodos de minería tan eficientes como sus interfaces visuales. [8]

Hace tiempo que la mayoría de los sistemas se concentraban o en los algoritmos de minería o en las técnicas de visualización. La desventaja que se tiene al separar el análisis de la visualización es que en general no tiene tanto apoyo una en otra, es decir, son aplicaciones diferentes que tratan de unirse, muy diferente a una aplicación cuya idea implique el uso de ambas desde un inicio. [4]

Para que la minería de datos sea efectiva, es importante incluir al humano en el proceso de exploración de los datos. En general es deseable combinar el conocimiento humano con la enorme capacidad de almacenamiento que se tiene en las máquinas así como su poder de procesamiento.

Las exploraciones visuales de datos tienen como objetivo la integración del humano en el proceso de exploración, gestión que tienen los sistemas computacionales. [6]

La finalidad es aprovechar las capacidades humanas, como la deducción y la intuición, permitiendo así obtener una minería más completa, tomando en cuenta las capacidades inherentes a nuestra especie. [7]

Para apoyar este tipo de exploraciones se propone el uso de una pared de video que permita una amplia visualización de los datos, al mismo tiempo que se realiza el proceso de minería. Es por eso que el objetivo final es crear un sistema que permita visualizar bases de datos científicas y mostrar los gráficos obtenidos de sus atributos sobre una pared de video.

Cabe recalcar que la pared de video es un medio tecnológico eficaz, que aún no ha sido desarrollado, ni estandarizado, por lo que involucra la investigación sobre su manejo, es decir, requiere conocimiento computacional que todavía no ha sido desarrollado, y del cual sólo se han encontrado ejemplos de uso e ideas y sugerencias para su implantación.

Existen diversas investigaciones para crear visualizadores de bases de datos, que varían en la forma en la que despliegan o manipulan los datos.

Estas investigaciones iniciaron desde hace tiempo, debido a que dentro de diversas áreas del conocimiento se han generado grandes cantidades de datos. En ocasiones son tan numerosos que su manejo es difícil.

1 de agosto 2011 • Volumen 12 Número 8 • ISSN: 1067-6079

Lo que necesitamos es localizar patrones que nos ayuden a ver cómo es el comportamiento de los datos y de ahí tratar de comprender la información contenida en ellos.

Existe también el problema del transporte en un tiempo relativamente corto. Se debe mencionar el tiempo de procesamiento que incrementa en proporción a la cantidad de datos que se presente.

Si unimos todos estos problemas tenemos uno que básicamente consiste en obtener conocimiento de grandes bases de datos científicas, disminuyendo en la medida de lo posible el costo computacional que esto representa.

En cuanto a la solución propuesta para estos problemas, planteamos lo siguiente. Podemos aprovechar las ventajas que la visualización nos proporciona para la mejor y más fácil apreciación de los datos. Desplegando los gráficos en un área mayor, como es una pared de video, podemos observar más de un gráfico a la vez, permitiendo una mejor apreciación del comportamiento de los datos.

Si permitimos el manejo de los objetos gráficos, a disposición del usuario, por toda el área de despliegue, el podrá hacer conjeturas mediante comparaciones evidentes. Se propone la correlación de *Pearson* [5] como el método que mostrará evidencia de comportamiento similar en los datos. Se propone el manejo de objetos distribuidos para evitar la interpretación centralizada.

Proyecto

Se han encontrado diversos trabajos que con tecnología especializada, como redes ópticas y grandes cantidades de memoria Ram, obtienen información de las bases de datos. Eso las hace en ciertas ocasiones demasiado complicadas de conseguir, debido a su alto costo, [9] [10][11] Es por eso que la idea de este proyecto es aprovechar en la medida de lo posible las características del manejo de objetos distribuidos en la plataforma Mac, para disminuir en lo posible costos. Ciertamente esta tecnología es más cara, pero a su vez permite emplear menos para obtener la más alta resolución, que es lo que necesitamos.

La idea es implementar un sistema que conste de 4 partes principales, que solucionarán en conjunto el problema que se describió anteriormente:

Parte 1. En cuanto a la obtención de la información de la base de datos, debemos tomar en cuenta el manejador de la base de datos. Debemos generar la conexión con el servidor, de forma que emplee el menor tiempo posible. Se debe verificar la forma en la que se manejarán los datos.

Parte 2. En cuanto a las operaciones que se realizarán para la obtención de la correlación de *Pearson*, se debe tomar en cuenta que los datos pueden ser manipulados desde un archivo, ya que la consulta se haya terminado. Debemos tener una forma segura para realizar las operaciones y evitar desbordes. Tomando en cuenta que no sabemos qué tan grande es la base a manipular, se debe ser cuidadoso en ese aspecto.

Parte 3. La parte que realiza el manejo de los objetos distribuidos, contiene una variante, debido a que el tipo de objetos que se manipularán son gráficos. El protocolo que se utiliza es cliente-servidor, donde los servidores ponen sus objetos a disposición del cliente, quien los manipula de manera transparente una vez que esté funcionando la conexión.

Parte 4. La generación del gráfico *OpenGL* representa la parte que ayudará al usuario con la apreciación de la información. Es una parte muy importante dentro de la solución.

Si unimos estas partes podremos generar un sistema que pueda obtener información de bases de datos de gran tamaño, generando su correlación y un gráfico que muestre su comportamiento y permita una buena apreciación.

Se realizará un sistema que permita el manejo de bases de datos científicas, para obtener sus atributos y generar la correlación de datos sobre 18 variables distintas.

Los resultados de las correlaciones se mostrarán en una matriz de colores, la cual permitirá generar las gráficas de los pares de variables, que el usuario desee. Los contenedores de gráficos serán de tipo *OpenGL*, y podrán moverse por el área de las pantallas que formen parte del cluster de visualización.

Un cluster de visualización es un conjunto de nodos que comparten su salida de video, con el objetivo de formar un dispositivo visual de mayor dimensión y resolución. Un cluster de visualización consta de un servidor de video, que es el encargado de hacer disponibles los objetos visuales, para que los clientes accedan a ellos y a los nodos que permitirán la interpretación y el despliegue de los mismos.

La computación distribuida es un modelo que permite resolver problemas de computación masiva, utilizando un gran número de computadoras organizadas en grupos incrustados en una infraestructura de telecomunicaciones distribuida, que ha sido diseñada para resolver problemas demasiado grandes para cualquier supercomputadora, mientras se mantiene la flexibilidad de trabajar en múltiples problemas más pequeños. Para que un cluster funcione como tal, no basta sólo con conectar entre sí los ordenadores, sino que es necesario proveer un sistema de manejo del cluster, el cual se encargue de interactuar con el usuario y los procesos que corren en él para optimizar el funcionamiento.

1 de agosto 2011 • Volumen 12 Número 8 • ISSN: 1067-6079

Los mensajes remotos en *C-objetivo* proporcionan un sistema en tiempo de ejecución, que permite establecer conexiones entre objetos en diferentes espacios de direcciones, reconociendo cuando un mensaje es invocado por un objeto en una dirección remota y transferir los datos de una dirección a otra. [1]

Usando objetos distribuidos, se pueden enviar mensajes en *C-objetivo* a objetos en otras tareas o tener mensajes ejecutados en otros hilos de la misma tarea. Para enviar un mensaje remoto, una aplicación debe primero establecer la conexión con el objeto receptor.

Un objeto gráfico debe ser eficiente en cuanto a su manipulación desde la interfaz. Debe además permitir la manipulación de diversos tipos de eventos. [2][3] Podemos aprovechar las ventajas conocidas en cierto tipo de objetos visuales, que permiten ciertas características, como manejo de eventos del ratón, para desplegar la información necesaria dentro de una aproximación a la minería de datos visual. En esta parte es necesario considerar los aspectos que incluyen su interpretación desde los datos, el tipo de gráfico que se generará, así como la forma en la que puede manipularse en tiempo real desde el servidor.

Otro aspecto importante incluye la dimensión que se manejará, ya que depende en gran medida del tipo de datos contenidos por la base de datos, pero puede también obtenerse como una simulación de su comportamiento, es decir, a partir de los datos obtenidos inferir un comportamiento y generar un gráfico representativo.

Conclusiones del sistema distribuido

Tomando en cuenta el diseño de este sistema distribuido, se llegó a una primera conclusión que indica que este tipo de manejo distribuido de objetos, es en si una buena forma de control para una pared de video, aunque se debe tener en cuenta que algunas características deben ser mejoradas para beneficiar el desempeño.

En cuanto a la escalabilidad, el sistema es funcional, debido a que es fácil agregar máquinas que permitan la ampliación de la pared de video, lo cual se probó paulatinamente con el incremento de máquinas.

Se obtuvo una solución que permite tener objetos visuales distribuidos, que pueden contener visualizaciones complicadas, elaboradas con tecnología de *Open GL*. Esto trae consigo que este tipo de objetos puedan contener gráficos simples en formatos comunes, como jpg, bmp, png. Tomando en cuenta lo anterior, obtuvimos una herramienta que permite la manipulación de imágenes sobre la pared de video.

La implantación de la red *Ethernet* fue más veloz que la de red inalámbrica en el inicio de la ejecución, pero una vez que están conectados el retraso ya no es por la red, sino por el número de mensajes enviados.

A pesar de haber perdido un poco de velocidad, este tipo de implantación permite el uso de menos recursos para su desempeño, por lo menos en hardware, lo cual puede darle cierta ventaja en ciertos casos.

La idea de usar índices en lugar de compartir todo un bloque de imagen, ha resuelto satisfactoriamente la desventaja del retraso en tiempo por el envío de mensajes, ya que el dato es un simple entero, aunque por otro lado sí se considera que la replicación de la información puede llegar a ser un problema en cierto momento.

No se implementó un sistema de archivos compartido, porque la estrategia de solución implica el manejo de la menor cantidad de información a través de los mensajes. La conclusión de esto es que sí es funcional, con sus respectivas mejoras.

Se logra tener la transparencia parcial dentro del sistema, demeritada solamente por la velocidad de despliegue de cada máquina. La confiabilidad del sistema fue comprobada y es buena aunque implique no ser tan bueno en otros aspectos.

La idea de aprovechar las ventajas de los objetos distribuidos y los objetos con- tenedores de gráficos de la plataforma de MAC OSX, fue buena y se concluye que con esto sólo se ha puesto la primera piedra para la construcción de un sistema que permita más opciones en un futuro.

Esta primera implementación sólo muestra datos en 2D, pero deja abierta las posibilidades de crear cualquier modelo complicado, con un mínimo cambio dentro de la clase del objeto visual.

Conclusiones de la minería visual

La minería que se desarrolló en este proyecto permite el manejo de los datos, de manera que el usuario decida qué parte quiere ver desplegada.

La interfaz que se emplea trata de evitar ambigüedades en la medida de lo posible. El manejo de una pantalla principal permite el movimiento de las gráficas en la pared de video.

La obtención de la correlación de *Pearson* nos permite obtener una información muy básica de los datos. Es buena si se trata de una primera exploración.

Las gráficas que se generan en este caso no son las más deseables, pero cumplen con su función informativa y son buenas para ejemplificar valores simples.

1 de agosto 2011 • Volumen 12 Número 8 • ISSN: 1067-6079

Conclusiones de los gráficos

Los gráficos que se obtuvieron en este sistema no son de alta calidad y no usan ni la mitad de las capacidades que tiene, por su generación con la biblioteca de *OpenGL*. Se pueden mejorar de una manera no tan compleja, ya que están implementados.

Los contenedores que se usan para el despliegue, demostraron ser eficientes en cuanto a la capacidad de manejo que permiten, así como por las capacidades gráficas con las que cuentan. Se puede pensar en funciones como *zoom* o como *resize*, que no serían tan complicadas de establecer gracias a la implementación que está ahora en funcionamiento.

Etapa de minería de datos

La minería de datos está lejos de ser un área totalmente descubierta, ya que dentro de ella se engloban muchos aspectos muy importantes; por ejemplo la manera en la que se puede descubrir conocimiento dentro de un conjunto de datos no explorados.

El tipo de datos que se pueden manipular son tan variados, que es necesario encontrar maneras para su manejo. Si hablamos de multimedia no sólo debemos encontrar una forma para identificarlo, sino además para su procesamiento y análisis preliminar.

Hablamos de memoria, de tiempo en el procesador y de técnicas en el envío de red para distribuir esta carga que tiende a ser cada vez más pesada, por el tamaño de la información y por el tipo. Para ejemplificar un poco estas consideraciones, basta hablar de las manipulaciones de imágenes de gran tamaño. Ahora, si en lugar de imágenes tenemos video, es mayor el incremento en el procesamiento y el almacenamiento.

Cabe hacer notar que dentro del manejo de diversos tipos de información, se ha dado solución a muchos problemas mediante el uso de metadatos, lo cual en sí tiene su propio problema anexado con respecto a cómo se deben utilizar, es decir, cómo tratar los datos que no tienen una estructura lineal, evitando en lo posible las ambigüedades que la misma estructura puede traer consigo.

Una vez que se pueda resolver el problema del tipo de datos que se manipulará y disculpando el problema que implica la estructura que los contendrá, está el problema de la comunicación de la aplicación con la base de datos.

Etapa de comunicación con la base de datos

En este trabajo aprovechamos las ventajas de la definición de objetos distribuidos, dentro del lenguaje de programación *objetive-C*. Es por eso que la comunicación entre los mensajes la

hacemos mediante el protocolo ip, definido por el lenguaje.

Realmente la forma en la que se comunican en este trabajo no causa problema, debido a que los mensajes que se envían no contienen datos más grandes que un entero o una cadena, debido precisamente al paradigma de solución que se implantó, pero si lo vemos desde un punto de vista más amplio, es decir que la aplicación realmente envía datos más grandes como imágenes o video.

La solución debe cambiar para utilizar un tipo de protocolo que permita la transmisión de paquetes de datos más grandes. Un ejemplo de esto es un proyecto que planea generar mapas dinámicamente, desde una base de datos geográfica. Hablamos de obtener en tiempo real la información de cada uno de los puntos que formen el mapa.

En otras palabras, trataremos de obtener la descripción de cada píxel, de forma que pueda verse y manipularse cada uno de ellos con sus propias características. Hablamos de 1920x1200 pixeles por cada pantalla conectada.

Además hablamos de una manera tal de comunicación, que permita a cada máquina estar consciente de su posición con respecto a las demás, para así obtener los datos que requiere para desplegar su parte.

Se trata de evitar el uso de un servidor centralizado que envíe órdenes a cada cliente. Se trata de que cada nodo sepa qué parte de imagen debe desplegar con respecto a las demás pantallas conectadas. De esta forma se generará el mapa en forma distribuida. El problema se queda abierto, definido como la manera en la que un cluster de visualización puede generar un mapa sin necesidad de un control externo.

Etapa de generación de gráficas

En esta solución se generan gráficos simples en dos dimensiones, debido a que el tipo de método de minería elegido maneja pares de variables. Aun así se han usado objetos que poseen las capacidades de *OpenGL*, con lo cual surgen muchas ideas que pueden realizarse en trabajos posteriores, mediante la generación de modelos más complejos, en 3D, con texturas que permitan una mejor apreciación del comportamiento de los datos.

En un futuro se propone tomar en cuenta más de dos grupos de variables. Se propone la búsqueda de modelos adecuados que permitan el mejor uso de esta tecnología de visualización. Existen características que pueden ser aprovechadas de mejor modo si se toma otra posición en cuanto al desarrollo de las gráficas. Se hace evidente que usarlo en este caso parece un desperdicio de tecnología.

1 de agosto 2011 • Volumen 12 Número 8 • ISSN: 1067-6079

Pero si tomamos en cuenta que este trabajo se puede tomar como parte de uno a futuro, que requiera emplear este tipo de gráficos, podemos aceptar el hecho de que éste sea el primer paso, y es mejor darlo con lo mejor que se tiene, aunque esté un poco desaprovechado.

Podría evitarse el retraso en la generación del gráfico si utilizamos primitivas básicas del contenedor estándar y generamos una imagen en baja resolución, la cual puede ser manipulada de forma más fácil.

Etapa de manejo de objetos distribuidos

Recordando un poco podemos encontrar, que:

Sistema distribuido es aquel en que los componentes de hardware y software, localizados en computadoras y unidos mediante una red, comunican y coordinan sus acciones mediante el paso de mensajes.

Esto trae consigo las siguientes consecuencias:

• Concurrencia

• Inexistencia de un reloj global

• Fallos independientes

Por otro lado podemos encontrar una de las definiciones más aceptadas de sistema distribuido, como:

Un sistema distribuido es un grupo de computadoras independientes que son percibidas por los usuarios como una sola.

De ahí encontramos que los objetos distribuidos son módulos de software que son diseñados para trabajar juntos, pero están ubicados en múltiples máquinas, conectadas mediante la red. Un objeto envía un mensaje a otro objeto en una máquina remota. El resultado se envía al objeto que lo llamó. Ahora bien, sobre la solución que se le dio a este concepto, cabe recordar que la idea fue aprovechar las ventajas que ofrece la tecnología del lenguaje *objetive-C*. Es por eso que se manejan tal y como son propuestos.

Podemos mejorar este tipo de mensajería si conocemos explícitamente cómo se manejan. Podríamos en otro contexto tratar de implementar este concepto con otro protocolo, evitando en lo posible los retrasos, pero esto, claro, requiere de una amplia investigación a este respecto.

Las características que debe cumplir un objeto distribuido se pueden limitar a las que podemos

realmente ofrecer. Sabemos que con respecto al concepto más general, un objeto distribuido no está en una máquina, sino en varias y de ahí que surjan la concurrencia y los demás problemas inherentes al concepto de distribuido.

Una idea que surge de la implementación realizada, hace sugerir una manera en la que el objeto realmente permanezca en varias máquinas a la vez.

A continuación se muestra la imagen de la pared de video, donde se probo (figura 1), la cual se encuentra en el laboratorio de computación científica, del departamento de computación, del CINVESTAV. La información contenida en este artículo se encuentra incluida dentro de la tesis de maestría llamada Minería de datos visual.

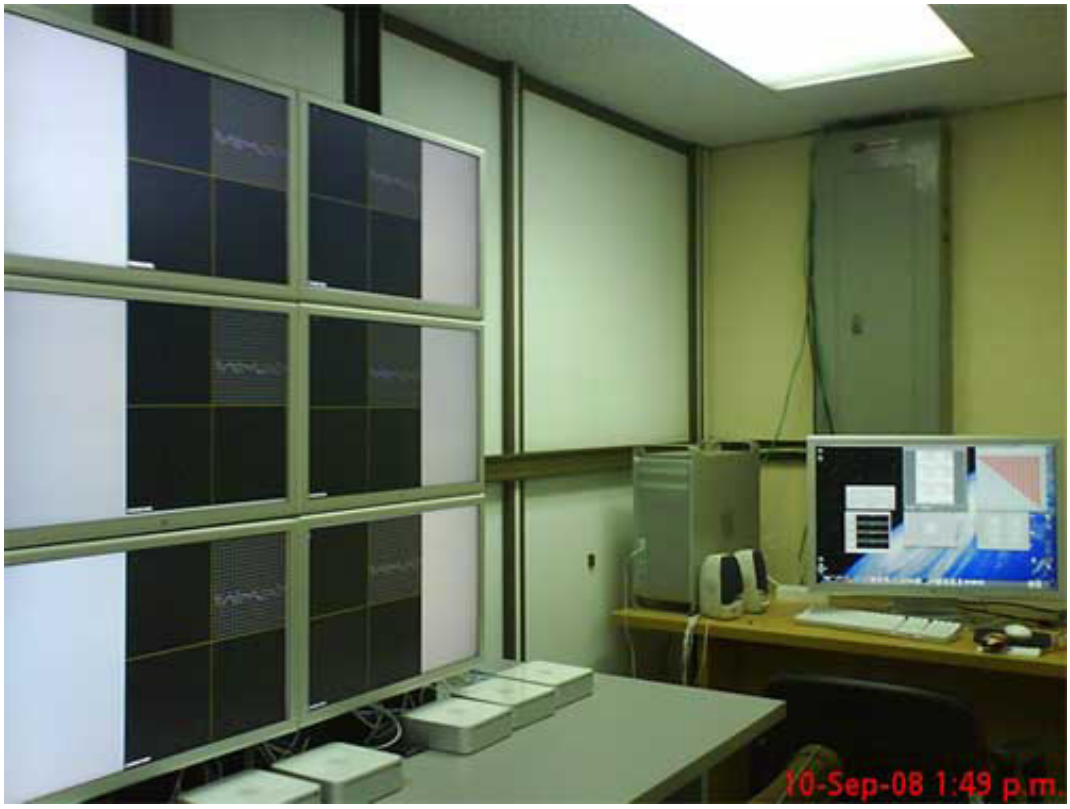

Figura 1. CinvesWall

Conclusión

La minería de datos visual es una nueva disciplina que surge a partir de la necesidad de explorar grandes cantidades de datos, ya que mediante la visualización de los datos, la búsqueda del conocimiento se realiza de un modo más práctico, empleando características como la intuición humana para evitar cálculos innecesarios. Las paredes de video son herramientas de visualización cientí¬fica que proporcionan procesamiento paralelo y alta resolución. Si empleamos la pared de video en la minerí-a de datos visual, tendremos una de las mejores herramientas para la exploración de datos masivos.

1 de agosto 2011 • Volumen 12 Número 8 • ISSN: 1067-6079

Bibliografía

[1] Apple Inc. Distributed objects programming topics cocoa interapplication communication.2003, 2007 Apple Inc., 2007. http://developer.apple.com/library/mac/documentation/cocoa/

[2] Apple Inc. Nsopenglview class reference cocoa user experience. 2003, 2007 Apple Inc., 2007. http://developer.apple.com/library/mac/#documentation/GraphicsImaging/Conceptual/OpenGL-MacProgGuide/opengl_drawing/opengl_drawing.html

[3] Apple Inc. Nsview class reference cocoa graphics and imaging. Apple Inc.2003, 2007 Apple Inc., 2008.

http://developer.apple.com/library/mac/#documentation/Cocoa/Reference/ApplicationKit/Classes/NSView_Class/Reference/NSView.html

[4] Cesar Ferri Ramírez José Hernández Orallo, Maria José Ramírez Quintana. Introducción a la minería de datos, volume 19. Marcel Dekker. INc., 2004.

[5] George W. Snedecor and William. Métodos estadísticos. CECSA, 1977.

[6] Mason J. Katz William J. Link Philip M. Papadopoulos, Caroline A. Papadopoulos and Greg Bruno. Configuring large highperformance clusters at lightspeed: A case study. IEEE Computer Graphics and Applications, 2005.

[7] Oreste Verta Domenico Talia, Paolo Trunfio. Weka4ws: a wrsf enabled weka toolkit for distribuited data mining on grids. Data Mining Grid digital library, 2005.

http://grid.deis.unical.it/weka4ws/

[8] Pak Chung Wong. Visual data mining. IEEE Computer Graphics and Applications, pages 2–3, 2002.

[9] Stephen W Michnick Kirill Tarassov. ivici: Interrelational visualization and correlation interface.

http://michnick.bcm.umontreal.ca/ivici/

[10] Singh Rajvikra. Sage: the scalable adaptive graphics environment.

http://www.evl.uic.edu/core.php?mod=4&type=1&indi=281

[11] Thomas A. DeFanti Chong Zhang, Jason Leigh. Terascope: Distributed visual data mining of terascale data sets over photonic networks. 19:935 – 943, 2003.