

Infraestructuras de bajo costo para la visualización de datos a gran escala y resolución, una opción accesible para instituciones educativas

Infraestructuras de bajo costo para la visualización de datos a gran escala y resolución, una opción para accesible Instituciones Educativas

Jesús Alberto Verduzco Ramírez
Instituto Tecnológico de Colima
javrtesis@yahoo.com.mx

Nicandro Farías Mendoza
Instituto Tecnológico de Colima
nmendoza@ucol.mx

Gilberto René Martínez Bonilla
Instituto Tecnológico de Colima
subdiracademica@itcolima.edu.mx

Pedro Rocha Medrano
Instituto Tecnológico de Colima
procha@itcolima.edu.mx

María Isabel Sáenz Rodríguez
Instituto Tecnológico de Colima
su_saenz16@hotmail.com

RESUMEN

La necesidad de disponer de superficies de visualización gráfica de datos de mayor tamaño y resolución que aquellas proporcionadas por el monitor se hace evidente al utilizar aplicaciones que generan cantidades significativas de datos gráficos como: la visualización científica, los entornos de la realidad virtual y aumentada, el diseño en ingeniería, el análisis de gráficas en finanzas, etcétera. El presente capítulo tiene como objetivo, en la primera parte, hacer una reseña de las tecnologías propietarias en el área de las plataformas de visualización intensiva de datos que emplean alta escala y alta resolución. En la segunda parte, se presentan las experiencias al construir tres sistemas de visualización de bajo costo, que pueden ser utilizadas como reales

alternativas a las costosas infraestructuras propietarias en apoyo a las actividades académicas en instituciones de educación con presupuesto limitado.

Palabras clave: visualización de datos, proceso gráfico distribuido, cluster, muro de imágenes, cave.

RESUMO

La necesidad de disponer de superficies de visualización gráfica de datos de prefiteo tamaño y resolución that aquellas proporcionadas por el monitorar se hace evidenciados al utilizar aplicaciones that generan cantidades significativas de datos Gráficos Como: la visualización Científica, los entornos de la realidad y Virtual Aumentada, el diseño en Ingeniería, el análisis de gráficas en Finanzas, etcétera. El presente Capítulo tiene Como Objetivo, en la primera parte, hacer una de las tecnologías reseña propietarias en el área de las Plataformas de visualización intensiva de datos that emplean Alta escala y Alta resolución. En La Segunda parte, se presentan las experiencias al Construir tres Sistemas de visualización de bajo costo, that pueden Ser utilizadas Como reales alternativas a las costosas infraestructuras propietarias en apoyo a las Actividades Académicas en Instituciones de educación con presupuesto limitado.

Palabras clave: visualización de datos, proceso Gráfico distribuido, cluster, muro de imágenes,, cave.

Fecha recepción: Septiembre 2013

Fecha aceptación: Octubre 2013

Introdução

O monitor é um dispositivo utilizado para exibir informações gráficas para um usuário de computador. No entanto, do ponto de vista das suas características físicas, os monitores têm evoluído lentamente. Monitores LCD atualmente substituir os monitores CRT, mas isso ocorre apenas alguns anos atrás. Em contraste, a maioria dos componentes do PC, como capacidades de memória, processador e disco têm visto a sua quase o dobro a cada ano, enquanto a resolução dos monitores só tem aumentado a uma taxa de 5% ano nas últimas duas décadas (Chen Y., Chen H., Clark, DW, Liu, Z., Wallace, G. & Li, K, 2001)

As dimensões da resolução do monitor e constituem um obstáculo para determinados tipos de aplicações cuja principal característica é a quantidade significativa de dados gráficos de geração. A visualização de dados científicos, ambientes virtuais e realidade aumentada, design de engenharia e análise de gráficos em finanças, são exemplos de aplicações que exigem superfícies de exibição gráfica maiores do que aquelas fornecidas pelo monitor. Rodando em um sistema com gráficos limitados poder, um aplicativo não pode explorar a sua plena capacidade na geração do ambiente gráfico. Como resultado, a exibição de gráficos complexos e interação com o usuário de processamento são fortemente perturbados. Por exemplo, um usuário interagir com um aplicativo que exibe mapas, é forçado a exibir frações do quadro total se você quer analisar os detalhes da imagem, isso envolve a perda da visão do contexto global, situação agravada ainda mais se envolvendo vários usuários.

O interesse em aumentar o tamanho e a resolução da superfície de exibição tem sido nos últimos anos. As soluções atuais de usar um centralizado ou uma arquitetura distribuída. A abordagem centralizada é fornecer um único nó de múltiplas placas gráficas e consequente alimentação mais do que um monitor. Como um bônus, temos um baixo custo e fácil de implementar; No entanto, este sistema sofre de alguns inconvenientes. Um nó executa o processo gráfico (rendering), o que significa baixa eficiência. Um problema importante é o número limitado de PCI e AGP portas disponíveis no nó. Uma solução mais atraente é a de dividir o processamento gráfico entre um grupo de nós interligados por uma rede de comunicação de dados e distribuir as imagens e gráficos de primitivas utilizando um protocolo eficiente. Neste modelo, os principais problemas são o consumo excessivo de largura de banda da rede e sincronização das atividades dos nós. Apesar destes inconvenientes, desenvolvendo projetos atualmente empregados como processo de distribuição de solução gráfica.

Sistemas de display gráfico de projetores de vídeo baseados tais como: exposição de parede (Li K., H. Chen, Y. Chen, DW Clark, P. Cook, 2000), Workbench (Wolfgang Krüger, Christian Bohn-A, Bernd Fröhlich Heinrich Schüth & Wolfgang Strauss, 1995) e CAVE (C. Neira Cruz, Daniel J. Sandin & Thomas A. Defanti 1993), têm sido amplamente utilizados na visualização de dados.

Estes sistemas permitem que aplicativos exibam imagens em alta resolução e grande acompanhados de visão estereó para aumentar o grau de imersão dos usuários. As seções a seguir fornecem uma análise dessas três soluções usando esta abordagem se torna.

Parede de exibição

Uma exposição de parede é um sistema capaz de exibir imagens de alta resolução e grande (Li K. et al., 2000). A imagem global é gerado em uma ou mais superfícies de um conjunto de projetores de vídeo, que são alimentados por tipo de gráfico emite um supercomputador SGI Onyx. Na Figura 1, uma parede de exposição consiste em oito superfícies de vídeo independentes iluminados por holofotes mostrados. Cada projector é responsável pela geração de uma fracção da imagem global projectada. Uma exposição de parede pode exibir imagens 2D em grandes superfícies, o que facilita detalhada, por exemplo, na medicina, imagens de satélite, arquitetura, etc. análise.



Figura 1. Display Wall gerado por un arreglo de ocho video proyectores

Responsive Workbench

O Workbench Responsive (W. Krueger et al., 1995), é uma ferramenta para gerar imagens 3D estereoscópico que são projetadas em duas superfícies perpendiculares vista com a ajuda de um sistema que consiste em um projetor e espelhos. O usuário pode interagir com aplicativos desenvolvidos usando luvas e um mouse 3D. Além disso, um sistema de monitoramento monitora a posição da cabeça do usuário para que esta exibição do ambiente virtual a partir do ângulo direito.

A bancada é altamente apreciado trabalhar com modelos eletrônicos para a proximidade que tem com a ferramenta mãos do usuário. A Figura 2 mostra um utilizador utilizando o Workbench.

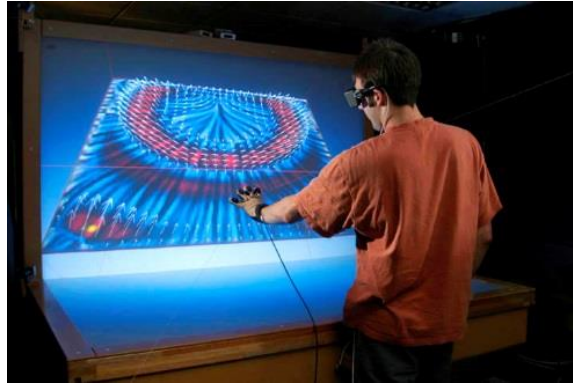


Figura 2. Interacción con una aplicación 3D ejecutándose en el Workbench

CAVE

O sistema CAVE é uma exibição de alto desempenho de gráficos 3D consiste em um cubo de seis lados. Os rostos são iluminados a partir do exterior por projetores de vídeo alimentados por canais de gráficos supercomputador, em que as aplicações que produzem o gráfico exibido nas superfícies da caverna prazo. Usuários localizados dentro da caverna fornecidos óculos 3D, está praticamente cercada por imagens estéreo acompanhado contribuindo para aumentar o grau de imersão. As aplicações se concentrar em modelos em escala de exibição CAVE de fenômenos físicos.

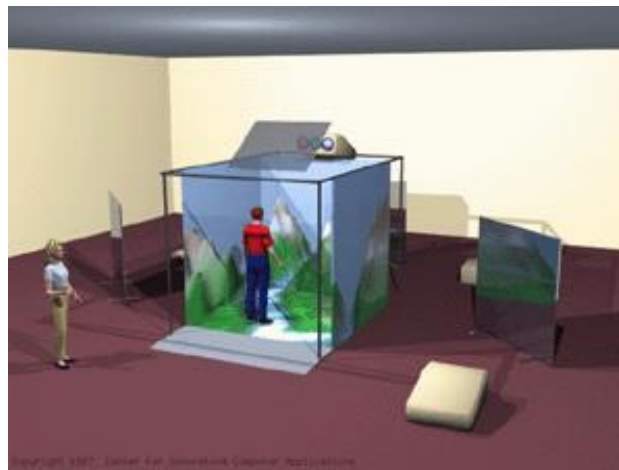


Figura 3. Estructura de una CAVE

Soluções de baixo custo

Computadores tradicionalmente aplicações de uso intensivo de visualização de dados têm sido utilizados equipado tecnologia proprietária de alta performance. Os três sistemas discutidos na seção anterior CAVE, exposição de parede e bancada compartilham uma característica comum: eles usam tecnologia proprietária como projetores de vídeo 3D e supercomputadores de classe SGI Onyx, resultando em um aumento significativo em seus custos tornando-os inacessíveis para as organizações orçamento. Uma alternativa atraente é o uso de computadores de cluster, ou seja, usar o poder de processamento adicional de um conjunto de equipamentos de informática e conciliar a sua capacidade de armazenamento e processamento de gráficos como uma alternativa econômica para supercomputadores caros.

O aumento tendência em rápida evolução do desempenho e baixo custo de componentes do PC, incluindo placas gráficas, faça o cluster uma alternativa viável para aplicações que requerem processamento de dados intensivo. A infra-estrutura deste tipo também são econômicos, pois você pode reutilizar equipamentos existentes nas organizações, por exemplo, nas instituições de ensino.

VISUALIZAÇÃO DE PLATAFORMAS PARA BAIXO CUSTO DISTRIBUÍDO

Atualmente, existem várias plataformas que seguem os componentes distribuídos integrados com baixo custo, um dos mais destacados é Grimage (Jérémie Allard, Clément Menier, Bruno Raffin, Edmond Boyer & François Faure, 2007) abordagem de exibição, que consiste de quatro elementos : um conjunto de processamento de gráficos, um sistema de imagem, uma biblioteca para desenvolvimento de aplicações paralelas e uma superfície de tela grande. Nesta plataforma, você pode comprar um objeto a ser instantaneamente entrou e modelagem 3D em um mundo virtual com o qual você pode interagir. A Figura 2 mostra imagens de Grimage.



Figura 2. Funcionamiento de la Plataforma Grimage

Na Figura 3 é mostrado em pormenor um diagrama da operação de Grimage. Em primeiro lugar, fluxos de vídeo a partir de um conjunto de câmaras são capturadas, o objecto desejado não surge e cada elemento de fundo é subtraído, portanto, um modelo 3D do objecto é criado capturado e processado em uma superfície de exibição grande.

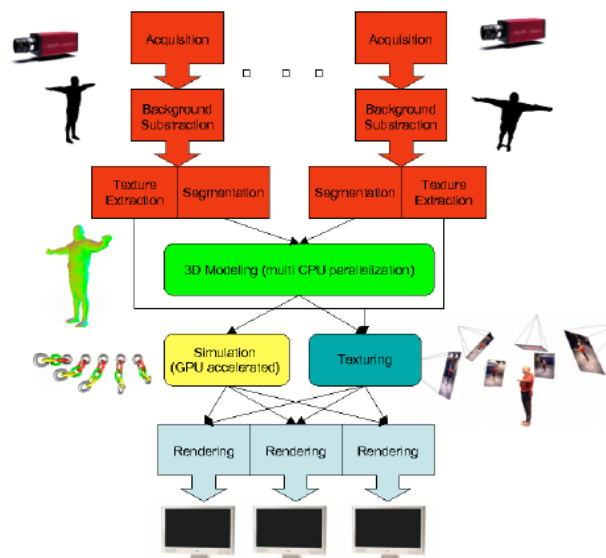


Figura 3. Esquema de funcionamiento de Grimage

Grimage necessária para operar uma equipe de alta performance, a configuração existente no INRIA Rhone Alpes é composto por 6 câmaras de alta definição, uma parede de imagens a partir de 16 projectores de vídeo e 2 clusters interconectados, um com 11 dual-Xeon de 2,6 GHz para

aplicação e outra com 16 Opteron dual-2 GHz para a parte de apresentação. Para a distribuição de mensagens entre processos e estes, Grimage usa uma biblioteca de software chamado FlowVR, que é descrito na secção seguinte.

FlowVR

FlowVR (Jérémie Allard, Valérie Gouranton Sebastien Liimet, Emmanuel Melin, Bruno Raffin & Sophie Robert, 2004) é uma biblioteca que fornece aos usuários as ferramentas para desenvolver e executar aplicações interativas em Computação de Alto Desempenho ferramentas de cluster Grades s. As principais aplicações desta biblioteca são as soluções de realidade virtual e visualização científica. FlowVR facilita a programação modular que reduz as dificuldades de engenharia de software, permitindo execuções de aplicações de alto desempenho em ambientes paralelos e distribuídos. Arquitetura e aplicativos são mostrados na FlowVR e 7.

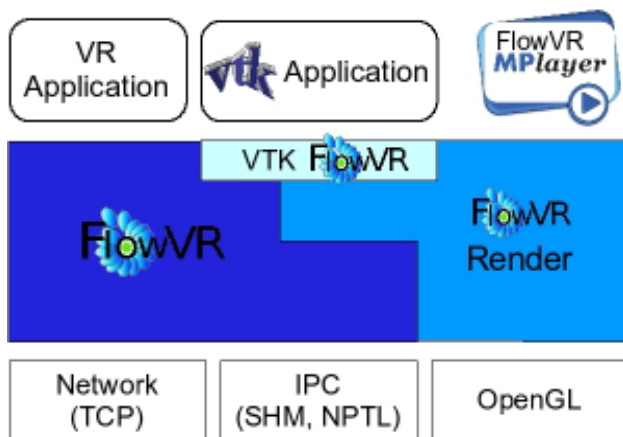


Figura 4: Arquitectura de FlowVR

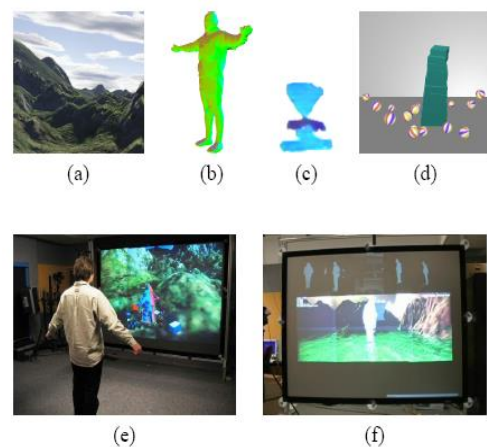


Figura 5. Diferentes aplicaciones de FlowVR

Na Figura 5 mostrado FlowVR aplicações diferentes:

(A) a prestação de terra, (b) modelagem de um corpo de câmera, (c) distorção de objetos, (d) simulação de corpos rígidos, todos eles podem ser combinados em uma aplicação rodando em um cluster.

SISTEMAS DE EXIBIÇÃO DE IMPLEMENTAÇÃO DE BAIXO CUSTO

Nesta secção, a construção de três protótipos utilizando componentes de baixo custo de operação é descrito. Nós definimos a baixo custo, como o uso de componentes de arquitetura aberta, tais como PCs, projetores de vídeo, cluster de computadores, também chamados de componentes de commodities, como uma alternativa à infra-estrutura proprietária caros, como supercomputadores.

Prototype 1: Imagem da parede de baixo custo

Nesta seção, vamos descrever o trabalho feito para implementar uma parede de imagens usando componentes baratos. Uma parede de imagens é um sistema capaz de exibir imagens de alta resolução em grandes superfícies. A imagem global é gerado em uma ou mais superfícies de um conjunto de projetores de vídeo, que são alimentados pelas saídas gráficas dos gânglios que constituem um cluster. Na Figura 8 uma parede de imagens compostas de quatro superfícies independentes (chamado de azulejos) mostrados. Cada nó de cluster é responsável por gerar para cada rede de imagem e transmissão de dados de azulejos move primitivas gráficas e comandos de sincronização entre eles. No conjunto, há um ou mais nós que permitem a interação do usuário com os aplicativos, usando o teclado eo mouse ou por outros meios, tais como reconhecimento de gestos, acompanhamento de posição do usuário e assim por diante.

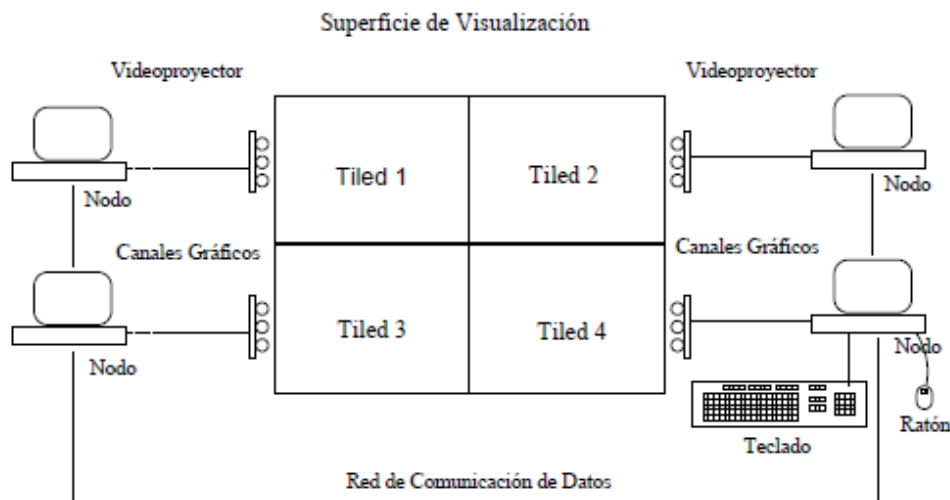


Figura 6. Arquitectura de un muro de imágenes de bajo costo

A solução de proxy

Nesta seção, vamos descrever o trabalho feito para espalhar a exibição do X Window Server (Robert W. & Jim Gettys Scheifler, 1986) e visualizar gráficos de X Window (clientes X) aplicação em vários nós em um cluster. As mensagens produzidas pelos clientes X precisam ser modificados e distribuídos entre o grupo de nós. Do mesmo modo, as mensagens de regresso, a interação do utilizador do produto deve ser recuperado e levado para X. O cliente arquitectura física do sistema (ver Figura 9) é constituída por:

Nó Master: Host uma instância em execução do proxy. Este nó deve imperativamente conectar os clientes, usando o soquete em que o proxy escuta.

Renderização nós: Executar uma instância do servidor X Window e tem a missão de receber os pedidos gráficos enviados pelo cliente através do proxy, execute o processamento e visualização posterior.

Cada nó executa o processamento da fração da imagem que corresponde à vista. Com este procedimento, o conjunto de monitores ou projetores, organizados no âmbito de um acordo especial tem uma superfície de exibição no qual as imagens são clientes implantado.

Nó interação: Permite que dispositivos de dados de entrada, teclado e mouse, pelo qual o usuário pode interagir com aplicações. Este nó também realiza a exibição gráfica.

Visor Virtual A.

A primeira tarefa do nosso protótipo é feito na época de seu lançamento, é para definir a configuração da tela virtual. O usuário fornece linha de comando na lista de nós e da prestação disposição armazenados no visor virtual. O proxy envia primitivas gráficas de visualização (XOpenDisplay) tornando cada servidor, a fim de recuperar as suas características gráficas, particularmente as dimensões da tela de abertura. Com esta informação, as construções de proxy e determina a resolução da tela virtual, por exemplo: 4 servidores de renderização A, B, C e D cuja exibe estão disponíveis 1,280x960 pixels e forma um arranjo 2x2, irá formar um monitor virtual 2,560x1,920 pixels, como mostrado na Figura 9. No momento em que um cliente X

conecta, o proxy envia as características do monitor virtual. Com esta informação, o cliente X acredita que ter uma área de visualização maior do que a prevista pelo monitor do PC no qual ele está sendo executado.

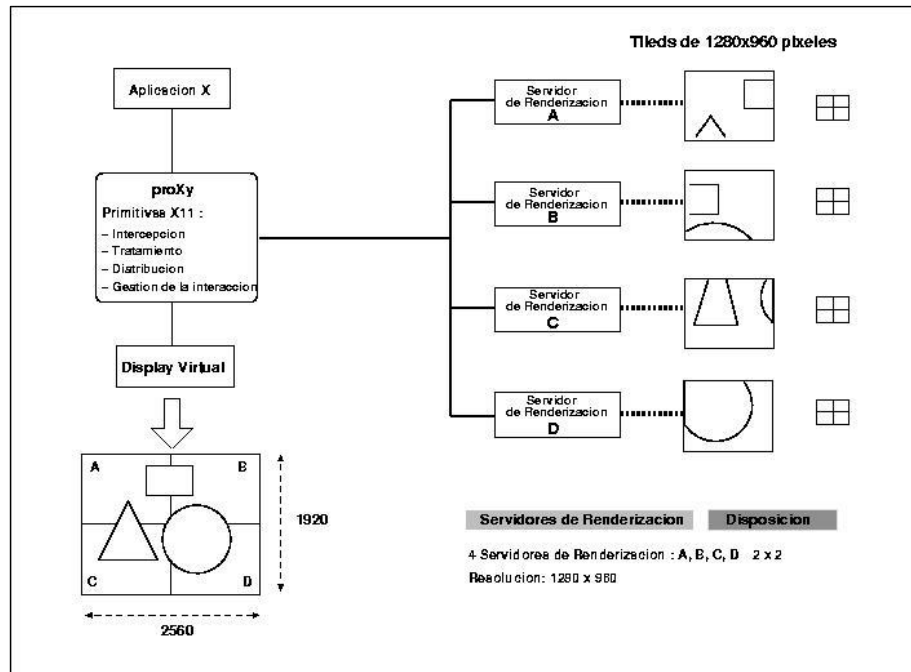


Figura 9. Estructura del Display Virtual en proXY1

B. Tornando o fluxo

Nós desenvolvemos duas estratégias para o tratamento de gráficos de primitivas. A primeira estratégia é aplicada pelo protótipo Proxy1 chamado e envolve a interceptação e coordenar adaptação realização primitivas gráficas, a fim de se ajustar ao âmbito da visualização virtual. Esta solução é expressa pela seguinte expressão:

$$X(u,v) = X_{virtual} - U * X \quad (1)$$

$$Y(u,v) = Y_{virtual} - V * Y \quad (2)$$

Onde (1) X, Y identifica a resolução da tela virtual (u, v) servidor tornando posição numeradas de (0,0), (Xvirtual, Yvirtual) as coordenadas de um ponto no visor virtual e (X (U, V), Y (u, v)), as coordenadas correspondentes no que diz respeito à prestação servidor (U, V). Graças às ações da

função de recorte tornar nodos feitas somente a prestação do monitor virtual cada administrada. 9 mostra este procedimento.

A segunda solução é utilizada para o protótipo chamado Proxy2, esta proposta amplamente descentralizar funções que adaptam à prestação primitivas nós de gráficos. Para este fim, Proxy2 constrói uma janela visor em cada nó de prestação. A característica especial desta janela é que tem uma dimensão semelhante à exibição virtual. Esta janela é colocada em cada nó render o intervalo de coordenadas de visualização virtual de cada nó é responsável pela composição de exibição, o resto da janela é eliminado pela função de recorte.

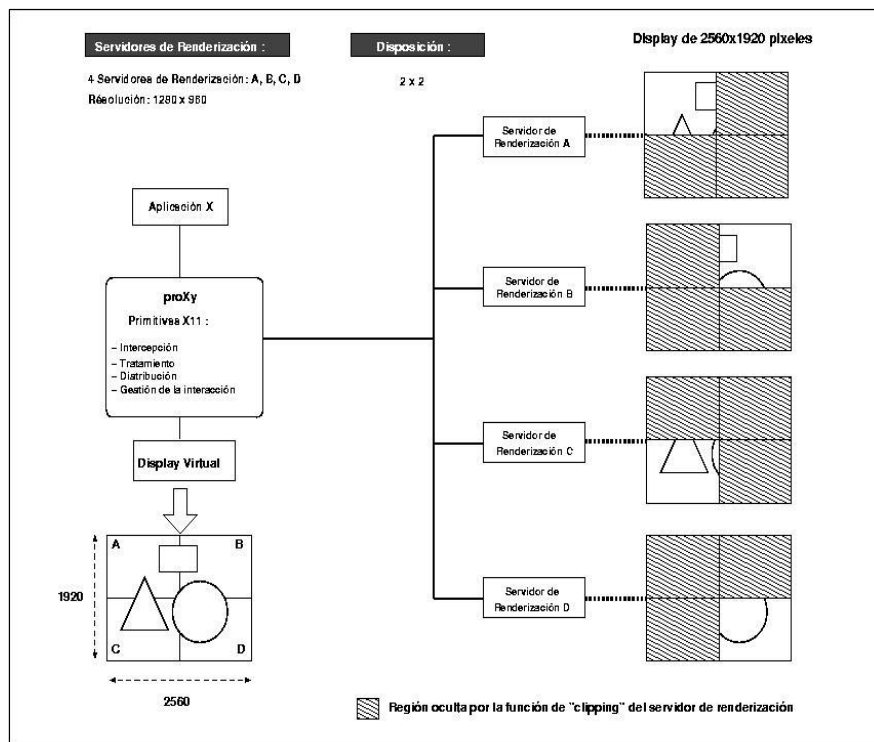


Fig. 10. Construcción del Display Virtual en la solución proXY2

C. Fluxo de interação

Para recuperar as ações do usuário sobre o mouse eo teclado mensagens de interação nó cruzam. O servidor X tem o mecanismo de focagem para determinar a janela em que os resultados de render e interação deve ser dirigido.

Embora o foco permanece confinado à eventos de interação nó azulejos teclado e mouse são

tratados localmente. A situação muda quando o usuário arrasta o cursor do mouse para um vizinho azulejos. Em seguida, os eventos gerados no servidor de interação deve ser dirigida ao nó possuir tornando o foco naquele momento. Nós usamos a extensão xtext apresentar no servidor de X Window que lhe permite enviar eventos para um display remoto. A nossa solução baseia-se este código. Os resultados são mostrados na Figura 11.



Fig. 11 Muro de imágenes de bajo costo en el Tecnológico de Colima

Discussão dos Resultados

Nesta seção, vamos mostrar o desenvolvimento de um servidor proxy que recebe mensagens dos clientes X Window System e distribui-los a um conjunto de servidores X, que cooperam para construir a imagem do cliente, obtendo como resultado de gráficos maiores e maior resolução.

Para implantar a exibição do servidor X Window, usamos duas estratégias. A primeira consiste em adaptar o contexto gráfico do monitor virtual primitivo, esta tarefa é realizada no chamado Proxy1 protótipo. A segunda estratégia é implementado no protótipo Proxy2 é para executar o processo de adaptação de render os nós.

Com esta implementação, os benefícios para o usuário por ter uma ferramenta que exhibe gráficos dos aplicativos X Window System em grandes áreas, aumentando a quantidade de informação que pode ser percebido e melhorar o processo de interação com o usuário. Uma vantagem desta

solução é que ele pode ser implementado com elementos presentes na academia, por exemplo, o Instituto Tecnológico de Colima, com esta infra-estrutura estará disponível para exibir imagens 2D em tamanhos grandes.

II Prototype 2: plataforma barata para interação com objetos virtuais

Esta seção apresenta a segunda desenvolvimento, uma plataforma para executar aplicações de realidade aumentada.

A nossa solução é dividida em dois módulos funcionais:

- Um módulo para a aquisição e distribuição das imagens capturadas com webcams.
- Um módulo para o tratamento de renderização fluxo de vídeo e misturado com cenas interativas em 3D.

Nesta primeira fase, para testar a viabilidade da nossa ideia, que consiste em apenas dois nodos protótipo foi implementado, de aquisição e de nó do nó render interligados por uma rede, como mostrado na Figura 12.

O Node aquisição

O nó de aquisição liga uma ou duas câmaras de web, que adquirem imagens do meio ambiente. Um servidor armazena vídeo e transmite essas imagens para renderização nó.

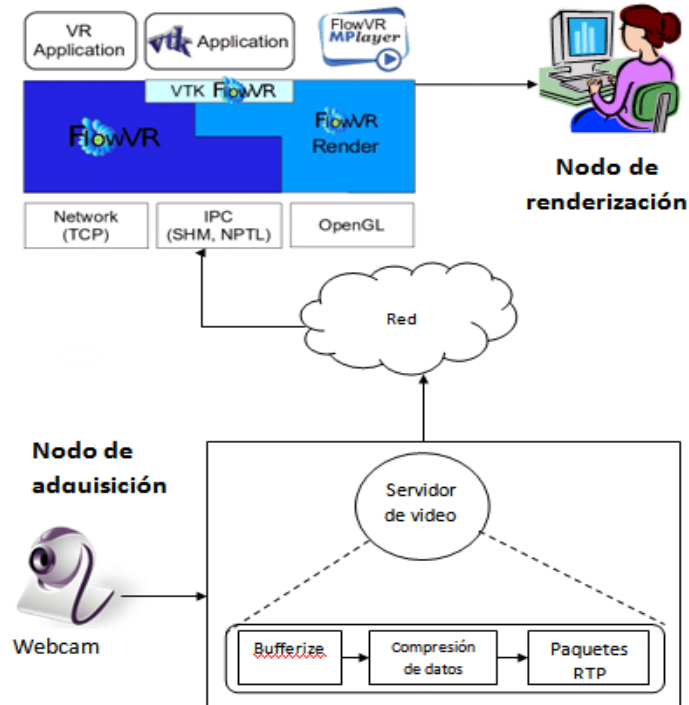


Figura 12. Arquitectura del prototipo de realidad aumentada de bajo costo

Renderização nó

O nó tornar executando uma instância do FlowVR, que recebe quadros de vídeo, executa o processamento e visualização posterior através FlowVR imerso no MPlayer.

Desempenho

O nó aquisição captura imagens e envia o fluxo de vídeo ao nó de processamento. Como servidor de vídeo, usamos o Mago software (SUSTO, 2008), um software gratuito que permite o envio de transmissões de vídeo através de redes IP. O software leva susto imagens armazenadas no buffer, os comandos sequencialmente, comprime os dados e envia-lo através da rede. Enquanto isso, na prestação de nó, o cliente inicia o jogador imerso em FlowVR MPlayer. Desta forma, você pode visualizar as imagens captadas pela webcam na prestação nó.

A mistura das imagens captadas pela webcam e a geração de cenas 3D é fornecido pelo componente FlowVR renderem, o que permite a combinação de transmissão de vídeo com um ambiente virtual render FlowVR.

Descrição do aplicativo

Para testar a funcionalidade da nossa plataforma, temos desenvolvido alguns exemplos em que o reconhecimento dos contornos capturados pela webcam e interação com esses objetos sem o uso de quaisquer objetos de interação do dispositivo é mostrado. Para ambas as operações têm usado a biblioteca (Intel Corporation, nd) OpenCV.

Reconhecimento Contour

Um esboço é uma lista de pontos que representam uma forma ou de outra, a curvatura de uma imagem. Esta representação pode ser diferente dependendo das circunstâncias. Há muitas maneiras de representar. Em openCV contornos são representados por sequências em que cada entrada da referida sequência codifica a informação sobre a localização do próximo ponto na curva.

Para a visualização dos contornos de uma imagem, é necessário em primeiro lugar um método para a gestão da memória. OpenCV possui uma entidade chamada memória de armazenamento, que é um método para gerenciar a localização de objetos dinâmicos na memória. O tipo de objecto que pode ser armazenado em uma memória de armazenamento é uma sequência. As sequências são listas de outras estruturas de dados vinculado.

O primeiro exemplo é o reconhecimento desenvolvido o esboço de uma série de cartas e desenhos à mão livre. O resultado é mostrado nas imagens seguintes..

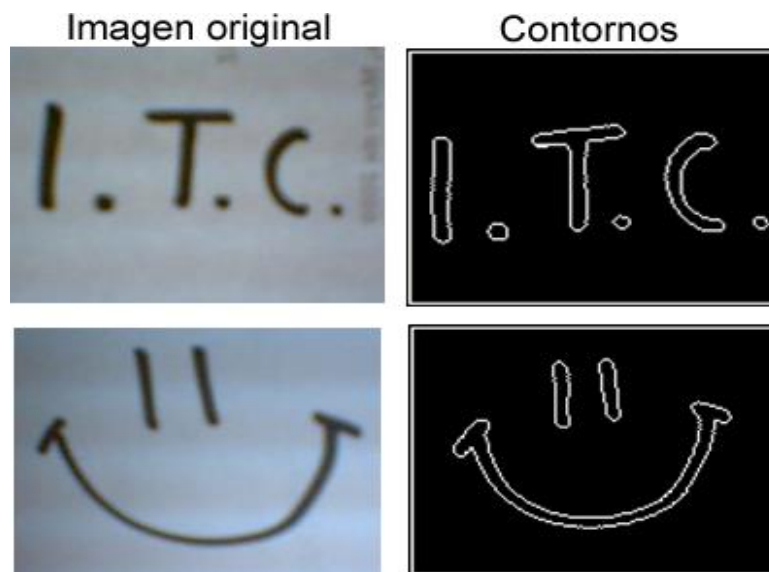


Figura 12. Detección de contornos

Interação com objetos virtuais

Usando este protótipo, você pode capturar objetos usando webcam e incorporá-los em um aplicativo 3D, que tem o potencial de interagir com objetos virtuais que compõem a cena 3D. Numa aplicação deste tipo foram identificados três elementos mostrados na Figura 7.

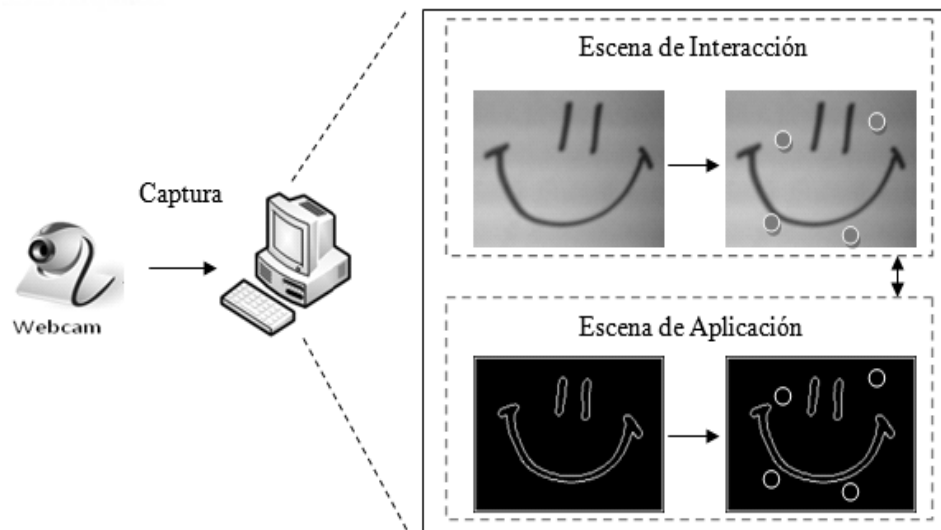


Figura 7. Arquitectura de las aplicaciones desarrolladas

- A. A cena de interação é representada por um conjunto de imagens capturadas pela câmera.
- B. A cena de aplicação é representado pelo conjunto de imagens que são algumas características do cenário de interação. Por exemplo: esboço, cantos e cores.
- C. Finalmente, os objetos virtuais: elas são incorporadas nas cenas de interação e aplicação programaticamente objetos que reagem a eventos no cenário de aplicação e são representados através de imagens ou texturas no cenário de interação.

Para verificar a interação com objetos virtuais, ele desenvolveu um par de exemplos destinados a demonstrar a maneira pela qual o usuário pode interagir com eles. A primeira aplicação mostra um conjunto de bolhas que o usuário pode explorar para fazer movimentos na cena de interação.

A cena da aplicação é a diferença entre uma imagem e seu antecessor. Assim, uma sombra branca é formada.

O segundo exemplo de aplicativo mostra um conjunto de bolas que saltam no cenário de interação, quando o usuário toca qualquer ou quando colidem, mudando sua direção.

O aplicativo consiste contornos cena das figuras que aparecem no cenário de interação, que, em contacto com objectos virtuais, eles executam uma rotina para mudar sua direção de movimento.

Um exemplo destas duas aplicações pode ser observado na Figura 8.

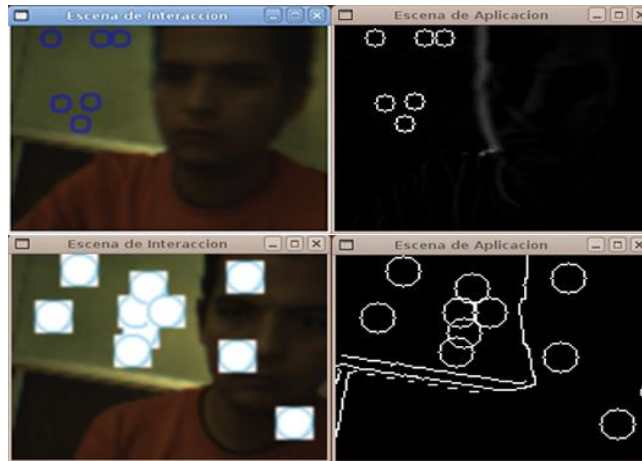


Figura 8. Aplicación que combina imágenes adquiridas y escenas virtuales

III.- Prototype 3: Implementando um CAVE barato

Nesta seção, vamos descrever o desenvolvimento de um protótipo de baixo custo CAVE. Para atingir o objetivo de implementar uma CAVE, dividimos o trabalho em quatro subsistemas são descritos em detalhes nas seções a seguir.

Área de visualização A.

A área de exibição é composto por um cubo de cinco lados. Nós construímos uma estrutura em forma de cubo de metal de 1,80 m de cada lado, então nós cobrir o balde com um tecido especial para permitir a projecção de imagens a partir do

exterior. Para a entrada de usuários na caverna, um acesso fornecido em uma das faces suficientes para permitir a passagem de um usuário tanto espaço implementado. Nesta área de exibição seis usuários podem ficar com algum conforto, detalhes são apresentados na Figura 9.

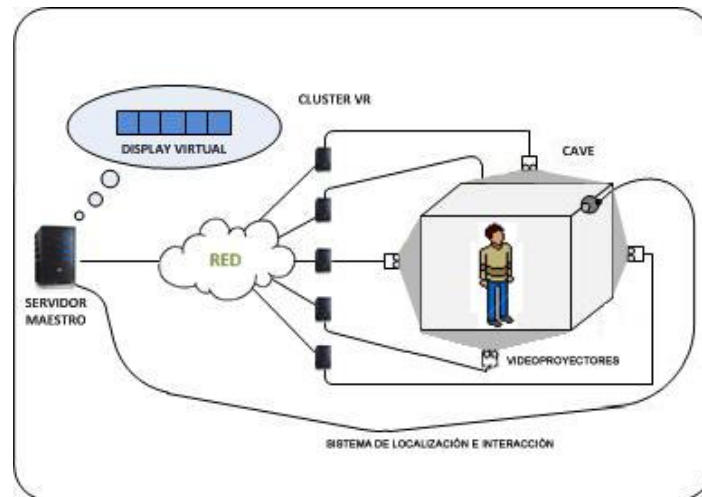


Figura 9: Subsistemas de CAVE de baixo custo

A. Clúster VR

Os aplicativos em execução para chave é suportado em um cluster que compreende seis VR PC equipado com 2 GB de RAM e 1.8 GHz. Os cartões PC gráficos são genéricos e não fornecem suporte para visualização estéreo. PCs são interligados por uma rede Fast Ethernet a 100 Mbps. A fim de reduzir os atrasos associados custo da comunicação no cluster VR, optou-se pelo modelo de execução distribuídas cópias sincronizadas. Neste modelo, cada PC Cluster VR mantém uma cópia completa do aplicativo em execução; embora tenha a desvantagem do consumo excessivo de

memória em cada nó, você ganha em termos de redução dos custos da comunicação. Você só precisa trocar mensagens de sincronização nas cópias hospedados no cluster nós de aplicação VR, estas mensagens geralmente são de tamanho pequeno, o que faz com que pequenas latências. Para suportar aplicações no CAVE, usamos o software sizígia (Schaeffer, Goudeseune, 2003) que permite a paralelização distribuir aplicativos de renderização de gráficos 3D cenas.

Sistema de Projeção C.

Ele consiste de um conjunto de cinco projetores de vídeo convenientemente localizados fora da superfície da tela, a uma distância e distribuídos em uma matriz de tal modo que iluminar completamente cada um dos lados da caverna. Cada projetor é ligado às saídas de cada VR nó do cluster. Nós usamos projetores de vídeo normalmente usados em salas de aula para apoiar a prestação de cursos. A resolução é de 1.024x768 pixels manipulados. É importante ressaltar que esses projetores de vídeo não tem a capacidade de estéreo.

D.- Sistema de acompanhamento e interação

Para ativar a interação do usuário com os aplicativos em execução no CAVE, foi necessária a implementação de uma alternativa para sistemas de rastreamento tradicionais. Nossa solução determina a posição de um usuário utilizando as imagens adquiridas por uma webcam interligado ao cluster de servidor mestre VR. Um utilizador equipado com um capacete em que um LED de alta intensidade é pilotos sujeitos que interagem com o aplicativo em execução no momento. A área interna do CAVE adquirida pela webcam representa a área total em que os usuários se movem. Qualquer

mudança na posição do usuário é detectada pela webcam e isso significa regeneração cena 3D e visualização posterior.

Para detectar alterações nas imagens obtidas pela câmera, vamos implementar um pequeno programa com a ajuda da biblioteca OpenCV. Neste maner, o movimento para os usuários CAVE pode navegar aplicativos 3D. Neste primeiro protótipo suportam apenas para foi implementada uma interação do usuário individual.

Os testes de função Prototype

Para medir a funcionalidade de nosso CAVE, usar alguns aplicativos OpenGL (Mark Segal & Kurt Akeley, 1994) biblioteca. Posteriormente, lançamos o cluster em execução nos projetores de vídeo e VR acesas para iluminar a caverna. Os resultados foram surpreendentes. Apesar de não usar dispositivos com suporte para estéreo, o realismo das imagens dentro da caverna e mergulho foram particularmente encorajadores. Os aplicativos em execução no cluster VR teve um desempenho bastante aceitável. Em relação à gestão de localização do usuário e interação com os aplicativos, a solução implementada adequadamente servido. Imagens de aplicativos em execução CAVE mostrados nas Figuras 14, 15 e 16.





Figura 11. Usuários dentro do CAVE

Figura 10: Exibição de moléculas no CAVE

Figura 15. O sistema de localização e interação CAVE

Discussão dos Resultados

Esta seção descreve as experiências adquiridas através da aplicação de uma caverna com itens de baixo custo, tais como PC, Internet, projetores de vídeo, o que resulta em uma redução significativa nos custos. Com esta infra-estrutura, o Instituto Tecnológico

de Colima tem um sistema de exibição para o desenvolvimento de alta performance e implementação de aplicativos orientados a educação.

CONCLUSÕES

Este artigo apresentou o desenvolvimento de três sistemas para a visualização gráfica de grandes áreas acompanhadas de alta resolução. A coisa notável é que estes desenvolvimentos têm sido usados itens de baixo custo na sua implementação. Os resultados mostram que eles são uma alternativa real para sistemas de visualização caras que usam tecnologias proprietárias e são inacessíveis para as organizações em um orçamento.

As soluções aqui apresentadas podem ser utilizadas por instituições de ensino em aplicações para a educação, com base em seu baixo custo eo fato de que muitos dos seus elementos podem ser reutilizados nos laboratórios dessas instituições.

Bibliografia

Chen, Y.; Chen, H.; Clark, D.W.; Liu, Z.; Wallace, G. & Li, K (2001). "Software environments for cluster-based display systems," Cluster Computing and the Grid, 2001. Proceedings. First IEEE/ACM International Symposium on , vol., no., pp. 202,210.

Cruz, Neira C; Sandin Daniel, J. & Defanti Thomas, A. (1993). "Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE". Proceedings SIGGRAPH , 135-142. Intel Corporetion (s.f.) "OpenCV". Recuperado el 2 de Febrero de 2009, de <http://opencv.org/>

Jérémie, Allard; Valérie, Gouranton; Loick, Lecointre; Sébastien, Liimet; Emmanuel, Melin; Bruno, Raffin & Sophie, Robert (2004). "FlowVR: A Middleware for Large Scale Virtual

Reality Applications". Euro-Par 2004 Parallel Processing, Lecture Notes in Computer Science, Volume 3149, 2004, pp 497-505.

Jérémie, Allard; Clément, Menier; Bruno, Raffin; Edmond, Boyer and François, Faure. (2007). "Grimage: markerless 3D interactions". In *ACM SIGGRAPH 2007 emerging technologies* (SIGGRAPH '07).

Li K.; Chen H.; Chen Y.; Clark D.W.; Cook P.; Damianakis S.; Essl G.; Finkelstein A.; Funkhouser T.; Housel T.; Klein A.; Liu Z.; Praun E.; Singh J.P.; Shedd B.; Pal J.; Tzanetakis G. & Zheng J. (2000). "Building and using a scalable display wall system," *Computer Graphics and Applications, IEEE* , vol.20, no.4, pp.29,37, Jul/Aug 2000.

Mark Segal and Kurt Akeley (1994). "The Design of the OpenGL graphics interface", *Silicon Graphics Computer System*.

Robert, W. Scheifler and Jim Gettys (1986). "The X window system". *ACM Trans. Graph.* 5, 2 (April 1986), 79-109.

Schaeffer, B.; Goudeseune C. (2003), "Syzygy: native PC cluster VR", *Virtual Reality, 2003. Proceedings. IEEE* , vol., no., pp.15,22, 22-26 March 2003.

Wolfgang, Krüger; Christian-A, Bohn; Bernd, Fröhlich; Heinrich, Schüth; Wolfgang, Strauss, and Gerold, Wesche (1995). "The Responsive Workbench: A Virtual Work Environment". *Computer* 28, 7 (July 1995), pp. 42-48.