

UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA

2010.1

INVESTIGAÇÃO DA VIABILIDADE TÉCNICA DA CONSTRUÇÃO
DE UMA VITRINE INTERATIVA

GUILHERME BARRETO DE MENDONÇA

Recife, julho de 2010

UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
CENTRO DE INFORMÁTICA

2010.1

INVESTIGAÇÃO DA VIABILIDADE TÉCNICA DA CONSTRUÇÃO
DE UMA VITRINE INTERATIVA

TRABALHO DE GRADUAÇÃO

Monografia apresentada ao Centro de
Informática da Universidade Federal de
Pernambuco, para obtenção do Grau de
Bacharel em Ciência da Computação.

Aluno: Guilherme Barreto de Mendonça

(gbm@cin.ufpe.br)

Orientador: Edson de Barros Carvalho

(ecdbcf@cin.ufpe.br)

Recife,
2010

INVESTIGAÇÃO DA VIABILIDADE TÉCNICA DA CONSTRUÇÃO
DE UMA VITRINE INTERATIVA

Guilherme Barreto de Mendonça

TRABALHO DE GRADUAÇÃO

*Trabalho encaminhado a Banca
Examinadora do Curso de Ciências da
Computação da Universidade Federal
de Pernambuco – UFPE.*

Judith Kelner
Avaliador (a)

Edson de Barros Carvalho
Professor Orientador

Guilherme Barreto de Mendonça
Universidade Federal de Pernambuco – UFPE
Curso de Ciências da Computação

Recife
Julho/2010

Agradecimentos

Primeiramente, a Deus por ter me iluminado durante toda minha vida e, em especial, durante todo o meu curso de Graduação no CIn. Consegui, com a força Dele, chegar ao final do curso, mesmo com tantas adversidades no meio do caminho.

A toda minha família, em especial aos meus pais, por terem me proporcionado uma educação de primeira qualidade, através da qual conquistei meu objetivo de estudar em um bom curso de uma Universidade Federal.

A minha namorada e aos meus amigos, por terem me dado força para terminar este Trabalho de Graduação e por me proporcionar muita alegria durante todo o curso.

Ao meu orientador, Edson de Barros Carvalho, que me deu a oportunidade de realizar este Trabalho de Graduação que será fundamental na minha conclusão de curso.

Por fim, a todos que torceram pelo meu sucesso, muito obrigado.

Resumo

Alguns conceitos de Realidade Aumentada e Interação Usuário-Máquina permitem uma grande variedade de aplicações interativas em tempo real.

Uma das mais conhecidas é a Vitrine Interativa, que pode ser definida como um produto composto de equipamentos, periféricos e softwares. Tal produto oferece uma interação em tempo real entre um usuário e uma aplicação projetada na vitrine de uma loja.

Para isso ser possível, esses equipamento precisam ter uma série de especificações mínimas e estarem posicionados corretamente. Além disso, esses softwares precisam estar configurados e executando devidamente, para permitir uma interação sem falhas e sem atrasos perceptíveis ao olho humano.

O estudo descrito neste trabalho representa as análises e conclusões de um Estudo de Viabilidade Técnica de dois tipos de Vitrine Interativa a serem desenvolvidas pela empresa Codemedia.

Por fim, serão apresentadas as principais contribuições deste trabalho, além de possíveis trabalhos futuros relacionados ao tema proposto.

Palavras-chave: Produto, Interação, Vitrine Interativa, Viabilidade Técnica, Codemedia.

Abstract

Some concepts of Augmented Reality and Human-Computer Interaction allow a large sort of interactive applications in real time.

One of the most popular is the Interactive Window, which can be defined as a composed product of equipment, peripherals and software. This product offer an interaction in real time between the user and an application projected into a store's window.

In order to make that possible, these equipments need to have several minimal specifications and they must be positioned correctly. Besides that, theses software need to be configured and executing properly to allow an interaction without errors and noticeable delays to the human eye.

The study described in this work represents the analyses and conclusions of a Technical Viability Study of two types of Interactive Window to be developed by Codemedia enterprise.

Lastly, it will be presented the main contributions of this work, and also possible future works related to the theme proposed.

Key-words: Product, Interaction, Interactive Window, Technical Viability, Codemedia.

SUMÁRIO

1. CONSIDERAÇÕES INICIAIS.....	10
1.1. INTRODUÇÃO.....	10
1.2. MOTIVAÇÃO.....	11
1.3. OBJETIVOS.....	12
1.4. METODOLOGIA DE TRABALHO.....	13
1.5. ESTRUTURA DO DOCUMENTO.....	13
2. FUNDAMENTAÇÃO TEÓRICA.....	15
2.1. CONCEITOS DE REALIDADE AUMENTADA E INTERAÇÃO USUÁRIO-MÁQUINA.....	15
2.2. VITRINE INTERATIVA E OUTROS PRODUTOS INTERATIVOS.....	17
2.2.1. Vitrine Interativa.....	18
2.2.1.1. Funcionamento básico.....	20
2.2.1.2. Exemplos reais.....	21
2.2.2. Piso interativo.....	23
2.2.2.1. Funcionamento básico.....	24
2.2.2.2. Exemplos reais.....	25
2.2.3. Outros produtos de interação em tempo real.....	26
3. ESTUDO DE VIABILIDADE TÉCNICA.....	29
3.1. VI e VI+.....	29
3.2. EQUIPAMENTOS E ACESSÓRIOS.....	30
3.2.1. Projetor.....	31
3.2.2. Câmera.....	33
3.2.3. Canhão Infravermelho	37
3.2.4. Computador.....	39
3.2.5. Acessórios.....	40
3.3. SOFTWARES.....	42
3.3.1. Tracker.....	42
3.3.2. Comunicação.....	52
3.3.3. Aplicação Front-End.....	56
3.3.4. Pré-requisitos de software.....	58
4. DESENVOLVIMENTO DA VI.....	60
4.1. EQUIPAMENTOS E ACESSÓRIOS.....	60
4.1.1. Projetor.....	60
4.1.2. Câmera.....	62

4.1.3.Canhão Infravermelho.....	63
4.1.4.Computador.....	63
4.1.5.Acessórios.....	65
4.2.SOFTWARES.....	65
4.2.1.Tracker.....	65
4.2.2.Comunicação.....	70
4.2.4.Pré-requisitos de software.....	72
5.CONSIDERAÇÕES FINAIS.....	74
5.1.CONCLUSÃO.....	74
5.2.PRINCIPAIS CONTRIBUIÇÕES.....	74
5.3.TRABALHOS FUTUROS.....	75
REFERÊNCIAS.....	77
ANEXOS.....	78

LISTA DE FIGURAS

Figura 1 - Exemplo de ARToolKit.....	17
Figura 2 – Exemplo de Vitrine Interativa.....	18
Figura 3 – Exemplo de Piso Interativo.....	23
Figura 4 - Equipamentos de um Piso Interativo.....	24
Figura 5 - Exemplo de Mesa Interativa Multitoque.....	26
Figura 6 - Tela principal do CCV 1.3.....	45
Figura 7 - Tela de calibração do CCV 1.3.....	48
Figura 8 - projeto de opencvExample.....	50
Figura 9 - opencvExample em execução.....	51
Figura 10 - UDP-TCP Bridge em execução.....	55
Figura 11 - Projetor.....	61
Figura 12 - Câmera.....	62
Figura 13 - Computador.....	64
Figura 14 - Modo Calibração.....	68
Figura 15 - Modo Configuração.....	69
Figura 16 - Modo Envio.....	70

1. CONSIDERAÇÕES INICIAIS

1.1. INTRODUÇÃO

O uso de produtos interativos está cada vez mais comum na Mídia Indoor. A variedade de abordagens e aplicações permite o desenvolvimento de vários tipos desses produtos e em vários tipos de ambiente.

Museus, aeroportos, exposições, shoppings e vários tipos de lojas passaram a investir pesado na Mídia Indoor e deixaram a Mídia Outdoor um pouco de lado. Propagandas em outdoors, comerciais de TV e anúncios de revista, entre outros, estão cedendo espaço para aplicações interativas projetadas no piso ou na vitrine.

Uma matéria do [Jornal Hoje, 2009] mostra que consumidores são “fisgados” pela vitrine, já que uma pesquisa apontou que mais de 80% das vendas de roupas acontecem por causa da vitrine. Esse valor comprova que a vitrine é uma grande aliada da Propaganda.

Impulsionado por esses dados e pela crescente da Mídia Indoor, um produto interativo conhecidos como Vitrine Interativa se torna cada dia mais utilizado por lojas que desejam aumentar as suas vendas chamando a atenção dos consumidores que passam próximo ao local.

O principal objetivo deste Trabalho de Graduação é a investigação da viabilidade técnica da construção de duas versões de Vitrines Interativas. Uma delas será mais simples, utilizando equipamentos de preços acessíveis e softwares com menos funcionalidades, enquanto que a outra fará uso de equipamentos mais completos e softwares mais robustos.

Esses, por sua vez, utilizam vários conceitos de áreas que trabalham com interação entre usuário e máquina em tempo real, como a Realidade Aumentada e a Interação Usuário-Máquina. Alguns dos principais conceitos serão abordados ao longo do capítulo.

Além desses conceitos e da apresentação e análise detalhada dos equipamentos e softwares utilizados em Vitrines Interativas, serão mostrados outros tipos de produtos interativos que estão cada dia mais comuns na Mídia Indoor, como o Piso Interativo e as Mesas Interativas Multitoques.

Nos anexos presente no final deste trabalho, pode-se encontrar links que mostram exemplos reais desses produtos em todo o mundo e em vários tipos de ambientes diferentes.

1.2. MOTIVAÇÃO

A motivação para a escolha desse tema se deu a partir do meu interesse nas aplicações de interação em tempo real. Já havia um interesse em estudar conceitos de Realidade Aumentada e Interface Usuário-Máquina que abordassem esse tipo de aplicação e, a partir daí, desenvolver algum produto nesta área

Esse interesse foi intensificado quando, em uma viagem no início de 2009, me deparei com um tipo de produto interativo, o Piso Interativo, e, com isso, foi possível observar, na prática, a capacidade que esse tipo de produto tem de envolver o usuário.

A oportunidade de desenvolver um produto como esse surgiu através de minha empresa, a Codemedia. É uma empresa formada por quatro sócios, sendo três alunos e um professor do CIn. O início da trajetória se deu na cadeira de “Projeto de Desenvolvimento”, mais conhecida como Projetão. Na época, alguns alunos se juntaram para fazer um projeto sobre compra e venda de ingressos pelo celular, mas não foi levado adiante, na época.

Porém, em 2009, alguns antigos membros da equipe decidiram se juntar novamente e tentar uma subvenção econômica. A partir daí, surgiu a Codemedia. O projeto, apelidado de SeuTicket, foi totalmente replanejado e demos início ao seu desenvolvimento.

Durante a fase inicial de implementação, no início de 2010, surgiu a oportunidade, através de um empresário da área de Publicidade, de desenvolvermos outro projeto. Esse, por sua vez, seria uma Vitrine Interativa.

Apesar de tal projeto não ter nenhuma ligação aparente com o SeuTicket, ele interessou bastante a empresa, especialmente a mim, e, por isso, decidimos aceitar a proposta de desenvolvê-lo.

Com esse novo projeto, decidimos conversar com um amigo e também aluno do CIn, Jerônimo Barbosa, sobre a possibilidade dele participar do

desenvolvimento, já que essa área era de seu interesse. Ele aceitou a parceria com a Codemedia e, a partir daí, iniciamos o planejamento.

Chegamos, então, a duas alternativas: a construção de uma versão simples, com poucos requisitos, e a de uma versão mais completa e robusta, com uma lista de requisitos mais extensa, ou seja, mais custosa de ser desenvolvida. Para uma melhor identificação, apelidamos a primeira versão de *VI* (sigla para *Vitrine Interativa*, nome pelo qual esse tipo de produto é, geralmente, conhecido no mercado) e a segunda de *VI+*.

A partir desse planejamento, o próximo passo seria o desenvolvimento de um Estudo de Viabilidade Técnica que abordasse ambas as versões. Esse estudo englobaria conceitos de Realidade Aumentada e Interação Usuário-Máquina, pesquisa de produtos similares já existentes e, principalmente, estudos sobre cada equipamento e software necessário para o desenvolvimento das versões *VI* e *VI+*.

Coincidentemente, na mesma época, me matriculei na cadeira de Trabalho de Graduação, do Centro de Informática, e conversei com Edson sobre a possibilidade dele ser o meu orientador. Ele aceitou e comentou que uma boa idéia para o trabalho seria esse Estudo de Viabilidade Técnica. Como estava bastante empolgado com o projeto, concordei com a idéia e decidi usá-la como tema deste Trabalho de Graduação.

1.3. OBJETIVOS

A proposta dos produtos interativos como a *Vitrine Interativa* é trazer mais clientes para as lojas, aumentando, conseqüentemente, a quantidade de vendas.

Além disso, esse produto pode oferecer serviços de compra de determinados itens da loja sem o cliente nem precisar entrar na mesma, ou simplesmente entreter as pessoas que passam pelo local com animações interativas em tempo real.

Os principais objetivos deste Trabalho de Graduação são:

- apresentar exemplos reais de produtos interativos;
- estudar o funcionamento de todos os equipamentos e softwares utilizados em *Vitrines Interativas*;

- investigar a Viabilidade Técnica do desenvolvimento de duas versões do produto Vitrine Interativa a serem desenvolvidas pela empresa Codemedia, em parceria com Jerônimo Barbosa.

1.4. METODOLOGIA DE TRABALHO

A metodologia utilizada para a realização do presente trabalho foi inicialmente a pesquisa bibliográfica sobre conceitos de Realidade Aumentada e pesquisas sobre produtos interativos já existentes.

Em seguida, foram realizados estudos de embasamento técnico de softwares e equipamentos necessários para o desenvolvimento desses produtos, em especial a Vitrine Interativa.

Além disso, foram realizados testes, com vários desses softwares e equipamentos, em lojas de rua e de shopping center. Com esses testes, foram obtidos vários resultados que contribuíram para a escolha do que será utilizado nas versões VI e VI+.

1.5. ESTRUTURA DO DOCUMENTO

A seguir será descrita a distribuição dos tópicos ao longo dos capítulos deste documento:

- **2 – Fundamentação Teórica:** este capítulo introduz as principais definições envolvidas no domínio abordado. Conceitos, modelos e exemplos são exibidos nesse capítulo na intenção de serem utilizados eventualmente durante o desenvolvimento do trabalho aqui descrito;
- **3 – Estudo da Viabilidade Técnica:** apresenta e analisa, do ponto de vista técnico, os equipamentos, acessórios e softwares necessários para o desenvolvimento de ambas as versões VI e VI+;
- **4 – Desenvolvimento da VI:** discute sobre os itens apresentados no capítulo anterior que foram utilizados no desenvolvimento e nos testes da versão VI;

- **5 – Considerações Finais:** neste capítulo, se encontram as principais conclusões retiradas desse trabalho. Nele, são descritos os conteúdos de maior importância gerados durante esse estudo, além de apresentados os trabalhos futuros que possam surgir a partir deste trabalho.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, será, inicialmente, apresentada uma fundamentação teórica nas áreas de Realidade Aumentada e Interação Usuário-Máquina que apresentará conceitos importantes no desenvolvimento de produtos interativos, que, por sua vez, serão apresentados e terão exemplos reais mostrados.

2.1. CONCEITOS DE REALIDADE AUMENTADA E INTERAÇÃO USUÁRIO-MÁQUINA

Segundo [Ronald Azuma, 1993], a Realidade Aumentada é um ambiente que envolve tanto realidade virtual como elementos do mundo real, criando um ambiente misto em tempo real. Ou seja, é há uma integração entre o mundo real e objetos virtuais criados pelo computador.

É definida, no geral, como um sistema que:

- É interativo e tem processamento em tempo real;
- Combina elementos do mundo real com elementos virtuais;
- É, geralmente, concebida em três dimensões.

Em várias aplicações de RA, objetos virtuais tridimensionais, gerados por computador, são sobrepostos em um ambiente real. Para isso, são necessários três componentes básicos:

- Objeto real com alguma marca de referência, que possibilite a interpretação e criação do objeto virtual pelo computador;
- Câmera capaz de transmitir a imagem desse objeto real;
- Software de visão computacional capaz de analisar o vídeo transmitido pela câmera e, a partir dele, calcular a posição exata relativa ao marcador do objeto real. Com isso, o objeto virtual pode ser gerado, em tempo real, na localização exata do marcador.

Estes softwares de visão computacional utilizam conceitos de Realidade Aumentada e Interação Usuário-Máquina e algoritmos que são fundamentais para que a interação ocorra em tempo real. Os principais são:

- **Video tracking:** processo de localização de um ou mais objetos reais através de uma ou mais câmeras. O tipo de video tracking mais utilizado é o de *blob* tracking, que pode ser definido como regiões de uma imagem que diferem da imagem anterior captada pela câmera. Esse processo de atualização das imagens ocorre, em média, 60 vezes por segundo, caracterizando o processo como video tracking.
- **Visão Computacional:** estuda a tecnologia das máquinas que “vêem”. Analisa sistemas que extraem informações de imagens. Esses sistemas são responsáveis, na maioria dos casos, por calcular a localização exata de gestos do usuário que está sendo capturado por uma câmera.
- **Reconhecimento de objetos:** define técnicas de encontrar objetos em imagens ou vídeos. Alguns exemplos de técnicas são: Reconhecimentos por partes, Métodos baseado em aparência e Métodos baseados em características.
- **Detecção de blobs:** é uma técnica de processamento de imagens que consiste em aplicar algoritmos com o objetivo de detectar *blobs*, que podem ser gestos de um usuário ou algum marcador utilizado pelo usuário. Depois de calculada a posição desse marcador, ela também é usada para a criação do objeto virtual. Com isso, através do computador, será possível a visualização desse objeto virtual sobre o marcador real, pertencendo ao mesmo ambiente.

A variedade de aplicações que utilizam esses conceitos é muito grande. Uma das mais conhecidas e praticadas em todo o mundo é a biblioteca *ARTool/Kit*. É disponibilizada como open-source no site do laboratório HITL da Universidade de Washington.

Ela emprega métodos de visão computacional para detectar marcadores (ou tags) na imagem capturada por uma câmera. Depois de calculada a localização exata desse marcador, um objeto virtual é atrelado a este marcador, provocando a mistura entre ambiente real e virtual. A partir daí, o usuário pode mover o marcador e o objeto virtual vai interagir em tempo real.

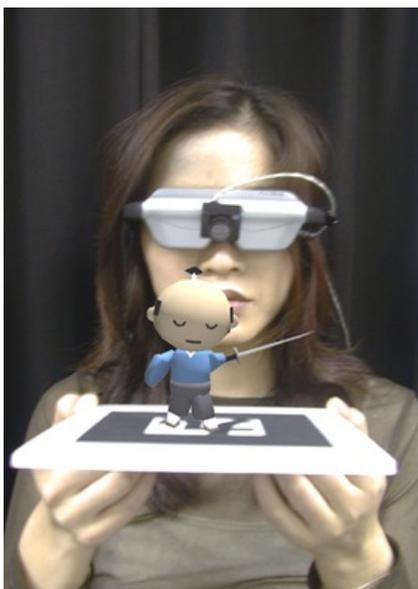


Figura 1 - Exemplo de ARToolKit

Na figura 1, é possível identificar o objeto virtual (o samurai) no mesmo ambiente dos objetos reais “atrelado” ao marcador.

Algumas das aplicações de Realidade Aumentada são utilizadas pela Mídia para a divulgação de produtos e para entreter e chamar atenção dos consumidores. Estas aplicações podem ser encontradas em revistas ou outros meio de Propaganda e em programas de televisão, por exemplo.

Essas aplicações serão abordadas ao longo deste trabalho como Produtos Interativos, pois são produtos compostos de equipamentos, acessórios e softwares que proporcionam uma interação em tempo real entre usuários e aplicações projetadas na vitrine, no piso, ou em mesas, dependendo do tipo de produto.

Um dos mais conhecidos é a Vitrine Interativa, que será abordada detalhadamente na próxima seção.

2.2. VITRINE INTERATIVA E OUTROS PRODUTOS INTERATIVOS

Nesta seção, serão apresentados e definidos os produtos interativos mais conhecidos e utilizados: a Vitrine Interativa e o Piso Interativo. Será mostrado, também, o funcionamento básico e alguns exemplos reais, com ênfase à Vitrine Interativa, por ser o tema deste trabalho.

Além disso, também serão apresentados alguns conceitos e exemplos reais de outros tipos de produtos interativos, como a Tela Multitoque e a Mesa Multitoque. As referências desses exemplos reais estão listadas nos anexos deste trabalho.

2.2.1. Vitrine Interativa



Figura 2 – Exemplo de Vitrine Interativa

A Vitrine Interativa pode ser definida como um produto que proporciona uma interação em tempo real entre um ou mais usuários e uma aplicação projetada na vitrine de uma loja. Essa interação pode ser feita através de:

- toques no vidro (*multitouch*);
- gestos ou outros movimentos realizados na frente do vidro (*touchless*).

Alguns produtos permitem a interação de mais de um usuário ao mesmo tempo, enquanto que outros só permitem uma pessoa por vez. Os produtos multitouch permitem a captação simultânea de vários toques, o que significa que mais de um usuário pode interagir ao mesmo tempo. Porém, alguns

produtos touchless não têm a capacidade de detectar mais de um usuário por vez.

Qualquer tipo de Vitrine Interativa que faça uso da abordagem *touchless*, ou seja, sem toques, deverá possuir um conjunto básico de equipamentos e softwares para se tornar funcional. Tal conjunto pode ser dividido da seguinte maneira:

Equipamentos e acessórios:

- **projektor:** necessário para projetar a aplicação no vidro. Precisa ser posicionado na horizontal.
- **câmera:** utilizada para captar os movimentos e a exata posição do usuário que participa da interação;
- **computador:** utilizado para executar os softwares envolvidos no processo e conectá-los a outros equipamentos;
- **tela de projeção:** usada para tornar a projeção mais visível.
- **cabos:** necessários para as conexões entre os equipamentos e entre eles e as fontes de energia.

Softwares:

- **tracker:** software de visão computacional utilizado na detecção dos movimentos do usuário;
- **aplicação final:** aplicação projetada na vitrine com a qual o usuário vai interagir;
- **pré-requisitos de software:** softwares necessários para o funcionamento do *tracker* e da aplicação final.

Essa lista pode apresentar mais ou menos equipamentos e/ou softwares, dependendo de fatores como:

- **Iluminação:** tipo de iluminação do ambiente em que o produto for ser utilizado. Caso o ambiente tenha iluminação inconstante, como as lojas de rua que sofrem incidência dos raios solares, será necessário

a presença de um ou mais canhões (ou refletores) de luz infravermelha. Com isso, a câmera precisará ser adaptada para reconhecer apenas este tipo de luz.

- **Espaço:** tamanho do espaço disponível para a projeção. Grandes vitrines podem requerer grandes projeções. Com isso, mais de uma unidade de alguns equipamentos deverão atuar interligados à apenas um conjunto de softwares. Ou seja, serão necessários mais de um projetor e mais de uma câmera, que, por sua vez, deverão se conectar a apenas um tracker.
- **Equipamento de saída de vídeo:** na grande maioria dos casos, um projetor é utilizado. Porém, ele pode ser substituído por uma TV LCD, que será fixada na vitrine e reproduzirá a aplicação final.

Depois de apresentado o conjunto básico que faz uma Vitrine Interativa funcionar, será explicado como é o funcionamento básico deste produto.

2.2.1.1. Funcionamento básico

Para que o produto funcione, todos os equipamentos precisam estar devidamente fixos, posicionados e ligados, os softwares devem estar instalados, configurados e no estado de execução e todas as conexões entre esses softwares e equipamentos precisam estar feitas.

A partir daí, o funcionamento básico da Vitrine Interativa pode ser descrito da seguinte maneira:

1. O usuário se posiciona à frente da vitrine, entrando no campo de visão da câmera;
2. Esta câmera prontamente detecta que há “corpos estranhos” no seu campo de visão, que são representados por toques ou movimentos do usuário;
3. O *tracker* (que atua diretamente ligado à câmera) calcula as posições exatas desses corpos;

4. O *tracker* formata essas posições em mensagens e as envia pela rede à aplicação final;
5. A aplicação recebe e decodifica essas mensagens;
6. Os pontos formados pelas posições desses corpos estranhos interagem com a aplicação que está projetada na vitrine.

Essas “etapas” são realizadas em intervalos de tempo mínimos, ou seja, imperceptíveis ao olho humano. Por isso, esse funcionamento é caracterizado como um funcionamento em tempo real.

Esse processo de funcionamento, juntamente com todos os itens citados, serão detalhados no capítulo seguinte.

2.2.1.2. Exemplos reais

O produto da Vitrine Interativa, do ponto de vista comercial é, geralmente, utilizado como objeto de Marketing e Propaganda. Na grande maioria dos casos, esse produto é utilizado por lojas e bares/restaurantes para a divulgação de seus produtos/serviços.

É um dos artefatos mais usados na Mídia Interna ou Mídia Indoor, que pode ser definida como qualquer tipo de propaganda ou divulgação feita na área interna de algum estabelecimento.

Na maioria dos casos, as empresas que desenvolvem produtos como a Vitrine Interativa são empresas de tecnologia que trabalham com soluções de Publicidade. Geralmente, elas “alugam” o produto para lojas que desejam chamar a atenção de seus clientes através da vitrine. O preço desse “aluguel” é, na maioria dos casos, cobrado de acordo com o tempo que a loja deseja usufruir desse produto.

A variedade de lojas que alugam esse produto permite um grande número de aplicações finais interativas que podem ser utilizadas. As mais comuns são: animações simples, cardápios virtuais, sites, manequins virtuais e imagens em geral.

A seguir, alguns exemplos utilizados em lojas de rua ou de shopping center:

Vitrine Interativa em lojas de rua:

- **Vitrine Interativa da *La Vitrine (CAN)*** - desenvolvido pela empresa canadense Moment Factory, esse produto foi instalado na vitrine dessa loja, situada no centro da cidade de Montréal, no Canadá. A instalação incluía dispositivos de rastreamento combinados com 35.000 lâmpadas leds e é capaz de mostrar vários conteúdos baseados na presença e circulação de pessoas.
- **Vitrine Interativa da *Ralph Lauren (EUA)*** - instalada numa loja de rua, oferecia um catálogo na vitrine. Com isso, o usuário tocava na tela e adicionava roupas a um carrinho virtual, deixando seu telefone para contato, no final da compra;
- **Vitrine Interativa da *Apple Store (ALE)*** - desenvolvido pela empresa Media Zest, este produto dispara umas animações à medida que alguém passa pela frente das vitrines da loja.
- **Vitrine Interativa da *Virgin (EUA)*** - desenvolvido pela Adobe, esse produto ativa animações diversas na vitrine quando pessoas passam pela frente da loja.

Vitrine Interativa em lojas de shopping center:

- **Vitrine Interativa da *LG (BRA)*** - desenvolvido pela agência criativa, este produto está sendo utilizado para divulgar a nova tecnologia presente nas TVs de LCD da LG, o NetCast™. Esta tecnologia oferece acesso à Internet através da TV.
- **Vitrine Interativa da *C&A (BRA)*** – Vitrine Interativa que foi utilizada para mostrar alguns produtos da coleção de 2008.
- **Vitrine Interativa da *Diesel (ALE)*** – o cenário projetado na vitrine interagia com as pessoas à medida que elas se movimentavam;

- **Vitrine Interativa do *Atlântico Shopping (BRA)*** - neste caso, o sistema detectava o movimento das mãos do público. Com isso, o usuário podia interagir com um celular gigante projetado na vitrine. Além disso, o projeto contava com transmissão ao vivo do local, através da Internet.
- **Vitrine Interativa da *Orange (ING)*** - produto que permitia que o usuário controlasse a aplicação projetada através de gestos, sem necessidade de toques.

2.2.2. Piso interativo



Figura 3 – Exemplo de Piso Interativo

Outro tipo de produto interativo muito utilizado na Mídia Indoor é o Piso Interativo, também conhecido por Chão Interativo, que pode ser definido como um produto que proporciona uma interação em tempo real entre um ou vários usuários e uma aplicação projetada no piso de algum estabelecimento. Essa interação é feita através da detecção de movimentos sobre a projeção.

Em sua maioria, são produtos que permitem a participação de mais de um usuário. Muitos exemplos, inclusive, são de jogos que exigem dois usuários.

Este produto utiliza os mesmos equipamentos e softwares da Vitrine Interativa. No entanto, esses equipamentos precisam ser posicionados no teto ou em outra estrutura, de maneira que fiquem voltados para baixo, na vertical, como ilustrado na figura 4.



Figura 4 - Equipamentos de um Piso Interativo

Devido ao posicionamento, esse tipo de produto não pode ser utilizado em ambientes com teto baixo, pois, nesse caso, a projeção será muito pequena, inviabilizando uma boa interação. Além disso, a altura não pode ser baixa, pois, caso contrário, os equipamentos podem atrapalhar o fluxo de pessoas sobre a área do piso onde se encontra a projeção.

2.2.2.1. Funcionamento básico

O Piso Interativo apresenta um funcionamento básico muito similar ao da Vitrine Interativa. As etapas são basicamente as seguintes:

1. O usuário anda sobre a área do piso onde se encontra a projeção;
2. A câmera prontamente detecta que há “corpos estranhos” no seu campo de visão, que são representados por movimentos do usuário;
3. O *tracker* (que atua diretamente ligado à câmera) calcula as posições exatas desses corpos;

4. O *tracker* formata essas posições em mensagens e as envia pela rede à aplicação final;
5. A aplicação recebe e decodifica essas mensagens;
6. Os pontos formados pelas posições desses corpos estranhos interagem com a aplicação que está projetada no piso.

Assim como na Vitrine Interativa, os intervalos de tempo entre essas etapas são imperceptíveis ao olho humano, caracterizando o processo como interação em tempo real.

2.2.2.2. Exemplos reais

O Piso Interativo também é muito utilizado por lojas que desejam divulgar seus produtos. Porém, ao contrário da Vitrine Interativa, a projeção não tem como objetivo principal atrair novos clientes que passam pela frente da loja, e sim chamar a atenção dos clientes que já estão do lado de dentro.

Esse produto também pode ser utilizado apenas como forma de entretenimento para as pessoas que estiverem passando no local. Podem ser encontrados em eventos, como feiras e exposições, e em locais públicos com grande circulação de pessoas, como aeroportos, museus e corredores de shopping center.

Assim como na Vitrine Interativa, a variedade de aplicações finais é muito grande. As mais utilizadas em Pisos Interativos são: jogos, animações simples e imagens.

A seguir, uma lista com alguns exemplos em diferentes ambientes:

Piso Interativo em exposições:

- **Piso Interativo da *Innovation Expo (HK)*** - Piso Interativo utilizado na exposição Innovation Expo 2005, em Hong Kong.

Piso Interativo em cassinos:

- **Piso Interativo do *MGM Grand (EUA)*** - permitia uma série de animações diferentes.

Piso Interativo em locais públicos:

- **Piso Interativo da *Barclays (POR)*** - utilizado no aeroporto de Lisboa, esse produto mudava a sequência de imagens exibidas, à medida que as pessoas passavam sobre a projeção.
- **Piso Interativo da *Nissan (POR)*** - também utilizado no aeroporto de Lisboa, este Piso provocava uma série de animações no logotipo da Nissan quando alguém passava sobre a área do Piso Interativo.
- **Piso Interativo da *Coca-Cola (BRA)*** - desenvolvido pela YDreams, empresa com foco na área de Realidade Aumentada, este produto disparava uma série de animações em imagens com produtos da Coca-Cola.

2.2.3. Outros produtos de interação em tempo real



Figura 5 - Exemplo de Mesa Interativa Multitouch

Outros tipos de produtos interativos podem ser encontrados. Em especial, os produtos *Multitouch* ou *Multitoque*, que são sensíveis a toques do usuário.

Os equipamentos e softwares são, basicamente, os mesmos citados anteriormente. Porém, grande parte desses produtos faz uso de sensores instalados em placas de acrílico que são responsáveis por detectar a posição exata dos toques do(s) usuário(s).

Segue, abaixo, uma pequena lista com alguns exemplos de aplicações multitoque:

Tela Multitoque:

- **Tela Multitoque do *Fantástico (BRA)*** - tela existente no programa de Jornalismo da Rede Globo que responde aos toques do apresentador;

Mesa Multitoque:

- **Mesa Multitoque da *Microsoft (EUA)*** - conhecida como Microsoft Surface, essa mesa se trata de um “computador de superfície” vendido mundialmente, que pode ser utilizada para qualquer funcionalidade disponível programada pelo cliente;
- **Mesa Multitoque da *SmartTech (EUA)*** - Mesa interativa conhecida como Smart Table. É muito similar à Microsoft Surface. Porém, é voltada para a educação infantil.
- **Mesa Multitoque *iBar (ING)*** - Mesa Interativa utilizada em bares, boates e restaurantes.

Como se pôde perceber, existem várias abordagens para a construção de produtos que ofereçam uma interação com o usuário em tempo real

Para a escolha de um desses produtos, a empresa deve levar em consideração, principalmente, qual:

- aplicação final será utilizada;
- o ambiente disponível para a projeção dessa aplicação.

A partir daí, é possível definir qual tipo de produto interativo é o mais adequado para a situação.

Conforme mencionado anteriormente, os produtos mais conhecidos e comercializados são:

- Vitrine Interativa
- Piso Interativo
- Tela Multitoque
- Mesa Multitoque

Porém, este Trabalho de Graduação terá como foco apenas a Vitrine Interativa. As outras abordagens, portanto, não serão mais abordadas neste trabalho.

3. ESTUDO DE VIABILIDADE TÉCNICA

Neste capítulo, as duas versões serão apresentadas, inicialmente, e analisadas do ponto de vista técnico, ou seja, se são viáveis tecnicamente de serem desenvolvidas. Ou seja, serão estudados a disponibilidade no mercado e o desempenho dos equipamentos necessários para os dois tipos de VI mencionados na Motivação deste trabalho, além de um estudo sobre os softwares que deverão estar presentes em cada uma das versões e também sobre como eles deverão se comunicar com os equipamentos para oferecer uma interação de qualidade em ambas as versões.

Abaixo, segue a divisão de seções deste capítulo:

- VI e VI+: define e diferencia as duas versões que serão discutidas ao longo do trabalho.
- Equipamentos: lista os equipamentos e acessórios necessários, com suas devidas especificações.
- Softwares: apresenta os softwares fundamentais para o funcionamento de ambas as versões.

3.1. VI e VI+

Nesta seção, será apresentado um breve detalhamento sobre os requisitos e especificações de cada versão.

- **VI** - A versão mais básica consiste num produto interativo em tempo real baseado numa abordagem touchless de uma Vitrine Interativa. Este produto vai utilizar um projetor com especificações relativamente modestas, uma câmera simples adaptada para captar apenas luz visível e um computador de performance razoável que possua conexão com todos esse equipamentos e seja compatível

com todos os softwares necessários. Desses softwares, o *tracker* deverá calcular apenas um ponto central relativo ao gesto do usuário. Com isso, a aplicação final deve ser pontual. Ou seja, utilizar apenas esse ponto, simulando o ponteiro de um mouse. De maneira resumida, a VI pode ser definida como um produto que vai utilizar equipamentos e acessórios de preços relativamente acessíveis, e softwares com poucas funcionalidades. Além disso, vai trabalhar apenas com interações pontuais e exigirá ambiente com iluminação constante, devido ao não uso de canhões infravermelho.

- **VI+** - A versão mais completa também consiste num produto interativo em tempo real baseado na abordagem Vitrine Interativa. Esta versão fará uso de um projetor com especificações mais robustas, uma câmera adaptada para captar apenas luz infravermelha e um computador de grande performance que seja compatível com todos os equipamentos e softwares utilizados. Dentre eles, um *tracker* que consiga calcular tanto o ponto central relativo ao usuário como todos os pontos da silhueta do usuário. A aplicação, portanto, poderá ser pontual ou utilizar todos esses pontos da silhueta na interação. Resumindo, a VI+ será um produto que vai utilizar equipamentos e acessórios de preços mais caros, e softwares com requisitos mais complexos. Vai trabalhar tanto com interações pontuais como com interações que utilizarão a silhueta de todo o corpo do usuário ou de parte dele e poderá ser instalado em qualquer ambiente, com iluminação constante ou variável.

Nas próximas seções deste capítulo, serão discutidas as especificações mais importantes tanto dos equipamentos e acessórios, como dos softwares. No final da discussão de cada item, serão apresentadas quais as especificações mínimas que a VI e a VI+ devem ter pra este item.

3.2. EQUIPAMENTOS E ACESSÓRIOS

Nesta seção, todos os equipamentos e acessórios para cada versão do produto, como computadores, projetores e câmeras, serão analisados individualmente, levando em consideração as especificações de cada um exigidas por cada versão.

3.2.1. Projetor

O projetor tem a função de projetar a aplicação flash na vitrine. Ele deve ser fixado no teto ou em alguma estrutura alta. Para isso, é necessário, na maioria dos casos, um suporte que deixe o aparelho firme e, ao mesmo tempo, não atrapalhe a circulação de pessoas no ambiente.

Para uma projeção de qualidade, não pode haver nenhum objeto ou estrutura. Para que isso seja possível, a arrumação da vitrine deve ser adaptada para que não haja nenhum objeto, como manequins, por exemplo, entre o projetor e a área do vidro reservada à projeção. Além disso, o foco deve ser ajustado para proporcionar a imagem de maneira mais nítida possível.

Em relação às especificações de um projetor a ser utilizado num produto como a Vitrine Interativa, deve-se destacar:

- **Luminosidade:** a luminosidade de um projetor é a quantidade de fluxos luminosos, ou seja, a quantidade total de luz emitida pelo projetor que pode produzir estímulo visual, medida em LUMENS. Porém, o grau de luminosidade de projetores são medidos em ANSI LUMENS, um padrão determinado pela Instituição Americana de Padrões (ANSI). Os projetores utilizados em produtos interativos possuem, normalmente, entre 2.000 e 5.000 ANSI LUMENS.
- **Tipo de display:** Os mais comuns são: DLP e LCD. O primeiro, Digital Light Processing, é mais recente e consiste em milhares de espelhos movendo-se para trás e para frente dentro do equipamento. Já o segundo, Liquid Crystal Displays, é bem mais comum e consiste em milhares de pontos que se acendem e apagam quando necessário.

- **Distância focal:** indica a menor/maior distância do projetor pra tela, de modo que permita que este tenha foco. Essa distância pode ser calculada através de vários atributos, como as distâncias focais mínima e máxima das lentes do equipamento
- **Razão de aspecto:** é uma média da divisão entre largura e altura da imagem do projetor. O valor mais comum é o de 4:3.
- **Resolução nativa:** pode ser definida como a resolução padrão que o projetor foi configurado para exibir. As resoluções mais comuns são as SVGA (800 x 600) e XGA (1024 x 768).
- **Vida útil média da lâmpada:** se define como a duração da vida útil da lâmpada. Normalmente, varia de 2.000 a 5.000 horas.
- **Entradas de vídeo:** são as entradas de vídeo disponíveis no projetor. Para poder se conectar ao computador, ele deve ter, no mínimo, a entrada padrão VGA. Porém, pode-se ter também outros tipos de entradas que podem melhorar o seu tempo de resposta e a qualidade da imagem projetada, como a DVI, e até permitir o uso de sons na aplicação, como a HDMI.

O projetor, porém, não é o único tipo de equipamento de saída de vídeo que pode ser usado por produtos interativos. Outro tipo bastante utilizado é a TV LCD. A vantagem de usá-la é o seu baixo custo, quando comparada ao projetor. Entretanto, a montagem se torna um pouco mais complexa, já que a TV precisa ser fixada na vitrine rente ao vidro, para que o reflexo do vidro não atrapalhe a visibilidade da projeção.

Ambas versões discutidas neste trabalho farão uso de um projetor de mais de 2500 ANSI LÚMENS, visto que os testes realizados em lojas de shopping center e de rua mostraram que projetores com luminosidade mais baixa que esse valor ofereceriam uma área insuficientemente brilhante para o propósito do produto, o que tornaria a imagem da projeção pouco visível.

Com isso, as especificações básicas do projetor para cada versão da VI puderam ser divididas da seguinte maneira:

- **VI:**
 - **Luminosidade:** 2.500 a 3.000 ANSI LUMENS;
 - **Tipo de display:** LCD;
 - **Distância focal:** 3 metros;
 - **Razão de aspecto:** 4:3;
 - **Resolução nativa:** SVGA (800 x 600) ;
 - **Vida útil média da lâmpada:** 2.000 a 3.000 horas;
 - **Entradas de video:** VGA.

- **VI+:**
 - **Luminosidade:** 3.000 a 4.000 ANSI LUMENS;
 - **Tipo de display:** LCD;
 - **Distância focal:** 5 metros;
 - **Razão de aspecto:** 4:3;
 - **Resolução nativa:** XGA (1024 x 768) ;
 - **Vida útil média da lâmpada:** 3.000 a 5.000 horas;
 - **Entradas de video:** VGA, DVI e HDMI.

3.2.2. Câmera

A câmera pode ser considerada como um dos equipamentos mais importantes de qualquer produto interativo, visto que é utilizada como principal meio de entrada de informação.

Este equipamento é responsável por detectar “corpos estranhos” no seu campo de visão. Estes corpos podem ser algum gesto do usuário ou até a silhueta do corpo do próprio usuário. A partir dessa detecção, esses corpos, mais conhecidos como *blobs*, terão suas posições exatas calculadas pelo *tracker*.

Para ser reconhecida e utilizada no computador, a câmera precisa ter um software instalado neste computador para poder ser reconhecida por ele, chamado de *driver*. Este deve ser compatível com a plataforma e com a versão do Sistema Operacional do computador.

Após a instalação desse driver e da devida configuração do equipamento, é necessário definir o melhor posicionamento para a câmera. Geralmente, esse posicionamento acontece nas partes superior ou inferior do vidro da vitrine, com a câmera voltada para fora da loja. Dessa forma, ela terá uma visão ampla da área da frente da vitrine e não terá obstáculos que possam atrapalhar seu campo de visão.

O próximo passo é a calibração, que é feita através do software de trackeamento, o *tracker*. Por isso, esse processo será explicado na seção que aborda este software.

Em relação às especificações deste tipo de equipamento que mais devem ser levadas em consideração para o uso em produtos interativos, é possível listá-las da seguinte maneira:

- **Resolução:** quanto maior a resolução, maior é a precisão, pois, mais pixels da tela estarão “disponíveis” para detectar *blobs*. Ou seja, os *blobs* poderão ser representados por mais pixels, aumentando a realidade. A resolução mais comum em câmeras para produtos como a Vitrine Interativa é de 320 x 240. Porém, também é possível encontrar algumas com 640 x 480 ou mais.
- **Frames por segundo:** representa a velocidade de atualização dos *frames* (quadros) gravados pela câmera. Em outras palavras, representa quantos frames a câmera captou em um segundo. Quanto maior essa velocidade, mais informação foi gravada num

determinado período de tempo. Um valor mínimo satisfatório para essa especificação de câmeras usadas em produtos interativos é de 30 FPS (frames por segundo). Com esse valor, é possível captar movimentos rápidos, sem perdas significativas de informação. Valores mais altos representam captações mais suaves e fidedignas. Esta especificação está diretamente relacionada à resolução, pois, quanto maior a resolução, menor o FPS, já que imagens maiores levam mais tempo para serem processadas. Ou seja, o computador a que a câmera está conectada pode não ter capacidade de receber imagens numa resolução de 640 x 480 a uma velocidade de 60 FPS, pois muito processamento é exigido. Neste caso, ele só será capaz de receber imagens nesta resolução a velocidades menores.

- **Plataforma e versão do S.O. (driver):** indica se a câmera está ou não disponível para serem usadas em certos computadores. Por exemplo, uma câmera só possui drivers desenvolvidos para a plataforma Linux. Com isso, ela não vai ser reconhecida em outro tipo de plataforma e, portanto, não poderá ser usada por um computador cuja plataforma não seja Linux.
- **Tipo de plug:** grande parte das câmeras utilizadas em produtos interativos possuem conexão USB ou FireWire (tecnologia de entrada e saída de dados em alta velocidade definida no IEEE como padrão 1394). As que utilizam USB são mais comuns por serem mais baratas e pelo fato desse tipo de plug ser muito popular. Já as que utilizam o segundo tipo são menos encontradas por terem preços elevados. Porém, têm como vantagem a pequena latência na transmissão da imagem da câmera para o computador, representando maiores valores de FPS.
- **Filtros de luz:** grande parte das câmeras possuem um filtro bloqueador de luz infravermelha em suas lentes, com o objetivo de não permitir que essa luz chegue ao sensor do equipamento. Isso é feito, basicamente, para evitar distorções na imagem e detecção de falsos *blobs*. Porém, alguns produtos interativos utilizam a idéia

contrária, ou seja, a câmera deve ser capaz de captar apenas luz infravermelha. Para isso, elas precisam ser adaptadas. Esse processo de modificação se trata da retirada do filtro das lentes do equipamento. Ele deve ser aberto cuidadosamente e o filtro bloqueador de luz infravermelha localizado entre as lentes e o sensor deve ser removido. Nesse momento, outro filtro deve ser colocado no mesmo local. Este, no entanto, deve ser um filtro bloqueador de luz visível ao olho humano, a luz branca. Este filtro pode ser feito por algumas camadas de filme de fotografia. O processo detalhado de adaptação da câmera podem ser conferidos em tutoriais na Internet. A partir dessa adaptação, a câmera está apta a captar apenas luz infravermelha. Esse não é um procedimento obrigatório, pois, como já mencionado, produtos interativos também funcionam (em ambientes com iluminação constante) sem o uso de canhões infravermelho. No entanto, a maior parte dos desenvolvedores desse tipo de produto prefere trabalhar apenas com luz infravermelha, com o objetivo de evitar distorções de imagem com mudanças da iluminação local que possam vir a surgir. É importante mencionar que esse procedimento de modificação na câmera não é válido para as câmeras mais sofisticadas, pois elas utilizam bloqueadores de infravermelho nas próprias lentes. Nesses casos, é necessário a troca das lentes.

De acordo com os resultados dos testes, foram definidos estas especificações mínimas para cada versão da VI:

- **VI:**
 - **Resolução:** 320 x 240;
 - **FPS:** 30;
 - **Plataforma e versão do S.O. (driver):** Windows, Windows 7 Professional;

- **Tipo de plug:** USB;
- **Filtros de luz:** Filtro de luz infravermelha (já que não será usado canhão nessa versão).
- **VI+:**
 - **Resolução:** 640 x 480;
 - **FPS:** 60;
 - **Plataforma e versão do S.O. (driver):** Windows, Windows 7 Professional;
 - **Tipo de plug:** FireWire;
 - **Filtros de luz:** Filtro de luz visível (já que será usado o canhão).

3.2.3. Canhão Infravermelho

Como afirmado anteriormente, apenas a versão VI+ fará uso do equipamento descrito ao longo desta subseção, o canhão infravermelho.

Ele atua como uma fonte luminosa e é composto por um conjunto de LEDs infravermelho, ou seja, diodos semicondutores que emitem luz infravermelha (não visível ao olho humano) quando energizados.

O uso desse equipamento é muito indicado em ambientes com iluminação inconstante, como, por exemplo, as lojas de rua. Em locais como esses, o posicionamento do sol provoca uma mudança constante no grau de luminosidade que incide sobre o vidro, podendo acarretar no surgimento de sombras dentro da área de captura da câmera. Dependendo do tamanho delas e da configuração do software de captura, esta câmera pode reconhecer essas sombras como *blobs*, quando não deveria, resultando na interação entre esses “falsos *blobs*” e a aplicação final.

Conforme mencionado na subseção anterior, quando esse canhão é utilizado, a câmera precisa captar apenas luz infravermelha. Caso ela não venha de fábrica com essa propriedade, ela vai precisar ser adaptada para ser capaz de “enxergar” apenas este tipo de luz. Caso contrário, o uso do canhão será inútil e o produto não funcionaria da melhor maneira.

Além disso, é de fundamental importância que todo o campo de visão da câmera esteja bem iluminado pelo canhão, evitando o surgimento de áreas da projeção que a câmera não será capaz de visualizar, devido à falta de luz.

As especificações mais importantes de canhões infravermelhos para o uso em produtos interativos são:

- **Quantidade de LEDs:** quantidade de LEDs infravermelho do canhão. Quanto mais, maior é a intensidade de luz. Essa quantidade varia muito entre os modelos, mas os mais comuns possuem entre 50 e 150 LEDs.
- **Alimentação:** a mais comum é de saída 12V e potência 500mA. Com isso, cada canhão requer uma fonte que atenda a essas especificações. As mais conhecidas no mercado são conhecidas como fontes 12V 500mA.

Nos testes, foi utilizado apenas um canhão de 50 LEDs, que se mostrou insuficiente para o campo de visão da câmera utilizado. Porém, duas unidades deste equipamento seria suficiente para iluminar toda a área da projeção.

A partir daí, foi possível a definição de uma especificação mínima para o uso de dois canhões na VI+.

- **VI+:**
 - **Quantidade de LEDs:** 50 em cada;
 - **Alimentação:** uma fonte 12V 500mA para cada.

3.2.4. Computador

O uso de um computador é fundamental para qualquer produto interativo. Ele é responsável pela execução de todos os softwares e por conectá-los aos equipamentos.

O conjunto de especificações mais importantes de um computador a ser utilizado num produto como a Vitrine Interativa pode ser definido como:

- **Plataforma:** se define como o padrão da arquitetura e do Sistema Operacional (S.O.) de um computador. As plataformas mais usadas em produtos interativos são: Windows e Mac.
- **Versão do S.O.:** se refere à versão do Sistema Operacional utilizado no computador. As mais utilizadas são as versões mais estáveis e compatíveis com todos os softwares e equipamentos utilizados. Atualmente, as versões mais utilizadas são: Windows XP, Windows 7 e o Mac OS X.
- **Configurações de Hardware:** especificações dos hardwares mais voltados para a performance do computador, como: processador, memória RAM e placa de vídeo.

De acordo com os testes realizados, os requisitos de sistema básicos do computador para cada versão da VI puderam ser divididas da seguinte maneira:

- **VI:**
 - **Plataforma:** Windows;
 - **Versão do S.O.:** Windows 7 Professional;
 - **Configurações de Hardware:**
 - **Processador:** Pentium Core 2 Duo T7250 2 GHz;

- **Memória RAM:** 2 GB DDR2;
 - **Placa de vídeo:** 512 MB, com saída VGA.
- **VI+:**
 - **Plataforma:** Windows;
 - **Versão do S.O.:** Windows 7 Professional;
 - **Configurações de Hardware:**
 - **Processador:** Pentium Core 2 Quad Q6600 2.66 GHz;
 - **Memória RAM:** 4 GB DDR3;
 - **Placa de vídeo:** 1 GB, com saídas VGA, DVI e HDMI.

3.2.5. Acessórios

Os acessórios utilizados em produtos interativos têm a função facilitar a montagem dos equipamentos e, também, de tornar a projeção mais visível.

Podem ser de simples fitas adesivas até sofisticadas películas de projeção. O uso desses acessórios varia bastante entre produtos interativos.

Os principais acessórios utilizados em produtos interativos podem ser classificados da seguinte maneira:

- **Cabos:** todos os cabos necessários para conectar os equipamentos com o computador. No caso do projetor, um cabo VGA, DVI ou HDMI, dependendo do modelo. Dependendo do espaço destinado à montagem, pode ser necessário também um ou mais cabos extensores USB e/ou cabos extensores elétricos.
- **Tela de projeção:** tela ou película fixada no vidro, utilizada como fundo para a projeção. Os tipos mais comuns são papéis fosco, filmes plásticos adesivados, películas PVC adesivadas e translúcidas

e bases de acrílico anti-brilho de fina espessura. A escolha do tipo depende do ambiente para instalação e, principalmente, do preço. A vantagem de um papel fosco, como o papel manteiga, é o baixo custo. Porém, ele tem como desvantagem a fragilidade. O uso de telas ou películas sem transparência alguma não permite visão dos produtos da vitrine que estiverem por trás da projeção. Ou seja, o uso destas telas só é indicado para vitrines que serão exclusivas para a projeção.

- **Suporte para projetor:** suporte para prender o projetor no teto.
- **Estrutura para câmera:** estrutura que fixe a câmera quando esta for instalada na parte superior da vitrine. Pode ser utilizada, também, para os canhões infravermelhos, quando aplicados.
- **Fita adesiva:** fita necessária para prender a tela de projeção na vitrine, caso essa tela não seja adesivada.
- **Fontes de alimentação:** fontes de alimentação necessárias para o funcionamento dos canhões infravermelho, presentes apenas na versão VI+.

De acordo com os resultados obtidos nos testes, os acessórios que serão utilizados nas duas versões serão os seguintes:

- **VI:**
 - **Cabos:** um cabo VGA para conexão projetor-computador;
 - **Tela de projeção:** papel manteiga de 1,20m x 0,60m;
 - **Suporte para projetor:** suporte universal para projetor;
 - **Fita adesiva:** fita adesiva transparente;
 - **Fontes de alimentação:** não necessário.
- **VI+:**

- **Cabos:** um cabo HDMI para conexão projetor-computador;
- **Tela de projeção:** película PVC adesivada e translúcida;
- **Suporte para projetor:** suporte universal para projetor;
- **Fita adesiva:** não necessário;
- **Fontes de alimentação:** duas fontes 12V 500mA.

3.3. SOFTWARES

Neste tópico, os softwares exigidos para cada versão serão analisados individualmente. Primeiramente, serão apresentados os resultados da pesquisa para encontrar softwares já disponíveis no mercado, como open-source ou pagos, que possam ser utilizados em cada versão. Em seguida, serão definidos os drivers dos equipamentos mencionados na seção anterior e os requisitos e configurações de outros softwares necessários para o funcionamento de ambas as versões.

3.3.1. Tracker

O *tracker* é um software de visão computacional que pode ser considerado como o software mais importante de qualquer produto interativo. Sua função principal é a de calcular as coordenadas de cada ponto dos *blobs* detectados e passá-las à aplicação final, que, por sua vez, estará sendo projetada.

Ele atua diretamente ligado à câmera e é o responsável por realizar várias configurações deste equipamento, como, por exemplo, a definição do nível de detalhes que a câmera vai detectar movimentos do usuário, e outros que serão apresentados a seguir.

É também conhecido como *software de trackeamento* e possui diversas funções indispensáveis ao funcionamento do produto, como:

- **Configuração:** através deste software, é possível configurar vários filtros que vão influir direta e indiretamente na detecção de *blobs* pela câmera. Também é possível, em alguns *trackers*, modificar configurações da câmera, além de escolher qual câmera será utilizada, caso tenha mais de uma conectada ao computador. Os filtros mais utilizados serão apresentados ao longo desta seção.
- **Calibração da câmera:** técnica que permite que a visão da câmera (imagem real) seja alinhada com a visão do *tracker* (imagem trackeada), através de funções de mapeamento. Ao final da calibração, o posicionamento de uma mão do usuário, por exemplo, que for captado pela câmera será exatamente igual ao posicionamento do *blob* referente a essa mão. Pode ser considerada como uma técnica de sincronização entre o que a câmera e o *tracker* vêem. Além disso, essa técnica permite limitar as dimensões do campo de visão da câmera.
- **Detecção de blobs:** técnica que consiste em detectar pontos e/ou regiões mais brilhantes ou escuras que as áreas vizinhas nos frames provenientes da captação da câmera. Há diversos algoritmos que podem ser usados por esta técnica. O resultado da execução desses algoritmos é a identificação dos *blobs*, que são associados a um ID (identificador).
- **Cálculo das coordenadas dos pontos dos blobs:** após a calibração da câmera, o *tracker* passa a utilizar a imagem trackeada como um plano cartesiano bidimensional. Com a identificação dos *blobs* nesta imagem, ou seja, neste plano, é possível calcular as coordenadas X e Y de todos os pontos pertencentes aos blobs, associando cada conjunto de coordenadas ao ID do *blob* que ela representa.

- **Codificação e envio das coordenadas:** essas coordenadas, associadas ao respectivo ID do *blob*, precisam ser codificadas em mensagens e enviadas através de uma conexão socket à aplicação final. A codificação das mensagens pode ser através de armazenamento em *Arrays* de *String*, por exemplo, ou por formatos de texto, como o *JSON*, um formato de texto derivado da linguagem *JavaScript* que representa estruturas de dados e arrays, tratando-os como *objetos*. Pode ser adotado em qualquer linguagem e está descrito na RFC 4627.

Com a apresentação de *trackers* já desenvolvidos e utilizados em produtos interativos, os termos mencionados nesses tópicos ficarão mais claros.

Alguns desses *trackers* são desenvolvidos pela NUI Group (Natural User Interface Group), uma comunidade open-source fundada em 2006 e formada por vários desenvolvedores de todo o mundo, voltados para a área de pesquisa e desenvolvimento de Interação Homem-Máquina.

Um software produzido membros da NUI Group, é o *Community Core Vision*. Ele possui, atualmente, três versões e é bastante usado em produtos interativos, como a Vitrine Interativa. Sua primeira versão, a 1.1, é mais conhecida como *Tbeta*. A versão seguinte, a 1.2, é multiplataforma (Windows, Mac e Linux), enquanto que a 1.3, ainda em desenvolvimento, está disponível apenas na plataforma Windows, por enquanto.

O CCV, como é mais conhecido, é um software open-source com soluções de visão computacional e sensibilidade a toques. Ele dá suporte a diversas câmeras e formatos de video. Ou seja, um video pode ser utilizado para simular a visão da câmera.

A imagem a seguir ilustra a interface principal da versão 1.3 do software, com suas principais funções.

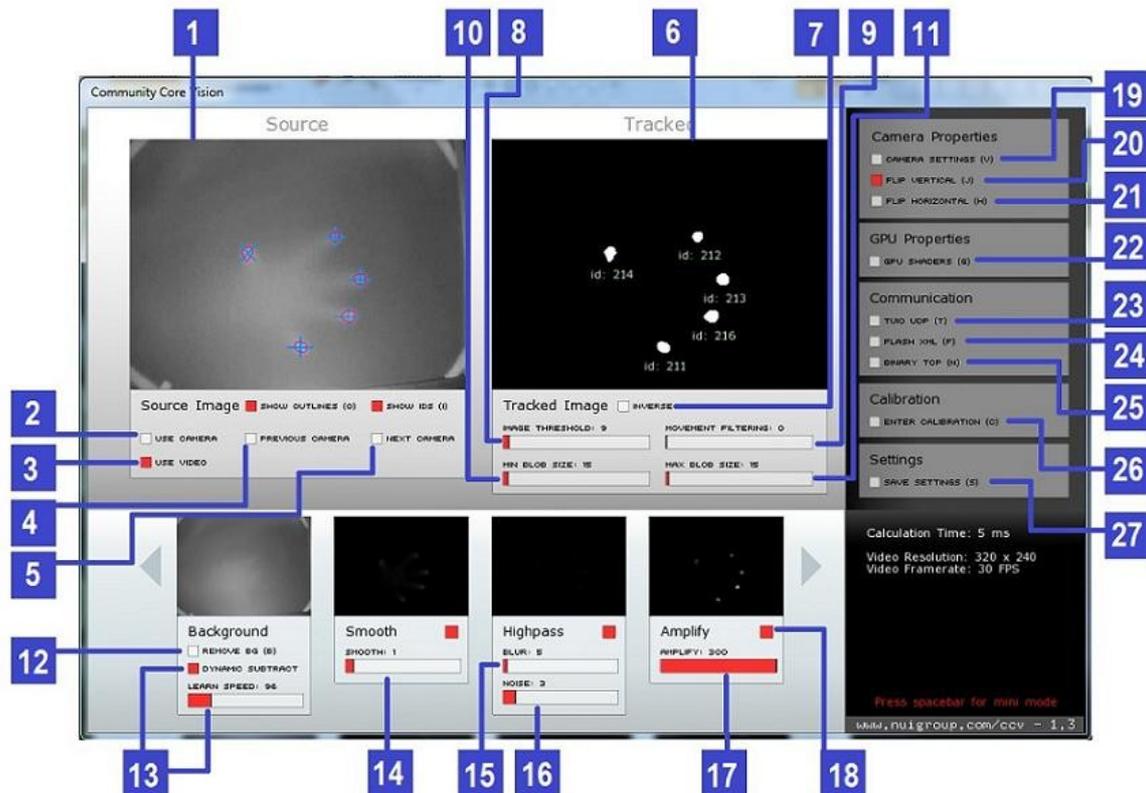


Figura 6 - Tela principal do CCV 1.3

1-Imagem real: imagem real do video ou da câmara.

2-Botão “usar câmara”: botão para se utilizar a câmara e captar imagens da câmara seleccionada.

3-Botão “usar video”: botão para se utilizar o arquivo de video, que deve estar em na pasta *data* do software.

4-Botão “câmera anterior”: botão para seleccionar a câmara anterior, caso tenha mais de uma conectada.

5-Botão “câmera seguinte”: botão para seleccionar a câmara seguinte, caso tenha mais de uma conectada.

6-Imagem Trackeada: imagem final, depois da aplicação dos filtros que serão explicados a seguir. Esta imagem é utilizada pelo software na detecção de *blobs*.

7-Inverso: mostrar *blobs* pretos ao invés de brancos.

8-Limiar: define o nível de captura de pixels que formarão os *blobs*. Ou seja, define o nível de detecção de *blobs*. Quanto maior for esse valor, mais *blobs* serão detectados.

9-Filtro de movimento: define a distância mínima (em pixels) que um *blob* deve percorrer para ser considerado um “movimento”.

10-Tamanho mínimo do blob: quanto maior essa opção, maior é o tamanho mínimo para a identificação de um *blob*.

11-Tamanho máximo do blob: quanto maior essa opção, maior a quantidade de pixels que pode ser identificada como um *blob*.

12-Botão “remove background”: técnica que captura o frame (quadro da imagem real) atual e passa a usá-la como uma imagem de fundo que deve ser comparada, a cada atualização de frame, com o frame atual. É a partir dessas comparações que os *blobs* são identificados.

13-Botão “subtração dinâmica”: botão que deve ser marcado para permitir a atualização automática do background. Indicado para ambientes com iluminação inconstante ou quando a detecção de falsos *blobs* acontece constantemente, devido à mudanças no ambiente. A barra determina a velocidade da atualização do background.

14-Suavidade: filtro que “arredonda” a imagem real.

15-Highpass Blur: têm a função de desfocar a imagem real.

16-Highpass Noise: filtro cuja função é a de adicionar ruídos à imagem real.

17-Amplitude: quanto maior, mais brilhantes os pixels *fracos*, ou seja, pouco visíveis. Com isso, torna os *blobs* mais visíveis.

18-Botão “liga/desliga”: está presente em cada filtro e tem como função habilitar ou não o filtro.

19-Botão “configuração da câmera”: abre uma janela de configurações da câmera.

20-Botão “girar verticalmente”: gira a imagem real verticalmente.

21-Botão “girar horizontalmente”: gira a imagem real horizontalmente.

22-Botão “modo placa de vídeo”: habilita o uso da placa de video. Modo ainda em desenvolvimento.

23-Botão “enviar via UDP”: botão que habilita o envio de mensagens controladas pelo protocolo TUIO (apresentado ainda nesta seção) via UDP. Exige o uso da biblioteca FLOSC, uma biblioteca de conversão UDP-TCP que será apresentada na próxima seção deste capítulo.

24-Botão “Flash XML” (disponível apenas nesta versão): botão que habilita o envio de mensagens diretamente ao Flash via TCP, sem necessidade de bibliotecas adicionais. As mensagens são convertidas em XML antes de serem enviadas.

25-Botão “TCP binário” (disponível apenas nesta versão): botão que habilita o envio, via TCP, de mensagens brutas, ou seja, as coordenadas dos blobs.

26-Botão “calibração”: carrega a tela de calibração da câmera selecionada. Deve ser o primeiro passo a se fazer depois de iniciado o software. Algumas instruções de calibração estão escritas na própria tela. Para iniciar esse processo, os equipamentos precisam estar devidamente posicionados, conectados e ligados. Com isso, a tela de calibração deve estar projetada na vitrine. Nessa tela, será possível ajustar o tamanho do campo de visão da câmera desejado. Em seguida, deve-se calibrar individualmente alguns pontos das extremidades desse campo, como ilustrado na figura abaixo. Essa ação é feita pondo, por alguns segundos, alguma fonte de luz próxima à região da vitrine onde cada ponto se encontra.

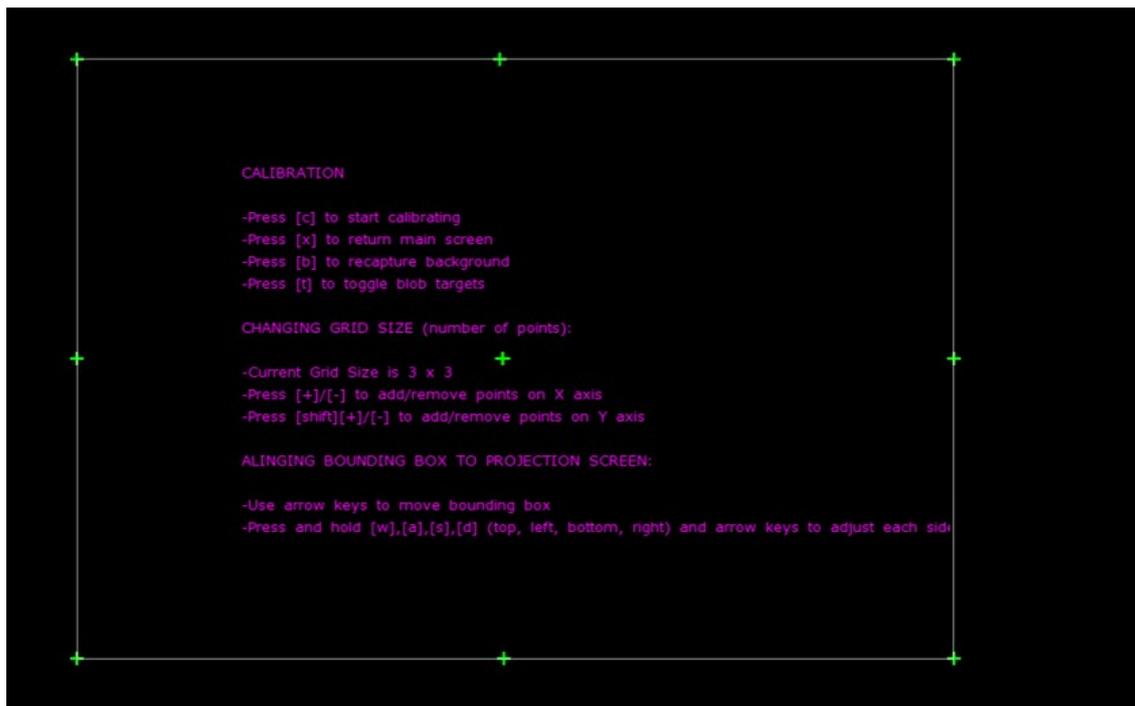


Figura 7 - Tela de calibração do CCV 1.3

27-Botão “salvar configurações”: salva todos os valores dos filtros e as outras configurações em um arquivo XML de configuração. Com isso, quando o software for iniciado novamente, as configurações permanecerão as mesmas.

A tela principal do software, representada pela figura 5, mostra, também, algumas informações da câmera, como a resolução e a velocidade (em FPS) que ela está captando no momento.

O CCV trabalha, em todas suas versões, com o protocolo alto nível TUIO (Table-Top User Interfaces Objects), que foi desenvolvido para prover uma interface de comunicação versátil entre aplicações alto nível de toque e aplicações baixo nível. Isso caracteriza o CCV como um software voltado para aplicações de toque. Porém, ele também é usado para aplicações touchless, como a Vitrine Interativa, como já mencionado.

O protocolo TUIO foi implementado utilizando como base o protocolo baixo nível OSC (Open Sound Control), que foi criado com o propósito de

controlar uma moderna rede de comunicação entre computadores, sintetizadores de som e outros tipos de dispositivos multimídia. As mensagens OSC são transportadas através do protocolo UDP.

Por ser baseado em OSC, o protocolo TUIO herda esta e outras características pertencentes ao OSC, como a sintaxe e a compatibilidade com vários softwares, como o CCV.

Uma biblioteca também muito utilizada por desenvolvedores de software de visão computacional é a *openFrameworks*. É uma biblioteca open-source, multiplataforma (Windows, Mac e Linux), voltada para artistas, designers e programadores com foco em trabalhos audio-visuais, fornecendo para isso uma interface simples e padrão para manipular diferentes tipos de mídia. A *openFrameworks* preza pela simplicidade para facilitar o uso por profissionais que não dominam técnicas de programação e que não costumam trabalhar diretamente com desenvolvimento de softwares, como artistas e designers.

Além disso, foi criada com o objetivo de ser uma biblioteca pequena e gerenciável, mas que, ao mesmo tempo, possa ter sua funcionalidade estendida através da adição de outras bibliotecas, conhecidas como *addons*. Vários deles estão presentes na versão mais recente da biblioteca, a v0.061.

Essa versão contém os arquivos que compõem a biblioteca *openFrameworks* e mais algumas aplicações de exemplo que utilizam tanto esses arquivos quanto os *addons* presentes na versão. Essas aplicações de exemplo seguem um padrão de arquitetura, que está ilustrada na figura abaixo:

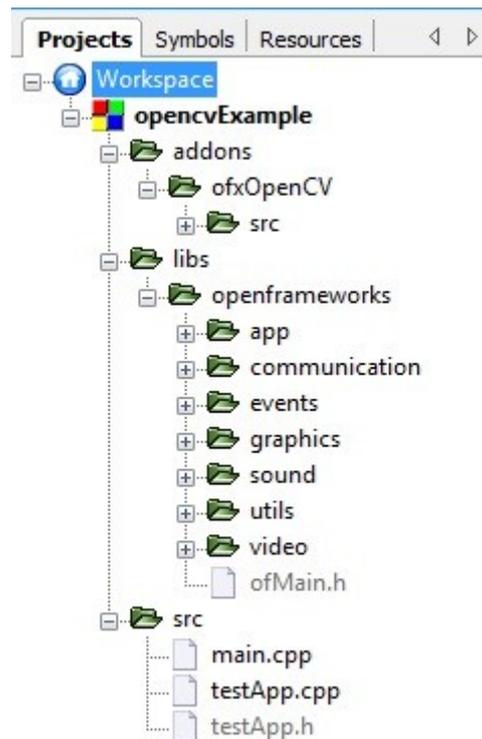


Figura 8 - projeto de opencvExample

- **addons**: pasta que contém os arquivos dos addons utilizados no projeto.
- **libs**: pasta que contém os arquivos da biblioteca openFrameworks.
- **src**: pasta que contém o código-fonte do projeto. Por padrão, contém os seguintes arquivos:
 - **main.cpp**: classe que contém apenas um método *main*. Nesse método são chamadas as funções da classe testApp.cpp.
 - **testApp.cpp**: principal classe. Contém o corpo das funções descritas na testApp.h.
 - **testApp.h**: interface que define as funções que serão implementadas na testApp.cpp.

O exemplo acima, cujo nome é *opencvExample*, mostra a arquitetura padrão do *addon* utilizado nos softwares de produtos interativos, o *ofxOpenCv*. Ele utiliza conceitos da biblioteca *OpenCV*, desenvolvida pela Intel.

Pode ser definida como uma biblioteca multiplataforma voltada para aplicações de visão computacional em tempo real. A seguir, uma figura que apresenta o *opencvExample* em execução, detectando *blobs* de um video.

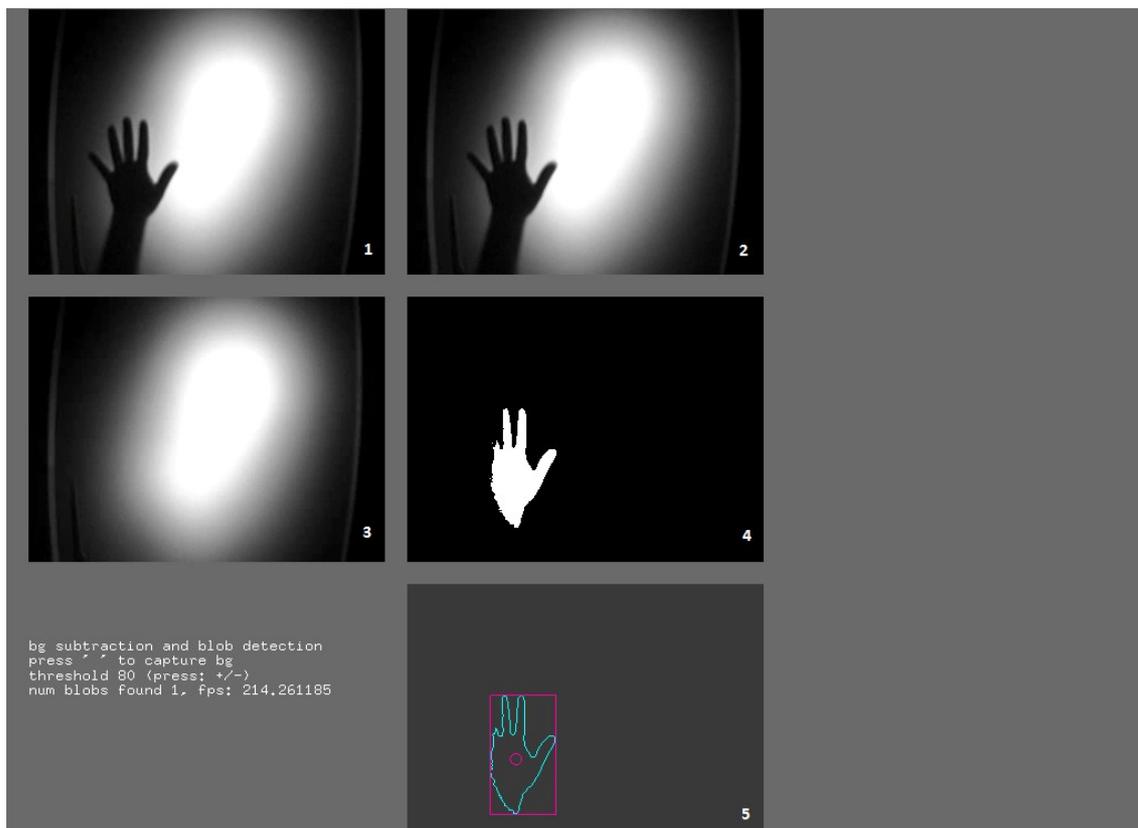


Figura 9 - opencvExample em execução

Os quadros de número 1 e 2 representam a imagem real. O de nº 3 representa a imagem utilizada como background. O de nº 4 representa a imagem trackeada, enquanto que o de nº 5 representa os blobs capturados. No caso, apenas um. Ao lado do último quadro, encontram-se algumas instruções de configuração e algumas informações sobre o video, como o nº de blobs detectados no momento e a velocidade de atualização dos frames desse video, em FPS.

Devido à simplicidade no desenvolvimento e ao fato desse exemplo ser muito utilizado em *trackers* que trabalham com aplicações pontuais, foi definido que a VI fará uso de um *tracker* baseado no addon *ofxOpenCv*. O CCV 1.3, por apresentar mais funcionalidades e ser voltado para aplicações pontuais ou que utilizem a silhueta do usuário, vai ser utilizado como *tracker* da VI+.

3.3.2. Comunicação

A transmissão das mensagens entre o *tracker* e a aplicação é feita através da rede. O que vai definir se esse tráfego vai ser na rede local, ou se vai precisar passar por redes externas é a arquitetura de distribuição desses softwares. Podem ser consideradas dois tipos principais dessa arquitetura:

- **Arquitetura Integrada** - ambos softwares estão rodando na mesma máquina. Neste caso, o tráfego das mensagens se dá apenas pela rede local da máquina. Este tipo de arquitetura é o mais indicado, pois não há tráfego externo, que pode ser considerado um grande causador de atrasos no funcionamento do produto. Porém, tem a desvantagem de concentrar todo o processamento dos dois softwares na mesma máquina, exigindo dela uma boa performance.
- **Arquitetura Distribuída** - com este tipo de arquitetura, os softwares rodam em máquinas diferentes. Ou seja, a máquina na qual a câmera está conectada roda o *tracker*, enquanto que a outra, que está ligada ao projetor ou à outro equipamento de saída de vídeo, roda a aplicação. Esta arquitetura tem a vantagem de distribuir o processamento dos softwares em máquinas diferentes, minimizando a exigência de performance de um só computador. Entretanto, há necessidade de tráfego das mensagens através de alguma rede externa, além das redes internas de cada. Caso essa rede externa esteja congestionada, há grandes chances de atrasar o processo de funcionamento do produto, acarretando em atrasos perceptíveis ao olho humano durante a interação. Neste tipo de arquitetura, ambas

máquinas devem estar conectadas na mesma rede, seja ela a Internet ou alguma rede interna local.

Esse tráfego é controlado pelos dois protocolos da camada de transporte: UDP e TCP. O primeiro é descrito na RFC 768 e ideal para aplicações em tempo real, pois é um serviço sem conexão. Ou seja, não há necessidade de se estabelecer uma conexão entre cliente e servidor. A principal desvantagem é a falta de garantia de entrega de pacotes. Porém, devido a grande quantidade de pacotes e a grande velocidade de transmissão, a perda de alguns pacotes acaba se passando despercebida.

Já o TCP, descrito na RFC 793, garante a entrega dos pacotes, exigindo, porém, uma série de passos para estabelecimento de uma conexão entre cliente e servidor, o que torna o fluxo de pacotes bem mais demorado do que com UDP.

Devido à necessidade de enviar mensagens em tempo real, os *trackers* geralmente enviam suas mensagens à aplicação final em pacotes UDP. Porém, vários tipos de aplicação não conseguem ler pacotes UDP.

Isso acontece com as aplicações Flash, que podem ser consideradas como o tipo de aplicação mais utilizado em produtos como a Vitrine Interativa. Com isso, surge a necessidade de converter esse pacotes UDP em pacotes TCP, durante o trajeto entre as duas aplicações.

Para solucionar esse tipo de impasse, foram criadas as pontes que têm como principal função receber os pacotes UDP do *tracker* e redirecioná-las, de imediato, como pacotes TCP à aplicação final, sem alterar o conteúdo das mensagens empacotadas por eles. Essa conversão é fundamental para a comunicação entre as aplicações e não atrasa o fluxo, pois as latências são desprezíveis.

O processo de conversão pode ser realizado por diferentes tipos de bibliotecas e plataformas. Por esse motivo, várias pontes estão disponíveis na Internet e, por isso, o desenvolvedor deve escolher a que mais se adeque ao

seu projeto. Grande parte delas são open-source, multiplataforma e desenvolvidas em várias linguagens, principalmente *Java* e *C++*.

Essas pontes possuem uma porta UDP de entrada, por onde ela recebe os pacotes UDP, e uma porta TCP de saída, por onde ela envia os pacotes TCP. A primeira deve ser igual à porta configurada pelo *tracker* para envio de seus pacotes UDP. Já a segunda deve ser a mesma porta configurada pela aplicação final para recebimento dos pacotes TCP.

Alguns exemplos de pontes UDP-TCP são:

- **FLOSC**: uma aplicação multiplataforma implementada em Java que atua como uma ponte entre aplicações que enviam pacotes OSC via UDP e aplicações que recebem mensagens XML via TCP, como Flash. Ele converte as mensagens OSC que chegam via UDP em XML e as envia por TCP ao cliente Flash. Esse cliente utiliza a conexão a um objeto do tipo Socket, descrito na API de ActionScript 3.0, através do IP da máquina onde o *tracker* está rodando e da porta pela qual o *tracker* envia os pacotes, para receber essas mensagens e, por fim, analisar a informação XML contida nelas. Essa ponte é utilizada pelo software CCV, mencionado anteriormente, caso seja escolhida a opção de envio por UDP.
- **OSCAR**: similar à FLOSC, porém implementada em *Cocoa* e *Python*.
- **UDP-TCP Bridge**: desenvolvida por Mehmet Akten, é uma aplicação multiplataforma (Windows e Mac) implementada em C++, utilizando alguns *addons* da biblioteca openFrameworks, como ofxNetwork, ofxThread e ofxXMLSettings, que recebe qualquer tipo de mensagem via UDP e a redireciona, sem nenhum tipo de modificação, via TCP. Abaixo, uma figura ilustrativa desta ponte:

```
UDP->TCP Bridge v0.2 by Mehmet Akten (c) 2009 | www.memo.tv
listening on UDP port: 3333
sending on TCP port: 3000
at 60 Hz | actual : 64.12 Hz
Press 'd' to toggle debug info (check console for more) | ON
07/05/10 00:10:56 | 1966,359 ; sending 46 bytes
```

```
D:\Guilherme\Documents\Code Media\VI\bin - tracker + bridge + flash\udp-tcp-bridg...
26 8=56
27 7=55
28 5=53
29 0=48
30 0=48
31 0=48
32 ,=44
33 0=48
34 .=46
35 5=53
36 0=48
37 8=56
38 3=51
39 3=51
40 3=51
41 3=51
42 1=93
43 >=125
44 1=93
45 >=125
46 bytes -----
```

Figura 10 - UDP-TCP Bridge em execução

Na primeira tela, encontram-se algumas informações do tráfego, como data, frequência de envio, duração do tráfego, quantidade de bytes já enviadas e quantidade de bytes que estão sendo enviados no momento. Além disso, pode-se destacar duas outras informações:

- **UDP port:** porta UDP pela qual a ponte está “listening”, ou seja, recebendo. Deve ser igual à porta configurada pelo *tracker* para envio dos pacotes UDP.
- **TCP port:** porta TCP pela qual a ponte está “sending”, ou seja, enviando. Deve ser igual à porta configurada pela aplicação final para recebimento dos pacotes TCP.

Essas duas portas e a frequência de envio podem ter seus valores alterados no arquivo settings.xml, localizado na pasta raiz da aplicação.

Na segunda tela, é possível se observar o caracter que está sendo enviado no momento.

É possível desenvolver uma aplicação própria que faça a mesma função das listadas acima. Porém, por elas serem open-source e multiplataforma e por desenvolverem muito bem suas funções, o desenvolvimento de uma ponte própria acaba se tornando, em muitos casos, desnecessário.

Por ser uma aplicação de simples uso e que recebe qualquer tipo de mensagem via UDP, a UDP-TCP Bridge listada acima será utilizada na VI. Já a VI+ não vai utilizar nenhuma ponte, já que o *tracker* (CCV 1.3) utilizado nessa versão permite o envio de mensagens já em pacotes TCP, não necessitando, portanto, de conversão durante o trajeto.

Sobre a arquitetura de montagem, ambas versões utilizarão a Arquitetura Integrada.

3.3.3. Aplicação Front-End

Também chamada de aplicação final, pois é nela que o processo de interação termina, essa aplicação pode ser definida como a aplicação front-end de um produto interativo, pois é com ela que o usuário vai interagir. Pode estar projetada numa vitrine, num piso, numa parede, numa mesa. O local da projeção vai ser depender do tipo do produto.

Essa aplicação pode ser de qualquer tipo, desde que possua uma interface visual interativa e seja compatível com o recebimento de pacotes TCP. Além disso, deve decodificar as mensagens e utilizar os pontos extraídos dessas mensagens corretamente e em tempo real, sem causar atrasos perceptíveis ao olho humano.

O tipo de aplicação final mais utilizado em produtos interativos, seja ele *multitouch* ou *touchless*, é o Flash, uma interface de programação visual que utiliza ActionScript como linguagem de programação e foi desenvolvida pela Macromedia, empresa comprada pela Adobe, em 2005.

Uma aplicação flash é desenvolvida numa ferramenta fechada de desenvolvimento da Adobe. A versão mais recente é a Adobe Flash CS5. Esta ferramenta possibilita o desenvolvimento de projetos de aplicações Flash. Esses projetos contêm a classe principal, representada por um arquivo de extensão *.fla*, e alguns arquivos de código-fonte adicionais com extensão *.as*, quando necessário. No caso de projetos para produtos interativos, esses arquivos serão classes da linguagem de programação mais utilizada em projetos de aplicações Flash, a *ActionScript*, cuja versão mais recente é a ActionScript 3.0 (AS3).

Na classe principal são definidas as configurações do projeto, além da criação da animação que será gerada e da implementação, em código AS3, das interações dessa animação.

Essas interações ocorrem quando um evento de determinado tipo, descrito em alguma biblioteca presente na API de ActionScript 3.0, altera o estado da aplicação. No caso de aplicações Flash que vão ser interagidas pelo mouse, a biblioteca utilizada é a *MouseEvent*. Um outro exemplo seria as aplicações que recebem mensagens TUIO. Neste caso, a biblioteca utilizada é a *TouchEvent*. Caso seja necessário, o desenvolvedor pode desenvolver o próprio tipo de evento que vai ser usado no projeto.

O recebimento das mensagens é realizado através da conexão a um objeto Socket (*flash.net.Socket*) através do IP da máquina onde o *tracker* está rodando e da porta TCP especificada na ponte.

Depois de compilado o projeto, as animações são criadas em arquivos independentes do projeto, com formato *.swf* (Shockwave Flash File). Esse tipo de animação está presente em vários websites e, para ser visualizada, é necessário a instalação do Adobe Flash Player como plugin do browser. Esse

player se encontra, atualmente, na versão Adobe Flash Player 10. Os arquivos .swf podem ser extraídos para arquivos executáveis (.exe).

Por ser o tipo de aplicação visual mais indicado para produtos interativos, ambas VI e VI+ utilizarão aplicações Flash como front-end do produto.

3.3.4. Pré-requisitos de software

Para o funcionamento ideal dos produtos interativos, alguns pré-requisitos de software devem estar devidamente instalados no sistema.

Conforme mencionado na seção de discussão da câmera, o driver específico para a plataforma e versão do S.O. do computador precisa ser instalado para que o equipamento seja reconhecido pela máquina.

Na maioria dos casos, o driver é acompanhado por um software de configuração e testes da câmera. Nele, é possível modificar, por exemplo, a velocidade de captação da câmera (FPS) e a resolução. Além disso, geralmente também se pode testar se o equipamento realmente está captando imagens nas devidas configurações.

Assim como a câmera, outros equipamentos utilizados podem exigir a instalação de drivers.

Além dos drivers, alguns outros softwares podem ser necessários para o funcionamento dos produtos interativos. Por exemplo, para o uso do *tracker* CCV (ou Tbeta) mencionado na seção “Tracker”, é necessário a instalação do reprodutor de vídeos Quicktime, disponível para download no site da Apple. Um outro exemplo é a biblioteca FLOSC, mencionada na seção anterior. Para o seu funcionamento, assim como o de qualquer outra biblioteca Java, é necessário a instalação do Ambiente de Tempo de Execução Java (JRE), disponível para download no site da Sun, desenvolvedora de Java.

4. DESENVOLVIMENTO DA VI

Após o Estudo de Viabilidade Técnica apresentado no capítulo anterior, foi dado início ao desenvolvimento de um protótipo da versão VI.

A opção por essa versão foi devido à maior simplicidade na implementação tanto do tracker como de uma aplicação Flash para testes, em relação à VI+. Outro fator importante para a escolha da VI é o fato de os equipamentos exigidos por ela serem mais simples e, conseqüentemente, mais baratos e fáceis de serem encontrados.

A VI é um produto destinado a apenas um usuário por vez. Este usuário deverá estar numa posição pré-determinada, a uma distância de aproximadamente 1m em relação à vitrine.

Os equipamentos e acessórios utilizados serão: projetor, câmera, computador, cabos e papel fosco. Enquanto que os softwares serão um *tracker*, uma ponte UDP-TCP e uma aplicação Flash simples.

Neste capítulo, todos esses itens que foram utilizados na implementação e nos testes deste protótipo da VI, serão apresentados e detalhados.

4.1. EQUIPAMENTOS E ACESSÓRIOS

Esta seção vai mostrar, individualmente, cada um dos equipamentos e acessórios utilizados no desenvolvimento da VI. Cada um deles atende às especificações mínimas de cada apresentadas no capítulo anterior.

4.1.1. Projetor

O projetor foi escolhido como equipamento de saída de vídeo. Uma TV LCD, além de mais cara, apresenta algumas dificuldades de manuseio e

instalação devido aos seus tamanho e peso. As principais especificações do projetor utilizado encontram-se a seguir.



Figura 11 - Projetor

- **Marca:** Sony;
- **Modelo:** VPL-ES4;
- **Luminosidade:** 2.200 ANSI LUMENS;
- **Tipo de display:** LCD;
- **Distância focal:** 3 metros;
- **Razão de aspecto:** 4:3;
- **Resolução nativa:** SVGA (800 x 600) ;
- **Vida útil média da lâmpada:** 2.500 horas;
- **Entradas:** VGA.

A tela de projeção utilizada será apresentada na subseção “Acessórios”.

4.1.2. Câmera

A VI fará uso de uma câmera que é considerada pela comunidade NUI Group como a câmera mais usada no desenvolvimento e nos testes de produtos interativos por vários motivos: atende à todas as especificações mínimas de câmeras para produtos interativos com bom desempenho, tem fácil modificação (caso o produto vá utilizar canhões infravermelho) e preço acessível.

Este equipamento foi lançado com o único propósito de ser utilizada como acessório do PlayStation 3 (video-game da Sony). Entretanto, pelos fatores citados acima, entre outros, ela é bastante conhecida e utilizada por desenvolvedores de softwares de visão computacional. É grande o número de drives e informações sobre este equipamento disponíveis na Internet.

Suas principais especificações são:



Figura 12 - Câmera

- **Marca:** Sony;
- **Modelo:** PS3Eye;
- **Resolução:** 320 x 240;
- **FPS:** 30;

- **Plataforma e versão do S.O. (driver):** Windows, Windows 7 x86 (32 bits);
- **Tipo de plug:** USB;
- **Filtros de luz:** Filtro de luz infravermelha que já acompanha o produto (já que não será usado canhão infravermelho nesta versão).

Se o produto interativo incluir o uso de canhões infravermelho, esta câmera precisa ser adaptada. O processo é simples e pode ser resumido pela seguinte sequência:

1. Abertura da câmera;
2. Remoção do filtro de luz infravermelha;
3. Colocação do filtro de luz visível;
4. Fechamento da câmera.

Videos que mostram esse processo completo podem ser encontrados na Internet. Por exemplo, <http://vimeo.com/2939528> e <http://vimeo.com/2975294>.

4.1.3. Canhão Infravermelho

Conforme mencionado no capítulo anterior, a VI não vai utilizar canhões infravermelho, visto que será usada apenas em locais com iluminação constante. Por isso, esse item não será abordado neste capítulo.

4.1.4. Computador

O computador mais utilizado para desenvolvimento e testes da VI é um notebook cujas principais especificações são as seguintes:



Figura 13 - Computador

- **Marca:** Dell;
- **Modelo:** Inspiron 1525;
- **Plataforma:** Windows;
- **Versão do S.O.:** Windows 7 Professional x86 (32 bits);
- **Configurações de Hardware:**
 - **Processador:** Pentium Core 2 Duo T7250 2 GHz;
 - **Memória RAM:** 2 GB DDR2;
 - **Placa de video:** 512 MB, com saída HDMI;
 - **Disco Rígido:** 120 GB;
 - **Portas USB:** 4 portas.

O equipamento, apesar de simples, funcionou perfeitamente e não gerou nenhum tipo de atraso significativo no processo de interação.

4.1.5. Acessórios

Os acessórios utilizados nas montagens realizadas nos testes do protótipo podem ser divididos da seguinte maneira:

- **Cabos:** um cabo VGA para conexão projetor-computador;
- **Tela de projeção:** papel manteiga de aproximadamente 1,20m x 0,80m;
- **Suporte para projetor:** não foi utilizado durante os testes. O projetor foi posto em cima de mesas ou outros móveis semelhantes.
- **Fita adesiva:** fita adesiva transparente, utilizada para fixar a câmera no alto da vitrine e o papel manteiga no vidro.

O cabo VGA já acompanha o projetor, enquanto que o cabo USB já é acoplado à câmera.

4.2. SOFTWARES

Nesta seção serão apresentados os softwares necessários para a versão VI. Dentre eles, alguns já estão disponíveis na Internet como open-source e outros precisaram ser desenvolvidos.

4.2.1. Tracker

O *tracker* utilizado pela versão VI foi desenvolvido em cima do `opencvExample`, apresentado na seção “Tracker” do capítulo anterior. Com isso, pode-se afirmar que é um software de visão computacional desenvolvido em C++, com uso da biblioteca `openFrameworks`, que tem como objeto a detecção de *blobs* referentes à gestos do usuário. O software possui três modos, que serão apresentados mais à frente.

A ferramenta utilizada para o desenvolvimento deste software foi a `Code::Blocks`, um ambiente de desenvolvimento C++ open-source, multiplataforma, simples de ser configurado e muito utilizado por desenvolvedores de *trackers*.

Além de algumas classes da biblioteca `openFrameworks`, e do *addon* `ofxOpenCv` (já apresentado anteriormente), o projeto do *tracker* da VI utiliza alguns *addons* do `openFrameworks` necessários para o controle do envio das mensagens. São eles: `ofxTCPClient`, `ofxTCPServer`, `ofxTCPManager`, `ofxThread` e `ofxUDPManager`. Essas bibliotecas são utilizadas pelas outras classes do projeto, que são as classes padrão do `opencvExample` modificadas e outras classes que precisaram ser implementadas. Todas essas classes serão apresentadas a seguir:

Classes padrão do `opencvExample` adaptadas:

- **main.cpp**: classe que define o tamanho da janela do *tracker* que será exibida quando executado e executa a classe `testApp.cpp`.
- **testApp.cpp**: classe principal de configuração do *tracker*. Nela, são definidos os valores do IP da máquina que executa a aplicação Flash e a porta UDP que vai enviar os pacotes, necessários para a conexão socket realizada na classe `interface.cpp`. Os valores padrão são `localhost` e `3333`, respectivamente, considerando que o *tracker* será executado na mesma máquina da aplicação Flash. Além disso, é definido se vai ser utilizado um vídeo (utilizado em testes) ou a câmera como fonte de imagens para detecção de *blobs*. Esta classe é a responsável por executar as outras, mencionadas abaixo.

Outras classes implementadas:

- **interface.cpp**: classe responsável por criar a conexão socket, informando os valores de IP e porta definidos na classe testApp.cpp. Além disso, tem a função de formatar a mensagem e enviá-la. É responsável pelo terceiro modo do software, o Modo Envio, desenhando o contorno dos *blobs* e apresentando, neste contorno, qual o ponto será utilizado para o processo de interação, ou seja, qual ponto será formatado juntamente com suas coordenadas na mensagem que será enviada via socket. Para a formatação, é utilizado o formato de texto JSON (JavaScript Object Notation), simples de ser escrito e de fácil leitura. O formato da mensagem foi definido como:

- { object : [{ id : number , point : [x , y] }] }, sendo o id referente ao id do *blob* e o x,y referente às coordenadas do ponto selecionado num determinado instante.

Essa mensagem, estando nesse formato é enviada pela rede à aplicação Flash. A conversão realizada durante o caminho não modifica esse formato. Por isso, a mensagem chega intacta à aplicação Flash, que por sua vez já estará desenvolvida para decodificar mensagens neste formato.

- **mapper.cpp**: calcula as configurações de calibração da câmera. Responsável pelo processo da calibração e pelo desenho do primeiro modo do *tracker*, o Modo Calibração, onde é possível realizar esse processo.
- **tracker.cpp**: classe que tem o papel de aplicar os filtros definidos pelo usuário na imagem trackeada, definir o background adequado e, a partir daí, executar os algoritmos de detecção de *blobs*. As configurações dos filtros (limiar, brilho e contraste) e do background são realizadas no segundo modo do software, o Modo Configuração.

A seguir, algumas ilustrações de exemplo dos três tipos de modos do *tracker* em funcionamento, utilizando um video como fonte de imagens para detecção de *blobs*.

Modo Calibração

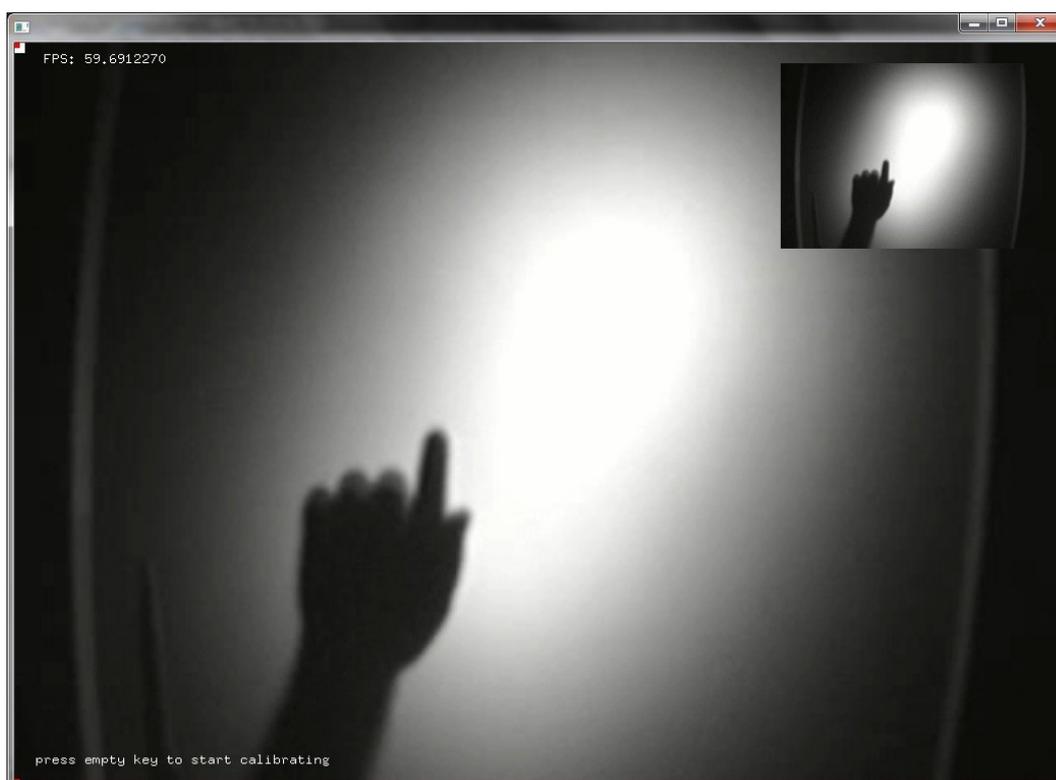


Figura 14 - Modo Calibração

Modo Configuração

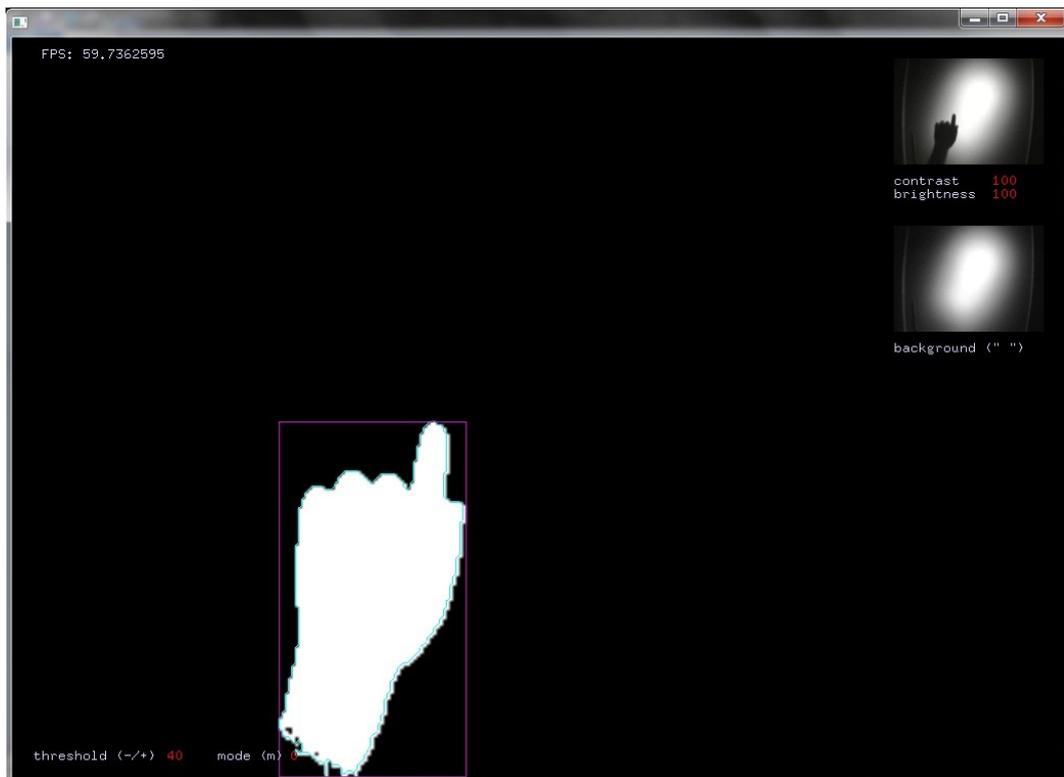


Figura 15 - Modo Configuração

Modo Envio

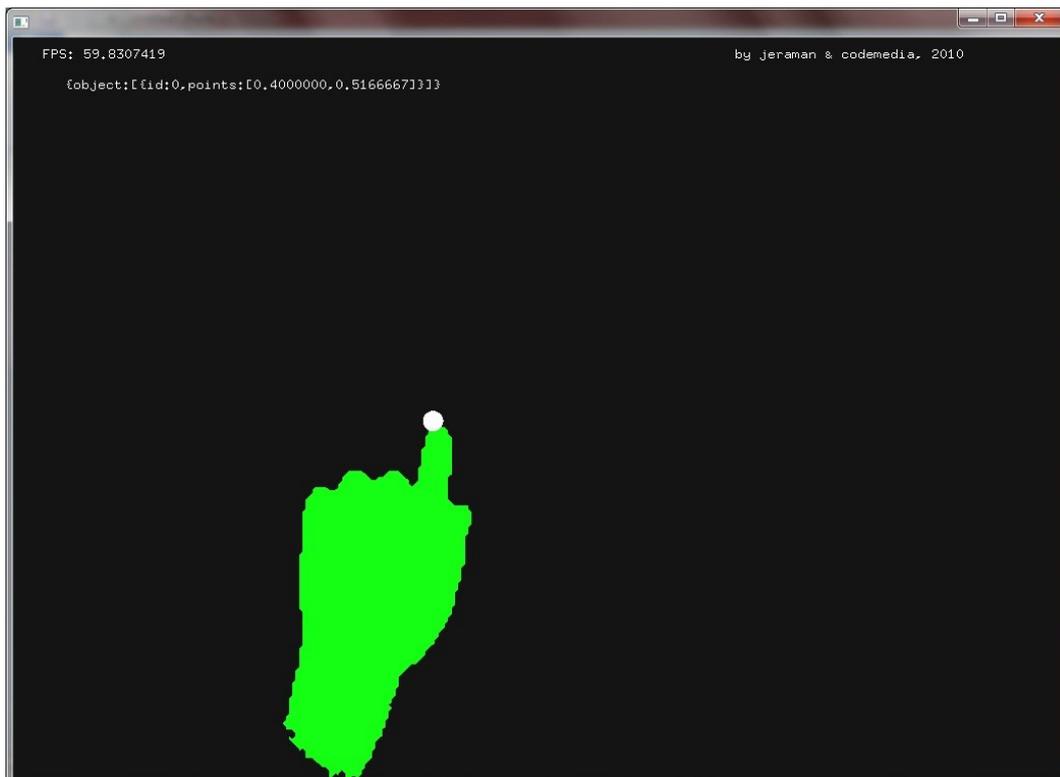


Figura 16 - Modo Envio

Para alternar entre os modos, basta pressionar a tecla “a”. Ao chegar no Modo Envio, o ponto selecionado (representado pela bola branca) e suas coordenadas passam a ser enviados pela rede. Esse envio só acontece caso o *tracker* esteja nesse modo.

4.2.2. Comunicação

Para realizar a conversão de pacotes UDP em pacotes TCP, permitindo uma comunicação em tempo real entre *tracker* e aplicação final, será necessário o uso de uma ponte UDP-TCP apresentada no capítulo anterior, a UDP-TCP Bridge.

Um link para o download deste software pode ser encontrado no seguinte endereço: <http://code.google.com/p/udp-tcp-bridge/>.

4.2.3. Aplicação Flash

O tipo de aplicação que a VI vai utilizar é a aplicação Flash, que vai usar como linguagem de programação o ActionScript 3.0. Para os testes e demonstração, foi desenvolvida uma animação Flash simples, que divide a tela em alguns quadrantes de cores diferentes.

A interação é feita através de gestos com a mão do usuário. Quando ele aponta para algum dos quadrantes, o posicionamento do dedo é calculado pelo *tracker*, e o ponto formado simula o ponteiro de um mouse, fazendo com que o quadrante apontado brilhe intensamente.

Essa aplicação foi dividida em dois módulos. Um, apelidado de API Flash, composto por duas classes AS3, e outro, chamado de Aplicação Flash, composta por uma classe Flash.

Módulo API Flash:

- **Vi.as:**
 - cria um objeto Socket (`flash.net.Socket`);
 - conecta a ele através dos valores de IP e porta passados pelo outro módulo;
 - recebe, através da conexão socket, os pacotes TCP e decodifica as mensagens contidas neles.

- **ContourEvent.as:**
 - classe que estende da classe Event (`flash.events.Event`) e cria o tipo de evento ContourEvent, que será utilizado no outro módulo.

Módulo Aplicação Flash:

- **main.fla:**
 - classe principal que contém o código fonte da aplicação Flash propriamente dita. Nesta classe, o método de conexão socket da classe Vi.as é chamado e são passados os valores do IP e porta que serão utilizados pelo outro módulo para esta conexão socket. Esses valores são, por padrão, localhost e 3000, respectivamente, lembrando que o valor do IP deve ser o IP da máquina que roda o *tracker* e o valor da porta se trata da porta TCP que será criada para recebimento das mensagens. Além disso, essa classe contém o desenho e a definição das cores dos quadrantes. Eles, juntamente com o ponto, são definidos como objetos do tipo MovieClip (flash.display.MovieClip). Esse ponto é, então, convertido em um objeto do tipo ContourEvent, definido no outro módulo. Quando ele entra na área de algum quadrante, o evento é disparado e provoca um brilho intenso neste quadrante.

Quando a aplicação é executada pela primeira vez, é criado um arquivo main.swf no mesmo diretório do arquivo main.fla. O main.swf representa o main.fla compilado e, por isso, não pode ser modificado. A partir dele, é possível extrair o código compilado para diversos formatos, inclusive o executável .exe, o mais utilizado para execução de aplicações Flash em produtos interativos.

4.2.4. Pré-requisitos de software

Para o funcionamento ideal da câmera PS3Eye no computador utilizado nos testes, foi necessário a instalação do driver do equipamento. Atualmente, podem ser encontrados várias versões de drivers disponíveis como open-

source para esta câmera. O driver para a plataforma Windows mais indicado pelos desenvolvedores da comunidade NUI Group foi desenvolvido pela Code Laboratories e é chamado de CL-Eye Driver. Está disponível no site da empresa, através do link: <http://codelaboratories.com/downloads/>.

Em relação ao *tracker*, ele possui apenas um software pré-requisito, o reprodutor de vídeos QuickTime, da Apple, cujo download está disponível no site do fabricante, através do link: <http://www.apple.com/quicktime/download/>.

Os outros softwares utilizados na VI não requerem nenhum tipo de pré-requisito.

5. CONSIDERAÇÕES FINAIS

Neste capítulo, serão apresentadas as principais conclusões e resultados desse Trabalho de Graduação, além das principais contribuições e de possíveis trabalhos futuros na área.

5.1. CONCLUSÃO

Este Trabalho de Graduação apresentou conceitos utilizados em produtos interativos e deu ênfase ao Estudo de Viabilidade Técnica de duas versões de um produto interativo conhecido como Vitrine Interativa, apelidadas de VI e VI+. A segunda engloba todas as funcionalidades da primeira, além de mais algumas.

Primeiramente, foram apresentados os principais tipos de produtos interativos, como a própria Vitrine Interativa, o Piso Interativo e a Mesa Interativa Multitoque. Isso proporcionou um grande aprendizado sobre quais os tipos mais utilizados no mercado e sobre qual o tipo de ambiente que mais aluga esse tipo de produto. Além disso, essa pesquisa mostrou vários desenvolvedores de produtos interativos, que podem vir a ser concorrentes diretos da Codemedia.

Em seguida, foi feito um estudo sobre quais os equipamentos, acessórios e softwares necessários para a construção de uma Vitrine Interativa. Os resultados foram bastante proveitosos, pois, a partir desse estudo, as principais especificações dos equipamentos e as funcionalidades mais importantes dos softwares foram conhecidas.

Como resultado desse estudo, foi desenvolvido um protótipo baseado na versão mais simples, a VI. Esse protótipo será de muita utilidade no desenvolvimento completo das versões VI e VI+.

5.2. PRINCIPAIS CONTRIBUIÇÕES

Como principais contribuições deixadas por este trabalho, pode-se destacar:

- O aprendizado de novas tecnologias, como softwares de visão computacional e aplicações Flash. Essa experiência pode ajudar bastante no desenvolvimento de outros projetos.
- O conhecimento de outro segmento do mercado, o de Mídia Indoor.
- O aprendizado de novos conceitos. Em sua maioria, da área de Realidade Aumentada. Esse aprendizado pode facilitar bastante o desenvolvimento de outras aplicações na área, no futuro.

5.3. TRABALHOS FUTUROS

Após a conclusão deste Trabalho de Graduação, é possível identificar o desenvolvimento dos seguintes trabalhos futuros relacionados ao tema:

- **Desenvolvimento da versão VI:** Com algumas melhorias, o protótipo já desenvolvido vai ter todas as funcionalidades e vai utilizar todos os equipamentos que foram planejados para a VI.
- **Desenvolvimento da versão VI+:** Com o desenvolvimento de algumas funcionalidades extras em cima da versão VI e com a compra de equipamentos mais robustos, a versão mais complexa das propostas estará construída.
- **Desenvolvimento de um Piso Interativo:** Para isso, basta o reposicionamento dos aparelhos utilizados em uma das versões acima.
- **Desenvolvimento de outros produtos interativos:** Com dois tipos de software de visão computacional já desenvolvidos, caso as versões acima sejam finalizadas, é possível a construção de outros tipos de produtos, como mesas multitoques. Para isso, será necessário um estudo em tecnologias de toque e a aquisição de equipamentos sensíveis a toques.

REFERÊNCIAS

AZUMA, Ronald. **Tracking Requirements for Augmented Reality**. Nova Iorque, 1993. Disponível em: <http://www.cs.unc.edu/~azuma/cacm.html>. Acesso em: 22 de junho de 2010.

BARBOSA, Jerônimo. **Hacking PS3 Eye!**. Recife, 2009.

Disponível em: <http://jeraman.wordpress.com/2009/08/17/hacking-ps3eye/>. Acesso em: 02 de julho de 2010.

BEZERIANOS, Anastasia. **Using Alternative Views for Layout, Comparison and Context Switching Tasks in Wall Displays**. Toronto, 2007.

Codemedia. Disponível em: <http://www.codemedia.com.br>

Consumidores são "fiscados" pela vitrine. **Jornal Hoje**. Rio de Janeiro, Rede Globo, 8 de setembro de 2009. Programa de TV.

Disponível em: <http://g1.globo.com/jornalhoje/0,,MUL1296176-16022,00-CONSUMIDORES+SAO+FISGADOS+PELA+VITRINE.html>

FRANÇOIS, Alexandre R.J. **Real-Time Multi-Resolution Blob Tracking**. California, 2004.

GEIBLER, Jörg. **Shuffle throw or take it! Working efficiently with an interactive wall**. Darmstad, 1998

OLIVEIRA, André Luís Belini. **A melhor interação entre o homem e a máquina**. Revista Custo Brasil, ano 4, n. 21. Rio de Janeiro, 2009. Disponível em: <http://www.revistacustobrasil.com.br/21/pdf/04.pdf>. Acesso em: 01 de julho de 2010.

PAPE, Dave & ANSTEY, Josephine. **"Falling Over You" Interactive Wall**. Buffalo, 2003.

PARADISO, Joseph A. Tracking contact and free gesture large interactive. **Communications of the ACM**, Cambridge, vol 46, n.7, 2003.

REKIMOTO, Jun. **SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces**. Toquio, 2002.

SAWCHUK, Alexander A., et al. **From Remote Media Immersion to Distributed Immersive Performance**. Los Angeles, 2003.

ANEXOS

ANEXO 1 – Exemplos reais de produtos interativos

ANEXO 1.1 – Vitrine Interativa

Adobe - <http://gizmodo.com/278038/adobe-pimps-creative-suite-3-with-interactive-wall-in-union-square>

Apple Store - <http://www.youtube.com/watch?v=74AY1hfr7Yg&feature=related>

C&A - <http://www.youtube.com/watch?v=RoY058-1nYs>

Diesel - <http://www.youtube.com/watch?v=6wnX41ZdHUA>

La Vitrine - <http://vimeo.com/7644351>

LG - <http://www.lge.com/br/press-release/article/vitrine-interativa-da-lg-mostra-na-pratica-o-funcionamento-da-linha-de-tv-infinita-com-acesso-internet.jsp>

Multicamera - <http://codelaboratories.com/showcase/>

Orange - <http://www.youtube.com/watch?v=G507OZGaGC8>

Ralph Lauren - <http://about.polo.com/awards/chicago.asp>

ANEXO 1.2 – Piso Interativo

Barclays - http://www.youtube.com/watch?v=51cRL_S8Atk&feature=related

Innovation Expo 2005 - <http://www.youtube.com/watch?v=qo2ilsBoUwM>

MGM Grand - <http://www.youtube.com/watch?v=GH587GTiSXE&feature=related>

Nissan - <http://www.youtube.com/watch?v=ekuoqgC9CvE>

YDreams (Coca-Cola) - http://www.youtube.com/watch?v=Blcap_YRpi0&feature=related

ANEXO 1.3 – Outros produtos interativos

Fantástico - <http://fantastico.globo.com/Jornalismo/FANT/0,,MUL1579071-15605,00.html>

iBar - <http://www.mindstorm.eu.com/products/ibar>

Microsoft Surface - <http://www.microsoft.com/surface/>

SmartTech -

<http://smarttech.com/us/Solutions/Education+Solutions/Products+for+education/Complementary+hardware+products/SMART+Table>