



INSTITUTO FEDERAL
GOIÁS

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
DEPARTAMENTO DE PESQUISA E INOVAÇÃO

Construção de um software para interação com computadores utilizando processamento de imagens

André Ricardo do Carmo Santarém (Bolsista)

Gleidson Moraes Silva (Voluntário)

Heverton Barros de Macedo (Orientador)

Agosto / 2011

Tecnologia em Sistemas de Informação

4º Período

É Bolsista de Renovação: () SIM (X) NÃO

JATAÍ, JULHO DE 2012

Estrutura do relatório final

- 1 – Identificação do Projeto e Componentes;
- 2 – Introdução;
- 3 – Material e Métodos;
- 4 – Resultados;
- 5 – Conclusão;
- 6 – Perspectivas de continuidade ou desdobramento do trabalho;
- 7 – Publicações e participações em eventos técnico-científicos;
- 8 – Apoio e Agradecimentos;
- 9 – Referências Bibliográficas;
- 10 – Bibliografia.

1 – Identificação do Projeto e Componentes

Título do Projeto: Construção de um software para interação com computadores utilizando processamento de imagens

Bolsista: André Ricardo do Carmo Santarém

Voluntário: Gleidson Moraes Silva

Orientador: Heverton Barros de Macedo

Local de execução: Jataí - Goiás

Vigência: Agosto 2011 – Julho 2012.

2 – Introdução

À medida que a tecnologia evolui, novas formas de interação são criadas com o objetivo de melhorar a comunicação entre as máquinas e os homens e vice-versa. Um dos dispositivos que tem ganhado mercado nestes últimos anos consiste na manipulação de informações através de telas sensíveis ao toque, conhecidas também por mesas digitalizadoras, presentes nos computadores denominados tablets PCs.

As empresas desenvolvedoras de vídeo games estão criando formas de interação entre o dispositivo e seus utilizadores de forma surpreendente. Um exemplo revolucionário de interação homem-máquina é o Kinect, lançado pela Microsoft em 2009. O acessório é conectado ao vídeo game Xbox 360 e possui um sistema que reconhece vozes, gestos e movimentos permitindo ao jogador se tornar o próprio controle dos jogos.

Durante uma conferência que antecede a *Electronic Entertainment Expo* 2009, importante feira do mundo dos videogames, popularmente conhecida como E3 2009, a Microsoft revelou um passo gigantesco no avanço tecnológico. Trata-se de algo além dos controles de movimento presentes no Wii, a interação desta vez é para valer, e ela é capaz até de conversar com você! [...] Para navegar entre os menus, nada de setas para a esquerda e para a direita. Pequenos “tapas” com a mão, como se empurrassem os quadros da *Dashboard* substituem quaisquer botões, e um comando de voz como “Play” inicia o conteúdo selecionado. (ABRÃO, 2009).

Nesse cenário de jogos e interatividade pode-se observar a utilização de câmeras auxiliando no processamento de imagens e movimentos. Porém as câmeras não estão sendo utilizadas apenas neste universo do entretenimento, mas em diversas outras áreas como: saúde, pesquisas espaciais, mapeamento e rastreamento geográfico entre outras.

A área de processamento de imagens vem sendo objeto de crescente interesse por permitir viabilizar grande número de aplicações em duas categorias bem distintas: (1) o aprimoramento de informações pictóricas para interpretação humana; e (2) a análise automática por computador de

informações extraídas de uma cena (MARQUES FILHO e VIEIRA NETO, 1999, p.1).

Na área educacional, existem novos dispositivos que proporcionam grande interatividade entre o homem e a máquina, como as lousas interativas, que tem auxiliado professores de diversas áreas do conhecimento a tornar o estudo uma atividade mais agradável, produtiva e dinâmica.

O presente trabalho consiste no desenvolvimento de um sistema de baixo custo e boa usabilidade que possa manipular as imagens de uma webcam para funcionar como uma lousa interativa, porém com recursos simplificados, apenas efetuando o processamento das imagens como em GONZALEZ e WOODS (2000).

De forma geral, o sistema construído detecta movimentos realizados pelo usuário do computador e seus movimentos são transformados em comandos, evitando assim o uso do mouse ou teclado na operação do computador.

No desenvolvimento do software foi utilizada a linguagem de programação JAVA e o ambiente de desenvolvimento Netbeans 6.9.1. Existem várias Interfaces de Programação de Aplicativos - API's (*Application Programming Interface*) que auxiliam no desenvolvimento de aplicações JAVA. Neste projeto foi utilizada a API DSJ (*Direct Show Java*).

O sistema desenvolvido pode ser utilizado para apresentações, seminários, palestras, workshop, além de prover uma nova ferramenta para ministrar aulas.

3 - Material e Métodos

3.1 Materiais:

- Microcomputador, Instituto Federal de Goiás – Campus Jataí;
- Web Cam, Instituto Federal de Goiás – Campus Jataí;
- Retroprojetor, Instituto Federal de Goiás – Campus Jataí;
- Linguagem de Programação Java – Oracle;
- API Direct Show Java (DSJ) – Humatic;
- Ambiente de Desenvolvimento Netbeans6.9.1 – Netbeans.org.

3.2 Métodos:

O projeto teve o objetivo de desenvolver um sistema para ser utilizado em apresentações sem o uso de periféricos (teclado, mouse). Para desenvolver a aplicação mencionada, foi utilizado um microcomputador com sistema operacional Windows 7, com os seguintes programas previamente instalados:

- JDK (Java Development Kit) 6 update 27 ou superior;
- IDE Netbeans 6.9.1 ou superior;
- API DSJ (Direct Show Java);

Utilizando a IDE Netbeans, foi criado um novo projeto utilizando a linguagem Java. Nesse projeto foram criadas as seguintes classes:

InicializaDispositivos.java – Essa classe tem a função de buscar os dispositivos de vídeo conectados ao computador (webcam). Essa classe utiliza algumas classes da API DSJ, para capturar os dispositivos e a partir dessas classes são criados objetos que retornam essas informações;

Capturalmagens.java – Essa classe tem a função de obter imagens através dos objetos de armazenam os dispositivos de vídeo já localizados anteriormente pela classe **IncializaDispositivos.java**;

Manipulalmagens.java – Essa classe tem a função de receber objetos do tipo imagem e realizar operações com essas imagens. Operações como “desenhar quadrados virtuais” na imagem, comparar imagens e verificar se existem diferenças entre as mesmas e outras operações;

InterfaceUsuario.java – Essa classe tem a função de criar um formulário base onde serão criados “botões virtuais” que serão utilizados pelo usuário para manipular uma aplicação, uma

apresentação de slides, por exemplo. Nesse formulário seria apresentada a imagem capturada pela webcam, e através das classes referidas anteriormente, são realizados os processamentos na imagem, “imprimindo” os “botões virtuais” na imagem e fazendo comparações se houve alteração nos pontos delimitados pelos “botões virtuais”.

A comparação é realizada, de forma simplificada, da seguinte maneira: A webcam captura imagens e a sua reprodução assemelha-se a um vídeo, que podemos considerar como um conjunto de várias imagens com pequenas alterações que estão sendo mostradas sequencialmente. Os “Botões Virtuais” determinam qual o local deve ser observado em caso de alterações no quadro atual de imagens. Se houve alteração, em algum dos locais pré-determinados pelos “botões virtuais”, um comando do teclado é executado.

A execução de comandos do teclado são realizados com a implementação de um objeto da classe Robot, que faz parte do pacote do próprio Java e permite executar comandos do teclado ou mouse através da codificação. Dessa forma cada “botão virtual” já possui sua tarefa bem definida, por exemplo, um dos botões “Avança um slide” da apresentação e outro botão “Volta um slide”.

Foram realizados diversos testes até se obterem os resultados esperados, uma vez que as alterações de sombra e luminosidade também provocam alterações nos pixels entre as imagens capturadas. Observando esse fato foi implementada uma “faixa de tolerância” entre as alterações dos pixels, que estariam associadas às oscilações da luminosidade do ambiente, ou seja, somente seria registrado que houve alteração de uma imagem para outra caso o número de pixels alterados seja superior a “faixa de tolerância” utilizada. Esse parâmetro é definido internamente no sistema, ou seja, via código, não sendo possível sua alteração pelo usuário.

A aplicação final possibilita ao usuário informar onde deseja que sejam localizados os “botões virtuais”, pois de acordo com o local onde será realizada a apresentação pode ser necessário um melhor ajuste das posições dos botões.

4 – Resultados

Como resultado, o software proposto permite que seu utilizador crie diversos “botões virtuais” vinculados a comandos também definidos pelo usuário. Por exemplo: um professor, poderia usar o software para passar os slides de sua apresentação como se utilizasse uma lousa interativa.

No caso hipotético do uso em sala de aula, uma possível ação de um “botão virtual” seria o acionamento da tecla *page down*, permitindo que os slides sejam passados ao realizar um movimento, feito com a mão, na região delimitada pelo botão virtual previamente configurado.

Tanto a ação a ser realizada pelos botões quanto o seu tamanho e posicionamento na tela são definidos pelo utilizador do software. Dessa forma sua utilização não está restrita apenas ao uso em sala de aula, mas também em ambientes onde o teclado ou mouse possa ser substituído por ações simples que serão identificadas com movimentos feitos com as mãos.

É possível visualizar na Figura 1 os botões virtuais no canto esquerdo da imagem. Os botões correspondem as regiões delimitadas pelos quadrados pretos presentes na figura. Uma configuração desse tipo, com três botões, poderia ser usada para executar ações como: passar os slides para frente, passar os slides para trás e, por fim, desabilitar temporariamente os botões virtuais. Vale destacar que os botões virtuais estarão presentes apenas no momento em que eles são criados e configurados, desaparecendo assim que a apresentação for iniciada.



Figura 1 – Botões virtuais criados pelo usuário.

Fonte: Autoria própria.

5 - Conclusão

O desenvolvimento de um software capaz de detectar movimentos a partir de uma webcam foi realizado. Sua construção foi projetada com o objetivo de executar interação com um computador sem os dispositivos básicos de entrada, como o teclado e mouse. O software pode ser utilizado como ferramenta de auxílio ao professor em sala de aula funcionando como uma lousa interativa, embora esse não seja o seu único propósito.

No funcionamento do software as detecções de movimentos são realizadas em regiões específicas da imagem, definidas pelo seu utilizador. Tais regiões são consideradas “botões virtuais” que podem executar ações predefinidas como, por exemplo, a transição de slides em uma apresentação.

Comparar o alto grau de interatividade alcançado pelos equipamentos da indústria dos videogames com o presente trabalho, que utiliza apenas uma webcam, é algo muito pretensioso. Porém, o desenvolvimento do projeto permitiu uma nova forma de interação com a máquina, indicando a viabilidade de se construir novos projetos na área de processamento de imagens e visão computacional.

Atualmente o software está em fase de testes nos laboratórios do IFG – campus Jataí e posteriormente será disponibilizado para a comunidade acadêmica.

6 – Perspectivas de continuidade ou desdobramento do trabalho

O projeto tem a possibilidade de ser continuado na forma de um trabalho de conclusão de curso, utilizando os conceitos apresentados e implementando novas funcionalidades à aplicação desenvolvida.

7 – Publicações e participações em eventos técnico-científicos

Participação no Seminário local de Iniciação Científica IFG Campus Jataí, 26/03 de 2012, Jataí – GO.

8 – Apoio e Agradecimentos

O projeto teve financiamento do CNPq/AF.

Colaboraram: Heverton Barros de Macedo (Professor Orientador), André Ricardo do Carmo Santarém (Bolsista) e Gleidson Moraes Silva (Voluntário), Coordenação dos Cursos de Informática, IFG – Campus Jataí.

9 – Referências Bibliográficas

ABRÃO, Gustavo Bonato. **Microsoft revoluciona a forma de se jogar videogame.** Em: <<http://www.tecmundo.com.br/2180-microsoft-revoluciona-a-forma-de-se-jogar-videogame.htm>>, Acesso em 20, mar. de 2011.

GONZALEZ R.C, WOODS R.E. **Processamento de Imagens Digitais.** Edição 1. São Paulo: Edgard Blücher, 2000.

MARQUES FILHO, Ogê; VIEIRA NETO, Hugo. **Processamento Digital de Imagens.** Rio de Janeiro: Brasport, 1999.

10 – Bibliografia

ANDRADE, Maria Margarida. **Introdução à metodologia do trabalho científico**, São Paulo: Atlas, 2007.

MUSEU DO COMPUTADOR. Em: <<http://www.museudocomputador.com.br/enciteclado.php>>, Acesso em 20 de março de 2011.

ORACLE. **Class Robot.** Em: <<http://docs.oracle.com/javase/1.4.2/docs/api/java.awt/Robot.html>>, Acesso em 21 maio de 2012.

PERANI, Letícia; BRESSAN, Renato Teixeira. **Wii Will Rock You: Nintendo Wii e as Relações entre Interatividade e Corpo Nos Videogames.** Em: <http://www.petfacom.ufjf.br/arquivos/artigos/Artigo_3_Wii.pdf>, Acesso em 21 de março de 2011.

PETERS, Nils, **DSJ Javadoc.** Em: <<http://www.humatic.de/htools/dsj/javadoc/index.html>>, Acesso em 21, maio de 2012.

ROCHA, Heloisa Vieira da, BARANAUSKAS, Maria Cecilia Calani. **Design e Avaliação de Interfaces Humano-Computador.** São Paulo – Instituto de Computação: Unicamp, 2003.

TANENBAUM, Andrew S.; WOODHULL, Albert S. **Sistemas Operacionais Projeto e Implantação.** 2. ed. Porto Alegre: Bookman, 2000.

VIDAGEEK. **Como funciona o Wii Remote.** Em: <<http://vidageek.net/2009/07/01/como-funciona-o-wii-remote/>>, Acesso em 28 de março de 2011.

YOUTUBE. **Matéria Jornal da Globo (KINECT XBOX 360)** - 04/11/2010 Em: <<http://www.youtube.com/watch?v=1JdWomEyyr0>>, Acesso em 29 março de 2011.