E-lousa: uma lousa eletrônica baseada em análise de imagens

Moacir Francisco da Silva Junior[1], Regis da Silva Avansini[1], Almir Olivette Artero[1]

[1]Faculdade de Informática de Presidente Prudente - FIPP, UNOESTE. mjunior@unoeste.edu.br, avansini@unoeste.edu.br, almir@unoeste.br

Resumo

Este trabalho apresenta um sistema de lousa eletrônica (e-lousa) de baixo custo, que usa um hardware bastante simples e comum nas salas de aulas atuais. Composto por uma webcam e um projetor de vídeo, conectados a um computador, o sistema apresenta as imagens (Slides, Textos, etc) em um quadro ou mesmo em uma parede, e também realiza a captura da movimentação de um ponteiro laser sobre a imagem projetada, de modo que anotações feitas pelo professor possam ser imediatamente incorporadas à apresentação. Os experimentos realizados têm mostrado que o sistema proposto torna as aulas mais dinâmicas, pois as anotações feitas durante as explicações se tornam disponíveis para o professor e também para seus alunos. Embora o sistema tenha revelado um grande diferencial em aulas presenciais, é no sistema de ensino à distância que o mesmo se torna de fundamental importância. Diferente de outros sistemas de lousa eletrônica, onde o professor deve permanecer próximo do quadro durante a interação com ele, no sistema proposto é possível fazer anotações e também acessar o seu menu a uma boa distância, o que garante também uma maior mobilidade ao professor dentro da sala de aula.
Palavras-Chave: lousa eletrônica, ensino à distância, processamento de vídeo.

E-panel board: an electronic panel board system based on pictures analysis

Abstract

This work presents a low cost electronic panel board system, which uses a simple and common hardware in the current classes. Composed by a Webcam and a video projector, connected to a computer, the system presents the images (slides, texts etc) in a white panel or even in a wall, and it also accomplishes the capture of the movement of a pointer laser on the projected image, so that annotations done by the teacher can be immediately incorporated to the presentation. The accomplished experiments have been showing that the proposed system possibilities most dynamic classes, because the annotations done during the explanations become available for the teacher and also for their students. Although the system has revealed a big one differential in classes, it is in the distance learning that it turns of fundamental importance. Different from other systems of electronic panel board, where the teacher should stay close of the panel during the interaction with it, in the proposed system it is possible to do annotations and also access its menu at a good distance, what guarantees a larger mobility to the teacher in the classroom.
Keywords: eletronic board, e-learning, video processing.

1. Introdução

A utilização da informática nas escolas tem se tornado cada vez mais comum, pois contribui bastante para aumentar a interatividade do aluno com a aula, porém, no princípio a informática era apenas uma ferramenta de apoio ao ensino, visando principalmente prender a atenção dos alunos à aula. Com a popularização dos computadores, novos recursos de hardware foram sendo desenvolvidos, e o projetor de vídeo (ALECRIM, 2007) se tornou um hardware indispensável nas aulas. Porém, apesar de todo o aparato tecnológico, o conforto proporcionado pela escrita na lousa, ainda é maior do que o obtido ao se desenhar usando um mouse. Deste modo, as últimas novidades tecnológicas de hardware incluem um quadro eletrônico, que tenta disponibilizar ao professor a mesma conveniência de um quadro negro, porém com a vantagem de que tudo que é escrito nele é enviado para um computador, e poderá ser distribuído posteriormente aos alunos, por meio de arquivos digitais ou impressos. Os quadros eletrônicos existentes no mercado atual, chamados Panel Board (WEDGWOOD AV LTD, 2009), como por exemplo, Interwhite Board (modelo 1077 – custo R$ 6.342,00), possibilitam o armazenamento das imagens desenhadas sobre eles, funcionando através de sensores baseados em ultra-sons que trabalham com uma caneta eletrônica bem específica e, deste modo, não reconhecem anotações feitas com caneta laser, mais comum nas salas de aula atuais.

Com o crescimento do número de cursos à distância, geralmente, com aulas via Internet, o uso deste tipo de dispositivo tem se tornado um grande diferencial, pois oferece uma maior interatividade entre o professor e os alunos e, pode-se afirmar que, no caso do aprendizado à distância (*E-Learning*), o quadro eletrônico é a principal ferramenta do professor. Contrapondo-se às vantagens do quadro eletrônico está o seu ainda elevado custo, que o tem mantido afastado da maioria das escolas e, o que se observa na prática é que as escolas, quando conseguem adquirir um destes dispositivos, os empregam em poucas de suas aulas.

Diante deste cenário, este trabalho propõe a implementação de um quadro eletrônico de baixo custo (E-Lousa), que funciona a partir do processamento de imagens obtidas com uma câmera digital (WebCam) e um projetor de vídeo, atualmente, muito comum nas salas de aulas atuais. A Figura 1 ilustra o esquema usado nesta implementação, que mostra que a câmera de vídeo captura a imagem gerada pelo próprio computador em que ela está conectada. Em seguida, o computador processa a imagem e gera uma nova imagem, para ser apresentada pelo projetor. Nesta figura também se nota que o usuário usa uma caneta laser comum para desenhar sobre a imagem gerada e, tudo o que ele desenha sobre a imagem é capturado pela câmera e introduzido na própria imagem. Assim, o problema todo a ser resolvido neste sistema consiste em identificar a posição do ponteiro laser na imagem e realizar as ações desejadas sobre ela, o que inclui também acessar menus do sistema, selecionar ações como apagar o quadro, trocar as cores e espessuras de desenho e também selecionar o desenho de objetos gráficos primitivos, como retas, círculos, retângulos e outras formas.

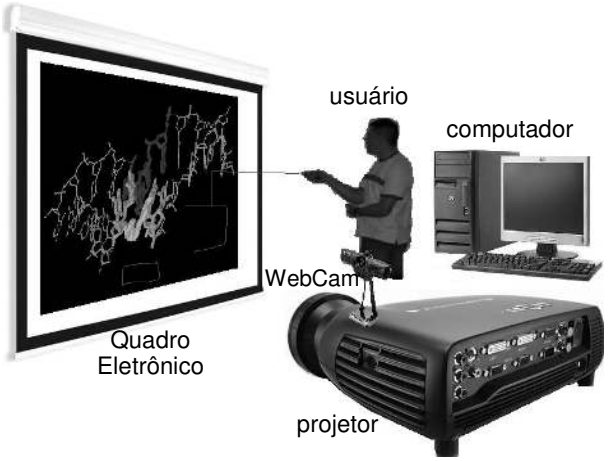


Figura 1. Esquema do Quadro Eletrônico.

As demais seções deste trabalho estão organizadas da seguinte maneira: na Seção 2 é apresentada uma introdução ao processamento digital de imagens, com ênfase nas técnicas usadas neste projeto; na Seção 3 são apresentadas as tecnologias, tanto de hardware, quanto de software, empregadas no desenvolvimento do quadro eletrônico; A Seção 4 apresenta o programa de controle e descreve processamento realizado pelo sistema; Por fim, a Seção 5 apresenta as conclusões e propostas para trabalhos futuros.

2. Processamento de imagens digitais

O Processamento Digital de Imagens (PDI) envolve um conjunto de tarefas interconectadas que começa com a captura de uma imagem. Em seguida, é realizado o pré-processamento, que envolve passos como a filtragem de ruídos introduzidos pelo próprio dispositivo de captura, pela interferência de aparelhos externos ou mesmo por desordens do cenário capturado (GONZALEZ; WOODS, 1992).

No passo seguinte podem ser feitos processamentos objetivando separar os objetos do plano de fundo (background), o que significa que é necessário identificar, através de um processo de segmentação, características constantes de uma parte da imagem e descontinuidades dentro da imagem. Nesta etapa também podem ser extraídas algumas características ou atributos das imagens, tais como as bordas e texturas, capazes de caracterizar os objetos presentes nas imagens. Outra característica importante é o movimento que pode ser medido a partir de várias imagens em uma sequência de vídeo.

2.1. Imagens digitais

Uma imagem digital pode ser tratada como uma matriz I, cujos índices de linhas e de colunas identificam um ponto na imagem, e o correspondente valor do elemento da matriz identifica o nível de cinza naquele ponto (CASTLEMAN,1996). Os elementos dessa matriz digital são chamados de elementos da imagem ou elementos da figura, ou ainda "pixels".

2.2. Segmentação de imagens

A segmentação tem a função de dividir a imagem nas partes que a compõem (GONZALEZ; WOODS, 1992), sendo que o nível da subdivisão depende do problema a ser resolvido. A segmentação deve ser finalizada quando o objeto de interesse for separado da imagem. Também se pode dizer que cada objeto que foi segmentado, em uma imagem, está representado pelo conjunto de pixels que foram agrupados segundo um critério de homogeneidade. Os algoritmos de segmentação para imagens monocromáticas, geralmente, levam em conta duas propriedades das imagens, a similaridade e a descontinuidade. A descontinuidade é responsável por separar uma imagem levando em conta a mudança abrupta dos valores de nível de cinza sendo, normalmente utilizada na detecção de pontos isolados, linhas e bordas. Por outro lado, os algoritmos baseados em similaridade tentam agrupar os pixels de acordo com alguma medida de similaridade.

Embora as imagens coloridas carreguem mais informações do que as imagens monocromáticas e existam vários estudos sobre a segmentação de imagens coloridas, as técnicas mais utilizadas são baseadas nas técnicas de segmentação de imagens monocromáticas, que levam em conta apenas as diferenças de brilho dos componentes da imagem (GONZALEZ; WOODS, 1992). Neste projeto, por causa da necessidade de detectar a posição da imagem formada pelo apontador laser sobre a imagem, utiliza-se uma segmentação baseada na informação de cor. Assim, a localização do ponteiro laser é identificada a partir de sua cor.

3. Captura de imagens usando as bibliotecas do windows

A API VFW foi desenvolvida especificamente para o Windows16 bits e provê funções que permitem uma aplicação processar dados de vídeo. Embora muitas de suas funcionalidades tenham sido substituídas no Directx (JONES, 2004; LUNAS, 2008), seu desempenho ainda permite a sua aplicação neste trabalho, o que torna este sistema mais simples de se portado para outros computadores. Como esta API opera em 16 bits e o Windows, desde a versão 95 trabalha com 32 bits, foi criado, pela Microsoft, um conversor entre as operações desta API e o Windows e, este conversor foi denominado MSVFW32. Essa API possibilita incorporar a captura de vídeo, em uma aplicação utilizando a classe AVICAP, que proporciona aplicações com uma interface simples.

Segundo a *Microsoft*, existem dois passos necessários para chamar uma função *API* do Windows: 1 - Declarar a função em um módulo; 2 - Chamar a função no seu programa como se fosse uma função da linguagem de programação. As funções da *API* ficam disponíveis através de uma biblioteca *DLL*.

3.1. Biblioteca *VFW.DCU*

As funções das API's VFW e AVICAP estão implementados na biblioteca VFW.DCU (MICROSOFT, 2009; MICROSOFT, 2009a). Dentre estas funções estão:

CapCreateCaptureWindow – cria uma janela onde será capturado o vídeo. Esta janela é acessada por meio de um handle de um componente Panel; CapDriverConnect – função que retorna verdadeiro caso um dispositivo de captura esteja conectado ao computador, retornando falso caso contrário; FHasDlgVideoSource – função que abre a caixa de diálogo para os dispositivos de captura do Windows; FhasDlgVideoFormat – função que abre a caixa de diálogo, do Windows, para configurar o tamanho da imagem a ser capturada e o tipo de imagem; CapPreviewRate – configura a taxa de captura dos frames; CapPreview – habilita a visualização do vídeo; CapSetCallbackOnFrame – causa uma interrupção no computador e chama o procedimento ou a função desejada cada vez que ocorrer uma interrupção; CapCaptureSetSetup – envia parâmetros de configuração do modo de captura: capparam.dwRequestMicroSecPerFrame – envia a taxa de captura desejada; capparam.fYield – seleciona captura em segundo plano; capparam.fMakeUserHitOKToCapture – habilita exibição de caixas de diálogo; capparam.fAbortLeftMouse – aborta a captura caso o mouse seja movido para a esquerda; capparam.fAbortRightMouse – aborta a captura caso o mouse seja movido para a direita; capparam.wPercentDropForError – exibe mensagem de erro caso uma quantidade de frames cheguem antes que o frame atual tenha sido completamente processado. Isto ocorre porque enquanto o frame atual é processado, não há como armazenar outros frames; capparam.fCaptureAudio – habilita captura de áudio; capparam.AVStreamMaster – controla o

tamanho do espaço para armazenamento de imagens. A biblioteca *VFW.DCU* implementa um registro com o nome *TVIDEOHDR*, que recebe os dados capturados a cada frame. Na posição *lpData* desse registro fica armazenada uma matriz com as dimensões [1..*Altura*, 1..*Largura*, 1..3] contendo os dados da imagem capturada (coordenadas de cor R,G,B). A cada frame capturado, esta matriz é preenchida novamente, o que possibilita o processamento das imagens diretamente nesta matriz, sem precisar copiá-la para outra estrutura.

3.1.1. Programação usando threads

Em sistemas operacionais tradicionais, cada processo tem um espaço de endereçamento e um fluxo de controle. Contudo, há situações em que é desejável ter múltiplas threads de controle, no mesmo espaço de endereçamento, executando em paralelo, como se fossem processos distintos (TANENBAUM, 2005). Uma thread pode controlar vários processos, mas também é possível que um processo contenha várias threads. Embora as threads criadas em um mesmo programa compartilhem os mesmos recursos (processador, memória, etc), cada thread é executada individualmente e sequencialmente, possuindo seu próprio contador de instruções e sua própria pilha de execução. Cada thread tem seu ciclo de vida, podendo também criar threads filhas. Neste trabalho, as threads são usadas para permitir o tratamento de eventos através do posicionamento do ponteiro laser.

4. Projeto do Quadro Eletrônico (E-Lousa)

O programa de controle captura as imagens projetadas no quadro, utilizando uma câmera de vídeo. Deste modo, as anotações feitas sobre o quadro com o uso de uma caneta laser são também capturadas e, com o processamento desta imagem, é identificado o local onde o apontador laser está posicionado.

Posteriormente, os desenhos feitos com o apontador são incorporados aos demais desenhos da imagem atual e, o processo de realimentação segue adiante.

Antes de iniciar a operação do quadro, o usuário deve acessar um menu (utilizando o mouse), para selecionar a origem e o formato das imagens. Por este menu, se inicia e se interrompe o sistema. Como existe uma grande variedade de cores para os apontadores laser, o sistema proposto prevê também uma calibragem, que é feita pelo usuário antes de começar a operar com o quadro. Neste processo de calibragem, o usuário deve identificar na imagem capturada, a região que contém o marcador formado pela caneta laser. A cor assim identificada é então armazenada pelo sistema e servirá de referência para identificar as posições seguintes do marcador laser obre o quadro eletrônico, até que o usuário resolva fazer uma nova calibragem. Após a realização da calibragem, o sistema localizará a posição do apontador a partir de uma varredura na imagem, até encontrar a cor da caneta. Ao localizar um pixel com a mesma cor do apontador, sua posição (x,y) é armazenada. Para manter linhas contínuas, a cada ponto identificado é traçada uma linha reta entre o ponto anterior e o ponto atual, sendo que a cor e a espessura desta linha pode ser trocada pelo usuário, usando para isto, o próprio apontador laser. Por meio de funções pré-definidas os usuários podem desenhar retângulos, quadrados e círculos. Também usando o apontador, o usuário pode acessar algumas funções do sistema, como: abrir e salvar imagens; apagar, recortar, copiar e colar a imagem da tela.

4.1. Hardware utilizado no projeto

Os dispositivos usados no projeto do quadro eletrônico, ilustrados na Figura 2 são descritos a seguir.



(a) (b) (c) (d) (e)

Figura 2. Dispositivos usados no projeto do quadro eletrônico.

4.1.1. Apontador laser

O quadro eletrônico necessita de um dispositivo capaz de projetar um ponto de luz a certa distância, e que este ponto tenha brilho, suficiente, para aparecer sobre a imagem projetada no quadro. A melhor alternativa neste caso é usar um apontador laser, que consegue atender esses pré-requisitos e ainda apresenta um baixo custo. O laser é formado por um feixe de luz coerente que se concentra numa área pequena e bem definida. Praticamente não existindo o espalhamento deste feixe luminoso. A Figura 2(a) mostra o apontador laser usado neste projeto.

4.1.2 Câmera de vídeo

O dispositivo escolhido neste projeto (Dr. Hank – Figura 2(b)) consiste em uma WebCam, conectada ao computador através da porta USB. O fator que deve ser considerado na escolha da WebCcam é que todos os dispositivos com conexão USB permitem o uso da biblioteca VFW (Vídeo for Windows), o que simplifica bastante a sua programação. O modelo utilizado neste projeto possui uma resolução de 1.3M pixels (640 X 480) com uma taxa de até 30 fps. Embora este tipo de câmera utilize um sensor CMOS, que tem uma qualidade de imagem inferior a um sensor CCD, ela é usada porque apresenta um baixo custo e ainda consegue resultados satisfatórios. Para facilitar o posicionamento da câmera em relação ao quadro é indicado que esta possua opções de zoom digital, ajuste de foco e de ângulo da imagem.

4.1.3. Projetor de vídeo

O projetor de vídeo usado neste projeto (Figura 2(c)) é da marca Panasonic, modelo PT-LC55U, que projeta imagens com uma resolução máxima de 800x600 pixels. Tais características são importantes para determinar a distância entre o projetor e o local em que a imagem é projetada, formando o quadro eletrônico.

4.1.4. Tela de projeção

O local onde a imagem é projetada pode ser uma parede, de preferência branca, ou uma tela de plástico vinil (Figura 2(d)), construída especificamente para esta finalidade. Observado que a cor da tela ou parede não pode ter a mesma cor do apontador laser, pois deste modo não é possível identificá-lo na imagem capturada pela câmera. Também é recomendável que a tela não seja reflexiva, pois os reflexos causam dificuldades no processamento da imagem.

4.2. Programação e estrutura do Quadro Eletrônico

A implementação deste projeto foi feita usando o ambiente de programação Borland Delphi 7, que dispõe de diversos recursos que facilitam a sua construção, tais como acesso à biblioteca VFW e também o suporte ao uso das Threads necessárias para a execução de rotinas em paralelo. A Figura 3 apresenta o diagrama de funcionamento do Quadro eletrônico, sendo os seus blocos descritos nas seções seguintes.

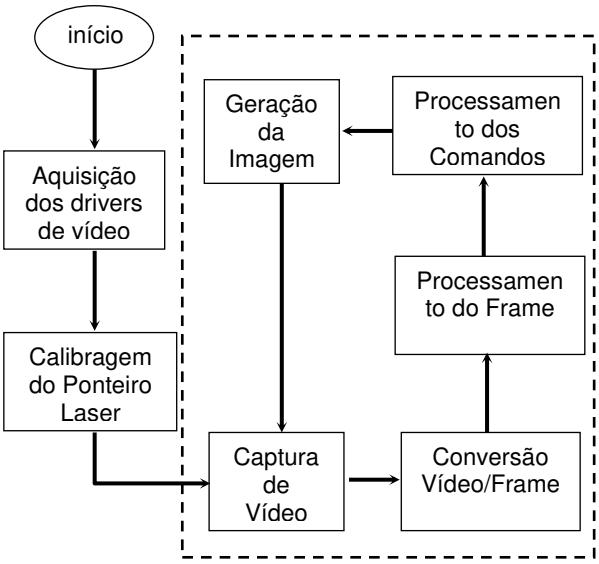


Figura 3. Funcionamento interno do Quadro Eletrônico.

4.3. Calibragem do ponteiro laser

Antes de usar o sistema é necessário fazer a calibração da cor do apontador laser que será usado e, para isto, deve-se clicar no botão *Configura Cor*, que abre uma tela auxiliar, contendo o botão *Captura Imagem*, utilizado para capturar a imagem do quadro. Em seguida, a imagem da caneta laser deve ser projetada no quadro e, com o mouse indica-se a região contendo a imagem do apontador. Em seguida, clica-se no botão *Captura Imagem* para que as informações de cor desta região sejam guardadas pelo computador. Também possível usar mais de um apontador laser, desde que tenham cores diferentes. A Figura 6 mostra a calibração do apontador, onde a região clara, no centro do retângulo é a imagem do apontador.

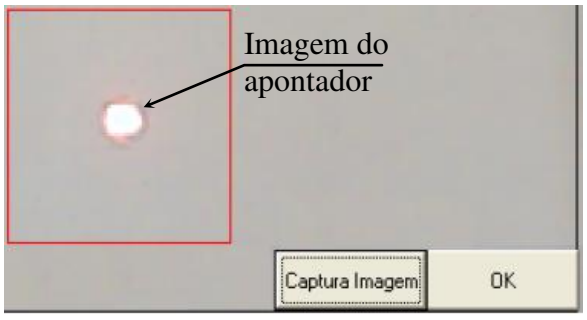


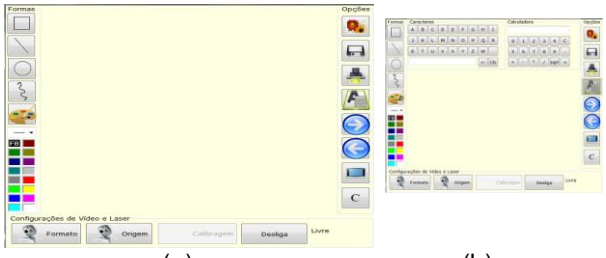
Figura 6. Imagem do apontador projetado sobre o Quadro Eletrônico.

Após o frame ter sido recebido, o comando *CapSetCallbackOnFrame* dispara a função que é executada para tratá-lo, o que é feito colocando a imagem capturada na memória do computador, para que possa ser processada. Neste caso, a imagem é colocada numa matriz tridimensional, de modo que o primeiro índice corresponda à posição *y* do pixel, o segundo índice corresponda à posição *x* do pixel e o terceiro índice corresponda aos três canais de cores (vermelho, verde, azul) da imagem. Duas cores C_1 e C_2, compostas pelas componentes (R_1, G_1, B_1) e (R_2, G_2, B_2) são comparadas medindo a distância Euclidiana entre elas, dada por:

$$Dist = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2} \quad (1)$$

4.4. Captura de vídeo

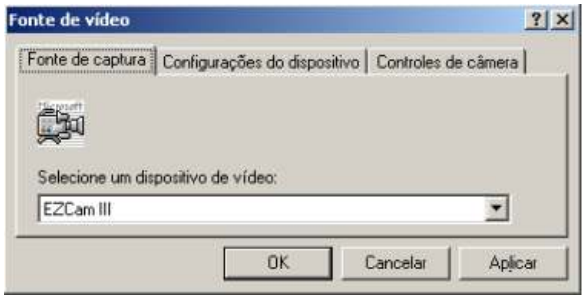
A captura do vídeo é feita usando a biblioteca VFW.DCU, juntamente com diversas outras API's do Windows, o que garante uma ótima portabilidade entre máquinas usando este sistema operacional, pois não é necessário programar cada dispositivo de captura independente, além disso, todos as câmeras de captura compatíveis com VFW podem ser usadas com o sistema. Quando o sistema é acionado, cria-se uma ligação entre o aplicativo e o dispositivo de vídeo do Windows por meio do comando CapCreateCaptureWindow, que envia um parâmetro contendo o handle do local onde será exibido o vídeo. A Figura 4(a) apresenta a tela inicial do sistema E-Lousa, enquanto que em (b) tem-se a tela com o teclado e a calculadora acionados pelo usuário, para fazer algumas anotações e cálculos.



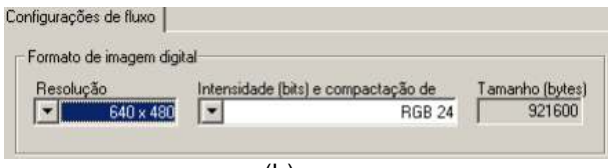
(a) (b)

Figura 4. Tela do Quadro Eletrônico.

Antes de usar o quadro é preciso definir o Formato, usando o botão Formato, que aparece na Figura 5. Quando este botão é pressionado, surge a janela apresentada na Figura 5, que contém todos os dispositivos de captura instalados na máquina. Nesta janela também é possível realizar as configurações do dispositivo selecionado, como a resolução de espaço e cor. Dependendo da qualidade da WebCam, o sistema consegue trabalhar com uma taxa de captura em torno de trinta frames por segundo (incluindo o processamento destes frames). Quando são utilizadas placas de capturas e uma câmera de vídeo convencional, com custos mais elevados, é possível trabalhar com taxas de captura ainda mais altas, porém, com uma taxa superior a dez frames por segundo, o sistema consegue fazer o tratamento das imagens e desenhar as linhas sem deformá-las. Entretanto, quando a taxa de captura cai para menos de dez frames por segundo, o sistema se mostra lento e os resultados deixam a desejar.



(a)



(b)

Figura 5. a) Seleção do dispositivo de captura; b) Seleção do formato e resolução das imagens.

4.5. Conversão vídeo/frame e processamento dos frames

Após os ajustes iniciais, o sistema já pode ser usado, bastando para isto clicar no botão "*Inicia*", que então envia os parâmetros de configuração à biblioteca *VFW.DCU.* A partir deste momento, as imagens são capturadas e, a cada frame recebido pela câmera, é gerada uma matriz contendo os dados. O comando *CapSetCallbackOnFrame* registra uma função que será executada a cada frame recebido do dispositivo de captura e, nessa função devem estar todas as rotinas de processamento das imagens, o que inclui algumas funções do sistema, como por exemplo, a movimentação da caneta, o desenho das linhas e polígonos, a chamada aos métodos de abrir, salvar, apagar, copiar, recortar, colar e ainda ligar ou desligar a exibição do ponteiro do mouse.

A cada imagem inserida na matriz, o sistema do Quadro Eletrônico realiza a sua segmentação através de uma varredura pela imagem, procurando por pixels que tenham as cores do apontador (definidas na Seção 4.4). Quando é encontrado um pixel similar ao do apontador, suas coordenadas, *x* e *y* são armazenadas na memória e as buscas nesta matriz são encerradas. Em seguida, os dados do novo frame são colocados na matriz e, após a localização do apontador neste frame, é traçada uma linha partindo da posição encontrada anteriormente e chegando até a posição atual.

4.6. Processamento dos comandos e geração da imagem

O sistema utiliza variáveis lógicas para controlar as funções que são utilizadas no momento e, por meio dessas variáveis decide se vai ser desenhada uma linha, um círculo, um retângulo, ou se a caneta ganhará a função de mouse, utilizado no momento de abrir ou salvar uma imagem, ou ainda se a função utilizada será a de copiar, recortar ou colar uma imagem.

Para garantir o funcionamento das opções de *Abrir* e *Salvar*, é usado o modelo de processamento paralelo, o que é implementado usando um *thread*, que chama a caixa de diálogo do Windows. A intenção de criar threads é controlar o mouse do computador com a própria caneta, inclusive com a opção de clique. Os *threads* foram criados com a prioridade *tpTimeCritical*, que é a maior prioridade possível a uma *threads* da classe *Tthread*. Com a criação de *thread*, o sistema operacional abre a caixa de diálogo do Windows como um processo distinto do quadro eletrônico, fazendo que o sistema do quadro não perca o foco, e continue a capturar e processar as imagens. Em testes realizados verificou-se que quando o programa que controla o processo de captura perde o foco, ele continua recebendo as imagens, porém não retorna a matriz com os dados para serem processados.

Após a identificação das ações do usuário, através do reconhecimento da imagem do ponteiro laser, o sistema executa as ações solicitadas pelo usuário (desenhos, texto digitado no teclado, etc), gerando a próxima imagem a ser exibida no projetor de vídeo.

4.7. Seleção de linhas e cores de desenho

O Quadro Eletrônico dispõe de um menu interativo que pode ser acessado com a própria caneta a laser, a uma distância considerável, sendo que a distância mais crítica é determinada pela câmera de captura até o quadro. O Quadro Eletrônico pode usar todas as cores disponíveis pelo sistema operacional no traçado das linhas, retângulos e círculos. A escolha da cor não altera anotações e desenhos feitos anteriormente. Também é possível alterar as espessuras das linhas, o que pode ser feito durante o funcionamento do sistema. Estas definições de cor e espessura são realizadas pelo usuário usando as caixas ilustradas na Figura 7.

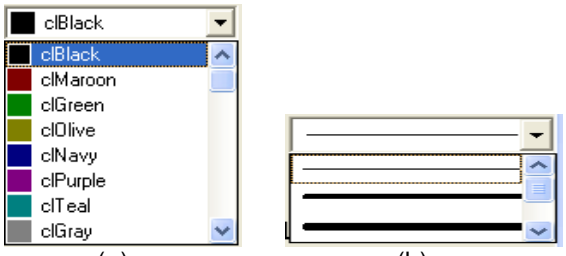


Figura 7. a) Caixa de seleção de cores; b) Caixa com espessuras de linhas.

4.8. Objetos de desenho

Para se desenhar um retângulo ou um quadrado é necessário clicar e arrastar a caneta. O ponto inicial da figura é a posição onde a caneta foi localizada, o usuário arrasta a caneta na diagonal, definindo com isso o tamanho da figura, e a última posição onde a caneta for localizada é o ponto final do desenho. Logo após a caneta ser desligada a imagem é desenhada no quadro. O processo de se desenhar um círculo é praticamente o mesmo, porém o usuário arrasta a caneta definindo o raio desejado para o círculo. A espessura e a cor da linha podem ser aplicadas também ao desenho dos retângulos e círculos. Para facilitar a localização da caneta no quadro, o sistema possibilita a exibição do cursor do mouse na tela, porém há um campo, de fácil acesso, mas não com a própria caneta, que possibilita ocultar o cursor. Se o campo estiver checado o cursor fica visível, caso não esteja checado o cursor fica oculto. Este campo pode ser alterado pelo usuário usando o menu que aparece na Figura 8.

4.9. Menu interativo

O menu interativo, que é apresentado na Figura 8, também é acessado com a própria caneta laser, de uma distância considerável do computador e do quadro. Nesse menu encontram-se as opções de abrir e salvar uma imagem, iniciar uma imagem nova em branco, copiar ou recortar a imagem do quadro para a

memória do computador, colar sobre o quadro a imagem que estiver contida na memória do computador, além de ativar as funções, pré-definidas, para desenhar retângulo ou círculo.

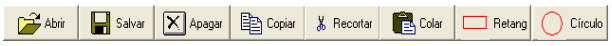


Figura 8. Barra de menu interativo.

Durante o processamento das imagens capturadas são feitas várias verificações sobre o posicionamento do apontador e, caso o apontador esteja sobre a barra de menus, é feita mais uma varredura para se verificar sobre qual botão do menu ele está e, com isso, chamar a função que efetua o processamento desejado. Por exemplo, para acessar a opção "*Abrir*" o apontador deve ser deslocado para cima da barra de menu, sobre esta posição. Com isso, o sistema de controle identifica que a função "*Abrir*" foi chamada. Nesse momento é criado o thread que chama a caixa de diálogo "*Abrir Arquivo*" do *Windows*. Com a utilização de *threads* é possível executar vários processos ao mesmo tempo, assim, neste projeto os processos de captura e processamento de vídeo são realizados pelo sistema de controle, enquanto que o mouse é controlado pela *thread*.

4.10. Impressão

O Quadro Eletrônico também possui funções de impressão e de configuração de impressora, importante para que as anotações feitas no quadro possam ser impressas e distribuídas a todos os participantes de uma atividade realizada com o quadro eletrônico.

5. Otimização

Objetivando amenizar os processos de interpretação e analise das imagens capturadas, foram adotadas duas estratégias importantes para acelerar o processamento. A primeira medida foi a utilização do Bounding Box [FOLEY et al., 1991] para detectar se o ponteiro laser colidiu com algum objeto do Quadro Eletrônico, a definição dos Bounding Box foram demarcadas por "retângulos" onde seu volume representa um Bounding Box. Toda a área de desenho do quadro foi a principio dividido em três regiões, porém quando o teclado virtual é ativado estas regiões passam a ser cinco devido ao aumento de objetos que podem ser manipulados, quatro destes Bounding-Box volumes são disjuntos, ou seja, não existe sobreposição entre eles e apenas um, no caso o que delimita os caracteres não numéricos contém outro Bounding-Box completamente, esta associação de Bounding-Box é mostrada na Figura 9.

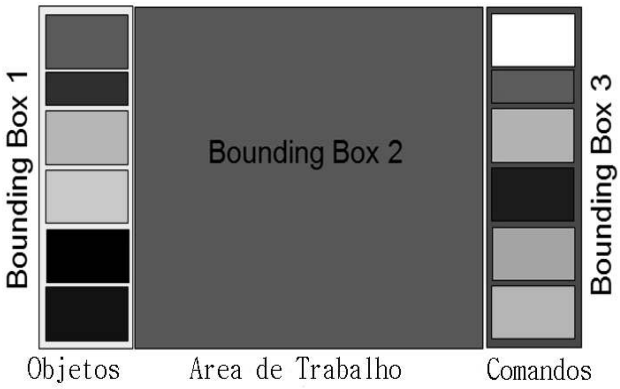


Figura 9. Regiões bounding box do Quadro Eletrônico

A segunda estratégia adotada consiste em realizar uma varredura menos densa da imagem como mostra a Figura 10, para assegurar a consistência das interpretações das ações, ao se encontrar o ponteiro laser na imagem será ativada os elementos adjacentes da imagem no ponto encontrado, apesar de ser uma abordagem simples, garantiu algum ganho com relação ao tempo de processo dos frames.

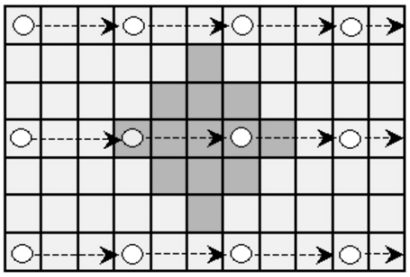


Figura 10. Varredura menos densa dos pixels em uma malha 3 por 3.

6. Resultados

Os experimentos realizados com usuários utilizando o Quadro Eletrônico mostraram uma grande facilidade de uso, se comparado ao tradicional trabalho utilizando o mouse, principalmente para se fazer desenhos a mão livre. A possibilidade de acessar menus do Windows e selecionar os arquivos também usando o apontador também se revelou de grande importância, pois evita que o usuário do sistema tenha que ir até o computador usar o mouse ou o teclado para realizar esta tarefa. O fato da maior parte dos equipamentos necessários já estarem presentes nas salas de aulas atuais, e também a facilidade para se montar o sistema, tem motivado o uso deste recurso em diversas aulas onde existe a necessidade de se fazer desenhos e anotações. Na comparação com o uso de uma quadro disponível comercialmente, foi usado o modelo 1077 da Interwhite Board (custo R$ 6.342,00), os usuários apontaram como principal diferencial do Quandro Eletrônico propostos neste projeto, a possibilidade de se fazer anotações a uma boa distância da tela, o que não é possível no modelo comercial, que obriga o usuário a tocar sobre a tela.

6.1 Interação com o Quadro Eletrônico

A Figura 11 apresenta a tela do Quadro Eletrônico com uma imagem previamente carregada e também algumas anotações realizadas pelo usuário, usando traçados a mão-livre.

Conclusões

Os resultados obtidos com o sistema implementado nesse projeto mostraram que é possível construir um quadro eletrônico de baixo custo, pois utiliza apenas equipamentos já disponíveis em muitas salas de aula. Os experimentos realizados com usuários têm indicado uma grande variedade de aplicações, pois as anotações feitas sobre o quadro eletrônico podem ser facilmente distribuídas aos participantes da aula, de maneira local ou mesmo remota, o que contribui bastante para as aulas realizadas na modalidade de ensino a distância.

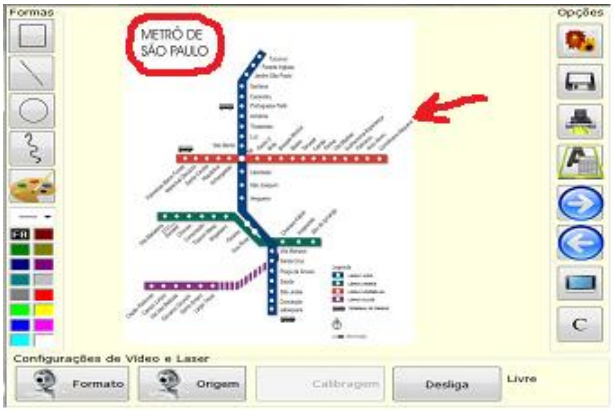


Figura 11. Tela do quadro eletrônico cm anotação sobre uma imagem.

Uma das principais dificuldades observadas com esta versão do sistema é que enquanto o apontador laser está ligado, é realizado o desenho sobre o quadro e, quando o usuário precisa interromper o traçado em um local e continuar um pouco mais adiante, precisa desligar o laser e posicioná-lo na posição desejada para prosseguir, entretanto, com o laser desligado fica difícil posicioná-lo adequadamente. Assim, entre os trabalhos futuros espera-se dotar o sistema de dois apontadores lasers, um verde e um

vermelho, de modo que o verde sempre fique ligado, fornecendo um referencial constante para o usuário, enquanto que o vermelho é acionado apenas quando se deseja realizar alguma ação no quadro. Outra possibilidade é manter apenas um ponteiro laser, porém, adicionando a ele, um sinal de rádio (por exemplo, bluetooth). Uma terceira opção que sendo investigada é controlar a potência do laser, através de um botão. Assim, quando não está pressionado, o laser fica em baixa potência, aparecendo sobre o quadro, mas não sendo reconhecido pelo sistema e, quando o botão é pressionado, é liberada a sua potência normal de trabalho e, então, o sistema reconhece e insere as anotações realizadas. Deste modo o apontador laser sempre ficará acionado, fornecendo o referencial para o usuário e, o traçado de desenhos ou outra ação será realizado apenas se o sinal de rádio for emitido.

Os experimentos com os usuários também têm contribuído para a definição de novos recursos a serem incorporados ao projeto atual, como: o reconhecimento automático de caracteres desenhados manualmente; navegação pela Internet com o próprio quadro eletrônico – a partir do desenvolvimento de um *browser* que permita ao usuário capturar telas e logo em seguida fazer anotações e desenhos sobre elas; possibilidade de abrir mais de um arquivo ao mesmo tempo; suporte para a exibição de vídeos, permitindo a captura de telas e também a realização de anotações e desenhos sobre as mesmas; Incorporação de conceitos de Física para a realização de experimentos – colisões de objetos, alavancas, roldanas, etc. Por fim, poderá ser investigada a possibilidade de gravação de áudio juntamente com as imagens, o que irá complementar as explicações dos conteúdos abordados em uma aula usando o Quadro Eletrônico. A versão atual do sistema desenvolvido neste projeto está disponível para download no endereço *http://fipp.unoeste.br/~almir/researchs/DIP/E-lousa*. Como o hardware necessário para o seu uso é bastante comum nas salas de aulas atuais, espera-se que o seu uso possa atender um grande número de usuários que precisam de um recurso como este para tornar mais fáceis a realização de anotações e a distribuição de suas aulas, principalmente, realizadas na modalidade de ensino a distância.

Referências

ALECRIM, E. **Projetores de vídeo**: principais características, 2007. Disponível em: <http://www.infowester.com/projetores.php>. Acesso em 22 ago 2009.

CASTLEMAN, K. R. **Digital Image Processing**. New York, Prentice Hall, 1996.

FOLEY, J. et al. **Computer Graphics**: Principles and Practice, New York, Addison-Wesley Publishing Company, 1991.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**, London, Prentice-Hall, 1992.

JONES, W. **Beginning Directx 10**, New York, Course Technology PTR, 2004. (Game Development Series).

LUNA, F. **DIntroduction to 3D Game Programming with DirectX 10**, São Paulo, Wordware Publishing, 2008.

MICROSOFT. 2009. Disponível em: <Http://Msdn.Microsoft.Com/En-Us/Library/ Dd757677(Vs.85).Aspx>. Acesso em 22 ago 2009.

MICROSOFT. 2009a. Disponível em: Http://Msdn.Microsoft.Com/En-Us/Library/ Ms645533(Vs.85).Aspx, Acesso em 22 ago 2009.

TANEMBAUM, A. S. **Sistemas Operacionais Modernos**, São Paulo, Prentice-Hall do Brasil, 2006.

WEDGWOOD AV LTD. Interactive whiteboards general information. 2009. Disponível em: <http://www.interactive-whiteboards.co.uk/whiteboard_help.htm>, Acesso em 22 ago 2009.