Julio C. S. Jacques Junior

Utilizando Visão Computacional para Simular Comportamentos de Multidões de Humanos Virtuais

São Leopoldo

2006

Julio C. S. Jacques Junior

Utilizando Visão Computacional para Simular Comportamentos de Multidões de Humanos Virtuais

Dissertação submetida a avaliação como requisito parcial para a obtenção do grau de Mestre em Computação Aplicada.

Orientador:
Soraia Raupp Musse

São Leopoldo

2006

FOLHA DE APROVACAO

AGRADECIMENTOS

Agradeço a meus pais, Julio (*in memorian*) e Lucia, às minhas irmãs, Letícia e Luciana, e a todos os familiares e amigos que nunca me deixaram sem o apoio necessário para que eu pudesse chegar ao fim desse trabalho.

A minha orientadora, professora Soraia Raupp Musse e ao professor Cláudio Rosito Jung (co-orientação), pela confiança, incentivo e amizade conquistados no decorrer do curso. Pela atenção com que sempre me receberam e pela dedicação, orientação segura e esforços, que tornaram possível a execução desse trabalho.

A todo o pessoal do laboratório CROMOS, que me ajudaram direta ou indiretamente, em especial à Adriana Braun, André Tavares e Marcelo Borba.

Aos professores e funcionários da UNISINOS, que de alguma maneira, também contribuiram para o meu desenvolvimento.

A HP Brasil, pelo apoio financeiro.

RESUMO

Este trabalho apresenta um modelo para extrair informações do mundo real, capturadas com a utilização de técnicas de visão computacional, no que tange acompanhamento de indivíduos, com o fim de simular e validar comportamentos de multidões de humanos virtuais.

Uma grande dificuldade ao se tentar reproduzir de forma realista (por meio de simulação) o comportamento de uma multidão em um determinado espaço é informar para o modelo de simulação todos os atributos necessários para descrever o movimento das pessoas virtuais. Além das características individuais e coletivas das pessoas poderem produzir uma grande variedade de comportamentos, tornando sua modelagem complexa, o espaço também contém restrições que podem interferir no comportamento das pessoas.

Neste trabalho é proposto um modelo onde pessoas do mundo real têm suas trajetórias capturadas de forma automática. Numa etapa de pós-processamento, as trajetórias capturadas são utilizadas para gerar campos de vetores velocidade que serão utilizados para auxiliar no cálculo do movimento dos humanos virtuais, provendo assim simulações mais realistas. Também está sendo proposta uma métrica de comparação qualitativa entre dois grupos de trajetórias (Mapas de Ocupação Espacial), objetivando validar resultados de simulação com os dados capturados com o modelo proposto. Conforme os resultados obtidos até o momento, pode-se afirmar que a utilização desse modelo apresenta resultados aceitáveis. Dessa forma, objetiva-se contribuir para o aumento de realismo comportamental em simulação de multidões de humanos virtuais.

Palavras-chave: Simulação de multidão, acompanhamento de objetos, detecção de sombra, subtração de fundo.

ABSTRACT

This study presents a model to extract information from the real world using computer vision techniques. In particular, we use tracking algorithms to extract the trajectories of filmed people, aiming to simulate and validate the behavior of virtual human crowds.

A great challenge when trying to reproduce in a realistic manner (by means of simulation) the behavior of a crowd in a determined space is to inform to the simulation model all necessary attributes to describe the movement of virtual people. Individual and general features of people can produce a great variety of behaviors, making its modeling complex. Furthermore, the space also contains restrictions that can interfere on people behavior.

In this study it is proposed a model in which people from the real world have their trajectories captured in an automatic manner. In a post-processing step, captured trajectories are used to generate velocity fields that will be used to help in the calculation of virtual human movement, providing more realistic simulations. It is also being proposed a tool for qualitative comparison between two groups of trajectories (Spatial Occupation Maps), with the purpose of validating simulation results with data captured from real life. According to results obtained until this moment, it can be assumed that the model presents acceptable results. It is expected that the proposed model can contribute to the improvement of behavioral realism in simulations of virtual human crowds.

Keywords: crowd simulation, object tracking, shadow detection, background subtraction.

LISTA DE FIGURAS

1 Exemplo de um espaço de análise. 17

2 Exemplo de um ambiente simulado. 22

3 Exemplo de um ambiente simulado. 22

4 Multidão seguindo as trajetórias descritas pelo *designer*. 23

5 Configuração de trajetórias observadas. À esquerda, a saída de uma biblioteca e a direita, o corredor de uma universidade. 24

6 (a) Ilustração da interface com o usuário e (b) estudo de caso exibido pelos autores. 25

7 (a) Modelo de *background*, (b) imagem em análise e (c) *foreground* detectado (em preto) 27

8 Problema ocasionado pela sombra. 27

9 *Foreground* detectado e classes de objetos, respectivamente 30

10 Arquitetura do sistema proposto. 36

11 (a) exemplo de visão de câmera oblíqua; (b) exemplo de visão de câmera *top-down*. 39

12 (a) Modelo de *background*, (b) imagem em análise e (c) *foreground* detectado (em preto). 42

13 (a), (b) e (c) 3 dentre os 100 quadros usados para gerar os modelos de *background*; (d) imagem em análise; (e) subtração de fundo utilizando o desvio padrão individual de cada *pixel* e (f) subtração de fundo utilizando a mediana dos desvios padrões. 43

14 Detecção inicial de sombra usando diferentes valores de limiar L_{ncc}. (a) $L_{ncc} = 0.90$, (b) $L_{ncc} = 0.95$ e (c) $L_{ncc} = 0.98$ 46

15 (a) Resultado final da detecção da sombra (*pixels* da sombra são representados por cinza claro). (b) Objetos do *foreground* após a remoção da sombra. (c) Eliminação de "buracos" e *pixels* isolados, com a utilização de operadores morfológicos. 47

16 Exemplo de detecção de sombra mal sucedida. (a) Imagem em escala de cinza. (b) *Pixels* do *foreground*. (c) Sombra removida. (d) Pós-processamento morfológico 48

17 Diferença entre usar ou não o NCC na etapa de remoção da sombra. (a) Imagem em análise. (b) *foreground* detectado. (c) em vermelho, *pixels* que passaram no teste do NCC e não passaram no teste da razão; em azul, *pixels* que passaram no teste do NCC e no teste da razão. (d) em azul, *pixels* que passaram no teste da razão (sem utilizar o NCC). (e) resultado final do caso (c). (f) resultado final do caso (d). 49

18 Modelo adaptativo. Primeira coluna: imagens de entrada. Segunda coluna: sem atualizar o modelo de *background* (pontos em preto representam os objetos do *foreground* e em azul da sombra). Terceira coluna: atualizando o modelo. 53

19 Análise da área e distância das bordas da imagem do objeto. 55

20 Exemplo de estimativas para a cabeça. O quadrado vermelho representa o *template* capturado e o ponto azul o seu centro, posição estimada. (a) Centro. (b) Ponto mínimo em y e médio em x. (c) Histograma em y e ponto médio em x. (d) Transformada da distância. 56

21 Exemplo de estimativas para a cabeça. O quadrado vermelho representa o *template* capturado e o ponto azul o seu centro, posição estimada. (a) Centro. (b) Ponto mínimo em y e médio em x. (c) Histograma em y e ponto médio em x. (d) Transformada da distância. 56

22 Exemplo de estimativas para a cabeça. O quadrado vermelho representa o *template* capturado e o ponto azul o seu centro, posição estimada. (a) Centro. (b) Ponto mínimo em y e médio em x. (c) Histograma em y e ponto médio em x. (d) Transformada da distância. 57

23 Exemplo de correlação. (a) *bounding-box* do objeto. (b) *template* de correlação do tempo i. (c) área de busca do tempo $i + 1$. (d) área de busca binária (informa onde a SSD deve ser computada - em branco). 58

24 Resultado do *tracking*. O quadrado vermelho representa o *template* da pessoa. (a) *template* inicial. (b) posição da cabeça após 10 quadros. (c) posição da cabeça após 20 quadros. 60

25 Trajetórias capturadas. 61

26 Campo de vetores gerado a partir de 17 trajetórias capturadas. 62

27 Exemplo de classificação manual das trajetórias. 62

28 (a) Trajetórias capturadas. (b) Trajetórias classificadas em duas classes. (c) Campo de vetores gerado para a Classe 1. (d) Campo de vetores gerado para a Classe 2. 64

29 Trajetórias agrupadas em 4 classes diferentes. 65

30 Campos de vetores gerados para as classes de pessoas que se locomovem nos sentidos: (a) superior → inferior, (b) esquerda → direita, (c) inferior → superior e (d) direita → esquerda. 66

31 (a) Trajetórias capturadas; SOMs obtidos utilizando sub-retângulos com tamanhos (b) 21×21 e (c) 5×5. 67

32 Primeira linha: quadros de uma seqüência de vídeo. Segunda linha: objetos do *foreground* detectados com a técnica de subtração de fundo descrita. Terceira linha: remoção da sombra. Quarta linha: resultado após aplicação de operadores morfológicos de pós-processamento (concatenação de fechamento e abertura). 69

33 (a) Quadro de uma seqüência de vídeo em escala de cinza; (b) *pixels* do *foreground*; (c) sombra removida e (d) pós-processamento morfológico. . . . 70

34 Primeira linha: quadros de uma seqüência de vídeo. Segunda linha: objetos do *foreground* detectados com a técnica de subtração de fundo descrita. Terceira linha: remoção da sombra. Quarta linha: resultado após aplicação de operadores morfológicos de pós-processamento 71

35 Trajetórias detectadas. 73

36 (a) SOM das pessoas reais e (b) SOM dos humanos virtuais. 74

37 Classes geradas para o conjunto de trajetórias. 75

38 Número e local de origem aproximado das pessoas virtuais simuladas para o caso 1. 76

39 As linhas denominadas A, B e C são utilizadas para medir a velocidade média dos pedestres, na Tabela 2. 76

40 (a) SOM das pessoas reais e (b) SOM dos humanos virtuais do caso 1, para o cenário "bifurcação em T". 77

41 Número e local de origem aproximado das pessoas virtuais simuladas para o caso 2. 78

42 (a) SOM das pessoas reais e (b) SOM dos humanos virtuais do caso 2, para o cenário "bifurcação em T". 79

43 Número e local de origem aproximado das pessoas virtuais simuladas para o caso 3. 79

44 (a) SOM das pessoas reais e (b) SOM dos humanos virtuais do caso 3, para o cenário "bifurcação em T". 80

45 (a) campo de visão da câmera e (b) classes de trajetórias geradas. 81

46 (a) classe esquerda/direita e (b) classe direita/esquerda. 81

47 (a) SOM das pessoas reais e (b) SOM dos humanos virtuais, para o cenário "calçadão". 82

48 (a) SOM das pessoas reais e (b) SOM dos humanos virtuais para o cenário "calçadão" extrapolado. 83

49 Distância entre pessoas. Nesse caso, as pessoas 1, 2, 3 e 5 podem estar influenciando no caminhar da pessoa 4. 86

50 Diagrama de Voronoi, campos de visão e região relacionada ao espaço pessoal percebido, PPS. 88

51 Dois grupos detectados. 90
52 Exemplos de PPS e distâncias individuais. 91
53 Humanos virtuais confortáveis e desconfortáveis. 92

LISTA DE TABELAS

1 Tempo médio e desvio padrão para percorrer todo o espaço para a seqüência filmada e para a simulação. 74

2 Métricas quantitativas para validação do caso 1, para o cenário “bifurcação em T”. 77

3 Métricas quantitativas para validação do caso 2, para o cenário “bifurcação em T”. 78

4 Métricas quantitativas para validação do caso 3, para o cenário “bifurcação em T”. 80

5 Velocidade média e desvio padrão para percorrer todo o espaço para a seqüência filmada e para a simulação, para o cenário “calçadão”. 82

6 Velocidade média e desvio padrão para percorrer todo o espaço para a seqüência filmada e para a simulação, para o cenário “calçadão” extrapolado. 83

7 Intervalos de distâncias entre pessoas estabelecidos por Hall 85

8 Intervalos de distâncias propostos nesse trabalho 85

LISTA DE ABREVIATURAS

FPS – *Quadros por segundo (Frames Per Second)*
NCC – *Correlação Cruzada Normalizada (Normalized Cross-Correlation)*
SSD – *Soma das Diferenças Quadráticas (Sum of Squared Difference)*
SOM – *Mapas de Ocupação Espacial (Spatial Occupancy Maps)*
PPS – *Espaço Pessoal Percebido (Perceived Personal Space)*

SUMÁRIO

1 Introdução **14**

1.1 O Problema 16

1.2 Objetivos Gerais 18

1.3 Objetivos Específicos 18

2 Revisão Bibliográfica **20**

2.1 Sistemas computacionais para animação e simulação 20

2.2 Sistemas de acompanhamento de objetos utilizando visão computacional . 25

2.2.1 Conceitos 25

2.2.2 Detecção e *Tracking* 26

2.3 Contexto deste trabalho no estado-da-arte 32

3 Modelo Proposto **36**

3.1 Posicionamento da câmera 38

3.2 Detecção de objetos em movimento 38

3.2.1 Modelagem do *background* 39

3.2.2 Detecção da sombra 42

3.2.2.1 Detecção de *pixels* candidatos a estar na sombra 44

3.2.2.2 Sombra: refinamento e segmentação 46

3.2.3 Atualização do modelo de *background* 48

3.3 *Tracking* 52

3.3.1 Análise do objeto do *foreground* 52

3.3.2 Estabelecendo uma correlação entre quadros consecutivos 57

3.4 Análise dos dados 60

3.4.1 Geração dos campos de velocidades 61

3.4.2 Mapas de Ocupação Espacial 65

4 Resultados Experimentais **68**

4.1 Subtração do fundo e segmentação da sombra 68
4.2 Simulando multidões de humanos virtuais de forma realista, auxiliado por visão computacional 70
4.2.1 Integração dos dados com o simulador 72
4.2.2 Caso A: passagem de pedestres 73
4.2.3 Caso B: bifurcação em T 74
4.2.3.1 bifurcação em T: caso 1 75
4.2.3.2 bifurcação em T: caso 2 77
4.2.3.3 bifurcação em T: caso 3 78
4.2.4 Caso C: calçadão 80

5 Outras Aplicações - Detecção de Eventos na Multidão 84

6 Considerações Finais e Trabalhos Futuros 93

Referências

Anexo - Pulbicações 99

1 INTRODUÇÃO

O homem, ao estudar sistemas, objetos ou fenômenos, muitas vezes depara-se com dificuldades em analisá-los na sua forma natural de existência, por dificuldades de acesso, medição ou mesmo altos riscos e custos envolvidos. Por isto são utilizadas formas de representação que permitam manipular e compreender as entidades estudadas, sendo em seus aspectos qualitativos, ou quantitativos. Esta representação é feita por meio de modelos (STRACK, 1984; LAW; KELTON, 1991). Um modelo de simulação pode prover boas estimativas de como um determinado sistema do mundo real se comportaria sem a necessidade de se correr grandes riscos, possibilitando adquirir informações úteis sobre o seu comportamento.

O comportamento de multidões humanas vem sendo investigado por vários grupos de pesquisa, para diversos propósitos. Por exemplo, a modelagem de multidões humanas pode ser usada em áreas de entretenimento, para simular convincentemente o movimento de um número grande de pessoas virtuais (podendo ser usada em produção de filmes e jogos de computador); povoar espaços virtuais imersivos visando aperfeiçoar o senso de presença (espaços virtuais colaborativos); prover simulações de multidões para avaliação de espaços complexos (simular um fluxo de pessoas deixando um estádio de futebol após uma partida), etc. De fato, modelos de simulação de multidões podem ter um papel importante, tanto para determinar o nível de conforto de pessoas em um grande espaço público como para avaliar sua segurança.

Entretanto a descrição das propriedades de multidões tem se revelado um desafio em função de sua complexidade e diversidade. Isso ocorre pois as características individu-

ais e coletivas das pessoas produzem uma grande variedade de comportamentos, tornando sua modelagem complexa e requerendo uma grande quantidade de atributos a serem considerados para que se consiga uma descrição autêntica. (ZELTZER, 1991 apud BRAUN, 2004).

Atualmente, um grande número de aplicações de visão computacional visa extrair características importantes de uma imagem de entrada, de forma que sua descrição, interpretação ou entendimento, possa ser processado por uma máquina. Por exemplo, um sistema de visão pode distinguir partes em uma linha de montagem e listar algumas de suas características, como tamanho e número de objetos. Sistemas mais sofisticados de visão estão aptos para interpretar resultados do processamento e descrever os vários objetos e seus relacionamentos na cena.

Em sistemas de visão computacional, a imagem de entrada é inicialmente processada, o que pode envolver restauração, melhoramento, ou uma representação apropriada dos dados. Assim, certas características podem ser extraídas, segmentando-se a imagem em componentes. A imagem segmentada é então utilizada por um classificador, ou um sistema de entendimento por imagens (*image understanting system*). Na etapa de classificação da imagem, regiões diferentes são mapeadas, ou segmentadas em um ou diversos objetos, cada qual identificado por um rótulo. Sistemas de entendimento por imagem determinam o relacionamento entre os diferentes objetos em uma cena objetivando prover sua descrição. Por exemplo, tais sistemas deveriam ser capazes de enviar um relatório: "há uma estrada de terra rodeada por vegetação no campo de visão" (GONZALEZ; WINTZ, 1987; JAIN, 1989). Técnicas de visão computacional também podem ser utilizadas como um sistema de medição (PONTE et al., 2004), por exemplo, inserindo-se partículas artificiais em um fluxo de água e detectando seus movimentos para medir a velocidade, direção e sentido que o fluxo está seguindo.

A utilização de técnicas de visão computacional para análise de imagens na dinâmica humana é uma área importante de pesquisa que visa detectar pessoas e entender

seu comportamento em um ambiente complexo. Pode ser usada em diversas aplicações, como vigilância visual, monitoramento de tráfego, interface homem-máquina, aplicações biomecânicas, como por exemplo, a análise do modo de andar das pessoas, provendo diagnóstico médico, análise e treinamento de performance atlética, etc (WANG; SINGH, 2003; WANG; HU; TAN, 2003).

1.1 O Problema

Em muitos trabalhos apresentados na literatura, as simulações do comportamento da multidão é de alguma maneira pré-programado, usualmente relacionado com as restrições do espaço virtual, reproduzindo um mundo virtual artificialmente povoado por pessoas que parecem comportar-se de maneira randômica (ULICNY; CIECHOMSKI; THALMANN, 2004; HELBING; FARKAS; VICSEK, 2000; BRAUN, 2004; CHENNEY, 2004). Em outros trabalhos, o usuário pode observar situações da vida real e adquirir empiricamente dados para calibrar o simulador, reproduzindo as direções, pontos de atração, etc, que motivam a movimentação das pessoas (BROGAN; JOHNSON, 2003). Mesmo que essa tarefa manual possa reproduzir resultados realísticos, ela irá representar apenas a situação observada, e possivelmente muitas alterações nos dados de entrada devem ser feitas para simular diferentes situações, por exemplo, na inserção de novas limitações no espaço, como obstáculos, novos pontos de interesse, ou na mudança das velocidades devido a alterações da densidade ocupacional.

Uma grande dificuldade ao se tentar reproduzir de forma realista (por meio de simulação) o comportamento de uma multidão em um determinado espaço é informar para o modelo de simulação todos os atributos necessários para descrever o movimento das pessoas virtuais. A Figura 1 é utilizada para exemplificar algumas dificuldades que podem ser encontradas ao se tentar observar (de forma empírica) o comportamento das pessoas em um determinado espaço. Por exemplo, pode-se fazer os seguintes questionamentos: qual a velocidade média das pessoas que passam na localização **a** (nos sentidos esquerda-direita

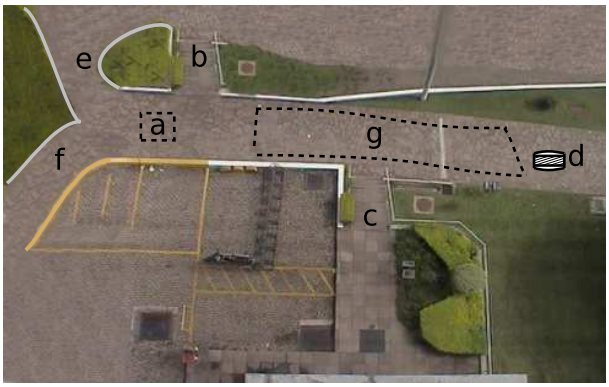


Figura 1: Exemplo de um espaço de análise.

e direita-esquerda)? Como as pessoas desviam do obstáculo **d**? As pessoas desaceleram quando chegam próximas do obstáculo? Com que velocidade média as pessoas sobem e descem as escadas **b** e **c**? Como as pessoas contornam as curvas **e** e **f**? Mesmo que essas observações possam ser realizadas de forma empírica, apesar de exigir um grande trabalho para o observador, pode-se pensar que ao invés de utilizar uma velocidade média para uma região grande, como a região **g** na Figura 1, poderia-se subdividir o espaço e considerar velocidades médias para pequenas regiões, como a região **a**, ou ainda até mesmo a velocidade média em cada ponto da imagem, tornando impraticável uma observação manual.

Não foi encontrado na literatura pesquisada nenhum trabalho que utilize dados de entrada capturados por técnicas de visão computacional para auxiliar na simulação de um fluxo grande de pessoas, em ambientes abertos ou restritos. Informações oriundas do mundo real podem prover resultados mais realistas em simulações de multidão, devido a vantagem de se poder reproduzir velocidades, direções e sentidos semelhantes às observadas. Aqui concentra-se o principal objetivo dessa pesquisa.

Salienta-se que nesse trabalho visa-se capturar dados de ambientes estruturados (como pedestres que caminham na calçada, por exemplo, onde há um fluxo de pessoas). Ambientes não estruturados (como uma partida de futebol, ou uma festa) exigiriam outras considerações que não são o foco deste trabalho. Na verdade, nestes casos citados, os

movimentos são dificilmente generalizados e não representam o problema que visa-se tratar neste trabalho.

1.2 Objetivos Gerais

A grande motivação para o desenvolvimento desse trabalho é a utilização de técnicas de visão computacional para investigação de padrões do comportamento humano em ambientes públicos que possam ser usados para calibrar dados de simuladores e validar seus resultados em comparação com a realidade.

O objetivo é identificar as trajetórias das pessoas de forma automática e quantificar suas velocidades, direções e sentidos. Além disso, visa-se investigar trajetórias com comportamentos semelhantes e classificá-las, de maneira a agrupar trajetórias similares, generalizando comportamentos.

Também visa-se gerar campos de velocidades (por interpolação ou extrapolação), a partir das trajetórias já classificadas, que poderão ser usados para guiar uma multidão de humanos virtuais. E finalmente, propor métodos para comparar os comportamentos de dois grupos de humanos, tanto quantativamente quanto qualitativamente.

1.3 Objetivos Específicos

Com a finalidade de solucionar o problema acima, definiu-se os seguintes objetivos específicos:

- Desenvolver um modelo para detecção automática de pessoas, a partir de registros de vídeo, utilizando imagens em escala de cinza.
- Identificar trajetórias com comportamentos semelhantes e dividir em grupos (pós-processamento).
- Gerar campos de velocidades, a partir dos grupos de trajetórias.

- Desenvolver um protótipo do modelo elaborado, que seja capaz de detectar e armazenar as trajetórias das pessoas.

- Utilizar uma ferramenta de simulação de multidão existente, ou desenvolver um protótipo simples para aplicar a técnica proposta.

- Prover medidas para comparação e validação do espaço analisado (em simulação e vídeo), tais como velocidade média das pessoas, ocupação espacial, etc.

No próximo capítulo serão apresentados alguns trabalhos considerados estado-da-arte em sistemas computacionais para animação e simulação de indivíduos, grupos ou multidões de entidades. Também será apresentada uma visão geral da literatura sobre técnicas de visão computacional que podem ser aplicadas para capturar dados do mundo real a partir de uma seqüência de imagens. No capítulo 3 é descrito o modelo de visão computacional proposto para capturar dados do mundo. Também é descrita a metodologia utilizada para gerar informações para o simulador e métodos para análise qualitativa e quantitativa dos resultados. No capítulo 4 são apresentados alguns resultados experimentais obtidos com a utilização do modelo proposto. Uma possível aplicação para analisar o comportamento das pessoas, considerando aspectos individuais e de grupos é apresentado no capítulo 5. Por fim, no capítulo 6 são expostas as conclusões e sugestões para aperfeiçoamentos futuros. Em anexo, segue a lista de artigos desenvolvidos no decorrer da dissertação.

2 REVISÃO BIBLIOGRÁFICA

Este capítulo destina-se a apresentar alguns trabalhos de extrema importância, ou considerados estado-da-arte, em sistemas computacionais para animação e simulação de indivíduos, grupos ou multidões de entidades, focalizando aspectos de controle de movimento. Como o objetivo principal desse trabalho é capturar informações do mundo de forma automática, para validar e auxiliar em simulações de multidões humanas, também será apresentada uma visão geral da literatura sobre técnicas de visão computacional que podem ser aplicadas para captura de informações do mundo real a partir de uma seqüência de imagens. Estas poderiam ser utilizadas, por exemplo, como dados de entrada para um simulador de indivíduos, grupos ou multidões de humanos, de maneira a calibrá-lo ou validá-lo.

2.1 Sistemas computacionais para animação e simulação

Nesta seção são apresentados alguns trabalhos relacionados a simulação de indivíduos, grupos, ou multidões de entidades, focalizando na forma pela qual essas entidades são controladas, ou seja, como é a gerência e controle de seus movimentos. Propõe-se a classificação dos trabalhos em dois grupos: modelos comportamentais e modelos baseados em usuário.

No grupo de modelos comportamentais são incluídos os trabalhos nos quais as entidades controladas (indivíduos, grupos, ou multidões) são regidas por regras ou forças controladoras, necessitando pouca interação do usuário e com resultado final não deter-

minístico. No grupo de modelos baseados em usuário são incluídos os trabalhos nos quais há bastante interação com usuário, podendo assumir um comportamento determinístico ou não. Os próximos três trabalhos apresentados estão relacionados à primeira classe de modelos, ou seja, comportamentais.

Reynolds (REYNOLDS, 1987) propôs uma abordagem para simular bandos de pássaros, como entidades denominadas *boids*. Ele obteve animações realísticas visualmente utilizando apenas simples regras locais: o movimento de cada *boid* é uma combinação de seu desejo de evitar colisão com *boids* próximos, de aproximar sua velocidade à de *boids* vizinhos e de se mover ao redor do centro do bando. Estes comportamentos não são necessariamente adequados para animação de humanos virtuais. No entanto, este trabalho é citado por prover um modelo pioneiro e pela proposição do termo animação comportamental.

Helbing et al (HELBING; FARKAS; VICSEK, 2000) propõem um modelo baseado em física para simular grupos de pessoas em situação de pânico. Utiliza um sistema de partículas para modelar o movimento da multidão. Nesse sistema, cada partícula i de massa m_i tem um valor de velocidade desejado v_i^g em uma direção indicada por um vetor unitário $\mathbf{e}_i^g$, tendendo a adaptar sua velocidade instantânea $\mathbf{v}_i$ a essas condições desejadas dentro de um certo intervalo de tempo τ_i. Simultaneamente, as partículas tendem a manter uma distância dependente da velocidade em relação à outras partículas e paredes. Este modelo gera resultados observados em cenários reais como a formação de arcos na saída, (Figura 2), além do aumento no tempo de evacuação com o aumento do módulo da velocidade desejada. Este modelo considera várias simplificações no que tange principalmente os seguintes aspectos: (i) todos os indivíduos são simulados de maneira a não possuírem individualidades, ou seja, o comportamento dos agentes é único; (ii) as simulações são realizadas em ambientes geometricamente simples e (iii) a resposta dos agentes à situação de emergência se reduz a fugir do local pela única saída que esse possui.

Braun et al (BRAUN et al., 2003; BRAUN, 2004; BRAUN; BODMAN; MUSSE, 2005)

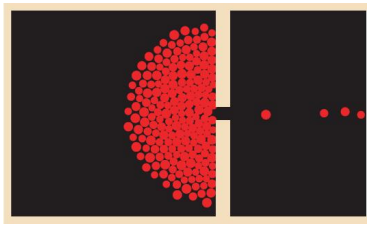


Figura 2: Exemplo de um ambiente simulado.

estenderam o modelo de Helbing (HELBING; FARKAS; VICSEK, 2000) adicionando individualidades aos agentes e incluindo o conceito de grupos, também focado em situação de pânico. Nesse trabalho, são atribuídos níveis de altruísmo (relacionado com a capacidade de se salvar sem ajuda dos outros), e dependência aos agentes (uma pessoa altruísta pode ir em direção ao perigo para resgatar outra pessoa que precise de ajuda em vez de se dirigir imediatamente para a saída). A formação de grupos está relacionada à força de altruísmo que é implementada como uma interação entre dois ou mais agentes de uma mesma família. Nesse modelo o ambiente pode ser subdividido em contextos hierarquicamente relacionados objetivando simular ambientes virtuais com múltiplas salas (Figura 3). No caso de Braun como de Helbing, o controle do movimento dos agentes é feito através de forças motivadoras, baseadas na mecânica newtoniana, que estimula os agentes a irem para as saídas do ambiente, provendo comportamentos que emergem do modelo.

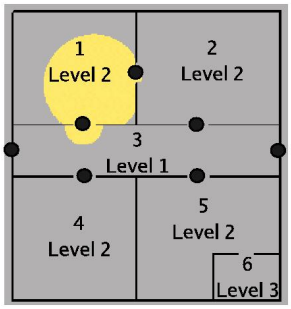


Figura 3: Exemplo de um ambiente simulado.

Os próximos três trabalhos apresentados estão relacionados à segunda classe de modelos, baseados em usuário.

Ulicny et al (ULICNY; CIECHOMSKI; THALMANN, 2004) propõem uma ferra-

menta (*Crowdbrush*), para distribuir, modificar e controlar membros de uma multidão em tempo real. O *designer* trabalha em um ambiente 2D, utilizando o *mouse* e o teclado para manipular os objetos correspondentes em 3D. Pode-se incluir ou remover membros da multidão, alterar suas aparências, animação, configurar um comportamento de alto nível, pré-determinar trajetórias para a multidão, ou enviar eventos para um subsistema de comportamentos. A Figura 4 exibe uma multidão seguindo as trajetórias especificadas pelo *designer*. Este trabalho é um recente exemplo de controle de multidões baseada em intervenção com o usuário (*user-based*).

Figura 4: Multidão seguindo as trajetórias descritas pelo *designer*.

Brogan e Johnson (BROGAN; JOHNSON, 2003) apresentam uma abordagem baseada em observações empíricas de situações da vida real para descrever planos de trajetórias para indivíduos. Neste trabalho, um grupo de pessoas foi conduzido em um experimento controlado, cujo objetivo era fazer com que os participantes caminhassem de um ponto de partida até um ponto desejado, sob circunstâncias distintas, para gerar uma base de dados para análise. Usando o vídeo gravado nos experimentos (com 640 x 480 de resolução, numa taxa de aquisição de 10 quadros por segundo), os autores digitalizaram as trajetórias percorridas, projetando o centro de massa de cada participante no plano de solo. Posteriormente os pontos projetados são interpolados (*Catmull-Rom Spline*) para gerar uma trajetória contínua. Duas configurações distintas de trajetórias observadas são exemplificadas com auxílio da Figura 5.

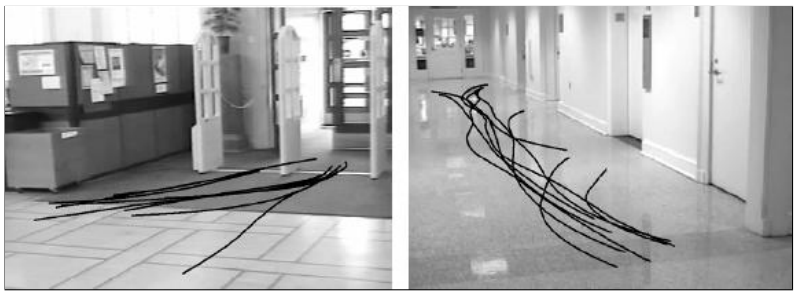


Figura 5: Configuração de trajetórias observadas. À esquerda, a saída de uma biblioteca e a direita, o corredor de uma universidade.

Explorando os dados dos experimentos, Borgan e Johnson chegaram a três conclusões: (1) as pessoas aceleram e desaceleram com freqüências similares, alcançando uma velocidade máxima similar; (2) os pedestres exibem limitações radiais quando evitam obstáculos; (3) planos de trajetórias são influenciados pela inércia dos pedestres.

Stephen Chenney (CHENNEY, 2004) apresenta uma abordagem para representação e desenvolvimento de campos de velocidades, objetivando descrever o movimento de um grande número de entidades. Nesse trabalho, um conjunto de pequenos campos de velocidades (*flow tiles*), gerados a partir de funções, é usado para formar um grande campo de velocidades, que pode ser usado para guiar fluidos ou multidões (utilizando campos estáticos), ou gerar suaves redemoinhos (*swirling fog*) em uma piscina (utilizando campos que variam no tempo). A Figura 6 ilustra a interface utilizada para construção do campo de velocidades e exibe as pessoas seguindo as trajetórias descritas. Uma grande vantagem de se utilizar essa abordagem é a facilidade de armazenamento e reutilização dos campos (*flow tiles*).

O presente trabalho difere das classificações apresentadas por não tratar-se de um modelo de simulação comportamental, nem exigir intervenção com o usuário, visto que utiliza-se dados capturados da vida real.

(a) (b)

Figura 6: (a) Ilustração da interface com o usuário e (b) estudo de caso exibido pelos autores.

2.2 Sistemas de acompanhamento de objetos utilizando visão computacional

O objetivo dessa seção é introduzir alguns conceitos e apresentar alguns dos principais trabalhos encontrados na literatura relacionados a detecção automática de pessoas com a utilização de técnicas de visão computacional.

2.2.1 Conceitos

Segundo Wang et al. (WANG; HU; TAN, 2003), a análise visual do movimento humano busca detectar (*human detection*), acompanhar (*human tracking*) e entender o comportamento das pessoas (*human behavior understanding*). De uma maneira mais geral, interpretar seu comportamento a partir de seqüências de imagens. Tais análises tem atraído grande interesse em pesquisas envolvendo visão computacional, devido a suas aplicações promissoras em muitas áreas, como por exemplo, vigilância visual, interface com usuário, análise de performance atlética, realidade virtual, etc.

A etapa de detecção objetiva segmentar as regiões correspondentes às pessoas em relação ao resto da imagem. Este é um passo importante para sistemas de análise do movimento humano, já que os processos subseqüentes como *tracking* e entendimento do comportamento são bastante dependentes disso. O processo de detecção normalmente envolve

segmentação dos objetos em movimento e classificação. Por exemplo, imagens capturadas por uma câmera de vigilância posicionada em uma auto-estrada podem incluir veículos, pessoas, pássaros, nuvens, etc, como objetos em movimento. Para futuramente acompanhar (*tracking*) as pessoas, é necessário distinguí-las corretamente dos outros objetos. Dessa forma, essa etapa de classificação pode não ser necessária em algumas situações, por exemplo, quando se sabe que os objetos em movimento tratam-se somente de pessoas.

O acompanhamento (ou rastreamento) de objetos em seqüências de vídeo, é um passo fundamental em visão computacional para se entender a dinâmica do comportamento humano. O objetivo principal é acompanhar o movimento dos objetos em uma seqüência de quadros do vídeo. Posteriormente, o resultado do *tracking* pode ser analisado matematicamente para se interpretar o comportamento do objeto em estudo. O acompanhamento no decorrer do tempo envolve normalmente estabelecer uma relação coerente para um objeto em quadros consecutivos do vídeo, usando características como pontos, linhas, ou regiões.

2.2.2 Detecção e *Tracking*

Subtração de fundo (*background subtraction*) é uma abordagem bastante utilizada em aplicações que utilizam câmeras fixas (como por exemplo, câmeras de vigilância) para detecção de objetos em movimento. Subtração de fundo consiste basicamente em se obter um modelo matemático de fundo (*modelo de background*) da cena, e subtrair cada quadro da seqüência de imagens por esse modelo. Pontos (*pixels*) que atingem um determinado limiar são associados a objetos em movimento (*foreground*) e os demais são associados ao fundo da imagem (*background*). A Figura 7 é utilizada para exemplificar de uma maneira bem simples o processo de detecção utilizando-se um algoritmo de subtração de fundo.

As abordagens para subtração de fundo diferenciam-se umas das outras pelo tipo de modelo usado para gerar o fundo e pelas funções utilizadas para atualizar o modelo de fundo. Um modelo bastante simples de fundo pode ser adquirido com o valor médio de

(a) (b) (c)

Figura 7: (a) Modelo de *background*, (b) imagem em análise e (c) *foreground* detectado (em preto)

cada *pixel* da imagem, para uma seqüência de imagens, aproximando o fundo a uma cena estática.

Entretanto, a maioria dos algoritmos de subtração de fundo produzem falsos resultados quando a câmera se desloca, por exemplo, devido a correntes fortes de vento, resultando em uma imagem tremida. É uma abordagem normalmente bastante sensível a ruído, mudanças bruscas de iluminação e pode detectar indesejavelmente sombras como *foreground*. Em particular, nessa abordagem a detecção de sombras como objetos do *foreground* é bastante comum. Por exemplo, sombras podem causar conexões de diferentes objetos de um grupo, conforme ilustrado na Figura 8, gerando um único objeto (geralmente denominado *blob*) como resultado da subtração do fundo. Em casos como esse, torna-se mais difícil isolar e acompanhar cada objeto no grupo.

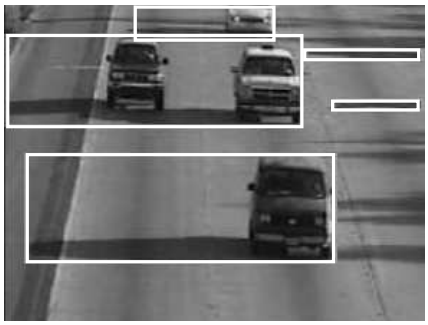


Figura 8: Problema ocasionado pela sombra.

Métodos de fluxo óptico (*optical flow*) também são geralmente utilizado para descrever movimentos coerentes de pontos, ou características entre quadros de uma seqüência

de imagens. Esse método pode ser usado para detectar independentemente objetos em movimento, mesmo na presença de movimento da câmera. Entretanto, a maioria dos métodos de fluxo óptico são computacionalmente complexos e muito sensíveis a ruído, não podendo ser utilizados para detecção em tempo-real sem um *hardware* especializado. Uma discussão mais detalhada sobre fluxo óptico pode ser encontrada no trabalho de Barron (BARRON et al., 1994 apud WANG; HU; TAN, 2003).

Outra abordagem utilizada para detecção de objetos em movimento é a da diferença temporal (*temporal differencing*). Essa abordagem faz uso da diferença dos *pixels* entre dois ou mais quadros consecutivos, em uma seqüência de imagens, para extrair objetos em movimento. É uma técnica capaz de se adaptar facilmente à espaços dinâmicos, mas geralmente realiza um trabalho pouco considerável em se tratando de extrair o total de características relevantes dos *pixels*, por exemplo, possivelmente gerando buracos, ou falhas dentro de objetos em movimento (WANG; HU; TAN, 2003). A seguir, serão analisadas algumas técnicas que podem ser consideradas estado-da-arte na detecção e acompanhamento de pessoas.

McKenna et al (MCKENNA et al., 2000) propõem um método de subtração de *background* que combina informação de cor (RGB e cromaticidade) e informação do gradiente para detecção de *foreground*, tratamento de sombras e valores de *pixels* contendo pouca informação de cor. Detecta regiões, pessoas e grupos de pessoas; cria modelos de aparência de pessoas para que elas possam ser rastreadas também em situações de oclusão. Utiliza informação de cor para prover estimativas de profundidade durante a oclusão e reconhece quando uma pessoa deposita ou retira um objeto da cena.

Haritaoglu et al (HARITAOGLU; DAVIS, 2000) propõem um modelo estatístico de *background*, utilizando imagens em tons de cinza, e representando cada *pixel* por três valores: mínimo e máximo valor de intensidade, e diferença máxima de intensidade entre quadros consecutivos do vídeo, durante um período de tempo, denominado período de aprendizado. O sistema (chamado W4) é aplicado em vigilância visual para detectar e

rastrear múltiplas pessoas, e monitorar suas atividades em ambientes abertos. O sistema opera com uma câmera, podendo capturar imagens em tons de cinza ou infra-vermelho (pois durante a noite se tem muito pouca informação de cor). Não faz nenhum tratamento para sombras. Utiliza uma combinação de análise de forma e *tracking* para localizar pessoas e suas partes (cabeça, mãos, pés e tronco) e para criar modelos de aparência de pessoas para que elas possam ser rastreadas também em situações de oclusão. O sistema pode determinar quando uma região de *foreground* contém múltiplas pessoas e consegue segmentar as pessoas para acompanhá-las individualmente. Também detecta quando uma pessoa está carregando um objeto; cria modelos de aparência para os objetos para identificá-los em quadros futuros e reconhece quando uma pessoa deposita ou retira um objeto da cena. Esse modelo foi utilizado como base neste trabalho e será melhor detalhado no próximo capítulo.

Em (ELGAMMAL et al., 2002) é utilizado um modelo de *background* não paramétrico que pode ser utilizado em imagens em tons de cinza ou coloridas. Possui uma vantagem de detectar objetos mesmo na presença de árvores (o que é considerado um fator agravante em subtração de fundo, devido a não se tratar de um objeto totalmente estático) e mudanças de iluminação. Entretanto, para detecção de sombra, é utilizado informação de cor (utilizando o espaço de cor rgb normalizado).

Cucchiara et al (CUCCHIARA et al., 2003) apresentam um método que combina suposições estatísticas com conhecimento de níveis dos objetos. Tais objetos podem ser o objeto em movimento propriamente dito, *ghosts* (um conjunto de pontos conectados, detectados na subtração do fundo, que não correspondem a nenhum objeto em movimento) e sombras (podendo estar conectada ao objeto ou não). Utiliza fluxo óptico para diferenciar um objeto em movimento de um *ghost*, para todos os pontos do objeto encontrado, assumindo que objetos em movimentos possuem um movimento significante, enquanto *ghosts* possuem uma média próxima de zero. O modelo de fundo é gerado a partir de um filtro temporal da mediana, no espaço de cor RGB. Também utilizam o espaço de cor HSV para fazer a detecção de sombras em movimento.

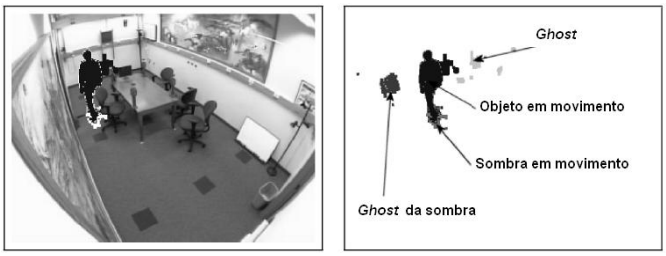


Figura 9: *Foreground* detectado e classes de objetos, respectivamente

Em (KUMAR; SENGUPTA; LEE, 2002) foi desenvolvido um estudo comparativo sobre qual espaço de cor melhor se adapta à detecção de *foreground* e sombras, para sistemas de monitoramento de tráfego. Os autores concluíram que o espaço de cor "YCrCb" gera melhores resultados para a detecção esperada. Nesse estudo foram analisados os sistemas de cores RGB, XYZ, YCrCb, HSV e rgb normalizado. Para realizar este estudo os autores utilizaram técnicas estatísticas (basicamente, média e desvio padrão) para modelar o *background*. Os resultados são comparados em termos de "detecção verdadeira", "não detecção", e "detecção falsa" para cada pixel, também como detecção de objetos em movimentos como regiões.

Em (PRATI et al., 2001) é realizado um estudo comparativo de duas técnicas de detecção de sombras para análise de trafego de veículos. A primeira - *SAKBOT* (*Statistical And Knowledge-Based Object Tracker*) -, utiliza técnicas de subtração de fundo para detectar objetos em movimento. Detecta sombras com o uso da luminância e posteriormente faz um refinamento utilizando propriedades de cor. Pra isso, SAKBOT converte as informações dos *pixels* de valor RGB para HSV. Não explora propriedades espaciais e não provê nenhum pós-processamento após a detecção da sombra (como por exemplo, operações morfológicas). O segundo algoritmo - *ATON* (*Autonomous Transportation agents for On-scene Networked incident management*) faz utilização de informações locais, espaciais (assumindo que objetos e sombras são regiões conexas na cena) e temporais (assumindo que a posição dos objetos e das sombras podem ser previstas a partir de quadros anteriores) para detectar objetos e sombras. Dado os valores de média e variância para cada canal de cor do ponto de referência, pode-se derivar seus valores correspon-

dentes para *pixels* na sombra. Também é associado a cada *pixel* uma probabilidade de ele ser classificado como *background*, *foreground*, ou sombra. Nesse estudo comparativo concluiu-se que ATON distingue melhor entre *pixels* do *background*, *foreground* e sombras enquanto que SAKBOT detecta melhor sombras. Os autores sugerem que uma possível integração das duas técnicas, combinando suas vantagens e eliminando suas desvantagens, pode ser um estudo bastante atrativo e de grande interesse.

Stauffer e Grimson (STAUFFER; GRIMSON, 2000) propõem um sistema de monitoramento visual que passivamente observa os objetos detectados para identificar e classificar padrões de atividades. A detecção dos objetos é baseada em um método adaptativo de subtração de *background* que modela cada *pixel* como uma mistura de Gaussianas. O sistema é capaz de acumular ocorrências de atividades semelhantes para criar uma árvore binária hierárquica de classificação.

Alguns autores propõem diferentes abordagens para detectar sombra em imagens monocromáticas. Stauder et al (STAUDER; MECH; OSTERMANN, 1999) utilizam quatro critérios para classificar se uma região da imagem pertence a uma sombra em movimento: (i) assume-se que o foco de luz é forte, assim mudanças de iluminação são grandes em amplitude quando causadas por sombras em movimento, ocasionando uma diferença considerável entre dois quadros consecutivos; (ii) assume-se que a câmera é estática e que o *background* é texturizado, assim regiões alteradas pela sombra podem ser distinguidas de regiões alteradas por objetos em movimento, verificando-se as bordas (estáticas) do *background*; (iii) assume-se que o *background* é plano, assim mudanças de iluminação causadas por sombras em movimento devem ser suaves; (iv) assume-se que a luz que causa a sombra possui uma determinada extensão, assim, a sombra irá possuir uma penumbra (que é uma transição suave de iluminação de uma região na sombra para uma fora dela). Rosin (ROSIN; ELLIS, 1995) interpreta a sombra causada na imagem como regiões semi-transparentes, que possuem uma atenuação na reflexão local. Identifica tais regiões analisando suas propriedades fotométricas: primeira, *pixels* na sombra devem ser mais escuros que os da imagem de referência; segundo, as regiões na sombra devem possuir

uma intensidade de atenuação homogênea (exceto nas bordas, devido à penumbra).

2.3 Contexto deste trabalho no estado-da-arte

Simulação de multidões vem sendo estudada a bastante tempo, para diferentes propósitos. Em todos os trabalhos encontrados na literatura revisada, sobre animação e simulação computacional, cada indivíduo, ou membro da multidão virtual possui um vetor velocidade associado, para cada instante de tempo, que controla o seu movimento (seja esse vetor retornado por um conjunto de regras, informado por um usuário, ou resultado de uma equação).

No trabalho de Reynolds (REYNOLDS, 1987), as regras que regem o movimento dos agentes não são necessariamente adequadas para simulação de humanos. No trabalho de Ulincy et al (ULICNY; CIECHOMSKI; THALMANN, 2004), é necessário que o usuário especifique quais os tipos de comportamento que os agentes devem ter, onde a maior dificuldade seria atribuir uma grande quantidade de parâmetros, para reproduzir de forma realista, um padrão de comportamento encontrado em um espaço complexo. Nos trabalhos de Helbing et al (HELBING; FARKAS; VICSEK, 2000) e Braun et al (BRAUN et al., 2003; BRAUN, 2004), onde o foco da pesquisa está voltado para situações de emergência, o objetivo dos agentes é evacuar de forma segura um determinado ambiente. Nesses dois trabalhos, para reproduzir o comportamento dos agentes em uma situação de vida normal (em um ambiente complexo), de forma realista, deve-se atribuir uma grande quantidade de parâmetros, o que pode se tornar uma tarefa bastante árdua e não necessariamente ser eficiente na solução do problema, além de dificultar pequenas alterações.

No trabalho de Borgan e Johnson (BROGAN; JOHNSON, 2003), mesmo que sejam capturados de maneira empírica dados oriundos do mundo real para descrever planos de trajetórias para indivíduos, tal tarefa é feita manualmente e a consideramos de difícil execução, e novamente de difícil alteração. o trabalho de Chenney (CHENNEY, 2004) utiliza campos de velocidades definidos interativamente para guiar seus movimentos, mas

nosso trabalho se diferencia na maneira como os campos são gerados (usando visão computacional, capturando informações do mundo real).

Objetivando capturar informações do mundo de forma automática, para validar e auxiliar em simulações de multidões humanas, foram revisados diversos trabalhos encontrados na literatura, sobre técnicas de visão computacional empregadas na identificação de pessoas ou objetos em movimento. Diversos trabalhos (WANG; HU; TAN, 2003; MCKENNA et al., 2000; HARITAOGLU; DAVIS, 2000; ELGAMMAL et al., 2002; STAUFFER; GRIMSON, 2000) poderiam ser utilizados para capturar a trajetória de pessoas do mundo real de forma automática, a partir de seqüências de vídeo. Eles diferenciam-se uns dos outros por diversos fatores, podendo-se considerar, por exemplo, a complexidade do modelo (podendo estar associado ao custo computacional e de implementação), tipos de dados empregados (imagens coloridas ou monocromáticas), resultados obtidos (detecção verdadeira ou robustez, por exemplo), dentre outros aspectos.

A grande maioria dos trabalhos utilizam imagens coloridas, e um campo de visão de câmera oblíquo ou lateral (que dificulta o processo de mapeamento de coordenadas de *pixel* para coordenadas do mundo, e na maioria das vezes gera problemas de oclusão entre pessoas, não sendo considerado o campo de visão ideal para utilização neste trabalho). Acredita-se que a utilização de imagens coloridas beneficia o processo de análise e entendimento automático das imagens, uma vez que os espaços de cores utilizam três canais de informação, podendo ser citados os espaços de cores RGB, HSV, YCrCb, etc. Entretanto, a utilização de somente imagens em escala de cinza pode ser vantajosa, pois diminui o número de dados de entrada, podendo diminuir o número de cálculos envolvidos (de três canais de cores para somente um), e em situações noturnas, também há muito pouca informação de cor [1].

Dessa forma, foi utilizado como base, um modelo bastante referenciado na literatura, denominado W4 (HARITAOGLU; DAVIS, 2000), pela sua simplicidade e adequação a imagens em escala de cinza. Tal modelo sofreu duas modificações, nas etapas de geração

[1] muitos estabelecimentos comerciais também utilizam apenas câmeras monocromáticas.

do modelo e teste de objetos em movimento, o que consideramos um aperfeiçoamento. Também foi adicionado uma etapa de identificação e remoção de sombra (já que o modelo W4 identifica a sombra como parte do objeto em movimento e não faz nenhum tratamento especial). Além disso, a grande maioria dos trabalhos que identificam a sombra em seqüências de imagens, também o faz com a utilização de imagens coloridas. Assim também está se propondo uma nova abordagem para remoção de sombras em imagens em escala de cinza.

Embora existam diversas abordagens para realizar o acompanhamento dos objetos (WANG; HU; TAN, 2003; MCKENNA et al., 2000; HARITAOGLU; DAVIS, 2000; ELGAMMAL et al., 2002; STAUFFER; GRIMSON, 2000), optou-se pela utilização da correlação de máscaras neste trabalho. De fato, em vistas de topo (utilizadas neste trabalho), a cabeça das pessoas é aproximadamente constante, sendo uma boa escolha para a máscara de correlação.

Em uma etapa de pós-processamento do sistema de visão computacional, os dados capturados do mundo real são utilizados para simular e validar comportamentos de multidões de humanos virtuais. A partir de trajetórias capturadas de pessoas, de forma automática, são extraídos comportamentos semelhantes e gerados um ou mais campos de velocidades associado a cada padrão de comportamento, que poderão guiar uma multidão de humanos virtuais com comportamento bastante similar ao das pessoas do ambiente estudado. Também está sendo proposta uma métrica de comparação qualitativa entre dois grupos de trajetórias, Mapas de Ocupação Espacial (SOM – *Spatial Occupancy Maps*), objetivando validar resultados de simulação com os dados capturados com o modelo proposto.

Dessa forma, acreditamos que este trabalho possa ter contribuições nas áreas de visão computacional e simulação de humanos virtuais, pois pode-se capturar informações do mundo através de algoritmos de visão por computador e reproduzir comportamentos bastante condizentes com a realidade, que talvez, outros modelos comportamentais ou

baseados em usuários não fossem tão eficientes. O próximo capítulo descreve o modelo proposto.

3 MODELO PROPOSTO

Neste capítulo é apresentado o modelo proposto para detecção e acompanhamento de pessoas a partir de registros de vídeo. Subdividiu-se o modelo em 2 fases de processamento, sendo o sistema de visão computacional e a análise de dados. A Figura 10 exibe uma visão geral do sistema proposto.

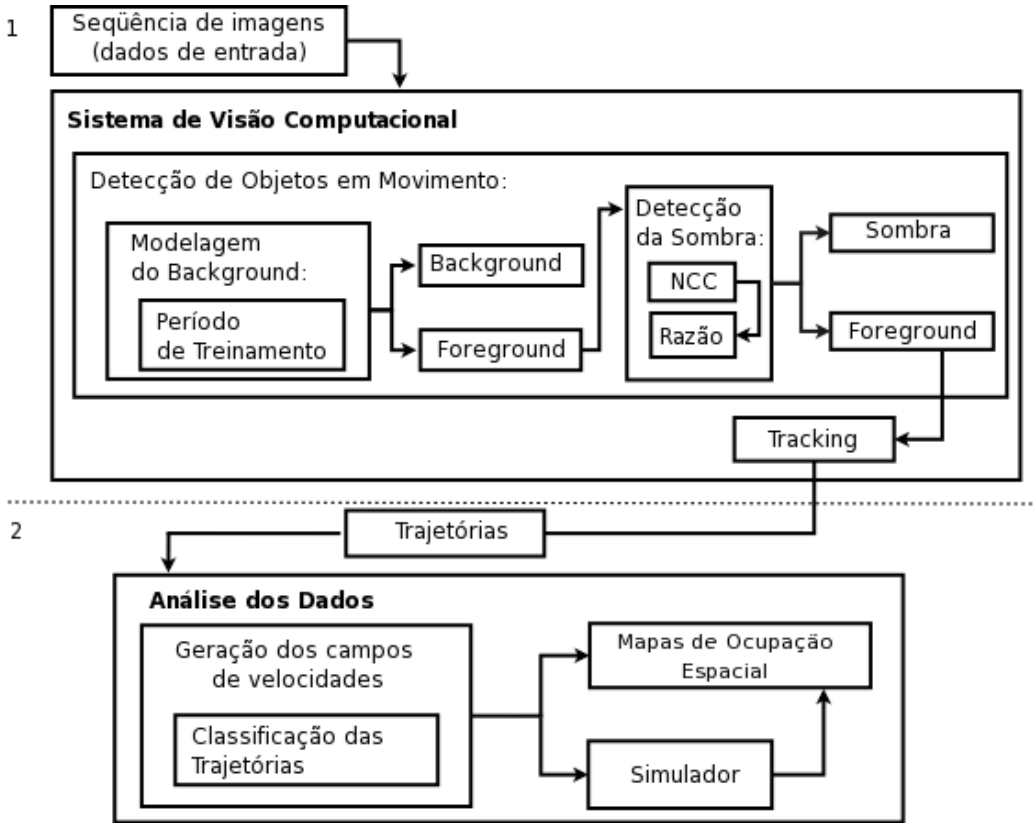


Figura 10: Arquitetura do sistema proposto.

Na Figura 10, o módulo denominado *Sistema de Visão Computacional* é responsável por processar as imagens de entrada (1) e retornar um conjunto de trajetórias

capturadas (2). Para tanto, será utilizada uma abordagem para *Detecção de Objetos em Movimento* que faz a utilização de um algoritmo de subtração de *background* (referenciado na Figura 10 como *Modelagem do Background*). Ao final da etapa de *Modelagem do Background* é retornada uma matriz binária (correspondente à imagem em análise) que representa o conjunto dos *pixels* classificados como objetos em movimento na cena (*foreground*).

Em particular, técnicas de subtração de *background* possuem a característica de detectar, indesejavelmente, sombras como objetos do *foreground*. Dessa forma, para todo *pixel* do *foreground* é verificado se trata-se de um ponto pertencente ao objeto ou à sombra. Essa verificação é realizada em duas etapas: na primeira é utilizada a Correlação Cruzada Normalizada (*normalized cross-correlation – NCC*) para comparar os *pixels* do *foreground* com o modelo de *background*; na segunda etapa, todos os *pixels* que alcançam um determinado valor de limiar, no teste do NCC, são pós-processados verificando-se, numa região de vizinhança $M \times M$, se a razão dos *pixels* é aproximadamente constante. No final dessa etapa é retornado o conjunto de *pixels* que assume-se pertencer ao objeto em movimento. A etapa de *Tracking* é responsável por acompanhar os objetos classificados como *foreground* no decorrer da cena e gerar trajetórias.

O módulo denominado *Análise dos Dados* recebe um conjunto de trajetórias capturadas e é responsável por gerar os campos de vetores velocidades que poderão ser usados por um simulador para guiar uma multidão de humanos virtuais, e por comparar resultados da simulação com os dados capturados pelo sistema de visão computacional (*Mapas de Ocupação Espacial*).

Na seção 3.1 é apresentado o tipo de visão de câmera ideal para ser utilizado no desenvolvimento do trabalho. Na seção 3.2 é apresentada a técnica utilizada para fazer a detecção automática das pessoas (detecção de objetos em movimento). Na seção 3.3 é apresentado o método que faz o relacionamento de uma pessoa entre dois quadros consecutivos da seqüência do vídeo (*tracking*). A metodologia utilizada para gerar os

campos de velocidades é descrita na seção 3.4.1. Por fim, na seção 3.4.2 são introduzidos os Mapas de Ocupação Espacial, que se propõem servir como uma métrica de comparação coerente entre dois grupos de trajetórias.

3.1 Posicionamento da câmera

Muitos trabalhos baseados em técnicas de visão computacional para detecção de pessoas são aplicados em problemas de vigilância. A maioria desses trabalhos utilizam uma câmera fixa com um campo de visão oblíquo (ou lateral), pois nessas aplicações é de grande interesse a identificação das faces das pessoas. Nesse tipo de aplicação, onde o campo de visão é oblíquo, existe um fator complicador em se tratando de detecção automática de pessoas, que é o acompanhamento das pessoas que estão completa ou parcialmente obstruídas por outras pessoas ou objetos, conforme ilustrado na Figura 11 (a). Além disso, o mapeamento das coordenadas de *pixel* da imagem para coordenadas do mundo pode não ser preciso, devido à projeção da câmera.

Como o objetivo desse trabalho é extrair trajetórias de indivíduos, escolheu-se uma configuração de câmera *top-down*, que provê um campo de visão normal em relação ao solo, ilustrado na Figura 11 (b), diminuindo ou eliminando o problema de oclusão e fornecendo um mapeamento simples entre coordenadas de mundo e de imagem. Por outro lado, a utilização dessa configuração de câmera nem sempre é adequada para ambientes internos, pois o campo de visão é reduzido se a altura de fixação da câmera é baixa.

3.2 Detecção de objetos em movimento

Nesta seção é descrita a metodologia utilizada para detecção de objetos em movimento a partir de registros de vídeo, com a utilização de técnicas de visão computacional. Devido à abordagem de subtração de fundo ser uma técnica bastante utilizada, computacionalmente barata e de fácil implementação, optou-se por utilizá-la. O algoritmo utilizado para geração do modelo de fundo foi o W4 (HARITAOGLU; DAVIS, 1998, 2000), devido

(a) (b)

Figura 11: (a) exemplo de visão de câmera oblíqua; (b) exemplo de visão de câmera *top-down*.

a sua robustez, simplicidade e adequação para imagens monocromáticas. Entretanto, foi feita uma pequena modificação nessa abordagem no que diz respeito à modelagem do *background* e no teste realizado para identificar se um *pixel* pertence ao fundo, como será descrito a seguir.

Como é bastante comum algoritmos de subtração de fundo detectarem sombras como objetos do *foreground*, também é proposta uma nova abordagem para detecção de sombras, utilizando imagens em tons de cinza.

3.2.1 Modelagem do *background*

W4 utiliza um modelo de *background* construído a partir de estatísticas de valores do *background* durante um período de treinamento, obtendo um modelo de fundo robusto mesmo se houverem objetos se movimentando na cena, como pedestres, automóveis, etc. O modelo é dividido em dois estágios, como descrito a seguir.

Num primeiro momento, é utilizado um filtro da mediana em cada *pixel* durante um determinado período de tempo (normalmente de 20 a 40 segundos) para distinguir *pixels* em movimento de *pixels* estáticos (entretanto, nossos experimentos mostraram que 100 quadros $\approx$ 3.3 segundos são suficientes para o período de treinamento, se não houver muitos objetos em movimento na cena). Num segundo estágio, apenas aqueles *pixels* considerados estáticos são utilizados para construção inicial do modelo de *background*.

Considere V uma pilha contendo N imagens consecutivas, $V^k(i,j)$ sendo a intensidade do *pixel* (i,j) na *k-ésima* imagem de V, $\sigma(i,j)$ e $\lambda(i,j)$ o desvio padrão e mediana da intensidade do *pixel*(i,j) para todas imagens em V, respectivamente. O modelo inicial de *background* para um *pixel* (i,j) é formado por um vetor tridimensional: o valor mínimo $m(i,j)$ e máximo $n(i,j)$ de intensidade e máxima diferença de intensidade $d(i,j)$ entre quadros consecutivos observados durante esse período de treinamento. O modelo de *background* $\mathbf{B}(i,j) = [m(i,j), n(i,j), d(i,j)]$, é obtido da seguinte maneira:

$$\begin{bmatrix} m(i,j) \\ n(i,j) \\ d(i,j) \end{bmatrix} = \begin{bmatrix} \min_z V^z(i,j) \\ \max_z V^z(i,j) \\ \max_z |V^z(i,j) - V^{z-1}(i,j)| \end{bmatrix}, \tag{3.1}$$

onde z são os quadros que satisfazem a condição:

$$|V^z(i,j) - \lambda(i,j)| \leq 2\sigma(i,j). \tag{3.2}$$

Essa condição, representada pela equação (3.2) garante que apenas os *pixels* considerados estáticos serão utilizados no cálculo do modelo de *background*. Porém, quando há bastante objetos em movimento na cena, o desvio padrão em determinados pontos da imagem pode ser alto, fazendo com que o intervalo de *pixels* considerados *background* para aquele ponto seja conseqüentemente grande. Dessa forma, optou-se por alterar a condição anterior para:

$$|V^z(i,j) \ - \ \lambda(i,j)| \ \leq \ 2\beta, \tag{3.3}$$

onde β é a mediana de todos os desvios padrões (assumindo-se que a variação de iluminação afeta a cena de uma forma global). Essa pequena modificação nos permite gerar o modelo de fundo da cena em um tempo relativamente pequeno, mesmo se houver objetos em movimento na cena. Entretanto, pode ocorrer que alguns *pixels* (que tiveram o desvio padrão bastante elevado) não passem na condição representada pela equação (3.3).

Assim, quando o vetor resultante para aquele ponto for vazio, ou tiver apenas um elemento (apenas a mediana passou na condição), utiliza-se a condição representada pela equação (3.2).

Após o período de treinamento, o modelo inicial de *background* $\mathbf{B}(i,j)$ é obtido. Então, cada imagem $I^t(i,j)$ da seqüência de imagens do vídeo é comparada com $\mathbf{B}(i,j)$, e um *pixel* (i,j) é classificado como *background* (*pixel* estático) se:

$$|I^t(i,j) - m(i,j)| < k\mu \quad \text{ou} \quad |I^t(i,j) - n(i,j)| < k\mu, \tag{3.4}$$

onde μ é a mediana das máximas diferenças absolutas entre quadros $d(i,j)$, e k é uma constante (em W4 é sugerido $k = 2$). Pode ser observado que, se um determinado *pixel*(i,j) tem um valor de intensidade $m(i,j) \leq I^t(i,j) \leq n(i,j)$ em um determinado quadro t, ele deve ser classificado como *background* (porque ele pertence a uma região limitada pelo valor mínimo e máximo do modelo de *background*). Entretanto, a equação (3.4) pode classificar erroneamente um *pixel* como *foreground*, dependendo de k, μ, $m(i,j)$ e $n(i,j)$. Por exemplo, se $\mu = 5$, $k = 2$, $m(i,j) = 40$, $n(i,j) = 65$ e $I^t(i,j) = 52$, a equação 3.4 irá classificar $I^t(i,j)$ como *foreground*, mesmo que ele pertença ao intervalo entre $m(i,j)$ e $n(i,j)$. Para solucionar esse problema, se propõe um teste alternativo para a detecção do *foreground*, o qual classifica $I^t(i,j)$ como um *pixel* pertencente ao *background* se:

$$m(i,j) - k\mu \leq I^t(i,j) \leq n(i,j) + k\mu. \tag{3.5}$$

Um exemplo de detecção de *foreground* é ilustrado na Figura 12. Os objetos do *foreground* foram efetivamente encontrados, mas dois tipos de sombra também foram detectados: no objeto da esquerda a sombra foi causada por obstrução da luz indireta; no objeto da direita a sombra foi causada por obstrução direta da luz solar.

A Figura 13 ilustra a diferença de se utilizar a mediana dos desvios padrões ao invés do desvio padrão individual de cada *pixel* para gerar o modelo de fundo da cena (com-

(a) (b) (c)

Figura 12: (a) Modelo de *background*, (b) imagem em análise e (c) *foreground* detectado (em preto).

paração entre as equações (3.2) e (3.3)). A primeira linha exibe alguns quadros usados para gerar o modelo de fundo a partir de 100 imagens. A segunda linha exibe a imagem em análise, o resultado da subtração de *background* utilizando o desvio padrão individual para cada *pixel* (para modelagem da cena) e o resultado da subtração de *background* utilizando a mediana de todos os desvios padrões, respectivamente. Pode ser observado na imagem (e) que foram detectados menos *pixels* em movimento do que na imagem (f) (utilizar essa abordagem permite incorporar um intervalo maior de valores para o *background*), conseqüentemente, fazendo com que alguns objetos do *foreground* contenham algumas falhas de preenchimento, o que foi amenizado na imagem (f). Embora a modificação proposta gere alguns *pixels* isolados erroneamente classificados como *foreground*, tais *pixels* podem ser facilmente descartados usando algum critério de área mínima ou operadores morfológicos.

3.2.2 Detecção da sombra

Em regiões sombreadas, é esperado que uma determinada fração α de luz incidente seja bloqueada, como relatado por outros autores (ELGAMMAL et al., 2002). Mesmo que possam existir diversos fatores que podem influenciar o valor de intensidade de um *pixel* na sombra (SALVADOR; CAVALLARO; EBRAHIMI, 2004), nesse trabalho assume-se que a intensidade observada de um *pixel* na sombra é diretamente proporcional à luz incidente; conseqüentemente, *pixels* na sombra são versões escaladas (escurecidas) dos

(a) quadro 1 (b) quadro 50 (c) quadro 100

(d) imagem em análise (e) subtração de fundo (f) subtração de fundo

Figura 13: (a), (b) e (c) 3 dentre os 100 quadros usados para gerar os modelos de *background*; (d) imagem em análise; (e) subtração de fundo utilizando o desvio padrão individual de cada *pixel* e (f) subtração de fundo utilizando a mediana dos desvios padrões.

pixels correspondentes no modelo de *background*.

A correlação cruzada normalizada (*normalized cross-correlation – NCC*) é uma ferramenta matemática utilizada para procurar versões escaladas de um sinal/imagem em uma amostra maior (ROSENFELD; KAK, 1982). Basicamente o NCC é a normalização do produto interno (correlação cruzada) entre um *template* e uma imagem maior em relação a uma energia medida, produzindo o valor máximo se a intensidade do *template* é uma versão escalada de uma região da imagem maior.

O NCC já foi utilizado por outros autores para identificar e remover sobra em objetos do *foreground*, (GREST; FRAHM; KOCH, 2003; TIAN; LU; HAMPAPUR, 2005). Em (GREST; FRAHM; KOCH, 2003) o NCC é utilizado para fazer a detecção de *pixels* que pertencem à sombra (utilizando informação de luminosidade), entretanto é aplicado um pós-processamento (ou refinamento) utilizando informação de cor (no espaço de cor HSV). Diferentemente em nosso trabalho, o limiar do NCC é relaxado (objetivando-se detectar todos os *pixels* que pertencem à sombra – mesmo que *pixels* do *foreground* também sejam detectados como sombra) e posteriormente é proposto uma etapa de refinamento, utilizando imagens em escala de cinza, que visa eliminar *pixels* do *foreground* como parte da sombra.

3.2.2.1 Detecção de *pixels* candidatos a estar na sombra

Considere $\mathrm{B}(i,j) = \lambda(i,j)$ a imagem representativa do *background*, formada pela mediana de cada *pixel* no período de treinamento, e $I(i,j)$ uma imagem da seqüência do vídeo. Para cada *pixel* (i,j) pertencente ao *foreground*, um *template* T_{ij} com dimensões $(2N+1)\times(2N+1)$ tal que $T_{ij}(n,m) = I(i+n, j+m)$, para $-N \leq n \leq N$, $-N \leq m \leq N$ (T_{ij} corresponde a uma vizinhança do *pixel* (i,j), na maioria dos experimentos, utilizando $N = 4$). Dessa forma, o NCC entre o *template* T_{ij} e a imagem B no *pixel* (i,j) é dado por:

$$NCC(i,j) = \frac{ER(i,j)}{E_B(i,j)E_{T_{ij}}} \ , \tag{3.6}$$

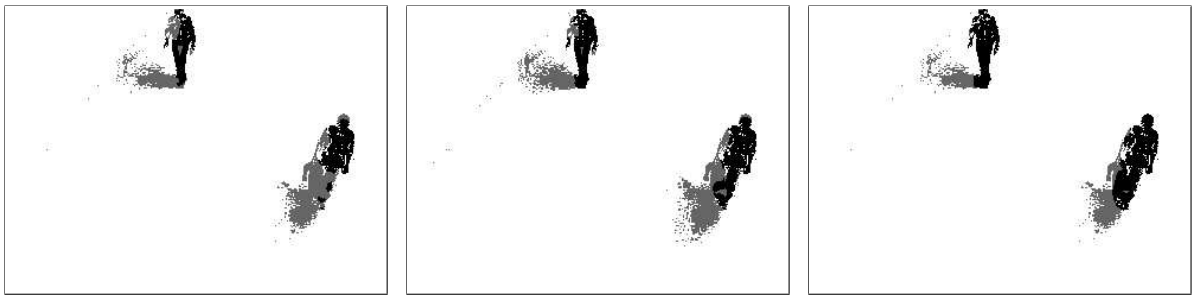
onde

$$\begin{aligned} ER(i,j) &= \sum_{n=-N}^{N}\sum_{m=-N}^{N} B(i+n,j+m)T_{ij}(n,m) \quad , \\ E_B(i,j) &= \sqrt{\sum_{n=-N}^{N}\sum_{m=-N}^{N} B(i+n,j+m)^2} \qquad \text{e} \\ E_{T_{ij}} &= \sqrt{\sum_{n=-N}^{N}\sum_{m=-N}^{N} T_{ij}(n,m)^2}. \end{aligned} \tag{3.7}$$

Para um *pixel* (i,j) em uma região sombreada, o NCC em uma vizinhança de T_{ij} deve ser alto (próximo de 1), e a energia $E_{T_{ij}}$ dessa região deve ser menor que a energia $E_B(i,j)$ da região correspondente na imagem do *background*. Dessa forma, um *pixel* (i,j) é pré-classificado como pertencente à sombra se:

$$NCC(i,j) \geq L_{ncc} \quad \text{e} \quad E_{T_{ij}} < E_B(i,j), \tag{3.8}$$

onde L_{ncc} é uma constante. Se L_{ncc} tiver um valor baixo, muitos *pixels* correspondentes ao *foreground* podem ser erroneamente classificados como pertencentes a sombra. Por outro lado, um valor elevado para L_{ncc} resulta numa diminuição de detecções erradas, porém *pixels* pertencentes à sombra podem não ser detectados. De fato, a influência do limiar L_{ncc} para a detecção *pixels* pertencentes à sombra pode ser observado na Figura 14. Essa figura ilustra a aplicação da técnica proposta para detecção de sombras nos *pixels* do *foreground* da Figura 12 (c), para diferentes valores de limiar (L_{ncc}). *Pixels* em preto são considerados *foreground* e *pixels* em cinza correspondem a *pixels* na sombra, de acordo com a equação (3.8). Nossos experimentos indicam que valores de $L_{ncc} = 0.95$ resultam em um bom comprometimento entre detecções falsas e não detecções.



(a) (b) (c)

Figura 14: Detecção inicial de sombra usando diferentes valores de limiar L_{ncc}. (a) $L_{ncc} = 0.90$, (b) $L_{ncc} = 0.95$ e (c) $L_{ncc} = 0.98$

3.2.2.2 Sombra: refinamento e segmentação

O NCC provê boas estimativas sobre a localização de *pixels* pertencentes à sombra, detectando *pixels* em uma vizinhança, que possuem aproximadamente uma versão escalada em relação ao modelo de *background*. Entretanto, alguns *pixels* do *background* relacionados à objetos em movimento podem ser classificados erroneamente como *pixels* pertencentes à sombra. Objetivando solucionar esse problema, um estágio de refinamento é aplicado para todos os *pixels* que satisfizerem a equação (3.8).

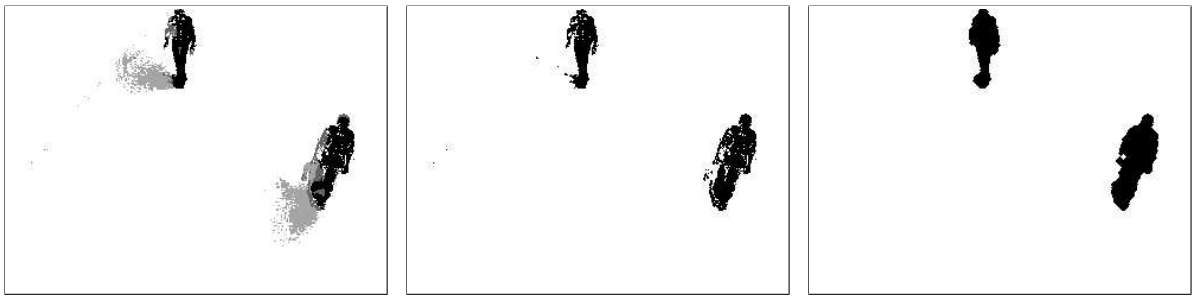
O estágio de refinamento proposto consiste em se verificar, para cada *pixel* candidato à estar na sombra, se a razão $I(i,j)/B(i,j)$, em uma vizinhança, é aproximadamente constante. Mais especificamente, considera-se uma região R com dimensão $(2M+1) \times (2M+1)$ *pixels* (normalmente $M = 1$) centrada em cada *pixel* candidato (i,j), classificando-o como pertencente à sombra se:

$$STD_R\left(\frac{I(i,j)}{B(i,j)}\right) < L_{STD} \quad \text{e} \quad L_{low} \leq \left(\frac{I(i,j)}{B(i,j)}\right) < 1, \tag{3.9}$$

onde $_{STD_R}\left(\frac{I(i,j)}{B(i,j)}\right)$ é o desvio padrão da razão $I(i,j)/B(i,j)$, quantificado em toda a região R, e L_{STD} e L_{low} são constantes. Mais precisamente, L_{STD} controla o desvio máximo para a região em análise (experimentos indicam que um valor adequado é $L_{STD} = 0.05$), e L_{low} previne a classificação errada de objetos muito escuros, *pixels* com valores de intensidade

muito baixos, como *pixels* pertencentes à sombra (experimentos indicam que um valor adequado é $L_{low} = 0.5$). Salienta-se que em dias ensolarados a sombra pode ser muito forte, e a informação sobre a intensidade dos *pixels* contidos na sombra pode ser perdida. Em casos como esse, $I(i,j)/B(i,j)$ é normalmente um valor muito baixo, e *pixels* da sombra podem ser classificados erroneamente como objetos do *foreground*.

Um exemplo da técnica de refinamento aplicada à detecção inicial da sombra exibida na Figura 12 (c) é ilustrado na Figura 15 (a). Na Figura 15 (a), *pixels* em cinza correspondem à detecção inicial da sombra e *pixels* em cinza claro correspondem à detecção final da sombra. Na Figura 15 (b) são exibidos todos os *pixels* do *foreground* após a remoção da sombra, e na Figura 15 (c) é exibido o resultado final após a aplicação de operadores morfológicos (concatenação de fechamento e abertura utilizando um elemento estruturante 5×5 com forma de diamante) para preencher "buracos" e remover *pixels* isolados.

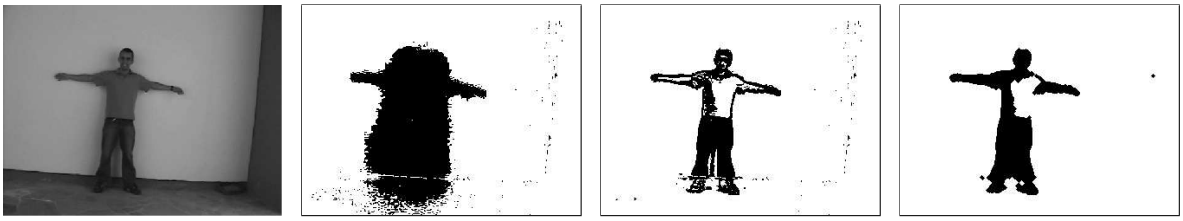


(a) (b) (c)

Figura 15: (a) Resultado final da detecção da sombra (*pixels* da sombra são representados por cinza claro). (b) Objetos do *foreground* após a remoção da sombra. (c) Eliminação de "buracos" e *pixels* isolados, com a utilização de operadores morfológicos.

Um problema ocasionado com a utilização da técnica proposta é a classificação errada de objetos válidos do *foreground* como *pixels* da sombra, em seqüências de vídeo contendo um *background* homogêneo com objetos do *foreground* homogêneo (e mais escuro). Tal problema pode acontecer porque o NCC pode ser muito alto em tal situação e o desvio padrão bastante baixo. Um exemplo dessa classificação errada pode ser visto na Figura 16, onde uma pessoa com uma camiseta de "cor" homogênea está em frente

a um fundo homogêneo (mais claro). A sombra é corretamente detectada ao redor da pessoa, porém a camiseta e partes da pele são classificadas erroneamente como sombra. Felizmente, na maioria das aplicações, o *background* apresenta algum tipo de textura, tornando este tipo de classificação falsa bastante incomum.



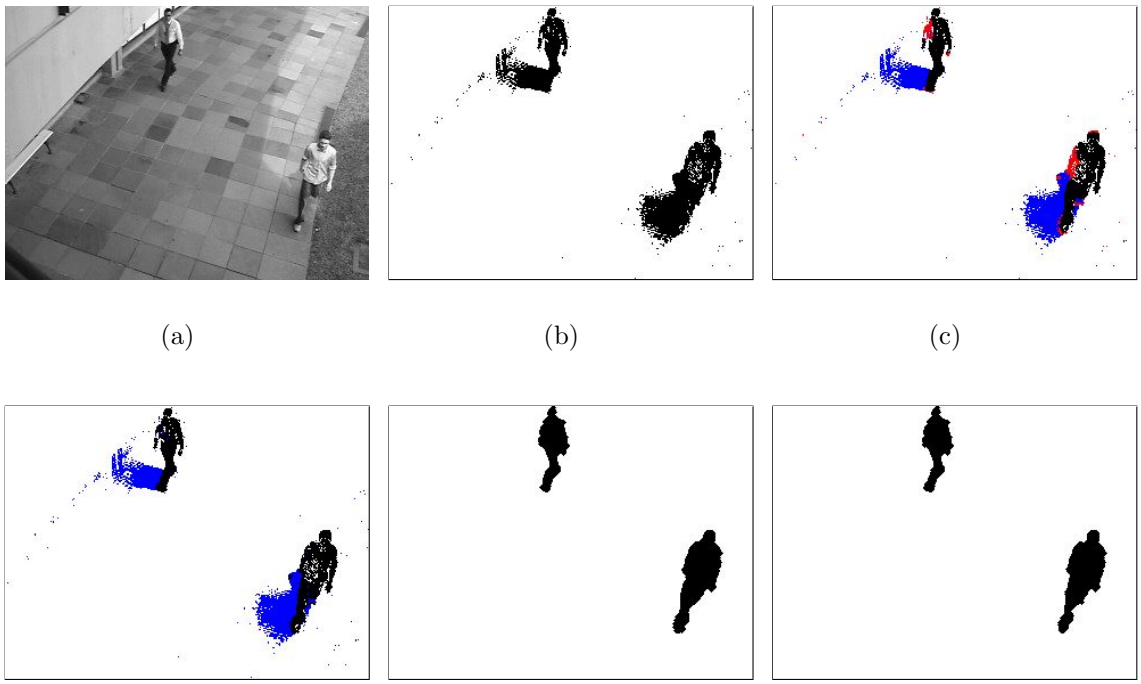
(a) (b) (c) (d)

Figura 16: Exemplo de detecção de sombra mal sucedida. (a) Imagem em escala de cinza. (b) *Pixels* do *foreground*. (c) Sombra removida. (d) Pós-processamento morfológico

Objetivando reduzir o custo computacional na etapa de remoção da sombra, o seguinte experimento foi conduzido: ao invés de aplicar o refinamento para os *pixels* que satisfazem a condição do teste do NCC, essa verificação (razão) foi feita diretamente em todos os *pixels* do *foreground*. Constatou-se que o resultado visual é muito semelhante ao obtido pela técnica descrita na seção 3.2.2, porém com um custo computacional reduzido. Utilizar a técnica com o teste do NCC resulta em uma detecção mais precisa, mas não utilizá-lo pode ser vantajoso por questões de simplicidade e ganho computacional, sem comprometer o resultado final de uma forma global. A Figura 17 ilustra o resultado obtido pela técnica de remoção de sombra, com e sem o cálculo do NCC. Uma análise quantitativa sobre a diferença entre usar ou não o teste do NCC pode ser objeto de estudo futuro. Os resultados apresentados no decorrer do texto foram gerados utilizando a descrição parcial, ou seja, sem a utilização do cálculo do NCC.

3.2.3 Atualização do modelo de *background*

Para que o modelo consiga suportar mudanças de iluminação, ou uma possível movimentação (indesejada) da câmera, o mesmo é atualizado periodicamente. Mudanças bruscas de iluminação afetam o desempenho do sistema como um todo, porém são repa-



(a) (b) (c)

(d) (e) (f)

Figura 17: Diferença entre usar ou não o NCC na etapa de remoção da sombra. (a) Imagem em análise. (b) *foreground* detectado. (c) em vermelho, *pixels* que passaram no teste do NCC e não passaram no teste da razão; em azul, *pixels* que passaram no teste do NCC e no teste da razão. (d) em azul, *pixels* que passaram no teste da razão (sem utilizar o NCC). (e) resultado final do caso (c). (f) resultado final do caso (d).

radas nessa etapa de atualização. Mudanças suaves de iluminação se adaptam ao modelo de *background* de forma pouco perceptível.

Usualmente, o modelo de *background* é atualizado a cada 100 quadros (aproximadamente 3.3 segundos, em uma seqüência de vídeo adquirida a 30 fps). Para isso, sempre são armazenadas as ultimas 100 imagens geradas na etapa de subtração de fundo (antes da remoção da sombra) também como as imagens em escala de cinza em análise. Existem três condições para que um *pixel* seja atualizado:

- *pixel* do *background*: para cada *pixel* (i, j) da imagem que for classificado como *background* em mais de 80% dos quadros do período de atualização (usualmente 100 quadros), são armazenados em um vetor $V^t(i, j)$ (onde t representa o tempo), os seus valores em escala de cinza dos quadros nos quais foi classificado como *background*. Dessa forma, para esse *pixel* (i, j) serão atualizados seus valores mínimo, máximo, mediano e máxima diferença absoluta entre quadros consecutivos da seguinte maneira:

$$m(i,j)_{new} = \alpha m(i,j) + (1-\alpha)\ \min_t(V^t(i,j))\ , \tag{3.10}$$

$$n(i,j)_{new} = \alpha n(i,j) + (1-\alpha)\ \max_t(V^t(i,j))\ , \tag{3.11}$$

$$\lambda(i,j)_{new} = \alpha \lambda(i,j) + (1-\alpha)\ \text{median}_t(V^t(i,j)), \tag{3.12}$$

$$d(cont)_{new} = \ \max_t\ |V^t(i,j) - V^{t-1}(i,j)|, \tag{3.13}$$

  onde α é o fator de atualização utilizado (usualmente $\alpha = 0.5$, para $0 < \alpha < 1$), m, n, λ e d são os valores mínimo, máximo, mediano e máxima diferença absoluta entre quadros consecutivos, respectivamente. *cont* é um contador para os *pixels* que satisfazem essa condição. Utilizar um valor muito baixo para α faz com que o valor da variável seja atualizado dando mais prioridade à informação presente do que passadas (MCKENNA et al., 2000).

- *pixel* do *foreground*: para cada *pixel* (i, j) da imagem que for classificado como *pixel*

do *foreground* em mais de 80% dos quadros do período de atualização (usualmente 100 quadros), são armazenados em um vetor $S^t(i,j)$ (onde t representa o tempo), os seus valores em escala de cinza dos quadros nos quais foi classificado como *foreground*. Então, para todo *pixel* (i,j), que satisfizer essa condição e cuja mediana das máximas diferenças absolutas de S for menor ou igual que a mediana das máximas diferenças de todo o modelo de *background* é feita a atualização da seguinte forma:

$$m(i,j)_{new} = \min_t(S^t(i,j)) \ , \tag{3.14}$$

$$n(i,j)_{new} = \max_t(S^t(i,j)) \ , \tag{3.15}$$

$$\lambda(i,j)_{new} = \text{median}_t(S^t(i,j)) \tag{3.16}$$

Essa condição faz com que *pixels* considerados *foreground* por muito tempo, cuja variação de intensidade seja muito pequena, sejam incorporado ao *background*. Dessa forma, um objeto inserido na cena é incorporado ao fundo após um determinado tempo. Da mesma forma, quando um objeto é retirado da cena, o modelo de *background* é capaz de suportar tal modificação, adaptando-se à mudança.

- os demais *pixels* que não satisfizerem nenhuma das duas condições anteriores permanecem com seus valores inalterados.

Após o processamento de toda a imagem, é então atualizado o valor da mediana das máximas diferenças absolutas da seguinte forma:

$$\mu_{new} = \alpha\mu + (1-\alpha)\ \text{median}_t(d(cont)), \tag{3.17}$$

onde d é o vetor que contém as máximas diferenças absolutas entre quadros consecutivos dos *pixels* que passaram na primeira condição do período de atualização (equação (3.13)), μ é a mediana das máximas diferenças absolutas entre quadros consecutivos antes da atualização, e α é o fator de atualização utilizado (usualmente $\alpha = 0.5$, para $0 < \alpha < 1$).

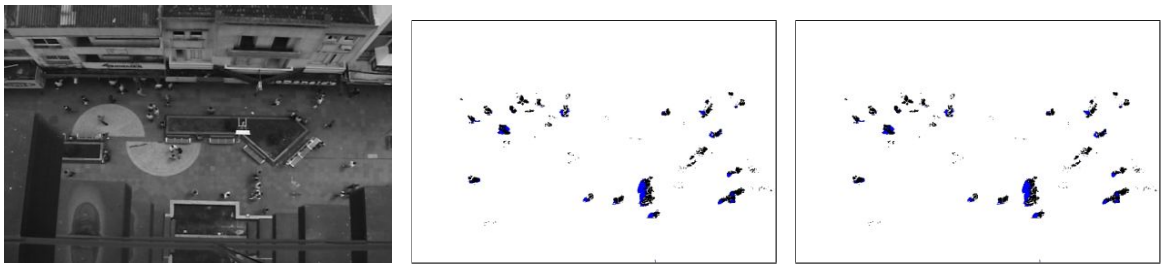
A Figura 18 exibe o resultado da subtração de fundo para quatro 4 de uma seqüência de vídeo, com e sem atualização do modelo de *background*. A primeira coluna exibe as imagens em análise e a segunda coluna exibe o resultado da subtração de fundo sem atualização do modelo de *background*. Nesse caso, uma nuvem causou um bloqueio parcial da luz por um determinado tempo, e pode-se observar que o algoritmo de remoção de sombra foi capaz de identificar essa mudança. Entretanto, como o algoritmo de remoção de sombra é aplicado à todos os *pixels* classificados como *foreground*, em casos onde muitos *pixels* são classificados como *foreground*, há um custo computacional envolvido. Se o problema fosse invertido, ou seja, uma nuvem que estava causando um bloqueio parasse de causá-lo, faria com que muitos *pixels* fossem classificados como *foreground* (indesejavelmente) mesmo após a remoção da sombra. A terceira coluna da Figura 18 exibe o resultado da subtração de fundo com a utilização da adaptatividade do modelo de *background*.

3.3 *Tracking*

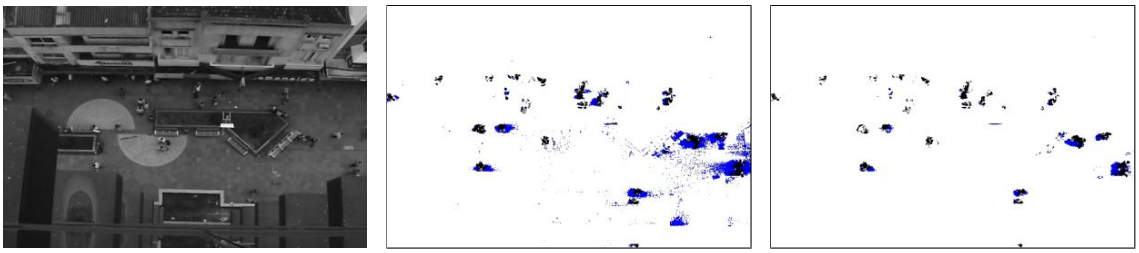
Nesse trabalho utiliza-se um procedimento de *tracking* automático simples, baseado em características de correlação. Basicamente, a idéia é analisar cada objeto do *foreground* (como por exemplo, sua área, seu ponto de referência, sua distância das bordas da imagem, etc.) e verificar se o mesmo é um novo objeto na cena ou não. Se for um novo objeto, deve ser capturado seu *template* inicial e correlacionado com o próximo quadro da seqüência de vídeo, caso contrário, o objeto já identificado anteriormente continua sendo acompanhado. Esse procedimento será melhor descrito no decorrer da seção.

3.3.1 Análise do objeto do *foreground*

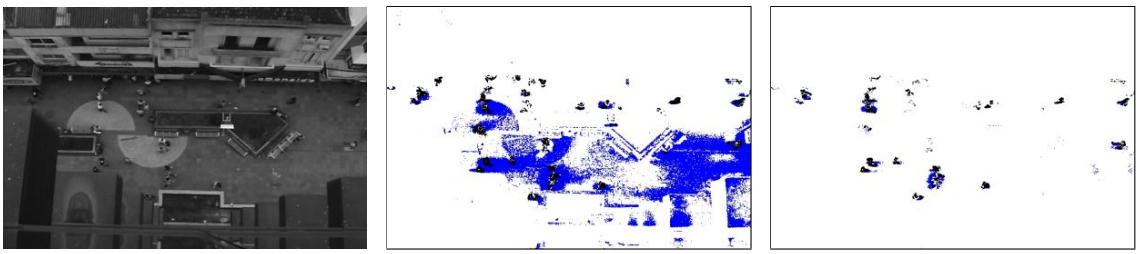
No resultado da técnica de subtração de fundo (após a remoção da sombra e operações morfológicas) é gerada uma imagem binária contendo os elementos do *foreground*. Para acompanhar as pessoas no decorrer do tempo pode ser necessário distingui-



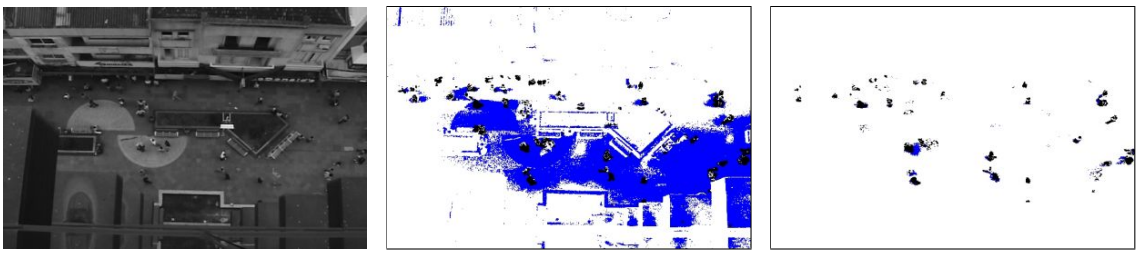
(a) 1 (b) 1 (c) 1



(d) 250 (e) 250 (f) 250



(g) 500 (h) 500 (i) 500



(j) 1000 (k) 1000 (l) 1000

Figura 18: Modelo adaptativo. Primeira coluna: imagens de entrada. Segunda coluna: sem atualizar o modelo de *background* (pontos em preto representam os objetos do *foreground* e em azul da sombra). Terceira coluna: atualizando o modelo.

las corretamente dos outros objetos (como por exemplo veículos, pássaros, vegetação, etc.). Entretanto, essa etapa de classificação foi bastante simplificada nesse trabalho, pois assume-se que os objetos capturados pela câmera são apenas pessoas. Essa classificação é feita apenas pela área de cada objeto.

O primeiro critério para capturar o *template* inicial de um objeto é sua medida de área. A área de cada objeto é medida pelo número de elementos conexos (*pixels*) que ele possui. Dessa forma, todo objeto que possuir uma área maior que um determinado valor de limiar é um possível candidato a ser acompanhado. O valor desse limiar é escolhido previamente. Baseando-se na posição da câmera, é estabelecida uma relação de metros $\leftrightarrow$ *pixels* e o limiar é definido como a área de uma elipse com aproximadamente 30cm x 23cm, ou seja, a metade da área média de uma pessoa vista de cima (FRUIN, 1971) (utilizou-se esse valor com o objetivo de não descartar pessoas que possam apresentar uma área pequena). Acredita-se que esse valor de limiar pode ser obtido automaticamente em um período de aprendizado, tornando o algoritmo mais robusto, pois as pessoas apresentam um padrão de forma, aparência, e movimento diferente de outros objetos, que pode ser usado como característica em uma etapa de classificação.

O segundo critério para que o objeto possa ser acompanhado é que ele esteja a uma distância mínima das bordas da imagem, assumindo que pessoas só possam surgir nas extremidades da imagem (objetos muito próximos das bordas não são levados em consideração). A Figura 19 ilustra um objeto que possui uma área considerável porém ainda está muito próximo da borda da imagem (a pessoa não entrou completamente na cena).

Em diversos trabalhos encontrados na literatura, o centróide do objeto é utilizado como um ponto para sua referência. Alguns trabalhos buscam identificar as partes do corpo, podendo estimar a posição da cabeça da pessoa. Nesse trabalho optamos por estimar a posição da cabeça da pessoa, pois acreditamos que dessa forma conseguiremos acompanhar a pessoa de forma suave (de uma vista superior, a cabeça normalmente

Figura 19: Análise da área e distância das bordas da imagem do objeto.

permanece invariante, o que facilita a correlação entre quadros consecutivos).

Nesse trabalho foram testados 4 métodos para estimar a posição da cabeça da pessoa, descritos a seguir. Dentre os 4, optou-se por utilizar a "transformada da distância" quando a câmera é de topo (pois apresentou os melhores resultados na maioria dos experimentos, por ser invariante ao sentido em que a pessoa caminha e por se mostrar independente das posições das pernas e braços - o que é um complicador quando se deseja utilizar o centróide por exemplo) e "ponto mínimo em y e médio em x" em alguns experimentos cujo campo de visão da câmera é oblíquo (por questão de simplicidade, já que essa não é a principal posição de câmera usada nesse trabalho).[1]

- centro: é posicionado um *bounding-box* nos limites do objeto em análise e é capturado o seu ponto médio em x e em y.

- ponto mínimo em y e médio em x: é posicionado um *bounding-box* nos limites do objeto em análise e é capturado o seu ponto médio em x e mínimo em y (mais um determinado valor - usualmente a metade do tamanho do *template*). Assumindo-se que a cabeça da pessoa esteja sempre direcionada para a parte superior da imagem.

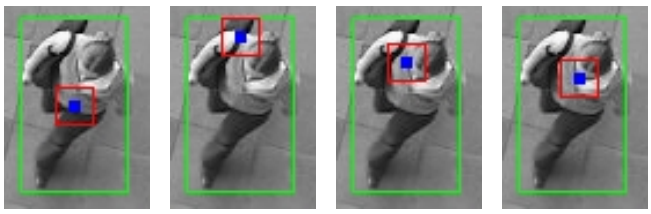
- histograma em y e ponto médio em x: projeta-se os pontos do objeto (imagem binária) sobre o eixo y. A coordenada de máxima ocorrência é a estimativa da posição da cabeça na coordenada y. Essa técnica é afetada quando a pessoa está

[1] Neste trabalho, os eixos x e y representam, respectivamente, as orientações horizontal e vertical, com a origem no canto superior esquerdo da imagem.

caminhando no mesmo sentido do eixo x, devendo ser utilizado a projeção sobre o eixo x.

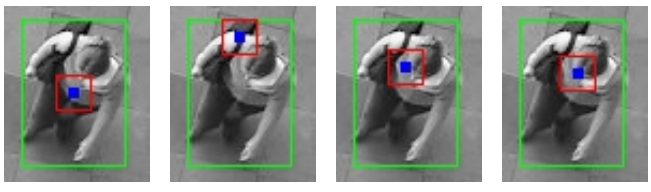
- transformada da distância: o ponto de máximo global da transformada da distância retorna o centro do maior círculo inscrito no objeto. Em vistas de topo, tal posição deve coincidir aproximadamente com o centro da cabeça da pessoa.

As Figuras 20, 21 e 22 exibem o resultado dos quatro métodos aplicados em três situações distintas. Considere $I(i, j)$ a imagem em análise, em escala de cinza. É definido o *template* T_{ij} com dimensão $(2N + 1) \times (2N + 1)$ tal que $T_{ij}(n, m) = I(i + n, j + n)$, para $-N \leq n \leq N$, $-N \leq m \leq N$ (T_{ij} corresponde à vizinhança do $pixel(i, j)$, centro do *template*). Após capturar o *template* inicial da pessoa, objetiva-se correlacionar esse *template* com o próximo quadro da seqüência de vídeo.



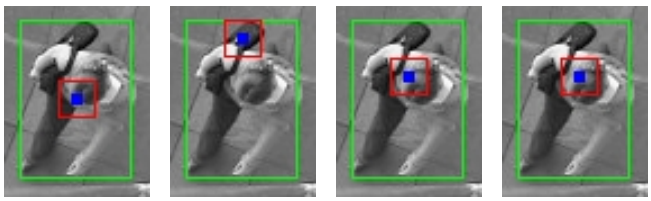
(a) (b) (c) (d)

Figura 20: Exemplo de estimativas para a cabeça. O quadrado vermelho representa o *template* capturado e o ponto azul o seu centro, posição estimada. (a) Centro. (b) Ponto mínimo em y e médio em x. (c) Histograma em y e ponto médio em x. (d) Transformada da distância.



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

Figura 21: Exemplo de estimativas para a cabeça. O quadrado vermelho representa o *template* capturado e o ponto azul o seu centro, posição estimada. (a) Centro. (b) Ponto mínimo em y e médio em x. (c) Histograma em y e ponto médio em x. (d) Transformada da distância.



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

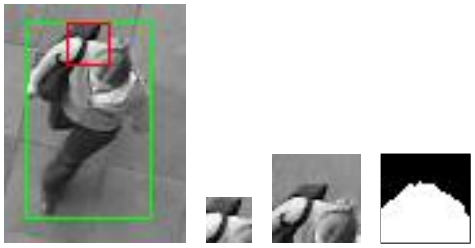
Figura 22: Exemplo de estimativas para a cabeça. O quadrado vermelho representa o *template* capturado e o ponto azul o seu centro, posição estimada. (a) Centro. (b) Ponto mínimo em y e médio em x. (c) Histograma em y e ponto médio em x. (d) Transformada da distância.

3.3.2 Estabelecendo uma correlação entre quadros consecutivos

Há diversas métricas de correlação desenvolvidas para estabelecer uma relação entre uma imagem pequena e outra grande, como a Soma das Diferenças Quadráticas (*Sum of Squared Difference - SSD*), Correlação Cruzada Normalizada (*Normalized Cross-Correlation - NCC*), Correlação Cruzada Normalizada com média zero (*Zero-Mean Normalized Cross-Correlation - ZMNCC*). Entretanto, Martin e Crowley (MARTIN; CROWLEY, 1995) concluíram que a SSD provê resultados mais estáveis que NCC ou ZNCC em aplicações genéricas, influenciando a decisão do autor, de maneira que neste trabalho também seja utilizado a SSD na etapa de *tracking*.

É esperado que o *template* de cada pessoa seja deslocado para uma região vizinha no próximo quadro, com o movimento da pessoa. Um deslocamento máximo pode ser estimado levando-se em consideração a taxa de aquisição do vídeo (*Frames Per Second - FPS*) e a velocidade máxima permitida para cada pessoa. Por exemplo, se a taxa de aquisição é de 15 FPS e a velocidade máxima permitida é de 5m/s, então o centro do *template* não pode ser deslocado mais do que 5 x (1/15) = 0.33 metros no quadro seguinte. Então a correlação entre o *template* original T e a região de sua vizinhança é computada, e o centro do *template* é movido para o ponto relacionado ao mínimo global da SSD. Tal procedimento é repetido para todo quadro subseqüente, até que a pessoa desapareça do campo de visão da câmera.

Salienta-se que a área de busca está restrita aos objetos do *foreground* do próximo quadro. Isso é um fator que pode ser utilizado para otimização computacional, já que a SSD não precisa ser computada para todos os pontos da área de busca, também como elimina a possibilidade do *template* possuir uma correlação grande com um objeto do *background*, o que poderia ocorrer se o *template* da pessoa possuísse mais pontos do *background* do que do *foreground*. A Figura 23 exibe um *template* capturado (utilizando-se o "ponto mínimo em y e médio em x") e sua área de busca. Um fator complicador quando se utiliza essa abordagem é quando o objeto pessoa (retornado da subtração do *background*) é dividido (*split*)[2] e por possuir uma área muito pequena não é levado em consideração no quadro atual. Para tentar solucionar isso, foi introduzido uma condição onde o objeto que está sendo acompanhado pode ficar no máximo m quadros sem um respectivo objeto de *foreground* (com $m = 5$ para todos resultados apresentados no decorrer do texto). Quando isso ocorre, a área de busca binária para esse objeto é preenchida com 1 (ou seja, a SSD é computada em todos os pontos). A pessoa que permanecer mais que m quadros sem pontos do *foreground* terá sua trajetória finalizada.



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

Figura 23: Exemplo de correlação. (a) *bounding-box* do objeto. (b) *template* de correlação do tempo i. (c) área de busca do tempo $i + 1$. (d) área de busca binária (informa onde a SSD deve ser computada - em branco).

Uma limitação de técnicas de correlação baseadas em *templates* é que o objeto de interesse representado pelo *template* deve permanecer invariante (por exemplo, o *template* não deve sofrer rotação, escala, e sua intensidade deve permanecer a mesma durante

[2]em dois ou mais blocos conexos de pequena área, de tal modo que nenhum desses blocos satisfaça o critério de área mínima.

diversos quadros). Tal condição não é encontrada em configurações de câmeras oblíquas, porque a aparência das pessoas é extremamente dependente de suas orientações e posições com relação a câmera. Entretanto, assume-se que a vista de topo da cabeça de uma pessoa é aproximadamente circular, que é invariante a rotações. Além disso, para trabalhar com mudanças suaves de iluminação e pequenas variações da posição da cabeça, o *template* é atualizado a cada Q quadros (usualmente, $Q = 5$, para seqüências adquiridas a 15 FPS) da seguinte forma:

$$T_{new} = \alpha T_{old} + (1 - \alpha) T_{last}, \tag{3.18}$$

onde α é o fator de atualização utilizado (usualmente $\alpha = 0.5$, para $0 < \alpha < 1$), T_{old} é o *template* antes da atualização e T_{last} é o *template* capturado para o tempo atual. Utilizar um valor muito baixo para α faz com que o *template* seja atualizado dando mais prioridade à informação presente do que passadas (MCKENNA et al., 2000).

Outro fator que deve ser levado em consideração é quando mais de uma pessoa entram muito próximas, originando um único objeto conexo de *foreground*. Quando isso ocorre, o objeto é acompanhado como sendo uma única pessoa. Se em algum momento essas pessoas de dividirem, originando dois objetos com área suficiente para serem acompanhados, o algoritmo é capaz de capturar o *template* inicial para o novo objeto detectado e inicializar uma trajetória para o mesmo.

Quando duas pessoas que estão andando separadamente se aproximam, originando um único objeto (e as duas já estavam sendo acompanhadas) o algoritmo também é capaz de continuar acompanhando as mesmas de forma natural, pois o *tracking* é realizado em duas etapas. Em uma primeira etapa, todo objeto, que contém um identificador único (associado a sua trajetória), deve ser correlacionado com o quadro seguinte. Após correlacionar todos os objetos do quadro atual, verifica-se se algum objeto que estava sendo acompanhado não foi atualizado, caso afirmativo, esse objeto é então correlacionado (caracterizando a segunda etapa do *tracking*). Para verificar a qual trajetória um

determinado objeto pertence é feita a seguinte verificação: a trajetória que possuir a coordenada (no tempo corrente) dentro do objeto do *foreground* é atribuída ao mesmo. Dessa forma, quando dois objetos se fundem, um deles será acompanhado pela primeira etapa do *tracking* e o outro na segunda etapa.

A Figura 24 exibe o resultado do *tracking*, para uma determinada seqüência de vídeo, utilizando a técnica proposta nesse trabalho. A imagem (a) representa o quadro inicial, ou seja, quando o *template* da pessoa foi capturado pela primeira vez. As imagens (b) e (c) exibem a posição do *template* atualizada pelo método de correlação.

(a) quadro 1 (b) quadro 10 (c) quadro 20

Figura 24: Resultado do *tracking*. O quadrado vermelho representa o *template* da pessoa. (a) *template* inicial. (b) posição da cabeça após 10 quadros. (c) posição da cabeça após 20 quadros.

3.4 Análise dos dados

Com o resultado do procedimento de *tracking*, pode-se determinar a trajetória de cada pessoa capturada pela câmera. Acredita-se que é mais adequado utilizar seqüências de vídeo não muito densas para gerar os campos de velocidades, porque em situações de alta densidade as pessoas podem sofrer muitas alterações nas suas trajetórias, devido à interação com outras pessoas (além disso, o procedimento de *tracking* pode apresentar resultados errados em situações muito densas). Nesta seção é descrita a abordagem utilizada para gerar campos de velocidades densos (onde é estimado um vetor velocidade para cada ponto do espaço - por extrapolação), assim como a métrica proposta para comparar grupos de trajetórias (Mapas de Ocupação Espacial).

3.4.1 Geração dos campos de velocidades

Cada trajetória capturada é representada por uma seqüência de vetores velocidade. A cada *pixel* onde uma pessoa foi acompanhada é associado o respectivo vetor velocidade, que será utilizado para alimentar o simulador de multidões. Entretanto, normalmente há diversas regiões da imagem por onde nenhuma pessoa passou (ou seja, as trajetórias acompanhadas geram um campo vetorial esparso).

Há diversas abordagens para se gerar um campo de vetores de velocidades denso a partir de outros esparsos, podendo ser citados métodos de interpolação, como *nearest neighbor*, linear, cúbica, *spline*. Em nossos experimentos, não há trajetórias nas bordas da imagem, indicando que uma técnica de extrapolação deve ser usada. De fato, a interpolação *nearest neighbor* pode ser facilmente estendida para extrapolação, sem propagar muito o erro das regiões extrapoladas como em outras técnicas de interpolação (como por exemplo, linear ou cúbica). Um exemplo de um campo extrapolado a partir de 17 trajetórias capturadas, ilustradas com auxílio da Figura 25, pode ser observado na Figura 26. Salienta-se que, nesse experimento controlado, as pessoas foram orientadas a andar em um único sentido (sul → norte), diferentemente de uma situação real normal, onde as pessoas poderiam caminhar em diversas direções.

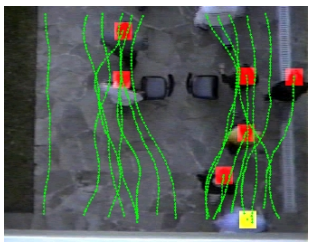


Figura 25: Trajetórias capturadas.

Quando as pessoas caminham em diversas direções, o seguinte problema pode acontecer, no processo de extrapolação das trajetórias: trajetórias com sentidos opostos podem se anular e indesejavelmente, gerar pontos no espaço com velocidade igual a zero, ou incoerentes. O campo de vetores também poderia conter diversos vetores adjacentes com direções opostas, o que poderia resultar em em trajetórias oscilantes não realistas.

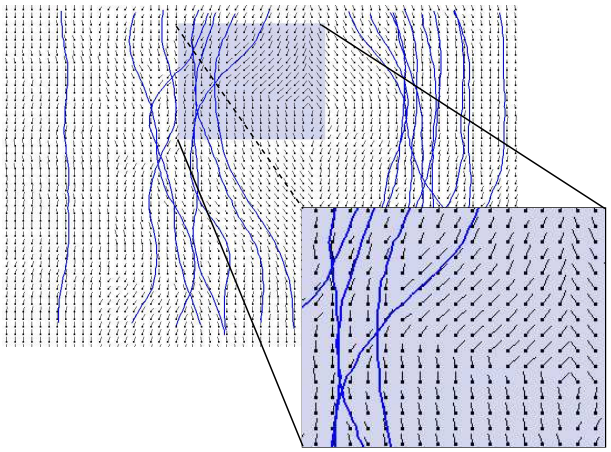
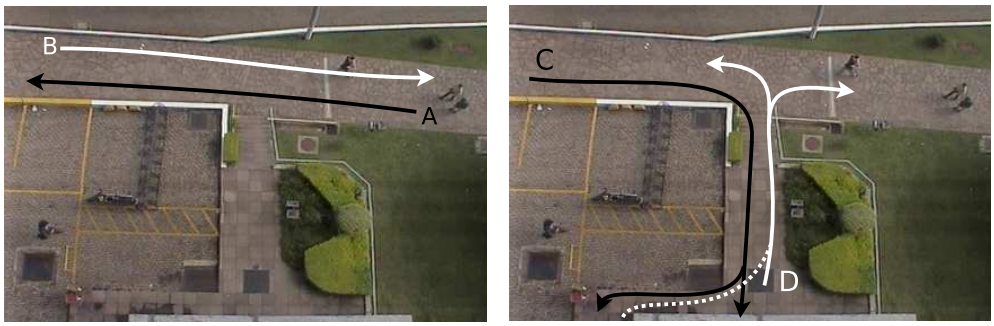


Figura 26: Campo de vetores gerado a partir de 17 trajetórias capturadas.

Objetivando solucionar esse problema, é feita uma classificação das trajetórias, como exemplificado a seguir.

Uma classificação bastante simples é exemplificada com auxílio da Figura 27, onde diversas pessoas foram acompanhadas, utilizando-se o procedimento de *tracking* descrito anteriormente, e suas trajetórias foram manualmente classificadas em 4 tipos: (A) todas as trajetórias com origem no lado direito que foram em direção à esquerda; (B) trajetórias com origem no lado esquerdo que foram em direção à direita; (C) trajetórias com origem no topo que foram em direção à parte inferior e (D) trajetórias com origem na parte inferior que se dirigiram à parte superior. Nesse caso, 4 campos de velocidades independentes poderão ser gerados baseados nas trajetórias de mesma classe.



(a) Trajetórias do tipo A e B (b) Trajetórias do tipo C e D

Figura 27: Exemplo de classificação manual das trajetórias.

Salienta-se que no procedimento manual de classificação das trajetórias, ilustrado

na Figura 27, foram levadas em consideração apenas a posição inicial e final da pessoa no decorrer da cena, ou seja, de onde ela vem e para onde ela vai. Objetivando automatizar esse processo, foi utilizada uma técnica de *clustering*, descrita a seguir.

A definição de trajetórias coerentes (ou similares) é muito dependente da aplicação. Por exemplo, em (JUNEJO; JAVED; SHAH, 2004) e (MAKRIS; ELLIS, 2005) a distância espacial entre trajetórias é uma característica importante para o processo de *clustering*. Para o nosso propósito, trajetórias semelhantes são aquelas que possuem aproximadamente o mesmo vetor deslocamento (por exemplo, duas trajetórias que vão da esquerda para a direita, desconsiderando sua velocidade média e distância entre as mesmas). Para uma classificação automática e agrupamento de trajetórias semelhantes, primeiramente são extraídas características relevantes e então uma técnica não supervisionada de *clustering* é aplicada.

Considere que $(x(s), y(s)), s \in [0, 1]$, denote uma trajetória reparametrizada pelo comprimento de arco (normalizada à unidade), tal que $(x(0), y(0))$ seja o ponto inicial e $(x(1), y(1))$ o ponto final da trajetória. Cada trajetória é então caracterizada por um conjunto de N_d vetores deslocamento $\mathbf{d}_i = (\Delta x_i, \Delta y_i)$ computados a distâncias equidistantes:

$$\mathbf{d}_i = (x(t_i + 1) - x(t_i), y(t_i + 1) - y(t_i)),\ t_i = \frac{i}{N_d},\ i = 0, ..., N_d - 1 \tag{3.19}$$

Para cada trajetória j, é obtido um vetor ($2N_d$-dimensional) de características $\mathbf{f}_j$ pela combinação dos N_d vetores deslocamentos associados à trajetória:

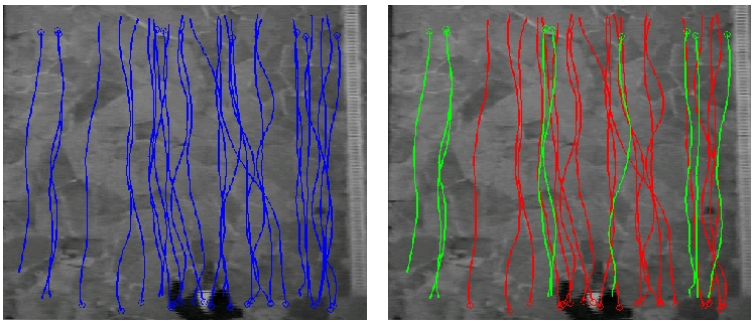
$$\mathbf{f}_j = (\Delta x_0, \Delta y_0, \Delta x_1, \Delta y_1, ..., \Delta x_{N_d-1}, \Delta y_{N_d-1}), \tag{3.20}$$

e o algoritmo de *clustering* (FIGUEIREDO; JAIN, 2002) é aplicado aos vetores de características $\mathbf{f}_j$.[3] O número N_d, de vetores deslocamento usados para montar $\mathbf{f}_j$, é escolhido baseado em quão estruturado o fluxo das pessoas é. Para trajetórias relativamente sim-

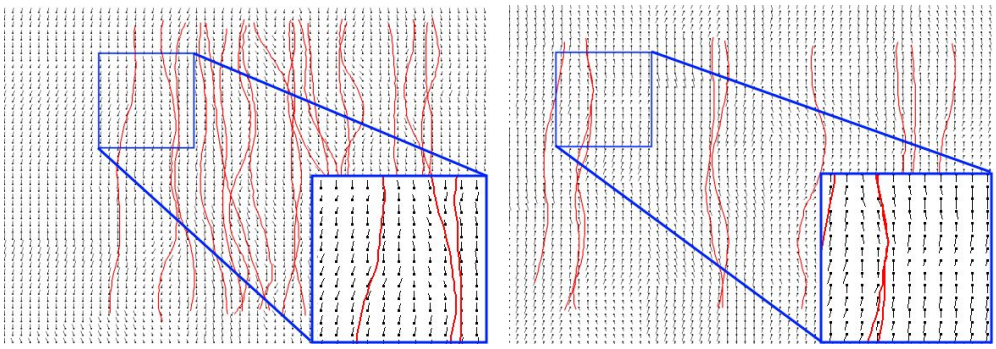
[3]Tal técnica de *clustering* foi selecionada por ser não supervisionada, e apresentar código disponível.

ples, valores pequenos para N_d podem capturar a essência das trajetórias. Por outro lado, trajetórias mais complicadas (com muitas curvas) são melhores caracterizadas usando-se valores maiores para N_d. Em geral, espaços públicos tendem a apresentar direções principais de fluxo, e $N_d = 2$ é usualmente uma boa escolha. Deve ser salientado que essa técnica de *clustering* não é apropriada para movimentos não estruturados (como por exemplo, jogadores em uma partida de futebol, ou crianças brincando), desde que suas trajetórias não apresentem semelhanças como um todo.

A Figura 28 exibe o resultado da técnica de *clustering* aplicada em um conjunto de trajetórias, ilustradas na Figura 28(a). Nesse caso, trajetórias que se deslocam para cima como para baixo foram postas corretamente em duas diferentes classes, marcadas com diferentes cores (Figura 28(b), vermelho e verde, respectivamente). Os campos de vetores extrapolados gerados podem ser visualizados nas Figuras 28(c) e (d).



(a) (b)



(c) (d)

Figura 28: (a) Trajetórias capturadas. (b) Trajetórias classificadas em duas classes. (c) Campo de vetores gerado para a Classe 1. (d) Campo de vetores gerado para a Classe 2.

A Figura 29 exibe o resultado da técnica de *clustering* aplicada a um conjunto de trajetórias capturadas no estudo de caso denominado "Bifurcação em T" ("região T"). Cabe salientar que a classificação manual, ilustrada na Figura 27 para a "região T" enfatiza os benefícios da técnica de *clustering* utilizada, já que para o mesmo cenário, as classificações automática e manual se mostraram bastante semelhantes, gerando exatamente o mesmo número de classes. Na Figura 30 são exibidos os 4 campos de vetores velocidades que foram gerados a partir das 4 classes de trajetórias (retornadas da técnica de *clustering* utilizada).

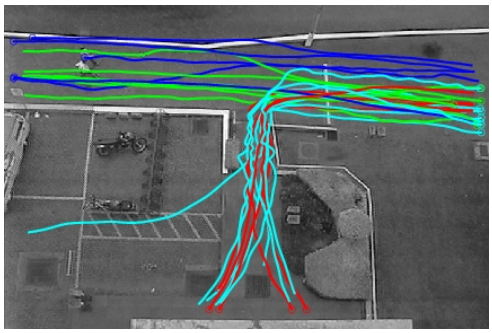
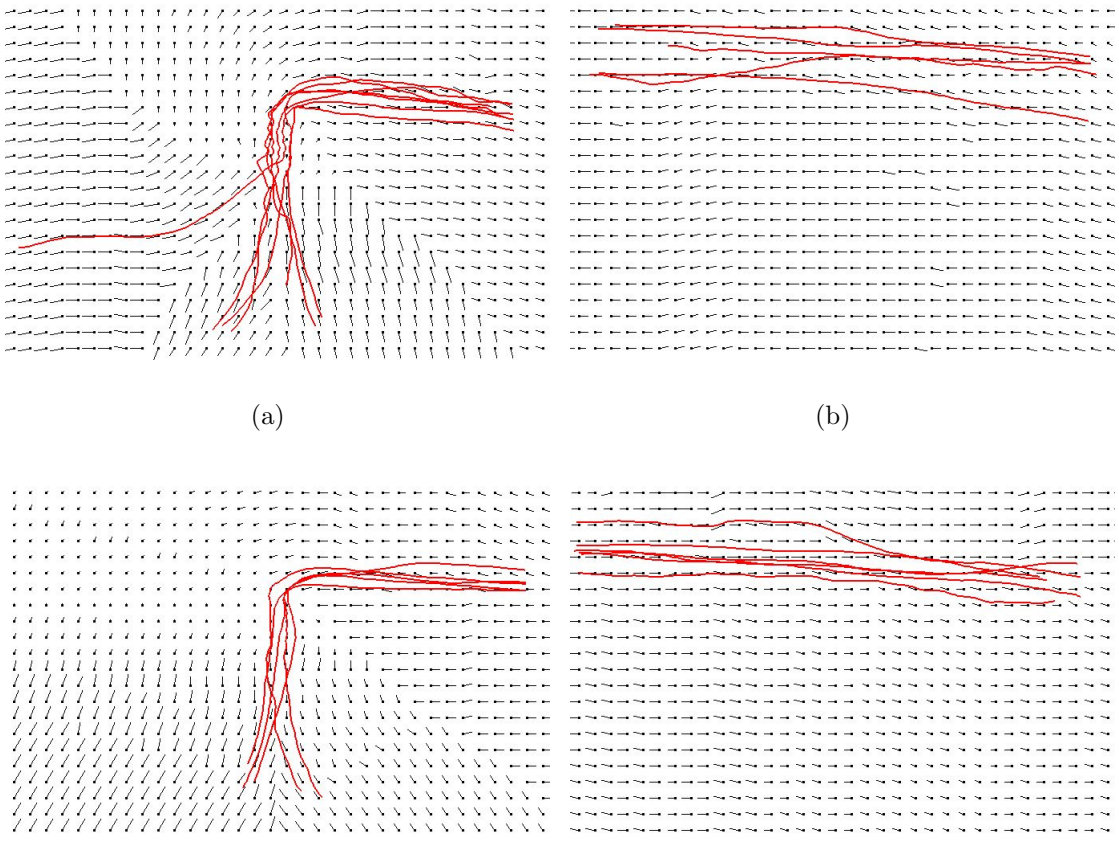


Figura 29: Trajetórias agrupadas em 4 classes diferentes.

3.4.2 Mapas de Ocupação Espacial

Objetivando comparar resultados de simulação com os dados capturados da vida real, introduzimos os Mapas de Ocupação Espacial (SOMs - *Spatial Occupancy Maps*) como uma métrica de comparação coerente entre dois grupos de trajetórias. Em geral, comparar trajetórias individuais de uma maneira microscópica não é adequado quando estamos interessados na dinâmica de multidões (pessoas de dois grupos diferentes podem se comportar diferentemente sob um ponto de vista micro, e de maneira similar sob uma visão global do movimento da multidão). Por outro lado, comparar grupos de pessoas apenas utilizando informação global (por exemplo, tempo total de evacuação) pode não representar reais similaridades entre duas multidões. Outras métricas que utilizam informação global e local também podem ser usadas, como tempo total para percorrer uma determinada distância, ou velocidade local em regiões específicas.

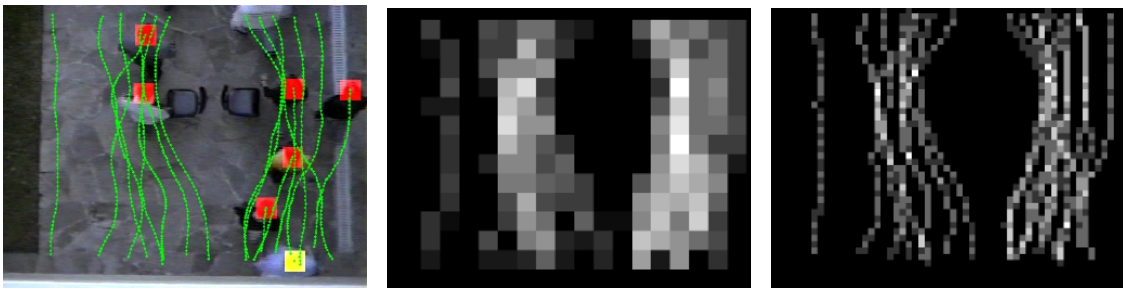


(a) (b)

(c) (d)

Figura 30: Campos de vetores gerados para as classes de pessoas que se locomovem nos sentidos: (a) superior → inferior, (b) esquerda → direita, (c) inferior → superior e (d) direita → esquerda.

O SOM é uma matriz de mesmo tamanho da região de interesse (assumindo-se que seja retangular) inicializada com zeros. A matriz é então sub-dividida uniformemente em $N_s \times M_s$ sub-retângulos, correspondendo a $N_s \times M_s$ sub-matrizes do SOM. A cada passo de tempo (*timestep*) que uma pessoa é detectada em qualquer dos sub-retângulos, todos os elementos da sub-matriz correspondente são incrementados. Após um determinado intervalo de tempo, cada sub-matriz do SOM representa o número de vezes (ou número de quadros) que o retângulo correspondente foi ocupado por uma pessoa. Salienta-se que o SOM provê informação sobre a ocupação do espaço, o que é uma combinação do número de pessoas e suas velocidades (uma pessoa movendo-se lentamente, ocupa o mesmo espaço por um período de tempo maior que uma pessoa mais veloz). Também, o tamanho dos sub-retângulos influenciam a escala da análise: sub-retângulos grandes provêm uma descrição mais global do espaço, enquanto que sub-retângulos menores podem ser usados para análise local, como ilustrado na Figura 31.



(a) (b) (c)

Figura 31: (a) Trajetórias capturadas; SOMs obtidos utilizando sub-retângulos com tamanhos (b) 21 × 21 e (c) 5 × 5.

Nas Figuras 31 (b) e (c), são exibidos dois SOMs, gerados a partir das trajetórias exibidas na Figura 31 (a). Sub-retângulos mais claros indicam regiões mais ocupadas. Dois tamanhos de sub-retângulos são ilustrados, 21 × 21 (Figura 31 (b)) e 5 × 5 (Figura 31 (c)). A utilização dos SOMs para comparação visual de multidões será posteriormente discutida no capítulo 4.

4 RESULTADOS EXPERIMENTAIS

Neste capítulo, são exibidos 3 casos de estudo onde as abordagens propostas, de subtração de *background* e segmentação automática da sombra, foram aplicadas. Também são apresentados 3 casos onde grupos de pessoas foram acompanhados, com a técnica de *tracking* descrita. Nesses 3 últimos experimentos foram gerados campos de vetores (conforme descrito anteriormente), a partir das trajetórias detectadas, que foram utilizados por um modelo de simulação (BRAUN et al., 2003; BRAUN; BODMAN; MUSSE, 2005), desenvolvido no âmbito do Laboratório CROMOS[1], objetivando simular um fluxo de pessoas com características semelhantes às das observadas no mundo real. Nesses 3 casos, os humanos virtuais simulados por Braun, utilizam os campos de velocidades gerados nesse trabalho no cálculo de seus movimentos (como velocidade e direção desejadas).

É importante salientar que, a exemplo do simulador utilizado, este trabalho também poderia ser integrado com outros sistemas similares para simulação de multidões, desde que os vetores velocidade representassem um estímulo para o movimento da população.

4.1 Subtração do fundo e segmentação da sombra

Nessa seção é analisada a performance dos algoritmos de subtração de fundo e subtração da sombra, para ambientes internos e externos, utilizando seqüências de vídeos em escala de cinza.

[1] *Virtual Human Simulation Lab* – `http://www.inf.unisinos.br/~cromoslab`

Na Figura 32 são exibidos 4 quadros de uma seqüência de vídeo externa, na qual uma parte da imagem é iluminada diretamente por luz solar, enquanto que em outras partes há apenas incidência de luz indireta. Como conseqüência, existe a produção de diferentes tipos de sombra (fracas e fortes). Na primeira linha, as imagens em escala de cinza originais são exibidas; na segunda e terceira linha, respectivamente, *pixels* do *foreground* antes e após a remoção da sombra, são exibidos. Objetos do *foreground* após a aplicação de operadores morfológicos são exibidos na quarta linha. Pode ser notado que a sombra foi efetivamente detectada e removida em ambos os tipos de sombra.

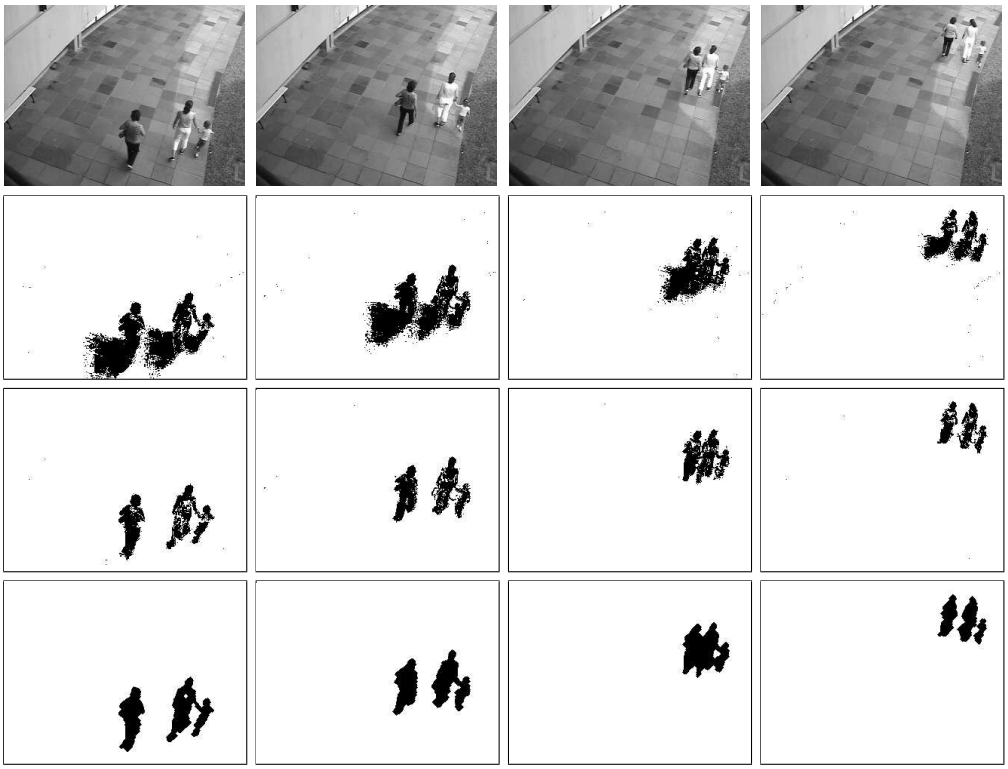
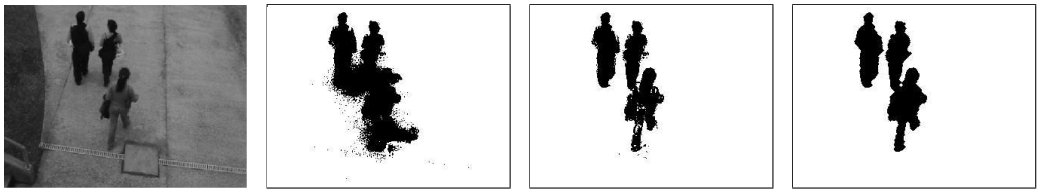


Figura 32: Primeira linha: quadros de uma seqüência de vídeo. Segunda linha: objetos do *foreground* detectados com a técnica de subtração de fundo descrita. Terceira linha: remoção da sombra. Quarta linha: resultado após aplicação de operadores morfológicos de pós-processamento (concatenação de fechamento e abertura).

Um outro caso de ambiente externo é ilustrado na Figura 33, correspondendo a uma seqüência de vídeo adquirida em um dia nublado, produzindo sombras fracas. A Figura 33 (a) exibe um determinado quadro da seqüência de vídeo, a Figura 33 (b) exibe os objetos do *foreground* detectados. Pode ser observado que é bastante difícil identificar

a pessoa na parte inferior da imagem. As Figuras 33 (c) e (d), respectivamente, ilustram o resultado da técnica de remoção de sombra proposta nesse trabalho, antes e após a aplicação de operadores morfológicos. Na quarta imagem, todas as três pessoas podem ser identificadas mais facilmente.



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

Figura 33: (a) Quadro de uma seqüência de vídeo em escala de cinza; (b) *pixels* do *foreground*; (c) sombra removida e (d) pós-processamento morfológico.

Na Figura 34 é exibido o resultado da aplicação da técnica proposta para a seqüência interna do *Hall Monitor*[2]. Como essa seqüência é colorida originalmente, nesse trabalho foi transformada para escala de cinza (usando o comando rgb2gray do MATLAB[3]). A sombra foi corretamente detectada e removida. Objetos válidos do *foreground* foram corretamente segmentados, apesar do fato que apenas 7 quadros tenham sido utilizados para a etapa de treinamento do *background*.

4.2 Simulando multidões de humanos virtuais de forma realista, auxiliado por visão computacional

Nessa seção são exibidos alguns resultados para demonstrar o potencial do modelo proposto no que tange a geração de dados para simulação de multidões humanas. Em particular, são apresentados 3 estudos de caso implementados em três cenários. Na seção 4.2.1 é descrito como é feita a integração dos campos de velocidades com o simulador. Na seção 4.2.2 são exibidos resultados experimentais e validações para o primeiro

[2]disponível para download em: `http://www.ics.forth.gr/cvrl/demos/NEMESIS/hall_monitor.mpg`, visitado em 09/01/2006.

[3]`http://www.mathworks.com/`

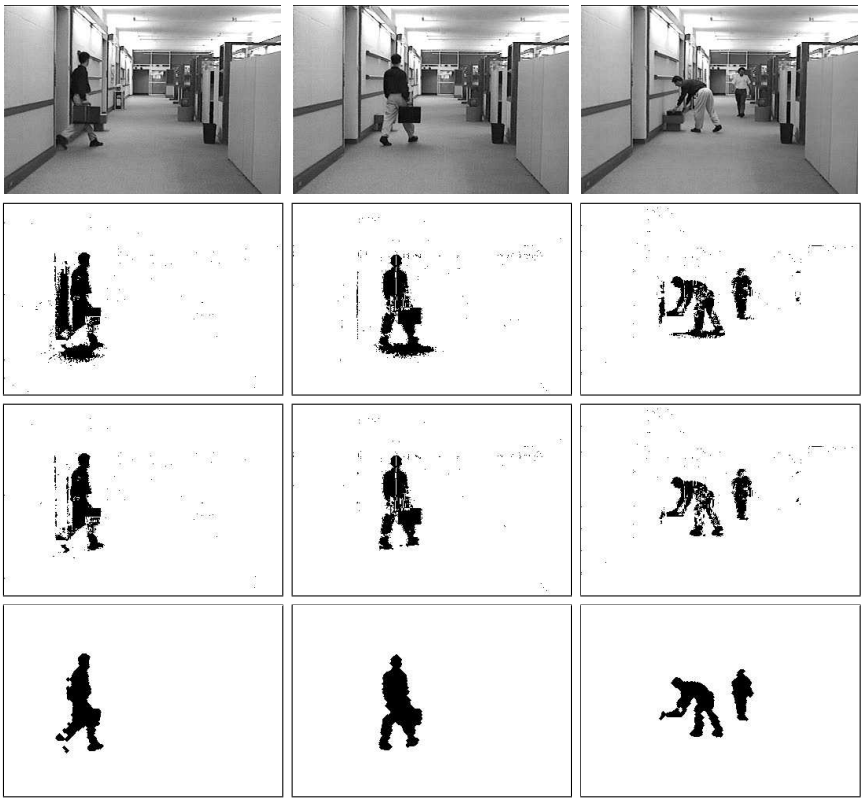


Figura 34: Primeira linha: quadros de uma seqüência de vídeo. Segunda linha: objetos do *foreground* detectados com a técnica de subtração de fundo descrita. Terceira linha: remoção da sombra. Quarta linha: resultado após aplicação de operadores morfológicos de pós-processamento

cenário (passagem de pedestres), na seção 4.2.3 são exibidos resultados experimentais e validações para o segundo cenário (bifurcação em T), e na seção 4.2.4 são exibidos resultados experimentais e validações para o terceiro cenário (calçadão).

Salienta-se que, nesses três experimentos, as trajetórias utilizadas para gerar os campos de velocidades foram pós-processados como descrito na seção 3.4.1, de maneira a servirem como dados de entrada para o simulador.

4.2.1 Integração dos dados com o simulador

Nessa seção é descrito como os dados gerados, com a utilização do modelo proposto, são integrados com o simulador de multidões desenvolvido no laboratório CROMOS. Salienta-se que este simulador utiliza como base o modelo de Helbing (HELBING; FARKAS; VICSEK, 2000), que foi extendido por Braun (BRAUN et al., 2003; BRAUN; BODMAN; MUSSE, 2005).

Como brevemente descrito na seção 2.1, Helbing et al (HELBING; FARKAS; VICSEK, 2000) utilizam um sistema de partículas para modelar o movimento da multidão. Nesse sistema, cada partícula i de massa m_i tem um vetor velocidade desejada $\mathbf{v}_i^g$ (*goal velocity*), tendendo a adaptar sua velocidade instantânea $\mathbf{v}_i$ dentro de um certo intervalo de tempo τ_i. Simultaneamente, as partículas tendem a manter uma distância dependente da velocidade em relação à outras partículas j e paredes w, controladas por forças de interação $\mathbf{f}_{ij}$ e $\mathbf{f}_{iw}$, respectivamente. A alteração da velocidade em um tempo t para cada partícula i é dado pela seguinte equação dinâmica:

$$m_i \frac{d\mathbf{v}_i}{dt} = m_i \frac{\mathbf{v}_i^g - \mathbf{v}_i(t)}{\tau_i} + \sum_{j \neq i} \mathbf{f}_{ij} + \sum_{w} \mathbf{f}_{iw} \tag{4.1}$$

Em simulações que não utilizam dados de visão computacional, a velocidade desejada de cada agente ($\mathbf{v}_i^g$) é descrita pelo ambiente, apontando para as portas de saída dos ambientes.

Para integrar os dados capturados de vídeo no fluxo de execução do simulador, cada agente i utiliza a informação de velocidade desejada vinda do campo de velocidades (conforme explanado na seção 3.4.1), na mesma posição do agente i ($\mathbf{v}_i^g = \mathbf{v}^{campo}$). Se uma seqüência de vídeo contém pessoas caminhando em diferentes velocidades, essa característica será mantida no campo de velocidades, e os agentes simulados irão tentar reproduzir aproximadamente os mesmos movimentos, como exibido nas seções 4.2.2, 4.2.3 e 4.2.4, onde são apresentados 3 estudos de caso.

4.2.2 Caso A: passagem de pedestres

Nesse experimento, um grupo de pessoas foi filmado em uma passagem de pedestres localizada no campus da UNISINOS. Um fragmento dessa passagem é exibido na Figura 35. As pessoas tiveram suas trajetórias capturadas e classificadas em um único tipo: pessoas que andam no sentido sul → norte. As trajetórias das pessoas são utilizadas para gerar um único campo de velocidades.

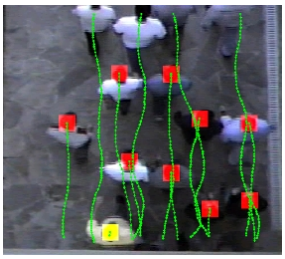


Figura 35: Trajetórias detectadas.

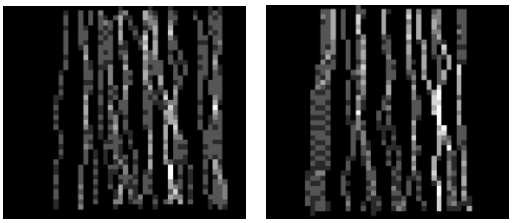
Após gerar o campo de velocidades para esse grupo, foi utilizado o simulador desenvolvido por Braun (BRAUN et al., 2003; BRAUN; BODMAN; MUSSE, 2005), com o objetivo de simular um grupo de humanos virtuais utilizando as informações capturadas do mundo. Na simulação, foi reproduzido o mesmo cenário e, um grupo com mesmo número de pessoas virtuais foi conduzido pela passagem virtual utilizando o campo de velocidades gerado (que informa a velocidade e direção desejada de cada agente, para cada ponto do espaço). Salienta-se que o modelo para tratamento de colisão entre as pessoas, e entre pessoas × obstáculos utilizado na simulação faz parte do modelo de Braun, e

ultrapassa o escopo desse trabalho.

Na Tabela 1 é feita uma comparação entre a seqüência filmada e o cenário simulado, onde pode ser observado a pequena diferença entre os tempos medidos. Na Figura 36 são apresentados resultados qualitativos de validação utilizando SOMs. Visualmente, os SOMs indicam que a ocupação espacial é similar em ambos ambientes, no real e no simulado.

	Tempo	Desvio Padrão
Seqüência real	2.21s	0.25s
Simulação	2.19s	0.22s

Tabela 1: Tempo médio e desvio padrão para percorrer todo o espaço para a seqüência filmada e para a simulação.



(a) (b)

Figura 36: (a) SOM das pessoas reais e (b) SOM dos humanos virtuais.

4.2.3 Caso B: bifurcação em T

Nesse experimento, um grupo de pessoas foi filmado em uma outra passagem de pedestres também na UNISINOS. Um fragmento dessa passagem é exibido na Figura 37. As pessoas tiveram suas trajetórias capturadas com a técnica de *tracking* descrita na seção 3.3 e classificadas, com a técnica de *clustering* utilizada, de forma automática em 4 tipos, como descrito na seção 3.4.1 (também ilustrado na Figura 37). Conseqüentemente, foram gerados 4 campos de velocidades (exibidos na Figura 30 –, 4 camadas a serem usadas pelo simulador), um para cada classe de trajetória. Salienta-se que o vetor de características originado por cada trajetória, utilizado para o *clustering* nesse experimento, é formado por 2 vetores deslocamento.

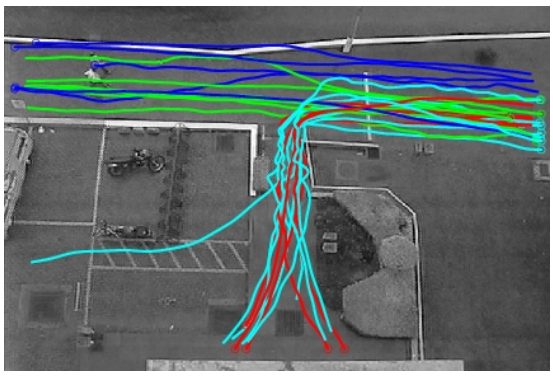


Figura 37: Classes geradas para o conjunto de trajetórias.

4.2.3.1 bifurcação em T: caso 1

Na simulação, foi reproduzido o mesmo cenário e um grupo, com aproximadamente mesmo número de pessoas (23) que na cena filmada, foi conduzido pela passagem virtual utilizando os campos de velocidades gerados (que informa a velocidade e direção desejada de cada agente, para cada ponto do espaço). Para essa simulação se definiu um pequeno grupo de pessoas na parte direita de imagem e um outro grupo de pessoas na parte esquerda da imagem. Tais pessoas virtuais devem seguir o campo de vetores gerado pelas pessoas reais que foram originadas no mesmo lugar – o grupo que segue o sentido direita → esquerda foi subdividido em 2 subgrupos, para que uma parte também siga o sentido superior → inferior. Finalmente, um terceiro grupo é originado na parte inferior da imagem, o qual deverá seguir o campo de vetores para as pessoas que partem daquela região. Deve-se salientar que, apesar dos agentes usarem campos de vetores diferentes, eles interagem entre si de acordo com o modelo de Braun (BRAUN et al., 2003; BRAUN; BODMAN; MUSSE, 2005), evitando interpenetração. A Figura 38 exibe o número de pessoas virtuais inicializadas para cada região.

Na Figura 39 é exibido o espaço utilizado para comparar resultados da simulação com dados capturados do vídeo, descrito na Tabela 2. Nessa Figura, as linhas A e C representam o espaço percorrido, medido em um corredor com uma pequena declividade, ou seja, as pessoas que caminham na direção esquerda → direita estão descendo. Na Tabela 2

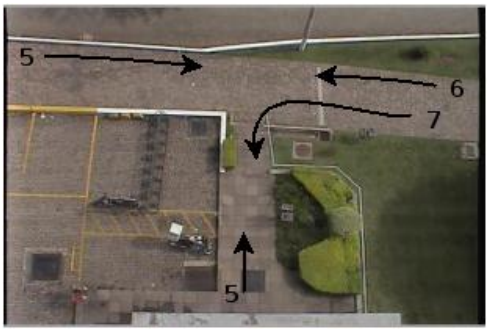


Figura 38: Número e local de origem aproximado das pessoas virtuais simuladas para o caso 1.

pode ser observado uma diminuição considerável de velocidade na região demarcada pela linha B, que representa uma escada.

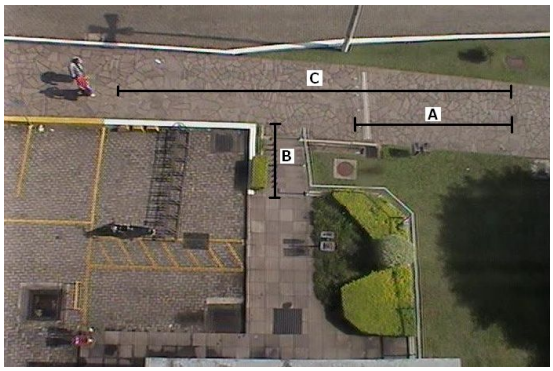


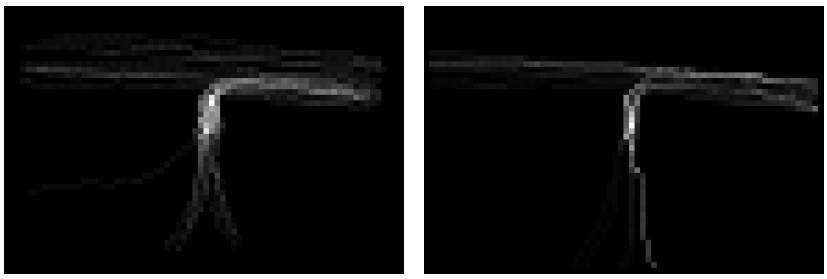
Figura 39: As linhas denominadas A, B e C são utilizadas para medir a velocidade média dos pedestres, na Tabela 2.

Na Tabela 2 é feita uma comparação entre a seqüência filmada e o cenário simulado. Com base nesses dados pode-se inferir que os humanos virtuais tiveram um comportamento bastante semelhante aos dos humanos reais. Deve-se salientar que, apesar das velocidades serem bastante semelhantes, algumas das pessoas virtuais demoraram um tempo maior para percorrer a região C. Isso ocorre por causa do modelo utilizado para tratamento de colisão entre as pessoas, que pode gerar um pequeno atraso dependendo de como elas colidem (ou evitam a colisão). No caso do modelo de Braun (BRAUN et al., 2003; BRAUN; BODMAN; MUSSE, 2005), o tratamento da colisão é implementado usando forças de repulsão radiais, conseqüentemente os agentes não são capazes de prever a colisão, e evitá-la alterando suas trajetórias, previamente ao momento da colisão.

Na Figura 40 são apresentados resultados qualitativos de validação utilizando SOMs. Visualmente, os SOMs indicam que a ocupação espacial é similar em ambos ambientes, no real e no simulado.

Região	Direção	Vídeo (vel.)		Simulação (vel.)	
		média	desvio	média	desvio
A	→	0.95998m/s	0.17493m/s	0.95582/s	0.26204m/s
	←	1.0024m/s	0.19033m/s	0.94205m/s	0.24599m/s
B	↓	0.40749m/s	0.23703m/s	0.43244m/s	0.27446m/s
	↑	0.41951m/s	0.18849m/s	0.51103m/s	0.29232m/s
C	→	1.028m/s	0.20061m/s	0.98517m/s	0.2701m/s
	←	1.0625m/s	0.20418m/s	0.99671m/s	0.19928m/s

Tabela 2: Métricas quantitativas para validação do caso 1, para o cenário "bifurcação em T".



(a) (b)

Figura 40: (a) SOM das pessoas reais e (b) SOM dos humanos virtuais do caso 1, para o cenário "bifurcação em T".

4.2.3.2 bifurcação em T: caso 2

Também foi analisado para este cenário a influência de se aumentar o número de pessoas virtuais, usando os campos de velocidades obtidos com um pequeno grupo de trajetórias (as mesmas utilizadas anteriormente). Tal experimento pode ser útil para prever o fluxo das pessoas em espaços públicos durante ocasiões especiais (por exemplo, em um *shopping center* nas vésperas do Natal). Extrapolou-se o número de pessoas virtuais deste cenário de 23 (como na seção 4.2.3.1) para 70. Poderia ser interessante realizar a mesma análise quantitativa (velocidade média e desvio padrão) e qualitativa (SOMs) como um procedimento de validação para esse experimento com o número de

pessoas simuladas extrapoladas, porém não possuímos vídeos reais com esse número de pessoas. Entretanto, é importante observar, na Tabela 3 e na Figura 42, que toda a multidão nesse cenário preservou as mesmas características das pessoas reais (informação espacial e decaimento da velocidade em função da escada). A Figura 41 exibe o número de pessoas virtuais inicializadas para cada região.

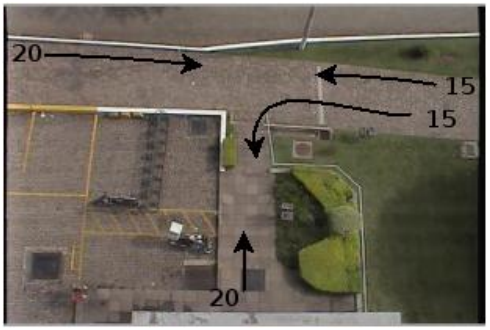


Figura 41: Número e local de origem aproximado das pessoas virtuais simuladas para o caso 2.

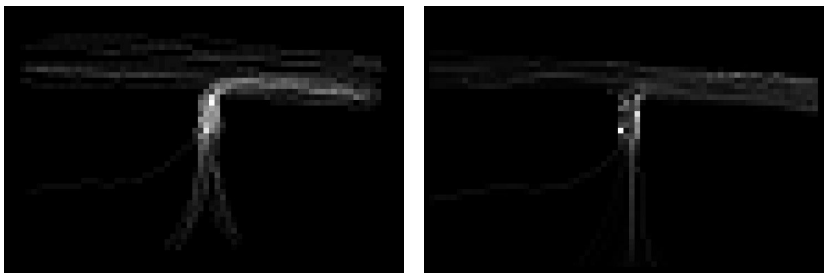
Região	Direção	Vídeo (vel.)		Simulação (vel.)	
		média	desvio	média	desvio
A	→	0.95998m/s	0.17493m/s	0.86385/s	0.31848m/s
	←	1.0024m/s	0.19033m/s	0.86917m/s	0.2945m/s
B	↓	0.40749m/s	0.23703m/s	0.41637m/s	0.34696m/s
	↑	0.41951m/s	0.18849m/s	0.42659m/s	0.28291m/s
C	→	1.028m/s	0.20061m/s	0.89804m/s	0.33736m/s
	←	1.0625m/s	0.20418m/s	0.94548m/s	0.31595m/s

Tabela 3: Métricas quantitativas para validação do caso 2, para o cenário "bifurcação em T".

É importante salientar que as velocidades médias das pessoas virtuais diminuíram com relação ao experimento simulado com 23 pessoas, demonstrando o impacto do aumento de pessoas simuladas.

4.2.3.3 bifurcação em T: caso 3

Em outro resultado experimental de simulação, para o mesmo cenário, as pessoas virtuais foram originadas apenas na parte esquerda da imagem e na parte inferior, totalizando 200 pessoas virtuais (100 para cada região). Dessa forma, elas deverão seguir os



(a) (b)

Figura 42: (a) SOM das pessoas reais e (b) SOM dos humanos virtuais do caso 2, para o cenário "bifurcação em T".

campos de velocidades das pessoas reais que partiram das mesmas regiões, sendo que não deverão evitar colisão frontal, já que todas as pessoas se locomovem basicamente para um único ponto final (parte direita da imagem). A Figura 43 exibe o número de pessoas virtuais inicializadas para cada região.

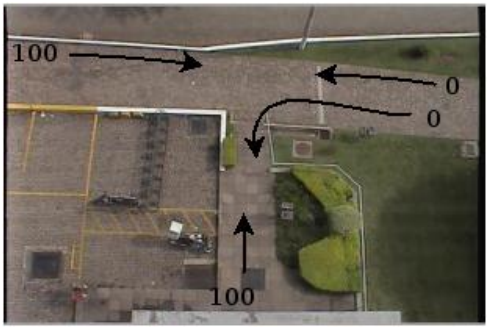


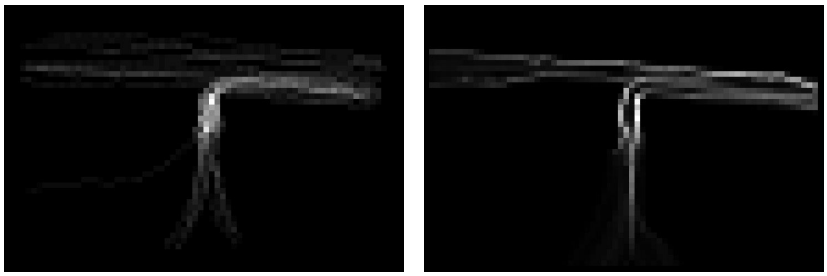
Figura 43: Número e local de origem aproximado das pessoas virtuais simuladas para o caso 3.

Na Tabela 4 é feita uma comparação entre a seqüência filmada e o cenário simulado. Com base nesses dados pode-se inferir que os humanos virtuais tiveram um comportamento mais semelhante aos dos humanos reais do que nos casos anteriores (pois praticamente não evitaram colisão frontal).

Na Figura 44 são apresentados resultados qualitativos de validação utilizando SOMs. Visualmente, os SOMs indicam que a ocupação espacial é similar em ambos ambientes, no real e no simulado.

Região	Direção	Vídeo (vel.)		Simulação (vel.)	
		média	desvio	média	desvio
A	→	0.95998m/s	0.17493m/s	0.93663/s	0.18841m/s
B	↑	0.41951m/s	0.18849m/s	0.51198m/s	0.2502m/s
C	→	1.028m/s	0.20061m/s	0.9798m/s	0.20787m/s

Tabela 4: Métricas quantitativas para validação do caso 3, para o cenário "bifurcação em T".



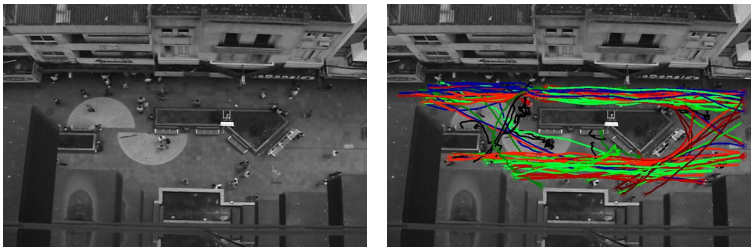
(a) (b)

Figura 44: (a) SOM das pessoas reais e (b) SOM dos humanos virtuais do caso 3, para o cenário "bifurcação em T".

4.2.4 Caso C: calçadão

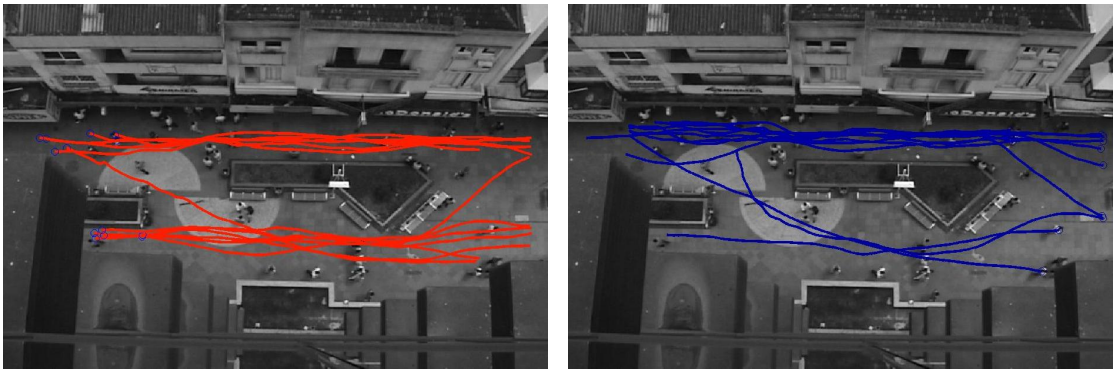
Nesse experimento, um grupo de pessoas foi filmado em um calçadão, localizado na cidade de Santa Maria - RS. Um fragmento dessa passagem é exibido na Figura 45(a). As pessoas tiveram suas trajetórias capturadas com a técnica de *tracking* descrita na seção 3.3 e classificadas de forma automática em 8 tipos, como ilustrado com diferentes cores na Figura 45(b). Dentre as 8 classes geradas automaticamente, 6 delas contêm trajetórias pequenas (pessoas que apenas atravessam a rua, entrando em lojas), e 2 delas contêm trajetórias mais longas (e com um número maior de pessoas acompanhadas). Essas duas últimas classes (Figura 46) representam melhor o fluxo global de movimentação de pessoas no ambiente, sendo selecionadas para alimentar o simulador de multidões. Salienta-se que o vetor de características originado por cada trajetória, utilizado para o *clustering* nesse experimento, é formado por 4 vetores deslocamento, objetivando-se capturar mais adequadamente as características das curvas. As trajetórias capturadas possuem, em geral, um deslocamento bastante grande, e utilizando somente 2 vetores de características fez com que diversas trajetórias, visualmente distintas, fossem postas em uma mesma

classe.



(a) (b)

Figura 45: (a) campo de visão da câmera e (b) classes de trajetórias geradas.



(a) (b)

Figura 46: (a) classe esquerda/direita e (b) classe direita/esquerda.

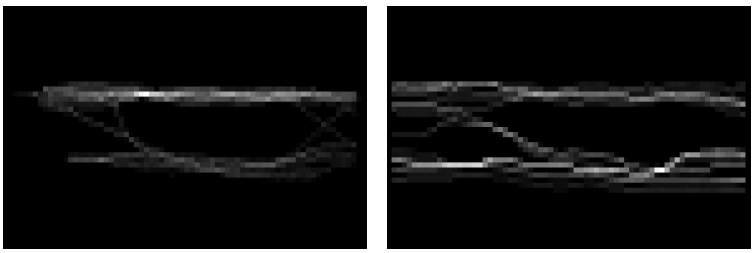
Na simulação, foi reproduzido o mesmo cenário e um grupo, com aproximadamente mesmo número de pessoas (30) que na cena filmada, foi conduzido pela passagem virtual utilizando os campos de velocidades gerados (que informa a velocidade e direção desejada de cada agente, para cada ponto do espaço). Para essa simulação se definiu um pequeno grupo de pessoas na parte direita da imagem (17) e um outro grupo de pessoas na parte esquerda da imagem (13). Tais pessoas virtuais devem seguir o campo de vetores gerado pelas pessoas reais que foram originadas no mesmo lugar. Deve-se salientar que, apesar dos agentes usarem campos de vetores diferentes, eles interagem entre si de acordo com o modelo de Braun (BRAUN et al., 2003; BRAUN; BODMAN; MUSSE, 2005).

Na Tabela 5 é feita uma comparação entre a seqüência filmada e o cenário simulado, onde pode ser observado a pequena diferença entre as velocidades medidas. Na Figura 47

são apresentados resultados qualitativos de validação utilizando SOMs. Visualmente, os SOMs indicam que a ocupação espacial é similar em ambos ambientes, no real e no simulado (desconsiderando que no ambiente real as pessoas andaram mais pela parte superior da imagem do que na inferior - isso poderia ser solucionado se na simulação as pessoas fossem originadas mais na parte superior da imagem, ao invés de serem originadas de forma aleatória) .

	Direção	Velocidade	Desvio Padrão
Seqüência real	→	1.3321m/s	0.29475
Simulação	→	1.3273m/s	0.28023
Seqüência real	←	1.1253m/s	0.32684
Simulação	←	1.1483m/s	0.27464

Tabela 5: Velocidade média e desvio padrão para percorrer todo o espaço para a seqüência filmada e para a simulação, para o cenário "calçadão".



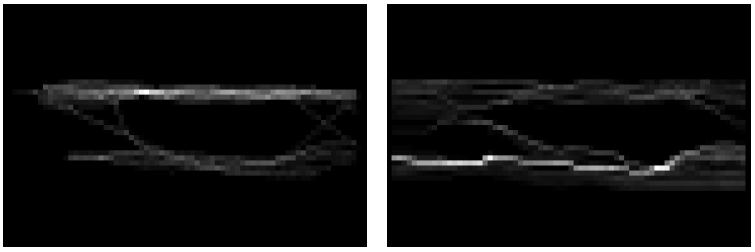
(a) (b)

Figura 47: (a) SOM das pessoas reais e (b) SOM dos humanos virtuais, para o cenário "calçadão".

Também foi analisado para este cenário a influência de se aumentar o número de pessoas virtuais, usando os campos de velocidades obtidos com um pequeno grupo de trajetórias. Nós extrapolamos o número de pessoas virtuais deste cenário de 30 para 100, sendo 60 pessoas virtuais inicializadas na parte direita da imagem e 40 na parte esquerda. Poderia ser interessante realizar a mesma análise quantitativa (velocidade média e desvio padrão) e qualitativa (SOMs) como um procedimento de validação para esse experimento com o número de pessoas virtuais extrapoladas, porém não possuímos vídeos reais com esse número de pessoas. Entretanto, é importante observar, na Tabela 6 e na Figura 48, que toda a multidão nesse cenário preservou as mesmas características das pessoas reais.

	Direção	Velocidade	Desvio Padrão
Seqüência real	→	1.3321m/s	0.29475
Simulação	→	1.2893m/s	0.31864
Seqüência real	←	1.1253m/s	0.32684
Simulação	←	1.1778m/s	0.31377

Tabela 6: Velocidade média e desvio padrão para percorrer todo o espaço para a seqüência filmada e para a simulação, para o cenário "calçadão" extrapolado.



(a) (b)

Figura 48: (a) SOM das pessoas reais e (b) SOM dos humanos virtuais para o cenário "calçadão" extrapolado.

5 OUTRAS APLICAÇÕES - DETECÇÃO DE EVENTOS NA MULTIDÃO

O modelo proposto neste trabalho foi aplicado em um protótipo para detectar eventos na multidão. Este capítulo visa discutir alguns resultados desta aplicação bem como explorar outras possibilidades de utilização do modelo.

A análise do movimento das pessoas em seqüências de vídeo pode ser muito útil em diversas aplicações. Pode ser interessante analisar o movimento das pessoas para utilização em projetos arquitetônicos de espaços públicos, objetivando aumentar o nível de conforto das pessoas. Por exemplo, tal análise poderia ser utilizada para decidir onde posicionar uma peça de arte em um museu de modo que pudesse ser visualizada confortavelmente por um grande grupo de pessoas. Uma outra possibilidade pode estar relacionada à engenharia de segurança, a qual poderia utilizar tal análise para prever comportamentos suspeitos em uma multidão, desde que eventos pré-determinados possam ser entendidos e possivelmente detectados.

Nessa seção é descrita uma possível aplicação para analisar o comportamento das pessoas, considerando aspectos individuais e de grupos, com a utilização de Diagramas de Voronoi (BERG et al., 1998; GUIBAS; STOLFI, 1985). Para isso, a posição de cada pessoa é utilizada como um ponto gerador de uma célula de Voronoi, e a evolução temporal dos polígonos de Voronoi é analisada. A posição das pessoas para cada instante de tempo pode ser capturada a partir de seqüências de vídeos reais, com a utilização da técnica de

tracking descrita na seção 3.3, ou obtidas diretamente do simulador de multidões (o que pode ser útil quando se deseja reproduzir, de forma controlada, situações específicas).

Essa aplicação faz uso de diversos conceitos relacionados ao espaço pessoal, proxemics e relacionamento espacial entre as pessoas, que são relatados na literatura por alguns pesquisadores de áreas como psicologia e sociologia. O termo proxemics foi proposto por Edward Hall (HALL, 1973), visando descrever a utilização social do espaço (em particular, espaço pessoal). Espaço pessoal é relatado à área, de fronteiras invisíveis, que rodeia o corpo de cada pessoa. Essa área estabelece uma zona de conforto durante a comunicação inter-pessoal, e pode desaparecer em espaços ou situações específicas (como por exemplo, elevadores ou multidões muito densas). Edward Hall estabelece quatro tipos de distâncias entre pessoas, exibidas na Tabela 7 (intervalo de distâncias em centímetro), os quais são utilizados como base nessa aplicação, para estabelecer se uma pessoa está andando confortavelmente ou não.

Distância (cm)	Próxima	Afastada
Íntima	< 15	15 - 45
Pessoal	50 - 80	80 - 120
Social	120 - 210	210 - 350
Pública	350 - 750	> 750

Tabela 7: Intervalos de distâncias entre pessoas estabelecidos por Hall

Neste trabalho, as distâncias estabelecidas por Hall foram simplificadas e se estabeleceu a classificação exibida na Tabela 8. Basicamente eliminou-se os atributos “próximo” e “afastado”.

Distância (cm)	intervalo
Íntima	≤ 45
Pessoal	$45 < d \leq 120$
Social	$120 < d \leq 350$
Pública	> 350

Tabela 8: Intervalos de distâncias propostos nesse trabalho

Conforme Robert Sommer (SOMMER, 1973), há considerável semelhança entre *espaço pessoal* e *distância individual*, ou espaçamento característico dos membros da

espécie. A distância individual existe apenas quando dois ou mais membros da espécie estão presentes, e sofre muita influência da densidade da população e do comportamento territorial. A distância pessoal e o espaço pessoal interagem em sua influência sobre a distribuição de pessoas. A violação da distância individual é uma violação das expectativas da sociedade; a invasão do espaço pessoal é uma intrusão nas fronteiras do eu da pessoa.

A Figura 49 ilustra a utilização do Diagrama de Voronoi como uma maneira de medir o espaço pessoal de cada pessoa. O uso do Diagrama de Voronoi também possibilita detectar quais pessoas poderiam estar, de alguma maneira, interferindo no espaço pessoal de outra.

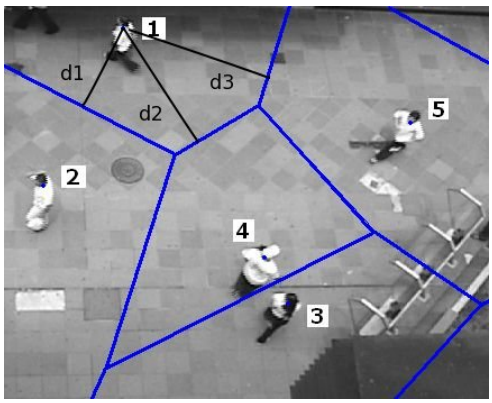


Figura 49: Distância entre pessoas. Nesse caso, as pessoas 1, 2, 3 e 5 podem estar influenciando no caminhar da pessoa 4.

Nesse trabalho definiu-se que o espaço pessoal de cada indivíduo é a área do polígono de voronoi correspondente, já que cada ponto dentro do polígono do voronoi está mais próximo do ponto que o gerou (a própria pessoa, ou *site*) do que qualquer outro ponto. O polígono de voronoi também é utilizado para calcular a distância de uma pessoa à seus vizinhos, como ilustrado na Figura 49, com os valores $d1$, $d2$ e $d3$ (que equivalem a metade das distâncias entre as pessoas vizinhas à pessoa 1). De fato, um ponto localizado nas bordas do polígono de voronoi está equidistante de dois *sites* vizinhos (duas pessoas vizinhas). Assim, a distância ortogonal de um *site*, de um polígono de voronoi, à uma de suas bordas, representa a metade da distância ao seu respectivo "vizinho de borda". Com essas distâncias, pode-se classificar o tipo de interação entre duas pessoas vizinhas

de acordo com a Tabela 8.

Como já conhecemos a trajetória de cada pessoa, podemos computar o Diagrama de Voronoi dinamicamente para cada quadro da seqüência de imagens, e extrair o espaço pessoal e distância dos vizinhos para cada indivíduo, como uma função do tempo. A análise temporal de tal informação pode ser usada para detectar alguns eventos específicos. Por exemplo, se o espaço pessoal de uma pessoa diminui em função do tempo, pode ser provável que ela esteja entrando em um lugar bastante ocupado (alta densidade). Eventos suspeitos também podem ser detectados se o espaço pessoal de um indivíduo é invadido por outra pessoa, em relação à distância pessoal, não sendo coerente ao relacionamento entre as duas pessoas (por exemplo, se uma pessoa desconhecida permanece à uma distância íntima de outra, mesmo que tenha espaço considerável para se manter à uma distância pública, uma possível situação de roubo poderia ser detectada). Salienta-se que esse trabalho não está focado especialmente nesse tipo de análise, mas acredita-se que tal análise pode representar um futuro aperfeiçoamento.

De acordo com Sommer (SOMMER, 1973), o espaço pessoal está relacionado com o nível de conforto das pessoas. Mesmo que seja intuitivo pensar que uma pessoa sinta-se confortável se o espaço ao seu redor for suficientemente grande, acreditamos que o espaço pessoal não seja a melhor métrica para estimar o nível de conforto. De fato, o conceito psicológico de espaço pessoal é estático, significando que é considerada a região ao redor do indivíduo, desconsiderando-se sua direção. Uma pessoa pode ter um espaço consideravelmente grande atrás de si, mas pode não se sentir confortável quando há outra pessoa ou um obstáculo à sua frente. O espaço pessoal percebido (*Perceived Personal Space*, ou *PPS*) proposto neste trabalho, provê uma métrica mais acurada para estimar o nível de conforto, pois é levado em consideração o campo de visão do indivíduo e sua direção. O campo de visão de cada indivíduo é modelado como um setor circular, com um ângulo α, simétrico em relação ao seu vetor velocidade. O PPS é então definido como sendo a área da região formada pela intersecção do campo de visão com o polígono correspondente, formado pelo Diagrama de Voronoi, como exibido na Figura 50 (em nossos

experimentos utilizou-se $\alpha = 120°$).

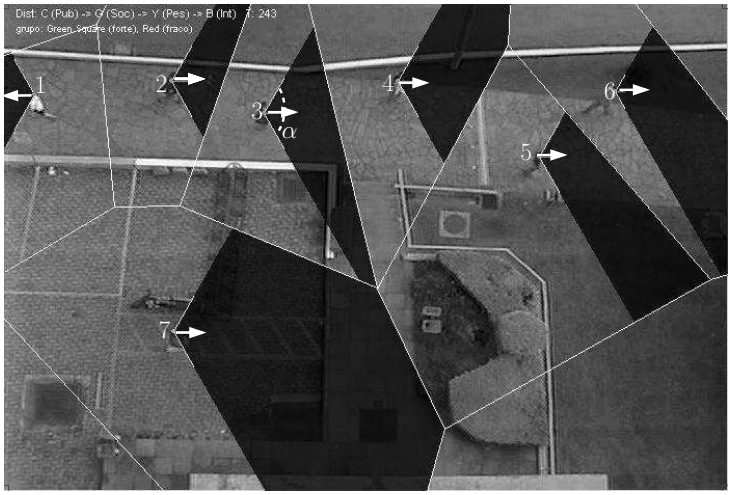


Figura 50: Diagrama de Voronoi, campos de visão e região relacionada ao espaço pessoal percebido, PPS.

Acredita-se que uma pessoa sinta-se confortável se há espaço suficiente no seu campo de visão em relação à uma determinada distância R_c. São utilizados os valores de distâncias da Tabela 8 para determinar diferentes níveis de conforto para R_c. Por exemplo, utilizando $R_c = 3.5$ metros, se provê um nível de conforto público, enquanto que ao utilizar-se $R_c = 0.45$ metros, estará se propondo um nível de conforto íntimo. Matematicamente dizemos que um indivíduo sente-se confortável em um quadro t, da seqüência de imagens, se:

$$PPS(t) \geq \frac{\alpha}{2} R_c^2, \tag{5.1}$$

onde $\frac{\alpha}{2} R_c^2$ é a área do setor circular com um ângulo α (em radianos) e raio R_c, e o $PPS(t)$ é o espaço pessoal percebido, estimado no quadro t. De acordo com a métrica proposta para conforto, por exemplo, o indivíduo 7 está mais confortável que o indivíduo 2, na Figura 50.

Em particular, também podemos utilizar o nível de conforto íntimo para identificar formação de grupos, como descrito a seguir. Para detectar a formação de grupos, o sistema monitora o nível de distância de cada pessoa para seus vizinhos, no decorrer do

tempo. Se duas ou mais pessoas mantiverem distâncias pequenas entre elas, durante um período de tempo (chamado de período de agrupamento, e denotado por T_g, medido em quadros), consideramos que elas formam um grupo. Na prática, até mesmo um grupo fortemente conectado (por exemplo, um casal de namorados) pode se separar durante alguns quadros, quando evitando obstáculos ou outras pessoas, por exemplo. Para tratar esse tipo de situação consideramos que dois indivíduos formam um grupo se eles permanecem à uma distância íntima (um do outro), por pelo menos uma fração p do período T_g, onde $0 \leq p \leq 1$.

Formalmente, considere dois indivíduos, $I_i(t)$ e $I_j(t)$, no quadro t, onde a função de agrupamento (g) é definida:

$$g(i,j,t) = \begin{cases} 1 & \text{se } d(I_i(t), I_j(t)) \leq D_{\text{íntima}} \\ 0 & \text{caso contrário} \end{cases}, \tag{5.2}$$

onde $d(I_i, I_j)$ representa a distância entre os agentes $I_i(t)$ e $I_j(t)$ no quadro t, e $D_{\text{íntima}} = 0.45$ metros é a distância para relacionamento íntimo, como definido na Tabela 8. Dessa forma, $I_i(t)$ e $I_j(t)$ são considerados um grupo no quadro t se:

$$\sum_{k=t-(T_g-1)}^{t} g(i,j,k) \geq pT_g \tag{5.3}$$

Nossos resultados experimentais indicam que 5 segundos são o suficiente para formação de grupo e $p = 0.8$ (para seqüências de vídeo adquiridas a uma taxa de 10 FPS, dessa forma, usualmente $T_g = 10 \times 5 = 50$).

Também definimos que a propriedade de formação de grupos seja associativa, por exemplo, se I_i e I_j estão agrupados e I_j e I_k também estão agrupados, conseqüentemente, I_i e I_k também estão agrupados. Dois exemplos de formação de grupos são ilustrados na Figura 51. Essa Figura ilustra um quadro de uma seqüência do vídeo, onde as pessoas pertencentes à dois grupos distintos aparecem destacadas.

Após detectar a formação de grupos, pode ser desejado identificar o grupo como

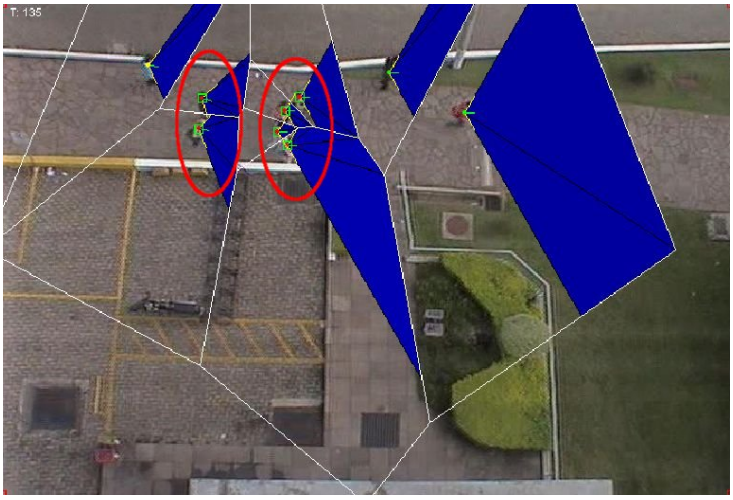


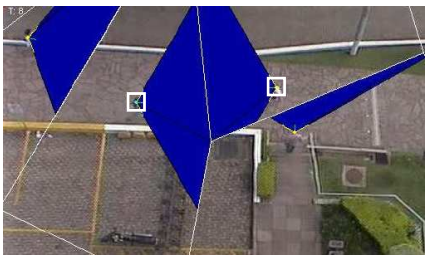
Figura 51: Dois grupos detectados.

voluntário ou involuntário. Acreditamos que há duas causas principais para formação de grupos: primeira, pessoas formam um grupo porque querem (por exemplo, dois amigos) e segunda, porque são forçadas, devido à restrições no espaço (por exemplo, uma multidão evacuando um estádio lotado).

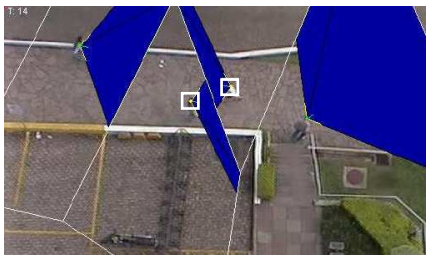
Para caracterizar um grupo como voluntário ou involuntário, é analisado o PPS de todos os indivíduos do grupo. É esperado que em grupos voluntários, a maioria das pessoas possuam um PPS elevado durante o período de agrupamento (T_g). Entretanto, alguns indivíduos podem caminhar atrás de outros (por um determinado tempo), conseqüentemente, tornando seus PPSs pequenos. Assim, para detectar formação de grupos voluntários, é verificado se pelo menos a metade das pessoas do grupo possuem um PPS relativamente grande. Mais especificamente, um grupo de N indivíduos é classificado como voluntário se, pelo menos $N/2$ elementos, estão confortáveis (em relação à $D_{\text{íntima}}$) durante uma fração p do tempo T_g. Caso contrário, o grupo é caracterizado como involuntário.

A Figura 52 exibe 3 quadros de uma seqüência de vídeo, onde indivíduos caminham livremente em uma situação não densa. Nessa Figura, o PPS de cada indivíduo é exibido em tonalidade escura. Em particular, pode ser notado que o PPS de dois indivíduos (destacados com quadrados brancos), que caminham em sentidos opostos, decresce em

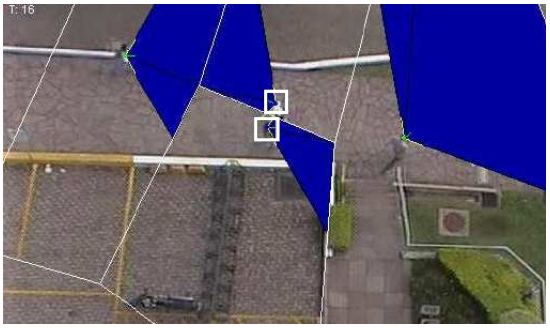
função do tempo. A distância entre eles também decresce, de acordo com a classificação estabelecida por Hall (HALL, 1973): na Figura 52 (a), os indivíduos estão à uma distância social um do outro; na Figura 52 (b), estão à uma distância pessoal um do outro e na Figura 52 (c), estão à uma distância íntima um do outro. Entretanto, não formam grupo pois permanecem um período de tempo muito curto em distância íntima um do outro.



(a) Quadro 8



(b) Quadro 14



(c) Quadro 16

Figura 52: Exemplos de PPS e distâncias individuais.

Em um experimento simulado, foi criado um cenário com 120 metros de comprimento, contendo um corredor (com aproximadamente 20 metros de comprimento) estreito no meio do cenário. A razão para incluir também dados gerados por um simulador está relacionada com a flexibilidade de se gerar situações controladas com uma variedade de parâmetros e eventos conhecidos, como geração de grupos, tamanho da população, velocidade das pessoas, restrições no espaço, localizações com pontos de afunilamento, etc. Tal ambiente foi povoado com 300 agentes, utilizando-se o simulador desenvolvido por Braun (BRAUN et al., 2003; BRAUN; BODMAN; MUSSE, 2005), onde o objetivo era movimentar as pessoas da esquerda para a direita. Algumas pessoas fazem parte de uma mesma

família, significando que os agentes se conhecem e pretendem permanecer juntos durante toda a simulação. A Figura 53 ilustra um quadro dessa simulação, onde indivíduos que não possuem um nível pessoal de conforto (por exemplo, agentes cuja equação (5.1) não é satisfeita para $R_c = 1.20$ metros) foram agrupados por causa da restrição do ambiente. De fato, nas regiões A e C, poucos agentes apresentaram estar em desconforto, e o oposto pode ser percebido na região B, como esperado, já que os agentes estão mais agrupados no corredor (alta densidade). Os humanos virtuais considerados desconfortáveis são marcados com pontos escuros na Figura 53.

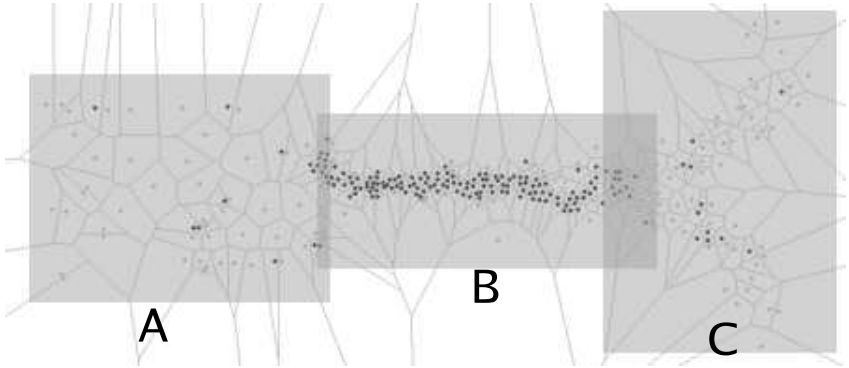


Figura 53: Humanos virtuais confortáveis e desconfortáveis.

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Esse trabalho apresentou um modelo para extrair informações do mundo real, capturadas com a utilização de técnicas de visão computacional, com o fim de simular e validar comportamentos de multidões de humanos virtuais. Conforme os resultados apresentados até o momento, pode-se afirmar que a utilização desse modelo apresentou resultados aceitáveis, conforme análise feita no capítulo 4.

As pessoas tiveram suas trajetórias capturadas de forma automática, através de algoritmos de visão computacional, e foram reproduzidos nas simulações comportamentos bastante semelhantes aos capturados da realidade. Invariavelmente, outros modelos, comportamentais ou baseados em usuários, não o fizessem de maneira tão eficiente (dificilmente um modelo comportamental conseguiria exibir tais resultados, entretanto, um modelo baseado em usuário poderia reproduzi-los, demandando trabalho exaustivo por parte do usuário).

Também não foi encontrado, na literatura revisada, nenhum trabalho que utilize dados extraídos por visão computacional que sejam utilizados para simular ou validar comportamentos de multidões de humanos virtuais, fazendo com que este trabalho também possua características inovadoras. Entretanto, salienta-se que os humanos virtuais devem possuir algum estímulo para movimentarem-se até a região que possue os campos de velocidades (pontos de interesse e velocidade média, por exemplo), fazendo com que o modelo proposto neste trabalho venha a acrescentar informações em simulação de multidões, sem abrir mão de métodos tradicionais de animação e simulação.

O resultado final deste trabalho, após integração com o simulador, também é bastante dependente do modelo utilizado para tratamento de colisão entre as pessoas virtuais, pois sem esse tratamento, as pessoas virtuais poderiam reproduzir as velocidades, direções e sentidos de forma bastante similar às do mundo real, porém iriam passar umas dentro das outras (gerando um comportamento irreal). Dessa forma, também seria interessante analisar o modelo para tratamento de colisão entre pessoas reais (ou outros modelos encontrados na literatura), de forma a calibrar o simulador, tornando o movimento dos humanos virtuais o mais realista possível.

Foram feitas contribuições em um modelo de subtração de *background* bastante referenciado na literatura, tornando possível a geração do modelo de fundo da cena em um tempo relativamente menor, bem como detectando de forma mais precisa os objetos que se movimentam na cena. Também foi apresentada uma nova técnica para detecção e remoção de sombra em seqüências de imagens em escala de cinza, para utilização em ambientes internos e externos, capaz de detectar sombras fracas e fortes.

O modelo proposto neste trabalho também foi aplicado em um protótipo para monitorar e detectar eventos na multidão, podendo ser utilizado para analisar o comportamento das pessoas quando andam sozinhas ou em grupos. Tal análise poderia ser utilizada para identificar de forma automática grupos de pessoas e analisar o comportamento do grupo como um todo, objetivando entender melhor o comportamento das pessoas quando não estão sozinhas e tentar reproduzir alguns padrões detectados, em um ambiente simulado. Tal aplicação também poderia ser utilizada para detecção de eventos suspeitos (como por exemplo, identificar uma aproximação rápida entre duas pessoas e prever uma situação de roubo). Como aperfeiçoamento dessa aplicação, objetivamos investigar o tempo para formação de grupos em outras seqüências de vídeo, assim como validar nossas regras de agrupamento. Acredita-se que a caracterização do grupo também pode ser utilizada para detectar automaticamente a localização de obstáculos e pontos de afunilamento no ambiente. Também se pretende aplicar a técnica a uma variedade de seqüências de vídeos, objetivando adaptar as distâncias de Hall para outras culturas.

Contudo, alguns pontos do modelo ainda estão em aberto, e serão atividades a serem desenvolvidas em estudos futuros. Para tornar a detecção dos objetos mais robusta, pretende-se utilizar informação temporal (quadros passados, por exemplo) na etapa de subtração de *background* e remoção da sombra. Também se pretende analisar quantitativamente o impacto da utilização do cálculo do NCC na etapa de detecção de sombra. Por fim, apresentar medidas de performance do sistema de visão computacional, como por exemplo, FPS e resolução máxima permitida em tempo real.

O presente trabalho é base de uma linha de pesquisa do Projeto CSHuV, desenvolvido em cooperação com HP Brasil.

REFERÊNCIAS

BARRON, J. L. et al. Performance of optical flow techniques. *Int. J. Computer Vision*, v. 12, n. 1, p. 43–77, 1994. ISSN 0920-5691.

BERG, M. de et al. *Computational Geometry*. 2nd. ed. Verlag Berlin Heidelberg New York: Springer, 1998.

BRAUN, A. *Modelagem e Simulação de Multidões Humanas em Situações de Emergência.* Dissertação (Mestrado) — Universidade do Vale dos Sinos, 2004.

BRAUN, A.; BODMAN, B. J.; MUSSE, S. R. Simulating virtual crowds in emergency situations. In: *Proceedings of ACM Symposium on Virtual Reality Software and Technology (VRST)*. [S.l.: s.n.], 2005. p. 244–252.

BRAUN, A. et al. Modeling individual behaviors in crowd simulation. In: *CASA*. [S.l.]: IEEE Computer Society, 2003. p. 143–148. ISBN 0-7695-1934-2.

BROGAN, D. C.; JOHNSON, N. L. Realistic humanwalking paths. *Proceedings of Computer animation and social Agents, IEEE Computer Society*, p. 94–101, 2003.

CHENNEY, S. Flow tiles. *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, p. 233–242, 2004.

CUCCHIARA, R. et al. Detecting moving objects, ghosts, and shadow in video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 25, n. 10, p. 1337–1341, October 2003.

ELGAMMAL, A. et al. Background and foreground modeling using nonparametric kernel density estimation for visual surveilance. *Proceedings of the IEEE*, v. 90, n. 7, p. 1151–1162, July 2002.

FIGUEIREDO, M. A. T.; JAIN, A. K. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 24, n. 3, p. 381–396, 2002.

FRUIN, J. J. *Pedestrian. Planing and Design - Revised Edition.* 2nd. ed. Alabama, USA: Elevator World, Inc., 1971.

GONZALEZ, R. C.; WINTZ, P. *Digital Image Processing.* 2nd. ed. Reading, MA: Addison-Wesley, 1987.

GREST, D.; FRAHM, J.-M.; KOCH, R. A color similarity measure for robust shadow removal in real time. In: ERTL, T. (Ed.). *VMV*. [S.l.]: Aka GmbH, 2003. p. 253–260. ISBN 3-89838-048-3.

GUIBAS, L.; STOLFI, J. Primitives for the manipulation of general subdivisions and the computation of voronoi. *ACM Trans. Graph.*, ACM Press, New York, NY, USA, v. 4, n. 2, p. 74–123, 1985. ISSN 0730-0301.

HALL, E. T. *La Dimension Oculta - Tradução de Joaquin Hernandez Orozco do original: The Hidden Dimension*. Madrid: Instituto de estudios de administracion local, 1973.

HARITAOGLU, D. H. I.; DAVIS, L. S. W4: Who? when? where? what? a real time system for detecting and tracking people. In: *FG '98: Proceedings of the 3rd. Internacional Conference on Face & Gesture Recognition*. Washington, DC, USA: IEEE Computer Society, 1998. p. 222. ISBN 0-8186-8344-9.

HARITAOGLU, D. H. I.; DAVIS, L. S. W4: Real-time surveillance of people and their activities. *PAMI*, v. 22, n. 8, p. 809–830, August 2000.

HELBING, D.; FARKAS, I.; VICSEK, T. Simulating dynamical features of escape panic. *Nature*, v. 407, p. 487–490, 2000.

JAIN, A. K. *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice Hall, 1989.

JUNEJO, I. N.; JAVED, O.; SHAH, M. Multi feature path modeling for video surveillance. *Internacional Conference on Pattern Recognition*, v. 2, p. 716–719, 2004.

KUMAR, P.; SENGUPTA, K.; LEE, A. A comparative study of different color spaces for foreground and shadow detection for traffic monitoring system. *The IEEE 5th International Conference on Intelligent Transportation Systems*, p. 100–105, September 2002.

LAW, A. M.; KELTON, W. D. *Simulation Modeling and Analysis*. 2nd. ed. New York: McGraw-Hill, 1991.

MAKRIS, D.; ELLIS, T. Learning semantic scene models from observing activity in visual surveillance. *IEEE Transactions on Systems, Man and Cybernetics - Part B*, v. 35, n. 3, p. 397–408, 2005.

MARTIN, J.; CROWLEY, J. L. Comparison of correlation techniques. In: AL., U. R. et (Ed.). *Intelligent Autonomous Systems - IAS-4*. Karlsruhe, Germany: [s.n.], 1995. p. 86–93.

MCKENNA, S. J. et al. Tracking groups of people. *Computer Vision and Image Understanding*, v. 80, n. 1, p. 42–56, October 2000.

PONTE, S. D. et al. Novel particle image velocimetry system based on three-color pulsed lamps and image processing. *IEEE Transcactions on Instrumentation and Measurement*, v. 53, n. 1, p. 175–180, February 2004.

PRATI, A. et al. Shadow detection algorithms for traffic flow analysis: a comparative study. *Intelligent Transportation Systems, IEEE*, p. 340–345, 2001.

REYNOLDS, C. W. Flocks, herds and schools: a distributed behavioral model. *SIGGRAPH*, v. 21, n. 21, p. 25–34, July 1987.

ROSENFELD, A.; KAK, A. C. *Digital Picture Processing*. 2nd. ed. New York, NY: Academic Press, 1982.

ROSIN, P. L.; ELLIS, T. Image difference threshold strategies and shadow detection. *6th British Machine Vision Conf., Birmingham*, p. 347–356, 1995.

SALVADOR, E.; CAVALLARO, A.; EBRAHIMI, T. Cast shadow segmentation using invariant color features. *CVIU*, v. 95, n. 2, p. 238–259, August 2004.

SOMMER, R. *Espaço Pessoal - Tradução de Dante Moreira Leite do original: Personal Space: the behavioral basis of design*. São Paulo: Editora Pedagógica e Universitária Ltda, 1973.

STAUDER, J.; MECH, R.; OSTERMANN, J. Detection of moving cast shadows for object segmentation. *IEEE Transactions on Multimedia*, v. 1, n. 1, p. 65–76, 1999.

STAUFFER, C.; GRIMSON, W. E. L. Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, v. 22, n. 8, p. 747–757, 2000.

STRACK, J. *GPSS: modelagem e simulação de sistemas*. Rio de Janeiro: LTC - Livros técnicos e científicos Editora S. A., 1984.

TIAN, Y. li; LU, M.; HAMPAPUR, A. Robust and efficient foreground analysis for real-time video surveillance. In: *CVPR (1)*. [S.l.]: IEEE Computer Society, 2005. p. 1182–1187. ISBN 0-7695-2372-2.

ULICNY, B.; CIECHOMSKI, P. de H.; THALMANN, D. Crowdbrush: Interactive authoring of real-time crowd scenes. *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, p. 243–252, 2004.

WANG, J. J.; SINGH, S. Video analysis of human dynamics - a survey. *Real-Time Imaging*, v. 9, p. 321–346, 2003.

WANG, L.; HU, W.; TAN, T. Recent developments in human motion analysis. *Pattern Recognition*, v. 36, p. 585–601, 2003.

ZELTZER, D. Task-level graphical simulation: Abstraction, representation, and control. In: BADLER, N. I.; BARSKY, B. A.; ZELTZER, D. (Ed.). *Making Them Move: Mechanics, Control, and Animation of Articulaited Figures*. San Mateo, CA: Morgan Kaufmann, 1991. p. 3–33.

ANEXO - PULBICAÇÕES

•JACQUES JR, J. C. S.; JUNG, C. R.; MUSSE, S. R. **Background Subtraction and Shadow Detection in Grayscale Video Sequences**. In: SIBGRAPI, 2005, Natal, RN. Proceedings of SIBGRAPI 2005 (XVIII Brazilian Symposium on Computer Graphics and Image Processing). Los Alamitos, USA: IEEE Computer Press, 2005. v. I, p. 189-196.