JOÃO LUIZ BERNARDES JÚNIOR

**MODELO ABRANGENTE E RECONHECIMENTO DE GESTOS COM
AS MÃOS LIVRES PARA AMBIENTES 3D**

São Paulo

2010

JOÃO LUIZ BERNARDES JÚNIOR

**MODELO ABRANGENTE E RECONHECIMENTO DE GESTOS COM
AS MÃOS LIVRES PARA AMBIENTES 3D**

**Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia.**

**São Paulo
2010**

JOÃO LUIZ BERNARDES JÚNIOR

**MODELO ABRANGENTE E RECONHECIMENTO DE GESTOS COM
AS MÃOS LIVRES PARA AMBIENTES 3D**

**Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia.**

**Área de Concentração:
Engenharia de Computação e Sistemas
Digitais**

**Orientador: Prof. Livre-Docente
Romero Tori**

**São Paulo
2010**

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 17 de dezembro de 2010.

Assinatura do autor _____

Assinatura do orientador _____

FICHA CATALOGRÁFICA

Bernardes Júnior, João Luiz

**Modelo abrangente e reconhecimento de gestos com as
mãos livres para ambientes 3D / J.L. Bernardes Júnior. -- ed.rev.
-- São Paulo, 2010.
170 p.**

**Tese (Doutorado) - Escola Politécnica da Universidade de
São Paulo. Departamento de Engenharia de Computação e Sis-
temas Digitais.**

**1. Interação homem-máquina 2. Realidade virtual 3. Visão
computacional I. Universidade de São Paulo. Escola Politécnica.
Departamento de Engenharia de Computação e Sistemas
Digitais II. t.**

DEDICATÓRIA

Dedico este trabalho à minha família:
João, Diana, João Marcos e Alair.

AGRADECIMENTOS

Ao Professor Dr. Romero Tori, pela inestimável orientação, além de paciência e incentivo ao longo de todo este trabalho.

A todos os membros do Interlab, atuais e que passaram pelo laboratório ao longo dos anos, que colaboraram direta ou indiretamente com este trabalho, e em particular aos professores Ricardo Nakamura e Maria Alice por sempre se mostrarem mais que dispostos a ajudar, ao Makoto e ao Tsuda pela participação nos projetos exploratórios, ao Bruno pelas posturas e à Mariza por todo seu apoio.

RESUMO

O principal objetivo deste trabalho é possibilitar o reconhecimento de gestos com as mãos livres, para uso em interação em ambientes 3D, permitindo que gestos sejam selecionados, para cada contexto de interação, dentre um grande conjunto de gestos possíveis. Esse grande conjunto deve aumentar a probabilidade de que se possa selecionar gestos já existentes no domínio de cada aplicação ou com associações lógicas claras com as ações que comandam e, assim, facilitar o aprendizado, memorização e uso dos gestos. Estes são requisitos importantes para aplicações em entretenimento e educação, que são os principais alvos deste trabalho. Propõe-se um modelo de gestos que, baseado em uma abordagem linguística, os divide em três componentes: postura e movimento da mão e local onde se inicia. Combinando números pequenos de cada um destes componentes, este modelo permite a definição de dezenas de milhares de gestos, de diferentes tipos. O reconhecimento de gestos assim modelados é implementado por uma máquina de estados finitos com regras explícitas que combina o reconhecimento de cada um de seus componentes. Essa máquina só utiliza a hipótese que os gestos são segmentados no tempo por posturas conhecidas e nenhuma outra relacionada à forma como cada componente é reconhecido, permitindo seu uso com diferentes algoritmos e em diferentes contextos. Enquanto este modelo e esta máquina de estados são as principais contribuições do trabalho, ele inclui também o desenvolvimento de algoritmos simples mas inéditos para reconhecimento de doze movimentos básicos e de uma grande variedade de posturas usando equipamento bastante acessível e pouca preparação. Inclui ainda um *framework* modular para reconhecimento de gestos manuais em geral, que também pode ser aplicado a outros domínios e com outros algoritmos. Além disso, testes realizados com usuários levantam diversas questões relativas a essa forma de interação. Mostram também que o sistema satisfaz os requisitos estabelecidos.

Palavras-chave: Interação Humano-Computador, Reconhecimento de Gestos, Ambientes Virtuais e Aumentados, Visão Computacional.

ABSTRACT

This work's main goal is to make possible the recognition of free hand gestures, for use in interaction in 3D environments, allowing the gestures to be selected, for each interaction context, from a large set of possible gestures. This large set must increase the probability of selecting a gesture which already exists in the application's domain or with clear logic association with the actions they command and, thus, to facilitate the learning, memorization and use of these gestures. These requirements are important to entertainment and education applications, this work's main targets. A gesture model is proposed that, based on a linguistic approach, divides them in three components: hand posture and movement and the location where it starts. Combining small numbers for each of these components, this model allows the definition of tens of thousands of gestures, of different types. The recognition of gestures so modeled is implemented by a finite state machine with explicit rules which combines the recognition of each of its components. This machine only uses the hypothesis that gestures are segmented in time by known posture, and no other related to the way in which each component is recognized, allowing its use with different algorithms and in different contexts. While this model and this finite state machine are this work's main contributions, it also includes the development of simple but novel algorithms for the recognition of twelve basic movements and a large variety of postures requiring highly accessible equipment and little setup. It likewise includes the development of a modular framework for the recognition of hand gestures in general, that may also be applied to other domains and algorithms. Beyond that, tests with users raise several questions about this form of interaction. They also show that the system satisfies the requirements set for it.

Keywords: Human Computer Interaction, Gesture Recognition, Virtual and Augmented Environments, Computer Vision.

LISTA DE FIGURAS

Figura 1 - Articulações da mão <i>Fonte: Pavlovic et al. (1997)</i>	41
Figura 2 - Componentes do modelo de gestos.....	74
Figura 3 - Máquina de estados para o reconhecimento	79
Figura 4 - Contorno	85
Figura 5 - Exemplos de vetores que representam contornos das posturas	86
Figura 6 - Exemplo de rotação do vetor	87
Figura 7 - Árvore para classificação dos movimentos	88
Figura 8 - Processo de reconhecimento.....	96
Figura 9 - Modelo de fluxo de objetos do <i>framework</i>	98
Figura 10 - Gestos com uma das mãos nos testes preliminares.....	103
Figura 11 - Exemplos de imagens usadas nos testes estáticos.....	106
Figura 12 - Alternativa para a postura 7	107
Figura 13 - Calibração.....	110
Figura 14 - Teste interativo para posturas.....	114
Figura 15 - Gestos sem movimento	115
Figura 16 - Teste dos movimentos	116
Figura 17 - Teste da orientação	116
Figura 18 - Gestos bimanuais e local inicial	118
Figura 19 - Manipulação de objetos no <i>desktop</i>	119
Figura 20 - Movimento com tapete no CoH ⁴	123
Figura 21 - Usuário inexperiente na segunda sequência ⁴	126
Figura 22 - Usuário experiente na terceira sequência ⁴	127
Figura 23 - Comandos para o ambiente de iluminação	133
Figura 24 - Metodologia de trabalho.....	156
Figura 25 - Casos de uso iniciais	158

LISTA DE TABELAS

Tabela 1 - Requisitos	22
Tabela 2 - Características para reconhecimento de posturas	56
Tabela 3 - Reconhecimento de posturas.....	60
Tabela 4 - Abordagens para segmentação	65
Tabela 5 - Exemplo de reconhecimento estático de posturas	108
Tabela 6 - Teste da calibração	111
Tabela 7 - Tempos de processamento.....	120

LISTA DE ABREVIATURAS E SIGLAS

3D	Tridimensional ou três dimensões
API	<i>Application Programming Interface</i>
BMP	<i>Bitmap</i> (formato de imagem)
CoH	<i>City of Heroes</i> (jogo utilizado nos testes)
DLL	<i>Dynamic Link Library</i>
DTW	<i>Dynamic Time Warping</i>
EBNF	<i>Extended Backus-Naar Form</i> (sistema de notação)
ES	Estimação por Síntese
HMM	<i>Hidden Markov Model</i>
HSV	<i>Hue-Saturation-Value</i> (sistema de cores)
LED	<i>Light Emitting Diode</i>
MMO	<i>Massively Multiplayer Online</i> (estilo de jogo)
RAM	<i>Random Access Memory</i>
RGB	<i>Red-Green-Blue</i> (sistema de cores)
SPM	Mapas de Probabilidade de Pele (sigla em inglês)
VC	Visão Computacional
XML	<i>eXtensible Markup Language</i>

LISTA DE SÍMBOLOS

F	Vetor de características para cada mão
r	Rotação de uma postura
$\bar{D}$	Vetor normalizado de distâncias que representa o contorno de uma postura
n	Número (de elementos em um vetor, quadros para aceitar uma postura etc.)
G	Posição do centróide da mão
a	Área da mão
(x_i, y_i)	Coordenadas de imagem de um ponto i
$d_j^{i^2}$	Distância ao quadrado do ponto j ao baricentro para a postura i
M_{ab}	Momento de imagem
P	Conjunto dos vetores que representam o contorno das posturas possíveis
p	Número de posturas possíveis
q_i	Erro quadrático médio entre o vetor que representa o contorno da postura da mão e o que representa a postura i

SUMÁRIO

1 INTRODUÇÃO.....	15
1.1 HIPÓTESE, OBJETIVO E REQUISITOS	16
1.2 JUSTIFICATIVA.....	23
1.2.1 Mérito acadêmico	23
1.2.2 Aplicações	24
1.2.3 Importância das aplicações.....	29
1.3 METODOLOGIA	31
1.3.1 Aplicativos específicos.....	32
1.3.2 Desenvolvimento do software.....	35
1.3.3 Avaliação.....	36
1.4 ESTRUTURA DO TRABALHO	37
2 REVISÃO DA LITERATURA SOBRE GESTOS	39
2.1 DEFINIÇÃO E CLASSIFICAÇÃO	40
2.2 MODELAGEM E RECONHECIMENTO	44
2.2.1 Modelagem.....	48
2.2.2 Análise e reconhecimento dos movimentos.....	52
2.2.3 Posturas	54
2.2.3.1 Características	54
2.2.3.2 Reconhecimento	57
2.3 SEGMENTAÇÃO DA IMAGEM	60
2.4 INTERAÇÃO 3D E GESTOS.....	65
3 MODELO DE GESTOS	70
3.1 MODELO	70
3.2 RECONHECIMENTO	77
4 O SISTEMA GESTURES2GO.....	82
4.1 ANÁLISE DA IMAGEM E CÁLCULO DAS CARACTERÍSTICAS	84
4.2 CLASSIFICAÇÃO DOS MOVIMENTOS	87
4.3 RECONHECIMENTO DAS POSTURAS	89
4.4 INTEGRAÇÃO COM OUTROS SISTEMAS.....	91

4.5 SEGMENTAÇÃO.....	92
4.6 <i>FRAMEWORK</i>	96
5 TESTES E RESULTADOS	101
5.1 TESTES PRELIMINARES	102
5.2 TESTES ESTÁTICOS DE RECONHECIMENTO DE POSTURAS.....	105
5.3 TESTES COM INTERAÇÃO	109
5.3.1 Teste da calibração	110
5.3.2 Segmentação e posturas	111
5.3.3 Gestos sem movimento	114
5.3.4 Movimentos	115
5.3.5 Orientação.....	116
5.3.6 Gestos bimanuais e local inicial	117
5.4 MANIPULAÇÃO DE OBJETOS NO <i>DESKTOP</i>	118
5.5 TEMPO DE PROCESSAMENTO	120
5.6 INTEGRAÇÃO E PASSO-A-PASSO COGNITIVO.....	121
5.6.1 CoH.....	122
5.6.2 Ambiente de Iluminação	131
6 CONCLUSÕES.....	134
REFERÊNCIAS	144
APÊNDICE A - HISTÓRICO DO DESENVOLVIMENTO.....	156
A.1 RESULTADOS DOS TESTES PRELIMINARES	160
A.2 RESULTADO DOS TESTES COM O CoH	164
APÊNDICE B - FERRAMENTAS COMUNS.....	169

1 INTRODUÇÃO

O problema que se aborda neste trabalho é possibilitar a interação através de gestos com as mãos livres em ambientes 3D, escolhidos dentre um grande número de gestos possíveis, de forma simples tanto para o usuário final como para o desenvolvedor da interface. Permitir este grande número de gestos possíveis deve ser a principal contribuição do trabalho.

A possibilidade de interagir com computadores usando as próprias mãos e gestos vem há décadas despertando o interesse de pesquisadores e a imaginação do público. Esse interesse pode ser observado inclusive figurando em diversas obras de ficção e existe por diversas razões mas principalmente por ser uma forma de interação mais similar àquela usada no cotidiano para comunicação com outras pessoas e interação física que o uso de interfaces convencionais como as baseadas em *mouse* e teclado (Bowman et al., 2005). Quek et al. (2002) apontam limites dessa similaridade para a maioria das interfaces baseadas em gestos, que em geral os utilizam de maneira discreta ou dissociados da fala, o que raramente ocorre na comunicação entre pessoas. Há evidências, no entanto, que mesmo esse grau menor de similaridade pode trazer vantagens em determinados contextos de interação.

Comparados à interpretação da fala, os gestos tem como principais vantagens (Bowman et al., 2005) não serem afetados pelo nível de ruído do ambiente, utilizarem um canal distinto da principal forma de comunicação entre humanos, a verbal, e não competirem com a resolução de problemas por recursos cognitivos, como ocorre com a fala. A principal desvantagem dos gestos é a maior dificuldade para entrada de informação simbólica.

Atualmente, com maior acesso a dispositivos de captura de imagens a taxas interativas e a capacidades suficientes de processamento, é possível fazer uso de gestos com as mãos livres, sem a necessidade de luvas ou sensores presos ao usuário, mesmo em plataformas domésticas. Com isso, a pesquisa para seu uso em interação, que já ocorre há décadas, têm se intensificado, principalmente com base em imagens em vez de sensores no usuário.

Já existem até mesmo produtos comerciais, como televisores ou computadores portáteis, controlados em parte por gestos simples com uma das

mãos. Na indústria de jogos eletrônicos, em particular, os três maiores fabricantes de consoles têm investido em soluções para interação baseada em gestos mais complexos (Edwards et al., 2004; Allive, 2007; Snider, 2009), com as mãos ou com o corpo, e duas dessas três soluções fazem uso de câmeras. No entanto, a maioria dos sistemas existentes, comercialmente ou em pesquisa, limita-se ao uso de um único tipo de gestos, como somente gestos estáticos ou somente dinâmicos, com um pequeno conjunto de gestos possíveis, o que pode dificultar seu uso. Muitos sistemas utilizam ainda equipamentos menos acessíveis, por exemplo capazes de fornecer mapas de profundidade. Além disso, muitas soluções comerciais são proprietárias e muitas das acadêmicas exigem conhecimentos específicos dos desenvolvedores da interface, como reconhecimento de padrões ou aprendizado de máquina (Wobbrock et al., 2007). É comum que também necessitem de ambiente ou iluminação controlados, posicionamento preciso do equipamento ou processos complexos de calibração. Cada um desses fatores restringe o uso desse tipo de interface e, neste trabalho, a intenção é minimizá-los, o que foi expresso na definição do problema como "de forma simples para o usuário e o desenvolvedor".

1.1 HIPÓTESE, OBJETIVO E REQUISITOS

Como uma solução para o problema de permitir a interação através de gestos escolhidos dentre um grande conjunto de gestos possíveis, inclusive de tipos de gestos diferentes (como estáticos e dinâmicos), neste trabalho adotou-se a seguinte hipótese.

Utilizando-se na definição e no reconhecimento de gestos um modelo que os divide em elementos que os caracterizam, é possível obter um grande conjunto de possíveis gestos a partir da combinação e da classificação de um número bem menor de cada um desses elementos.

Esta hipótese traduz uma abordagem linguística para o reconhecimento de gestos que, de acordo com Derpanis et al. (2004) é muito comum no reconhecimento de fala, mas raramente usada para gestos (o que foi confirmado na revisão da literatura): a partir da combinação de um pequeno número de elementos gera-se um número muito grande de possíveis sinais. Enquanto no caso da fala

esses elementos podem ser os fonemas, o acento com que são pronunciados, sua duração ou sua junção, para gestos pode-se adotar a divisão feita no trabalho revolucionário de Stokoe (1960) para uso em línguas de sinais, a partir da qual gestos deixaram de ser interpretados como unidades indivisíveis. Stokoe lista como componentes de um gesto as posturas que as mãos assumem (definidas pelos ângulos assumidos pelas articulações da mão, não incluindo o pulso e chamada por Stokoe de "dez"), sua orientação, o movimento realizado pelas mãos ("sig") e seu local em relação ao gesticulador ("tab"). Em outro trabalho na área linguística relacionado a línguas de sinais, Liddell e Johnson (1989) notam que as mudanças de postura realizadas durante o movimento raramente tem significado.

Outra vantagem desta abordagem linguística é que, dependendo do subconjunto de gestos que é utilizado em uma determinada interface, ou mesmo em um dos modos de operação dela, é possível utilizar subconjuntos ainda menores para cada um dos componentes dos gestos.

Com base nesta hipótese, o objetivo deste trabalho é projetar, implementar e testar um sistema que permita o reconhecimento de gestos para uso em interação em ambientes 3D, que possibilite a seleção dos gestos dentre um grande conjunto de possibilidades com base nesta abordagem linguística para seu reconhecimento, para uso principalmente em aplicações de educação e entretenimento. Está incluído neste objetivo a análise, durante o projeto do sistema, de pelo menos uma aplicação específica em cada um destes domínios, além de testes utilizando gestos com estes dois aplicativos específicos. Além disso, este sistema de reconhecimento de gestos tem requisitos descritos a seguir, que devem inclusive ser avaliados durante os testes para obter conclusões sobre o cumprimento do objetivo.

O principal requisito e a principal contribuição deste trabalho é justamente, com base no modelo desenvolvido, permitir a escolha dentre um número muito elevado de gestos para uso em interação. Esse requisito é importante pois, para interfaces de aplicações de educação ou entretenimento, a satisfação subjetiva do usuário, a facilidade de seu aprendizado e a de seu uso estão entre os principais objetivos de projeto (Shneiderman et al., 1998), mas gestos constituem uma interface "invisível", suas opções não são mostradas continuamente ao usuário (Bowman et al., 2005). Por isso, para aumentar a facilidade de aprendizado e a satisfação do usuário, cada gesto usado na interface deve possuir uma associação lógica e simples com a ação que comanda e, preferencialmente, deve ser escolhido

dentre gestos já utilizados no domínio da aplicação (Nielsen et al., 2003; Bowman et al., 2005). Por exemplo, um comando para mudar marchas em uma simulação de condução de automóveis poderia ser associado aos gestos de trazer um punho fechado na direção do corpo ou levá-lo para frente, com a mão em posição relativamente baixa, como se a mão estivesse movimentando a alavanca de câmbio. Da mesma forma, um comando de parar ou pausar poderia ser associado a mostrar a palma da mão com o braço estendido para frente, um gesto bastante conhecido. Aplicações de entretenimento ou educação, no entanto, podem versar sobre temas e domínios bastante distintos e não é viável definir um conjunto mínimo de gestos que seja demonstravelmente suficiente para encontrar aqueles com associações lógicas com quaisquer ações possíveis nesses domínios. A alternativa é, portanto, permitir o uso de gestos selecionados dentre um grande conjunto de escolhas possíveis, assumindo que quanto maior o conjunto de escolhas, maior a probabilidade de se encontrar nesse conjunto um gesto adequado para cada ação e em cada domínio e, portanto, de aumentar a facilidade de aprendizado e a satisfação com a interface. Essa necessidade de grande variedade de gestos implica em incluir neste requisito o reconhecimento de diferentes tipos de gestos: os estáticos, definidos somente por uma postura e orientação e possivelmente pelo local, os dinâmicos onde há somente mudança de postura ou de sua orientação mas sem movimento significativo da posição da mão, os dinâmicos onde ocorre esse movimento e, por fim, os paramétricos, onde dados como a posição da mão ou orientação da postura podem ser usados na interação como parâmetros contínuo, por exemplo em tarefas de seleção e manipulação.

Nielsen et al. (2003) afirmam ainda que um sistema de interação por gestos deve ser suficientemente flexível para permitir sua adaptação a usuários de diferentes culturas (e que, portanto, podem ter diferentes associações lógicas entre gestos e ações) e a interfaces diferentes para cada tipo de aplicação, cada aplicação ou mesmo cada modo de operação de uma interface. Considera-se, assim, que esta flexibilidade é um requisito importante do sistema. No caso de jogos baseados em gestos, em particular, a adaptação dos mesmos a cada cultura pode ser parte importante do que se denomina "localização" do jogo, processo ao qual é atribuída grande importância dada a abrangência global desse mercado (Bernal-Merino, 2007). Sistemas que precisam ser retreinados para cada conjunto de possíveis

gestos, por exemplo, podem violar este requisito, a menos que esse treinamento possa ser automatizado e rápido.

O reconhecimento também deve aceitar pequenas variações para cada gesto (Nielsen et al., 2003; Bowman et al., 2005), reduzindo a ocorrência de erros de entrada involuntários cometidos pelo usuário. Exigir que posturas ou movimentos sejam realizados de forma precisa pode tornar o reconhecimento mais simples, mas faz com que a interface seja consideravelmente mais difícil para aprender e usar, exigindo não só que o usuário aprenda os gestos e seus significados mas também que treine como realizá-los com precisão. Pode ainda causar a frustração do usuário e reduzir sua satisfação. Além disso, sistemas que aceitam pouca variação em geral precisam de mais calibração.

Em alguns casos, como aplicações para treinar habilidades motoras específicas ou em alguns tipos de jogo, pode-se argumentar que alguns ou todos dentre estes últimos requisitos (grande número de gestos selecionáveis, vocabulário modificado rapidamente devido a mudanças culturais e aceitar variações) poderiam ou deveriam ser ignorados, de forma que aprender os gestos e como realizá-los com precisão poderia justamente ser o objetivo ou o desafio da aplicação, mas neste trabalho optou-se por adotar estes requisitos, seguindo a recomendação mais geral de Shneiderman et al. (1998) de evitar erros e privilegiar facilidade de aprendizado e de uso e satisfação do usuário.

O sistema deve também permitir o uso doméstico com o menor tempo de preparação ou calibração possível. Hong (2008) realizou uma pesquisa medindo diversos sinais vitais e ondas cerebrais de usuários de diversos jogos e, dentre outras conclusões, descobriu que atualmente os usuários preferem jogos em que sejam introduzidos à ação o mais cedo possível, mesmo enquanto ainda estão aprendendo sobre o sistema e a interface. Acredita-se que esse resultado possa ser extrapolado para outras aplicações tanto no entretenimento quanto na educação. Sendo assim, não se deve exigir condições específicas de iluminação ou fundo da cena (que consomem tempo de preparação), nem treinamento (do sistema ou do usuário) ou procedimentos de calibração que demorem mais que um ou dois minutos no total.

Fornecer *feedback* visual para o usuário também é um requisito bastante importante. Prover alguma forma de feedback é uma das regras de ouro de Shneiderman et al. (1998) para qualquer interface e Bowman et al. (2005)

concordam que esse fator é tão ou mais importante em interfaces 3D que nas convencionais. Além disso, essa necessidade tornou-se bastante clara depois de testes preliminares e em projetos exploratórios relacionados a gestos, realizados em conjunto com alunos de graduação (Tsuda et al., 2007; Tokunaga et al. 2008; Bernardes et al., 2009). Em um desses projetos em particular, Tsuda et al. (2007) discutem em seus resultados e conclusões como a necessidade de fornecer feedback visual para o usuário em interfaces baseadas em câmeras mostrou-se importante durante os testes, visto a dificuldade exibida pelos usuários para mapear seu posicionamento ou movimentos do mundo real para o plano da imagem. Essa necessidade tornou-se um dos requisitos para sua aplicação final, ao ponto que foi considerado necessário reduzir o número de elementos gráficos virtuais presentes na aplicação para que não houvesse oclusão desse feedback (que no caso era fornecido através de uma imagem no fundo da aplicação).

Outro requisito do sistema é permitir o uso não só de gestos realizados com uma das mãos, mas também de gestos bimanuais. Além de aumentar o conjunto de gestos possíveis, o uso de ambas as mãos pode aumentar perceptivelmente a eficiência mesmo em tarefas em que uma delas é usada primariamente. Guiard (1987) relata experimentos em que o uso da mão não-dominante simplesmente para definir o espaço para um trabalho mais fino da mão dominante (por exemplo apoiando-se sobre uma superfície sobre a qual a mão dominante escreve) traz ganhos de eficiência de até 20% quando comparados ao uso somente da mão dominante. Esse é um exemplo de interação bimanual assimétrica, onde as ações que cada mão realiza são diferentes mas coordenadas. Na interação simétrica ambas as mãos realizam ações equivalentes, o que pode ocorrer de forma síncrona (por exemplo levantando pesos) ou assíncrona (por exemplo ao digitar).

Com relação ao requisito de precisão do reconhecimento de gestos, Karam & Schraefel (2006) utilizaram testes do tipo "Mágico de Oz"¹ para determinar até que nível os usuários toleram suas falhas. O estudo mostrou que taxas de erro de até 40% são toleradas antes que os usuários buscassem outra forma de interação, caso não houvesse uma alternativa convencional de interface para ser utilizada, e esse

¹ Testes em que um ser humano interpreta as entradas do usuário, fazendo as vezes do sistema computacional, em geral para permitir testes de técnicas e outros aspectos de interface sem a necessidade ou interferência de uma implementação específica.

número foi adotado como requisito neste trabalho. Se a interface convencional existir, seu uso for conveniente e o usuário estiver familiarizado com interfaces computacionais, no entanto, essa tolerância cai drasticamente e os usuários rapidamente optam pela interface mais familiar se o reconhecimento de gestos se mostra pouco preciso ou robusto. Além disso, caso ocorram erros no reconhecimento, é preferível que sejam falsos negativos, dos quais os usuários podem se recuperar rapidamente simplesmente repetindo o gesto, que falsos positivos, que podem disparar ações indesejadas.

Os tempos de processamento e de resposta são requisitos importantes. É desejável que o tempo de processamento seja o mais curto possível, liberando capacidade de processamento para outras atividades comuns em ambientes 3D, como renderização dos gráficos, simulação física, registro de objetos reais e virtuais, inteligência artificial etc. Este tempo também é limitado pelo tempo de resposta aceitável que, por sua vez, depende da aplicação. O uso de gestos não é adequado, por exemplo, a aplicações que dependam de reflexos rápidos, visto que o próprio ato de fazer um gesto quase sempre é mais demorado do que outras formas de dar comandos, como por exemplo apertar uma tecla. Sistemas multiusuário distribuídos, que devem lidar com latência da rede no processamento das entradas do usuário (como por exemplo jogos massivamente multiusuário, ou MMOs), no entanto, em geral precisam buscar soluções para que atrasos na entrada não penalizem os usuários. Os MMOs desenvolveram diversas estratégias para lidar com essa questão (Fritsch et al., 2005) e que também resultam em não penalizar o usuário pelo tempo extra gasto ao fazer um gesto quando comparado, por exemplo, com apertar um botão. Tais estratégias incluem reduzir o número de comandos necessários por unidade de tempo alternando entre estados distintos em vez de usar vários comandos individuais (por exemplo "entrando ou saindo do modo de ataque" em vez de utilizar um comando para cada ataque) ou aceitando que somente mais um comando seja colocado na fila de eventos enquanto a ação disparada pelo último não tiver terminado (e as ações são configuradas para consumir um certo tempo, por exemplo com animações ou efeitos gráficos). Fritsch et al. (2005) afirmam que no jogo Everquest 2, por exemplo, o uso dessas estratégias com ações consumindo aproximadamente 1000ms (esse valor e intervalos maiores são bastante comuns, inclusive na aplicação de entretenimento usada neste trabalho) permitem interação satisfatória com valores de latência de até 1250ms. Um limite mais prático para

latência para jogos interativos, apontado após a análise do trabalho de diversos autores é de aproximadamente 250ms (Henderson & Bhatti, 2003). Tomando esses valores como referência e novamente extrapolando os dados de jogos para outros ambientes 3D com aplicações semelhantes pode-se concluir que, em um sistema com algum mecanismo para lidar com atrasos nas entradas do usuário como nos exemplos citados acima, resta aproximadamente um segundo para ser dividido entre fazer o gesto e o tempo de resposta do sistema

Por fim, permitir o uso de gestos com juntamente com outros modos de interação e com aplicativos convencionais, projetados sem o uso de gestos em mente, também é um requisito, aumentando a abrangência do sistema e permitindo que o usuário experimente aplicativos já conhecidos de uma forma diferente. Do ponto de vista do desenvolvedor de software, a integração da interação por gestos a outros sistemas deve ser tão simples quanto possível, sem exigir conhecimentos específicos de áreas como reconhecimento de padrões ou aprendizado de máquina.

A Tabela 1 resume esta lista de requisitos.

Tabela 1 - Requisitos

Grande número de gestos possíveis e de diferentes tipos, inclusive bimanuais
Permitir adaptação a diferentes usuários, culturas, interfaces
Aceitar pequenas variações na execução do gesto
Pouca preparação e calibração para uso doméstico
Fornecer <i>feedback</i> visual
Taxas de erro < 40%, preferir falsos negativos
Tempo de resposta < 1s
Permitir integração com outros modos de interação e com aplicativos convencionais

É importante notar que se decidiu limitar o escopo deste trabalho a gestos com as mãos, não com o corpo inteiro, embora essa seja uma extensão interessante para trabalhos futuros.

1.2 JUSTIFICATIVA

Evitar o uso de luvas, sensores ou marcadores presos ao usuário é decisão consequente da intenção de simplificar o uso do sistema. Erol et al. (2005) afirmam que o uso de *datagloves* e sensores (e dos cabos que muitas vezes se fazem necessários) em geral aumentam os custos do sistema, requerem calibração complexa e restringem alguns movimentos, tanto das próprias mãos quanto do usuário. Além disso, as luvas podem não se ajustar ao tamanho das mãos de todos os usuários e podem até causar alergias ou, quando vestidas por mais de uma pessoa, problemas de higiene. Além disso, o uso de câmeras como sensores não fazem uso de dispositivos com partes móveis e minimizam o problema de desgaste de equipamentos. Essas são as principais razões pelas quais, agora que tecnologias acessíveis de captura de imagens e processamento são capazes de fazê-lo, não só interfaces baseadas em gestos mas também outras formas de interação como interfaces multimodais ou atentas (Turk & Kolsch, 2004), cada vez mais optam por reduzir o uso de sensores no usuário em favor de imagens dele.

Embora essas razões expliquem a decisão pelas mãos livres, é útil responder a três outras questões para justificar o trabalho e a escolha deste problema. O problema se adequa a pesquisa acadêmica deste nível ou trata-se de problema bem resolvido ou de pouco interesse? Há aplicações que se beneficiam da interação por gestos? E, por fim, qual a importância, comercial ou acadêmica, dessas aplicações?

1.2.1 Mérito acadêmico

O reconhecimento de gestos manuais, com diferentes tipos de sensores, e seu uso em interação são temas que têm gerado (Mitra, 2007; Jaimes e Sebe, 2007; Erol et al. 2005) e continuam a gerar grande interesse e volume de pesquisa. Com os avanços tecnológicos já discutidos, o foco dessas pesquisas tem se afastado do uso de luvas com sensores, caso em que determinar a postura e posição das mãos são problemas bem resolvidos (ainda que o reconhecimento do gesto em si e,

principalmente, seu uso para interação não o sejam), em favor do uso de câmeras como sensores.

Tratam-se, no entanto, de problemas complexos. Gestos mostram grande variação mesmo quando realizados por um mesmo usuário, o que só se intensifica quando se trata de usuários diferentes, inclusive com características físicas distintas; as mãos tem um grande número de graus de liberdade e ainda a capacidade de movimentar-se e mover os dedos com grande velocidade, mesmo ao realizar ações simples e quando se usam câmeras como sensores, podem existir muitas configurações possíveis da mão que geram imagens com grande similaridade (Erol et al., 2005). Além disso, há problemas comuns do uso de imagens, como variações na iluminação ou nas câmeras, ruído etc. Para uso em interação, gestos se comportam como uma "interface invisível", ou seja, ao contrário do que ocorre, por exemplo, em menus ou caixas de diálogo, as ações que os usuários podem tomar com gestos não são mostradas a eles constantemente e deve-se utilizar gestos que tenham associações lógicas com essas ações, mas essas associações podem variar com base em uma série de fatores, inclusive culturais e pessoais (Nielsen et al., 2003). Por tudo isso, a revisão da literatura mostra que, embora haja um grande interesse e volume de trabalho sobre esse problema, envolvendo diversas áreas de estudo, não se trata de um problema bem resolvido e um grande número de soluções coexistem e estão em constante evolução, em domínios, aplicações e com limitações distintas, como restrições aos tipos de gestos que se pode usar, ao conjunto de gestos possíveis ou às condições de uso. Sob o ponto de vista acadêmico, portanto, pode-se considerar o problema adequado ao desenvolvimento de uma tese, principalmente visando a contribuição de permitir a escolha de gestos dentre um grande conjunto de possibilidades.

1.2.2 Aplicações

Mas quais são os fatores práticos que motivam esse interesse acadêmico, em particular na área de interação, e que tipos de aplicação se beneficiam desse tipo de interface?

Bowman et al. (2005) afirma que, em geral, o uso de gestos (mesmo que não sejam familiares), faz uso de habilidades motoras bem desenvolvidas desde as fases iniciais da vida e usadas com menor esforço cognitivo que, por exemplo, *mouse* e teclado. Mas os mesmos autores afirmam que existe um grande número de aplicações, em geral uni- ou bidimensionais como manipulação de textos, sons e imagens, para as quais as interfaces convencionais estão muito bem adequadas e, ainda que existam interfaces alternativas para essas tarefas (baseadas em interpretação da fala, gestos ou rastreamento da direção do olhar, entre outras), seu uso se mostra menos eficiente que as convencionais e em geral é restrito a situações específicas em que é menos prático usá-las, como a presença de certas deficiências do usuário, restrições do ambiente ou do equipamento disponível.

Algumas dessas situações são suficientemente (e cada vez mais) comuns para justificar o uso de interfaces alternativas mesmo em aplicações convencionais. Um exemplo é o uso de um sistema fixo durante uma atividade que se beneficie de movimento ou em que o foco da atenção do usuário não seja necessariamente o sistema e sua interface. Palestras, demonstrações ou performances são exemplos em que ambas as condições são verdadeiras. Outra situação cada vez mais comum é justamente o uso de plataformas móveis e de tamanho reduzido, que tornam o uso de teclado e apontadores menos convenientes. Por fim, a relação entre usuários e dispositivos computacionais evoluiu de muitos-para-um, passando pela computação pessoal um-para-um, para uma situação em que, atualmente, um usuário frequentemente pode interagir simultaneamente com diversos dispositivos com capacidade de processamento, como um computador portátil, um sistema de som e um celular (Turk & Kolsch, 2004). Nesse caso, despendar a atenção exigida por uma interface convencional para cada um desses dispositivos torna-se menos prático, por exemplo, que o uso de gestos, como demonstrado pelos experimentos de Karam & Schraefel (2005).

Mas além desses casos específicos, em que aplicações bem adequadas às interfaces convencionais se beneficiam do uso de outras alternativas, Bowman et al. (2005) afirmam que há toda uma classe de interfaces para as quais as técnicas e dispositivos convencionais se mostram menos adequados: as interfaces 3D, ou seja,

aquelas em que tarefas² são realizadas diretamente num contexto espacial.

Por possuírem menos graus de liberdade, dispositivos como mouse e teclado, quando usados para realizar essas tarefas em três dimensões, tornam necessário o uso de técnicas² complexas de interação, que em geral demandam esforço para ser aprendidas e resultam em interação mais difícil e menos eficiente que o uso de outras técnicas, dentre as quais estão os gestos manuais. Um exemplo comum dessas diferentes complexidades para realizar tarefas em 3D com dispositivos convencionais é a rotação de um objeto, para visualização do mesmo de diferentes pontos de vista ou para colocá-lo em uma posição específica. Essa tarefa é realizada de forma simples e corriqueira com objetos físicos, usando as mãos, mas em geral é bastante complexa em ambientes virtuais 3D usando mouse e teclado.

Além disso, da mesma forma que o uso das mãos é uma interface de ação (uma forma com que o usuário se comunica com o sistema) mais similar a forma com que se comunica e interage com pessoas e objetos no cotidiano, interfaces 3D também constituem uma interface de percepção (aquelas em que a comunicação se dá na direção inversa, do sistema para o usuário) mais próxima da forma como se percebe o mundo, outro fator que torna adequado o uso de gestos em interfaces 3D.

Devido a essas características da interação em ambientes 3D, virtuais ou aumentados, que tornam mais atrativo o uso de gestos, e porque, como alertam Nielsen et al. (2003), uma das principais dificuldades no desenvolvimento de sistemas de interface com gestos é que procuram desenvolver uma solução geral, enquanto o sistema deve ser projetado para um domínio mais específico, considera-se justificado restringir o domínio de interesse deste trabalho aos ambientes 3D. Ainda que seja não só possível como mesmo desejável que o sistema resultante possa ser usado em outros tipos de aplicação, como por exemplo nas convencionais

² **Tarefa** é qualquer ação que o usuário deseje e possa realizar através da interface ou, mais formalmente, é a entrada de uma unidade de informação pelo usuário. Essa unidade de informação pode ser decomposta em unidades menores, ou sub-tarefas. Tarefas podem ser definidas de forma independente de sua implementação. Uma implementação particular de uma tarefa, a forma que se permite que o usuário a realize em uma interface, é chamada **técnica**. Da mesma forma que uma tarefa é dividida em sub-tarefas, uma técnica constitui-se de componentes de software e hardware. Os componentes de hardware são os dispositivos de entrada e saída do sistema, enquanto os de software mapeiam a informação dos dispositivos de entrada como alguma ação no sistema e sua saída para alguma forma que possa ser exibida pelos dispositivos de saída. Uma forma de projetar diferentes técnicas para a realização de uma dada tarefa é decompô-la em suas sub-tarefas e associar a cada uma um componente de técnica. Da mesma forma, um componente de técnica pode ser utilizado em várias tarefas diferentes. (Foley et al., 1996).

em situações adequadas, a pesquisa, os requisitos, o desenvolvimento e os testes devem ser baseados nos ambientes 3D.

De acordo com Bowman et al. (2005), as tarefas mais comuns em interfaces 3D são navegação, seleção e manipulação de objetos, comandos para controle de sistema e entrada simbólica de dados e é possível utilizar gestos para realizar todas essas tarefas, mas eles se adequam muito bem ao envio de comandos, e em menor grau à manipulação de objetos, à navegação e a alguns casos específicos de entrada simbólica.

A navegação consiste na movimentação da representação do usuário ou de seu ponto de vista dentro do ambiente 3D e pode ser dividida em locomoção (o movimento em si) e na busca e determinação de caminhos (o componente cognitivo da navegação, que pode ser auxiliado pela interface). É comum que o usuário não possa olhar para cima ou para baixo nem que possa mover-se fora de um plano (por vezes deformado com relevo) que representa o chão. Nesses casos, a navegação torna-se bem mais simples mesmo com dispositivos convencionais, visto que pode ser reduzida a uma tarefa bidimensional, de mover-se sobre esse plano.

Seleção de objetos é auto-explicativa, já a sua manipulação é definida por Bowman et al. (2005) como a realização no espaço de transformações de corpo rígido sobre o objeto, o que é consistente com definições anteriores bidimensionais e com o que se faz com objetos reais com as mãos na maior parte dos casos. Por essa definição, mudanças na forma de objetos não-articulados ou mudanças de cor não são classificadas como manipulação, mas são exemplos de ações de controle.

O controle de sistema pode ser confundido com o envio de comandos e é uma forma crítica de acessar sua funcionalidade. Essas funcionalidades podem incluir, por exemplo, o salvamento ou carregamento de arquivos, o início, pausa ou fim de processos, desfazer a última ação, dentre outros. Incluem ainda a mudança do modo de interação ou mudanças de estado do sistema ou de um objeto. O que diferencia um comando das tarefas descritas anteriormente é que um comando simplesmente especifica o que deve ser feito, mas não como. Já em tarefas como navegação, seleção e manipulação, o usuário em geral especifica de forma contínua o que e como deve ser feito. Apesar disso, comandos podem ser utilizados para a realização de outras tarefas. Por exemplo, definindo um comando para selecionar o próximo objeto de uma lista e outro para o anterior, um usuário pode selecionar objetos percorrendo essa lista. Definindo um comando para começar ou parar de se

mover para frente e mais dois, para virar para a esquerda ou direita, realiza-se uma navegação simples.

Por fim, a entrada simbólica consiste na entrada de texto e números, usada por exemplo para dar nomes a arquivos ou objetos, especificar quantidades (principalmente grandes números ou quantidades precisas) ou para comunicação entre usuários. De acordo com Bowman et al. (2005) essa é a tarefa presente com menos frequência em ambientes 3D, não por ser menos necessária, mas sim porque técnicas para realizá-la de forma eficiente são de difícil projeto e implementação se o uso de um teclado não for prático. O reconhecimento de fala ou o uso de teclados virtuais são exemplos de técnicas que podem substituir o teclado mas que são de difícil implementação e nem sempre são aplicáveis. Gestos podem ser usados para a entrada de números pequenos ou de um pequeno vocabulário de palavras, mas para vocabulários maiores seu uso se torna bastante complexo.

Mesmo as interfaces 3D ainda podem ser consideradas como uma classe de aplicação muito ampla, abrangendo um grande número de aplicações com características, requisitos e abordando conhecimentos bastante distintos. Por isso, decidiu-se por restringir ainda mais o domínio, a aplicações de entretenimento, em particular jogos eletrônicos, e educação.

A escolha destes dois tipos de aplicação é conveniente pois, de acordo com Shneiderman et al. (1998), eles tem requisitos semelhantes no que se refere à interface. Para ambos, dentre as qualidades da interface, devem ser privilegiadas a facilidade de aprender e lembrar seu uso, tolerância a erros, baixo custo e satisfação subjetiva. Além disso, ambos se incluem em linhas de pesquisa do Interlab e tanto o autor quanto este laboratório, onde se desenvolve esta pesquisa, têm experiência prévia com esses dois tipos de aplicação em ambientes 3D.

Como já foi discutido, nas principais plataformas de jogos tem havido investimento em alternativas de interação, inclusive gestos, o que concorda com a análise feita até o momento visto que, nessas plataformas, grande parte dos jogos se dá em ambientes 3D. O uso de mouse, teclado ou *gamepads* nesses ambientes, e das técnicas complexas de interação que tornam necessárias, são vistos como um dos principais obstáculos para o uso desses sistemas por um número maior de usuários e o uso de alternativas como gestos para interação é considerado uma das principais formas de aumentar esse número, além de melhorar a interação para os usuários já existentes (Kane, 2005). Além de permitir o uso de técnicas de interação

menos complexas, Payne et al. (2007) mostram que interfaces baseadas em gestos podem aumentar a satisfação subjetiva dos usuários de jogos.

Enquanto nos jogos os ambientes 3D, com sua adequação a técnicas não-convencionais de interação, são largamente utilizados, em educação seu uso ainda é restrito, principalmente devido a seu custo, mas essa é considerada uma das principais aplicações de ambientes virtuais (Youngblut, 1998) e aumentados (Azuma, 1997), pois permitem, de uma forma mais próxima àquela com que se experimenta a realidade, visualizar e interagir com conceitos abstratos, objetos e eventos em escalas atômicas ou interplanetárias ou que não seriam observáveis devido a tempo, distância ou fatores de segurança. Além disso, em educação à distância, podem ser utilizados para aproximar os participantes (Tori, 2010). Nesse aspecto, tanto em educação como em entretenimento, e não somente em aplicações 3D, o uso de gestos pode ainda mostrar outra vantagem sobre técnicas convencionais. Gestos são mais visíveis que o uso de teclas ou botões para uma platéia ou para outros participantes do ambiente que possam ver o usuário (por exemplo representado por um vídeo-avator), mostrando de forma mais clara o mapeamento entre suas ações e mudanças no sistema ou no ambiente. Isso facilita inclusive o aprendizado do uso do sistema por outros usuários.

Pode-se afirmar, portanto, que existem diversas situações em que qualquer aplicação pode se beneficiar de interfaces baseadas em gestos mas que uma das maiores dificuldades desse tipo de interface é procurar uma solução geral, que se aplique a diversos domínios, dadas as características e requisitos diferentes dos mesmos. E que no caso de interfaces 3D, devido ao menor número de graus de liberdade das interfaces convencionais, o uso de gestos mostra-se ainda mais adequado e isso se estende ao uso de ambientes 3D em jogos e em educação, o que justifica a escolha destas aplicações para este trabalho.

1.2.3 Importância das aplicações

Ambientes 3D são utilizados em diversas aplicações de grande importância econômica e para pesquisa (Bowman et al., 2005), dentre as quais pode-se citar educação e treinamento, entretenimento, projetos e cirurgias auxiliados por

computador, tratamentos médicos, visualização de dados científicos cada vez mais complexos e volumosos, computação móvel e ubíqua. Algumas dessas áreas são de âmbito mais recente, mas todas tem ganhado importância e abrangência com o aumento de capacidade e redução dos custos de sistemas computacionais. Atualmente, pode-se afirmar que um obstáculo ao crescimento da utilização de ambientes 3D é justamente a inadequação das interfaces e dispositivos convencionais a esses ambientes.

Dentre essas aplicações, a importância da educação é auto-evidente. Já o entretenimento, em particular os jogos eletrônicos, também se mostram importantes tanto enquanto atividade econômica quanto para pesquisa. Economicamente, o faturamento da indústria de jogos (incluindo vendas de hardware, software e serviços) somente nos Estados Unidos em 2008 ultrapassou US\$42.5 bilhões (IBISWorld, 2009), ultrapassando diversas outras atividades econômicas, como por exemplo o cinema.

Do ponto de vista de pesquisa, jogos eletrônicos são uma aplicação de grande interesse para o estudo de interação. Swartout & van Lent (2003) discutem como as formas de projetar a interação normalmente usadas em jogos podem ser aplicadas a diversas outras áreas, inclusive à educação. Essas técnicas, baseadas na experiência do usuário, resultam em aplicações capazes de prender sua atenção por longos períodos e que são aprendidas, mesmo em casos que retratam sistemas e interface complexos, de forma rápida e agradável sem a necessidade de cursos ou materiais suplementares.

No caso de interfaces inovadoras, que tem relação mais estreita com este trabalho, Starner et al. (2004), afirmam que jogos são uma plataforma ideal para testá-las e popularizá-las. Eles discutem como sua natureza lúdica aumenta a disposição dos usuários a testar protótipos e explorar diversos aspectos dessas interfaces mesmo antes de terem atingido um estado mais desenvolvido, bem acabado ou agradável. Observam também que, ao contrário de outros sistemas, jogos podem ser projetados especificamente para destacar ou esconder aspectos específicos da interface, permitindo a avaliação mais independente desses aspectos.

De fato existem diversos exemplos de pesquisa de novas interfaces utilizando jogos, em particular na área de Realidade Aumentada como relatado por Bernardes et al. (2008). A própria indústria de jogos eletrônicos introduziu ou popularizou

diversas interfaces de uso previamente restrito. Desde *hardware* de baixo custo e algoritmos para renderização 3D em tempo real³, do joystick a *gamepads* cada vez mais complexos e controladores na forma de instrumentos musicais, de *datagloves* a pistolas que na verdade são dispositivos para apontar e até mesmo dispositivos hápticos (Novint, 2009), existem diversos exemplos dessa popularização de interfaces. Atualmente, há até profissionais que são encorajados a fazer uso de jogo para melhorar habilidades com interfaces usadas em seus trabalhos (Dobnik, 2004).

1.3 METODOLOGIA

Para cumprir o objetivo deste trabalho com todos os seus requisitos, adota-se uma metodologia que combina pesquisa acadêmica com projeto centrado em usuário (Norman & Draper, 1986) e o processo unificado de desenvolvimento de *software* (Jacobson et al., 1999).

Ao longo de todo o trabalho, principalmente e mais intensamente durante sua fase inicial (que precede o desenvolvimento do *software*), faz parte da metodologia realizar uma constante pesquisa e revisão crítica da vasta literatura sobre reconhecimento de gestos e seu uso em interação, dentre outros temas relevantes ao trabalho. Além de delimitar o estado da arte desse tipo de sistema e identificar possíveis contribuições, essa pesquisa tem entre suas principais funções auxiliar em diversas decisões de projeto (como a escolha inicial de um ou outro algoritmo de classificação ou conjunto de características relevantes à análise da imagem), tomadas com base em relatos na literatura, poupando o tempo e o esforço necessários para implementar, testar e comparar diversas alternativas para cada funcionalidade necessária ao sistema, tempo e esforço esses que poderiam inclusive inviabilizar o trabalho, ao menos em sua forma atual. Ainda durante essa pesquisa na fase inicial, a metodologia inclui a participação de forma mais indireta, orientando alunos de graduação, de projetos distintos, de menor escopo e com requisitos diferentes, para explorar aspectos específicos do uso de gestos em interação.

³ Atualização de 10Hz ou mais pode ser considerada como tempo real (Kang, Lee & Jung, 2004).

Seguindo a metodologia de projeto centrado em usuário, o primeiro passo é especificar o contexto de uso, ou seja, que tipo de público fará uso do sistema, para que e em que condições. Uma vez estabelecida a aplicação em ambientes 3D para educação e entretenimento e uso doméstico, um grupo é formado por usuários com diferentes perfis pertinentes (como jogadores, desenvolvedores, professores e alunos) e é utilizado em testes e discussões preliminares cujos resultados auxiliam na determinação dos requisitos e de outros aspectos do sistema. Já nessa fase preliminar são escolhidos dois aplicativos específicos cuja principal função é ilustrar de forma mais tangível o uso de gestos em ambientes 3D e facilitar essa análise inicial, além dos testes. O desenvolvimento é iterativo (como também ocorre no processo unificado) e cada interação deve resultar em um protótipo que, por sua vez, deve ser usado para testes, não só funcionais como também com um subconjunto desse grupo de usuários, fornecendo dados para as próximas interações. O processo unificado é utilizado também para determinar como o desenvolvimento e os testes são documentados.

Utiliza-se ainda nessa metodologia uma abordagem do específico para o geral, escolhendo aplicativos específicos para serem analisados e testados durante o trabalho para, a partir daí, obter conclusões mais gerais. Essa abordagem é útil pois, como já foi discutido, tanto aplicações de entretenimento como de educação podem versar sobre temas e domínios bastante distintos de forma que se torna inviável comprovar que o sistema desenvolvido adequa-se a todas as aplicações desse tipo. Ao invés disso, a metodologia prevê a realização de testes para mostrar a adequação do sistema a uma aplicação específica de entretenimento e uma de educação e, com base nesses e em testes funcionais adicionais, a discussão e obtenção de conclusões mais gerais, sobre a adequação do sistema a outras aplicações do mesmo tipo.

1.3.1 Aplicativos específicos

Para análise durante todo o trabalho e para realizar testes com o sistema desenvolvido, utiliza-se como aplicação de entretenimento um jogo do tipo MMO. Como já foi discutido, esse gênero conta com diversas características que se

adequam ao uso de gestos. São jogos em que, além da navegação no ambiente 3D, o envio de comandos é uma das tarefas principais e mais frequentes, tarefa para a qual o uso de gestos está entre as técnicas mais indicadas de acordo com Bowman et al. (2005) e Nielsen et al. (2003). Além disso, para reduzir o efeito da latência de rede sobre a jogabilidade, a maior parte desses jogos conta com mecanismos que limitam a quantidade de entradas de usuário por unidade de tempo, o que também tem como consequência a não penalização do usuário pelo tempo que consome para realizar um gesto, ainda que seja maior do que, por exemplo, pressionar uma tecla ou botão.

Nesse gênero, optou-se por utilizar um jogo comercial já existente, por três razões. Uma delas é justamente verificar a capacidade de integrar o sistema com aplicações já existentes sem a necessidade de modificá-las de forma alguma, o que aumenta a possibilidade de uso do sistema e é um de seus requisitos. Além disso, testes com um aplicativo relativamente complexo, já existente e sem modificações permitem explorar o uso de gestos com diversas tarefas (como controle do sistema, comandos de jogo e seleção de objetos) que não foram otimizadas especificamente para o uso dessa técnica, condição que poderia mascarar positivamente os resultados. Por fim, esta escolha permite que o foco do trabalho permaneça no reconhecimento de gestos e não no desenvolvimento do aplicativo para testes.

O jogo selecionado chama-se "*City of Heroes*⁴" (CoH). Nesse jogo, por padrão, a interação se dá com o mouse e, principalmente, com o teclado. As setas do teclado são usadas para navegação no ambiente 3D e a tecla de espaço para pulos. Objetos e oponentes controlados pelo computador ou por outros jogadores podem ser selecionados utilizando um clique do mouse ou uma tecla (por exemplo a tecla tab para selecionar o oponente mais próximo, técnica que em geral é preferida por ser mais rápida) e então uma grande variedade de ações podem ser realizadas sobre a entidade selecionada, através de comandos associados a uma tecla, uma combinação de teclas ou a um ícone na tela que pode ser clicado (novamente, o uso do teclado é preferido). Alguns objetos, como portas, não podem ser selecionados e só permitem a interação através de cliques do mouse (City of Heroes, 2010).

⁴ *City of Heroes*® é marca registrada da *NCSoft Corporation* e é usada nesse trabalho, assim como imagens mostrando conteúdo do jogo, sem fins lucrativos e com permissão, de acordo com o item 6b de seu *End User License Agreement*.

Esse jogo é líder no mercado de MMOs com seu tema e conta com aproximadamente 125 mil assinantes (NCSoft, 2008), com o valor mensal da assinatura de US\$15,00 nos EUA, e permanece no mercado (com investimentos contínuos em desenvolvimento, atualizações, suporte, manutenção etc.) mesmo seis anos após seu lançamento, duração incomum para MMOs baseados em assinatura. Além disso, uma das principais vantagens de sua escolha é a facilidade, para um grande número dos personagens com os quais se pode jogar, de se encontrar um conjunto de gestos adequado, já existentes no domínio do jogo e, portanto, mais fáceis de aprender e memorizar. Enquanto um grande número de MMOs enfatiza o uso de armas pelos personagens, os mesmos representam super heróis no CoH e grande parte deles não utiliza arma alguma. Ao invés disso, o uso de cada um de seus "super poderes", ou ações que podem realizar durante o jogo, é claramente identificável a partir de um gesto feito pela representação 3D do personagem e associado àquela ação. Com raras exceções, um único gesto é associado a cada ação. Ações diferentes, no entanto, podem estar associadas a um mesmo gesto, o que é uma desvantagem. Outra desvantagem é que, devido a limitações técnicas das animações realizadas no jogo, os dedos dos personagens não são animados e, por isso, seus gestos se limitam ao uso de duas posturas, o punho fechado e a mão aberta com os dedos unidos. Mesmo nesse caso, outras posturas são úteis para tarefas como controle do sistema ou em casos em que duas ações compartilham um mesmo gesto no jogo.

O segundo aplicativo a ser usado para análise e testes, na área de educação, é um ambiente 3D interativo e colaborativo para demonstrar os efeitos de diferentes fontes de luz, como pontual ou direcional e direta ou indireta, e de diferentes algoritmos de iluminação, como Gouraud ou Phong, na renderização de imagens em tempo real. O aplicativo inclui uma representação de um usuário (em geral o professor) como um vídeo-avatar (Tokunaga et al., 2009), ou seja, permitir que sua imagem, capturada com câmeras de vídeo, seja renderizada integrada a esse ambiente, possibilitando, dentre outras vantagens, uma comunicação consideravelmente mais rica, fazendo pleno uso de gestos e expressões faciais. O uso de gestos em conjunto com o vídeo-avatar traz a vantagem de permitir que os gestos sejam usados para envio de comandos e controle do sistema ao mesmo tempo em que pode ser visto pelos outros usuários do ambiente e servir como ferramenta de comunicação, permitindo que esses usuários associem os gestos e

seus significados e suas consequências sobre o sistema e sobre o ambiente, o que facilita inclusive o aprendizado do uso do aplicativo. Nesse ambiente os gestos são usados principalmente como comandos para ligar ou desligar diversas fontes de luz, para abrir ou fechar janelas (que são fontes de luz direcional) e para mudar o algoritmo de iluminação usado. Esse aplicativo se insere no projeto TIDIA-Ae (TIDIA, 2010) como ação de aprendizado.

Outra possibilidade de aplicação na área de educação é a integração do sistema com o VIDA, um atlas de anatomia virtual já existente que utiliza realidade aumentada, estereoscopia e gestos (atualmente, em lugar de gestos específicos, o movimento das mãos obtido através do fluxo óptico é utilizado para realizar algumas tarefas no VIDA). Para esse fim, já foi realizado um estudo com usuários para determinar que gestos melhor se adequam a essa aplicação, como descrito por Tori et al. (2009). Esse estudo é utilizado neste trabalho para análise dos requisitos do sistema, indicando, por exemplo, a utilidade da orientação das posturas como um parâmetro. A integração de fato e os testes com essa aplicação, no entanto, devem figurar somente entre os trabalhos futuros.

1.3.2 Desenvolvimento do software

Para o desenvolvimento do *software* necessário neste trabalho, adota-se como metodologia uma customização do Processo Unificado (Jacobson et al., 1999) para um projeto de pesquisa de pequena escala com um único desenvolvedor e baixo risco financeiro, descrita por Bernardes (2004). De acordo com Smith (2002), nesse caso as fases de concepção e elaboração podem ser bastante reduzidas a uma rápida verificação da viabilidade e ao traçado de um cronograma de mais alto nível para as atividades do trabalho, que é refinado em cada etapa. A fase de transição também só se inicia após o término da implementação. Este trabalho consiste, assim, quase que unicamente da fase de construção. Dentro dessa fase, de acordo com Jacobson et al. (1999) a etapa de análise também pode ser absorvida pelas etapas de requerimento, projeto e implementação em projetos de pequeno porte e, portanto, também não é realizada explicitamente. A etapa de projeto, no entanto, é bastante estendida no caso de um projeto de pesquisa, pois

inclui o desenvolvimento de, e experimentação com, protótipos. Uma última simplificação do processo é que, embora tenha sido dividido em diversas interações, os artefatos de cada uma não são mantidos separadamente mas sim integrados, de forma que ao fim do projeto há somente um conjunto de artefatos que reflete o sistema final.

1.3.3 Avaliação

Antes mesmo do início do desenvolvimento, foram realizados alguns testes preliminares com possíveis usuários, principalmente para verificar seu comportamento ao utilizar gestos nas condições específicas para o uso deste sistema, ou seja, utilizado principalmente para transmitir comandos sem o auxílio da fala, utilizando uma câmera cuja posição é conhecida pelo usuário e sem a capacidade de determinar diferenças de profundidade. Enquanto os projetos exploratórios citados anteriormente também tenham colaborado com a determinação e a confirmação de diversos requisitos do sistema, estes testes preliminares tiveram um caráter bem mais específico e foram mais úteis para esse fim. Seus resultados são usados ao longo do projeto principalmente para determinar que tipos de gestos e que posturas e movimentos devem ser reconhecidos pelo sistema, além de quais gestos devem ser associados a cada ação prevista nos testes com aplicativos e de algumas simplificações possíveis para a implementação do sistema.

Além destes testes preliminares, deve-se realizar testes durante todo o desenvolvimento já que, enquanto alguns dos requisitos do sistema podem ser verificados diretamente com base na definição do modelo de gestos, como é o caso do número de gestos possíveis, a maior parte precisa ser medida com base em testes com usuários e os resultados desses testes podem variar ao longo do desenvolvimento.

As principais medidas relacionadas aos requisitos e que podem ser obtidas diretamente com testes são o tempo necessário para preparação e calibração do sistema, o tempo médio consumido para o reconhecimento do gesto (em uma configuração específica de *hardware* para servir como indicador do processamento exigido por essas atividades) e as taxas de erros do reconhecimento (tanto de falsos

negativos como de falsos positivos), inclusive como são afetadas por mudanças em parâmetros do sistema (como tolerâncias, conjuntos de gestos utilizados) e no ambiente de teste. Durante todo o desenvolvimento, devem existir mecanismos no sistema para medir e registrar esses tempos e taxas, conforme cada funcionalidade é implementada. Além disso, a metodologia prevê a implementação de aplicativos interativos simples para testar os diferentes tipos de gestos, o uso de seus parâmetros e outras funcionalidades ao fim da implementação de cada uma e fornecer dados qualitativos, além dos quantitativos. Todos estes aplicativos de teste e mecanismos de medida e registro devem ser mantidos atualizados a cada nova interação de projeto, de forma que sejam utilizados como uma forma simples de teste de regressão.

Além destas medidas e testes funcionais, que pecam por apresentar o uso de gestos desligado de qualquer contexto, seu uso integrado às duas aplicações descritas anteriormente (CoH e ambiente de iluminação) deve ser testado por um número limitado de usuários utilizando uma variação do método de Passo-a-Passo Cognitivo (Rowley & Rhoades, 1992), em que as perguntas realizadas a cada passo refiram-se especificamente ao uso de gestos, procurando desacoplar o máximo possível questões relacionadas aos gestos de questões relacionadas à interface e às tarefas específicas de cada aplicativo mas ainda permitindo a contextualização do uso desta forma de interação.

1.4 ESTRUTURA DO TRABALHO

Esta introdução define o problema, a hipótese e os objetivos centrais deste trabalho, bem como seu contexto, motivação e justificativa, a metodologia usada para seu desenvolvimento e testes. O restante do documento segue a seguinte estrutura:

No capítulo dois são apresentados e discutidos conceitos importantes presentes na literatura, relativos a gestos e sua definição, classificação, seu reconhecimento e uso em interação. Esses conceitos são necessários para o entendimento do trabalho. A principal função deste capítulo, entretanto, é analisar e discutir as principais abordagens e algoritmos utilizados e descritos na literatura para

o reconhecimento de gestos como um todo bem como para cada uma de suas etapas, permitindo e justificando a seleção das abordagens utilizadas no desenvolvimento deste trabalho.

O modelo de gestos adotado, que permite a definição de um grande número de gestos, inclusive de tipos diferentes, e a estratégia usada para reconhecer os gestos por ele definidos são detalhados no capítulo três. Esse capítulo contém as principais contribuições do trabalho.

O restante do desenvolvimento do sistema de reconhecimento de gestos é descrito no quarto capítulo, incluindo as abordagens adotadas para resolver cada etapa do problema e sua implementação, além da arquitetura de *software* utilizada.

O capítulo cinco discute os testes do sistema desenvolvido e seus resultados e o sexto e último capítulo apresenta as conclusões do trabalho e possíveis trabalhos futuros.

2 REVISÃO DA LITERATURA SOBRE GESTOS

No capítulo anterior, diversos conceitos são utilizados sem que sejam formalmente definidos, como por exemplo posturas, gestos, seus tipos e as etapas de seu reconhecimento. Esses conceitos e suas definições adotadas são discutidos neste capítulo. Mas sua principal função, ao mesmo tempo em que apresenta o estado da arte em reconhecimento de gestos com as mãos livres, é analisar as diferentes abordagens utilizadas nesse reconhecimento para permitir a seleção de uma abordagem para o desenvolvimento.

Porque a bibliografia sobre reconhecimento de gestos é bastante vasta, delimitou-se o escopo da discussão de diferentes abordagens de reconhecimento aos trabalhos mais diretamente relacionados a este. Assim, embora existam na literatura diversos trabalhos e aplicações do uso de gestos reconhecidos com datagloves ou em mesas, esses sistemas tem características bastante distintas daquele que é objetivo aqui e, por isso, não foram pesquisados com abrangência. Mesmo dentre os trabalhos desse tipo que foram analisados durante a pesquisa, são discutidos somente alguns poucos, cujos resultados podem ser aproveitados apesar dessas diferenças. Entende-se que, em futuros trabalhos cujo principal foco seja o desenvolvimento e teste de técnicas de interação ou de aplicações com este sistema, este recorte não pode ser feito da mesma forma, sob pena de deixar de lado informações e conclusões importantes. Da mesma forma, evitou-se discutir o problema de reconhecimento e uso de gestos do corpo que, particularmente no caso de gestos dos braços, apresenta uma interseção com os gestos manuais, mas que muitas vezes é tratado de maneira bastante distinta, com menos graus de liberdade. Outros trabalhos que tem relação com o tema e foram identificados e analisados durante esta pesquisa mas que não são discutidos em maior detalhe aqui incluem algumas das abordagens com alto custo de processamento para reconhecimento da postura da mão utilizando uma câmera, baseadas em modelos 3D, e variações de técnicas para comparar imagens (usadas, por exemplo, para reconhecimento baseado na aparência da mão).

Além disso, diversas ferramentas que estão dentre as principais usadas para o reconhecimento de gestos são referenciadas neste capítulo sem que sejam discutidas em maior detalhe, visto que essa discussão está além do escopo deste

trabalho. Dentre essas ferramentas, podem ser citados filtros bayesianos (em particular filtros de Kalman e de partículas), *Hidden Markov Models* (HMMs), máquinas de estados finitos, auto-espacos, *Dynamic Time Warping* (DTW) e *Elastic Graph Matching* (EGM). Para permitir que o leitor que não tenha conhecimento prévio sobre essas ferramentas possa acompanhar o texto de forma mais clara, o Apêndice B fornece uma explicação bastante sintética sobre elas, sem a mínima pretensão de servir como introdução a seu uso, mas somente para facilitar a discussão dos trabalhos relacionados.

2.1 DEFINIÇÃO E CLASSIFICAÇÃO

Pavlovic et al. (1997) oferecem uma definição de gesto manual que combina a evolução da postura e da posição no tempo, com a intenção de comunicar ou manipular.

Com relação à postura da mão, coexistem duas definições bastante aceitas: pode significar a configuração que a mão assume definida pelos ângulos formados por todas as articulações de todos os dedos, independente da posição e da orientação da palma, ou pode ser independente somente da posição e definida pelos ângulos das articulações dos dedos e também pelos três ângulos que definem a rotação da mão como um todo. Neste trabalho, adota-se essa segunda definição e, de acordo com ela, a postura da mão aparenta ter 20 graus de liberdade, correspondentes às flexões ou extensões das três articulações e à rotação lateral de cada um dos cinco dedos. Na prática, principalmente para movimentos não forçados (aqueles feitos naturalmente, com ou sem esforço mas sem que se use a outra mão ou qualquer outro apoio como auxílio), esses movimentos não ocorrem de maneira independente e a dimensionalidade do problema pode ser reduzida.

A Figura 1, adaptada de Pavlovic et al. (1997), mostra essas articulações e algumas das restrições de seus movimentos. Além dessas regras explícitas, o uso de técnicas estatísticas como análise de componentes principais pode reduzir ainda mais essa dimensionalidade (Wu, Lin & Huang, 2001).

Figura 1 - Articulações da mão Fonte: Pavlovic et al. (1997)

A definição de Pavlovic et al. (1997) combina posturas e movimentos, o que de fato é aceita pela maioria dos autores, e limita-se a gestos voluntários, com a intenção de comunicar. Embora existam autores que incluam gestos involuntários em suas definições (inclusive podendo indicar a comunicação de diversos aspectos de que o gesticulador não está consciente), na maior parte das interfaces são úteis somente os gestos voluntários, por isso essa limitação da definição será mantida neste trabalho. A definição inclui também parâmetros espaço-temporais do gesto, como a velocidade com que é realizado. De fato, esses parâmetros podem alterar ou até inverter o significado do gesto. O mesmo movimento feito repetidas vezes com velocidade pode significar pressa ou rapidez, mas feito de forma lenta pode dar a entender calma ou que há tempo.

Essa definição peca, no entanto, em dois pontos: não inclui o local em que o gesto ocorre nem em relação a um referencial fixo no gesticulador nem a um fixo no ambiente. A maior parte dos autores concorda que o local onde o gesto é realizado (ou, por vezes, onde se inicia) pode modificar drasticamente seu significado. Por exemplo: um gesto feito com a mão em postura de apontar para o lado de dentro fazendo um movimento circular em torno de um eixo horizontal paralelo ao plano frontal do corpo pode significar continuidade, avançar ou voltar (dependendo do sentido de rotação) se for feito abaixo da altura da cabeça, mas quando feito

exatamente da mesma forma mas posicionado ao lado e próximo da cabeça, em geral indica demência. Em línguas de sinais, dentre outros contextos, a parte do corpo que é tocada ou não no início ou no final do gesto também pode alterar seu significado de forma semelhante. A posição do gesto em relação a elementos no ambiente e no contexto que cerca o gesticulador também pode modificar o seu significado, muitas vezes através do gesto de apontar, mas não necessariamente só nesse caso. Um gesto feito com uma das mãos enquanto a outra aponta para um prato de comida, por exemplo, pode assumir significado bastante distinto se ao invés disso apontar para uma pessoa.

Além disso, considera-se a tarefa de manipulação como distinta da gesticulação (poucas pessoas descreveriam o ato de pegar uma caneta, por exemplo, como um gesto). Por isso, neste trabalho é adotada a seguinte definição, que acrescenta àquela apresentada por Pavlovic et al. (1997) o local do gesto e elimina a manipulação: **um gesto manual é uma seqüência temporal e voluntária de posturas e posições das mãos, tendo como referencial de local o corpo do autor do gesto, podendo fazer referência ao ambiente que o cerca e contida num intervalo limitado de tempo, com a intenção consciente de comunicar.**

Embora esta definição inclua a evolução temporal do gesto, nada afirma sobre quando começa ou termina. Quek (1994) é um dentre a maioria dos autores que aceita uma divisão temporal dos gestos, proveniente de estudos psicológicos, em três fases: preparação, núcleo e retração. A preparação e a retração em geral são caracterizadas por movimentos rápidos da mão, respectivamente saindo de e voltando para uma posição de repouso, sem significado para a comunicação. O núcleo é a fase temporal do gesto em que ele assume significado, através da combinação dos fatores citados na definição acima. Pavlovic et al. (1997) discutem que a fase de retração pode ser suprimida, ou difícil de identificar, na transição entre dois gestos simultâneos. Quek (1994) afirma ainda que movimentos lentos entre as posições de repouso não constituem gestos e que gestos estáticos, aqueles definidos como uma única postura, devem ser mantidos por um certo tempo para terem significado.

Pesquisas da área de linguística (Liddell e Johnson, 1989) sugerem uma simplificação que pode ser aplicada à definição de gesto adotada ao afirmar que as mudanças na postura da mão que ocorrem durante o seu movimento não tem, em

geral, significado e, nesses casos, basta analisar as posturas assumidas pela mão antes de começar a se mover e depois que o movimento chega ao fim.

Com base no movimento, gestos manuais podem ser classificados como estáticos, definidos por uma única postura que, de acordo com Quek (1994), deve ser mantida por um certo tempo, e dinâmicos, em que há mudança de postura ou movimento da mão. Essa classificação é adotada neste trabalho, e os gestos dinâmicos são subdivididos em quatro tipos: aqueles em que ocorre somente uma rotação da postura em torno do pulso, os que consistem somente em uma mudança de postura, aqueles em que há somente movimento da mão e, por fim, os que combinam movimento e mudança de postura. É comum na literatura, no entanto, a utilização do termo "gestos dinâmicos" para indicar gestos definidos somente pelos movimentos da mão, ignorando mudanças na postura.

Classificações baseadas em outros fatores, como a função ou o significado dos gestos, são numerosas e raramente concordam. Neste trabalho adota-se uma combinação das classificações propostas por Nielsen et al. (2003) e Bowman et al. (2005):

- Gestos emblemáticos ou simbólicos são aqueles que tem significado próprio, sem a necessidade de imitar ou fazer alusão a nenhum objeto ou ação. Um exemplo é o gesto estático com o polegar levantado que indica aprovação e outro é o esfregar do indicador com o polegar que faz referência a dinheiro.
- Gestos mímicos em geral imitam um objeto ou uma ação para descrevê-los ou assumir seu significado, e podem ser exemplificados por fazer um círculo com as mãos ou imitar o ato de puxar um gatilho. Podem ser subdivididos em icônicos, que representam conceitos concretos como nos exemplos acima, ou metafóricos, que representam conceitos abstratos, como o gesto para indicar continuidade descrito anteriormente.
- Gestos deíticos são os gestos de apontar, indicando pessoas, objetos ou locais.
- Gestos proposicionais fazem referência consciente ao espaço ao redor do gesticulador e em geral ilustram tamanhos (por exemplo ao separar as mãos) ou movimentos.
- Gestos de batida são repetitivos e usados para indicar ritmo ou velocidade, ou para enfatizar palavras.

Os dois primeiros tipos de gestos, com significado próprio e discreto são classificados por Quek et al. (2002) como semaforicos e, de acordo com os autores, são usados com baixa frequência durante a comunicação espontânea, mas estão entre os gestos mais úteis para interação, inclusive para a realização de tarefas secundárias e não as que são o principal foco da interação, como mostrado por Karam & Schraefel (2005). Os três últimos tipos em geral são utilizados juntamente com a fala e são classificados dessa forma e em conjunto por Bowman et al. (2005).

Oviatt (1999), mostra que, durante a comunicação espontânea, somente 20% dos gestos utilizados por pessoas são deíticos mas o uso desse gesto é muito comum em interfaces baseadas somente em gestos e principalmente nas que os combinam com fala. Epps, Lichman & Wu (2006) teorizam que isso se dá principalmente quando é utilizada a metáfora de janelas, por associação do gesto de apontar com o ponteiro do mouse. Em testes realizados por eles com vinte voluntários instruídos para interagir utilizando somente posturas de mão (mas podendo escolher qualquer postura para a realização de qualquer tarefa) com uma interface convencional baseada em janelas, mais de 70% dos usuários utilizou a postura de apontar com o indicador estendido ou uma variação dela, na média de todas as 36 tarefas. Para algumas tarefas, como seleção, essa porcentagem superou 90%, mas para nenhuma (incluindo, por exemplo, rotação) foi inferior a 50%.

A maioria dos autores cita ainda outra função ou classificação de gestos, as línguas de sinais. Como se tratam, no entanto, de línguas tão ou mais complexas que as verbais, com regras próprias complexas que inclusive regem sua evolução e que utilizam não só gestos da mão mas do corpo inteiro, além de expressões faciais, considera-se simplista classificá-las como meros gestos. Neste trabalho, o problema de língua de sinais não será abordado, exceto quando trazer contribuições úteis para o uso de gestos em geral.

2.2 MODELAGEM E RECONHECIMENTO

De acordo com Marcel (2002) e Bowman et al. (2005), o reconhecimento de gestos por sistemas computacionais é feito principalmente de três formas: através de

datagloves, Visão Computacional (VC) ou de superfícies de interação sensíveis a um ou mais toques (seja usando VC ou outro tipo de sensor). Este trabalho discute principalmente a segunda forma.

Para esse caso, Marcel (2002) cita as principais etapas em que o reconhecimento pode ser subdividido: modelagem do gesto, segmentação da imagem, análise da imagem e reconhecimento ou classificação do gesto.

A modelagem define quais características do gesto (dentre as que constituem, por exemplo, a definição de gesto adotada neste trabalho) serão consideradas pelo sistema e de que forma são quantificadas por ele.

A segmentação é responsável por encontrar nas imagens as regiões de interesse, em geral as mãos e alguma referência da posição do corpo do gesticulador, como a cabeça. Separar as mãos dos braços é outra responsabilidade comum na segmentação.

Já a segmentação do gesto no tempo em geral ocorre durante a análise e o reconhecimento, levando em conta as três fases de um gesto apresentadas anteriormente (Quek, 1994) ou delimitadores mais claros, como posturas específicas ou locais específicos para a realização do gesto. Além disso, durante a análise são determinadas as principais características, ou *features*, usadas para definir e reconhecer os gestos e é feito o seu mapeamento no modelo de gesto adotado.

Por fim, o reconhecimento em geral consiste na classificação do gesto, utilizando as características já determinadas, dentre um vocabulário conhecido de possíveis gestos. Como essa breve descrição sugere, análise e reconhecimento muitas vezes estão intimamente ligados, uma vez que os algoritmos usados para a classificação do gesto o fazem com base nas características definidas e encontradas na análise.

Enquanto no restante deste capítulo são discutidos diversos trabalhos com soluções específicas para o problema de reconhecimento de gestos com VC e para cada uma das etapas em que pode ser subdividido, existem revisões da literatura mais gerais e bastante completas já publicadas sobre esse problema, cuja análise pode ser útil e nas quais se baseiam diversos pontos discutidos neste trabalho. A mais antiga, mas que contém uma série de conceitos importantes, é a de Pavlovic et al. (1997). As revisões mais recentes feitas por Turk (1998) e Marcel (2002) apresentam muitos dos mesmos conceitos sob óticas diferentes e complementam e atualizam o trabalho de Pavlovic et al. (1997). Mitra & Acharya (2007) se concentram mais nas soluções técnicas para o problema que nos conceitos mais básicos de

gestos e apresentam uma revisão bastante atual e Jaimes & Sebe (2007) discutem questões, soluções e aplicações do uso de gestos sob a ótica de interação multimodais.

Pode-se identificar duas categorias em que se dividem as principais estratégias para modelagem e reconhecimento de gestos. Uma delas trata cada gesto como um todo indivisível e procura reconhecê-lo assim, apesar do fato de geralmente lidar com gestos complexos, onde ocorrem movimentos e mudanças de postura da mão. A outra categoria inclui as estratégias que optam por dividir o gesto em elementos componentes e utilizar essa divisão em seu reconhecimento, classificando cada componente independentemente. Essas estratégias por vezes são chamadas de "dividir para conquistar" e dividem o gesto somente em postura e movimento, como é feito por exemplo por Fei & Reid (2003) e Wu & Huang (1999), mas podem ser vistas como uma abordagem linguística para a modelagem e reconhecimento de gestos, como discutido anteriormente e proposto por Derpanis et al. (2004), que dividem os gestos em postura, movimento e local com base na análise de línguas de sinais publicada por Stokoe (1960). Podem ser classificados também nessa segunda categoria trabalhos que utilizam um modelo mais simples de gestos e reconhecem somente posturas ou somente movimentos das mãos, mas não sua combinação.

As abordagens que procuram reconhecer o gesto como um todo, sem dividi-lo, são aplicadas, em sua maior parte, não para interação mas sim no domínio de línguas de sinais, em particular para reconhecer gestos que representam palavras inteiras. Quase todos esses gestos são definidos tanto por posturas específicas quanto por movimentos de uma ou de ambas as mãos (Stokoe, 1960), ao contrário do que ocorre para soletrar palavras, caso em que muitas letras são representadas por gestos estáticos. Em geral, nessas aplicações de reconhecimento de gestos, estados são definidos usando a posição da mão e algumas medidas simplificadas relativas à sua postura. Esses sistemas frequentemente fazem uso de vocabulários limitados e regras gramaticais para auxiliar no reconhecimento, já que utilizam esse estado onde a modelagem da postura pode ser tão ambígua.

A solução utilizada por Starner & Pentland (1995), por exemplo, utiliza HMMs com um modelo que, além da posição da mão, consideram a inclinação e a posição de uma elipse que a envolve como variáveis de estado. Bauer & Hienz (2000) também usam HMMs mas com um estado que define melhor a postura da mão,

baseado nas cores das partes de uma luva sobre a mão. Lichtenauer, Hendriks & Reinders (2008) utilizam a posição (em 3D) e somente o tamanho da imagem da mão e, além disso, mostram uma técnica que permite que um novo gesto seja reconhecido após ser demonstrado para treinamento uma única vez. E o GART ou *Gestures and Activities Recognition Toolkit* (Lyons et al., 2007), fornece ferramentas para facilitar o reconhecimento de gestos usando HMMs (e cujo estado, portanto, pode combinar características de postura e movimento, embora a ferramenta seja usada com mais frequência para reconhecimento de movimentos), inclusive com funcionalidades prontas para capturar entradas de sensores como câmeras e acelerômetros. Davis & Sha (1994) reconhecem gestos que incluem mudanças de postura e movimentos, modelados pelas trajetórias das pontas dos dedos, através de uma máquina de estados finitos. Florez et al. (2002) utiliza uma rede neural para gestos constituídos por uma sequência de duas posturas. Triesch & Malsburg (2001) usam EGM para reconhecer posturas e movimentos através da mudança das posições de seus nós e requerem apenas seis exemplos para o treinamento de cada gesto (relatam o uso de doze gestos estáticos). Darrel, Essa & Pentland (1996) usam DTW para reconhecer gestos quaisquer modelados como uma sequência de imagens da mão mudando de posição ou postura.

Em sua grande maioria, esses sistemas não são indicados para o reconhecimento de gestos estáticos, mas sim para os dinâmicos, visto que utilizam representações simplificadas da postura da mão para reduzir o tamanho do vetor de estado e, conseqüentemente, o custo de reconhecimento e treinamento, já que nesse caso o movimento é o componente mais significativo do gesto. Mesmo a diferenciação entre gestos com movimentos parecidos mas posturas distintas pode ser difícil nesses sistemas, já que muitas posturas são descritas de forma similar por seu número reduzido de variáveis de estado. Esse é um dos motivos pelos quais sistemas que utilizam essa abordagem têm como aplicação comum as línguas de sinais, que têm regras gramaticais conhecidas que podem ser utilizadas para auxiliar o reconhecimento. Há exceções a essa regra, como o sistema desenvolvido por Triesch & Malsburg (2001), mas Derpanis et al. (2004) afirmam que considerar o gesto como um todo em geral limita severamente o vocabulário possível ou aumenta consideravelmente a complexidade do sistema, tornando seu uso mais difícil e restrito, como relata Wobbrock et al. (2007) para o caso de reconhecimento de

gestos baseado em toque. Pavlovic et al. (1997) e Mitra & Acharya (2007) afirmam que, nesses sistemas, é raro o uso de mais de 50 ou 100 gestos possíveis.

Além disso, em muitos desses sistemas a inclusão de novos gestos ou o reconhecimento de diferentes subconjuntos de gestos são dificultados pela necessidade de treinamento, comum em todos esses casos. Enquanto no trabalho de Triesch & Malsburg (2001) e nos que fazem uso de máquinas de estados finitos ou HMMs, onde em geral cada máquina ou cada HMM representa um único gesto, essa desvantagem é minimizada uma vez que cada novo gesto requer somente um esforço relativamente pequeno para o treinamento. Em muitos outros casos, no entanto, como por exemplo em sistemas que utilizam redes neurais ou árvores de decisão treinadas, é comum que a inclusão ou exclusão de um único gesto torne necessário retreinar todo o sistema.

Foi com base nessas desvantagens das abordagens que consideram gestos como um todo, principalmente frente aos requisitos de grande número de gestos possíveis (inclusive os estáticos) e flexibilidade na escolha dos gestos utilizados (por exemplo para diferentes usuários, aplicações ou culturas), que neste trabalho opta-se por adotar a abordagem linguística desde a hipótese, utilizando-a para a modelagem e o reconhecimento de gestos. As subseções abaixo discutem em maior detalhe diferentes formas de modelar gestos e de fazer a análise e o reconhecimento de seus principais componentes, movimentos e posturas, em trabalhos que utilizam essa abordagem linguística ou que consideram somente um desses componentes no modelo de gestos.

2.2.1 Modelagem

Na vasta literatura sobre reconhecimento de gestos, são muito raros os casos em que o gesto é modelado como a combinação de elementos mais simples, reconhecidos separadamente, como se propõe fazer neste trabalho, combinando duas posturas, movimento e local, permitindo a realização de todos os tipos de gestos citados na introdução e que ocorreram nos testes preliminares.

Um dos únicos sistemas que adota e implementa essa estratégia foi desenvolvido por Liu & Fujimura (2004). Os autores citam a possibilidade de

combinar, no modelo do gesto com uma ou duas mãos, uma única postura com um movimento e um local para sua realização e discutem como essa combinação pode aumentar consideravelmente o vocabulário de gestos possíveis. Utilizando 12 movimentos, 7 posturas com uma das mãos e um exemplo de postura feita com ambas as mãos se tocando, reconhece aproximadamente 20 gestos, mas afirma que a combinação de todas as posturas, movimentos e locais que reconhece permitem a definição de até uma centena deles, o que ainda é um número relativamente pequeno de acordo com Derpanis et al. (2004). Esse sistema faz uso de uma câmera capaz de fornecer um mapa de profundidade da imagem.

O sistema OGRE (Salles et al., 2004) também permite a definição de gestos formados por uma série de posturas ligadas por movimentos retilíneos de uma das mãos (e os chama de gestos "ensaiados"). Além disso, define gestos estáticos como somente uma postura e dinâmicos como somente o movimento, ambos para uma das mãos. Dentre os trabalhos pesquisados, o OGRE foi o único que afirmou explicitamente que prevê a configuração do sistema para reconhecimento de um conjunto menor de gestos, selecionado a partir de um conjunto maior de gestos possíveis, o que, de acordo com os autores, torna o reconhecimento mais rápido e preciso.

Derpanis et al. (2004) também modelam o gesto, a partir de uma abordagem linguística, como a combinação de postura, movimento e local, e chamam atenção para como um número relativamente pequeno de cada um desses elementos pode gerar um grande número de sinais possíveis. Nesse trabalho, no entanto, é relatada somente a implementação e testes do reconhecimento de movimentos para uma das mãos e não desse grande número de sinais.

Outros exemplos onde o modelo de gestos inclui posturas ou movimentos mas não a combinação dos dois são os trabalhos de Mo (2007) e Sclaroff et al. (2005). Esse último, no entanto, é uma síntese de diversas pesquisas sobre gestos realizadas em uma mesma instituição e, por isso, não está claro se posturas e gestos estão combinados em um mesmo modelo, ou constituem modelos em sistemas separados.

A raridade da combinação de movimentos, posturas e local dos gestos durante essa revisão chegou a levantar dúvidas sobre a validade dessa combinação no uso cotidiano de gestos e não em línguas de sinais. É simples encontrar uma série de exemplos, no entanto, que mostram essa validade e isso ocorreu inclusive

durante os testes preliminares relatados na introdução. A importância do local já foi exemplificada. Gestos realizados com o mesmo movimento mas com posturas diferentes podem ter significados completamente diferentes, por exemplo como acenar ou pedir carona. O mesmo ocorre quando a mesma postura é usada com movimentos diferentes. Puxar um punho em direção ao corpo pode ter o significado oposto ou nem mesmo estar relacionado a estender bruscamente o braço levando o punho fechado à frente.

Dentre os trabalhos pesquisados, somente três permitem a modelagem de um gesto dinâmico como a transição entre duas posturas: Florez et al. (2002), Shamaie & Sutherland (2003) e Utsumi, Tetsutani, & Igi (2002). Assumindo que seja possível definir um movimento como nulo nos gestos ensaiados do OGRE, esse sistema também permite o uso desse tipo de gesto.

Um grande número de trabalhos reconhece somente gestos estáticos (na maior parte das vezes para uma única mão) e, portanto, modelam gestos como uma única postura, mantida por um certo tempo. Por vezes utilizam também parâmetros como posição para a interação e, em alguns casos, rastreiam essa posição em vez de detectá-la em cada quadro⁵.

Segen & Kumar (1999) são um exemplo desse tipo de modelagem, mas que se diferencia dos demais por estimar a postura da mão em três dimensões utilizando a sombra que ela projeta em um ambiente com fundo e iluminação controlados e conhecidos. Jennings & Murray (1997) utilizam posturas para controle de um robô, enquanto Bretzner, Laptev & Lindenberg (2002) o fazem para controle de eletrodomésticos e Souza et al. (2006) para controlar avatares em um ambiente 3D. Chen et al. (2007) utilizam comandos na tarefa de navegação, em um ambiente virtual que contém objetos de aprendizagem, usados em educação. Kolsch et al. (2004) implementaram o HandVu, um sistema que facilita o rastreamento da mão direita em seis posturas e que visa facilitar o uso de gestos, inclusive para computação móvel.

⁵ Essa frase indica a distinção entre detecção e rastreamento, que será utilizada ao longo do texto. A primeira utiliza somente as informações obtidas a cada instante de tempo para localizar objetos de interesse, enquanto o segundo aproveita a coerência temporal da maioria dos sistemas e utiliza também informações de instantes anteriores para auxiliar essa localização.

Outros exemplos de trabalhos que descrevem o uso de gestos modelados como uma única postura são Hardenberg & Bérard (2001), Mo & Neumann (2006), Maclean et al. (2001), Guan et al. (2006), Terrillon et al. (2002), Peixoto & Carreira (2005), O'Hagan et al. (2002), Du & Li (2000), Schlattmann et al. (2007), Liu & Jia (2004) e Black & Jepson (1998b).

A postura em si pode ser modelada através de características de sua imagem ou com modelos 3D ou simplificados da mão, usando, por exemplo, cilindros curvas quadráticas ou simplesmente retângulos em lugar das falanges (Erol et al., 2005). Todos os trabalhos citados acima modelam a postura da mão com propriedades da imagem.

Quase tão comum quanto modelar um gesto como somente uma postura é considerar somente o movimento da mão, descartando suas posturas ou local.

Kim et al. (2006), em um exemplo de computação atenta, modela um único gesto para reconhecimento por um robô, o de acenar para atrair sua atenção, mas o faz em condições adversas, até mesmo com o usuário deitado. Yang & Ahuja (1998), Ramamoorthy et al. (2003), Hong, Turk & Huang (2000), Rajko & Qian (2008), Marcel et al. (2000), Lee & Kim (1999), Mammen, Chaudhuri & Agarwal (2002), Shan et al. (2004), Black & Jepson (1998a) e Appenrodt et al. (2009) reconhecem gestos modelados como movimentos realizados com uma das mãos, enquanto Suk, Sin & Lee (2008), Just, Bernier & Marcel (2004), Mammen, Chaudhuri & Agarwal (2001), Shamaie & Sutherland (2004) fazem o mesmo para gestos bimanuais. A maior parte desses trabalhos faz o rastreamento da posição da mão e todos modelam a mão através de sua aparência e não com um modelo 3D.

Foram analisados poucos trabalhos que modelam gestos deíticos para reconhecimento por VC. Dentre eles, pode-se citar Park, Roh e Lee (2008), que identificam com uma única câmera a direção em 3D em que um usuário aponta utilizando uma extensão total ou parcial do braço. Marcel et al. (2000) também modela e reconhece gestos deíticos.

Como é comum que o modelo de gesto adotado e as características usadas na análise e no reconhecimento do gesto estejam intimamente relacionados, é preciso estar atento para evitar a confusão entre esses conceitos. O modelo do gesto está muito mais relacionado com conceitos de mais alto nível, como a definição de um gesto, que com a tecnologia usada para reconhecê-lo e é possível que nem mesmo especifique detalhes como, por exemplo, que características são

usadas para descrever a postura da mão. Essa foi inclusive a abordagem adotada na discussão de modelagem feita acima. Ainda que haja casos em que o modelo do gesto usado tem os mesmos elementos que o conjunto de características extraídos na análise (como, por exemplo, no caso de um gesto modelado como as trajetórias no espaço das pontas dos dedos, cujas posições espaciais são extraídas como características utilizando um conjunto de câmeras), o modelo e as características podem também ter naturezas bastante distintas (ainda usando as posições espaciais dos dedos como exemplo de características, o modelo poderia ser definido como o conjunto dos ângulos das articulações de um modelo 3D de mão).

2.2.2 Análise e reconhecimento dos movimentos

Para reconhecer o movimento, em geral basta utilizar como característica a trajetória que um ponto de referência (em geral o centróide da imagem da mão) percorre na imagem ou no espaço ao longo do tempo. Quando essa trajetória é bidimensional em vez de 3D, pode-se utilizar a área da mão como *feature* para estimar seu movimento em profundidade. Essas duas características (trajetória 2D do centróide e área da mão na imagem) são adotadas também neste trabalho.

Um grande número das soluções para reconhecimento dos movimentos opta por rastrear essa trajetória, em vez de detectar a posição independentemente em cada quadro. Mo (2007) chama atenção para o fato de que, embora isso torne a localização da mão mais robusta e até mais rápida, pode trazer problemas para o uso desses sistemas para interação, pois é comum que tais sistemas exijam calibração ou inicialização complexa e precisem de reinicialização após falhas no rastreamento. Sistemas com inicialização e reinicialização automáticas, em geral contornam esse problema passando a utilizar detecção automaticamente quando uma falha de rastreamento é percebida. Jennings (1999) e Schlattmann et al. (2007) mostram a implementação de sistemas com essa característica. No presente trabalho, optou-se por utilizar inicialmente somente a detecção dos objetos de interesse, em vez de seu rastreamento, com a possibilidade de implementar esse rastreamento caso a detecção não se mostre suficientemente rápida e robusta durante os testes. São três os objetos distintos cuja posição pode precisar ser

determinada continuamente neste trabalho: as duas mãos, no caso de gestos bimanuais, e a cabeça, para servir como referência para o local do gesto.

Muitas das ferramentas citadas anteriormente e usadas para o reconhecimento de gestos como um todo são de natureza suficientemente geral para poder ser utilizadas no reconhecimento de diversos outros padrões, dentre eles, de movimentos das mãos.

HMMs são a ferramenta mais utilizada para fazer esse reconhecimento. Para movimentos com uma só mão, é possível citar como exemplos os trabalhos de Appenrodt et al. (2009), Marcel et al. (2000) e Lee & Kim (1999). Rajko & Qian (2008) faz o mesmo, mas utilizando modelos com parâmetros reduzidos para ganhar velocidade e Ishihara & Otsu (2004) utilizam HMMs em conjunto com características *Haar-like* (Viola & Jones, 2001). Movimentos bimanuais também são reconhecidos com HMMs, como mostrado por Just, Bernier & Marcel (2004) e Shamaie & Sutherland (2004). Esses últimos utilizam uma rede bayesiana para, ao mesmo tempo em que rastreia a mão, fazer a fusão das saídas dos HMMs usados em conjunto com cada uma das mãos.

Além de HMMs, DTW é usado por Sclaroff et al. (2005), redes neurais por Yang & Ahuja (1998) e máquinas de estados finitos por Hong, Turk e Huang (2000).

Todas essas ferramentas são bastante úteis para o reconhecimento de movimentos genéricos e quaisquer, que podem ser treinados. Movimentos que vão de girar e lançar um laço virtual a traçar letras ou números no ar podem ser utilizados. Com base nos testes preliminares, no entanto, verificou-se que gestos mais complexos são difíceis ou demorados de realizar e em geral requerem movimentos de maior amplitude o que, de acordo com Nielsen et al. (2003), aumenta a possibilidade de problemas de ergonomia na interface baseada em gestos. Não é surpresa, assim, que em nenhum dos gestos sugeridos pelos usuários nos testes preliminares foram usados movimentos complexos. Por isso, neste trabalho optou-se por utilizar somente os doze gestos básicos cujo uso foi considerado fácil nos testes preliminares: os dois movimentos circulares, nos sentidos horário e anti-horário no plano da imagem, e os dez movimentos lineares, nos dois sentidos sobre três eixos, mais quatro diagonais no plano da imagem. Esses movimentos básicos são de uso bastante comum, por exemplo em línguas de sinais (Stokoe, 1960), como também para gestos em dispositivos de interação por toque (Elias et al., 2007).

Embora ferramentas como HMMs possam ser utilizadas para o reconhecimento também desses movimentos simples, trabalhos como o de Mammen et al. (2002) mostram que ferramentas mais simples são suficientes para reconhecê-los, como árvores de decisão simples, com poucos níveis e regras definidas explicitamente, sem treinamento. Neste trabalho, é feita a opção por utilizar esta abordagem mais simples, a menos que se mostre inadequada durante os testes.

Com relação aos dois sistemas citados anteriormente que combinam posturas e movimentos em seu modelo de gestos, o OGRE (Salles et al., 2004) utiliza uma biblioteca para interfaces baseadas em caligrafia para reconhecimento dos movimentos e Liu & Fujimura (2004) os classificam usando o mínimo erro quadrático entre as componentes x, y e z do movimento, uma das quais pode ser medida diretamente a partir do mapa de profundidade.

2.2.3 Posturas

Como as posturas das mãos são usadas em grande parte dos modelos de gestos e como são o componente do gesto com maior número de graus de liberdade, uma grande variedade de características extraídas na análise e abordagens para o reconhecimento coexistem.

2.2.3.1 Características

A literatura traz um grande número de exemplos de características diferentes que são utilizadas na análise das posturas da mão. Para Bauer e Hienz (2000), por exemplo, as características são as posições e tamanhos de *blobs* coloridos, cada cor correspondendo a um dedo e à palma da luva usada pelo usuário. Para Davis & Sha (1994), são as posições dos marcadores nas pontas dos dedos do usuário.

Uma das características mais utilizada para reconhecimento da postura são representações do contorno da imagem da mão. Nesse contorno os máximos e mínimos locais tem grande importância e, por vezes, são usados como únicas

características (O'Hagan et al., 2002) ou para alinhar os vetores que as contêm. É comum utilizar coordenadas polares com origem no centróide da imagem da mão para descrever os pontos do contorno e inclusive utilizar somente a distância (em geral normalizada) e desprezar a fase, principalmente caso os vetores sejam alinhados pelos mínimos e máximos locais. Imai et al. (2004) e Shimada et al. (2001) utilizam exatamente essa abordagem para permitir a diferenciação entre milhares de posturas. Lee et al. (1999) utilizam um histograma de orientação das bordas como *feature* da postura da mão e Freeman et al. (1996) faz o mesmo para posturas do corpo. No OGRE, Salles et al. (2004) testaram diversas características diferentes para o reconhecimento de posturas e selecionaram como a mais precisa histogramas das distâncias e ângulos entre pontos do contorno. Mo (2007) e Mo & Neumann (2006) utilizam alguns elementos do contorno, além das pontas dos dedos, como características. Dentre o grande número de características utilizadas no reconhecimento de posturas, a literatura indica que representações do contorno da mão estão entre as que permitem o reconhecimento da maior variedade de posturas com boa precisão. Por esses motivos, uma dessas representações é utilizada no desenvolvimento deste trabalho, mas pode-se destacar outras alternativas muito utilizadas que também foram pesquisadas.

As posições das pontas dos dedos também são características muito utilizadas, principalmente pela facilidade de sua segmentação sem a necessidade de conhecimento prévio da postura. O uso dessas características traz ainda a vantagem de não depender da segmentação entre a mão e o braço do usuário. Pode, no entanto, reduzir o número de posturas possíveis. Além de Davis & Sha (1994), Rehg & Kanade (1993), Hardenberg & Bérard (2001), Jennings (1999) e Maclean et al. (2001) também as utilizam. Schlattmann et al. (2007) reconstroem o *visual hull* da mão, uma representação espacial obtida combinando as silhuetas da mão vistas por múltiplas câmeras em diferentes posições, e o utilizam para encontrar pontas dos dedos em 3D.

Diferentes momentos da imagem⁶ da mão também são utilizados com frequência como características. Freeman et al. (1996) já o faz, assim como Fei &

⁶ Funções da intensidade dos pixels, para uma imagem ou parte dela, e de suas coordenadas elevadas a diferentes potências. Existem diversas formas de calcular conjuntos desses momentos para descrever padrões de forma que sejam invariantes a escalonamento, rotação ou espelhamento.

Reid (2003). Salles et al. (2004) também os utilizam como uma das formas de comparar posturas e concluem que tem baixa precisão. Terrillon et al. (2002) fazem uma pré-qualificação dos componentes conexos (restantes após a segmentação) baseada nos momentos de imagem, cujo principal objetivo é encontrar a posição da cabeça do usuário na imagem.

Algumas características são armazenadas em estruturas bidimensionais em vez de em vetores. Peixoto & Carreira (2005), por exemplo, utilizam as coordenadas polares dos pontos do contorno, como explicado anteriormente, mas combinam distância e fase para formar uma nova imagem, que chamam de imagem de assinatura do contorno. Terrillon et al. (2002) utilizam a fase da transformada de Fourier da imagem binária resultante da segmentação da mão para descrever sua postura, o que funciona como um filtro passa-alta e magnifica a contribuição das bordas na comparação de imagens. Liu e Fujimura (2004) utilizam a transformada de distância da imagem binária (em que cada pixel recebe o valor de sua distância a um ponto do fundo da imagem).

Sclaroff et al. (2005) combinam quatro tipos de características para descrever posturas: pontos do contorno da mão, um histograma de orientação dos pixels desse contorno, os dedos, definidos por suas pontas (pontos de máxima curvatura) e mínimos locais (a base do dedo) e momentos de imagem.

A Tabela 2 mostra um resumo das particularidades das principais características para reconhecimento das posturas discutidas nesta seção.

Tabela 2 - Características para reconhecimento de posturas

Característica	Particularidades
Marcadores (pontas dos dedos, luvas)	Detecção simples e precisa, marcadores presos ao usuário ou vestidos
Representações do contorno da mão	Maior variedade de posturas que podem ser diferenciadas
Pontas dos dedos	Simplicidade na detecção, poucas posturas que podem ser diferenciadas
Momentos de imagem	Maior simplicidade, menor número de posturas que podem ser diferenciadas

2.2.3.2 Reconhecimento

Pode-se dividir as abordagens para o reconhecimento da postura da mão, utilizando as características extraídas das imagens da mão, em dois grandes grupos: as baseadas em sua **aparência** e as baseadas em **modelos 3D** da mão.

As abordagens baseadas em aparência resumem-se a estratégias para classificar as posturas comparando a aparência da mão às aparências de um conjunto pré-definido de posturas, sem procurar inferir qualquer informação sobre o estado real da mão no espaço. Marcel (2002) afirma que são três as principais estratégias de reconhecimento baseado em imagens: busca por máxima verossimilhança, árvores de decisão e redes neurais.

Na grande maioria dos trabalhos analisados é utilizada a busca pela postura no espaço das possíveis que tenha máxima verossimilhança com a postura observada. Na maior parte das vezes isso significa a menor distância no espaço definido pelas características, como por exemplo no caso do OGRE (Salles et al., 2004), ou ainda maior correlação entre elas, como feito por Liu e Fujimura (2004). Menos comum é fazer essa busca em auto-espacos, como fazem Black & Jepson (1998), Peixoto & Carreira (2005) e Shamaie & Sutherland (2003).

As árvores de decisão muitas vezes são usadas com regras definidas explicitamente e sem treinamento, baseadas na posição das pontas dos dedos, como nos trabalhos de Hardenberg & Bérard (2001), Maclean et al. (2001) e Jennings (1999). Podem ser utilizadas também com treinamento, por exemplo para classificar as posturas com base em características do tipo *Haar-like* (Viola & Jones, 2001), como fazem Ong & Bowden (2004) e Chen et al. (2007). Redes neurais são usadas no reconhecimento de posturas por Souza et al. (2006) e Florez et al. (2002).

Fei & Reid (2003) não utilizam nenhuma dessas três estratégias. Ao invés disso utilizam HMMs exclusivamente para o reconhecimento da postura, o que se mostrou bastante incomum dentre os trabalhos pesquisados.

No reconhecimento baseado em modelos 3D da mão, procura-se determinar os parâmetros desse modelo, em geral os ângulos assumidos pelas articulações da mão, com base nas características observadas na imagem da mão real. Erol et al. (2005) apresentam uma revisão extensa sobre esse tipo de abordagem e mostram que uma de suas principais vantagens é que podem diferenciar um número muito

elevado (da ordem de dezenas de milhares) de posturas distintas, enquanto os métodos baseados em imagens perdem robustez conforme cresce o vocabulário de gestos possíveis em um momento. Erol et al. (2005) afirmam que os métodos baseados em modelos podem fornecer precisão comparável a obtida com *datagloves* sem apresentar as desvantagens desse dispositivo, já discutidas. Podem, portanto, ser utilizados como interface natural em aplicações que requerem esse nível de precisão como, por exemplo, as que envolvem o treinamento de habilidades motoras. Erol et al. (2005) listam diversas soluções para o reconhecimento de posturas baseado em modelos 3D e as classificam em três categorias: baseadas em detecção (ou "quadro único", o termo usado pelos autores), em rastreamento com uma única hipótese e rastreamento com múltiplas hipóteses.

A principal desvantagem dessa abordagem, e o motivo pelo qual não é discutida mais a fundo neste trabalho, é que, dado o grande número de graus de liberdade da mão, todos esses métodos tem um custo computacional extremamente elevado para serem utilizados em tempo real. Dentre os numerosos trabalhos analisados por Erol et al. (2005), existem sistemas que requerem tempos da ordem de uma hora para processar um único quadro enquanto somente um que utiliza exclusivamente um modelo 3D identifica as posturas da mão em tempo real. Trata-se do DigitEyes (Rehg e Kanade, 1993), curiosamente um dos sistemas mais antigos presentes na revisão. O DigitEyes só atinge execução em tempo real, no entanto, usando *hardware* específico para acelerar o processamento de imagens e, principalmente, limitando os graus de liberdade rastreados ao movimento de três dedos, com cinco graus de liberdade no total, e a rotações do pulso mas não translações da mão (o que é, na verdade, bastante comum nesses sistemas, cujo foco é a postura da mão), que somam mais três graus. Embora a capacidade de processamento tenha se multiplicado muitas vezes desde então, a complexidade do problema aumenta exponencialmente de forma ainda mais rápida ao serem considerados mais graus de liberdade e, por isso, não é surpreendente que até hoje não tenham sido publicadas soluções que tratem o problema completo em taxas próximas de tempo real.

Há, no entanto, uma terceira abordagem que, por vezes, também é considerada como baseada em modelos 3D mas que combina o uso de modelos com características da abordagem baseada em imagens. Nesse caso, o modelo de

mão é utilizado para gerar um grande número de imagens correspondentes a suas possíveis configurações antes da tarefa de reconhecimento, já levando em conta as restrições do movimento das mãos, como por exemplo as ilustradas na Figura 1. Assim, o problema de reconhecimento se reduz a um problema de busca de imagens em uma base bastante extensa, mas ainda é possível obter os ângulos das articulações da mão durante o reconhecimento, visto que os parâmetros do modelo 3D usados para gerar cada uma das imagens de comparação são conhecidos. Essa abordagem é chamada de Estimação por Síntese (ES).

Comparada à abordagem baseada somente em imagens, a ES permite um rastreamento mais simples e robusto da postura da mão por utilizar os parâmetros do modelo 3D como estado. Como afirmam Mammen, Chaudhuri & Agarwal (2001), enquanto as propriedades da imagem da mão são uma projeção bidimensional de uma estrutura com muitos graus de liberdade e por isso sofrem variações muito grandes, mesmo durante a movimentação natural, para que sejam rastreadas de forma simples, os ângulos das articulações variam de forma mais suave e podem ser rastreados com maior facilidade. Comparada às abordagens baseadas só em modelos, a ES pode ser bem mais rápida, principalmente se as imagens geradas forem organizadas de forma a facilitar estratégias mais eficientes de busca (em geral são organizadas por similaridade). Por vezes um modelo de transição de posturas está embutido nessa organização das imagens, de forma que o rastreamento da postura e a busca nas imagens tornam-se uma só tarefa.

A revisão de Erol et al. (2005) também inclui métodos ES. Ainda assim, somente mais um sistema (Shimada et al., 2001) dentre os analisados pelos autores mostra-se capaz de realizar o reconhecimento em tempo real. Atualmente o sistema desenvolvido por Imai & Shimada (2004), baseado em muitos dos mesmos princípios, também é capaz disso. Wang & Popovic (2009) utilizam outra abordagem para acelerar a busca a ponto de permitir o reconhecimento em tempo real, usando uma luva colorida com um padrão específico para eliminar as ambiguidades da mão.

Todos os métodos que utilizam modelos, ES ou não, para reconhecer posturas em tempo real realizam o rastreamento de uma única hipótese. Isso não é de surpreender pois, comparado com a detecção, o rastreamento pode reduzir drasticamente o espaço de busca. Já comparado ao uso de múltiplas hipóteses é menos robusto, mas bem mais simples e rápido.

Como neste trabalho, para permitir o uso de gestos em interação, é necessário que as posturas sejam identificadas em tempo real, decidiu-se utilizar ou reconhecimento de postura baseado em imagens ou em ES. A primeira alternativa consome menos processamento, enquanto a segunda é mais robusta, mas o rastreamento requer que posturas muito parecidas sejam diferenciadas, o que torna mais complexo atender o requisito de permitir variações na execução dos gestos pelos usuários. Por esses motivos, optou-se por utilizar o reconhecimento baseado em imagens, baseado na busca pela postura com máxima verosimilhança dentre as possíveis. Assim como nas abordagens de ES, no entanto, optou-se por utilizar um modelo 3D de mão para facilitar a criação do conjunto de posturas possíveis.

A Tabela 3 resume as particularidades destas três principais abordagens para o reconhecimento de posturas.

Tabela 3 - Reconhecimento de posturas

Abordagem	Particularidades
Propriedades da imagem	Requer menor quantidade de processamento, perde robustez com crescimento do número de posturas
Modelos 3D da mão	Ainda não é executável em tempo real, grande número de posturas
Estimativa por síntese (ES)	Pode ser feito em tempo real, grande número de posturas, rastreamento a partir de pequenas variações

2.3 SEGMENTAÇÃO DA IMAGEM

A segmentação dos objetos de interesse na imagem não é o foco deste trabalho, mas é necessária para permitir a análise e o reconhecimento dos gestos com base nas imagens. Em sua grande maioria, os trabalhos pesquisados fazem a segmentação de uma ou ambas as mãos (e por vezes também do rosto, como ponto de referência) nas imagens obtidas pelas câmeras classificando os pixels, um a um e com base em sua cor, como sendo pele ou não. Isso é possível visto que o tom de pele da maior parte das pessoas, se encontra em um intervalo relativamente pequeno entre tons de vermelho e amarelo, devido ao sangue e à melanina, respectivamente (Fleck & Forsyth, 2009).

Vezhnevets et al. (2003) apresentam uma extensa revisão da literatura sobre essa estratégia de segmentação, estudam a influência do uso de diversos espaços

de cores distintos (e conclui que, na prática, tem pouca influência) e divide as soluções encontradas em três grandes categorias.

A primeira é o uso de regras explícitas para definir se um pixel é ou não classificado como pele, com base em operações e comparações feitas sobre os componentes de sua cor.

A segunda são modelos não-paramétricos de pele, em que durante uma etapa de treinamento é atribuída a cada ponto do espaço de cores uma probabilidade individual de ser pele ou não, baseada em quantas vezes aquela cor foi classificada como pele nessa etapa. Nenhum modelo matemático é utilizado para sintetizar essa distribuição e, ao invés disso, o mapa completo é utilizado na classificação. Por isso essa categoria também é conhecida como mapas de probabilidade de pele, ou pela sigla em inglês SPM. Essa abordagem consome uma grande quantidade de memória para armazenamento do mapa de cores e probabilidades, mas seu treinamento e uso são bastante rápidos. É impraticável, no entanto, generalizar ou interpolar os dados obtidos durante o treinamento.

Por fim, modelos paramétricos da distribuição de probabilidade de uma cor ser pele no espaço de cores são representações mais compactas que um SPM, por exemplo como uma gaussiana ou uma mistura delas.

Uma comparação dessas abordagens mostra que os SPM têm o melhor desempenho, medida pelo número de falsos positivos e negativos, mas que alguns conjuntos de regras explícitas, em particular as determinadas de forma automática por Gomez e Morales (2002) e Gomez (2002), têm desempenho bastante similar ou até superior a algumas das abordagens mais custosas. As misturas de gaussianas utilizadas por Jones & Rehg (1999) têm desempenho similar a um SPM, até porque é utilizado um número suficientemente grande de gaussianas para que a mistura se aproxime de um SPM.

Vezhnevets et al. (2003) apresentam ainda uma categoria distinta, de métodos dinâmicos para classificação, utilizados quando se pode calibrar e atualizar o sistema para cada caso específico (ou seja, para um usuário, câmera e condições de iluminação e fundo específicas). Nesse caso, é comum o uso de modelos mais simples, como os paramétricos, para que possam ser atualizado de forma rápida e simples durante a calibração e o uso, e com desempenho similar ou até superior a de outros métodos. A alta taxa de falsos positivos que normalmente degrada o desempenho de métodos paramétricos não se constitui como problema tão grave

nesse caso, em que se usa um modelo mais específico para a cor de pele, em vez de um modelo geral para ser aplicado em diversas situações distintas.

O presente trabalho opta por utilizar essa estratégia simples de classificação baseada somente na cor do pixel, até mesmo visto que essa atividade não é seu foco. Para determinar qual algoritmo deve ser utilizado, métodos dinâmicos com base em uma rápida etapa de calibração devem ser comparados com as misturas de gaussianas utilizadas por Jones & Rehg (1999) e com as regras explícitas propostas por Gomez e Morales (2002) e Gomez (2002), métodos que têm desempenho comparável ao uso de SPMs .

O uso de algumas restrições com relação ao fundo da cena pode melhorar consideravelmente o desempenho da segmentação pixel a pixel baseada em cor. A remoção de um fundo estático conhecido, por exemplo, elimina um grande número de possíveis falsos positivos. Outra técnica comum é segmentar regiões baseadas em seu movimento, assumindo que o fundo seja estático, que tem o mesmo efeito. Essas estratégias são ainda mais comuns em interfaces gestuais baseadas em mesas, mas também são frequentemente aplicadas no domínio deste trabalho. Hardenberg & Bérard (2001), por exemplo, combinam as duas. Enquanto o uso do movimento não seja adequado a este trabalho, visto que se deseja detectar regiões que podem permanecer praticamente estáticas por longos períodos, como a cabeça do usuário enquanto ele gesticula com as mãos, a remoção de um fundo conhecido é uma possibilidade para melhorar a segmentação, caso os testes mostrem que isso é necessário. Outras alternativas que foram estudadas e que podem ser utilizadas em trabalhos futuros são descritas a seguir.

Existem alternativas para a classificação baseada em cor com maior custo computacional do que a feita pixel a pixel, mas que levam em conta também a vizinhança do pixel, atribuindo maior probabilidade de ser pele a pixels pertencentes a regiões onde essa probabilidade já é alta. Jedynak et al. (2002) apresentam um exemplo dessa abordagem.

Depois da baseada em cor, a estratégia de segmentação usada com mais frequência nos trabalhos analisados baseia-se em mapas de profundidade, em geral obtidos através de equipamento específico para esse fim, como câmeras *Time-of-Flight*. Assumindo a hipótese de que a mão do usuário é o elemento da cena mais próximo da câmera, um simples *threshold* na profundidade é suficiente. Mo & Neumann (2006), por exemplo, adotam um valor para esse *threshold* baseado no

tamanho da palma em quadros anteriores. Liu & Fujimura (2004) e Lichtenauer, Hendriks & Reinders (2008) utilizam uma abordagem similar. Appenrodt et al. (2009) comparam o uso de cor, profundidade e imagens térmicas e concluem que a segmentação baseada na profundidade traz os melhores resultados. Enquanto essa alternativa é descartada para uso no presente trabalho, devido ao requisito de uso de equipamentos acessíveis e de baixo custo, é uma opção de grande interesse para trabalhos futuro, visto que diversos outros projetos envolvendo gestos em ambientes 3D com os quais este sistema pode ser integrado já fazem uso de equipamentos capazes de gerar mapas de profundidade.

A segmentação baseada em informação de bordas, em vez de comparação de cores, em geral só é utilizada quando se possui um modelo para a forma dessa borda. Para gestos manuais, existem dois casos em que isso é comum: o uso de bordas para detectar as pontas dos dedos que, quando não estão oclusas, tem um padrão de reconhecimento simples em praticamente todos os ângulos, ou no caso em que a postura é conhecida, ou quando se conhece um pequeno número de posturas possíveis e um modelo de transição entre elas. Embora nenhuma dessas condições esteja presente neste trabalho, a informação das bordas na imagem pode ser usada de uma forma simples para auxiliar a segmentação separando regiões de cor de pele que se toquem, simplesmente subtraindo a imagem das bordas da imagem já segmentada, antes que sejam feitas sua binarização e a identificação dos contornos, como sugerido por Terrillon et al. (2002).

Como exemplo de segmentação usando informação de bordas, Rehg & Kanade (1993) identificam as pontas dos dedos nas imagens dessa forma e Caglar & Lobo (2006) fazem a segmentação baseada em modelos dos dedos e de suas pontas. Wang, Zhang & Dai (2007) combinam o rastreamento por cor, com o algoritmo Camshift (Bradski & Kaehler, 2008), e das bordas das pontas dos dedos, usando Condensation. Usando esse mesmo algoritmo, Isard & Blake (1998b) e Liu & Jia (2004) combinam mapas de profundidade com contornos de mão e modelos conhecidos de transição entre posturas para rastrear, segmentar e classificar quatro posturas. Hamada, Shimada & Shirai (2004) expressam o modelo de transição através de um grafo, que chamam de rede de transição, e rastreiam um número maior de posturas conhecidas. Nessa rede há inclusive um nó "nenhuma postura", principalmente para momentos em que o movimento rápido da mão borra a imagem e prejudica a detecção de sua postura.

Jennings (1999) deriva um modelo probabilístico para permitir a combinação de diversas estratégias de segmentação e utiliza a cor da pele, profundidade, pontas dos dedos e bordas combinados para um rastreamento robusto dos dedos utilizando múltiplas câmeras.

Nguyen, Binh & Bischof (2008) fazem uso de características da imagem do tipo *Haar-like* (Viola & Jones, 2001) e árvores de decisão para a detecção da mão, uma estratégia similar é bastante utilizada para detecção e reconhecimento de faces. Já Triesch & Malsburg (2001) combinam a segmentação por cor com filtros de Gabor, filtros lineares cuja resposta a impulso é dada pela multiplicação de uma função harmônica e uma gaussiana e que demonstram propriedades similares ao córtex visual humano (Daugman, 1985).

Por fim, existem trabalhos em que a segmentação é feita através do uso de luvas ou marcadores coloridos, como nos casos de Bauer e Hienz (2000) e Davis & Sha (1994), respectivamente e até utilizando marcadores ativos, como LEDs (Marcel, 2002).

Outra função da segmentação que tem bastante importância para algumas estratégias de análise e reconhecimento e que deve ser importante neste trabalho é a separação entre a mão e o braço do usuário. Deimel & Schroter (1999) utilizam a transformada de distância da imagem (já binarizada para remover o fundo) para detectar o centro da palma e, com base na interseção entre círculos em torno desse centro e os outros elementos da imagem, determina onde deve ser feita a segmentação do braço. Amayeh et al. (2007) também utilizam a palma da mão como referência, mas usam as bordas da imagem para determinar qual elemento conectado a palma é o braço, pois assumem que esse cruza a borda da imagem, enquanto os dedos não. Yoon et al. (2006) fazem a segmentação com base no braço, e não na palma, encontrando seu maior eixo de simetria e procurando o ponto de mínima espessura ao longo desse eixo.

A Tabela 4 compara as principais abordagens para segmentação discutidas nesta seção.

Tabela 4 - Abordagens para segmentação

Abordagem		Particularidades
Marcadores		Simplicidade e precisão, presos ao usuário ou vestidos
Bordas		Precisa, exige modelo do objeto
Profundidade		Precisa, exige equipamento especial
Baseadas em cor da pele (mais usadas)		
Pixel a Pixel	Regras explícitas	Simplicidade, altas taxas de erros
	Não-paramétrica	Menores taxas de erros, grande consumo de memória
	Paramétrica	Grande necessidade de processamento
	Dinâmica	Calibração permite usar técnicas mais simples
Vizinhança		Maior necessidade de processamento

2.4 INTERAÇÃO 3D E GESTOS

Já que o principal foco deste trabalho é o reconhecimento dos gestos, a implementação de técnicas de interação e aplicações que utilizam estes gestos está além de seu escopo, exceto talvez de forma resumida para testar o sistema no contexto de aplicações específicas. É importante, porém, que nesta revisão da literatura se analise como os gestos são utilizados para interação, pois essas informações afetam os requisitos do sistema. As informações desta seção complementam as já apresentadas na introdução com essa finalidade.

As principais técnicas e tarefas de interação 3D associadas ao uso de gestos são baseadas, como já foi indicado, no envio de comandos. Gestos semafóricos têm uma estreita relação com comandos, mesmo na comunicação natural. Até mesmo para realizar outras tarefas, como por exemplo navegação, é comum que gestos sejam utilizados com técnicas que envolvem comandos para realizá-las. Por exemplo, para navegação podem ser utilizados comandos gestuais para "começar a mover para frente", "parar", "virar para a esquerda" e "virar para a direita". Gestos deíticos também são muito utilizados na seleção de objetos. Gestos em geral, quando utilizados juntamente com parâmetros como posição ou orientação da mão, também podem ser utilizados para a manipulação de objetos. Até mesmo para a entrada simbólica de pequenos números, gestos manuais podem ser utilizados de forma natural.

Já foi discutido que uma das principais desvantagens de interfaces baseadas em comandos, inclusive as gestuais, é serem uma "interface invisível", em que as opções não são continuamente apresentadas ao usuário. Há técnicas, no entanto, que permitem minimizar essa desvantagem.

Bau & Mackay (2008), por exemplo, apresentam o OctoPocus que determina (para gestos definidos por uma trajetória, de mão ou caneta, por exemplo), a cada momento e com base no contexto e na realização parcial de um gesto pelo usuário, qual é o conjunto de gestos que ainda é possível realizar como continuação do gesto parcialmente realizado e apresenta os elementos desse conjunto e a que ação estão associados graficamente ao usuário.

Lenman, Bretzner & Thuresson (2002) descrevem o uso de menus radiais, em vez de verticais ou horizontais, e que por isso podem ser acessados de forma mais eficiente mesmo sem a necessidade de feedback utilizando a orientação de uma postura de mão. Para usuários iniciantes o menu auxilia o aprendizado, mas os experientes conseguem realizar a tarefa sem a necessidade de desviar o olhar para o menu. Propõe inclusive que, após o usuário demonstrar uma certa eficiência ou escolher essa opção, o menu nem mesmo seja mostrado como padrão mas somente se for detectada hesitação do usuário durante a interação. Combinando posturas distintas, essa técnica pode ser usada para acessar inclusive menus hierárquicos de forma rápida. Lenman, Bretzner & Thuresson (2002) comentam que isso é especialmente útil em domínios para os quais não haja referências culturais de gestos com significados adequados para associar a ações, como por exemplo no controle de certos eletrodomésticos. Boussemart et al. (2004) apresentam uma interface com os mesmos menus radiais, aqui semitransparentes e utilizados com interação bimanual. Os menus são posicionados sobre objetos com a mão não-dominante e selecionados com a dominante, indicando uma operação que deve ser realizada sobre aquele objeto.

O uso dessas técnicas deve ser feito no projeto de interface específicas, enquanto um sistema de reconhecimento como o desenvolvido neste trabalho deve prover recursos para que isso seja possível. Enquanto o uso de menus radiais deve ser suportado pelo uso de parâmetros dos gestos neste trabalho, o reconhecimento de gestos incompletos, usado no OctoPocus, não está previsto em seu escopo.

Por fim, uma técnica de interação baseada em VC que tem sido utilizada com crescente frequência por diversos autores e até em aplicações comerciais e que tem

sido classificada como uso de gestos manuais é a interação utilizando *hotspots* (Jaimes & Liu, 2005). A técnica consiste em aumentar o espaço em que o usuário se vê (em geral em um monitor) com elementos de interface que são acionados quando a câmera detecta algum movimento sobre eles em coordenadas de imagem (e como o usuário se vê nessas coordenadas isso é simples), em geral utilizando fluxo ótico. Essa técnica também não apresenta a desvantagem das "interfaces invisíveis", visto que os *hotspots* são sempre visíveis para o usuário. Gonçalves et al. (2008) apresentam um exemplo desse tipo de aplicação, um jogo utilizado para acrescentar informações a imagens e Cabral et al. (2005) descrevem um *engine* para facilitar a criação de jogos usando essa técnica. Essa forma de interação, no entanto, não se adequa bem à definições de gesto manual geralmente adotadas, pois não exige uma configuração específica ou movimento específico da mão nem constitui a manipulação de um objeto. Nem mesmo exige que se use a mão para realizar o movimento.

Com relação a aplicações específicas de gestos em interação, já na revisão feita por Pavlovic et al. (1997) podem ser encontrados diversos exemplos, como sistemas para controlar e interagir com robôs (uma aplicação de interação com gestos que continua sendo bastante frequente até hoje), para controle de elementos bidimensionais de interfaces gráficas e de aparelhos eletrônicos como TVs e aparelhos de som. Mais recentemente, Bretzner, Laptev & Lindenberg (2002) também implementam controle desse tipo de aparelho com gestos estáticos (quatro posturas) e Yoon et al. (2001) utilizam gestos (definidos por movimentos e reconhecidos com HMMs) para interagir com um aplicativo bidimensional, um editor gráfico desenvolvido pelos autores. Ainda na revisão de Pavlovic et al. (1997) é também discutido o DigitEyes (Rehg & Kanade, 1993), baseado em um modelo 3D e ainda assim permitindo o uso, em tempo real, da mão como um mouse 3D. Um pouco mais tarde, Maggioni and Kammerer (1998) atualizam as informações que já haviam sido listadas por Pavlovic et al. (1997). sobre o GestureComputer, um sistema que reconhece seis posturas da mão através de momentos de imagem e das pontas dos dedos e permite interação em tempo real utilizando essas posturas como gestos estáticos e a posição da mão como parâmetro. O sistema reconhece também movimentos da cabeça.

O uso de gestos para interação 3D em ambientes virtuais também é bastante explorado na literatura, embora com menos frequência quando se trata de

reconhecimento por VC. O'Hagan et al. (2002) discutem a implementação e o uso de um sistema que utiliza gestos deíticos para a tarefa de seleção e posturas para manipulação de objetos (ou da própria cena, permitindo a navegação) no ambiente 3D. Boussemart et al. (2004) também usam gestos deíticos para selecionar e manipular objetos em um ambiente desse tipo, mas o fazem com o auxílio de menus, como discutido anteriormente. Chen et al. (2007) utilizam comandos para navegar em um ambiente virtual que contém objetos de aprendizagem. Nijholt & Heylen (2002) e Paggio & Jongejan (2005) oferecem alguns dos muitos exemplos do uso de gestos em conjunto com a fala em ambientes virtuais e Malkawi & Srinivasan (2004) descrevem seu uso em um ambiente aumentado.

Embora existam exemplos do uso de gestos em jogos eletrônicos na literatura, eles aparecem com menos frequência nas revisões publicadas sobre a literatura da área de gestos. Um dos primeiros sistemas desenvolvido para o uso de gestos em jogos (Freeman et al., 1996) é uma das poucas exceções a essa regra. O sistema reconhece posturas da mão e do corpo e seus autores sugerem algumas maneiras nas quais essas funcionalidades poderiam ser utilizadas em jogos. Como jogos são uma das principais aplicações do presente trabalho, é interessante analisar como gestos tem sido utilizado nesse domínio.

Starner et al. (2004) discutem o jogo MIND-WARPING, que combina interface tangível⁷ em uma mesa aumentada com uma interface móvel de realidade aumentada. Um ou mais jogadores posicionam "monstros" na mesa, objetos tangíveis que são identificados e mostrados em primeira pessoa ao jogador com a interface móvel, aumentando o ambiente real visto por ele. Esse jogador utiliza quatro gestos definidos pela posição das mãos em relação a seu corpo e capturadas por uma câmera presa à sua cabeça e direcionada para as mãos. Esses gestos, segmentados no tempo por um "grito de kung fu", são utilizados para lançar ataques contra os monstros virtuais em três alturas distintas (há três tipos de monstros e cada um ataca em uma altura) e para se defender.

Kim et al. (2005) usam marcadores fiduciais nas costas de uma das mãos para detectar sua posição e definem quatro gestos estáticos, baseados em quantos

⁷ Interface em que um objeto real e tangível é associado biunivocamente a um objeto virtual de forma que, minimamente, mudanças no objeto real provoquem as mesmas mudanças no objeto virtual (Ishii & Ullmer, 1997). O contrário é desejável mas nem sempre prático.

dedos são mostrados estendidos, para o controle do sistema em um jogo. Os gestos são reconhecidos como comandos de começar/recomeçar, pausar, selecionar/mover e mostrar o mapa.

O Camera Kombat (Paula et al., 2006) define um único gesto com a mão como um ataque dirigido contra o oponente. Esse gesto consiste em estender o braço na direção do oponente, o que é reconhecido pelo ângulo formado pela mão, cabeça e pelo centróide do corpo.

Tse et al. (2007) mostram a aplicação em jogos de um sistema multimodal que combina fala e gestos manuais e testam seu uso para interagir com dois jogos bastante conhecidos. Os gestos são modelados como um gesto estático ou como a transição de uma postura para a outra e a posição das mãos é usada como parâmetro na interação. O sistema permite o uso de ambas as mãos, por mais de um jogador e é implementado em uma mesa sensível a toques, mas os autores afirmam que seus produtos e resultados podem ser aplicados a outros cenários de detecção de gestos.

Figueiredo et al. (2009) descrevem uma framework de código aberto para jogos de *air guitar* que detecta as mãos do usuário, que deve usar luvas coloridas, e determina a posição de uma guitarra virtual, bem como a frequência e a intensidade dos sons gerados, com base nas posições e movimentos das mãos. O comportamento da "guitarra" pode ser configurado de várias formas. Cutler and Turk (1998) já descrevem um sistema baseado em fluxo ótico que permite interagir com personagens e instrumentos musicais virtuais.

Kang, Lee & Jung (2004) integram a um jogo bem conhecido uma interface baseada em gestos reconhecidos com VC. Embora utilizem principalmente gestos do corpo, levantam a importante questão da necessidade de gestos contínuos e suaves e buscam uma solução para a segmentação do gesto no tempo, em particular à detecção do momento inicial do gesto.

3 MODELO DE GESTOS

Como um dos principais requisitos e contribuições deste trabalho é possibilitar o reconhecimento de gestos escolhidos dentre um grande conjunto de possibilidades, para aumentar a probabilidade de se encontrar gestos com associações lógicas com as ações a que estão ligados na interface, o modelo de gestos adotado deve ser flexível e incluir diferentes tipos de gestos, como gestos estáticos, definidos como uma única postura mantida por um certo tempo (Quek, 1994), aqueles em que a posição da mão não muda, mas sua postura ou orientação sim, gestos em que mão se move sem mudança de postura e, por fim os que combinam movimento e mudança de posturas. O modelo deve prever ainda que todos esses tipos de gesto podem ter seus significados modificados pelo local em que são realizados, em relação ao corpo do usuário. Dessa forma, o modelo é capaz de descrever todos os tipos de gestos descritos na introdução e usados nos testes preliminares.

Além de propor um modelo que satisfaça estes requisitos, este capítulo descreve também uma maneira de classificar gestos descritos a partir dele. Como este modelo descreve uma linguagem formal regular para gestos, é natural utilizar um autômato finito (e determinístico), ou uma máquina de estados finitos, para implementá-lo.

3.1 MODELO

Com inspiração em abordagens linguísticas, usadas frequentemente no reconhecimento de fala (Derpanis et al., 2004), decidiu-se neste trabalho modelar o gesto não como uma entidade atômica, mas sim como uma combinação de elementos mais simples, e tirar vantagem desta decomposição para implementar o reconhecimento.

Os quatro elementos em que o gesto foi decomposto neste modelo provêm da análise revolucionária de línguas de sinais feita por Stokoe (1960), a partir da qual, há meio século, gestos não são mais vistos como unidades indivisíveis. Esses

elementos, para cada mão, são: sua postura, chamada por Stokoe de "dez", a orientação da postura, o movimento global da mão, ou "sig", e o local onde ocorre o gesto em relação ao gesticulador, ou "tab".

A segmentação dos gestos no tempo é simplificada neste modelo pela restrição que todos são delimitados por um conjunto de postura e orientação inicial e um final, mantidos por um intervalo de tempo configurável e, em geral, curto. Embora a hipótese da manutenção da postura seja comum para gestos estáticos e para interação (Quek, 1994), pode ser adotada também para gestos dinâmicos semafóricos, principalmente para evitar a detecção de uma postura durante a transição entre duas outras. O modelo admite que as posturas inicial e final sejam iguais.

É possível omitir a definição da orientação de uma ou de ambas as posturas, do movimento e do local onde ocorre o gesto e nesse caso ele é reconhecido independente desses elementos. Dessa forma, a posição e a orientação podem ser usadas também como parâmetros contínuos dos gestos, por exemplo para tarefas de manipulação no ambiente 3D, e não só para transmitir um significado discreto. Também é possível omitir uma das mãos, mas não ambas, na definição de um gesto, de forma que tanto gestos bimanuais como aqueles com qualquer uma das mãos podem ser descritos e reconhecidos. Mudanças de postura durante a realização do movimento em geral não tem significado (Liddel & Johnson, 1989) e são desprezadas neste modelo.

Assim, um gesto nesse modelo é definido como uma sequência de duas posturas e orientações diferentes ou iguais, com ou sem um movimento delimitado por elas. Além disso, na definição de um gesto pode-se, opcionalmente, especificar o local onde se inicia (e onde ocorre, caso não haja movimento). Tudo isso é válido para qualquer uma das mãos e um gesto bimanual pode combinar gestos distintos feitos com ambas. Esta definição é mostrada de maneira mais clara e formal, utilizando a notação BNF estendida (ISO/IEC, 1996). Um gesto realizado com uma das mãos é definido como:

```
gesto manual = postura, [orientação], [local], [movimento],
               postura, [orientação];
```

Mas um gesto pode ser realizado com a mão direita, a esquerda ou ambas ao mesmo tempo. Como na maior parte das notações de linguagens, inclusive a EBNF,

não se prevê a ocorrência de elementos simultaneamente mas sim em sequências, neste texto utiliza-se o sinal de adição (+) para denotar essa ocorrência simultânea de dois ou mais elementos, sem impor nenhuma exigência sobre a sincronia das partes de cada um. Assim, um gesto que pode ser realizado com uma das mãos ou com ambas simultaneamente é definido como:

```
gesto = gesto manual + [gesto manual];
```

Uma limitação deste modelo é que prevê gestos com somente um ou dois conjuntos de postura e orientação, enquanto muitos gestos na prática tem as mesmas postura e orientação iniciais e finais, mas uma postura ou orientação intermediária distinta, como por exemplo um aceno. Esta limitação, no entanto, pode ser contornada com poucas linhas de código que interpretem um conjunto de gestos como um só. Dessa forma, o gesto definido pelas posturas inicial, intermediária e final pode ser substituído pelo gesto com as posturas inicial e intermediária seguido de outro com intermediária e final. Outra opção é simplesmente definir o gesto como a postura intermediária seguida da final.

Deve-se perceber que, até o momento neste modelo, os elementos local, postura e movimento não são especificados, de forma que ele pode ser aplicado a diferentes conjuntos de gestos e elementos, com diferentes formas de discretizar locais ou definir movimentos, por exemplo.

No desenvolvimento deste trabalho, no entanto, a modelagem inclui também a definição de quantos e quais elementos utilizar para cada um desses componentes do gesto, ou seja, dos símbolos terminais.

O movimento pode ser definido como "nenhum", com o centro da mão aproximadamente na mesma posição, ou como um dentre doze movimentos básicos: esquerda, direita, para cima, para baixo, 4 diagonais e movimentos no sentido horário e anti-horário, todos no plano da imagem, além de para frente e para trás, perpendiculares a esse plano. Esses movimentos foram escolhidos por serem movimentos básicos, usados como tal não só em línguas de sinais (Stokoe, 1960), como também para gestos em dispositivos de interação por toque (Elias et al., 2007). Movimentos mais complexos são combinações desses, mas os testes preliminares mostram que esses movimentos mais complexos são difíceis de executar no caso de gestos com mãos livres, em vez de com uma caneta ou mesmo a ponta do dedo numa superfície, e acabam exigindo movimentos amplos e

cansativos dos braços, o que deve ser evitado (Nielsen et al., 2003). Utilizando a notação EBNF, esse conjunto de movimentos pode ser expresso como:

```
movimento = "frente" | "trás" | "cima" | "baixo" | "direita" |
            "esquerda" | "direitaCima" | "direitaBaixo" |
            "esquerdaCima" | "esquerdaBaixo" | "horário" |
            "antiHorário" | "nenhum";
```

Nos testes preliminares, mesmo sem restringir os gestos dos usuários de forma que cada mão não cruzasse para o lado oposto do corpo, isso ocorre em aproximadamente 98% dos casos. Por isso e para simplificar a análise dos gestos, neste trabalho opta-se por impor essa limitação. Além disso, os gestos feitos espontaneamente nos testes preliminares raramente se iniciam afastados lateralmente do corpo, mesmo no teste direcionado ao uso do CoH onde são realizados gestos amplos. Nielsen et al. (2003) explicam que isso é natural, por razões de ergonomia. A altura em que os gestos são realizados durante estes testes, no entanto por vezes modifica drasticamente o seu significado, principalmente no caso do CoH, onde a escolha de posturas é restrita. Por isso, neste modelo os locais em que os gestos ocorrem são discretizados em quatro regiões baseadas somente na altura: acima da cabeça, na altura dela, na altura do peito (estimada como sendo uma vez e meia a altura da cabeça, logo abaixo dela) e abaixo dele, o que pode ser expresso em EBNF da forma abaixo. É importante notar que, nesse texto, o termo local é usado de forma bastante específica, denotando um destes locais discretizados e relativos à posição do gesticulador, enquanto o termo posição é usado de forma mais indiscriminada, em geral indicando uma medida contínua.

```
local = "alto" | "cabeça" | "peito" | "baixo";
```

Dado o grande número de graus de liberdade da mão, ao contrário do número relativamente baixo de locais e movimentos possíveis no modelo, opta-se por permitir o uso de um número maior de posturas, sem limitá-las a um conjunto predeterminado. Para os testes realizados neste trabalho, utiliza-se o conjunto de 17 posturas mostrado na Figura 2, que ilustra os três tipos de componentes dos gestos neste modelo. Outras posturas, no entanto, podem ser facilmente acrescentadas ao

sistema. Nessa figura, os locais estão nomeados em inglês e as posturas são mostradas para a mão direita mas espelhadas, como o usuário as veria se utilizasse a tela do computador como um espelho ao realizar os gestos.

Figura 2 - Componentes do modelo de gestos

Vários critérios são utilizados para a escolha dessas posturas. Em primeiro lugar, todas foram sugeridas pelos usuários durante os testes preliminares. Pelo menos mais nove posturas figuram nesses testes, mas ou são espelhamentos dessas, ou são citadas somente uma vez, ao contrário dessas 17 que são usadas em vários gestos distintos, ou no mesmo gesto, mas repetido por diversos usuários. Nos testes preliminares, as posturas usadas com maior frequência e com diversos significados diferentes são as de 1 a 4, a 8 e a 11.

Além disso, as posturas de 1 a 7 são as escolhidas com maior frequência para uso em interação em testes realizados por Epps, Lichman & Wu (2006) com 20 usuários realizando 36 tarefas distintas. Juntamente com as posturas 1 e 2, as posturas de 8 a 10 completam representações dos primeiros cinco algarismos que, de acordo com Nielsen et al. (2003) e Bowman et al. (2005) são utilizadas com frequência muito grande em interfaces baseadas em gestos, não só para a representação simbólica de números mas para diversos outros fins.

A postura 11, além de representar um gesto de "positivo" muito conhecido também é associada a diversos outros significados no testes preliminares, como "pedir carona", "reprovado" (com o polegar voltado para baixo) e até mesmo como um comando para o CoH. Já a postura 12, além de ser utilizada com frequência nos testes preliminares, é um dos gestos mais conhecidos em todo o mundo,

representando a expressão americana OK, presente em todas as versões da língua inglesa e em um grande número de outras línguas (Read, 1964), ainda que em muitas delas o gesto equivalente também tenha significados obscenos. A palavra é inclusive utilizada com frequência para indicar confirmação em interfaces e o gesto é usada também em diversos domínios específicos como o aeronáutico, mergulho e, assim como a postura 11, como *emote*⁸ em diversos jogos.

As posturas de 13 a 15, além de terem sido utilizadas nos testes preliminares, foram inseridas neste conjunto para demonstrar a capacidade de acrescentar e usar posturas pertencentes a domínios específicos que podem ser (e de fato são) explorados em aplicações de entretenimento e educação, onde podem ser comuns e significativas mesmo se não o forem no cotidiano. Cada uma dessas três posturas é bastante comum e reconhecida, respectivamente, nos domínios de música roque, surfe e do popular universo fictício de Jornada nas Estrelas.

As últimas duas posturas foram incluídas neste conjunto principalmente para fins de teste. A postura 16, referida como "garra" por dois usuários durante os testes preliminares, é útil nos testes para comparação com a postura 2, principalmente porque em diversos algoritmos (por exemplo baseados na detecção das pontas dos dedos) essas duas posturas são confundidas.

Surpreendentemente, a postura 17 foi sugerida com frequência nos testes preliminares e diversos dos trabalhos analisados durante a revisão da literatura fazem uso dela, por exemplo para controle de robôs ou navegação em ambientes 3D, como no caso de Chen et al. (2007). Essa postura também é muito utilizada em línguas de sinais (em LIBRAS, por exemplo, representa a letra i e é usada em diversos sinais) e tem sido inclusive usada como gesto obsceno na Austrália. Mas o principal motivo para sua inclusão neste conjunto foi testar seu reconhecimento, comparado com o da postura 11, visto que dependendo de como essas posturas são executadas, podem gerar contornos bastante semelhantes.

As posturas 6, 7 e 12 são úteis ainda durante os testes para verificar o comportamento do reconhecimento com posturas em que o lado e não a palma da mão está voltado para a câmera.

⁸ Animação realizada pela representação do usuário no jogo para expressar uma emoção ou outros conceitos.

É importante notar que o modelo é indiferente à forma como cada postura é armazenada internamente para o reconhecimento. Assim, pode-se usar um conjunto de ângulos para descrever a articulação da mão, um conjunto de pontos de seu contorno ou uma descrição em XML. Utilizando a numeração mostrada na Figura 2, este conjunto de posturas pode ser expresso em EBNF da seguinte forma:

```
postura = "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
| "10" | "11" | "12" | "13" | "14" | "15" | "16" | "17";
```

Por fim, as diferentes orientações que essas posturas podem assumir no plano da imagem também são discretizadas neste modelo, para serem usadas na diferenciação entre gestos. De forma contínua, podem ser usadas como parâmetros de gestos. A discretização mostrou-se necessária pois, de acordo com Nielsen et al. (2003) os usuários tem grande dificuldade para utilizar posturas com uma orientação precisa e, caso isso seja necessário, precisam ser treinados em cada postura. Assim, e de acordo com o requisito de tolerar variações nos gestos, decidiu-se adotar uma tolerância de $\pm 15^\circ$, o que é equivalente a discretizar a orientação em intervalos de 30° . Ao definir um gesto, o usuário do sistema pode utilizar qualquer valor de ângulo como orientação, mas será adotada a orientação mais próxima no conjunto de orientações mostrado em EBNF abaixo:

```
orientação = "0°" | "30°" | "60°" | "90°" | "120°" | "150°" |
"180°" | "-30°" | "-60°" | "-90°" | "-120°" |
"-150°" | "-180°";
```

Mesmo com esse intervalo relativamente grande, ainda há 13 orientações possíveis para cada postura no plano da imagem. A maioria dessas inclinações, no entanto, exige esforço ou posicionamento desconfortável do braço e da mão para ser atingida e somente 3 ou 4 delas são utilizáveis na prática.

Ainda assim, assumindo uma média de 3 orientações para cada postura, e combinando-as com somente as primeiras 11 posturas mostradas na Figura 2 (as mais comuns), os 12 movimentos e 4 locais possíveis, esse modelo permite a definição de mais de 56 mil gestos para uma única mão (mais de 135 mil com as 17 posturas e mais de 316 mil com as 26 citadas nos testes preliminares). A possibilidade de utilizar gestos bimanuais aumenta ainda mais esse número. Mesmo que muitos desses gestos não tenham associação lógica com nenhuma ação ou

domínio óbvios, pode-se afirmar que este modelo simples, utilizando a abordagem linguística, atende o requisito de permitir a escolha de gestos em um grande conjunto de possibilidades, de dezenas de milhares, a partir de números muito menores de cada componente do gesto.

Uma limitação imposta ao modelo para implementação neste trabalho por seus requisitos é o uso de uma única câmera, convencional e de baixo custo. O uso de câmeras que forneçam mapas de profundidade foi descartado por, atualmente, não se tratar de equipamento de amplo acesso, e o uso de duas ou mais câmeras convencionais para obter informação 3D, por estereoscopia, reconstrução ou outros métodos, foi descartado para a versão atual porque, em geral, essas abordagens exigem posicionamento preciso e processos mais complexos de calibração. Assim o sistema fica limitado ao uso de entradas bidimensionais e, com base nos requisitos de baixo processamento e simplicidade, optou-se por não procurar inferir a configuração 3D da mão com base nesses dados. Por isso, a única rotação medida para as posturas é no plano da imagem, como mostrado na definição da orientação em EBNF, enquanto rotações da mesma postura em torno de outros eixos são interpretados como posturas distintas (essa é a diferença, por exemplo, entre 3 e 6 na Figura 2, uma rotação de 90° em torno de um eixo vertical). Além disso, os movimentos "para frente" e "para trás", perpendiculares ao plano da imagem, são estimados através de mudanças de área da mão, e não de sua posição como os outros. Por isso, atualmente, restringe-se gestos que usam esses dois movimentos a ter a mesma postura inicial e final (embora futuros trabalhos devam eliminar essa restrição usando áreas relativas das posturas). Por fim, algumas posturas diferentes geram imagens 2D muito parecidas devido a auto-oclusões e o sistema não pode diferenciar entre elas.

3.2 RECONHECIMENTO

O reconhecimento do gesto a partir deste modelo é controlado por uma máquina de estados com regras explícitas, ou um autômato finito determinístico, que faz uso de outros módulos para o reconhecimento de movimentos e de posturas. A opção pelo uso de uma máquina de estados decorre principalmente do fato do

modelo descrever uma linguagem formal regular para gestos e de estabelecer que o gesto é segmentado no tempo por posturas, o que naturalmente leva a alguns estados, como busca de postura inicial e final.

Além do modelo já descrito, que permite a descrição de diversos tipos diferentes de gestos através de uma abordagem linguística, pode-se considerar também esse uso de uma máquina de estados para combinar os reconhecimentos de movimento e postura como uma das principais contribuições deste trabalho.

Como já discutido, a maior parte dos trabalhos nessa área ou trata o gesto como um todo, o que limita o conjunto de gestos utilizáveis, ou se limita a reconhecer um único tipo de gestos, como por exemplo somente gestos estáticos ou somente movimentos. Mesmo nos raros trabalhos que combinam posturas e movimentos no modelo de gestos que reconhecem, essa combinação é feita de forma mais simples. Liu & Fujimura (2004), por exemplo, utilizam uma postura, um movimento e um local de forma independente em seu modelo. Salles et al. (2004) realizam principalmente o reconhecimento de posturas e, mesmo em seus gestos "ensaiados", simplesmente registram os segmentos de reta que foram traçados pelo movimento do usuário e os enviam para uma biblioteca de reconhecimento de caligrafia. Derpanis et al. (2004) propõe um modelo linguístico mas reconhecem somente o movimento.

A forma proposta aqui para combinar movimentos e posturas permite o reconhecimento de diversos tipos de gestos, não requer treinamento e não faz hipótese alguma sobre como é feito o reconhecimento de cada um dos elementos do gesto, exceto que a movimento é segmentado temporalmente por posturas definidas. Por isso, esta máquina de estados pode combinar estes elementos independente da estratégia usada para reconhecê-los, podendo ser aproveitada em outros trabalhos com características distintas.

A Figura 3 ilustra a lógica utilizada por esta máquina de estados finitos que classifica o gesto. Além dos estados inicial (nenhum candidato a gesto detectado, buscando posturas iniciais) e final (gesto reconhecido), ela tem 3 estados para cada mão: 1) postura, orientação e local iniciais encontrados, mas postura ainda não aceita, 2) postura e orientação finais encontradas mas ainda não aceitas e 3) classificação do movimento. A orientação não é citada na figura somente por questão de espaço.

Para que um conjunto de postura e orientação seja aceito pelo sistema, deve ser reconhecido durante um número n de quadros em sequência, o que reduz a influência de eventuais reconhecimentos incorretos de postura (que podem ocorrer, por exemplo, durante a transição de uma postura para outra ou devido a problemas na segmentação). Utiliza-se como padrão o valor de $n = 3$ que, a 15 quadros/s, é praticamente imperceptível. Se durante esse tempo outra postura ou orientação é reconhecida o sistema retorna ao estado inicial, onde não há candidato algum encontrado. É necessário determinar inclusive, durante futuros testes, como um valor maior para n , mais perceptível para o usuário e que aumenta a chance de erros do sistema, pode afetar sua usabilidade.

Figura 3 - Máquina de estados para o reconhecimento

Para cada mão, se é reconhecida por esse tempo mínimo uma das posturas e orientações iniciais de um dos gestos definidos como possíveis no contexto atual e

se, caso o gesto defina um local inicial para essa postura, o local esteja correto, o sistema aceita a postura inicial e esse gesto passa a ser o candidato atual ao reconhecimento. Caso o local esteja incorreto, busca-se na lista de gestos algum outro com a mesma postura inicial e outro local ou um local não definido. Se encontrado, esse gesto se torna o candidato atual e a postura inicial é aceita. Senão, retorna-se ao estado inicial.

Uma vez que uma postura inicial é aceita, as posições e a área daquela mão começam a ser enviadas para o reconhecimento de movimentos e o sistema espera encontrar um conjunto de postura e orientação final. Se for aceito um que não corresponde ao definido para o candidato atual a gesto (o primeiro da lista de gestos possíveis com a postura, orientação e local iniciais corretos), busca-se na lista de gestos possíveis algum outro que tenha essas propriedades definidas corretamente. Se não for encontrado nenhum, a máquina retorna ao estado inicial mas, se for, esse passa a ser o novo candidato e a postura final é aceita. Se o candidato atual não tem o movimento definido, é aceito como o gesto reconhecido e o processo recomeça.

Uma vez que a postura final é aceita e a mão não faz nenhum movimento acima de uma tolerância durante n quadros (pois o movimento pode continuar mesmo depois que a postura final seja reconhecida, até porque caso haja movimento, essa postura pode ser igual a inicial), o movimento é classificado como um dos doze possíveis. Se é o movimento definido para o candidato atual, esse é aceito como o gesto reconhecido e o sistema retorna ao estado inicial. Senão, busca-se dentre os possíveis um gesto com os mesmo local inicial e postura e orientação iniciais e finais e com o movimento correto, ou sem movimento definido. Se for encontrado, esse é aceito como o gesto reconhecido mas mesmo se não for, o processo recomeça.

Para gestos bimanuais, outra limitação do modelo é que não define nenhuma forma que indique a sincronicidade entre as posturas ou movimentos feitos por ambas as mãos. Assim, o sistema é incapaz de diferenciar, por exemplo, entre um gesto de empurrar para frente simultaneamente com ambas as mãos e um de empurrar separadamente, primeiro com uma e depois com a outra (exceto se o desenvolvedor criar lógica adicional, por exemplo definindo um gesto diferente para cada mão e usando seus parâmetros de posição). Assim, um gesto bimanual é aceito pelo sistema desde que as condições acima sejam satisfeitas para ambas as

mãos, com a única condição adicional que as posturas iniciais para ambas as mãos sejam aceitas antes que a postura final para qualquer uma delas o seja.

Esse conjunto relativamente complexo de regras para o reconhecimento mostra ainda outra limitação para a definição do conjunto de gestos possíveis em cada contexto: **se um gesto é definido como um único conjunto de postura e orientação inicial e final, seja em algum local ou em local indefinido, nenhum outro gesto deve ser definido usando a mesma postura e orientação iniciais naquele local.**

Caso contrário, se o gesto com uma única postura e orientação preceder qualquer outro com a mesma postura e local iniciais na lista de gestos possíveis, sempre será reconhecido antes que seja possível reconhecer o outro. Mesmo se for o último da lista, pode ser reconhecido antes que o usuário tenha chance de continuar a realizar o outro gesto com a mesma postura inicial a menos que o valor de n seja aumentado e o usuário treinado para executar a postura inicial rapidamente.

4 O SISTEMA GESTURES2GO

O sistema de reconhecimento de gestos para interação em ambientes 3D deve ser desenvolvido neste trabalho respeitando os requisitos apresentados na introdução, aplicando a metodologia descrita no mesmo capítulo e utilizando o modelo e a estratégia de reconhecimento descritas anteriormente. O desenvolvimento faz uso também das análises e decisões tomadas durante a revisão da literatura. Adotou-se para este sistema o nome de Gestures2Go que traduz o objetivo de facilidade de uso, tanto por desenvolvedores quanto por usuários.

Neste processo de desenvolvimento, a fase de construção descrita pelo processo unificado está dividida em 5 interações, cada uma contando com desenvolvimento de protótipos, integração, testes e refinamento. Na primeira interação é desenvolvido o módulo de segmentação de regiões de interesse e a segunda é responsável pela análise e reconhecimento de posturas das mãos. Na terceira interação é desenvolvido o reconhecimento de gestos estáticos ou baseados em mudança de posturas, já levando em conta o local do gesto, mas ainda sem reconhecimento dos movimentos. Este reconhecimento é desenvolvido somente na quarta interação. A quinta é uma interação de integração com os aplicativos de teste e também prevê algumas melhorias globais no sistema, como por exemplo a inclusão de mais padrões reconhecíveis de posturas de mão. Parte dos artefatos gerados com o Processo Unificado durante este desenvolvimento se encontra neste capítulo e parte pode ser vista no Apêndice A, juntamente com uma descrição mais detalhada da metodologia usada e do decorrer do projeto.

Ainda que seja possível identificar a cada momento toda a grande variedade de gestos cuja descrição o modelo permite, visto que são combinações de um número muito menor de elementos, a primeira simplificação utilizada na implementação do sistema é que é possível reduzir a dificuldade do uso do sistema, a quantidade de processamento e a taxa de erros do reconhecimento reduzindo o número de gestos reconhecidos em cada instante. Como a aplicação do reconhecimento de gestos nesse caso é em interação e, como já discutido, eles constituem uma interface "invisível", vários autores como Nielsen et al. (2003), Bowman et al. (2005) e Shneiderman et al. (1998) recomendam que, em cada

contexto de cada aplicação, limite-se o número de gestos para o usuário memorizar e usar, em geral a um número entre 5 e 10 gestos, ou 7 ± 2 , já que esse é um caso em que a lei de Miller (1956) se aplica ao número de gestos na memória de curto prazo. **Por isso, o sistema implementado permite a especificação, em tempo de execução, do subconjunto de gestos reconhecíveis.** Nielsen et al. (2003), além de fazer essas considerações, propõem inclusive um método experimental para determinar o melhor conjunto de gestos para uma determinada aplicação e contexto. Utilizar um subconjunto menor de gestos possíveis na interface não só facilita o aprendizado e a memorização dos gestos pelo usuário e reduz a possibilidade que gestos involuntários do mesmo sejam reconhecidos como uma ação, como também permitem a utilização de conjuntos menores de posturas ou de movimentos o que, principalmente no caso das posturas, pode reduzir a taxa de erros (ao reduzir o uso de posturas semelhantes) e o processamento consumido pelo reconhecimento.

Outra simplificação adotada nesta implementação, devido às características de uso previstas para o sistema, é a hipótese que a câmera está fixa durante o uso e que não está desalinhada horizontalmente.

O processo de reconhecimento se inicia com uma imagem 2D proveniente da câmera. Essa imagem é devidamente segmentada para mostrar somente as mãos e a face e então uma série de características é extraída durante a análise e usada no reconhecimento. Essas características são a posição e área da cabeça e, para cada mão, uma representação do seu contorno, sua posição e sua área.

Alguns parâmetros de gesto podem ser obtidos diretamente a partir daí: a posição, a rotação e a área da mão no plano da imagem, que são extraídas da imagem como características para o reconhecimento, e a velocidade do gesto, que também pode ser derivada facilmente a partir das posições. Outros parâmetros, no entanto, como direção de apontamento, são de determinação mais complexa ou mesmo impossível nesse sistema.

Com as características calculadas, faz-se o reconhecimento das posturas e do movimento, que alimentam a máquina de estados usadas para o reconhecimento do gesto como um todo e descrita em detalhe no capítulo anterior. De posse desta visão mais geral do processo, as outras etapas do reconhecimento são descritas em detalhe no restante deste capítulo.

4.1 ANÁLISE DA IMAGEM E CÁLCULO DAS CARACTERÍSTICAS

Uma das primeiras atividades durante a etapa de análise é determinar que regiões correspondem às mãos e à cabeça do usuário. O sistema assume que tem um único usuário, de forma que, após a segmentação por cor de pele, os três maiores componentes conexos detectados correspondem a suas mãos e face. É adotado também um número mínimo de pixels para que um componente conexo seja aceito como uma dessas regiões de interesse.

Não é necessário que sempre estejam presentes na imagem ambas as mãos e a face. Como padrão, se somente uma região que satisfaça esse requisito for encontrada, o sistema assume que é a cabeça e, se houver somente duas regiões, que é a mão esquerda que está faltando. O sistema permite que esta ordem de prioridades pode ser modificada. Em muitos dos testes realizados focados no reconhecimento de posturas, por exemplo, foi atribuída a uma das mãos prioridade maior que à cabeça. Para simplificar a identificação desses elementos, essa implementação assume que as mãos não podem cruzar para o outro lado do corpo, determinado por uma linha vertical que passa pelo centro da cabeça, conforme já foi discutido. Assim, eles podem ser distinguidos entre si simplesmente por sua posição

O próximo passo e o objetivo principal da análise, é calcular as características usadas no reconhecimento dos gestos. Além da posição da cabeça, um vetor de características F para cada mão, calculado na análise, contém as características escolhidas, necessárias a ambas as atividades. F contém um valor de ponto flutuante r com o ângulo que descreve a rotação da postura da mão, um vetor normalizado que contém uma descrição de seu contorno, a posição G de seu centróide e sua área a , como mostrado a seguir:

$$F = (r, \bar{D}, G, a)$$

$$G = (x_G, y_G)$$

$$\bar{D} = (d_1^2, d_2^2, \dots, d_i^2, \dots, d_n^2) / \max(d_i^2)$$

$$d_i^2 = (x_i - x_G)^2 + (y_i - y_G)^2$$

Para a determinação de $\bar{D}$, o contorno é reamostrado para que sempre tenha no máximo n pontos. Atualmente o valor de n foi estabelecido, um tanto quanto arbitrariamente, como 128. Com imagens de resolução de 640 x 480 pixels e os

tamanhos mínimos aceitos para as mãos, esse número se mostra pequeno o bastante para que não haja sobreposição dos pontos e para permitir um processamento rápido, e suficientemente grande para que não seja necessário preocupar-se em escolher pontos do contorno próximos a características de interesse (como mínimos e máximos locais, que normalmente representam as pontas dos dedos e os "vales" entre eles). Assim, os pontos (x_i, y_i) usados no cálculo das distâncias ao quadrado d_i^2 e portanto de $\bar{D}$, representam a posição, em coordenadas da imagem, do i -ésimo ponto desse contorno reamostrado. Além disso, no mesmo laço em que é feita a normalização desse vetor, ele também é rotacionado, de forma que o ponto do contorno mais distante do baricentro seja sempre seu primeiro elemento ($i=1$). Os pontos que o antecederam são simplesmente copiados no fim do vetor. A Figura 4 ilustra essa representação do contorno da mão, mostrando os 128 pontos, o baricentro e uma distância (a máxima) entre ele e um ponto do contorno.

Figura 4 - Contorno

A rotação r da postura também é obtida a partir desse ponto de máximo: é o ângulo formado entre ele e um eixo com a mesma direção e sentido do eixo vertical da imagem (em geral apontando para baixo) e que passa pelo centróide.

Definindo o momento de imagem de um componente conexo como

$$M_{ab} = \sum_{i=1}^n I_i \cdot (x_i - x_o)^a \cdot (y_i - y_o)^b, \text{ onde:}$$

(x_i, y_i) é a posição de qualquer um de seus pixels em coordenadas da imagem;

I_i é sua intensidade, que neste caso vale 1 para o componente e 0 para o fundo;

e (x_o, y_o) são as coordenadas de uma origem;

a posição do baricentro e a área do componente conexo que representa cada mão são obtidos através de seus momentos de ordem 1 e 0 da seguinte forma:

$$a = M_{00}$$

$$x_G = M_{10} / a$$

$$y_G = M_{01} / a$$

A Figura 5 mostra uma plotagem dos valores de três vetores para três posturas diferentes da mão direita (espelhada).

Figura 5 - Exemplos de vetores que representam contornos das posturas

Os maiores picos na Figura 5 representam as pontas dos dedos, e em geral há um pico menor que representa um ponto de máxima distância no contorno da palma.

Já a Figura 6 mostra o resultado da rotação feita sobre esse vetor no caso de duas posturas que diferem principalmente por um ângulo de aproximadamente 90° . A linha tracejada mostra o vetor para a postura da esquerda antes de seu ponto de máximo ser alinhado como primeiro ponto. A diferença no segundo maior pico dos vetores alinhados deve-se à diferença na posição dos polegares.

Figura 6 - Exemplo de rotação do vetor

4.2 CLASSIFICAÇÃO DOS MOVIMENTOS

Como explicado anteriormente, os movimentos são necessariamente segmentados no tempo por posturas e classificados entre doze possibilidades. Nos quadros entre a aceitação da postura inicial e a parada da mão na postura final, as posições de seu centróide e sua área são enviadas para o módulo de reconhecimento do movimento.

Este reconhecimento é feito por uma árvore de decisão simples, de três níveis e com regras explícitas. Para estes doze movimentos básicos, esta implementação não necessita armazenar e comparar todos os pontos da trajetória percorrida pela mão, como fazem diversos autores, inclusive Mammen et al. (2002), mas somente

os valores máximos e mínimos de área e da posição horizontal (x) e vertical (y) da mão, além da ordem em que ocorrem. Com estes dados, é possível obter um conjunto simples de regras para a classificação do movimento, da seguinte forma.

No primeiro nível, a árvore classifica o movimento em uma dentre quatro classes possíveis, com base em sua característica dominante. Se o movimento vertical for muito maior que o horizontal, trata-se da classe **vertical**. O movimento é classificado como **horizontal** de forma análoga. Se tanto o movimento vertical quanto o horizontal são significativos o movimento é misto. Se a postura da mão não mudar mas sua área mostrar mudança significativa (aumentar ou diminuir mais que uma proporção constante, o padrão é 50%), o movimento é classificado como em **profundidade**, independente dos outros movimentos, já que esse tipo de movimento em geral é acompanhado por um movimento linear do centro da mão no plano da imagem, devido ao posicionamento da câmera em relação ao usuário.

O segundo nível da árvore é utilizado somente para dividir o movimento misto em mais duas classes. Se pares de máximos e mínimos de x e y acontecem em instantes próximos, trata-se de um movimento em **diagonal**, senão o movimento é classificado como **circular**.

Uma vez descoberta em qual destas cinco classes o movimento se encontra, basta analisar a ordem em que ocorreram os máximos e mínimos de x e y ou de área para determinar, nas folhas da árvore, qual dentre os doze movimentos foi executado. A Figura 7 ilustra esta árvore de decisão.

Figura 7 - Árvore para classificação dos movimentos

4.3 RECONHECIMENTO DAS POSTURAS

As posturas das mãos são reconhecidas por uma abordagem similar à ES, discutida no capítulo dois, ou seja, o conjunto de características obtidos a partir de imagens de mãos reais é comparado, usando o critério de máxima verossimilhança, com características de imagens sintetizadas a partir de um modelo de mão. Dessa forma, comparando somente propriedades de imagens 2D, sem manipular os parâmetros do modelo 3D durante o reconhecimento, é possível estimar a configuração espacial da mão, já que os parâmetros usados no modelo são conhecidos. A versão atual do sistema, no entanto, não faz uso desse fato.

Ao contrário do que ocorre na maioria dos métodos ES, no entanto, não se procura rastrear a postura continuamente diferenciando entre posturas com somente pequenas variações entre si. Visto que tolerância para variações nas posturas é um dos requisitos do sistema, é não somente aceitável como desejável que essas pequenas diferenças nas posturas sejam desprezadas, caso contrário seria necessário processamento adicional para obter a tolerância desejada. Sendo assim, esta implementação pode deixar de lado uma das maiores complicações de métodos ES. Em vez de precisar classificar a postura da mão em universo de milhares de possibilidades usando estratégias complexas de busca, só precisa fazê-lo dentre um pequeno número de posturas possíveis e pode percorrer todo o universo a cada passo, visto que, em cada momento, somente de cinco a dez gestos, cada um com no máximo duas posturas, devem estar disponíveis para o usuário. Se nenhuma correspondência aceitável for encontrada entre a imagem real e uma dessas posturas possíveis, o sistema pode simplesmente assumir que o usuário não está realizando um gesto de comando naquele momento. A desvantagem desta estratégia, comparada aos métodos ES, é que se perde a robustez do rastreamento de postura.

Também diferentemente de outros métodos ES, não é feito o alinhamento de todos os máximos e mínimos locais entre os vetores $\bar{D}$ da mão real e das imagens sintetizadas. Somente o máximo global (que também precisa ser encontrado para a normalização) é alinhado, sempre na primeira posição do vetor.

Assim, a postura i considerada como melhor correspondência para a postura da mão na imagem real, dentre todas as possíveis no momento, é aquela em que,

dado o conjunto $P = \{\overline{D}_1, \overline{D}_2, \dots, \overline{D}_i, \dots, \overline{D}_m\}$ de características que representam o contorno das m posturas possíveis, minimiza a distância entre $\overline{D}$ e $\overline{D}_i$, desde que a diferença entre a orientação r dessas duas posturas seja menor que a tolerância pré-definida (15° é o padrão) ou que a orientação não esteja definida para $\overline{D}_i$. Na prática, é utilizado o erro quadrático médio q ao invés da distância entre vetores, para evitar o cálculo de raiz quadrada e para que essa medida independa do número de pontos n usado no contorno. Também é adotada uma tolerância máxima para esse erro.

A expressão abaixo é utilizada para a determinação de q :

$$q_i = \frac{1}{n} \cdot \sum_{j=1}^n (d_j^2 - d_j^{i2})^2$$

Onde d_j^{i2} denota o elemento j do vetor $\overline{D}_i$ e n continua a ser o número de elementos desses vetores. Caso uma ou ambas as tolerâncias, para q ou r , seja ultrapassada para todos os elementos de P , o reconhecimento de postura retorna que nenhuma foi encontrada.

O valor de tolerância para q , além de ter um valor padrão (de 2%) pode ser especificado em tempo real, para permitir sua calibração em função do conjunto de posturas usado em cada momento. Quando esse conjunto inclui somente posturas bastante diferentes, como por exemplo as posturas 1 a 4 da Figura 2, essa tolerância pode ser aumentada, reduzindo o número de falsos negativos sem causar um aumento de falsos positivos. Já quando o conjunto de posturas possíveis inclui duas ou mais posturas semelhantes, como por exemplo as posturas 11 e 17 na mesma figura, a redução desta tolerância reduz o número de falsos positivos, ainda que possa aumentar o número de falsos negativos, principalmente na presença de problemas de segmentação.

O punho fechado, que não tem um ponto de distância máxima bem definido, tem o reconhecimento particularmente sensível à posição do polegar que, estando mais para fora ou para dentro do punho, pode definir esse ponto. Esse problema para o alinhamento dos vetores dessa postura, no entanto, têm seu impacto reduzido simplesmente associando a ela uma orientação não definida quando usada no conjunto de gestos possíveis. A normalização do vetor sem um valor máximo bem claro acaba por magnificar pequenas variações, como os nós dos dedos, que

podem estar desalinhadas mas mesmo assim essa postura ainda é bem reconhecida pelo sistema, desde que uma orientação não seja especificada para ela.

É interessante notar que esta estratégia para o reconhecimento de postura tem algumas similaridades com a utilizada no reconhecimento de trajetórias representando símbolos do reconhecedor de "um dólar" de Wobbrock et al. (2007). Em ambos os casos a reamostragem e a rotação dos vetores é importante, mas é mais simples no caso de Gestures2Go devido a quase sempre haver um ponto de máximo bem definido, o que ainda permite a diferenciação de posturas somente pelo ângulo, o que é impossível para os símbolos do algoritmo de "um dólar". Da mesma forma, enquanto Gestures2Go normaliza o vetor de distâncias, o reconhecedor de "um dólar" faz uma mudança de escala não linear do símbolo para que se encaixe em um quadrado de tamanho fixo, ambas estratégias para tornar o reconhecimento independente de escala.

4.4 INTEGRAÇÃO COM OUTROS SISTEMAS

O sistema implementado inclui uma interface responsável por traduzir uma descrição de gesto, resultante do reconhecimento, como uma entrada para outro sistema, como por exemplo um sistema operacional ou um *engine*⁹ de jogos. Essa interface permite ainda a interação multimodal, pois pode traduzir outros tipos de descrição, provenientes de outros sistemas, como por exemplo de reconhecimento de voz ou de um tapete de dança¹⁰.

Para tanto basta que a interface forneça duas funcionalidades: a associação de descrições a entradas e o envio de uma descrição, indicando que deve ser traduzida. Nesta versão do sistema foi desenvolvida somente uma implementação desta interface, que traduz uma descrição qualquer como um evento de entrada do Microsoft Windows. Nesta implementação deve-se utilizar os métodos

⁹ Conjunto de componentes de software integráveis e reutilizáveis que fornecem serviços utilizados em um jogo eletrônico, como renderização, simulação física etc. (Bernardes et al., 2004)

¹⁰ Controlador com oito botões geralmente acionados com pisadas.

`associate()` e `sendDesc()`. O método `associate()` recebe uma descrição na forma de *string* e a qual evento de entrada básico deve ser associada, dentre tecla abaixada, levantada ou apertada (abaixada e levantada), movimento do mouse e botão do mouse abaixado, levantado, clicado ou duplo-clique, especificando ainda a tecla (ou combinação de teclas) ou botão. O método `sendDesc()` simplesmente recebe a descrição e gera o evento associado, que é enviado para todas as janelas.

Além disso, devido à importância do *feedback* para o usuário, que já foi discutida, o sistema disponibiliza também para o desenvolvedor, além da descrição e dos parâmetros do gesto reconhecido, o acesso a diversas das imagens utilizadas durante o processamento do gesto, para que qualquer uma possa ser utilizada para fornecer *feedback* visual ao usuário. Essas imagens incluem a original, capturada pela câmera e espelhada em relação a uma reta vertical para facilitar o *feedback*, uma imagem colorida com o fundo removido, a imagem binária resultante da segmentação e uma imagem binária proveniente da análise, mostrando somente as mãos e a cabeça.

4.5 SEGMENTAÇÃO

A segmentação das mãos e da cabeça (utilizada como referência para determinar o local do gesto) na imagem não é o foco deste trabalho, mas é necessária. Para encontrar esses elementos, este trabalho opta por utilizar um modelo paramétrico de segmentação pixel a pixel baseada em cor da pele, binarizando a imagem através de um filtro passa-banda no tom e saturação do espaço de cores HSV. Como discutido no capítulo dois, nesse espaço há um sub-espaço do tom de formato simples que contém os valores de cor de pele para um grande número de pessoas e condições de iluminação. A saturação é utilizada principalmente para remover regiões muito claras ou escuras, onde o tom não está bem definido e que, portanto, podem resultar em falsos positivos nesta segmentação. Uma desvantagem desta abordagem é que a conversão do formato

RGB, utilizado pela maioria das câmeras, para o HSV consome tempo considerável, mas isso é compensado pela simplicidade da filtragem¹¹ dos pixels.

Este módulo adota valores médios e de tolerância fixos como padrão para o tom e a saturação da pele. Testes preliminares sob condições distintas de iluminação, em diferentes ambientes ou usando diferentes câmeras, no entanto, mostraram grandes variações nesses valores para as imagens capturadas, principalmente por diferenças na iluminação ou no algoritmo de ajuste de branco, ou *white balance* (Viggiano, 2004), executado automaticamente pelas câmeras e que, na maior parte dos dispositivos de menor custo, não pode ser desativado. Foi estabelecida então a necessidade de implementar métodos para acumular e calcular a média e o desvio padrão do tom e da saturação de uma ou mais áreas retangulares que podem ser definidas arbitrariamente. Isso permite um rápido procedimento de calibração que permita a amostragem de regiões de pele para que a segmentação utilize valores mais adequados na filtragem dos pixels.

Ainda durante os testes com protótipos do módulo de segmentação, falhas graves de segmentação ao testar em um ambiente cujas cores, quase em sua totalidade, tem tons muito próximos ao da pele (justamente o laboratório onde se desenvolve esta pesquisa), gerando muitos falsos positivos na abordagem simples de segmentação escolhida. Para minimizar esse problema, a opção de armazenamento e remoção de um fundo fixo e conhecido foi adicionada ao sistema, acrescentando mais um rápido procedimento de calibração. Considera-se que a hipótese de fundo conhecido é aceitável nesse caso, uma vez que o sistema já prevê o uso por somente um usuário e em ambiente doméstico, para aplicações de entretenimento e educação, contexto no qual um fundo praticamente fixo é muito mais comum que, por exemplo, em aplicações de computação móvel.

Mesmo com uma tolerância de cor de 50 num espaço RGB de 256x256x256 e sem mudança alguma na cena, durante os testes com protótipos uma grande parcela dos pixels, posicionados aparentemente de forma aleatória em cada quadro, não correspondiam ao valor do fundo armazenado. Esse problema é bastante reduzido simplesmente aplicando, após a remoção de fundo, uma operação de erosão com um *kernel* quadrado de 3x3 pixels.

¹¹ Filtragem é usada aqui no contexto de um filtro passa-banda, não de filtragem bayesiana.

Outras dificuldades encontradas para a segmentação nesse ambiente foram fontes de iluminação direta muito intensas, que causam reflexões especulares fortes na pele e efeitos de *bloom* (principalmente entre os dedos), e a ausência de paredes ou outros objetos brancos, o que torna bastante instáveis os algoritmos automáticos de *white balance* utilizados pelas câmeras. O módulo de análise e reconhecimento é capaz de lidar, até certo ponto, com as reflexões especulares, que são ignoradas exceto quando ocorrem sobre as bordas dos objetos de interesse, situação menos comum. O efeito de *bloom*, no entanto, sempre degrada a visão das bordas dos objetos. Mas como em geral acontece quando a mão é posicionada entre a câmera e uma fonte de luz, muitas vezes pode ser evitado pelo usuário. O usuário também pode contornar o problema de falta de regiões brancas, caso não seja possível desativar o *white balance* da câmera, simplesmente acrescentando essas regiões à cena, por exemplo usando uma camisa branca (um artifício útil mas longe do ideal, que no entanto pode ser necessário devido à limitação do *hardware*).

Foram testadas algumas alternativas ao uso deste filtro passa-banda para segmentação, também baseadas somente na cor dos pixels, como as misturas de gaussianas utilizadas por Jones & Rehg (1999) ou as regras inferidas por Gomez & Morales (2002) ou Gomez (2002). Nesses casos a remoção de fundo tornou-se menos necessárias, pois mesmo com diferentes condições de iluminação e ajustes de câmera foram detectados menos falsos positivos. O número de falsos negativos, no entanto, aumentou visivelmente, a ponto de prejudicar drasticamente o reconhecimento.

Não é de surpreender que a abordagem simples utilizada por este módulo de segmentação tenha desempenho superior à desses métodos mais complexos visto que, diferentemente deles, pode utilizar informações bastante específicas de fundo e cor de pele para uma câmera e condições específicas, obtidas durante uma rápida calibração. Vezhnevets et al. (2003) comentam que é comum que modelos para situações específicas em geral têm desempenho superior a modelos mais genéricos. É possível que utilizando abordagens similares a esses autores, mas retreinando seus sistemas para um grande número de imagens de pele obtidas por câmeras de baixo custo em diferentes condições (pois originalmente não foram treinados dessa forma), fosse obtido um método adequado, mas considera-se que esse esforço está fora do escopo deste trabalho.

O primeiro passo da análise de imagem, que seleciona os maiores componentes conexos na imagem como as mãos e a cabeça, também reduz o ruído restante após a segmentação e parte das mudanças no fundo.

O sistema, atualmente, não segmenta a mão do braço, o que impõe a limitação de que os usuários devem usar mangas compridas durante a interação. Alternativamente, se os gestos de uma determinada aplicação independem do local e, portanto, a cabeça não precisa ser localizada, o usuário pode optar por usar luvas com uma cor uniforme, já que o sistema permite que a cor da mão seja determinada a partir de amostras na calibração (considera-se inviável usar essa estratégia para segmentação da cabeça por não se esperar que os usuários, para interagir com o sistema, vistam máscaras). Isso pode inclusive reduzir problemas como fundos inadequados ou reflexos especulares, mas traz algumas das desvantagens do uso de luvas, citadas na introdução. Assim, embora exista, essa possibilidade não foi testada. A necessidade das mangas compridas é considerada uma limitação séria da implementação atual do sistema.

Embora existam métodos bem mais robustos para rastrear as mãos, além de serem mais complexos (inclusive dada a necessidade de reinicialização automática, discutida anteriormente), em geral só determinam a posição da mão, enquanto Gestures2Go precisa também de seu contorno. Outros métodos, como o Condensation (Isard & Blake, 1998b) rastreiam a posição e algumas variações em uma postura ou um pequeno conjunto delas, que no entanto precisam ser especificadas (no caso do algoritmo de Condensation, por exemplo, através de conjuntos de *splines* que representam seu contorno e ainda de um modelo das probabilidades de transição entre as posturas). Isso é pouco prático nesse caso, visto que as posturas da mão devem sofrer intensas variações durante a interação, representando diferentes gestos, e que as probabilidades de transição variam com cada conjunto de posturas e gestos usado. Sem informação sobre o formato do objeto de interesse, no caso a postura da mão, não é trivial utilizar informações de bordas para auxiliar a detecção ou o rastreamento.

A Figura 8 resume e ilustra, para uma das mãos, diversas etapas deste processo de reconhecimento de gestos implementado neste trabalho e discutido até o momento neste capítulo. A imagem capturada é espelhada e, durante a segmentação, tem um fundo conhecido removido, sofre uma erosão e é binarizada com base em um modelo paramétrico de cor de pele. A imagem segmentada é

utilizada pela análise, que encontra as mãos e a face do usuário e extrai as características do gesto, utilizadas no módulo de reconhecimento para fazer sua classificação combinando postura, movimento e local. A descrição do gesto reconhecido, bem como possíveis descrições provenientes de outros sistemas, são traduzidas pelo último módulo, para servirem como entradas para ambientes 3D ou outras aplicações.

Figura 8 - Processo de reconhecimento

4.6 FRAMEWORK

O sistema de reconhecimento descrito acima está implementado sobre um *framework*¹² mais genérico para reconhecimento de gestos, também desenvolvido como parte deste trabalho, que divide o reconhecimento em componentes que realizam a aquisição, segmentação e análise da imagem, além da classificação do gesto e que permite substituir facilmente qualquer um desses quatro componentes. O estudo de novas estratégias de reconhecimento e interfaces baseadas em gestos, além do que foi feito neste trabalho, é de grande interesse para a continuidade desta

¹² Nesse trabalho utiliza-se a definição de Johnson & Foote (1998) para *framework* de *software*: um projeto reutilizável para um sistema de *software* que pode ser representado por um conjunto de classes abstratas, com sua interface bem definida, que devem ser instanciadas para realizar a implementação de uma aplicação ou API específica.

pesquisa em trabalhos futuros, motivo pelo qual optou-se por criar este *framework* mais geral, com base na pesquisa de soluções alternativas para problemas similares analisadas durante a revisão de literatura. O *framework* deve ser útil não só para explorar o uso de gestos em outros domínios e aplicações como também para testar e comparar, com base em critérios como robustez ou tempo de processamento diferentes soluções e algoritmos para cada um de seus componentes, em diversos domínios.

Cada uma das atividades em que o reconhecimento de gestos geralmente se divide corresponde a um módulo nesta arquitetura. O modelo dos gestos está na classe `G2gGesture`. `G2gCapture` provê uma interface para capturar imagens 2D de uma ou mais câmeras ou sequências de vídeo pré-gravadas (usadas para testes, por exemplo). As imagens devem ter as mesmas dimensões, mas não necessariamente o mesmo número de cores. Um dispositivo pode, por exemplo, fornecer uma imagem colorida da cena e uma imagem em escala de cinza representando um mapa de profundidades. Sobre essas imagens é que é feita a segmentação, cuja interface é definida por `G2gSegmentation`. `G2gAnalysis` e `G2gRecognition` definem as interfaces para as classes que implementam a análise e o reconhecimento dos gestos, respectivamente.

O fluxo de informações pelo sistema em cada passo de tempo é o seguinte. Uma ou mais imagens servem de entrada para o módulo de captura de imagens, que as disponibiliza como um objeto `IplImage` do OpenCV (OpenCV, 2010). Esse formato de imagem equivale ao formato BMP (pixels armazenados em um vetor, sem compactação e com canais de cores entrelaçados) e foi escolhido não só pela comodidade em implementações que usem o OpenCV mas também pela velocidade de acesso a seus elementos.

O módulo de segmentação faz uso dessas imagens e provê um objeto da mesma classe (e tamanho, mas não necessariamente número de cores) contendo a imagem segmentada.

Baseada nessa imagem, a análise fornece um objeto `G2gFeatureCol`, uma coleção de características que, por sua vez, é usada pelo reconhecimento para, por fim, se obter um gesto como saída.

A Figura 9 mostra esse fluxo de informações em um diagrama de atividades da UML 2.2 que representa o modelo de fluxo de objeto da *Framework* de Gestures2Go.

Figura 9 - Modelo de fluxo de objetos do *framework*

`G2gFeatureCol` é uma coleção de objetos do tipo `G2gFeature`. Cada um desses objetos contém uma *string* de identificação que descreve a característica e um vetor de valores de ponto flutuante ou uma imagem. O vetor de valores, ou um único valor, são usados mais frequentemente, mas armazenar as características em uma imagem pode ser útil, por exemplo, para trabalhar no domínio da frequência. `G2gFeature` já define diversos identificadores, para as características utilizadas mais frequentemente na literatura de reconhecimento de gestos, visando facilitar a interface entre módulos de análise e reconhecimento desenvolvidos separadamente. O desenvolvedor pode, no entanto, utilizar o sistema com identificadores definidos por si mesmo.

`Desc2Input` é um módulo opcional que acompanha mas é na verdade distinto de `Gestures2Go`. Ele é responsável por facilitar, de uma forma bastante simples, tanto a integração com aplicações ou sistemas que não precisam estar cientes de `Gestures2Go` quanto interação multimodal. Esse módulo simplesmente traduz as entradas que recebe, que são descrições (como um identificador numérico, uma *string* ou uma descrição em XML, por exemplo), para outro tipo de entrada, por exemplo para um sistema operacional ou gerenciador de interface (como por exemplo um evento de *"key down"* ou *"mouse movement"*) ou para outro sistema, como um *engine* de jogos. As entradas de `Desc2Input` podem ser descrições de gestos fornecidas por `Gestures2Go` ou vindas de qualquer outro sistema, como por exemplo um sistema de reconhecimento de voz ou das entradas de um tapete de dança, e aí reside a possibilidade de interação multimodal. Uma implementação desse módulo para Microsoft Windows é descrita na seção 3.3.

Como esta arquitetura é formada em grande parte por interfaces, é possível criar uma única classe que, através de herança múltipla, implemente toda a funcionalidade do sistema. Isto em geral é considerado uma má prática em orientação a objetos e evitá-la é inclusive uma das razões por que a agregação é considerada preferível à herança (Eckel, 2003). Existem padrões de projeto (*design patterns*) que poderiam ter sido usados para forçar o uso de agregação e impedir a herança múltipla, mas esta arquitetura opta por permiti-la por uma razão. O reconhecimento de gestos pode ser bastante custoso em termos de processamento e, para ser usada como forma de interação, deve ser realizada em tempo real. Muitos algoritmos podem ser melhor otimizados para velocidade ao realizar mais de uma atividade (por exemplo, segmentação e análise) em conjunto. Além disso, a análise e o reconhecimento estão fortemente acoplados em alguns algoritmos e forçar sua separação poderia fazer com que o problema de reconhecimento se tornasse ainda mais complexo. Portanto, ainda que em geral seja recomendável evitar o uso de herança múltipla e implementar cada atividade em um classe distinta, facilitando a troca de módulos ou seu desenvolvimento em paralelo, `Gestures2Go` mantém a possibilidade de utilizá-la por boas razões.

Por fim, todas as classes neste sistema devem implementar os métodos `init()` e `cleanup()`, preferíveis ao uso dos operadores `new` e `delete` (o

sistema está implementado em C++) para evitar problemas com herança múltipla e sincronização (Eckel, 2003).

5 TESTES E RESULTADOS

Ao longo do desenvolvimento do trabalho e principalmente em suas etapas finais, foram realizados seis tipos de teste com o sistema. No primeiro, visando reduzir o acoplamento entre a segmentação e o reconhecimento de posturas para testar a estratégia adotada, foram feitos testes estáticos deste reconhecimento com 96 imagens de mão já segmentadas, incluindo variações para diversas posturas. O segundo tipo de teste foi interativo, utilizando um aplicativo que permite a calibração do sistema, exibe a postura detectada (se estiver entre as possíveis) e permite o envio de comandos para outras janelas. Esse aplicativo foi usado para testar todos os tipos de gestos e o uso de seus parâmetros. No terceiro tipo de teste esse mesmo aplicativo foi usado para avaliar a integração com outros sistemas e a precisão do uso da posição da mão como parâmetro para permitir a manipulação de objetos no *desktop*. A velocidade de diversas etapas do reconhecimento foi medida no quarto tipo de testes. Como já foi comentado, esses primeiros quatro testes foram utilizados constantemente durante o desenvolvimento, como forma simples de teste de regressão, mas os resultados desses testes durante o processo de desenvolvimento não foram, em sua maior parte, registrados. No quinto foram realizados os testes mais interessantes, a integração e o passo-a-passo cognitivo com outros aplicativos. Esses testes são descritos em maior detalhe nas subseções seguintes.

O último teste, que também se relaciona com as posturas, mediu simplesmente a dificuldade de se adicionar novos elementos ao conjunto de posturas conhecidas. Um usuário sem experiência com o sistema adicionou a esse conjunto 3 posturas: uma com somente os dedos indicador e médio estendidos e as posturas 12 e 16 (OK e garra) mostradas na Figura 2. Foram necessários menos de 10 minutos por postura para manipular o modelo 3D de mão utilizado, gerar figuras de referência e rodar um executável sobre elas para incluir suas características ao conjunto de posturas conhecidas.

5.1 TESTES PRELIMINARES

Como já foi mencionado, antes mesmo do início do desenvolvimento foram realizados testes preliminares com usuários de diferentes perfis (jogadores, desenvolvedores, professores e alunos) para verificar seu comportamento ao utilizar gestos principalmente para transmitir comandos sem o auxílio da fala, através de uma câmera com posição conhecida e sem a capacidade de determinar diferenças de profundidade

O primeiro destes testes, realizado com quatro usuários, consistiu simplesmente em pedir que fossem feitos gestos para a câmera (que estava desligada e serviu somente como ponto de referência) descrevendo o comando ou significado do gesto com uma ou duas palavras. Além do posicionamento da câmera, a única limitação tecnológica explicada aos usuários foi a impossibilidade de determinar diferenças de profundidade nas posturas, de modo que deveriam evitar aquelas para as quais isso fosse importante, como por exemplo apontar na direção da câmera.

Durante este teste, os quatro usuários listaram, no total, 62 pares distintos de gesto e significado e utilizaram 26 posturas diferentes da mão (desprezando mudanças de orientação no plano da imagem). Mais de um significado foi atribuído ao mesmo gesto em diversos casos e também houve casos em que um mesmo significado foi representado por dois gestos diferentes. As descrições dos gestos, posturas e significados obtidos neste teste estão consolidados no Apêndice A, mas seus principais resultados são relatados abaixo.

Dos 62 pares de gesto e significado registrados, a maior parte, 51 gestos, foi realizada com somente uma das mãos e 11 gestos, quase 18% do total, utilizaram ambas. Todos os gestos realizados com ambas as mãos utilizaram posturas específicas (em somente dois casos a postura foi diferente) e movimentos com ambas as mãos (6 com as mãos se afastando ou se aproximando, 2 com as mesmas girando em torno de um ponto entre elas e 3 em que se chocavam).

Dentre os gestos realizados com uma única mão, a maior parte, 29 gestos, consistiram simplesmente na apresentação e manutenção por um certo tempo de uma postura estática da mão, indicando que o reconhecimento de gestos estáticos é importante. Um total de 11 gestos consistiram de uma postura específica e um

movimento simples (para frente, para trás, para a esquerda, para a direita e movimentos circulares). Somente 5 envolviam uma postura mudando de orientação no plano da imagem, 4 foram definidos por uma postura inicial e uma postura final distinta e 2 utilizaram duas posturas distintas além de um movimento simples da mão. A Figura 10 ilustra estes resultados. Além disso, 5 dos gestos especificavam um local em que precisava ser realizado. Curiosamente, todos os usuários utilizaram a mão direita para realizar todos estes gestos, ainda que um deles fosse canhoto.

Figura 10 - Gestos com uma das mãos nos testes preliminares

Tanto para os gestos com somente uma das mãos como para os bimanuais, em somente um gesto uma das mãos mudou de lado em relação ao corpo do usuário, ou seja, imaginando um plano perpendicular ao plano da imagem e que divide o usuário em duas metades aproximadamente simétricas, uma das mãos cruzou esse plano em somente um dos 62 gestos (assumindo que nos 3 gestos bimanuais em que as mãos se chocaram esse plano foi tocado mas não ultrapassado). Esse fato é interessante dado que essa não foi uma limitação imposta aos usuários neste momento.

Os dois testes preliminares seguintes foram mais direcionados aos dois aplicativos descritos anteriormente e cada um foi realizado com somente dois usuários.

Para o ambiente interativo de iluminação, foram listados comandos para abrir e fechar uma janela (fonte de luz direcional) ligar e desligar três outros tipos de fontes de luz (iluminação indireta, um abajur e uma fonte de luz pontual representada por um lustre e é responsável pela maior parcela da iluminação no

ambiente) e também para selecionar o algoritmo de iluminação (*flat*, Gouraud ou Phong, em ordem crescente de qualidade). Coincidentemente, ambos os usuários optaram por numerar as luzes e os algoritmos da mesma forma: a luz principal como 1, o abajur como 2 (um dos usuários justificou essa escolha já que o abajur gera um cone de luz e que associou isso a um V formado pelos dedos indicador e médio), a luz indireta como três e os algoritmos também numerados de 1 a 3 em ordem crescente de qualidade. Um usuário sugeriu que simplesmente indicar o número da fonte de luz deveria mudar seu estado (de ligado para desligado ou vice-versa) e sugeriu que os números das luzes e dos algoritmos fossem diferenciados pela posição (com os algoritmos sendo sinalizados numa posição alta, com a mão acima da cabeça do usuário). O segundo usuário sugeriu que sinalizar com a mão direita o número de uma fonte de luz seguido da mão aberta a acendesse, e de um punho fechado a apagasse. Quanto ao número do algoritmo a ser usado, sugeriu que fosse sinalizado com a mão esquerda. Para abrir e fechar as janelas, ambos os usuários sugeriram movimentos da mão direita com o punho fechado (para os lados ou verticalmente, dependendo do modelo da janela). Mesmo nesse caso, com relativamente poucos comandos, foram sugeridos gestos que consistem de somente uma postura, de duas posturas, de uma postura e um movimento e do local em que o gesto é realizado.

No caso do CoH, foi selecionado um personagem arbitrariamente, com um conjunto de poderes ou ações associados a gestos e não ao uso de armas. O personagem conta com um grande número dessas ações (mais de 30), mas somente 11 que são utilizadas com frequência durante o jogo. As animações do personagem realizando cada uma dessas 11 ações foram mostradas a dois usuários e pediu-se que descrevessem que gestos fariam para representar as ações. Não foi surpresa que, nesse caso, ambos os usuários optaram por repetir os movimentos do personagem. Como já foi explicado, no CoH o número de posturas de mão é bastante limitado, portanto todos os gestos sugeridos envolveram movimentos. 3 dos 11 utilizam somente uma das mãos enquanto os 8 restantes utilizam ambas e em 4 dos 11, o local onde o gesto se inicia faz parte de sua definição. Em dois casos em que pares de ações tinham animações semelhantes, foram sugeridas posturas diferentes para distinguir os gestos. É interessante notar como a limitação no número de posturas aumenta a necessidade do uso de ambas as mãos e dos locais dos gestos, além dos movimentos. Comparando estes três testes preliminares

descritos até agora, também é possível notar como a frequência dos diferentes tipos de gestos varia em aplicações e domínios distintos.

Nestes três primeiros testes preliminares, foram sugeridos pelos usuários 93 gestos distintos, no total.

O último dos testes preliminares foi especificamente direcionado para analisar o uso dos movimentos da mão e foi realizado com três usuários. Pediu-se que realizassem, com uma das mãos e de frente para uma câmera desligada, usada como referência, uma série de 35 movimentos (com quantas pausas para repouso desejassem): os 9 algarismos com exceção do zero, 10 letras de forma que podem ser traçadas com um movimento contínuo (B, J, L, M, N, P, R, S, V e Z), 10 movimentos lineares (para frente e para trás, para cima e para baixo, para os lados e quatro diagonais), 2 quadrados, 2 triângulos e 2 movimentos circulares (as três formas feitas tanto no sentido horário como no anti-horário). Por fim, pediu-se aos usuários que classificassem os movimentos como fáceis, médios ou difíceis de realizar. Todos os usuários classificaram os 10 movimentos lineares como fáceis. Dois classificaram os movimentos circulares e aqueles formados por dois segmentos de reta (1, 7, L e V) como médios e justificaram dizendo que precisaram fazer movimentos mais amplos, ou mais demorados, enquanto o terceiro usuário classificou os mesmos movimentos, assim como o J, como fáceis. Todos os outros movimentos foram classificados como difíceis (e criticados como demorados) pelos três usuários. O algarismo 4 foi citado por dois deles como o pior movimento (ambos traçaram o algarismo como um h rotacionado de 180°, enquanto o terceiro o traçou como mostrado acima e considerou o R como o movimento mais difícil).

5.2 TESTES ESTÁTICOS DE RECONHECIMENTO DE POSTURAS

Este é o primeiro tipo de teste que foi realizado, após os testes preliminares, antes mesmo que a funcionalidade de segmentação estivesse completada ou que alternativas fossem pesquisadas para ela. Foi usada, assim, para testar diferentes tipos de características da imagem e formas de compará-lo, não só para permitir sua escolha como também para testar a viabilidade da estratégia selecionada, baseada em propriedades da imagem e não em modelos 3D e utilizando uma busca pela

postura com máxima verosimilhança. Além da forma de descrição do contorno que foi implementada neste trabalho, foi testado também o uso de Momentos de Hu, que se mostrou pouco preciso para diferenciar entre diversas posturas, e a correlação entre fases da transformada de Fourier das imagens binarizadas da mão, como descrito por Terrillon et al. (2002), que além de consumir mais processamento, forneceu resultados piores que o método adotado (mas melhores que o uso dos momentos). Nestes testes estáticos iniciais, utilizou-se um conjunto de duas imagens para cada uma dentre 12 posturas, binarizados e segmentados manualmente utilizando um pacote de edição de imagens.

Considerou-se útil manter este teste atualizado mesmo depois da implementação da segmentação para, como já foi discutido, permitir a análise do reconhecimento de posturas com menor influência da segmentação. Foram geradas mais 4 imagens de teste para cada uma das 12 posturas iniciais e 6 imagens para as 5 posturas adicionadas posteriormente (com uma exceção), totalizando 96 imagens, que foram capturadas binarizadas e segmentadas automaticamente usando a mesma estratégia implementada no sistema, já descrita. Essas imagens foram propositadamente geradas com pequenas diferenças na forma de executar as posturas, inclinações das mesmas e a diferentes distâncias da câmera, produzindo imagens de diferentes dimensões. O momento em que foram capturadas, no entanto, foi determinado manualmente, permitindo a escolha de imagens com segmentação sem grandes falhas. As imagens também foram recortadas manualmente, embora isso não tenha influência no reconhecimento. A Figura 11 mostra alguns exemplos de imagens utilizadas nestes testes, mantendo a proporção entre suas dimensões.

Figura 11 - Exemplos de imagens usadas nos testes estáticos

Já durante a fase de captura das imagens para o conjunto final de 17 posturas apresentado na Figura 2, decidiu-se por abolir o uso da postura 7. Ao tentar gerar imagens de teste para essa postura, ao menos da forma como está modelada, foi detectada uma dificuldade muito grande, mesmo para usuários experientes, para alinhar a mão corretamente com a câmera em dois eixos simultaneamente, de forma que nem as costas ou palma da mão, nem a parte de cima ou de baixo dos dedos (aproximadamente perpendiculares à palma), fossem vistos na imagem, modificando consideravelmente o perfil da mão. Após a geração de somente um par dessas imagens, com grande esforço, decidiu-se abolir seu uso. Uma postura semelhante que pode ser utilizada com a mesma finalidade e sem os mesmos problemas é mostrada na Figura 12, estendendo somente o polegar e o indicador em vez de todos os dedos. Essa alternativa, no entanto, não foi incluída nos testes. Essa dificuldade em alinhar a postura com a câmera também existiu, em grau muito menor, para a postura 6.

Figura 12 - Alternativa para a postura 7

Com essas imagens capturadas, foi criado um aplicativo de testes no qual são definidos dezesseis gestos estáticos, com cada uma dessas posturas como inicial e final, orientações e local inicial não definidos e sem movimento. Este aplicativo lê cada uma das imagens pré-segmentadas de postura, as utiliza como entrada para os módulos de análise e reconhecimento e registra seu erro quadrático médio q para cada uma das posturas. É importante que a orientação não seja definida para permitir a comparação com todas as posturas.

Nos 6 testes estáticos para cada uma das dezesseis posturas restantes, somente 5 posturas foram identificadas incorretamente. Em dois deles, a postura 17 foi identificada como a 11 (curiosamente, o contrário nunca ocorreu) e em um terceiro caso, com a postura 1. Esse último erro não ocorreria se tivesse sido levada em conta a orientação das posturas. Em uma ocasião a postura 6 foi identificada

como a 1 e, por fim, houve um teste em que a postura 5 foi identificada como a 13. A taxa de erro nestes testes, portanto, foi de 4.2%. Somente uma das imagens, da postura 16, não foi identificada como postura alguma.

Tabela 5 - Exemplo de reconhecimento estático de posturas

	1	2	3	4	5	6	8	9	10	11	12	13	14	15	16	17
1	0.36	10.04	3.65	28.98	5.58	2.20	6.69	7.21	8.88	1.41	6.13	3.44	3.70	6.35	14.28	0.75
2	8.30	1.18	7.69	17.54	10.42	10.66	11.75	8.92	4.59	6.53	5.21	7.65	7.60	8.20	4.75	6.81
3	2.62	9.44	0.67	15.32	5.16	2.93	6.53	7.86	9.00	1.80	4.66	3.80	4.34	5.05	8.28	2.16
4	13.50	10.23	6.98	1.09	10.80	15.53	12.56	11.54	11.10	9.68	10.08	11.07	11.46	8.41	3.98	11.20
5	2.20	9.34	4.38	24.61	1.36	5.04	1.99	7.05	8.70	1.43	8.35	1.45	4.15	3.56	11.50	1.42
6	2.09	10.29	3.64	26.26	6.56	1.82	7.19	7.63	9.40	1.89	5.43	3.67	4.76	6.69	13.68	2.09
8	6.09	9.55	9.40	22.07	4.79	12.47	1.53	7.30	9.32	4.55	10.91	2.94	6.20	2.89	11.34	4.80
9	7.70	8.58	8.35	19.20	5.22	11.83	5.53	1.67	7.18	5.46	6.74	7.84	8.77	7.12	8.67	6.74
10	8.29	4.96	8.52	16.97	5.93	12.29	6.58	6.36	1.85	5.78	8.33	7.09	8.36	8.60	8.11	6.79
11	1.71	4.84	2.91	17.10	2.31	5.13	2.88	4.40	4.53	0.66	4.71	0.89	3.01	2.40	6.71	0.86
12	4.70	8.47	3.91	23.35	8.77	5.10	10.01	7.35	8.04	3.81	1.13	7.31	6.20	9.59	10.37	3.85
13	1.74	7.86	5.04	24.39	3.01	5.91	3.21	7.59	7.16	1.60	8.06	0.43	4.43	3.44	11.35	1.47
14	2.26	7.05	2.87	21.55	5.79	2.94	7.51	4.87	5.67	1.59	4.86	3.37	0.23	3.28	10.33	1.68
15	6.20	7.27	6.44	14.94	5.20	9.49	6.49	5.12	6.53	3.96	8.60	4.17	3.66	1.05	8.96	4.84
16	15.97	11.93	10.16	8.29	12.09	19.59	13.99	13.46	11.64	12.05	9.72	12.69	11.90	10.17	6.63	13.19
17	1.34	6.33	2.55	16.72	2.59	4.22	3.06	4.49	5.77	1.08	5.13	2.23	3.92	2.91	7.35	0.87

A Tabela 5 exemplifica um destes seis testes, justamente aquele em que houve o falso negativo. Nesta tabela, cada linha representa uma imagem de teste para a postura que tem o mesmo número que a linha, e cada célula dela mostra o erro q , em porcentagem, calculado para essa imagem em relação a cada uma das posturas, numeradas nas colunas. Em negrito estão destacados os valores de erro mínimo de cada linha, que indicam qual postura (a coluna) é identificada para aquela imagem.

Nesta tabela todas as posturas estão identificadas corretamente, mas o erro para a postura 16, mostrado na célula com fundo cinza, tem um valor alto, maior que os 2% utilizados por padrão como tolerância. Esse erro é ainda mais alto ao comparar essa imagem com todas as outras posturas, o que sugere que a imagem utilizada para este teste é a fonte do problema. De fato, essa imagem, a terceira da esquerda para a direita mostrada na Figura 11, inclui uma região relativamente grande de braço não segmentado da mão (pode ser visto no canto inferior direito) que causa esse aumento de erro, mas ainda assim permite a identificação correta da postura, desde que se aumente a tolerância.

A Tabela 5 também mostra algo que ocorreu com frequência nos testes. Ainda que as posturas tenham sido identificadas corretamente, diversas outras

posturas tiveram um valor de erro abaixo da tolerância padrão, que já é relativamente estrita. Isso indica que, em um subconjunto de posturas possíveis que não incluísse a postura correta, a execução dessa postura pode gerar um falso positivo, sendo identificada incorretamente como outra postura quando deveria ser simplesmente ignorada. O próximo tipo de teste, no entanto, mostra que, na prática, isso ocorre com menos frequência, pois é raro que seja necessário um conjunto tão grande de posturas possíveis para um mesmo contexto de interação e o uso da orientação, ignorada nestes testes estáticos, elimina boa parte desses falsos positivos.

5.3 TESTES COM INTERAÇÃO

Para a realização do segundo tipo de testes com o sistema, desde o momento em que as primeiras versões da segmentação e do reconhecimento de posturas estavam implementadas, foi desenvolvido um aplicativo para testes interativos do sistema. Este aplicativo foi atualizado a cada nova funcionalidade incluída no sistema para permitir testes interativos dessas funcionalidades.

Primeiramente este aplicativo foi utilizado para testar o reconhecimento de posturas individuais com a mão direita e depois para o reconhecimento de gestos, sem e com movimentos e, por fim, bimanuais. Estes testes permitiram ainda a verificação visual da influência da segmentação e de diferentes condições de iluminação e fundo no reconhecimento. Durante alguns destes testes, as tolerâncias para r e q foram aumentadas para facilitar a identificação de falsos positivos nas posturas, e verificou-se que ocorrem principalmente na transição entre elas.

Este aplicativo de teste permite, através de comandos de teclado, que o usuário realize duas rápidas etapas de calibração: o armazenamento do fundo conhecido e a amostragem de regiões com cor de pele. Um dos testes realizados foi justamente para medir o tempo gasto com este processo de calibração. A amostragem da cor de pele é feita desenhando um retângulo vermelho na tela sobre o qual o usuário deve posicionar as regiões de pele que deseja amostrar. Apesar da aplicação permitir que isso seja feito diversas vezes, durante os testes a amostragem de uma única região ou, dependendo do usuário e da iluminação, de

duas (a palma e a parte de trás da mão) foi suficiente, inclusive para detectar a posição da face. Não houve ganho perceptível em aumentar o número de regiões amostradas.

A aplicação permite ainda, como *feedback*, três formas de visualização: em cores, inicialmente sem e depois com o fundo removido, ou em escala de cinza mostrando somente as regiões de interesse, cada uma em um tom de cinza. Ambas as formas se mostraram bastante úteis. A imagem colorida com o fundo removido permite identificar rapidamente mudanças no fundo detectadas pela câmera, mas a imagem monocromática é mais útil para visualizar as mãos e posturas e como é feita sua segmentação. Nesse modo, a aplicação exibe o nome (atribuído arbitrariamente) de uma postura quando identificada pelo sistema. Quando um gesto é realizado, seu nome (também arbitrário) é enviado para um console. As imagens rotuladas como "erosão" e "postura" na Figura 8 são, na verdade, regiões de telas capturadas desta aplicação.

5.3.1 Teste da calibração

A Figura 13 mostra o processo utilizado nestes testes para fazer a calibração necessária para a segmentação. Sua finalidade foi determinar o tempo gasto, em média, com esta tarefa.

Figura 13 - Calibração

Primeiramente o usuário deve se posicionar fora do campo de visão da câmera e pressionar a tecla b para gravar o fundo conhecido. Em seguida pressiona a tecla de espaço para exibir na tela o retângulo que indica a região onde será amostrada a pele, posiciona regiões de pele sobre esse retângulo e pressiona a

tecla i para incluir essa região na amostragem. Por fim, pressiona novamente a tecla de espaço para indicar o término da calibração.

Três usuários foram instruídos a realizar este teste exibindo duas regiões de mão, a frente e as costas, para amostragem. Um deles já tinha experiência com o sistema, mas pediu-se que os outros dois praticassem em duas tentativas antes que seus tempos para mais duas tentativas fossem registrados. O tempo medido vai do instante em que o usuário grava o fundo até o instante que finaliza a amostragem da pele. A Tabela 6 mostra os resultados deste teste. Em média o processo demorou 9.1s durante estes testes, com desvio padrão de 2.2s.

Tabela 6 - Teste da calibração

Usuário	Tempo (s)	
	Teste 1	Teste 2
1	10,9	9,0
2	5,7	7,3
3	12,2	9,5

5.3.2 Segmentação e posturas

Para este teste foi utilizado um conjunto de 16 gestos, com a mesma postura inicial e final, usando todas as posturas mostradas na Figura 2 para a mão direita, exceto a sétima, e orientação igual a zero, ou seja, usando a mesma orientação indicada na figura. Pediu-se aos usuários que simplesmente realizassem as dezesseis posturas na ordem que desejassem para verificar sua percepção qualitativa da qualidade do reconhecimento.

Este teste foi o que foi realizado no maior número de ambientes e com o maior número de usuários, oito no total, dentre eles três do sexo feminino, dois com tom de pele parda e três de descendência asiática. Buscou-se essa variedade de usuários para verificar o efeito de diferenças de tom de pele e do formato da mão na segmentação e no reconhecimento de posturas e não houve diferença perceptível no reconhecimento de posturas entre nenhum dos usuários, com uma exceção. Para um usuário em particular, no ambiente com as piores condições de iluminação e fundo, o polegar da mão direita era consistentemente separado da mão na

segmentação, impossibilitando o reconhecimento de diversas posturas. Não foi possível determinar a causa desse problema.

Se não houve variação perceptível em função dos usuários, o reconhecimento se mostrou drasticamente afetado pelo ambiente usado nos testes. Foram utilizados três ambientes e duas câmeras diferentes em cada um, ambas permitindo desativar o controle de exposição mas não o *white balance*, mas a escolha da câmera também não afetou os resultados. O primeiro ambiente tem um fundo com cor uniforme, branca e brilhante. Além disso, tinha fontes de luz muito intensas. No segundo o fundo contém diversos objetos, alguns brancos e inclusive estantes de madeira com regiões de tom bastante similar a tons de pele, mas a iluminação é suave. O terceiro ambiente é o pior caso. Não conta com nenhuma região branca, o que desestabiliza o algoritmo de *white balance* das câmeras, é quase que inteiramente pintado em tons de pele (a diferença entre o tom do ambiente e o da pele de um dos usuários, por exemplo, é de 1,7%) e tem muitas fontes de luz muito fortes que não só causam uma série de reflexos especulares na pele como também efeitos de *bloom* entre os dedos.

Os resultados destes testes são somente qualitativos, visto que a análise quantitativa das taxas de erros envolveria classificar manualmente as posturas em cada quadro de vídeos dos testes (e nem todos estes testes foram gravados) para comparação com as posturas reconhecidas. Além disso, logo no início dos testes percebeu-se que uma análise quantitativa simples das taxas de erros, nesse caso, é de pouca utilidade para avaliar o uso do sistema em interação, visto que muitos erros detectados durante os testes, mas que ocorrem somente por um ou dois quadros e principalmente durante transições entre posturas, são percebidos raramente pelos usuários e tem pouco ou nenhum impacto na interação.

O reconhecimento de postura teve seu melhor desempenho no segundo ambiente, indicando que a iluminação suave é mais importante que um fundo uniforme para a estratégia de segmentação utilizada. Neste ambiente, houve um usuário que não percebeu nenhum erro no reconhecimento, ainda que uma análise do teste gravado mostre diversos quadros em que a postura não foi reconhecida devido a problemas visíveis de segmentação e quadros de reconhecimento de falsos positivos durante a transição entre duas posturas, mas esses erros em geral ocorreram em quadros individuais separados. Os outros três usuários que fizeram o teste neste ambiente perceberam "piscadas" na imagem como erros na

segmentação e no reconhecimento, mas classificaram esses erros como "existentes mas praticamente imperceptíveis".

No primeiro ambiente os erros de segmentação e reconhecimento foram mais perceptíveis e três dos quatro usuários que realizaram os testes neste ambiente se queixaram da necessidade de ajustar a postura para que "a imagem da mão aparecesse inteira na tela". Na verdade estavam ajustando a posição da mão para reduzir as reflexões especulares na pele das intensas fontes de luz. Ainda assim, e principalmente com esses ajustes permitidos pelo *feedback*, todos os usuários classificaram o reconhecimento como "correto na maior parte do tempo" e estimaram que, em média, o reconhecimento ocorreu corretamente em 80% a 90% do tempo do teste (estimativa provavelmente pouco precisa já que, como no caso anterior, diversos erros de reconhecimento ocorreram durante somente um ou poucos quadros, tornando sua percepção difícil).

Os resultados no terceiro ambiente foram bem menos positivos. Três dos cinco usuários conseguiram, após um certo tempo de experimentação, encontrar posições em que nem o fundo nem as fontes de luz prejudicavam muito o reconhecimento, mas ainda assim classificaram o reconhecimento como "correto em mais da metade do tempo". Dois estimaram que o reconhecimento estava correto entre 70% e 80% das vezes enquanto o terceiro estimou entre 60% e 70%. Para os outros dois usuários, no entanto, o reconhecimento foi classificado como "correto em menos da metade do tempo" e não foi estimada uma porcentagem de acerto do algoritmo. Um deles tinha seu polegar consistentemente separado da mão pela segmentação.

Principalmente no terceiro ambiente, pode-se perceber ainda que, com uma das duas câmeras usadas nos testes, em alguns casos uma região ao redor deles aparece destacada do fundo, como uma "aura" (que pode ser vista na Figura 8), mesmo quando com os objetos de interesse suficientemente longe das fontes de iluminação e do fundo para não modificar sua aparência. Isso provavelmente se deve a correções locais feitas pela câmera. Essa região inclusive cresce em direção aos cantos da imagem caso a mão se aproxime deles.

Três dos oito usuários comentaram ainda que as posturas com a palma voltada para a câmera são mais fáceis e rápidas de realizar "corretamente", pois consomem menos tempo e esforço que alinhar o lado da mão com a direção de visualização da câmera.

A Figura 14 mostra uma tela do aplicativo para um dos usuários realizando o teste no primeiro ambiente.

Figura 14 - Teste interativo para posturas

5.3.3 Gestos sem movimento

Estes testes foram realizados por somente 2 usuários e, assim como todos os testes descritos a partir de agora, ocorreram no segundo ambiente descrito na seção anterior. Utiliza quatro gestos. O primeiro (No), tem como postura inicial o indicador apontando para cima e como final a mesma postura, mas com orientação de mais 30° (no sentido anti-horário). O segundo gesto (Click) tem a mesma postura inicial que o primeiro e tem como postura final o punho fechado. O terceiro (Bye) é igual ao primeiro, exceto que a postura usada é a mão aberta ao invés do indicador levantado. Por fim, o quarto gesto (Stop) tem as mesmas postura e orientação iniciais e finais, a mão aberta com os dedos unidos. Todas essas posturas são feitas com a mão direita e com a palma voltada para a câmera. Ao detectar o gesto, o aplicativo imprimia sua descrição em um console, separado da janela que mostra o *feedback* para o usuário.

Os dois usuários realizaram dois de cada um dos três primeiros gestos, nessa ordem. O último gesto encerrava o aplicativo de teste e por isso só foi realizado uma vez por ambos. Não houve nenhum erro no reconhecimento desses 14 gestos

executados, mas em um dos quatro gestos Click, houve uma demora no reconhecimento da postura final e, portanto, do gesto, que foi percebida e comentada pelo usuário. A Figura 15 mostra dois momentos próximos do final de um destes testes, em que a mudança de orientação da postura com a mão aberta é interpretada como o gesto chamado "Bye" e esta palavra é enviada a uma janela.

Figura 15 - Gestos sem movimento

5.3.4 Movimentos

Assim como o teste anterior, os movimentos foram testados por somente 2 usuários e imprimem o nome do gesto reconhecido em um console. Foi definido um gesto usando cada um dos doze movimentos básicos reconhecidos pelo sistema, todos segmentados por uma postura inicial de mão aberta e uma final de mão aberta com os dedos unidos. Pediu-se aos usuários que realizassem todos os doze movimentos, na ordem que preferissem. Todos os 24 movimentos executados nestes testes foram reconhecidos corretamente.

A Figura 16 mostra um destes testes em realização, mostrando o reconhecimento do movimento para baixo realizado entre a imagem da esquerda e a da direita. Curiosamente, a imagem da direita foi capturada durante a transição entre duas posturas, por isso nenhuma é identificada nela.

Figura 16 - Teste dos movimentos

5.3.5 Orientação

O aplicativo de testes interativos também foi utilizado para verificar o uso da orientação das posturas como parâmetro. Para tanto, foi modificado para exibir essa orientação, em graus, em vez de um nome para a postura sempre que uma das possíveis fosse reconhecida. A Figura 17 mostra um momento deste teste.

Figura 17 - Teste da orientação

Por se tratar de um teste em que não se espera haver variação entre os usuários, foi realizado por somente um deles para verificar esta funcionalidade. Foram definidos somente dois gestos para uso neste teste, um usando a postura com o indicador estendido mostrada na figura, e outro com a mão aberta como postura. Em ambos os casos a orientação foi exibida corretamente.

5.3.6 Gestos bimanuais e local inicial

Assim como no teste anterior, este teste foi realizado por um único usuário, para verificar o reconhecimento de gestos bimanuais pelo sistema, além da especificação do local onde o gesto se inicia. Faz uso da mesma funcionalidade de imprimir os nomes dos gestos detectados em um console usada nos testes de movimentos e de gestos sem movimento.

Com base na observação dos testes preliminares, decidiu-se por incluir movimento em todos os gestos definidos para este teste. Foram definidos quatro gestos, todos com posturas iniciais e finais iguais:

- as duas mãos abertas e os dedos unidos, afastando-se horizontalmente na altura do peito;
- igual ao anterior mas com as mãos se aproximando;
- as duas mãos em forma de punho fechado, a direita iniciando sobre a cabeça e descendo e a esquerda iniciando abaixo do peito e subindo
- a mão direita fica aberta com os dedos unidos e parada na altura da cabeça, enquanto a esquerda se move para frente com o punho fechado, sem local definido.

Todos estes gestos foram detectados corretamente durante o teste. A Figura 18 mostra o final deste teste.

Figura 18 - Gestos bimanuais e local inicial

5.4 MANIPULAÇÃO DE OBJETOS NO *DESKTOP*

Ainda que a principal aplicação de Gestures2Go deva ser a entrada de comandos em ambientes 3D, foi criado um aplicativo de teste para avaliar o uso de parâmetros dos gestos para outras tarefas e de Desc2Input para enviar eventos. Para deixar mais claro esse envio e também para facilitar o teste, tanto do ponto de vista de desenvolvimento quanto do ponto de vista do usuário, escolheu-se a tarefa simples e familiar de seleção de objetos no *desktop* 2D do MS Windows. Três usuários, também no segundo ambiente de teste, executaram essa tarefa usando gestos e forneceram *feedback* qualitativo.

Uma postura foi associada ao movimento do ponteiro do *mouse* e, enquanto reconhecida, mudanças na posição da mão na imagem capturada (e mostrada ao usuário) são mapeadas de forma absoluta para movimentos desse ponteiro. Como a mão se move somente em uma parte de uma imagem com 640 x 480 pixels e a mão em uma região de 1024 x 768, foi utilizado um mapeamento linear não-uniforme

entre esses movimentos, multiplicando os movimentos horizontal e vertical da mão por constantes diferentes antes de aplicá-los ao ponteiro. Um gesto que começa com a mesma postura do movimento e termina com outra foi associada ao clique do botão esquerdo do *mouse*.

O sistema provou ser de uso bastante intuitivo e foi pouco afetado por erros no reconhecimento, pois mesmo nas poucas vezes em que ocorreram, isso foi praticamente imperceptível durante o movimento, uma vez que no próximo quadro o ponteiro já era mapeado com a posição correta em relação à mão, devido ao uso de movimentos absolutos e não relativos do mesmo. Somente nas poucas vezes em que o gesto de clique não foi reconhecido o erro foi perceptível (foi escolhida como postura final desse gesto, propositadamente, o punho fechado, uma das posturas com reconhecimento mais problemáticas, conforme já discutido). Ainda assim, a recuperação desse erro é bastante simples, visto que é percebido rapidamente graças ao *feedback* visual do próprio *desktop* e o usuário pode repetir rapidamente o gesto para obter o resultado desejado.

O sistema mostrou-se ainda suficientemente preciso e estável no posicionamento do ponteiro para permitir a seleção de objetos relativamente pequenos (por exemplo, de letras individuais do nome de um ícone e não só do ícone em si), por todo os três usuários. A Figura 19 mostra uma região da tela durante a execução deste teste.

Figura 19 - Manipulação de objetos no *desktop*

Associando os gestos de forma um pouco diferente aos eventos é possível também manipular objetos do *desktop* e não só selecioná-los. Duas posturas podem ser associadas ao movimento, o gesto que se inicia com a primeira e termina com a segunda ao pressionamento do botão esquerdo e o que começa com a segunda e termina com a primeira pode ser associado a soltar o botão. Assim, além da seleção é possível, por exemplo, arrastar e soltar objetos.

5.5 TEMPO DE PROCESSAMENTO

O tempo de processamento necessário para as principais atividades realizadas em cada módulo do sistema foi medido, usando um processador Intel Core 2 Duo de 3GHz e 2GB de memória RAM. Gestures2Go foi executado em uma única *thread* e processador, mas essa limitação não foi imposta às funções do OpenCV. A Tabela 7 mostra valores aproximados das médias desses tempos para cada quadro capturado pela câmera, em 375 testes (ou cinco testes de cinco segundos a quinze quadros por segundo).

Tabela 7 - Tempos de processamento

Atividade		Tempo (ms)
Segmentação		13,600
Análise	Componentes conexos	0,650
	Cálculo de área e centro	0,013
	Cálculo das características	0,002
Reconhecimento	10 posturas	0,003
	Movimento	<0,001
	Gesto	<0,001

Procurou-se medir também o tempo gasto para trocar entre diferentes conjuntos de gestos possíveis (ambos os conjuntos com 10 gestos e 10 posturas), mas esse tempo também foi menor 1 μ s.

Estes resultados mostram como a segmentação consome um tempo muito maior que todo o resto do sistema. Isso não causa surpresa, visto que são aplicadas diversas operações sobre a imagem como um todo. Durante a análise, a identificação dos componentes conexos tem o maior custo, mas mesmo assim

consome menos que um milissegundo de processamento. O cálculo dos momentos de imagem usados para determinar a área e o centro das mãos e da face, trabalha com um número bem menor de pixels e é feito em aproximadamente treze microsegundos. Só não é mais rápido porque o OpenCV calcula todos momentos até terceira ordem, enquanto nesse caso somente os de ordem 1 e 0 são necessários. Enquanto essa função, portanto, pode ser acelerada com facilidade, isso claramente não deve ser uma prioridade. A partir de então, trabalha-se no espaço das características e não no da imagem, reduzindo ainda mais o custo de processamento. O cálculo do vetor de características F e o reconhecimento em si podem ser considerados bastante rápidos, consumindo somente cinco microsegundos, aproximadamente, para dez posturas possíveis e com a checagem da rotação r desativada para obter o pior caso em que o erro q precisa ser calculado para todas as posturas possíveis (nesse caso, o tempo de seu reconhecimento aumenta linearmente com o número de posturas possíveis). O reconhecimento dos movimentos e dos gestos em si consiste somente na avaliação de um número relativamente pequeno de condições e ocorreram em tempos muito pequenos para serem medidos com precisão.

Com estes resultados, considera-se o requisito de baixo tempo de processamento atendido e, caso seja necessário reduzir este tempo, a paralelização da maior parte das operações realizadas na segmentação, a atividade mais custosa, é trivial. Os resultados dos testes indicam, porém, que encontrar soluções mais robustas de segmentação, inclusive entre a mão e o braço, deve ter mais prioridade que tornar o sistema mais rápido.

5.6 INTEGRAÇÃO E PASSO-A-PASSO COGNITIVO

Enquanto os testes descritos acima são úteis para verificar as características e o funcionamento do sistema, inclusive durante o desenvolvimento, são de menor valia para analisar as possibilidades de uso do sistema em contextos específicos de interação em ambientes 3D. Por isso a metodologia do trabalho prevê a integração e testes do sistema com aplicativos desse tipo. O primeiro desses aplicativos é o CoH

e essa integração, sem nenhuma modificação ou mesmo acesso ao código do jogo, foi feita da forma descrita a seguir.

5.6.1 CoH

Uma das primeiras decisões nesse caso foi não utilizar gestos para controlar todos os aspectos do jogo e, ao invés disso, utilizar outro modo de interação para realizar a tarefa de navegação. Enquanto é possível realizar essa tarefa com gestos, o que foi inclusive testado (usando a mesma postura mostrada na Figura 17 para comandar movimento para frente, com a orientação utilizada para indicar mudança de direção), nesse caso isso dificulta, e em alguns casos elimina, a possibilidade de navegar e enviar comandos simultaneamente e, por isso, prejudica o uso do jogo, que prevê essa possibilidade. Bowman et al. (2005) afirmam que o uso de gestos é em interfaces 3D, em geral, beneficiado pelo acréscimo de outros modos de interação.

Tendo isso em mente, e como o módulo Desc2Input já facilita a integração de diferentes formas de interação com aplicativos convencionais, decidiu-se utilizar um tapete de dança para controlar a navegação, inclusive o movimento vertical, permitindo que o usuário utilize os pés para controlar sua locomoção no ambiente 3D do jogo, e as mãos para realizar todas as outras tarefas, usando gestos. Cada um dos 8 botões do tapete representa uma direção e é traduzido por Desc2Input como eventos *key down* e *key up* das setas do teclado e da tecla de espaço, que controla os saltos do personagem. Pisar simultaneamente em diagonais para lados opostos e para frente é traduzido como pular nessa direção e o mesmo ocorre com as duas diagonais para trás (uma das formas mais simples para o usuário pisar ao mesmo tempo nas duas diagonais é executar, de fato, um pequeno salto).

Gestos são usados para todas as outras ações no sistema: substituem os comandos de teclado para seleção e para ações sobre os objetos selecionados, além do clique com o mouse e seu posicionamento, usando a posição da mão na imagem como parâmetro. Desc2Input também foi utilizado para traduzir os gestos como eventos de *mouse* e, principalmente, de teclado.

Dada a importância do *feedback* para o usuário, já discutida, o fato que o CoH ocupa toda a tela ao ser executado e que não permite acesso às imagens que mostra ao usuário, criou-se uma janela que, mesmo sem foco, permanece sempre no topo das outras janelas, no canto da tela inferior esquerdo da tela, para permitir esse *feedback*. Nessa posição, a janela não prejudica consideravelmente a dinâmica do jogo, visto que ocupa o mesmo local que normalmente é utilizado pela janela de bate-papo e outras mensagens de texto.

Antes mesmo da inclusão de gestos no jogo, a movimentação usando o tapete foi testada por três usuários e comparada com o uso do teclado, com resultados interessantes. Como esses resultados são pouco relevantes ao uso de gestos, no entanto, estão registrados somente no Apêndice A. A Figura 20 mostra um usuário controlando o movimento de um personagem, inclusive na vertical.

Figura 20 - Movimento com tapete no CoH⁴

O personagem aparece translúcido na região inferior central da figura devido a um "poder de invisibilidade", durante um "super-pulo", outra capacidade fantasiosa que possui. Esta figura foi capturada enquanto somente os movimentos eram

testados, por isso não se está utilizando o *feedback* que mostra somente as mãos e face detectados em escala de cinza, mas somente a imagem espelhada do usuário. Também é possível ver nesta janela parte do fundo do segundo ambiente de testes, usado na maioria deles.

Como já foi explicado, os personagens do CoH utilizam somente duas posturas de mão devido a limitações da tecnologia de animação usada no jogo. Por isso, os gestos que foram associados às ações do jogador foram definidos fazendo uso somente dessas duas posturas (o punho fechado e a mão aberta com os dedos unidos). Mais duas posturas, no entanto, foram utilizadas. O ícone que o jogo mostra para representar a posição do ponteiro do *mouse* é justamente a mão direita aberta, por isso a escolha dessa postura para controlar esse ponteiro foi natural. O clique é associado com um gesto que tem essa postura como inicial e o punho fechado como postura final, como se indicando a ação de pegar algo. Por fim, a postura com somente o indicador e o polegar estendidos, usada em um gesto bimanual para formar os cantos inferior direito e superior esquerdo de um quadrilátero, foi usada para permitir o controle da câmera. Esse gesto ativa e desativa o controle da câmera, ação associada a uma tecla no jogo, e permite o uso do tapete para mudar sua posição, o que no jogo também é feito utilizando as mesmas teclas usadas para a navegação, as setas.

Como nesta aplicação utilizam-se somente quatro posturas bem diferentes entre si, a tolerância para o reconhecimento das posturas foi aumentada para evitar falsos negativos, principalmente dado que os usuários devem realizar sequências de movimentos relativamente rápidas. O teste utiliza o valor de 5% para essa tolerância. Além disso, não foi associada orientação fixa a nenhum dos gestos, exceto ao gesto de enquadrar, usado para controle de câmera.

Um personagem do jogo foi selecionado e foram definidos gestos para suas doze ações mais utilizadas, além de um gesto para selecionar o próximo alvo (comando de tecla muito usado no jogo), um para clicar com e mover o ponteiro do *mouse* e outro para controlar a câmera. Quinze é um número elevado de comandos, mas isso não parece ter afetado o teste, e dois desses comandos não foram usados. Os gestos associados a cada ação são os mesmos sugeridos nos testes preliminares e listados no Apêndice A, com somente uma mudança: para o poder chamado "*Soul Drain*" o personagem faz uma animação em que as mãos se cruzam no final, que tinha sido sugerida para associação com essa ação durante os testes

preliminares. Neste teste, um gesto similar é usado mas sem o cruzamento das mãos no final. Além disso, alguns gestos que mostram o lado da mão aberta para a câmera foram modificados para que mostrassem a palma (apesar de que, com o aumento de tolerância e as posturas usadas, essa mudança se mostrou pouco necessária).

Após esta rápida integração, os testes foram realizados com dois usuários. Um deles com experiência tanto com o jogo e o personagem quanto com o uso de gestos. O segundo não tinha experiência alguma com o jogo e nem mesmo com ambientes 3D, mas já havia participado de outros testes com o sistema de gestos (em particular os testes de posturas, movimentos e manipulação de objetos no *desktop*). Pediu-se a ambos os usuários que realizassem três sequências de ações, primeiro usando *mouse* e teclado para todas, e depois gestos. Nas descrições das três, mostradas a seguir, os nomes das ações usados no jogo, ou sua abreviação, estão indicados entre parênteses, somente para referência. Seu nome completo está no apêndice A, juntamente com a descrição do gesto associado a cada uma.

A primeira sequência inclui a ativação de dois poderes de longa duração (AM e *Hasten*), sem a presença de oponentes.

A segunda envolve a seleção de um oponente pertencente a um trio deles com o ponteiro do *mouse* (controlado por gestos), a realização de uma ação sobre ele para atrair sua atenção (NB) e um rápido movimento se afastando do oponente, para que somente ele seja atraído e entre em combate com o avatar do usuário, em vez de todo o grupo de oponentes. Continua com mais duas ações (RI e EF) que enfraquecem o oponente e, por fim, a repetição de três ações (CB, XRB e NBolt), até que o oponente seja vencido. Na prática, as três ações foram realizadas uma única vez, visto que o oponente foi rapidamente derrotado.

Na terceira sequência, o segundo oponente de outro grupo de três deve ser selecionado repetindo o comando de "selecionar próximo alvo", duas ações que afetam todo o grupo são realizadas (RI e EF, como antes) e o usuário deve fazer com que seu personagem avance em direção deles (para que o ataquem juntos em vez de se separar) e então deve realizar uma sequência de mais quatro ações que devem afetar e derrotar todo o trio de personagens (*Aim*, *Soul Drain*, *Irradiate* e NBomb). Caso isso não ocorra, as mesmas três ações usadas na sequência anterior (CB, XRB e NBolt) devem ser usadas para derrotar os oponentes, mas isso não foi

necessário. Dois dos treze comandos definidos (DC e controle de câmera) não foram usado nos testes.

A Figura 21 mostra o usuário inexperiente realizando a segunda sequência, a ponto de derrotar o único oponente que afastou dos outros dois, enquanto a Figura 22 mostra o usuário experiente na primeira metade da terceira sequência, com um grupo de oponentes.

Figura 21 - Usuário inexperiente na segunda sequência⁴

É interessante notar que em nenhuma destas duas imagens os usuários estão fazendo o gesto equivalente à ação e à animação sendo realizadas pelo personagem. Essa ação foi uma resposta ao gesto anterior e os usuários já estão realizando o gesto associado à próxima ação.

O jogo permite que uma (e somente uma) ação seja colocada numa fila de execução enquanto outra é realizada, como mecanismo para reduzir o impacto de atrasos na rede o que, como já foi comentado, favorece o uso de gestos, que tende a ser mais lento que o simples pressionar de uma tecla.

Figura 22 - Usuário experiente na terceira sequência⁴

Neste teste, as seguintes perguntas são feitas ao usuário antes da realização de cada passo:

- Você lembra qual gesto deve usar? Foi fácil lembrar desse gesto?
- O *feedback* visual auxilia na realização do gesto?

E depois da ação realizada, pergunta-se:

- A resposta ao gesto foi a esperada?

Para o teste usando teclado e mouse foram feitas as mesmas perguntas, substituindo a expressão "tecla ou ícone" pela palavra "gesto". Além das respostas a estas perguntas, ao final dos testes os usuários puderam fazer comentários qualitativos com relação a eles. As respostas para cada pergunta a cada passo e

para cada um dos dois usuários neste teste estão listadas no Apêndice A, enquanto esta seção resume e destaca seus principais resultados.

Ao contrário do que normalmente ocorre em um passo-a-passo cognitivo ao analisar uma interface, quando não há pressão de tempo, estas sequências de ações, com exceção da primeira, devem ser realizadas pelo usuário o mais rápido possível, para derrotar os oponentes e minimizar o risco a que se expõe. Por isso, na segunda e na terceira sequência, as perguntas e respostas do passo-a-passo cognitivo não foram feitas a cada passo do teste, como é o procedimento padrão, mas somente quando um dos usuários mostrava alguma dúvida ou, principalmente, ao final do teste, com o usuário revisando suas ações nos vídeos que o registram, pausando ao fim de cada passo e antes do próximo.

Os testes se iniciaram com o usuário experiente e o inexperiente pôde assisti-los, como auxílio para memorizar as sequências, as teclas pressionadas (relatadas em voz alta), os ícones clicados com o *mouse* e os gestos usados. Ambos os usuários realizaram todas as sequências uma vez, tanto com teclado quanto com gestos, como treino antes do teste. A calibração do sistema para cada um dos usuários, assim como o deslocamento do personagem até que fosse encontrado cada grupo de oponentes usado em cada sequência, foram feitos usando o teclado e não foram considerados como parte do teste. Para todas as sequências foram usados grupos de três oponentes da classe "*minion*" e de nível 49.

Essas ações que precederam o teste já mostraram duas falhas em sua execução. Em primeiro lugar, ambos os usuários já haviam participado de outros testes do sistema e estavam acostumados a fazer sua calibração individualmente, usando as teclas para gravar o fundo conhecido e amostrar a cor da pele. Para realizar este teste, no entanto, precisavam se posicionar mais afastados do computador e da câmera, para que seus gestos amplos com os dois braços fossem enquadrados por ela. A essa distância, não podiam usar o teclado. Entre a posição de calibração e a posição de teste havia uma fonte de luz que alterava suficientemente a cor da pele capturada pela câmera para prejudicar a segmentação. Isso foi resolvido com uma pequena mudança na posição do teste e com um usuário operando o teclado enquanto o segundo exibia a mão para amostragem, mas isso mostra que o procedimento de amostragem precisa ser modificado (visto que o sistema deveria se utilizável por um único usuário).

O segundo problema está na metodologia do teste em si, em particular na escolha dos oponentes. Encontrar esses grupos com o tamanho e composição desejada consome tempo antes de cada execução das sequências dois e três, o que teria causado problema em um teste com mais usuários. Além disso, o uso de oponentes reais, mesmo controlados pelo computador, inclui a pressão de tempo, que impossibilita o diálogo entre cada passo, e acrescenta variáveis adicionais ao teste. Na Figura 21, por exemplo, o oponente está fugindo do jogador. Felizmente ele permaneceu ao alcance por tempo suficiente para ser derrotado antes que alcançasse um grupo de oponentes, que de outra forma seriam atraídos para o jogador e perturbariam o teste. Na Figura 22 um dos oponentes acaba de desfazer parte do efeito das ações do usuário sobre todo o grupo. Felizmente isso ocorreu com o usuário experiente que, realizando um gesto além da sequência estabelecida, rapidamente corrigiu esse problema. O CoH tem um mecanismo que permitiria contornar esse problemas, permitindo que os jogadores criem missões e histórias que podem ser jogadas por eles mesmos e por outros jogadores. Inicialmente planejou-se utilizar esse recurso para criar uma missão com alvos estáticos para execução dos testes, mas por questões de tempo isso não foi feito. Estes testes mostraram que, no futuro, vale a pena esse investimento de tempo e esforço para criar um ambiente de teste controlado no jogo.

Ainda antes do início dos testes em si, durante as sequências de treinamento usando os gestos, ambos os usuários exibiram a tendência de, após realizar um gesto, esperar o personagem terminar a animação de seu gesto e ação equivalentes e só então os usuários iniciavam o próximo gesto. Esse comportamento causa pausas entre as ações realizadas pelo personagem cuja duração é o tempo de execução do gesto e que são indesejáveis nesse jogo. Além disso ambos os usuários, mas principalmente o inexperiente, relataram que a sensação de atraso entre seu gesto e o equivalente do personagem causava estranhamento no começo. Durante os testes, no entanto, ambos já se mostraram habituados a evitar essas pausas.

Durante o teste, cada um dos usuários realizou um total de 17 gestos, usando 13 dos quinze gestos definidos inicialmente, com 3 posturas (a quarta, para câmera, não foi usada nos testes), 6 movimentos e 2 locais diferentes usados nessas definições. Nenhum dos gestos foi reconhecido incorretamente ou deixou de ser

reconhecido e não foi necessário repetir nenhum gesto, nem mesmo durante os treinos.

O usuário inexperiente teve dificuldade de lembrar que gesto correspondia a uma determinada ação para duas ações durante os testes, e mais vezes durante os treinos. Curiosamente, para essas duas ações o personagem realiza animações que são difíceis de visualizar vendo-o de costas, como é mostrado por padrão no jogo. Esse mesmo usuário teve ainda mais dificuldade para lembrar as teclas ou ícones associados a cada ação usando teclado e *mouse*, quando necessitou de auxílio por três vezes, mesmo depois dos treinos.

Usando teclado e *mouse*, três das doze ações requerem o uso do *mouse* enquanto as outras nove permitem seu uso mas são preferencialmente disparadas pelo teclado. Dessas nove, cinco requerem que somente uma tecla seja apertada e quatro dependem da combinação de duas teclas, a tecla "Alt" e um número de um a cinco (nas teclas que ficam sobre as letras, não no teclado numérico). O usuário inexperiente, além da dificuldade para lembrar de como comandar algumas ações, relatou que teve grande dificuldade para utilizar essas combinações de teclas com a mão esquerda enquanto a direita permanecia sobre as setas para controlar o movimento e sua atenção devia se concentrar na tela e afirmou que, nesse aspecto, o uso de gestos foi "bem mais fácil". Considerou esse problema inclusive como mais grave que a dificuldade de lembrar dos comandos. Essa interface com teclado e *mouse* é padrão no CoH e não foi modificada nestes testes.

Como era de se esperar, o usuário que têm experiência com o jogo a mais de três anos não teve dificuldade alguma usando teclado e *mouse* e relatou que, na sua opinião, essa interface era mais fácil e rápida de utilizar que a interface com tapete e gestos, mas menos divertida. Também não teve dificuldade em lembrar os gestos associados a cada ação, uma vez que presenciou o personagem realizando os mesmos gestos por milhares de vezes nesse período. Durante o treino com os gestos, esse usuário realizou duas ações na ordem invertida, mas esse foi seu único desvio e não se repetiu durante os testes.

Curiosamente, o usuário experiente mostrou uma tendência de mover a cabeça inconscientemente durante o jogo, somente ao usar gestos, ainda que fosse obrigado a manter o foco de sua visão na tela. Por vezes a cabeça se inclinava junto com o movimento do personagem e muitas vezes o usuário abaixava a cabeça quando o personagem o fazia, principalmente antes de um gesto que exigisse que

as mãos começassem acima da cabeça, o que não era necessário para permitir que a câmera enquadrasse o gesto. O usuário inexperiente não demonstrou comportamento similar.

O uso do tapete de dança, além de constituir uma interface interessante para navegação, teve uma consequência não planejada mas que facilitou o reconhecimento dos gestos. O tapete demarca claramente uma região limitada onde o usuário deve permanecer e isso evita que ele se desloque perceptivelmente em relação às fontes de luz, auxiliando a segmentação.

5.6.2 Ambiente de Iluminação

A estratégia para integração de Gestures2Go com o ambiente 3D interativo para estudo de iluminação é a mesma utilizada com o CoH. O tapete de dança é utilizado para permitir a navegação e associado às setas do teclado, que também podem ser usadas para navegação no ambiente, enquanto gestos são associados às teclas usadas para acender e apagar as três luzes, abrir e fechar a janela e selecionar cada um dos três modos de renderização. O *feedback* é fornecido por uma janela que permanece sempre no topo das outras, mesmo sem o foco. Ao contrário do que ocorre no jogo, neste ambiente não existe uma funcionalidade de pulo.

O principal motivo de implementar também uma interface por teclado e não só através de gestos e utilizar Desc2Input para integrar os dois sistemas, em vez de fazer essa integração diretamente, é permitir o desenvolvimento deste aplicativo e do sistema de reconhecimento de gestos em paralelo e permitir testes com o ambiente inicialmente sem incluir variáveis relacionadas ao uso de gestos. A interface com teclado também amplia o potencial de uso do ambiente, ainda que o sistema de reconhecimento de gestos exija somente equipamento bastante acessível.

Neste ambiente não existe a mesma necessidade, que existia no CoH, de um modo de interação separado para navegação e poderiam ter sido definidos gestos para navegação, visto que não há necessidade do usuário se deslocarem e realizarem outras ações simultaneamente. A reação ao uso do tapete, no entanto, foi

tão positiva, tanto por parte de seus usuários quanto de terceiros que assistiram a um dos vídeos de teste, que decidiu-se repetir seu uso.

Foram definidos dois conjuntos de gestos para este aplicativo, baseados nas duas sugestões diferentes apresentadas durante os testes preliminares, usando as posturas com um, dois ou três dedos levantados para representar os números das luzes ou algoritmos e as posturas com a mão aberta e o punho fechado para representar, respectivamente, ligado e desligado, enquanto movimentos verticais do punho são interpretados como abrir ou fechar a janela. Para esse conjunto de posturas, utilizou-se uma tolerância de 4% para o reconhecimento.

Infelizmente, o desenvolvimento do ambiente de iluminação, que ocorreu independentemente deste trabalho, não foi concluído em tempo de realizar estes testes. Como se pode considerar que a navegação usando o tapete já foi plenamente testada usando o CoH e não é o foco deste trabalho, os testes relativos ao ambiente foram simplesmente testes dos gestos usados para comandar a iluminação e a renderização.

Foi criada uma aplicação de console separada do reconhecedor de gestos que, ao receber comandos do teclado, imprime mudanças no estado das luzes, da janela e da renderização e Desc2Input foi utilizado traduzir os gestos definidos para essas ações como esses eventos de teclado. A janela de *feedback* de Gestures2Go neste caso é similar à usada nos outros testes, mas como no CoH, não mostra a postura detectada.

Esta configuração foi testada por dois usuários. Um deles fazia a postura com dois dedos levantados de uma maneira em que ela era confundida, com alguma frequência, com a postura com os três dedos levantados, o que causou algumas entradas incorretas e não havia ocorrido durante os testes com as posturas, inclusive com o mesmo usuário. Nesse caso, o *feedback* sobre qual postura é detectada a cada momento, que foi eliminado neste aplicativo, teria auxiliado a detectar o motivo do erro com mais velocidade, inclusive pelo próprio usuário, que em seguida mudou a forma de realizar a postura e eliminou o erro.

Como os usuários que realizaram este teste foram os mesmos que realizaram o anterior, um ponto foi comparado entre ambos os testes. Neste caso, um gesto gera uma ação aparentemente instantânea (um texto aparece na tela ou, no aplicativo as mudanças nas luzes e no algoritmo), enquanto no CoH o gesto dava início a uma animação que durava entre um e três segundos e se assemelhava ao

gesto. Enquanto no jogo isso dá a impressão de um atraso entre o reconhecimento e a ação, no ambiente 3D, com as respostas parecendo instantâneas, os usuários têm a impressão que o sistema faz o reconhecimento de forma bastante rápida, o que de fato ocorre, mas em ambos os casos.

A Figura 23 mostra um momento da realização deste teste, com a aplicação de console sob a janela de *feedback* e ao fim da execução do sexto de uma série de comandos (acender uma das luzes).

Figura 23 - Comandos para o ambiente de iluminação

6 CONCLUSÕES

Neste trabalho desenvolveu-se um modelo de gestos e um sistema para seu reconhecimento com as mãos livres cuja principal característica e contribuição é permitir a escolha de gestos para interação dentre um grande conjunto de gestos possíveis.

As principais aplicações para as quais foi projetado são ambientes 3D aplicados a entretenimento ou educação, e daí são derivados seus requisitos.

O modelo de gesto, seguindo uma abordagem linguística, o descreve através da combinação de elementos mais simples para uma ou ambas as mãos: uma ou duas posturas de mão e sua orientação no plano da imagem, o movimento da mão e o local onde o gesto se inicia, em relação ao corpo do gesticulador.

Os gestos são segmentados no tempo mantendo posturas por um intervalo curto e configurável (nos testes utilizou-se 200ms), portanto ao menos uma postura sempre deve ser definida para cada gesto. Os outros elementos (orientação, movimento e local), no entanto, podem ser definidos ou não, o que permite o uso de diferentes tipos de gestos, sejam estáticos ou dinâmicos, com uma ou duas posturas, com ou sem movimento, com local inicial definido ou não e com qualquer uma das mãos, ou com ambas.

A abordagem implementada para combinar esses elementos de acordo com o modelo, usando uma máquina de estados finitos relativamente simples, com regras explícitas e sem necessidade de treinamento, é provavelmente a segunda contribuição mais importante deste trabalho, após o modelo.

Esta máquina de estados independe da segmentação, das estratégias usadas para reconhecer movimentos ou posturas e até da forma como são representados. O reconhecimento desses dois elementos não precisa nem mesmo ser independente, como ocorre neste trabalho. A única hipótese imposta por esta abordagem é segmentação do gesto no tempo por uma ou duas posturas conhecidas. Enquanto isso certamente não ocorre com o uso de gestos para auxílio da comunicação natural, é uma hipótese bastante válida para seu uso em interfaces e principalmente para envio de comandos, que são por natureza discretos.

Assim, esta máquina de estados permite combinar no modelo proposto saídas de diversas combinações de algoritmos ou estratégias para reconhecimento de

posturas ou movimentos que, como constatado na revisão da literatura, são abundantes, enquanto soluções que combinam esse elementos para permitir a definição de um grande número de gestos são bem mais raras ou inexistentes, principalmente com a flexibilidade que a solução implementada neste trabalho permite. Mesmo a possibilidade de usar ambas as mãos nos gestos não existe na maior parte dos trabalhos, de acordo com Jaimes & Sebe (2007).

Com este modelo e implementação, um número relativamente pequeno de elementos permite a definição e o reconhecimento de dezenas ou centenas de milhares de gestos. Como aplicações de entretenimento podem versar sobre temas e domínios bastante distintos e ter requisitos bastante diferentes entre si (como o exemplo do CoH que, por um motivo técnico, limita as posturas de mão usadas pelos personagens), essa variedade de gestos e de seus tipos permite uma maior probabilidade de definir um gesto que possa ter uma relação lógica, em cada contexto, específico com a ação de interface a que é associado, facilitando seu aprendizado, memorização e, portanto, uso. E esses, além do uso de equipamento acessível, estão entre os principais requisitos para tal tipo de aplicação.

Essa acessibilidade, não só do equipamento necessário como do ambiente para uso do sistema, também é alta, o que foi um dos requisitos discutidos na introdução e que também pode ser considerado cumprido. Somente uma câmera é usada, um dispositivo simples, com resoluções relativamente baixas (nos testes foram usadas câmeras com 640x480 e de 320x240 pixels) e baixas taxas de captura (como 15 quadros por segundo), o que permite sua comunicação através da porta USB sem a necessidade, por exemplo, de placas de captura de vídeo. Também não são utilizados infravermelho ou medidas de profundidade.

A principal característica que uma câmera pode ter para otimizar seu uso no sistema é a capacidade de desativar o ajuste automático de exposição e de *white balance*. Enquanto a primeira capacidade é somente incomum, a segunda era praticamente inexistente em câmeras de baixo custo até recentemente. Todos os testes, no entanto, foram realizados com câmeras que permitiam desativar somente o ajuste de exposição, demonstrando a viabilidade do uso desse equipamento.

Além disso, enquanto a segmentação exige um fundo conhecido, esse fundo não precisa ser completamente estático, muito menos uniforme, nem ter características especiais, exceto talvez alguma região branca para que, caso seja

impossível desativar o ajuste automático de *white balance*, esse ajuste ao menos permaneça mais estável.

A iluminação também não precisa ser especial, no entanto os testes mostraram que é um dos fatores que tem maior influência sobre a segmentação, principalmente na presença de fontes de luz intensa que podem causar reflexos especulares na pele do usuário, ou efeitos de *bloom*. O sistema, atualmente, não se adapta a diferentes condições de iluminação e há casos em que isso faz com que não apresente a qualidade mínima que se estabeleceu na introdução, de 60% de reconhecimento correto. Além disso, uma restrição do sistema considerada grave é a não segmentação entre a mão e o braço que implica na necessidade do uso de mangas compridas. Ainda que a solução proposta por Deimel & Schroter (1999) possivelmente seja capaz de eliminar esse problema, não foi possível implementá-la e testá-la por limitações de tempo. Devido a essas duas questões, iluminação e braço, considera-se o requisito de permitir o uso do sistema na maior parte dos ambientes e situações domésticas somente parcialmente cumprido.

Todos os outros requisitos do sistema podem ser considerados atendidos a contento por esta implementação. O sistema fornece imagens, inclusive de diferentes tipos, para permitir o *feedback* visual para o usuário. Também tolera variações na execução de todos os elementos dos gestos, conforme confirmado pelos testes interativos, com aplicativos e até nos testes estáticos de reconhecimento das posturas.

Facilitar a integração com outras aplicações, inclusive as que não foram projetadas para o uso de gestos, além de permitir interação multimodal, são atendidos por implementações de Desc2Input, através da tradução de descrições de gestos ou quaisquer outros eventos para eventos de entrada, seja do sistema operacional, de um gerenciador gráfico ou mesmo de outra API, como um *engine* de jogos (neste trabalho foi implementada somente a integração através de eventos do Microsoft Windows).

O requisito de baixo consumo de processamento também está satisfatoriamente cumprido, com o sistema concluindo o reconhecimento durante os testes, em média, em pouco mais de 14ms. Mais de 95% desse tempo é consumido pela segmentação e a quase totalidade dos outros 5% pela determinação dos contornos dos objetos de interesse, com todo o restante da análise e do reconhecimento consumindo menos de 20 μ s.

O sistema também permite a rápida troca de que gestos são associados a que ações, outro requisito atendido, e entre que conjuntos de gestos devem ser reconhecidos a cada momento, inclusive durante tempo de execução. Essa troca consumiu tempos menores que 1 μ s durante os testes e mesmo a adição de posturas totalmente novas ao sistema é um processo simples que consumiu poucos minutos de trabalho nos testes.

A possibilidade de configurar rapidamente subconjuntos menores de gestos que podem ser reconhecidos a cada instante, bem como de seus componentes, principalmente das posturas, traz outra grande vantagem ao sistema. Enquanto teoricamente é possível que reconheça todo o grande número de gestos que o modelo permite definir, já que esse número resulta da combinação de números muito menores de componentes, reconhecer todos esses gestos é desnecessário e até indesejável no contexto de interação. Limitar os conjuntos de gestos e componentes não só torna o reconhecimento mais rápido mas tem como maior vantagem permitir que, no projeto das interfaces, evite-se o uso de posturas similares no mesmo contexto, como por exemplo as posturas 11 e 17 da Figura 2 (uma com o polegar erguido, e uma com o dedo mínimo).

Em vez de remover essas posturas semelhantes do conjunto de possibilidades, ou de utilizar algoritmos com requisitos muito mais altos de processamento ou calibração, optou-se por permitir o uso de quaisquer posturas e inclusive de incluir posturas novas no sistema, mesmo que isso aumente sua taxa de erros de reconhecimento. Ainda que seja desejável usar todas essas opções em diferentes contextos, aplicações ou domínios, nunca deveria ser necessário utilizar todas ao mesmo tempo. Esse fato, além da facilidade de alterar os conjuntos de gestos usados, permitem que o projeto de cada interface ajuste os conjuntos de gestos escolhidos em cada contexto para que permaneçam com associações lógicas com as ações que comandam, mas sem utilizar gestos muito similares. O sistema conta até com uma funcionalidade que retorna uma medida de similaridade, em porcentagem, de duas posturas. Esta funcionalidade pode ser utilizada não só para informar o projetista da interface sobre potenciais problemas em conjuntos de gestos que escolheu como ainda para auxiliar na determinação de uma tolerância mais adequada a cada conjunto de posturas, para o seu reconhecimento, que o valor padrão de 2%.

Por fim, a calibração, ainda que necessária para a segmentação, consiste em dois passos bastante rápidos que em média consomem menos de dez segundos, de forma que se considera plenamente atendido o requisito de não exigir custosos ou longos treinamento ou preparação do sistema. A estratégia de calibração implementada nos testes, usando teclado para amostrar a pele, mostrou-se inadequada no teste com o CoH em que o usuário deve estar distante do teclado, mas alternativas para a segmentação, incluindo sua calibração, são uma das prioridades de trabalhos futuros e diversas alternativas podem sanar esse problema em particular.

Além das principais contribuições do trabalho, o modelo flexível de gestos e a forma implementada para combinar todos os seus elementos, podem ser citadas algumas contribuições menores do trabalho. Assim como as contribuições maiores, o *framework* proposto e utilizado na implementação do sistema, por exemplo, facilita o uso intercambiável de diferentes algoritmos para cada uma das principais atividades do reconhecimento de gestos com VC. Faz isso através da definição clara de módulos para cada uma delas e das interfaces entre esses módulos, inclusive permitindo que sejam combinados através de herança múltipla quando necessário, mas sem limitar-se a especificidades de alguma solução em particular, até como consequência de seu caráter abstrato. Assim, este *framework* permite não só a experimentação e comparação de diferentes soluções em um mesmo domínio mas também a exploração do uso de gestos em outros domínios. A maior parte dos trabalhos semelhantes analisados não tem estes aspectos mais gerais.

Além disso, o algoritmo usado para o reconhecimento de movimentos é inédito entre os trabalhos analisados, mais rápido e com menor consumo de memória, mas trata-se de uma solução bastante específica, simples e se aplica somente a um conjunto fixo e limitado de movimentos, por isso pode ser considerado, no máximo, como uma pequena contribuição. O mesmo se aplica ao reconhecimento de postura que, ainda que seja mais geral que o de movimentos e tenha diferenças em relação aos trabalhos semelhantes, tem caráter eminentemente derivativo.

Pode ser considerada uma limitação do sistema o fato de não conter mecanismos explícitos para definir a sincronicidade dos gestos, seja entre si, entre as ações das duas mãos, seja com ações realizadas por outras modalidades de interação, como o uso do tapete de dança. Parâmetros dos gestos podem, no

entanto, ser armazenados de forma simples nesta arquitetura e conter diversos valores para instantes de tempo, como por exemplo quando o gesto se inicia ou termina, separadamente para cada mão ou para o gesto como um todo. Isso permite a fusão desses dados já no nível das características do gesto, o que é adequado neste caso, como discutido por Jaimes & Sebe (2007), visto que a sincronia das ações das mãos pode ser considerada importante na sua interpretação. O mesmo mecanismo pode ser utilizado para a fusão dos gestos com outras modalidades de interação, mas como afirmam os mesmos autores, nesse caso eles geralmente estão menos vinculados temporalmente com as outras modalidades, pois se referem a informações diferentes e complementares, em vez de informações similares e redundantes. Quando isso ocorre, são mais indicados mecanismos de fusão semântica, que não devem ser definidos pelo *framework* pois ocorrem após o reconhecimento do gesto.

Os testes em ambientes 3D (não só das funcionalidades do sistema fora de contexto) também produziram alguns resultados interessantes, mesmo tendo sido realizados com poucos usuários. O comportamento de tentar sincronizar seus gestos com as animações equivalentes do personagem, bem como os comentários com relação a esse "atraso", que também ocorre mas não é percebido ao se utilizar a interface convencional, indicam que, numa aplicação desse tipo, em que os gestos do usuário e de seu avatar estão tão intimamente relacionados não só na forma como no tempo, é desejável utilizar mecanismos para reduzir esse atraso, como o reconhecimento de gestos incompletos. Existem trabalhos nessa área, especificamente para o reconhecimento de movimentos, usando, por exemplo, regras explícitas, predição-correção ou redes neurais, mas não foram encontrados resultados publicados sobre seu uso em interação nesse contexto, indicando uma interessante possibilidade para trabalhos futuros.

Na verdade, para parte dos movimentos básicos reconhecidos pelo sistema uma simples mudança na forma como são segmentados temporalmente é capaz de "prever" esses gestos. Basta que, como ocorria em uma versão do sistema durante o desenvolvimento, o reconhecimento do movimento não aguarde que a mão pare para terminar, mas somente que se desloque além de um limite configurável e que assuma a postura final. Assim, para gestos dinâmicos com as mesmas posturas inicial e final, o gesto em geral é reconhecido antes que o movimento termine. Se forem definidos gestos com a mesma postura inicial e final tanto para movimentos

horizontais ou verticais quanto para diagonais ou circulares, no entanto, esses últimos dois nunca são reconhecidos se o sistema não aguarda que o movimento das mãos pare, ao menos com o algoritmo simples utilizado atualmente, por isso na implementação atual o movimento só é classificado após o seu término.

Além dessa questão, percebida no teste com o CoH, e do problema já discutido em sua metodologia que poderia ter feito uso de um ambiente mais controlado, alguns outros fatores levam à conclusão que o cenário utilizado com essa aplicação não é tão adequado para esse tipo de teste como parecia ser a princípio.

Inicialmente foram considerados dois cenários de teste. Um deles seria utilizar um personagem recém-criado, com poucas ações, para que o usuário pudesse aprender a utilizar o jogo e o personagem usando gestos da mesma forma que se aprende jogando normalmente com teclado e *mouse*, com o personagem evoluindo lenta e gradativamente, com novas ações possíveis somente depois que as anteriores estão dominadas. O segundo cenário, o que foi adotado, é utilizar um personagem que já atingiu seu nível máximo. Assim, o personagem tem uma grande variedade de ações possíveis (o que, em geral, é mais divertido, desde que se saiba fazer uso de todas essas opções) e o usuário pode se concentrar na ação do jogo durante o teste, e não na evolução do personagem, que usa interfaces bidimensionais para as quais os gestos são menos adequados. Além disso, um personagem recém criado só possui três ou quatro ações possíveis, o que limita o número de gestos e força o usuário a repetir essas ações várias vezes, o que poderia se tornar cansativo.

O ideal seria um cenário utilizando um personagem de nível intermediário, ou ao menos limitar mais as ações utilizáveis pelo personagem de nível alto, o que, em parte, foi feito nesse teste. Mesmo tendo limitado as mesmas a um número relativamente elevado das ações mais frequentemente utilizadas, no entanto, o usuário inexperiente questionou e se sentiu tolhido ao não poder fazer uso de todas.

Uma última conclusão sobre os testes com esse aplicativo específico é que os jogos atuais do estilo MMO, apesar de apresentarem diversas vantagens para o uso de gestos, que já foram discutidas, também têm algumas desvantagens sérias. As interfaces bidimensionais para evolução do personagem, parte importante desse tipo de jogo, não favorecem o uso de gestos, como já foi dito. Além disso, mesmo em jogos como o CoH em que pequenas missões de dez a trinta minutos já podem

contribuir perceptivelmente para a evolução do personagem, para que essa evolução se aproxime do seu ápice são necessárias dezenas de horas de jogo. Sessões de jogo costumam durar entre uma e três horas, ou até mais. Ainda que isso não tenha sido registrado nos testes, como um usuário que experimentou realizar uma missão de quinze minutos usando a interface com gestos e tapete, o autor pode afirmar com segurança que uma sessão de uma hora exigiria um esforço considerável de boa parte dos usuários. Estudos sobre a usabilidade do jogo usando gestos e sobre a ergonomia desse tipo de interface também são interessantes trabalhos futuros.

Ainda com relação aos testes com aplicativos, o segundo objetivo que só pode ser considerado parcialmente cumprido são os testes com o ambiente de iluminação, ainda que isso tenha ocorrido por motivos além do controle deste trabalho.

Dentre os trabalhos futuros, portanto, uma das prioridades é realizar essa integração de fato e testar o uso de gestos no contexto do ambiente de iluminação. A integração e testes com o atlas anatômico virtual com realidade aumentada, o VIDA citado na introdução, também é um trabalho futuro interessante.

A principal prioridade para trabalhos futuros no entanto é, como já deve estar claro, a pesquisa e implementação de melhorias na segmentação que, como já se afirmou diversas vezes, foi relegada a um segundo plano, enquanto o foco deste trabalho foi o modelo para definição de um grande número de gestos com base em seus componentes e a forma implementada para combinar os mesmos. Trata-se, no entanto, de uma tarefa essencial, principalmente para a determinação da postura. Atualmente, a segmentação consome 95% do tempo de reconhecimento e é a maior fonte de seus erros. Também é responsável pela necessidade de calibração, por mais rápida que seja, e pela maior restrição do sistema, a necessidade de uso de mangas compridas.

Algumas medidas relativamente simples podem reduzir consideravelmente esses problemas, como modelos adaptativos para o fundo usando misturas de Gaussianas, a implementação do algoritmo de segmentação do braço de Deimel & Schroter (1999) ou o rastreamento das mãos e da cabeça que pode reduzir o custo computacional da segmentação ao permitir que ocorra em regiões menores da imagem e ainda torná-la mais robusta e permitir que as mãos cruzem o corpo. O rastreamento deve ter, no entanto, reinicialização automática para que seu uso seja

adequado no contexto de interação. Mais complexo que um modelo adaptativo da cor do fundo, devido ao movimento das mãos, é um adaptativo para a cor da pele, como no trabalho de Soriano et al. (2003), outra frente interessante para futuras pesquisas. A face, que se move menos durante os gestos e permanece com um padrão menos variável e, portanto, de detecção e rastreamento mais simples, pode ser útil para esse tipo de esforço. Se não for viável eliminar a amostragem ou fazer a auto-calibração da cor da pele, utilizar uma forma para sua fazer sua amostragem que não requeira o uso do teclado já elimina um dos problemas encontrados durante os testes. Uma postura específica, por exemplo, e portanto um padrão conhecido para detecção usando contornos em vez de cor, poderia ser associado à calibração. Atualmente, no mesmo laboratório em que se desenvolveu esta tese, há uma pesquisa de doutorado em andamento justamente sobre remoção de fundo e segmentação.

Outro trabalho deste laboratório que pode auxiliar não só na segmentação como no reconhecimento de uma maior variedade de gestos é o uso de informação de profundidade, como faz o sistema descrito por Nakamura & Tori (2008) para detecção de colisões entre objetos reais, e em particular um vídeo-avatar, e objetos virtuais. Tokunaga et al. (2009) também utilizam informação de profundidade para a renderização de seu vídeo-avatar e é outro trabalho que tem vantagens para a integração com gestos, pela sinergia já discutida entre os mesmos e a representação de vídeo do usuário gesticulando, inserida no ambiente 3D. A própria detecção de colisão entre objetos reais e virtuais já citada, assim como as interfaces tangíveis estudadas por Tsuda et al. (2007) e Calife et al. (2009), ou as que exploram dispositivos não convencionais (Bernardes et al., 2009), podem ser integradas ao reconhecimento de gestos para que se tenha um conjunto mais completo e flexível de ferramentas para interação 3D baseada em câmeras.

Além disso, o uso do sistema por desenvolvedores em um projeto ainda exige sua compilação e o uso de linguagem C para a definição dos gestos e sua associação com ações. Um sistema que permite essa configuração em um arquivo XML ou através de uma interface gráfica já está especificado para futuro desenvolvimento. O sistema também deve ser, em breve, fornecido como código aberto não só como código-fonte como também como componentes pré-compilados (como DLLs) para inclusão em outros projetos ou mesmo como aplicativo

independente, acessado usando também *sockets* e não somente eventos como feito atualmente.

Por fim, dentre as muitas possibilidades de futuras pesquisas que se abrem ao término desta etapa do trabalho, uma delas é interessante justamente por ter estado presente em seu início. Uma das primeiras propostas para esta pesquisa era de utilizar notações padronizadas de descrição de gestos, muito usadas no estudo de línguas de sinais, para facilitar a integração entre os módulos do sistema desenvolvido neste trabalho, sua configuração e até comunicação com outros sistemas. O trabalho evoluiu em outra direção e atualmente posturas, movimentos, locais e gestos fazem uso de descrições, mas aceitam descrições quaisquer. Existem diversos sistemas desses sistemas de notação, no entanto, cuja pesquisa e adoção, incluindo uma adaptação às necessidades dessa aplicação, podem ser úteis para estabelecer uma forma padrão para registrar e comunicar gestos. Stokoe (1960), por exemplo, além de formalizar a divisão de gestos em componentes, criou uma notação utilizando uma série de símbolos específicos, enquanto Liddell & Johnson (1989) apresentam uma notação semelhante mas utilizando caracteres alfa-numéricos. O Hamburg Notation System, ou HamNoSys (2010), utiliza uma série de símbolos gráficos específicos para codificar os diversos elementos que compõe um gesto e o SigML (Hanke, 2001) é uma versão desse sistema em XML. O uso dessas linguagens facilita inclusive a ligação entre o reconhecimento e a síntese de gestos, em geral feita utilizando avatares tridimensionais, que pode ser usada, dentre outras aplicações, como mais um canal para enriquecer a comunicação intermediada por avatares, controlados por humanos ou por agentes autônomos.

REFERÊNCIAS¹³

ALLIVE, 2007. LiveMove White Paper. Disponível em: <http://www.ikuni.com/papers/LiveMoveWhitePaper_en.pdf>. Acesso em 10 dez. 2009.

AMAYEH, G. ET AL. A Component-Based Approach to Hand Verification. In: COMPUTER VISION AND PATTERN RECOGNITION. **Anais...** 2007.

APPENRODT, J. ET AL. Data gathering for gesture recognition using mono color, stereo color and thermal cameras. In: INTERNATIONAL MEGA-CONFERENCE ON FUTURE GENERATION INFORMATION TECHNOLOGY, 1. **Anais...** 2009. p. 78-86.

AZUMA, R. A Survey of Augmented Reality. **Presence: Teleoperators and Virtual Environments**, v. 6, p. 355-385. 1997.

BAU, O. & MACKAY, W. OctoPocus: A Dynamic Guide for Learning Gesture-Based Command Sets. In: ACM SYMPOSIUM ON USER INTERFACE SOFTWARE AND TECHNOLOGY. **Anais...** 2008. p. 37-46.

BAUER, B. & HIENZ, H. Relevant Features for Video-Based Continuous Sign Language Recognition. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 4. **Anais...** 2000. p. 440-445.

BERNAL-MERINO, M. Localisation and the cultural concept of play in games. 2007. Disponível em: <http://www.gamecareerguide.com/features/454/localization_and_the_cultural_.php>. Acesso em: 30 jul. 2010.

BERNARDES, J. **Desenvolvimento de um ambiente para visualização tridimensional da dinâmica de risers**. 2004. 203 p. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo, São Paulo. 2004.

BERNARDES, J. ET AL. Augmented Reality Games. In: L. OLLI, H. WIRMAN AND A. FERNANDEZ (Ed.). **Extending Experiences**: Structure, analysis and design of computer game player experience. Lapland University Press, 2008. p. 228-246.

BERNARDES, J. ET AL. Jogos Eletrônicos e Realidade Virtual. In: KIRNER, C. & TORI, R. (ed.), **Realidade Virtual**: Conceitos e Tendências, SBC. 2004.

BERNARDES, J. ET AL. Integrating the Wii controller with enJine: 3D interfaces extending the frontiers of a didactic game engine. In: **Computers in Entertainment**, v. 7. 2009.

BLACK, J. & JEPSON, D. EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation. In: **International Journal of Computer Vision**, v. 26, p. 63–84. 1998.

¹³ De acordo com: ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6023: informação e documentação: referências: elaboração. Rio de Janeiro, 2002.

BOUSSEMARY ET AL. A Framework for 3D Visualisation and Manipulation in an Immersive Space using an Untethered Bimanual Gestural Interface. In: VRST. **Anais...** 2004. p. 162-165.

BOWMAN ET AL. **3D User Interfaces: Theory and Practice**. Addison-Wesley, 2005. 512 p.

BRADSKI, G. & KAEHLER, A. **Learning OpenCV: Computer Vision with the OpenCV Library**. O'Reilly Media, 2008. 576 p.

BRETZNER, L.; LAPTEV, I. & LINDEMBERG, T. Hand Gesture Recognition Using Multi-Scale Color Features. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 5. **Anais...** 2002. p. 405-410.

CABRAL, C. ET AL. Developing Games with Magic Playground: A Gesture-Based Game Engine. In: INTERNATIONAL CONFERENCE ON ADVANCES IN COMPUTER ENTERTAINMENT TECHNOLOGY. **Anais...** 2005.

CAGLAR, M. & LOBO, N. Open Hand Detection in a Cluttered Single Image using Finger Primitives. In: CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Anais...** 2006.

CALIFE, D. ET AL. Robot Arena. In: **Computers in Entertainment**, v. 7, 2009.

CAMURRI ET AL. Analysis of expressive gestures in human movement: the EyesWeb expressive gesture processing library. In: COLLOQUIUM ON MUSICAL INFORMATICS, 14. **Anais...** 2003.

CHEN, Q. ET AL. Navigating a 3D virtual environment of learning objects by hand gestures. In: **International Journal of Advanced Media and Communication**, v. 1, n. 4, p. 351-368. 2007.

CITY OF HEROES Homepage. Disponível em: <<http://www.cityofheroes.com/en.html>>. Acesso em: 30 jul. 2010.

CUTLER, R. & TURK, M. View-based interpretation of real-time optical flow for gesture recognition. In: INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 3. **Anais...** 1998.

DAUGMAN, G. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. In: **Journal of the Optics Society of America**, v. 2, p. 1160-1169. 1985.

DAVIS, J. & SHA, M. Visual Gesture Recognition. In: **IEEE Visual Image Signal Processing**, v. 141, n. 2, p. 101-106. 1994.

DEIMEL, B. & SCHROTER, S. Improving Hand-Gesture Recognition via Video Based Methods for the Separation of the forearm from the Human Hand. In: INTERNATIONAL GESTURE WORKSHOP. **Anais...** 1999.

DERPANIS, K. ET AL. Hand Gesture Recognition within a Linguistics-Based Framework. In: EUROPEAN CONFERENCE ON COMPUTER VISION. **Anais...** 2004. p. 282-296.

DOBNIK, V. Surgeons may err less by playing video games. 2004. Disponível em: <<http://www.msnbc.msn.com/id/4685909>>. Acesso em 30 jul. 2010.

DU, W. & LI, H. Vision based gesture recognition system with single camera. In: INTERNATIONAL CONFERENCE ON SIGNAL PROCESSING, 5. 2000. **Anais...** p. 1351-1357.

ECKEL, B. **Thinking in C++**. Prentice Hall, 2003.

EDWARDS, J. et al. Understanding Movement as Input for Interaction: a study of Two EyeToy Games. In: OZCHI, 2004. **Anais...** 2004.

ELIAS, J. ET AL. **Multi-Touch Gesture Dictionary**. US20070177803. 2007.

EPPS, J.; LICHMAN, S. & WU, M. A Study of Hand Shape Use in Tabletop Gesture Interaction. In: CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS. **Anais...** 2006. p. 748-753.

EROL, A. ET AL. A Review on Vision-Based Full DOF Hand Motion Estimation. In: IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Anais...** 2005.

FEI, H. & REID, I. Probabilistic Tracking and Recognition of Non-Rigid Hand Motion. In: IEEE INTERNATIONAL WORKSHOP ON ANALYSIS AND MODELING OF FACES AND GESTURES. **Anais...** 2003.

FIGUEIREDO ET AL. An open-source framework for air guitar games. In: BRAZILIAN SYMPOSIUM ON GAMES AND DIGITAL ENTERTAINMENT, 8. **Anais...** 2009. p. 69-75.

FILHOL, M. Sign description, how geometry and graphing serve linguistic issues. In: INTERNATIONAL CONFERENCE ON THEORETICAL ISSUES IN SIGN LANGUAGE RECOGNITION, 9. **Anais...** 2006.

FLECK, M. AND FORSYTH, D. Naked people Skin Filter. Disponível em: <<http://www.cs.hmc.edu/~fleck/naked-skin.html>>. Acesso em 30 jul. 2010.

FLOREZ, F. ET AL. Hand gesture recognition following the dynamics of a topology-preserving network. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 5. **Anais...** 2002. p. 303-308.

FOLEY, J. et al. **Computer Graphics: Principles and Practice**. 2nd. ed. Addison-Wesley, 1996. 1175 p.

FREEMAN ET AL. Computer Vision for Computer Games. In: INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 2. **Anais...** 1996. p. 100-105.

FRITSCH ET AL. The Effect of Latency and Network Limitations on MMORPGs: A Field Study of Everquest 2. In: NETGAMES '05. **Anais...** 2005.

GOMEZ, G. On selecting colour components for skin detection. In: INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION, 16. **Anais...** 2002.

GOMEZ, G. AND MORALES, E. Automatic Feature Construction and a Simple Rule Induction Algorithm for Skin Detection. In: ICML WORKSHOP ON MACHINE LEARNING IN COMPUTER VISION. **Anais...** 2002. p. 31-38.

GONCALVES ET AL. A Gesture Based Game for Image Tagging. In: CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS. **Anais...** 2008. p. 2685-2690.

GRIFFIN, S. Push. Play: An Examination of the Gameplay Button. In: DIGRA 2005 CONFERENCE: CHANGING VIEWS - WORLDS IN PLAY. **Anais...** 2005.

GUAN, H. ET AL. Appearance-based 3D Hand Pose Estimation. In: CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Anais...** 2006.

GUIARD, Y. Symmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model. In: **The Journal of Motor Behaviour**, v. 19 n. 4, p. 486-517. 1987.

HAMADA, Y.; SHIMADA, N. & SHIRAI, Y. Hand Shape Estimation under Complex Backgrounds for Sign Language Recognition. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 6. 2004. p. 589-594.

HAMNOSYS. Disponível em: <<http://www.sign-lang.uni-hamburg.de/projects/HamNoSys.html>>. Acesso em 30 jul. 2010.

HANKE, T. (ed.). **ViSiCAST Deliverable D5-1: Interface Definitions**. 2001. (Relatório Técnico D5-1).

HARDENBERG, C. & BÉRARD, F. Bare-Hand Human-Computer Interaction. In: WORKSHOP ON PERCEPTUAL USER INTERFACES. **Anais...** 2001.

HENDERSON, T. AND BHATTI, S. Networked Games – a QoS-Sensitive Application for QoS-Insensitive Users? In: ACM SIGCOMM. **Anais...** 2003.

HONG, P.; TURK, M.; & HUANG, T. Gesture modeling and recognition using finite state machines. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE GESTURE RECOGNITION, 4. **Anais...** 2000. p. 410–415.

HONG, T. Shoot to Thrill. In: **Game Developer**, v. 15, n. 9, p. 21-28. 2008.

IBISWORLD. **Video Games U.S. Industry Report**. 2009.

IMAI, A. ET AL. 3-D Hand Posture Recognition by Training Contour Variation. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE GESTURE RECOGNITION, 6. **Anais...** 2004. p. 895-900.

ISARD, M. & BLAKE, A. Condensation - Conditional Density Propagation For Visual Tracking. In: **International Journal of Computer Vision**, v. 28, n. 1, p. 5-28. 1998a.

ISARD, M., BLAKE, A. ICondensation: Unifying low-level and high-level tracking in a stochastic framework. In: EUROPEAN CONFERENCE ON COMPUTER VISION, 5. **Anais...** 1998b.

ISHIHARA, T. & OTSU, N. Gesture Recognition Using Auto-Regressive Coefficients of Higher-Order Local Auto-Correlation Features. In: AUTOMATIC FACE AND GESTURE RECOGNITION, 6. **Anais...** 2004. p. 583-588.

ISHII, H. & ULLMER, B. Tangible Bits: Towards Seamless Interfaces Between People, Bits And Atoms. In: SIGCHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS. **Anais...** 1997. p. 234-241.

ISO/IEC. International Standard 14977. Information technology: Syntactic meta-language: Extended BNF. Geneva, 1996.

JACOBSON, I.; Booch, G. & Rumbaugh, J. **The Unified Software Development Process**. Addison-Wesley, 1999. 463 p.

JAIMES, A. & LIU, J. Hotspot Components for Gesture-Based Interaction. In: HUMAN-COMPUTER INTERACTION - INTERACT. Springer, 2005. p. 1062-1066.

JAIMES, A. & SEBE, M. Multimodal human-computer interaction: A survey. In: **Computer Vision and Image Understanding** v. 108, p. 116-134. 2007.

JEDYNAK, B. ET AL. **Maximum entropy models for skin detection**. 2002. (Relatório Técnico XIII, Université des Sciences et Technologies de Lille).

JENNINGS, C. Robust Finger Tracking with Multiple Cameras. In: INTERNATIONAL WORKSHOP ON RECOGNITION, ANALYSIS, AND TRACKING OF FACES AND GESTURES IN REAL-TIME SYSTEMS. **Anais...** 1999.

JENNINGS, C. & MURRAY, D. Gesture Recognition for Robot Control. In: **Computer Vision And Pattern Recognition**. 1997.

JOHNSON, R. & FOOTE, B. Designing reusable classes. In: **Journal of object-oriented programming**, v. 1, n. 2, p. 22-35. 1988.

JONES, M. & REHG, J. Statistical Color Models with Application to Skin Detection. In: CVPR. **Anais...** 1999. p. 274-280.

JUST, A.; BERNIER, O. & MARCEL, S. Recognition of Isolated Complex Mono- and Bi-Manual 3D Hand Gestures. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 6. **Anais...** 2004. p. 571-576.

KANE, B. Beyond the Gamepad panel session. 2005. Disponível em: <http://www.gamasutra.com/features/20050819/kane_01.shtml>. Acesso em 30 jul. 2010.

KANG, H.; LEE, C. & JUNG, K. Recognition-based gesture spotting in video games. In: **Pattern Recognition Letters**, v. 25, n. 15, p. 1701-1714. 2004.

KARAM, M. & SCHRAEFEL, M. A study on the use of semaphoric gestures for secondary task interactions. In: CHI '05. **Anais...** 2005. p. 1961-1964.

KARAM, M. & SCHRAEFEL, M. Investigating User Tolerance for Errors in Vision-Enabled Gesture-Based Interactions. In: AVI. **Anais...** 2006. p. 225-232.

KIM, H. ET AL. Calling Motion and Natural Hand Detection for Gesture Recognition. In: SICE-ICASE INTERNATIONAL JOINT CONFERENCE. **Anais...** 2006. p. 313-316.

KOLSCH, ET AL. Vision-based Interfaces for Mobility. In: INTERNATIONAL CONFERENCE ON MOBILE AND UBIQUITOUS SYSTEMS. **Anais...** 2004.

LEE, H. & KIM, J. An HMM-Based Threshold Model Approach for Gesture Recognition. In: **IEEE Transactions On Pattern Analysis And Machine Intelligence**, v. 21, n. 10, p. 961-973. 1999.

LENMAN, S.; BRETZNER, L. & THURESSON, B. Using Marking Menus to Develop Command Sets for Computer Vision Based Hand Gesture Interfaces. In: NordiCHI. **Anais...** 2002. p. 239-242.

LICHTENAUER, J.; HENDRIKS, E. & REINDERS, M. Learning to Recognize a Sign from a Single Example. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 8. **Anais...** 2008.

LIDDELL, S. & JOHNSON, R. American Sign Language: The Phonological Base. In: **Sign Language Studies**, v. 64 p. 195-278. 1989.

LIU, X. & FUJIMURA, K. Hand Gesture Recognition using Depth Data. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 6. **Anais...** 2004. p. 529-534.

LIU, Y., JIA, Y. A Robust Hand Tracking and Gesture Recognition Method for Wearable Visual Interfaces and its Applications. In: 3RD INTERNATIONAL CONFERENCE ON IMAGE AND GRAPHICS, 3. **Anais...** 2004.

LYONS, H. ET AL. Gart: The gesture and activity recognition toolkit. In: INTERNATIONAL CONFERENCE ON HUMAN-COMPUTER INTERACTION, 12. **Anais...** 2007.

MACLEAN, J. ET AL. Fast Hand Gesture Recognition for Real-Time Teleconferencing Applications. In: IEEE ICCV WORKSHOP ON RECOGNITION, ANALYSIS, AND TRACKING OF FACES AND GESTURES IN REAL-TIME SYSTEMS. **Anais...** 2001.

MAGGIONI, C. & KAMMERER, B. GestureComputer: New Ways of Operating a Computer. In: **Computer Vision for Human-Machine Interaction**, Cambridge University Press, 1998.

MALKAWI, A. & SRINIVASAN, R. Multimodal human-computer interaction for immersive visualization: integrating speech-gesture recognition and augmented reality for indoor environments. In: INTERNATIONAL ASSOCIATION OF SCIENCE

AND TECHNOLOGY FOR DEVELOPMENT CONFERENCE ON COMPUTER GRAPHICS AND IMAGING. **Anais...** 2004.

MAMMEN, J.; CHAUDHURI, S. & AGARWAL, T. A Two-stage Scheme for Dynamic Hand Gesture Recognition. In: NATIONAL CONFERENCE ON COMMUNICATION. **Anais...** 2002.

MAMMEN, J.; CHAUDHURI, S. & AGARWAL, T. Simultaneous Tracking of Both Hands by Estimation of Erroneous Observations. In: BRITISH MACHINE VISION CONFERENCE. **Anais...** 2001.

MAPES, D., MOSHEL, J. A Two Handed Interface for Object Manipulation in Virtual Environments. In: **Presence: Teleoperators and Virtual Environments**, v. 4 n. 4, p. 403-416. 1995.

MARCEL, S. **Gestures for Multi-Modal Interfaces**: A Review. IDIAP, 2002.

MARCEL, S. ET AL. Hand Gesture Recognition using Input-Output Hidden Markov Models. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE GESTURE RECOGNITION, 4. **Anais...** 2000. p. 456-461.

MILLER, G. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. In: **The Psychological Review**, v. 63, p. 81-97. 1956.

MITRA, S. & ACHARYA, T. Gesture Recognition: A Survey. In: **IEEE Transactions on Systems, Man, and Cybernetics-part C: Applications and Reviews**, v. 37, n. 3, p. 311-324. 2007.

MO, Z. **Gesture Interface Engine**. 2007. Tese (Doutorado) - University Of Southern California.

MO, Z. & NEUMANN, U. Real-time Hand Pose Recognition Using Low-Resolution Depth Images. In: IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Anais...** 2006.

NAKAMURA, R., TORI, R. Improving Collision Detection for Real-Time Video Avatar Interaction. In: SYMPOSIUM ON VIRTUAL AND AUGMENTED REALITY. **Anais...** 2008. p. 105-114.

NCSoft **financial report**. 2008. Disponível em: <<http://www.ncsoft.net/global/ir/overview.aspx>>. Acesso em 30 jul. 2010.

NGUYEN, T.; BINH, N. & BISCHOF, H. An active boosting-based learning framework for real-time hand detection. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 8. **Anais...** 2008.

NIELSEN ET AL. A Procedure for Developing Intuitive and Ergonomic Gesture Interfaces for Man-Machine Interaction. In: INTERNATIONAL GESTURE WORKSHOP, 5. **Anais...** 2003.

NIJHOLT, A. & HEYLEN, D. Multimodal communication in inhabited virtual environments. In: **International Journal of Speech Technology**, p. 343–354. 2002.

NORMAN, D. & DRAPER, S. **User Centered System Design**: New Perspectives on Human-computer Interaction. CRC, 1986. 544 p.

NOVINT. Novint Falcon. Disponível em: <http://home.novint.com/products/novint_falcon.php>. Acesso em 30 jul. 2010.

O'HAGAN, R. ET AL. Visual Gestures for Virtual Environments. In: **Interacting with Computers**, v. 14, p. 231-250. 2002.

ONG, E. & BOWDEN, R. A Boosted Classifier Tree for Hand Shape Detection. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE GESTURE RECOGNITION, 6. **Anais...** 2004. p. 889-894.

OPENCV. Disponível em: <<http://sourceforge.net/projects/opencvlibrary/>>. Acesso em 30 jul. 2010.

OVIATT, S. Ten Myths of Multimodal Interaction. In: **Communications of the ACM**, v. 42, n. 11, p. 74-81. 1999.

PAGGIO, D. & JONGEJAN, B. Multimodal communication in the virtual farm of the staging Project. In: **Multimodal Intelligent Information Presentation**. Kluwer Academic Publishers, 2005. p. 27–46.

PARK, C.; ROH, M. & LEE, S. Real-Time 3D Pointing Gesture Recognition in Mobile Space. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 8. **Anais...** 2008.

PAULA ET AL. Camera Kombat: Interação Livre para Jogos. In: BRAZILIAN SYMPOSIUM ON GAMES AND DIGITAL ENTERTAINMENT, 5. **Anais...** 2006.

PAVLOVIC ET AL. Visual Interpretation of Hand Gestures for Human Computer Interaction: A Review. In: **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 19, n. 7, p. 677-695. 1997.

PAYNE, J. ET AL. Gameplay issues in the design of spatial 3D gestures for video games. In: CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS. **Anais...** 2006. p. 1217-1222.

PEIXOTO, P. & CARREIRA, J. A Natural Hand Gesture Human Computer Interface Using Contour Signatures. In: INTERNATIONAL CONFERENCE ON HUMAN-COMPUTER INTERACTION, 11. **Anais...** 2005.

QUEK, F. Towards a vision-based hand gesture interface. In: VIRTUAL REALITY SOFTWARE AND TECHNOLOGY CONFERENCE. **Anais...** 1994.

QUEK, F. ET AL. Multimodal Human Discourse: Gesture and Speech. In: **ACM Transactions on Computer-Human Interaction**, v. 9 n. 3, p. 171-193. 2002.

RAJKO, S. & QIAN, G. HMM Parameter Reduction for Practical Gesture Recognition. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 8. **Anais...** 2008.

RAMAMOORTHY, A. ET AL. Recognition of dynamic hand gestures. In: **Pattern Recognition**, v. 36. p. 2069-2081. 2003.

READ, A. Successive revisions in the explanation of "O.K.". In: **American Speech**, v. 39, n. 4, p. 243-267. 1964.

REHG, J. & KANADE, T. Digiteyes: Vision-based hand tracking for human-computer interaction. In: WORKSHOP ON MOTION OF NON-RIGID AND ARTICULATED BODIES. **Anais...** 1993. p. 16-24.

ROWLEY, D. & RHOADES, D. The cognitive jogthrough: a fast-paced user interface evaluation procedure. In: CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS. **Anais...** 1992. p. 389-395.

SALLES, J. ET AL. O.G.R.E. - Open Gestures Recognition Engine. In: BRAZILIAN SYMPOSIUM ON COMPUTER GRAPHICS AND IMAGE PROCESSING, 17. **Anais...** 2004.

SCHLATTMANN, M. ET AL. Markerless 4 gestures 6 DOF real-time visual tracking of the human hand with automatic initialization. In: **Computer Graphics Forum**, v. 26, n. 3. 2007.

SCLAROFF, S. ET AL. Tracking, Analysis, and Recognition of Human Gestures in Video. In: INTERNATIONAL CONFERENCE ON DOCUMENT ANALYSIS AND RECOGNITION, 8. **Anais...** 2005.

SHAMAIE, A. & SUTHERLAND, A. Accurate Recognition of Large Number of Hand Gestures. In: IRANIAN CONFERENCE ON MACHINE VISION AND IMAGE PROCESSING, 2. **Anais...** 2003. p. 13-15

SHAMAIE, A. & SHUTERLAND, A. Bayesian Fusion of Hidden Markov Models for Understanding Bimanual Movements. In: AUTOMATIC FACE AND GESTURE RECOGNITION, 4. **Anais...** 2004. p. 602-607.

SHIMADA ET AL. Real-time 3-D Hand Posture Estimation based on 2-D Appearance Retrieval Using Monocular Camera. In: INTERNATIONAL WORKSHOP ON RECOGNITION, ANALYSIS AND TRACKING OF FACES AND GESTURES IN REAL-TIME SYSTEMS, 2. **Anais...** 2001. p. 23-30.

SHNEIDERMAN, B. **Designing the user interface**: strategies for effective human-computer interaction. Addison Wesley, 3. ed., 1998.

SNIDER, M. Microsoft unveils hands-free gaming. In USA Today, 1 Jun. 2009. Disponível em: <http://www.usatoday.com/tech/gaming/2009-06-01-hands-free-microsoft_N.htm>. Acesso em 30 jul. 2010.

SORIANO, M ET AL. Adaptive skin color modeling using the skin locus for selecting training pixels. In: **Pattern Recognition**, v. 36, n. 3, p. 681-690. 2003.

SOUZA, N. ET AL. Animando Humanos Virtuais em Tempo-Real usando Visão Computacional e Redes Neurais. In: SYMPOSIUM ON VIRTUAL AND AUGMENTED REALITY, 8. **Anais...** 2006.

SMITH, J. A Comparision of RUP and XP. Rational Software, 2002. (White Paper).

STARNER, T. ET AL. Mind-Warping. In: ACM SIGCHI ADVANCES IN COMPUTER ENTERTAINMENT. **Anais...** 2004. p. 256-259.

STARNER, T. & PENTLAND, A. Visual Recognition of American Sign Language Using Hidden Markov Models. In: AUTOMATIC FACE AND GESTURE RECOGNITION. **Anais...** 1995. p. 189-194.

STOKOE, W. Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf. In: **Studies in linguistics: Occasional papers**, v. 8. 1960.

SUK, H.; SIN, B. & LEE, S. Recognizing Hand Gestures using Dynamic Bayesian Network. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE & GESTURE RECOGNITION, 8. **Anais...** 2008.

SWARTOUT, W. & VAN LENT, M. Making game of a system design. In: **Communications of the ACM**, v. 46, n. 7, p. 32-39. 2003.

TERRILLON, J. ET AL. Robust Face Detection and Hand Posture Recognition in: Color Images for Human-Machine Interaction. In: International Conference on Pattern Recognition, 16. **Anais...** 2002.

TIDIA - Aprendizagem Eletrônica homepage. Disponível em: <<http://tidia-ae.incubadora.fapesp.br>>. Acesso em 30 jul. 2010.

TOKUNAGA, D. ET AL. GESKTOP 3D: Explorando novas formas de interação 3D baseadas em gestos. In: Symposium on Virtual and Augmented Reality, 10. **Anais...** 2008. p. 125-128.

TOKUNAGA, D. et al. Video-Based Microfacet-Billboard Avatar for Educational Immersive Teleconference Systems. In: SYMPOSIUM ON VIRTUAL AND AUGMENTED REALITY, 11. **Anais...** 2009. p. 199-209.

TORI, R. ET AL. Design de Interação para um Atlas Virtual de Anatomia Usando Realidade Aumentada e Gestos. In: INTERACTION SOUTH AMERICA. **Anais...** 2009.

TORI, R. **Educação sem distância**: as tecnologias interativas na redução de distâncias em ensino e aprendizagem. Senac São Paulo e Escola do Futuro da USP, 2010.

TRIESCH, J. & MALSBURG, C. A System for Person-Independent Hand Posture Recognition against Complex Backgrounds. In: **IEEE Transactions On Pattern Analysis And Machine Intelligence**, v. 23, n. 12. 2001.

TSE ET AL. Multimodal Multiplayer Tabletop Gaming. In: **ACM Computers in Entertainment**, v. 5, n. 2. 2007.

TSUDA, F. ET AL. Integration of jARToolKit and enJine: extending with AR the potential use of a didactic game engine. In: SYMPOSIUM ON VIRTUAL AND AUGMENTED REALITY, 9. **Anais...** 2007. p. 51-69.

TURK, M. Gesture Recognition. 1998. Disponível em: <<http://citeseer.ist.psu.edu/564810.html>>. Acesso em 30 jul. 2010.

TURK, M. & KOLSCH, M. Perceptual interfaces. In: MEDIONI, G. & KANG, S. (ed.), **Emerging Topics in Computer Vision**. Prentice Hall, 2004.

UTSUMI, A.; TETSUTANI, N. & IGI, S. Hand detection and tracking using pixel value distribution model for multiple-camera-based gesture interactions. In: KNOWLEDGE MEDIA NETWORKING. **Anais...** 2002. p. 31-36.

VERTEGAAL, R. Attentive User Interfaces. In: **Communications of the ACM**, v. 46, n. 3, p. 30-33. 2003.

VEZHNEVETS ET AL. A Survey on Pixel-Based Skin Color Detection Techniques. In: GRAPHICON. **Anais...** 2003. p. 85-92.

VIGGIANO, J., Comparison of the accuracy of different white balancing options as quantified by their color constancy. In SPIE 5301. **Anais...** 2004. p. 323-333.

VIOLA, P. & JONES, M. Rapid Object Detection using a Boosted Cascade of Simple Features. In: CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Anais...** 2001.

WANG, X.; ZHANG, X. & DAI, D. Tracking of Deformable Human Hand in Real Time as Continuous Input for Gesture-based Interaction. In: IUI'07. **Anais...** 2007. p. 235-242.

WELSH, G. & BISHOP, G. **An introduction to the Kalman filter**. 1995. (Relatório Técnico TR95-041, University of North Carolina).

WILSON, A. & Bobick, A. Parametric Hidden Markov Models for Gesture Recognition. In: **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 21, n. 9. 1999.

WOBBROCK ET AL. Gestures without Libraries, Toolkits or Training: A \$1 Recognizer for User Interface Prototypes. In: ACM SYMPOSIUM ON USER INTERFACE SOFTWARE AND TECHNOLOGY. **Anais...** 2007.

WU, Y., HUANG, T. Capturing Articulated Human Hand Motion: A Divide-and-Conquer Approach. In: IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION. **Anais...** 1999. p. 606-611.

WU, Y., LIN, J. & HUANG, T. Capturing Natural Hand Articulation. In: INTERNATIONAL CONFERENCE ON COMPUTER VISION, 8. **Anais...** 2001. p. 426-432.

YANG, M. & AHUJA, N. Extraction and Classification of Visual Motion Patterns for Hand Gesture Recognition. In: COMPUTER VISION AND PATTERN RECOGNITION. **Anais...** 1998. p. 892-897.

YOON, H. ET AL. Hand gesture recognition using combined features of location, angle and velocity. In: **Pattern Recognition**, v. 34, p. 1491-1501. 2001.

YOON, T. ET AL. Image Segmentation of Human Forearms in Infrared Image. In: IEEE EMBS ANNUAL INTERNATIONAL CONFERENCE, 28. **Anais...** 2006. p. 2762-2765.

YOUNGBLUT, C. **Educational Uses of Virtual Reality Technology**. 1998. (Relatório Técnico D-2128, Institute for Defense Analysis).

APÊNDICE A - HISTÓRICO DO DESENVOLVIMENTO

Desde o início da pesquisa, o desejo de trabalhar com reconhecimento de gestos, ambientes aumentados ou virtuais, jogos e educação está presente, pelas motivações descritas na introdução, inclusive a de que esse trabalho se integre aos outros trabalhos do Interlab. De posse desse "esboço" de tema, uma revisão bibliográfica preliminar verifica a sua viabilidade. A partir daí o trabalho, ilustrado pela , se inicia de fato. Na Figura 24, as setas representam a sequência das atividades e também a influência de umas sobre as outras.

Figura 24 - Metodologia de trabalho

As atividades começam com uma revisão bibliográfica mais completa que a feita inicialmente. Os primeiros temas a pesquisar são técnicas de reconhecimento de gestos manuais e suas aplicações, não só por centrais para o trabalho, mas também aqueles com os quais se tinha menor experiência. Essa revisão se inicia por *surveys* sobre o assunto e se estende inclusive para temas, como reconhecimento de línguas de sinais ou de gestos como apoio à fala, além do escopo do trabalho, só definido com precisão numa etapa posterior. Paralelamente também são pesquisados jogos eletrônicos explorando novas formas de interação, em particular RA (Bernardes et al., 2008). Durante esse período são realizados também, juntamente com alunos de graduação, projetos explorando o uso de gestos com o HandVu em um ambiente de *Desktop* 3D (Tokunaga et al., 2008), sua detecção com acelerômetros (Bernardes et al., 2009) e jogos com RA (Tsuda et al., 2007). Esses projetos exploratórios influenciam o trabalho daí em diante, mas em geral de forma indireta.

A pesquisa sobre reconhecimento de gestos naturalmente leva ao projeto do *framework* do sistema, uma arquitetura mais geral e abstrata para esse reconhecimento. Leva também a uma pesquisa mais detalhada sobre o uso de gestos em sistemas de interface, em particular para interação 3D. A necessidade dessa pesquisa foi sentida inclusive em alguns dos projetos exploratórios. É somente depois dela que se define de forma mais específica a aplicação e os requisitos do sistema e se pode passar a implementação.

A implementação segue uma customização do Processo Unificado (Jacobson et al., 1999), já descrita na introdução. Após o projeto do *framework*, instancia-se uma implementação específica dele, para o domínio desse trabalho e incluindo o levantamento dos requisitos descritos anteriormente. A fase de construção dessa implementação é dividida em 5 interações, cada uma contando com desenvolvimento de protótipos, integração no *framework*, testes e refinamento.

A Figura 25 mostra os dez casos de uso definidos inicialmente para essa implementação e com base nos requisitos. Além dos casos mostrados, durante os testes realizados na quinta interação detectou-se a necessidade de acréscimo de mais dois, ligados ao desenvolvedor: permitir que modifique em tempo real a tolerância para o reconhecimento de posturas, com base nos conjuntos de gestos usados em cada contexto, e fornecer ao desenvolvedor o erro entre duas posturas,

de forma que possa prever eventuais problemas com conjuntos de gestos escolhidos.

Figura 25 - Casos de uso iniciais

Como a Figura 25 indica, a maior parte desses casos de uso descreve funcionalidades para o desenvolvedor, o usuário que tem acesso mais direto ao sistema, definindo assim sua interface de programação. Os últimos cinco casos, do ponto de vista do ator usuário, representam funcionalidades mínimas que uma aplicação que utilize o sistema deveria implementar em sua interface e os casos sete e oito, em particular, são bastante dependentes da aplicação.

Do ponto de vista do desenvolvedor, a inicialização consiste na instanciação e inicialização dos objetos do sistema e da passagem de alguns parâmetros, como número de câmeras e tamanho das imagens. A definição dos gestos possíveis difere da inicialização pois, apesar de dever ser feita antes do primeiro reconhecimento, esses gestos podem ser redefinidos a qualquer momento, por exemplo devido a uma mudança de contexto da aplicação. O reconhecimento de gestos é o principal objetivo do trabalho. O caso cinco já foi apresentado juntamente com a descrição do módulo de análise e consiste em definir quais dentre os objetos de interesse, uma

das mãos ou a face, tem prioridade na detecção quando nem todas estão presentes na imagem. O seis também já foi descrito juntamente com o módulo Desc2Input.

Como comentado anteriormente, a necessidade do *feedback* visual só foi confirmada depois de alguns experimentos e esse foi um dos últimos casos de uso elaborados antes da quinta interação. Os casos nove e dez também só foram elaborados mais tarde, após os primeiros testes de segmentação na primeira interação, quando sua necessidade tornou-se aparente e, assim como os casos cinco e seis, já foram descritos em mais detalhe juntamente com a discussão da segmentação. Em sua maior parte, esses casos de uso do ponto de vista do desenvolvedor são bastante simples: correspondem a chamadas de função que retornam ponteiros nulos quando ocorre algum erro (como por exemplo se não for encontrada uma câmera ou arquivo de vídeo na inicialização), onde o único participante é o desenvolvedor, com a única pré-condição de utilizar objetos já inicializados, sem variações importantes. Por isso não foram expandidos de maneira mais completa ou formal, mas simplesmente listados e descritos informalmente como mostrado acima.

Após a definição dos casos de uso, na primeira interação é desenvolvido o módulo de segmentação de regiões de interesse e a segunda é responsável pela análise e reconhecimento de posturas das mãos. Houve uma certa sobreposição entre essas duas primeiras interações. Na terceira interação são reconhecidos gestos definidos por vários fatores, mas ainda sem reconhecimento dos movimentos, desenvolvido somente na quarta interação. A quinta é uma interação de integração com outros aplicativos e refinamento global: o conjunto de posturas utilizado é expandido e são feitas melhorias e correções no sistema, como por exemplo o acréscimo da funcionalidade de ajustar a tolerância do reconhecimento de posturas. A confecção desta tese acontece em paralelo com as outras atividades e não é mostrada na Figura 24 para não poluí-la, visto que é influenciada e influencia todas elas. Um cronograma mostrando a ocorrência e duração de todas essas atividades ao longo da duração desse trabalho é apresentado no Apêndice A.

Uma das maiores alterações no projeto durante a fase de construção foi no reconhecimento de posturas. Durante toda a segunda interação da construção do sistema, responsável por implementar esse reconhecimento, não havia sido detectada a necessidade de fazer alinhamento algum entre os vetores do contorno, o que não foi feito nos protótipos ou testes iniciais. Isso se deve a dois fatores. Em

primeiro lugar, o algoritmo que identifica os componentes conexos o faz varrendo a imagem da esquerda para a direita e de cima para baixo, ou seja, na direção positiva de seus eixos. Além disso, um grande número de posturas tem um ponto mais alto bem definido na sua orientação padrão, mostrada na Figura 2, que equivale a ponta de um dedo. Assim, esses pontos mais altos bem definidos são sempre posicionado na primeira posição nos vetores, que ficam alinhados por eles automaticamente, como consequência do algoritmo que identifica os componentes conexos. Sem fazer o alinhamento dos vetores, em muitos casos a orientação da postura não era nem mesmo necessária para classificar posturas que diferissem somente por ela (o próprio erro entre elas era suficiente para isso, em grande parte dos casos) e era usada principalmente para acelerar o reconhecimento de postura, para evitar que o erro fosse calculado caso a diferença das rotações fosse maior que a tolerância. Além disso, a orientação era calculada com base no ponto mais alto e não no ponto de máximo.

Somente durante a quinta e última interação, quando se expandiu o número de posturas usadas (inclusive devido a necessidades levantadas para integração com outros projetos) é que o máximo global no vetor de distâncias entre o ponto de contorno e o centro da mão foi utilizado para alinhar os vetores e definir a orientação da postura. Além de uma grande melhora no reconhecimento de posturas sem um ponto mais alto bem definido e de uma pequena melhora mesmo no reconhecimento das outras posturas, uma vantagem de definir a orientação com base no ponto de máximo é que, para posturas cuja diferença seja somente a orientação, ela varia de forma mais contínua conforme a mão é rotacionada, enquanto o ponto mais alto pode variar consideravelmente. Assim, a rotação da postura pode ser melhor utilizada como parâmetro, por exemplo na tarefa de manipulação de objetos.

A.1 RESULTADOS DOS TESTES PRELIMINARES

Durante o teste em que se pediu a quatro usuários que fizessem gestos livremente na direção da câmera e lhes atribuíssem significados, foram listados os seguintes gestos (todas as posturas referidas por números correspondem às mostradas na Figura 2).

Utilizando somente posturas, todas realizadas com a mão direita, foram citados:

- Zero, um, dois, três, quatro e cinco: usando, respectivamente, as posturas 4, 1, 8, 9, 10 e 2.
- Ficar parado (usado por soldados e em jogos, durante manobras silenciosas): usando a postura 4, na altura da cabeça.
- Apontar (para cima, e para os dois lados) foi citado por diversos usuários, usando a postura 1. Apontar para baixo usou essa postura espelhada, mostrando as costas da mão.
- "V de vitória" foi outro significado atribuído à postura 8.
- "Pare!" foi atribuído a duas posturas: 2 e 3.
- Positivo: usando a postura 11. Espelhada para baixo, essa postura indicou reprovação ou condenação.
- Pedir carona: também usando a postura 11, com o polegar apontando para a direita.
- Visto: dois usuários usaram a postura 5, inclinada, com esse significado. Um deles a utilizou espelhada, mostrando as costas da mão.
- "Looser" foi outro significado atribuído à postura 5, mostrada em frente à testa.
- Garra: postura 16.
- OK: postura 12.
- Heavy Metal: postura 13.
- Hang Loose: postura 14. Telefone ou telefonar foi outro significado a essa postura, espelhada para mostrar as costas da mão e rotacionada com o polegar apontando para cima, ao lado da cabeça.
- "Vida longa e próspera" (cumprimento de Vulcanos em Jornada nas Estrelas): postura 15.
- A postura 17 foi citada por vários usuários como a letra i e usada para sinalizar a letra j em língua de sinais (curiosamente, foi a única menção a essas línguas), e também como um gesto obsceno na Austrália, popularizado por uma campanha publicitária.
- "I Love You" foi um significado atribuído à postura com os dedos mínimo, indicador e polegar estendidos. Um usuário também mencionou que tinha "significância mística"

- Luneta (ou câmera) foi um significado atribuído a uma postura com os dedos curvados e com a ponta do polegar tocando a ponta de um deles, com os dedos e a palma da mão formando um círculo aproximado no plano da imagem.
- Um gesto obsceno com o dedo médio erguido foi feito por todos os usuários.

Gestos dinâmicos foram realizados iniciando com as seguintes posturas:

- Postura 4: Soco (movimento para frente), Puxar (movimento para trás), Mudar Marcha (movimento, para frente, para trás ou para os lados, mas com a mão num local abaixo do peito).
- Postura 1: Não (girando em torno do pulso para um lado e para o outro), Loucura (apontando para a cabeça e girando em torno do pulso fazendo um círculo), Continuar (como o gesto anterior, mas à frente do corpo e abaixo do peito), Morte (apontando para a esquerda e movendo a mão para a direita na altura do pescoço).
- Postura 8: Tesoura ou Cortar (fechando os dois dedos. Um usuário também fez esse gesto com a postura espelhada, mostrando as costas da mão e apontando e movendo-se para a esquerda, um dos gestos que cruzou para o outro lado do corpo).
- Postura 2 - Tchau (girando em torno do pulso para um lado e para o outro), Empurrar (movimento para frente), Limpar (movimento circular), Pegar (fechando a mão), Parar Música (como o movimento de pegar, mas movendo a mão em um círculo no sentido horário).
- Postura 3 - Os mesmos dois primeiros significados e movimentos listados para a postura 2.
- Postura 6 - Golpe de Karatê (movimento para frente).
- Postura 11 - Pedir Carona (como no gesto estático com o mesmo nome, mas movendo a mão para a direita).
- Postura 5 - Atirar 1 (apontando para a esquerda e recolhendo o polegar), Atirar 2 (apontando para a esquerda e fazendo um rápido e breve giro em torno do pulso no sentido horário, imitando o recuo do tiro).
- Postura 7 - Pegar (fazendo com que o polegar toque as pontas dos dedos).

Foram utilizados ainda os seguintes gestos bimanuais:

- Zoom in e zoom out: usando a postura 5, com as mãos formando os cantos inferior direito e superior esquerdo de um quadrilátero e se afastando e se aproximando, respectivamente (embora um dos usuários tenha invertido esse sentido).
- Aumentar e diminuir: mãos na postura 6 com as palmas voltadas uma para a outra, se afastando e se aproximando, respectivamente.
- Esticar e encolher: uma variação do gesto anterior (até os significados podem ser sinônimos), com as mãos fechadas.
- Virar para a esquerda ou para a direita: os dois punhos fechados, e estendidos à frente do corpo, girando em torno do centro entre eles no sentidos anti-horário e horário, respectivamente.
- Indicação que alguém será prejudicado e gesto obsceno: A mão esquerda aberta sobre a direita fechada e batendo no lado esquerdo do punho com a palma.
- Ameaça ou violência: a mão direita dando um soco na palma da esquerda.
- Bater palmas (aprovação).

Nos testes preliminares direcionados a gestos associados a ações do CoH, foram definidos os seguintes gestos para cada uma:

- Selecionar o próximo alvo: gesto de pedir carona (a versão com movimento) com a mão direita.
- *Accelerate Metabolism* (AM): duas mãos abertas com os dedos unidos se movendo lateralmente para longe do corpo, na altura do peito.
- *Hasten*: duas mãos abaixo do peito, fechadas, movendo-se lateralmente para longe do corpo.
- *Radiation Infection* (RI): mão esquerda fechada, começando abaixo do peito e movendo-se para cima, sem movimento da mão direita.
- *Enervating Field* (EV): duas mãos fechadas, começam acima da cabeça e se movem para baixo.
- *Neutrino Bolt* (NBolt): mão direita aberta com os dedos unidos movendo-se para frente, sem movimento da outra mão.
- *X-Ray Beam* (XRB): duas mãos abaixo do peito, fechadas, movendo-se lateralmente para perto do corpo.

- *Cosmic Burst* (CB): mão esquerda aberta com os dedos unidos movendo-se para frente, sem movimento da outra mão.
- *Aim*: duas mãos abaixo do peito, abertas com dedos unidos e apontando para baixo, movendo-se lateralmente para longe do corpo.
- *Irradiate*: duas mãos fechadas se movendo lateralmente para longe do corpo, na altura do peito.
- *Neutron Bomb* (NBomb): duas mãos abertas com dedos unidos, começam acima da cabeça e se movem para baixo.
- *Soul Drain*: duas mãos abaixo do peito, fechadas, movendo-se para cima.
- *Dark Consumption* : duas mãos abertas com os dedos unidos e as palmas voltadas para o corpo, movendo-se para trás em sua direção.

A.2 RESULTADO DOS TESTES COM O CoH

Durante o passo-a-passo cognitivo com o uso de gestos integrado a esse aplicativo, o usuário experiente respondeu "sim" a todas as perguntas. As perguntas "O feedback visual auxilia na realização do gesto?" e "A resposta ao gesto foi a esperada?" também foram respondidas simplesmente com "sim" todas as vezes pelo usuário inexperiente. Abaixo seguem suas respostas às outras duas perguntas, para cada um dos passos do teste, listados em negrito.

AM

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim. No treino confundi com *Irradiate*, mas como *Irradiate* é ofensivo, associei com a mão fechada e esse com a mão aberta.

Hasten

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Mais ou menos, tem três poderes com um gesto bem parecido.

Seleção com ponteiro

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim, é como se eu pegasse o inimigo com a mão!

NB e movimento para trás

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim, só foi difícil lembrar que precisava me mover rápido para trás.

RI

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim.

EF

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim.

CB

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim.

XRB

Você lembra qual gesto deve usar?

Não.

Foi fácil lembrar desse gesto?

Não, como esse poder são raios que saem dos olhos, foi difícil lembrar que movimento de mão estava associado a eles. Só depois da dica e de ver o personagem fazendo o movimento lembrei do porquê.

NBolt

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim.

Selecionar próximo alvo (repetido por duas vezes)

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim, mas não sei porque.

Aim

Você lembra qual gesto deve usar?

Sim

Foi fácil lembrar desse gesto?

Mais ou menos, tem três poderes com um gesto bem parecido.

Soul Drain

Você lembra qual gesto deve usar?

Não.

Foi fácil lembrar desse gesto?

Não, pois não tinha prestado atenção no personagem quando fez da última vez. Depois me lembrei que era o gesto em que as mãos se cruzavam, como comentamos naqueles outros testes faz tempo, e foi mais fácil de lembrar.

Irradiate

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim. No treino confundi com AM, mas como esse é ofensivo, associei com a mão fechada.

NBomb

Você lembra qual gesto deve usar?

Sim.

Foi fácil lembrar desse gesto?

Sim, esse foi o gesto e o poder mais legal.

Por fim, com relação aos testes de movimento no CoH, usando o tapete e feitos com três usuários. Um desses usuários têm experiência com ambientes 3D, as outras duas são do sexo feminino, uma tem mais de 60 anos de idade e nenhuma tem experiência alguma com ambientes 3D. Foi pedido aos três que realizassem um percurso determinado com o personagem, primeiro utilizando o tapete e depois o teclado. Coincidentemente, o usuário experiente sofria de uma crise aguda de tendinite durante esse teste e não pode realizar o teste com o teclado, mas ainda assim opinou que o movimento com o tapete foi "um pouco mais lento mas mais divertido" do que o que costuma fazer usando o teclado. Ambas as usuárias inexperientes cometeram mais erros durante os testes, tanto com o teclado quanto com o tapete, e opinaram que a navegação utilizando o tapete foi mais fácil que com o teclado. A usuária com mais de sessenta anos se queixou que, ao preferir não pular nas diagonais do tapete para fazer com que o personagem pulasse, tinha seu

movimento prejudicado, pois ao pisar primeiro em uma diagonal e depois na outra, o personagem começava a se virar antes de pular. Essa mesma usuária comentou que a navegação pelo teclado seria impossível durante um longo período, por ter artrite nas mãos. Os comentários dos usuários com tendinite e artrite mostram que o uso do tapete dessa forma pode ser uma ferramenta para acessibilidade em alguns casos.

APÊNDICE B - FERRAMENTAS COMUNS

O filtro de Bayes é o principal algoritmo para calcular a crença ao longo do tempo para cada estado de um sistema, com base na regra de Bayes, que permite o cálculo da probabilidade de um evento dadas as probabilidades dos eventos que o afetam. Dentre esses estados, um ou mais que tenham valores mais altos de crença podem ser tomados como hipóteses sobre o estado do sistema em cada instante. Para todos os estados possíveis do sistema, esse algoritmo realiza uma predição, ou seja, calcula um valor de crença baseado no estado anterior, nas entradas de controle que foram impostas ao sistema e em um modelo de sua dinâmica. Depois atualiza ou corrige essa crença com base nas entradas que observou por meio de sensores (como, por exemplo, uma imagem vinda de uma câmera) e em um modelo de observação. A predição e a correção são realizadas a cada passo de tempo. Um filtro de Kalman (Welsh & Bishop, 1995) é uma implementação de filtros bayesianos em que o espaço é contínuo, uma única hipótese é utilizada e é representada como uma distribuição gaussiana e os modelos dinâmico e de observação são lineares (existem variações que aceitem modelos não lineares ou múltiplas hipóteses usando misturas de gaussianas). Um filtro de partículas, também conhecido como algoritmo de Condensation (Isard & Blake, 1998a), implementa o filtro de Bayes para múltiplas hipóteses com quaisquer distribuições de probabilidades, que são representadas por um conjunto de amostras.

Um HMM como, por exemplo, os utilizados e discutidos por Bauer e Hienz (2000), é uma cadeia de Markov (ou seja, vale a regra de Markov que afirma que o estado atual tem toda a informação necessária para definir o estado futuro), um conjunto de estados, cada um representando um instante de tempo e com probabilidades de transição para algum outro estado (ou de volta para o mesmo) e de gerar determinada observação. Isso permite que tolere variações na amplitude e no tempo dos padrões que procura reconhecer. Essas probabilidades são ajustadas durante o treinamento do sistema, quando são fornecidas entradas e as saídas corretas. Em geral cada HMM representa um gesto.

Uma máquina de estados finitos, de acordo com Hong, Turk e Huang (2000) é similar a um HMM, mas não tem estados pré-definidos nem há a necessidade de alinhamento dos dados durante o treinamento.

Auto-espacos (eigenspaces) utilizam, nesse contexto, um pequeno número de imagens base (representadas vetores com suas features) e combinações lineares delas para definir um espaço de imagens e permitir sua comparação nesse espaço. Black & Jepson (1998), por exemplo, fazem uso dessa abordagem.

DTW, nesse caso, consiste na aplicação de técnicas de programação dinâmica para alinhar gestos com durações diferentes, de acordo com Marcel (2002). Enquanto EGM tolera variações na forma da mão e da postura (Triesch & Malsburg, 2001). Mitra & Acharya (2007) discutem ainda diversos exemplos do uso de "soft computing" no reconhecimento de gestos. Dentre as abordagens dessa classe, a mais comum é o uso de redes neurais, mas também estão incluídos algoritmos genéticos e lógica fuzzy.