

## SHIVA: Dispositivo de Entrada de Dados e Interação Virtual por Webcam para Portadores de Necessidades Especiais

Danilo Cunha<sup>1</sup>, Flávio Ramoni<sup>1</sup>, Marcelo Chaar<sup>1</sup>, Ulysses Potiguara<sup>1</sup>, Alessandra Baganha<sup>1</sup>

<sup>1</sup>Curso de Bacharelado em Ciência da Computação - Área de Ciências Exatas e Tecnologia - Centro Universitário do Pará – Av. Gov. José Malcher, 1963 - CEP 66.060-230 – São Brás – Belém – Pará – Brasil.

{danilocunha85, marcelochaar, ulyssespotiguara, souramon}  
@yahoo.com.br, natasha@cesupa.br

**Abstract.** *Based on it offers of devices of data entry for disabled people and the popularity of webcams in the market, with its low cost of acquisition, was developed a solution based on the capture of position supervised for a common camera with the objective to increase the interaction of the users, mainly disabled people.*

**Resumo.** *Fundamentado na oferta de dispositivos de entrada de dados para Portadores de Necessidades Especiais (PNE's) e à popularidade das webcams no mercado, com seu baixo custo de aquisição, foi desenvolvida uma solução baseada na captura de posição supervisionada por uma câmera comum com o objetivo de aumentar a interação dos usuários, principalmente PNE's.*

### 1. Introdução

Desde 1974, [Monteiro, 2007] surgimento do caráter pessoal atribuído aos então chamados microcomputadores ou PC's (*Personal Computer*), a microeletrônica começou a ganhar força e investimento que culminou com os atuais portáteis: *palms*, *laptop*, celulares, até que a portabilidade começou a perder lugar para a mobilidade. Na definição de computação ubíqua, prevista por Mark Weiser já em 1991 [Coulouris, 2001]; a mobilidade, associada aos sistemas distribuídos encontrou apoio na Internet móvel e inauguram a Era da Computação em todo lugar.

Termos da atualidade, e frutos do avanço tecnológico de ponta, estão ainda, neste mesmo sentido, a realidade virtual e a nanotecnologia. Entretanto, face a todos os programas desenvolvidos, a forma de acesso usual aos sistemas computacionais permanece com a mesma classificação apresentada desde 1974, quando os dispositivos para entrada de dados, como o teclado e o *mouse* eram (como ainda são) os meios mais utilizados para estabelecer interface as máquinas.

O objetivo desta pesquisa é apresentar um protótipo de interação virtual por meio de uma câmera de vídeo comum de computador, onde os usuários podem adaptá-lo às suas necessidades. Deste modo, determinados tipos de portadores de necessidades especiais serão beneficiados, visto que um *layout* diferenciado de teclado poderá ser desenvolvido pela própria pessoa.

Ivan Laptev e Tony Lindeberg [Ivan Laptev and Tony Lindeberg, 2000] conseguiram um sistema de reconhecimento de vários estados da mão baseado em uma hierarquia de imagens de multi-escalas. Através de uma câmera o usuário interagia com

um programa que distinguia vários gestos da mão para efetuar diversas funções. Stiefelhagen [Stiefelhagen, 1997], com o auxílio de redes neurais, criou um sistema de rastreamento ocular em tempo real usando uma câmera que ficava localizada na parte inferior do monitor e com isso o usuário controlava um ponteiro na tela.

No presente estudo, será mostrado que, a partir da inspiração obtida no primeiro trabalho e visado estendê-lo a PNE's, bem como, baseado na carência de recursos financeiros para a obtenção de uma câmera de alta resolução - adequada ao segundo trabalho mas, limitando ao público geral devido ao seu alto custo, é pesquisado o uso de contraste com tonalidade ajustável em relação a uma cor base para entrada de dados através de uma *webcam* e um teclado virtual, podendo este ser alterado para atender necessidades específicas. Possibilitando que usuários com limitações, permanentes ou temporárias, nos membros superiores consigam interagir com o computador. Bastando que o usuário seja capaz de produzir contraste na área da tecla desejada.

O sistema de interação foi desenvolvido para a plataforma Windows®<sup>1</sup> e criado a partir do ambiente integrado de desenvolvimento Delphi®<sup>2</sup>.

## 2. Módulos

O sistema foi dividido em cinco módulos, com funções claras e diferenciadas e com finalidades futuras de reaproveitamento dos módulos em eventuais evoluções do sistema.

### 2.1. Módulo de Captura

Responsável pela conexão entre o programa e a câmera. No programa é utilizado um componente [MAsterC, 2006] que se encarrega de detectar a câmera e realizar as devidas conexões e operações. Cada *frame* é convertido em formato *bitmap* que é repassado em seguida para o módulo de processamento.

### 2.2. Módulo de Processamento

É o módulo encarregado de realizar modificações na imagem, como a captura dos contrastes e consequente transformação da imagem em preto e branco baseado na tolerância escolhida pelo usuário; inversão horizontal da imagem; inversão das cores preto e branco para captura em ambientes escuros; captura da cor secundária e o cálculo de sua posição em relação ao teclado virtual.

### 2.3. Módulo de Ação

Aqui o programa recebe a variável que contém o estado das teclas e “chama” o evento correspondente a cada tecla, caso a tecla se encontre “pressionada”. Este módulo também é encarregado de efetuar a repetição do evento em um determinado intervalo de tempo, tal como aconteceria se determinada tecla, do teclado, estivesse pressionada.

### 2.4. Interface Gráfica

A interface do protótipo apresenta, de forma intuitiva, a interação por meio de funções configuráveis do programa e opção de retorno do módulo de processamento, conforme será mostrado na figura 2, item 4 referente a demonstração da solução do teclado virtual

1 Windows® é marca registrada Microsoft Corp.

2 Delphi® é marca registrada Inprise Corp.

a seguir, neste artigo.

## 2.5. Teclados Pré-Definidos

Este módulo contém as classes de teclados criados e compilados junto ao protótipo, personalizáveis via alteração do código-fonte. Este é o módulo que permite futuras personalizações do sistema para adaptação em novos ambientes. Os que foram desenvolvidos para este projeto foram: Teclado Básico, Calculadora e Apresentação.

Esta organização por módulos, além de permitir futuras evoluções do sistema, também facilitou o desenvolvimento e a organização do código-fonte, permitindo que a manutenção seja específica ao módulo com o qual pretende-se alterar ou acrescentar algo, sem necessidade de alterações em todo o código.

Tendo o código-fonte um total de três classes (TPilha, TTeclado e Classe Principal), esta organização modular permitiu um maior compartilhamento do código entre os autores. Sendo assim possível torná-lo mais enxuto e agradável para entender “onde” e “como” cada módulo tem condições de atuar para contribuir com o sistema.

## 3. Algoritmo par captura da imagem e a captura de posição

Como cada câmera pode possuir um tipo diferente de resolução e ainda devido às diferenças de luminescência dos ambientes de uso, foi disponibilizado um meio do usuário adaptar o programa às mais diversas qualidades de *webcams* e claridades. Para tal, o usuário pode ajustar uma percentagem de determinada cor primária, onde qualquer outra cor, fora desta margem, será considerada como cor secundária. Deste modo, duas cores distintas são captadas pelo programa, sendo que a primeira é a cor neutra, que não dispara eventos, e a segunda o programa considera como ponteiro que, se detectada na área de qualquer tecla, o evento da mesma é chamado, como se esta fosse acionada.

Tratada a forma de acionamento da tecla, a partir do *layout* do teclado foi feito então com que o programa diferenciasse contrastes nesta captura com uma certa tolerância definida, sendo que apenas o preto e o branco seriam utilizados. É definida uma cor pela captura para servir de fundo e então o algoritmo, através da tolerância, determina o que é branco ou preto, sendo que cores mais próximas ao fundo são consideradas brancas.

Com o teclado mapeado virtualmente pelo programa e a imagem capturada pela câmera processada em preto e branco, o protótipo realizará a varredura em busca de áreas escuras. Escaneando as coordenadas das teclas e suas respectivas áreas, o programa confronta essas informações com a posição cartesiana dos pontos escuros obtidos anteriormente e incrementa 1 (uma unidade) a cada ponto escuro encontrado numa variável. Se após terminar a varredura, a percentagem de pontos escuros na área da tecla atingir uma determinada percentagem, o protótipo dispara o evento correspondente da tecla.

## 4. O Teclado Virtual

Buscando um meio físico de viabilizar o teclado virtual, foi desenvolvida uma estrutura para servir de base, como num teclado convencional. O dispositivo foi feito com uma borda de compensado fino e papel vegetal, a câmera fica localizada na parte inferior direcionada por trás da folha, enquanto o teclado é mapeado virtualmente pelo

protótipo na superfície da folha, como pode ser observado na Figura 1.

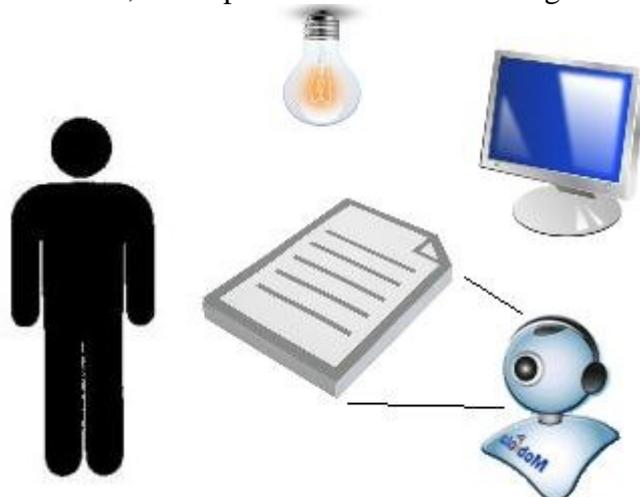


Figura 1. Ambiente Físico

As sombras projetadas no papel são capturadas pela câmera e o algoritmo converte em informações como “pressionado” e “liberado”, para cada tecla mapeada na posição onde a sombra foi projetada. Então, dependendo da ação, um evento pré-configurado é disparado.

Com isso, é possível a criação de novas teclas com funções especiais, definidas pelo usuário, podendo ser reajustado de acordo com suas necessidades. O teclado virtual também permitirá a total modificação de seu *layout*, ajustando-se tanto ao usuário comum quanto a um usuário especial.



Figura 2. Ambiente Gráfico I

Na Figura 2 tem-se um dos teclados pré-configurados e o algoritmo de captura da imagem trabalhando na imagem original. A câmera, localizada sob o protótipo, e o usuário repousando a ponta dos dedos sobre o papel vegetal. A inversão da imagem torna o uso mais intuitivo, já que a câmera localizada na frente do usuário capta uma imagem invertida se comparada com a posição real da pessoa fazendo com que cada movimento se equivalente ao movimento observado do ponto de vista do usuário.

#### 4.1. Considerações Finais

A utilização da programação voltada para a construção de sistemas com enfoque na interatividade humano-computador deve receber cada vez mais a atenção das equipes de desenvolvimento de *software* nas empresas especializadas.

O sistema foi baseado na idéia de que a interatividade é um dos elementos que deve ser levado em consideração na hora de escolher os dispositivos de entrada e saída

de um sistema computacional.

O modelo prevê a utilização de sistemas de entrada de dados baseados na utilização das populares *webcams* de baixo custo e tem um foco na possibilidade de expansão do mesmo para uso de pessoas com necessidades especiais. Propondo uma solução em *software* para um “problema” de *hardware*.

Como proposta de trabalhos futuros ficam, além do já citado modelo voltado para portadores de necessidades especiais, o aperfeiçoamento do protótipo, com o uso de interfaces tangíveis adequadas e especializadas para diversas outras finalidades.

Com baixo custo é possível o aperfeiçoamento dos meios de interação homem-máquina utilizando a realidade virtual, deixando o usuário à vontade para adaptar a própria ferramenta de trabalho. Devido a possibilidade de adaptação do programa, usuários portadores de necessidades especiais poderão usufruir de maneira não apenas mais confortável, mas também mais eficiente dos computadores.

## 5. Referências

- Laptev, Ivan; Lindeberg, Tony. (2000). “Tracking of multistate hand models using particle filtering and a hierarchy of multiscale image features”, Stockholm, Suécia.
- Stiefekhagen, Rainer; Yang, Jie; Waibel, Alex (1997). “Tracking Eyes and Monitoring Eye Gaze”, Universidade de Karlsruhe, Alemanha; Universidade Carnegie Mellon, EUA.
- Coulouris, George; Dollimore, Jean; Kinderg, Tim. Distributed systems: concepts and design. 3. ed. Massachusetts: Addison-Wesley, 2001.
- Monteiro, Mário A. Introdução à organização de computadores. 4. ed. Rio de Janeiro: LTC, 2002.
- MAsterC (2006). <<http://www.delphifr.com/membresite.aspx?ID=4058>>. Capturado em 27/08/2006.
- Planeta Delphi (2006). <<http://www.delphi.eti.br/>>. Capturado em 01/06/2006.