FÁBIO DITTRICH

MÉTODO PARA CONTAGEM DE PESSOAS EM MULTIDÕES UTILIZANDO MÚLTIPLAS VISÕES

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática.

Curitiba
2011

FÁBIO DITTRICH

MÉTODO PARA CONTAGEM DE PESSOAS EM MULTIDÕES UTILIZANDO MÚLTIPLAS VISÕES

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática.

Área de Concentração: Ciência da Computação

Orientador: Alessandro Lameiras Koerich
Co-orientador: Luiz Eduardo Soares de Oliveira

Curitiba
2011

Dittrich, Fábio
MÉTODO PARA CONTAGEM DE PESSOAS EM MULTIDÕES UTILIZANDO MÚLTIPLAS VISÕES. Curitiba, 2011.

Dissertação - Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática.

1. contagem de pessoas 2. múltiplas visões 3. CFTV I.Pontifícia Universidade Católica do Paraná. Centro de Ciências Exatas e Tecnologia. Programa de Pós-Graduação em Informática II - t

Dedico este trabalho ao vô Zeca, que sempre teve a ideia, assim como eu, de que educação é uma das coisas mais importantes da vida.

Agradecimentos

Ao meu orientador Alessandro Lameiras Koerich pela inúmeras reuniões para tirar dúvidas, muitas vezes marcadas em cima da hora.

Ao meu amigo Maurício pelas discussões, indicações e ideias dadas.

À minha namorada Mariana pela ajuda na criação de bases de cabeças.

E de forma indireta aos meus pais Roberto e Thaís e ao meu irmão Gustavo pela ajuda financeira, suporte moral, puxões de orelha e companheirismo.

Finalmente, a todos que de alguma forma ou outra me estimularam e apoiaram.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviações

CPs	*corner points*
SVM	*Support Vector Machine*
CFTV	*circuito fechado de televisão*
PETS	*Performance Evaluation of Tracking and Surveillance*
HSV	*Hue, Saturation, Value*
RGB	*Red, Green, Blue*
KNN	*K-Nearest Neighbors*
EMD	*Earth Mover's Distance*
BMA	*Block-Matching Algorithm*
HOG	*Histogram of Oriented Gradients*
CMD	*Coherent Motion Detection*
IPM	*Inverse Perspective Mapping*
ϵ-SVR	*ϵ-Support Vector Regressor*
LBP	*Local Binary Patterns*
MSE	*Mean Squared Error*
ASM	*Active Shape Model*
RNA	*Rede Neural Artificial*

Resumo

Este trabalho apresenta dois métodos inovadores para contagem de pessoas em multidão que combina informações de múltiplas câmeras para mitigar o problema de oclusão, que frequentemente afeta o resultado dos métodos de contagem de pessoas que utilizam somente uma visão da cena. O primeiro método proposto detecta CPs (*corner points*) associados às pessoas presentes na cena e computa os seus vetores de movimentação. O plano da imagem é transformado para o plano do chão utilizando homografia e pesos são atribuídos para cada *corner point* de acordo com a sua distância da câmera. O número médio de pontos por pessoa é estimado e usado para realizar a contagem. O segundo método proposto detecta cabeças utilizando dois classificadores diferentes: um classificador *Adaboost Perceptron* que utiliza características chamadas *Haar Features* e um classificador de Máquinas de Vetor de Suporte (SVM (*Support Vector Machine*)). O plano do chão é estimado através de uma homografia e correspondências entre detecções de cabeças de ambas visões são feitas, permitindo a contagem. Os testes foram feitos sobre a base de vídeo PETS2009 e comparados ao desempenho de outros métodos disponíveis na literatura que utilizam a mesma base de dados. Os resultados do primeiro método são promissores, mas o segundo método ainda precisa ser melhorado para obter resultados bons.

Palavras-chave: contagem de pessoas, múltiplas visões, CFTV

Abstract

This work presents a novel method for people counting in crowded scenes that combines the information gathered by multiple cameras to mitigate the problem of occlusion that commonly affects the performance of counting methods using single cameras. The first proposed method detects the corner points associated to the people present in the scene and computes their motion vector. The image plane is transformed to the ground plane using homography and weights are assigned to each corner point according to its distance to the camera. The mean number of points per person is estimated and used to perform the counting. The second proposed method detects heads with two different classifiers: a Adaboost Perceptron classifier which uses Haar Features and a SVM classifier. The ground plane is achieved through a homography and correspondences of head detections from both views are made, enabling the counting. The experiments were conducted using the video database PETS2009 and our results were compared to the performances of other available methods in the literature, which use the same database. The results of the first method are promising, but the second method still needs to be improved in order to achieve good results.

Keywords: people counting, multiple visions, CCTV

Capítulo 1

Introdução

A utilização de câmeras para realizar as mais diversas tarefas é uma tecnologia na qual se investe cada vez mais. As aplicações que utilizam câmeras vão desde monitoramento do tráfego em rodovias até a viabilização da observação de animais recém-nascidos em zoológicos sem que estes sejam perturbados pelos visitantes, passando por controle de qualidade de peças em indústrias e auxílio na realização de cirurgias [Sas10]. Entretanto, grande parte da utilização dessa tecnologia está nos sistemas de CFTV (*circuito fechado de televisão*) com o objetivo de aumentar a segurança [Sas10] do ambiente em que eles estão inseridos. Um exemplo de aplicação bastante importante é o monitoramento dos movimentos em um aeroporto, que tem o objetivo de melhorar o tráfego aéreo e as condições de segurança dos voos [DSG09]. Essa aplicação, em particular, possui duas partes principais: monitoramento das pessoas no aeroporto, com o intuito de evitar ataques terroristas (pessoas abandonando malas, por exemplo); e monitoramento das aeronaves para ajudar na decisão de suas movimentações.

Quando os circuitos fechados de televisão fazem parte de sistemas de segurança, eles são úteis de duas formas:

- Prevenção de ocorrência de eventos: Nesse caso, é necessário que haja um monitoramento ativo permanente das câmeras e, assim, quando uma ação suspeita estiver se desenvolvendo, é possível fazer algo para intervir.

- Auxílio em situações nas quais o evento já ocorreu: Neste caso, o resultado do monitoramento passivo pode ser analisado após a ocorrência para auxiliar em prevenções futuras e identificar culpados.

Nos sistemas de segurança de prevenção de ocorrência de eventos, são buscadas situações de pessoas entrando/saindo do ambiente, pessoas levando/deixando objetos, número de pessoas em multidão, número de pessoas em um local de alto fluxo, mudança na taxa de ocupação do local, formação de multidões, brigas, entre outros. Com a ajuda da Visão Computacional, é possível criar sistemas automáticos que realizam essas tarefas e auxiliam a tomada de decisões

ou até mesmo que possibilitam a substituição do monitoramento humano pelo monitoramento automático [Sas10].

1.1 Definição do Problema

Dentre as diversas dificuldades para criar um sistema que possa auxiliar o ser humano na tomada de decisões, existe o problema de contagem de pessoas. Genericamente, a contagem de pessoas pode ser feita de duas formas:

- Contagem de quantas pessoas existem numa determinada imagem.
- Contagem de quantas pessoas passaram por uma determinada região da imagem.

Ao tratar cenários com grande fluxo de seres humanos – que são bastante comuns em sistemas CFTV, as aplicações enfrentam problemas muito maiores do que contar pessoas individualmente, já que os indivíduos estão em multidão (ou seja, um grande número de pessoas). Essa maior dificuldade deve-se ao fato de ocorrer um grande número de oclusões dinâmicas e estáticas. As oclusões serão melhor explicadas e exemplificadas na seção 1.4 deste capítulo.

Além disso, a maioria dos métodos existentes que trata do problema de contagem de pessoas em multidão realiza a contagem de pessoas somente da imagem obtida pela câmera e não da cena na qual elas estão inseridas. A cena é o cenário real no qual realizamos a filmagem e do qual queremos extrair o número de pessoas. Ao utilizar a cena e não somente uma visão dela (imagens geradas de uma câmera), possuímos mais informações e estas são mais completas, o que acarreta numa contagem mais precisa.

Por isso, este trabalho visa tratar o problema de contagem de pessoas em multidões de uma cena e não somente de uma imagem. O método proposto não tratará do problema de contagem de pessoas que ultrapassam uma determinada região da imagem.

1.2 Hipótese

Para resolver o problema, partimos da hipótese de que é possível melhorar a contagem e diminuir o erro gerado por oclusões estáticas e dinâmicas se utilizarmos mais de uma câmera, com ângulos diferentes, capturando assim uma porção maior da cena.

1.3 Motivação

Bilhões de dólares são gastos com sistemas CFTV por governos, principalmente nos países desenvolvidos, e, além disso, as vendas para empresas e para propósitos domésticos têm

aumentado [Sas10]. Dessa forma, câmeras, muitas vezes empregadas com fins de segurança, estão amplamente difundidas. Por isso, uma grande motivação para realizar a pesquisa é que já existem, em diversos lugares, as câmeras instaladas. Entretanto, a grande maioria destas câmeras não têm um monitoramento adequado, fazendo com que o seu propósito não seja atingido. Realizando a pesquisa proposta, além de aumentar o nível de segurança, estas câmeras já instaladas e muitas outras que ainda serão instaladas terão sua compra e instalação justificadas. Exemplos de aplicações a que estas câmeras podem servir são: controle de fluxo de pessoas em locais de ocupação restrita, como um museu, um estádio de futebol, um zoológico, etc; controle dinâmico de sinaleiros baseado da análise do fluxo de pessoas no local para otimizar o fluxo de pedestres; e controle da entrada de passageiros em terminais de transporte coletivo. Outra motivação é o fato deste trabalho ser original, pois não existem outros que façam contagem de pessoas da cena e nem utilizando múltiplas câmeras.

1.4 Desafios

A contagem de pessoas em um ambiente pode ser feita de diversas formas, dentre as quais as mais comuns são:

- Contador de mão: um ser humano analisa o ambiente visualmente e para cada pessoa aperta o contador uma vez.
- Raios infravermelhos: quando uma pessoa passa pelo raio, ele é interrompido e o contador é incrementado.
- Imagens térmicas: o sistema utiliza sensores que buscam fontes de calor e as contam.
- Visão computacional: é o utilizado por este trabalho e utiliza câmeras convencionais para capturar imagens da cena. Em seguida, algoritmos de contagem são aplicados para estimar o número de pessoas.

Enquanto o contador de mão depende de uma pessoa para realizar a contagem, o sistema de raios infravermelhos realiza a contagem somente quando os indivíduos passam pela região onde eles estão. Os métodos de imagens térmicas e visão computacional são similares, mas este último é mais acessível em termos financeiros.

Existem diversas dificuldades para estimar quantos indivíduos existem num ambiente com um sistema que utiliza a visão computacional:

- Multidões: O cálculo do número de pessoas é mais simples quando se tratam de apenas alguns indivíduos. Multidões, como mostra a Figura 1.1 aumentam a complexidade do

problema com a ocorrência de oclusões e a necessidade de um maior tempo para realizar a contagem.

Figura 1.1: Uma multidão dificulta o cálculo do número de pessoas.

- Iluminação: Pode atrapalhar com a criação de sombras – o que poder gerar falsas contagens – e com a variação da luminosidade durante o passar do dia (Figura1.2 (a) e (b)), que atrapalha métodos baseados em modelo de fundo(Figura 1.2 (c);

(a) (b)

(c)

Figura 1.2: Iluminação: (a) pouca iluminação, (b) muita iluminação e (c) imagem de movimento com ruído gerado pela diferença de iluminação entre o quadro atual e o modelo de fundo.

- Oclusão dinâmica: quando uma pessoa encontra-se na frente da visão que a câmera possui de outro indivíduo, como mostra a Figura 1.3 (a);
- Oclusão estática: quando o indivíduo locomove-se para trás de um objeto da cena, como mostra a Figura 1.3 (b).

(a) (b)

Figura 1.3: (a) Exemplo de oclusão dinâmica; (b) Exemplo de oclusão estática.

Além das dificuldades citadas, que são comuns a todas as abordagens, existe um desafio relacionado ao objetivo deste trabalho:

- Uma câmera filmando um ambiente não o representa totalmente, como pode ser observado na Figura 1.4. Com a utilização de diversas câmeras, essa representação fica mais precisa. Dessa forma, existe o desafio de calcular o erro da contagem sobre a realidade da cena e não sobre o que a imagem de uma das câmeras mostra.

(a) (b)

Figura 1.4: (a) e (b) são quadros correspondentes da Visão 1 e 2, respectivamente. Em (a), há um indivíduo envolvido por um retângulo que não aparece na Visão 2. Em (b), há um indivíduo envolvido por um retângulo que não aparece na Visão 1. É possível perceber que as imagens individualmente não conseguem representar a cena completamente. Ao utilizar informação de ambas câmeras, melhoramos a representação do ambiente.

1.5 Objetivo

O principal objetivo é criar um método que seja robusto para tratar multidões e que estime o número de indivíduos da cena, não somente de uma das visões dela, melhorando a contagem.

1.6 Contribuições

Este trabalho traz diversas contribuições. Num aspecto social, o trabalho ajuda a melhorar sistemas CFTV de segurança, fazendo com que as pessoas que os utilizam sintam-se mais protegidas e possam agir rapidamente em caso de necessidade, ao serem auxiliadas pelo sistema. Já num aspecto econômico, as câmeras já instaladas e subutilizadas serão mais úteis e o dinheiro gasto com elas justificado. Concomitantemente, o trabalho ajuda a criar um sistema de segurança que ajuda o ser humano a tomar decisões, ao alertar um excesso de pessoas num ambiente, por exemplo. Finalmente, num aspecto científico, o trabalho apresenta um novo método, robusto e que visa resolver problemas de oclusão através da utilização de múltiplas câmeras, além de fornecer uma estimativa do número de pessoas da cena mais precisa.

1.7 Organização do Trabalho

O trabalho está organizado da seguinte maneira. No capítulo 2, é apresentada a Revisão Bibliográfica. Este capítulo distingue as técnicas de contagem de pessoas em dois conjuntos diferentes e as explica detalhadamente. Além disso, contém também outras seções que apresentam artigos que utilizam múltiplas câmeras, explicam como pode ser feita a avaliação do desempenho dos métodos, explicitam quais bases de vídeos foram usadas nos artigos, apresentam métodos que podem ser úteis para o problema de contagem e realizam uma síntese sobre elas. O capítulo 3 traz a metodologia utilizada para desenvolver os métodos a partir de duas abordagens apresentadas na revisão e a forma de utilizar duas câmeras para diminuir a quantidade de oclusões e estimar a contagem da cena. O capítulo 4 possui os resultados obtidos das implementações feitas e comparados aos resultados de outros artigos. E, finalmente, o capítulo 5 apresenta uma conclusão sobre os métodos propostos e a perspectiva para trabalhos futuros.

Capítulo 2

Revisão Bibliográfica

A contagem e estimação do número de pessoas em uma sequência de vídeo recebe cada vez mais atenção dos pesquisadores, pois é um ponto importante em aplicações de monitoramento de ambientes, principalmente em sistemas de prevenção de eventos, como foi citado na introdução. Entretanto, aplicações deste tipo enfrentam problemas como oclusão – quando uma pessoa bloqueia a visualização que a câmera possui de outro indivíduo ou este locomove-se para trás de um objeto – e iluminação – que afeta a detecção das pessoas através da sua variação com o passar do dia, por exemplo [Avi07, YJS06]. Desta forma, cada ambiente necessita de algoritmos robustos que ajudem a resolver seus problemas específicos.

Em um ambiente fechado, como uma sala ou um corredor, por exemplo, é mais fácil controlar a iluminação e pode-se posicionar a câmera de forma que não ocorra oclusão, como no teto, exemplificado na Figura 2.1. Ambientes abertos são mais problemáticos por não permitirem tanto controle e, dessa forma, eles necessitam de métodos diferentes dos que são empregados em ambientes fechados e, também, um outro posicionamento da câmera.

Figura 2.1: Não ocorrência de oclusões ao utilizar uma câmera no teto [KCCK02].

Como observaremos nos artigos apresentados, um dos principais pontos de qualquer abordagem de contagem de pessoas é fazer a detecção dos indivíduos ou da multidão em si. Para isso existem muitas técnicas:

- Detecção através da silhueta das pessoas [CGS02, LDT03, HF01]
- Detecção através da forma de andar (*Gait*) [CGS02]
- Detecção através da simetria das pernas [HSS04]
- Detecção através de padrões de movimento [VJS03]

Para esta revisão bibliográfica foram analisados artigos publicados no PETS (*Performance Evaluation of Tracking and Surveillance*) nos anos de 2002, 2009 e 2010. Nessas ocasiões, um dos desafios propostos foi a contagem de pessoas de uma multidão num ambiente aberto. Em cada um desses anos, uma base com sequências de vídeo foi disponibilizada de forma que os resultados obtidos pudessem ser comparados. Além desses três congressos, também foram usados artigos de outros periódicos como IEEE International Conference on Advanced Video and Signal Based Surveillance, IEEE International Conference on Computer Vision, e relatórios técnicos.

Ao analisar os artigos, é possível perceber duas classes de métodos para resolver o problema da contagem de pessoas: a primeira é composta pela segmentação de cada indivíduo e o possível rastreamento nos quadros seguintes; já na segunda, a multidão é detectada como um todo e uma análise estatística é feita de forma que seja possível estimar a quantidade de pessoas, por exemplo pela área ocupada.

A seguir, é feita uma análise crítica sobre os artigos de cada uma dessas classes.

2.1 Segmentação e Rastreamento de Indivíduos

Os métodos dessa classe realizam detecção de características individuais das pessoas como, por exemplo, a cabeça ou o corpo. Em seguida, geralmente é feito um rastreamento dessas características para validar a sua detecção e realizar a contagem. Ou seja, essa abordagem trata diretamente as características pessoais e as segmentam individualmente.

Sharma, Huang e Nevatia [SHN09] segmentam cada indivíduo e então rastream-os pelos outros quadros da sequência de vídeo. Para detectar as pessoas foi usado um algoritmo de Cluster-Boosted-Tree (CBT) para detecção de pedestres treinado com uma base do MIT [PEP98] com 2862 exemplos de humanos. A característica usada pelo CBT para detectar pedestres é o edgelet. Essa característica é baseada em silhuetas, como pode-se observar na Figura 2.2, e foram definidas *a priori*.

Uma vez detectados os indivíduos, eles são rastreados através de um algoritmo hierárquico baseado em aprendizagem que foi treinado com um conjunto da base TRECVID08 [SOK06]. Esse método não depende de subtração do fundo e por isso é pouco influenciado pela iluminação. Como ele segmenta os indivíduos e os rastreia, é possível extrair outras características de forma

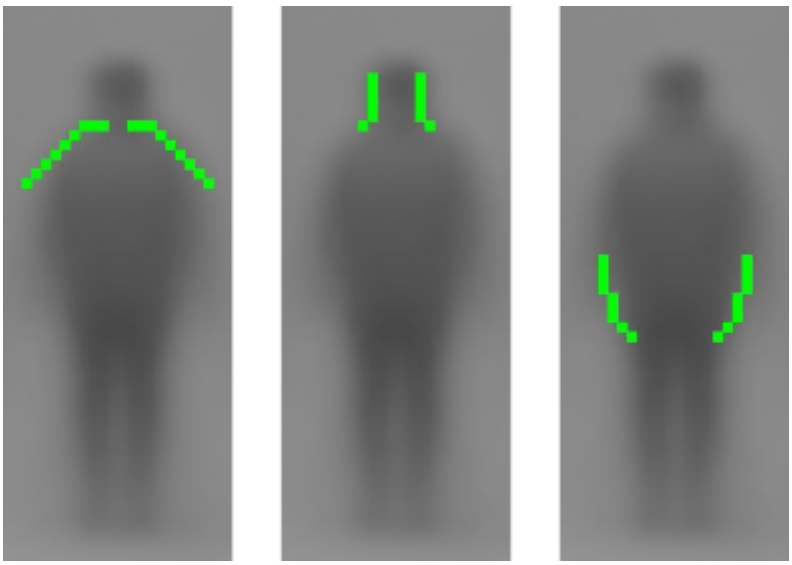


Figura 2.2: Exemplos das características Edgelets [SHN09].

que este método pode ser utilizado para outras aplicações além da contagem de pessoas. Além disso, essa abordagem permite identificar pessoas paradas. Entretanto, um dos principais problemas é a oclusão, já que não é possível obter os edgelets de um indivíduo oculto da câmera. Outro ponto negativo é o fato de tanto a detecção quanto o rastreamento serem baseados em aprendizagem, o que faz com que seja necessário realizar um treinamento previamente . Para avaliar o método desenvolvido, os autores utilizaram a view-001 das fontes de dados S1 e S2 da base de dados PETS2009 (esta e outras bases serão descritas adiante). Somente a visão de uma câmera foi usada pois o método proposto não trabalha com múltiplas visões da cena. Dez quadros de cada sequência foram selecionados e o seu *Ground Truth* (GT) foi gerado. O resultado do algoritmo de contagem é chamado de *Total Detections* (TD) e *Visible Ground Truth* (VG) indica as pessoas visíveis do *Ground Truth*. Ou seja, em VG só são consideradas as pessoas que não estão oclusas – com pelo menos metade do seu corpo visível. A Tabela 2.1 mostra os resultados apresentados no artigo. Ao comparar as contagens obtidas com o *Visible Ground Truth*, os resultados são bons. Entretanto, como o método não trata oclusão, ao comparar com o *Ground Truth* real, a contagem não atinge resultados promissores.

Tabela 2.1: Resultado da contagem de pessoas por edgelets [SHN09].

Sequência	GT	VG	TD
S1	227	189	160
S2	155	146	142

Cucchiara, Grana e Prati [CGP02] apresentam um método chamado Sakbot para detectar objetos em movimento (Moving Visual Objetcs - MVOs) e objetos que estavam em movimento e ficaram estáticos (Stopped Moving Visual Objects - SMVOs). Essa técnica utiliza informações de cor e movimento para detectar objetos, sombras e fantasmas (falsas detecções que são utilizadas para melhorar a detecção posteriormente ao atualizar o modelo de fundo).

Ele é baseado na supressão do fundo e utiliza informações de movimento e sombra para atualizar o plano de fundo. É utilizado o espaço de cores HSV (*Hue, Saturation, Value*) para facilitar na detecção de sombras e atualização do fundo.

Os passos usados pelo Sakbot são:

- Supressão do fundo: É feito com o espaço de cores RGB (*Red, Green, Blue*) entre o quadro atual e o fundo, conforme mostra a Figura 2.3.
- Detecção de sombras: Ao invés de suprimir as sombras, elas são detectadas para atualizar o quadro de fundo. O espaço de cores utilizado nessa etapa é o HSV. A detecção é feita baseada em [PCMT01].
- Computação de *Blobs*: Um algoritmo de rotulação computa os candidatos a objetos e extrai características como área, perímetro, textura, cores, etc.
- Validação e Classificação de Objetos: Utilizando as características extraídas, são detectados os MVOs e SMVOs e então classificados como objetos, sombras ou fantasmas.
- Rastreamento de MVOs e SMVOs: É utilizado um algoritmo simples de rastreamento, que mantém informações dos MVOs e SMVOs, como centroides e próxima posição esperada.
- Atualização do modelo de fundo: O fundo é atualizado constantemente, tornando o Sakbot robusto a mudanças constantes.

Figura 2.3: Interface do Sakbot: quadro atual, fundo, *blob* detectado e sombras (da esquerda para direita e de cima para baixo) [CGP02].

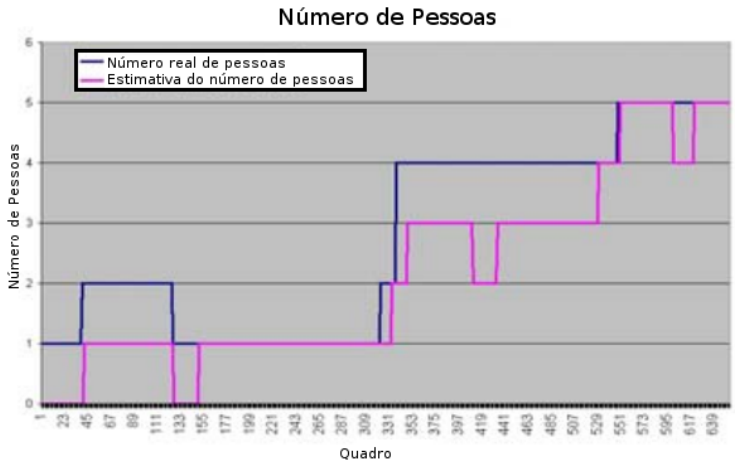
O principal problema deste método é que ele foi feito para trabalhar com poucas pessoas. Multidões são tratadas como somente uma pessoa. Além disso, o custo adicional de detectar sombra em todos os quadros torna necessário um poder computacional maior para continuar com um número razoável de quadros por segundo (entre 15 e 25). Como vantagem, o Sakbot consegue detectar pessoas paradas e faz emprego de sombras e falsas detecções para atualizar o quadro de fundo de forma que a iluminação não seja tão influente, mesmo utilizando uma abordagem com supressão de fundo. Por não possuir um algoritmo robusto de rastreamento e extrair características fracas – que não permitem uma boa detecção – como cor, o Sakbot gerou um erro muito grande na base de dados PETS2002, como mostra a Tabela 2.2

Tabela 2.2: Resultado da contagem de pessoas do Sakbot [CGP02].

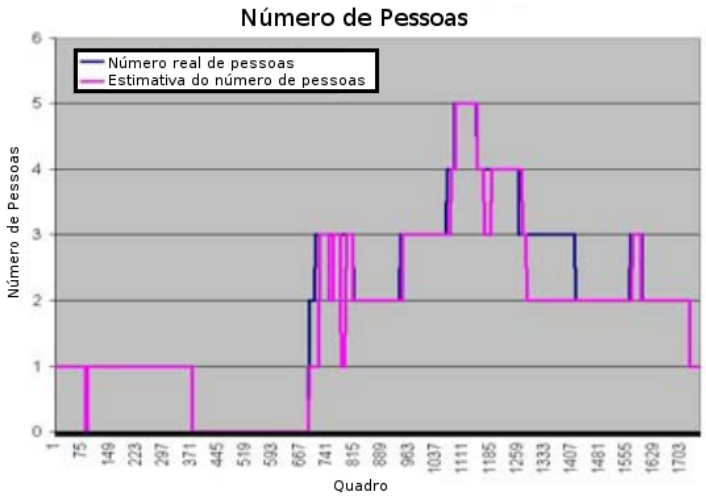
Base de Dados	Contagem do Sakbot	Ground Truth
Dataset1	10	6
Dataset2	17	12
Dataset3	49	15

Marcenaro, Marchesotti e Regazzoni [MMR02] mostram um método para rastreamento e contagem de pessoas em ambientes fechados. A partir dos quadros capturados pela câmera, é passado um filtro da mediana para redução de ruído. Em seguida, são detectadas as mudanças que ocorreram no quadro a partir da subtração do fundo. Como o artigo trata de ambientes fechados, não há uma grande variação na iluminação e por isso não é necessário atualizar o modelo de fundo. Outros tipos de ruído que possam vir a ocorrer são removidos com uma erosão seguida de uma dilatação. Um rastreador de objetos baseado em filtro linear de Kalman é empregado para extrair o contorno das pessoas [SA85], com o qual é feita a contagem. Uma desvantagem é que esse método somente é útil para ambientes fechados, pois ele utiliza subtração de fundo para detectar as pessoas e não possui uma forma de atualizar o modelo de fundo. Assim, variações na iluminação prejudicam o seu desempenho. Um ponto positivo é que existe uma forma de resolver oclusões dinâmicas entre dois objetos que se movimentam, ao utilizar as informações de rastreamento. Os resultados obtidos com esta abordagem são apresentados na Figura 2.4. A linha azul representa o número real de pessoas no vídeo, ou seja, o *Ground Truth*. A linha rosa representa o número de pessoas que o algoritmo detectou. O desempenho do método não é bom, pois além de sempre haver um grande erro, a quantidade de pessoas do vídeo é pequena e não pode ser considerada como multidão.

A abordagem utilizada por Kim et al. [KCCK02] trabalha com uma câmera no teto de forma a minimizar os problemas com oclusão. Eles utilizam uma imagem de fundo e subtração de dois quadros sucessivos para detecção dos objetos e do seu movimento. Há também uma correção feita na imagem de fundo para que a iluminação não seja tão influente. Cada pessoa é rastreada através de um retângulo envolvente, cujo centro é atualizado a cada quadro com relação a velocidade do objeto. A contagem é feita somente se a pessoa atravessa uma determi-



(a)



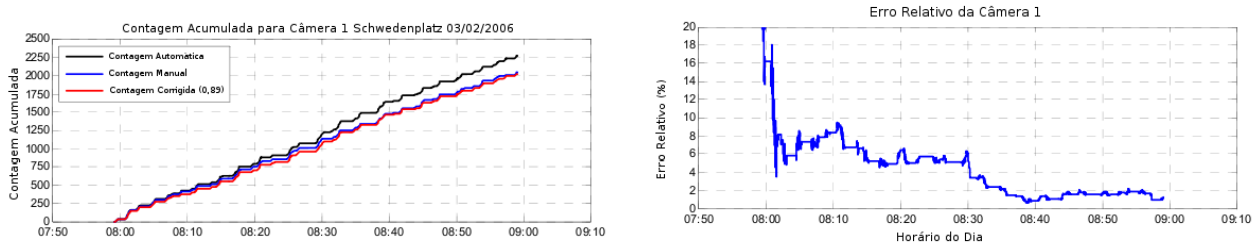
(b)

Figura 2.4: Resultado da contagem de pessoas para Dataset1 (a) e Dataset2 (b) [MMR02].

nada linha. Essa linha, assim como a região para detecção de objetos e a região de rastreamento podem ser definidas pelo usuário através de uma interface gráfica. Duas desvantagens desse método são: ele somente funciona para ambientes internos, já que não é possível colocar uma câmera no teto em um ambiente externo; não funciona para multidões, pois a área de busca pelo objeto é pequena e as pessoas são tratadas individualmente. Os resultados gerados são referentes a uma base própria dos autores. A taxa de contagem correta é de 96%. Como não é uma base comum para que outros métodos sejam testados, não há como fazer uma comparação fiel com outras abordagens.

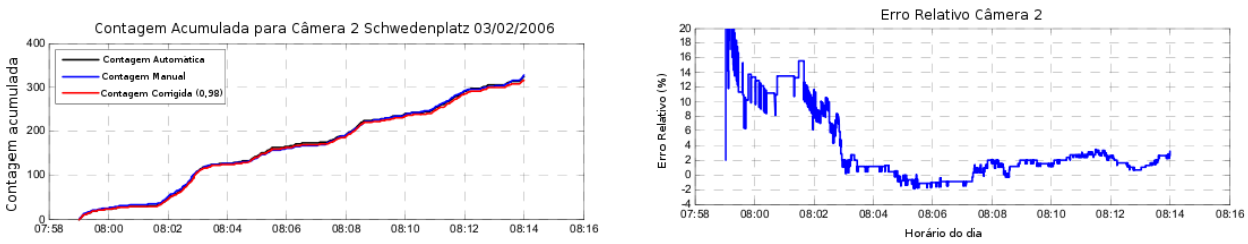
Sidla et al. [SLBS06] utilizam um modelo de fundo para detectar as regiões de interesse e a partir delas buscar pela forma da letra grega Ômega (Ω), que representa o contorno da cabeça e dos ombros de uma pessoa. Cada forma dessas é composta por 23 pontos, sendo que os que estão na cabeça possuem um peso maior para a detecção. Essa detecção é feita utilizando o detector de borda de Canny [GW02] na imagem dentro da região de interesse. Após os candidatos a

pedestre serem computados, a sua movimentação é analisada utilizando o algoritmo de Kanade-Lucas-Tomasi (KLT) [TK91] e uma nova posição para a forma é gerada. Ao redor dessa posição, uma forma é procurada e a mais parecida é associada com a forma do quadro anterior. Em seguida é feita a contagem de pessoas quando cada indivíduo atravessa uma determinada linha. Esse método funciona para contagem de pessoas em multidões, sendo sequências em ambientes internos ou externos. A maior desvantagem é, como nos outros trabalhos, quando ocorre oclusão. Os testes feitos foram em dois cenários diferentes: uma plataforma de metrô (ambiente fechado) e em um terminal de transporte público (ambiente aberto). Dessa forma, é mostrado que o método é robusto e funciona para diferentes ambientes, em diferentes situações. Na cenas em ambientes internos, foram utilizadas duas câmeras com 20 quadros por segundo e resolução de 640x480 pixels. Os resultados da contagem de pessoas e o erro relativo das câmeras são apresentados na Figura 2.5 e na Figura 2.6. Na Figura 2.5, podemos perceber que a contagem acumulada automática (linha preta) está acima da contagem acumulada manual (linha azul). Dessa forma, foi aplicado um fator de correção de 0,89, o que diminuiu o erro (linha vermelha). Já na na Figura 2.6, a contagem acumulada automática (linha preta) está mais precisa, similar à contagem acumulada manual (linha azul). Por isso, o fator de correção é apenas de 0,99 (linha vermelha). Para ambientes externos, foi usada uma câmera com resolução de 800x600 pixels e taxa de 15fps. As informações referentes a contagem de pessoas e erros são mostrados na Figura 2.7. Neste caso, a contagem acumulada automática (linha preta) diverge mais da contagem acumulada manual (linha azul) por causa dos problemas enfrentados em ambientes externos. Assim, há a necessidade de aplicar um fator de correção de 0,85 (linha vermelha).



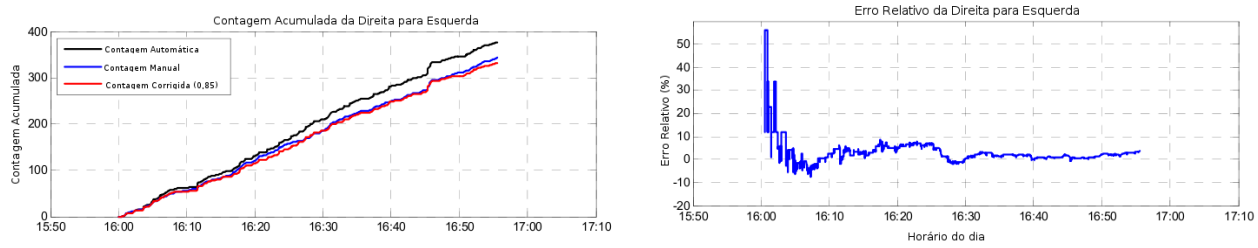
(a) (b)

Figura 2.5: Resultado da contagem de pessoas (a) e erro relativo (b) utilizando a Câmera 1 [SLBS06].



(a) (b)

Figura 2.6: Resultado da contagem de pessoas(a) e erro relativo (b) utilizando a Câmera 2 [SLBS06].



(a) (b)

Figura 2.7: Resultado da contagem de pessoas em ambiente externo (a) e erro relativo (b) [SLBS06].

Zhao, Dellandréa e Chen [ZDC09] apresentam uma abordagem para contagem de pessoas baseada na detecção e rastreamento de faces e classificação de trajetória no vídeo. Para realizar a detecção das faces é utilizado um algoritmo detector de faces simples baseado em [TCB04], enquanto para fazer o rastreamento dessas características usa-se a combinação de um filtro Kalman invariante à escala com um algoritmo de rastreamento baseado em *kernel*. De cada potencial trajetória de uma face é extraído um histograma angular dos pontos vizinhos e com um algoritmo de KNN (*K-Nearest Neighbors*) baseado na distância EMD (*Earth Mover's Distance*) [RTG98] é feita a classificação, separando trajetórias de faces verdadeiras de trajetórias de faces falsas. Então, a contagem é realizada levando em conta quantas trajetórias existem no quadro atual. O método proposto foi testado numa base de vídeos própria (cinco vídeos com 6345 quadros no total, adquiridos com uma câmera de resolução de 320x240) e obteve uma taxa de acerto de 93% como mostra a Figura 2.8. O classificador KNN foi testado para valores 1, 3, 5 e 7 para N e com 10, 15, 20, 25 e 30 trajetórias. Além disso, o algoritmo é robusto o suficiente para continuar rastreando a face depois que ela ficou oclusa por alguns quadros, como exemplifica a Figura 2.9.

O maior problema dessa abordagem é que na grande maioria das bases de vídeos e de sequências de vídeo reais, geradas por sistemas CFTV em funcionamento, não é possível obter informações suficientes referentes às faces para detectá-las, rastreá-las e fazer a contagem de pessoas. Geralmente, as câmeras não possuem uma resolução adequada e nem ficam próximas

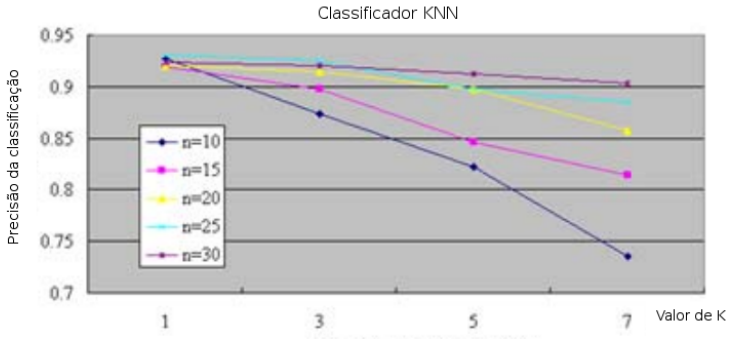


Figura 2.8: Resultados da classificação de trajetórias de faces por KNN [ZDC09].

Figura 2.9: Exemplo de rastreamento de face quando ocorre oclusão [ZDC09].

o suficiente dos indivíduos. Dessa forma, este método torna-se muito limitado.

Valle Jr. [VJ07] propõe um novo método para realizar contagem automática de pessoas em sequências de vídeo. Este método baseia-se na segmentação dos indivíduos, seu rastreamento e a contagem quando o indivíduo entra numa área pré-definida do vídeo. Antes de realizar a segmentação, é passado um filtro de mediana 3x3 em cada um dos canais RGB da imagem para redução de ruído. Já a segmentação é feita utilizando uma subtração de fundo simples, na qual a imagem de fundo é o primeiro quadro. Uma abertura binária é feita para eliminar algum ruído remanescente e um fechamento binário para conectar todas as partes do *blob* segmentado. Um retângulo envolvente é criado em volta de todas as componentes conexas do quadro e estes são rastreados. Em situações de oclusão, um algoritmo de previsão baseado em velocidade é acionado. Quando um *blob* entra na área de contagem, a sua largura é comparada a um limiar de largura média de uma pessoa (definido empiricamente). Se for menor ou igual ao limiar, só existe uma pessoa neste *blob*. Caso contrário, a área da região superior é comparada a um outro limiar de forma a determinar se existem duas ou três pessoas no *blob*. Uma outra forma

de realizar a contagem é aplicar um classificador KNN (K-Nearest Neighbors) treinado para determinar se existem uma, duas ou três pessoas no grupo. O método foi testado em duas bases de vídeo: CAVIAR [refa] e uma base criada pelo próprio autor. Os resultados são apresentados na Tabela 2.3. Podemos perceber que a utilização da classificação para determinar o número de pessoas dentro de grupos possui um resultado melhor que a abordagem de limiares.

Tabela 2.3: Resultados obtidos por [VJ07].

	Contagem Manual	Somente Rastrea-mento	Baseada em Limiares	Análise da Região Superior (11-nn)	Análise da Região Superior (5-nn)	Análise da Região Superior (1-nn)
CAVIAR	92	74	81	92	93	91
CCET	128	94	155	142	149	157

Essa técnica possui três grandes problemas: o modelo de fundo não é atualizado, o que faz com que a iluminação e a mudança gradual do fundo sejam fatores que geram ruído; oclusões mais complexas que pessoas andando perto umas das outras não são tratadas; e não é possível tratar multidões grandes – com grande número de pessoas –, somente grupos pequenos de pessoas.

No trabalho de Merad, Aziz e Thome [MAT10], foi proposta uma técnica que segue o fluxograma da Figura 2.10. Primeiramente, é feita a subtração de fundo como sugerido por Stauffer et al. em [SWG00]. Essa segmentação do primeiro plano é feita com um modelo de cada pixel de fundo através de uma mistura de três a cinco distribuições gaussianas, sendo que cada uma delas representa uma cor diferente. Os pesos de cada gaussiana na mistura correspondem à quantidade de tempo que essas cores permanecem na cena. Com os *blobs* computados e suavizados para remoção de ruído, são calculados os seus esqueletos. Os pontos que compõem os esqueletos têm um número de vizinhos variável: um ponto com um vizinho corresponde a um extremo do corpo da pessoa; um ponto com exatamente dois vizinhos é um ponto que liga um ponto extremo com um ponto inicial; e um ponto com mais de dois vizinhos é um ponto de onde os segmentos começam. Com os esqueletos dos *blobs*, são detectadas as cabeças procurando por pontos com somente um vizinho e cuja inclinação com relação ao eixo vertical seja pequena, como mostra a Figura 2.11. A detecção da pose da cabeça é feita para ignorar alguns falsos positivos.

Esse método depende demais da escolha dos pontos do esqueleto e, quando existem multidões e oclusões, os esqueletos podem ser montados incorretamente, acarretando em erro no estágio da detecção da cabeça, mesmo que esta esteja visível enquanto o corpo está ocluso, como mostra a Figura 2.12 (c). Já a Figura 2.13 mostra o resultado da contagem do método na sequência de vídeo S1_L1_Time13-57_view001 da base PETS2009 comparada ao seu *Ground Truth*. É possível perceber que há uma variação muito grande na contagem de um quadro para o outro. Mesmo assim, o método comporta-se bem até um determinado número de pessoas.

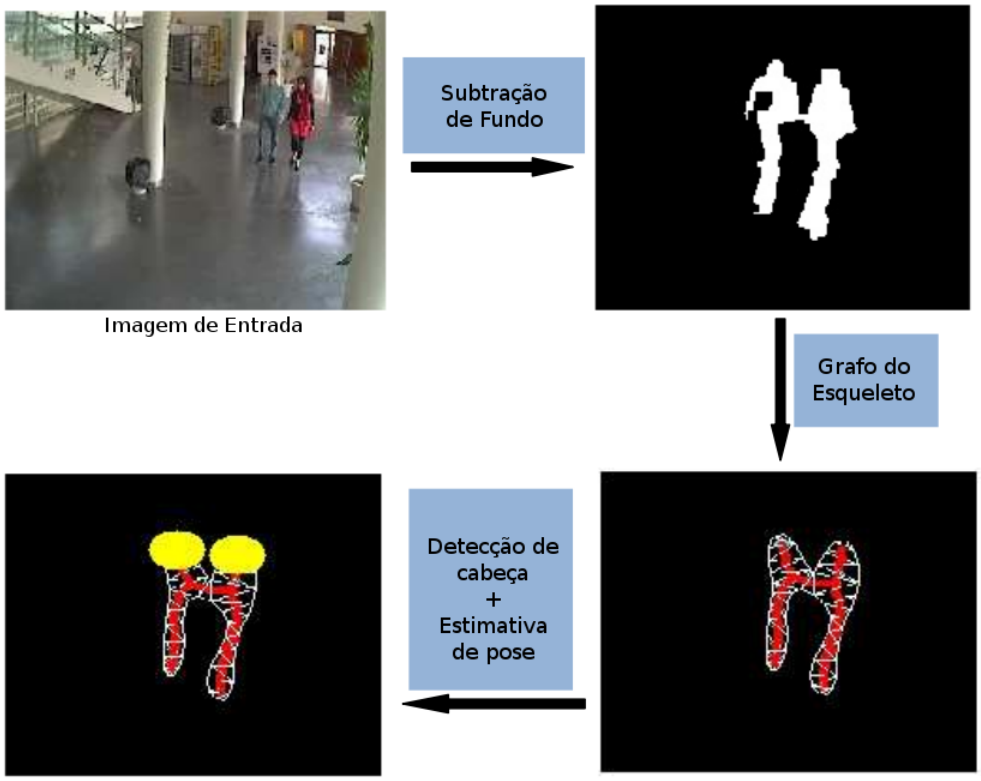


Figura 2.10: Fluxograma de como funciona o sistema de [MAT10].

Quando esse número aumenta, as oclusões atrapalham e a contagem piora consideravelmente, ainda tendo uma variação grande entre quadros consecutivos.

A seguir são apresentados alguns trabalhos nos quais a contagem é feita com uma abordagem probabilística, após detectar a multidão.

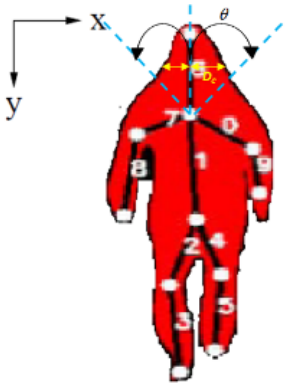


Figura 2.11: Detecção da cabeça de acordo com a inclinação [MAT10].

(a) (b) (c)

Figura 2.12: Construção do esqueleto para uma pessoa (a), para duas pessoas (b) e uma situação com oclusão parcial (c) [MAT10].

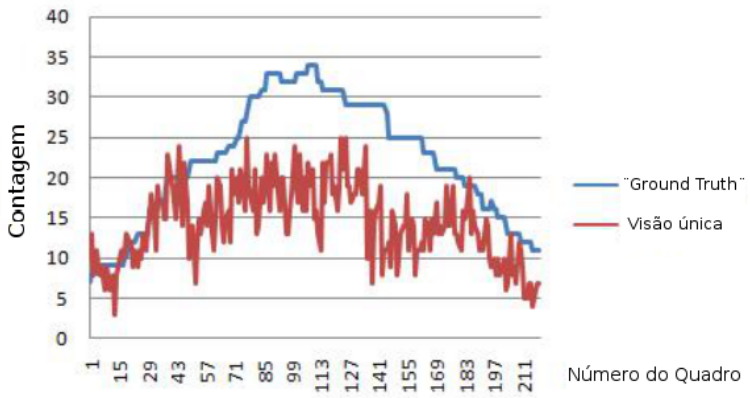


Figura 2.13: Contagem do método comparada ao *Ground Truth* [MAT10].

2.2 Segmentação de Multidões e Análise Probabilística

Os métodos dessa classe realizam detecção de características de grupos de pessoas como, por exemplo, a área ocupada ou o perímetro. Em seguida, geralmente é estimado o número de pessoas na imagem utilizando uma função previamente treinada que mapeia as características calculadas para o número de indivíduos da cena. Ou seja, essa abordagem trata de características do grupo, fazendo uma análise probabilística sobre elas.

O trabalho descrito por Albiol et al. [ASAM09] utiliza características chamadas de *corner point* e seus vetores de movimento para detectar multidões. Para definir os CPs, são calculados os gradientes horizontal e vertical de cada pixel da imagem através do filtro de Sobel [GW02], sua respectiva magnitude, a matriz de covariância do gradiente ao redor de cada pixel e os seus autovalores. Os vetores de movimento das características encontradas são estimados por um algoritmo de correspondência de blocos (BMA (*Block-Matching Algorithm*)) [Tek95] e é possível, então, distinguir *corner points* que possuem vetor de movimento nulo (pontos vermelhos) dos que possuem movimento de um quadro para o outro (pontos verdes), como mostra a Figura 2.14.

Figura 2.14: Exemplo de CP detectados e seus vetores de movimento [ASAM09].

Com a multidão detectada, temos como saber quantos *corner points* existem em cada imagem e, a partir de um treinamento com uma base previamente classificada manualmente, quantos existem por pessoa. Desta forma é estimado o número de indivíduos em cada quadro. A técnica descrita naquele artigo utiliza BMA para estimar os vetores de movimentação e, por isso, não faz a estimação do quadro de fundo, o que é uma vantagem já que a iluminação é um fator que atrapalha nesta etapa. Assim, é possível utilizar esta técnica para um ambiente externo. Outro ponto forte é que, por não utilizar modelo de fundo, não existe o problema de tratar as sombras das pessoas. Em contrapartida, esse método necessita de aprendizagem e calibração, ou seja, qualquer mudança no sistema de câmeras acarreta na recalibração e

num novo treinamento. Uma outra desvantagem é o fato de o número de CPs por pessoa ser uma estimativa que depende da calibração para cada cenário. Além disso, por tratar a multidão como um todo, não há como fazer rastreamento de um indivíduo para extrair mais características. Esse fator impede o uso desse método em algumas aplicações que necessitam de mais características além do número de pessoas. Outro ponto negativo é que se uma pessoa está estática no vídeo, os vetores de movimento de seus *corner points* serão nulos e ela não será detectada. Esse método também não considera a oclusão. Os resultados apresentados no artigo não foram comparados a um *Ground Truth* e por isso é difícil avaliar quão bom eles são. É possível fazer essa análise observando a comparação dos métodos submetidos ao PETS2009, apresentado no final deste capítulo. Na Figura 2.15 podemos ver o número de pessoas que esta abordagem detectou para duas regiões diferentes e tempos diferentes.

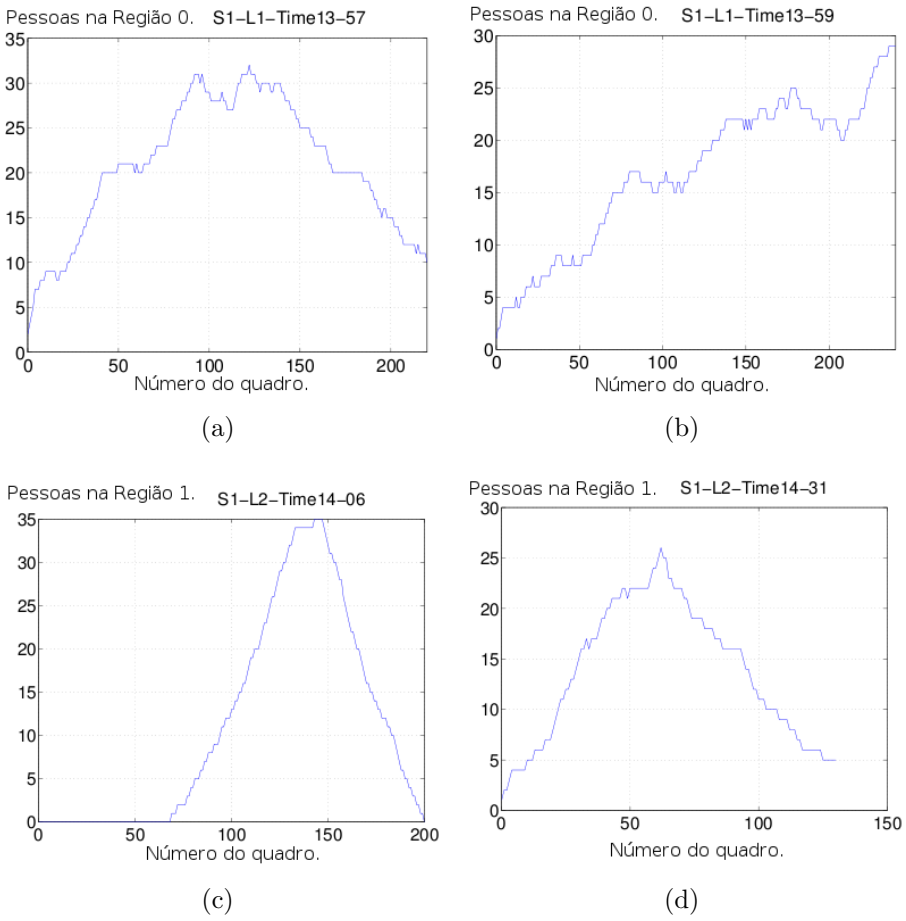


Figura 2.15: Número de pessoas detectadas na região R0 da base S1_L1 em tempos diferentes (a) e (b) e número de pessoas detectadas na região R1 da base S1_L2 (c) e (d) em tempos diferentes [ASAM09].

O trabalho de Chan, Morrow e Vasconcelos [CMV09] baseia-se em fazer uma regressão de características globais. As pessoas são segmentadas através de uma mistura dinâmica de texturas [CV08] que detecta multidões movendo-se em direções diferentes. De cada uma dessas regiões são extraídas 30 características como: área do segmento, perímetro do segmento, contagem do número de componentes conexas com mais de dez pixels do segmento, orientação

das bordas, homogeneidade, energia e entropia da textura, etc. Como essas características são variantes à distância que o objeto encontra-se da câmera, deve-se fazer uma normalização da perspectiva de forma que isso não influencie o resultado. Desta forma, quanto mais longe o objeto está, maior é o peso que é aplicado às suas características. Só então o modelo de mistura é aprendido por um algoritmo chamado de *expectation-maximization* [CV08]. Com isso, o número de pessoas é estimado fazendo-se uma regressão Gaussiana no vetor de características. Uma desvantagem é que há necessidade de fazer um ajuste na perspectiva, de forma que as características fiquem invariantes à distância, para qualquer movimento que a câmera sofra ou para cada câmera que está numa posição diferente. Além disso, é necessário fazer um treinamento sobre o modelo de texturas. Com esse método, diferente do apresentado por Albiol et al. [ASAM09], é possível rastrear cada segmento obtido de forma a obter novas características se necessário. Neste artigo, também foi usada a view-001 dos vídeos do PETS2009, o que facilita a comparação dos desempenhos. O *Ground Truth* do quinto quadro foi feito manualmente, contando indivíduos andando para a esquerda e para a direita, e a contagem dos outros quadros foi feita utilizando uma interpolação linear. A Tabela 2.4 mostra o número total de detecções da abordagem para cada vídeo, o *Ground Truth* calculado e o erro por quadro dos vídeos da base PETS2009. Os erros da sequência S1.L2 14-06 foram maiores pois esse vídeo possui uma multidão bastante densa, que não é representada na base de treinamento.

Tabela 2.4: Resultado da contagem de pessoas por texturas [CMV09].

Video	Região	Contagem	Ground Truth	Erro por quadro	Número de quadros
S1_L1 13-57	R0	4411	4838	2.46	218
S1_L1 13-57	R1	2301	2757	2.28	217
S1_L1 13-57	R2	1436	1437	0.99	201
S1_L1 13-59	R0	3455	3628	1.41	240
S1_L1 13-59	R1	1636	1539	0.69	217
S1_L1 13-59	R2	1313	1473	1.23	228
S1_L2 14-06	R1	1727	2462	5.89	131
S1_L2 14-06	R2	1078	1629	4.48	132
S1_L3 14-17	R1	518	481	0.98	50

Pätzold, Evangelio e Sikora [PES10], apresentam uma abordagem direta de contagem de pessoas em multidão, mostrada na Figura 2.16. Enquanto a maioria dos métodos atuais utiliza gradiente simples para analisar e detectar padrões humanos nas imagens, este utiliza um algoritmo chamado Histograma de Gradientes Orientados (HOG (*Histogram of Oriented Gradients*)), descrito por Dalal et al. em [DT05], que coleta o gradiente de pequenas regiões da imagem e as representam através de histogramas direcionais. Esses histogramas são normalizados e concatenados com os histogramas de regiões adjacentes e classificados por Máquinas de Vetor de Suporte. Entretanto, o algoritmo HOG foi modificado para trabalhar com somente as regiões da cabeça e ombros das pessoas por causa das oclusões, já que o algoritmo tradicional utiliza informação do corpo todo. Com este classificador, é possível encontrar regiões candidatas a cabeças, só que, por conta de ter sido treinado com uma pequena parte do corpo

humano, são detectados muitos falsos positivos. Por isso, é feita a fusão dessa informação de forma com a informação de movimento da imagem, de maneira que somente cabeças sejam detectadas, como mostra a Figura 2.17. Após essas etapas, é aplicada a informação referente ao rastreamento das cabeças detectadas de forma a validar a sua existência, através da Detecção de Movimentação Coerente (CMD (*Coherent Motion Detection*)), que impede que objetos que as pessoas carregam, e que se parecem com cabeças, façam com que trajetórias erradas sejam criadas. Na Figura 2.18, observamos o número de pessoas detectadas na sequência de vídeo S1_L1_Time13-57_view001 da base PETS2009 utilizando duas formas de calcular o fluxo ótico (diferença entre os quadros consecutivos) e comparada ao *Ground Truth* feito manualmente. O resultado é atingido satisfatório para as especificidades do problema tratado.

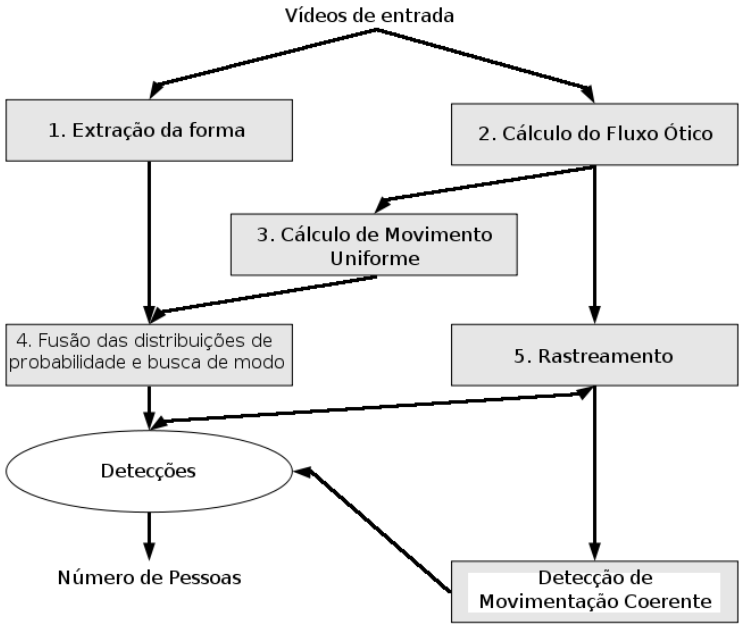


Figura 2.16: Fluxograma de como funciona o sistema de [PES10].

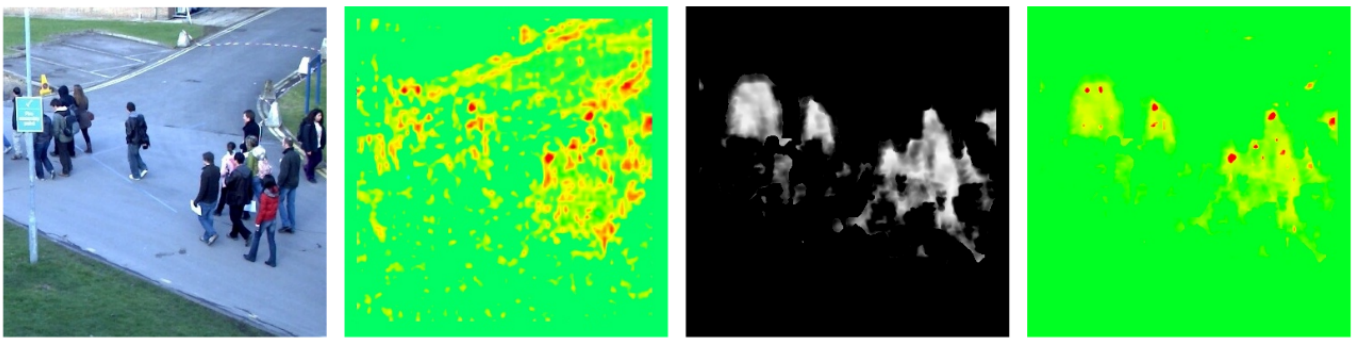


Figura 2.17: A imagem de entrada, o mapa de probabilidade do detector, a informação de movimentação e o mapa de probabilidade do detector com a informação de movimentação [PES10].

Ryan et al. [RDFS10] apresentam um método melhorado de uma abordagem previamente desenvolvida, que utiliza características locais de grupos de pessoas representadas por *blobs* no

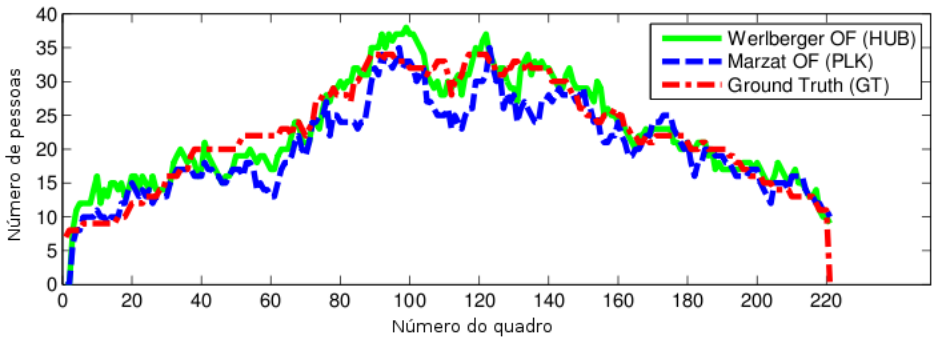


Figura 2.18: Resultados do método utilizando duas forma de calcular o fluxo ótico comparado ao *Ground Truth* [PES10].

primeiro plano e o seu subsequente rastreamento ao longo dos quadros para calcular o número de pessoas, de forma que a contagem total do quadro é a soma da contagem dos *blobs*. As características usadas são área, perímetro, razão entre perímetro e área, bordas e histograma dos ângulos das bordas. O uso de rastreamento dos grupos de pessoas permite analisar o seu histórico, como situações de fusão ou separação de grupos. Para que as características tenham consistência quanto à distância que estão da câmera, o método de mapa de perspectiva apresentado por Chan et al. [CMV09] é utilizado. A contagem de cada *blob* é feita com um modelo linear de mínimos quadrados, no qual cada característica tem um peso. Os resultados são mostrados na Tabela 2.5. O erro gerado pela estimativa da contagem foi diminuído com o sistema apresentado, obtendo bons resultados na base testada.

Tabela 2.5: Resultados sobre a base de pedestres UCSD de Chan et al. [CLV08]. Os testes incluem resultados do trabalho anterior e do proposto por Ryan, modificando diversos parâmetros [RDFS10].

Sistema	Erro	MSE
Ryan et al. original	1.2991	2.7470
Ryan novo, sem rastreamento	1.2586	2.5504
Ryan novo, atualização adaptativa (α=0.5)	1.2408	2.4872
Ryan novo, atualização adaptativa (α=0.25)	1.2307	2.4610
Ryan novo, atualização adaptativa (α=0.05)	1.2224	2.4402
Ryan novo, valor médio	1.2245	2.4078
Ryan novo, valor mediano	1.2212	2.3586

Conte et al. [CFP+10] apresentam um método de contagem de pessoas em multidão que foi extensivamente comparado ao método de Albiol et al. [ASAM09] por existirem algumas semelhanças entre eles. Entretanto, este não trata da perspectiva e da densidade dos grupos e é exatamente esse o maior diferencial do método proposto por Conte et al. [CFP+10]. O sistema descrito segue o fluxograma da Figura 2.19.

Um problema do trabalho de Albiol et al. [ASAM09] é que os *corner points* de um objeto não são estáveis, pois não se mantém ao longo dos quadros. Dessa forma, o algoritmo desenvolvido por Conte utiliza *SURF Points*, explicados em [BETVG08], como pontos de interesse, que são menos dependentes de escala e orientação. Após serem detectados, é feito um agrupamento

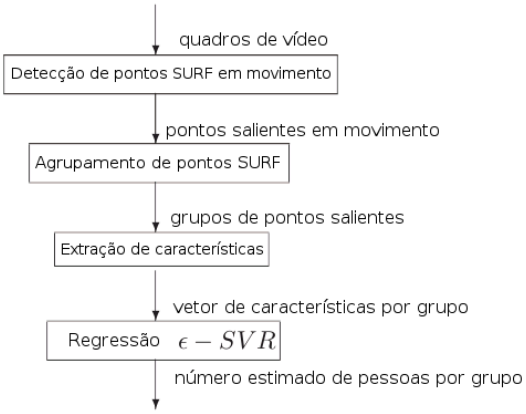


Figura 2.19: Fluxograma de como funciona o sistema de [CFP+10].

dos pontos com um algoritmo baseado em grafos, pois algoritmos tradicionais como *k-means* e *DBSCAN* não podem ser utilizados pelo problema não apresentar informações *a priori*, tais como: forma, tamanho e densidade dos grupos. Com esses grupos calculados, é possível resolver o problema de perspectiva ao calcular a distância dos grupos à câmera. Para isso é utilizado um Mapeamento de Perspectiva Inversa (IPM (*Inverse Perspective Mapping*)) nos pontos mais baixos dos grupos, assumindo que eles estão no chão. Outra melhoria é o calculo da densidade de cada grupo, feito através da razão entre número de pontos pela área ocupada. Com todas essas características calculadas, uma relação entre elas e o número de pessoas da cena não é uma simples proporção como em [ASAM09], já que temos que relacionar densidade e perspectiva, além dos pontos de interesse, com o número de pessoas da cena. Por isso, uma variante de SVM chamada ϵ-Regressão de Vetor de Suporte (ϵ-SVR (ϵ-*Support Vector Regressor*)) é treinada para estimar o número de pessoas dado o número de pontos de um grupo, a sua distância para a câmera e a sua densidade. Os resultados atingidos são comparados aos do método de Albiol et al. [ASAM09] na Tabela 2.6 utilizando alguns vídeos da base PETS2009. É possível perceber que o algoritmo proposto por Conte fazem com que o método atinja resultados melhores do que os atingidos por Albiol.

Tabela 2.6: Os resultados de Albiol et al. e de Conte et al. Os valores correspondem ao erro médio absoluto e erro médio relativo [CFP+10].

Video (visão)	Albiol et al.	Conte et al.
S1_L1 13-57 (1)	2.80 (12.6%)	1.92 (8.7%)
S1_L1 13-59 (1)	3.86 (24.9%)	2.24 (17.3%)
S1_L2 14-06 (1)	5.14 (26.1%)	4.66 (20.5%)
S1_L3 14-17 (1)	2.64 (14.0%)	1.75 (9.2%)
S1_L1 13-57 (2)	29.45 (106.0%)	11.76 (30.0%)
S1_L2 14-06 (2)	32.24 (122.5%)	18.03 (43.0%)
S1_L2 14-31 (2)	34.09 (99.7%)	5.64 (18.8%)
S3_MF 12-43 (2)	12.34 (311.9%)	0.63 (18.8%)

Analisando os artigos das duas abordagens apresentadas, percebemos que é possível tratar o problema de contagem de pessoas de formas diferentes: extrair características do

indivíduo ou da multidão. No geral, cada abordagem tem suas vantagens e desvantagens que devem ser analisadas de acordo com as necessidades do ambiente onde será realizada a contagem.

A próxima seção apresenta alguns artigos em que são utilizadas múltiplas câmeras. Esses trabalhos foram estudados para definir como combinar duas visões da cena na qual iremos realizar a contagem de pessoas e para minimizar a ocorrência de oclusões.

2.3 Múltiplas Câmeras

Um dos maiores e mais recorrentes problemas com sistemas de CFTV é a oclusão. Uma alternativa para contornar este problema é utilizar duas ou mais câmeras. Dessa forma, uma pessoa pode estar oculta da visão de uma câmera e visível para outra, resolvendo o problema da oclusão. Para atingir esses objetivos é necessário realizar uma correspondência entre as câmeras, ou seja, deve ser possível calcular o ponto correspondente em outra visão de um dado ponto na visão de origem. Por isso, com múltiplas câmeras é possível realizar uma contagem da cena mais precisa. Entretanto, não existe na literatura um método que utilize múltiplas câmeras para contagem de pessoas em vídeo. Esta seção apresenta trabalhos nos quais foram utilizadas múltiplas câmeras em aplicações que não a contagem de pessoas.

Wu et al. [WHH+09] propõem um método que utiliza diversas câmeras para rastrear um grande número de objetos em três dimensões. O problema de correspondência entre as múltiplas visões é resolvido com um procedimento de busca adaptativa e aleatória. Então, as trajetórias dos objetos são estimadas através de uma reconstrução estereoscópica usando uma busca epipolar na vizinhança. O uso de múltiplas visões ajuda a resolver o problema da oclusão de objetos, como mostra a Figura 2.20. Utilizando somente uma das visões, o objeto *O2* pode ficar ocluso pelo objeto *O1*. Assim, um algoritmo de rastreamento que utilize somente uma das câmeras pode deixar de seguir um dos objetos ou interpretar, erroneamente, o falso-positivo *Z1,3* como se fosse o objeto ocluso. Já um algoritmo que utilize as duas câmeras poderá continuar corretamente rastreando os dois objetos.

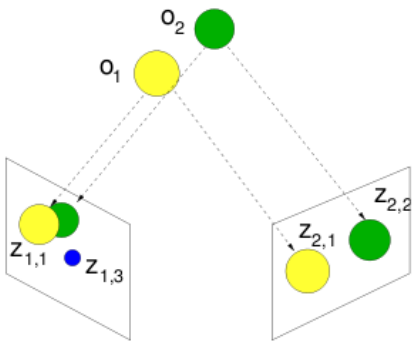


Figura 2.20: As visões de duas câmeras do mesmo cenário [WHH+09].

Essa abordagem foi testada para rastrear centenas de morcegos voando e apresentar uma

análise do comportamento do grupo. Os resultados apresentados na Tabela 2.7 mostram que grande parte das oclusões podem ser resolvidas por este método. Entretanto, não é possível resolver todas as ambiguidades referentes a oclusões em situações de alta densidade devido à insuficiente resolução das imagens (640x512). Além disso, o algoritmo proposto neste artigo busca durante cinco quadros a trajetória de um objeto perdido. Caso ele não seja encontrado, uma nova detecção, e com isso uma nova trajetória, são feitas. Dessa forma, o número de rastreamentos é maior que o número de morcegos, como mostra a Tabela 2.7, que apresenta o número de morcegos detectados, o número real de morcegos, a quantidade de trajetórias criada, o número de oclusões que ocorreram e quantas em quantas delas foi possível recuperar o resultado correto.

Tabela 2.7: Resultados obtidos por [WHH+09].

# of Bats/-Frame	True # of Bats	Computed # of Tracks	# of Occlusions	# of Recovered Occlusions
20	25	33	56	40
40	50	63	94	54
60	71	90	140	86
100	119	185	368	88

Snidaro, Visentini e Foresti [SVF09] apresentam um método para fundir informações de dois sensores. A detecção do objeto e seu rastreamento são feitos através de uma classificação feita por um conjunto de classificadores treinados previamente e atualizados em tempo de execução e que usam características heterogêneas para cada objeto. Então, é feita uma estimativa da posição do objeto em uma planta baixa através da fusão dos mapas de probabilidade de cada câmera e, depois, a aproximação desses mapas de probabilidade através de uma função gaussiana. As características utilizadas para detectar os objetos foram: características Haar, padrões binários locais (LBP (*Local Binary Patterns*)) e histogramas de cor. A Figura 2.21 exemplifica a transformação homográfica feita para gerar a planta baixa. Em seguida, a Figura 2.22 mostra as trajetórias que o objeto percorreu em cada câmera e as trajetórias colocadas na planta baixa. Em seguida, as informações são fundidas, gerando a trajetória para a imagem da transformação homográfica.

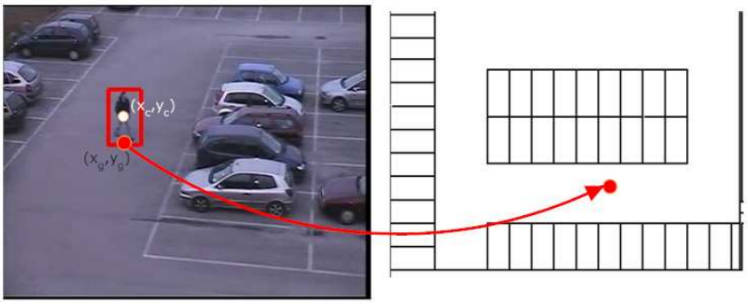
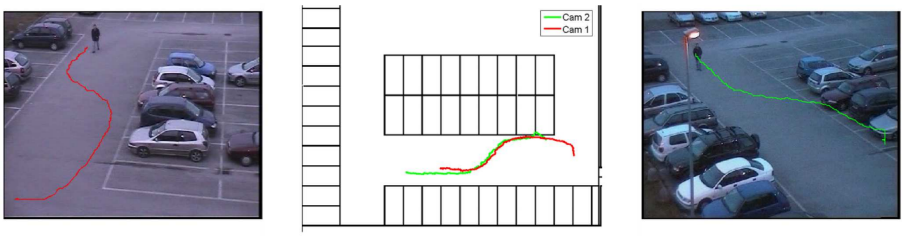
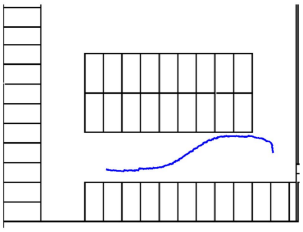


Figura 2.21: Criação da planta baixa através de uma transformação homográfica [SVF09].



(a)



(b)

Figura 2.22: Trajetórias das câmeras e sua projeção em planta baixa (a) e a trajetória fundida (b) [SVF09].

Verstockt et al. [VDBP+09] propõem uma abordagem para localização de objetos em sequências de vídeo utilizando múltiplas visões através de transformações homográficas. A segmentação do primeiro plano é feita em cada visão utilizando diferença temporal baseada em partições de macro blocos. Para extrair os objetos, são aplicados aos primeiros planos segmentados algoritmos de fusão de *blobs* (silhuetas), menor polígono envolvente e remoção de ruídos. Em seguida, esses objetos são projetados numa imagem de planta baixa através de uma transformação homográfica. Então, para localizar os objetos pode-se procurar por máximos locais na imagem de planta baixa, pois os primeiros planos de cada visão irão se sobrepor onde estão os objetos, como mostra a Figura 2.23. Já a Figura 2.24 mostra um fluxograma de como funciona esta abordagem.

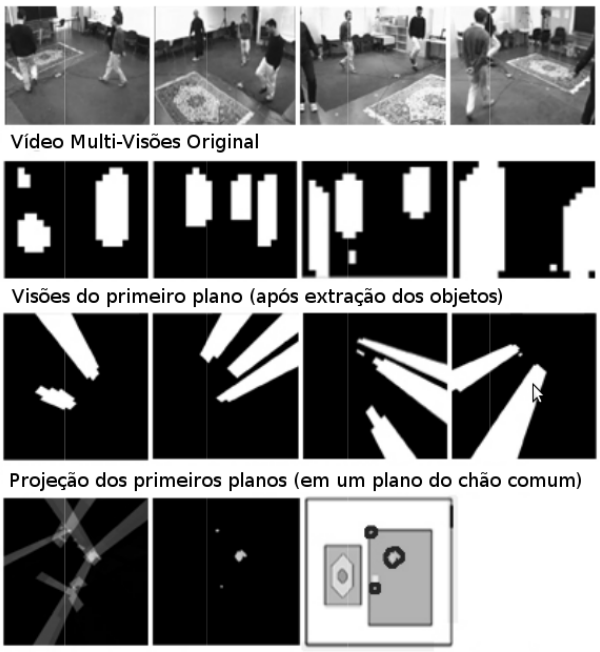


Figura 2.23: Exemplo da localização de objetos de [VDBP+09].

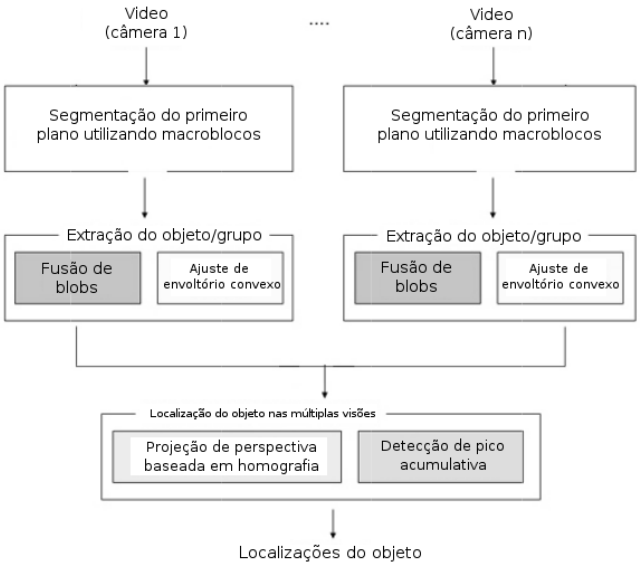


Figura 2.24: Fluxograma para localização de objetos utilizando uma técnica com múltiplas visões e macro blocos [VDBP+09].

Para testar o método, ele foi submetido à base CVLAB [FBLF08]. A Figura 2.25 mostra a taxa de acerto da localização de objetos comparada ao *Ground Truth* e usando limiares variáveis. Pode-se perceber que para o limiar ideal, o algoritmo possui um bom desempenho.

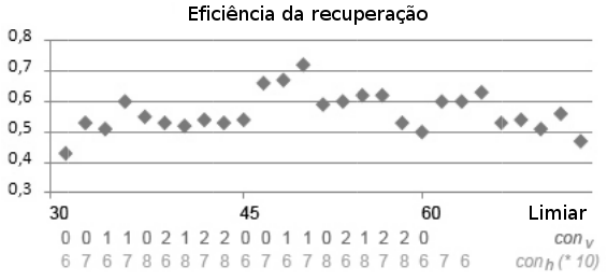


Figura 2.25: Resultado da localização de objetos [VDBP+09].

Os trabalhos que utilizam múltiplas câmeras nos dão uma ideia de como fazer a fusão das informações capturadas delas. Para os métodos propostos, decidiu-se utilizar a técnica de transformação homográfica, pois com os dados sobre as câmeras que foram fornecidos pela base é possível calculá-las facilmente. Esta técnica consiste em criar uma matriz homográfica que projeta os pontos de um plano em outro. Dessa forma, é possível transformar o plano de cada visão para a planta baixa e combinar as informações.

Na seção seguinte, falaremos sobre algumas bases de dados que foram utilizadas para realizar treinamentos de algoritmos e também para testar métodos. Além disso, ela abrangerá os métodos de avaliação utilizados pelos sistemas de contagem de pessoas apresentados.

2.4 Bases de Dados e Métodos de Avaliação

Diversos dos trabalhos apresentados comparam seus desempenhos ao avaliarem seus algoritmos numa mesma base de dados e utilizando um mesmo método de avaliação. Isso é importante para possibilitar uma comparação direta entre os métodos.

As conferências *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance* de 2002 e 2009 disponibilizaram bases de vídeos para que os participantes pudessem comparar os resultados. Em 2002, foram feitas seis sequências de vídeos – três para treinamento, totalizando 3145 quadros, e três para teste, totalizando 3709 quadros – de uma mesma câmera que filmou o interior de um shopping center. Já em 2009, a base possui vídeos de ambientes abertos filmados com oito câmeras diferentes (view-001 a view-008). Existem cinco fontes de dados: dados para calibração; dados para treinamento, chamado de S0; base de contagem de pessoas e estimação de densidade, S1; base de rastreamento de pessoas, S2; e base de análise do fluxo reconhecimento de eventos, S3. S0 contém três conjuntos *background*, *city center* e *regular flow*, assim como S1 e S2, que possuem L1, L2 e L3. Já S3 possui dois conjuntos: *event recognition* e *multiple flow*. Cada um desses conjuntos ainda possui alguns

vídeos, gravados em horários diferentes. Além disso, existem regiões marcadas nos vídeos de forma que o cálculo do número de pessoas seja feito dentro de cada uma delas (R0, R1 e R2), como mostra a Figura 2.26.

Figura 2.26: As regiões R0, R1 e R2 da base PETS2009.

Uma outra base de vídeos é a TRECVID08 [SOK06] que contém nove sequências de vídeo capturadas de três câmeras em diferentes locais fechados de um aeroporto. Cada uma das sequências possui 5000 quadros numa resolução de 720x576. Esta base foi utilizada como treinamento em [SHN09].

A base CAVIAR Database [refa] possui 28 vídeos dos quais três são de pessoas andando, seis de pessoas perambulando, quatro de pessoas descansando, caindo ou desmaiando, cinco de pessoas deixando malas para trás, seis de grupos de pessoas se encontrando e grupos de pessoas se dispersando e quatro de pessoas lutando. Esta é uma base que poderá ser utilizada para avaliação dos resultados do método proposto neste trabalho.

A base CVLAB [FBLF08] foi utilizada para testes no artigo de Verstockt et al. [VDBP+09]. Esta base contém uma sequência de vídeo filmada com quatro sensores em ângulos diferentes, com duração de dois minutos e meio, e na qual quatro pessoas andam aleatoriamente por uma sala.

A base UCSD apresentada por Chan et al. [CLV08] foi utilizada por Ryan et al. [RDFS10] para treinar e testar o desempenho do método. Essa base possui milhares de quadros de multidões de 11 a 45 pessoas andando em duas direções.

Cada artigo tem uma forma de avaliar o seu método. Sharma et al. [SHN09] selecionam manualmente dez quadros de cada sequência de vídeo da base PETS2009, geram o *Ground*

Truth visualmente, assim como o número de pessoas visíveis, ou seja não oclusas, e o número de pessoas que o método calculou para fazer a avaliação. Já Sidla et al. [SLBS06] utilizam todos os quadros de uma sequência, calculam a soma acumulada de pessoas e compara com o *Ground Truth*, que é gerado manualmente. Além disso, eles também calculam o erro relativo da câmera de acordo com o quadro. Chan et al. [CMV09] calculam o *Ground Truth* do quinto quadro e utilizam uma interpolação linear para os outros. Com o valor para cada quadro, eles comparam com o valor contado pela abordagem e calculam o Erro Quadrático Médio (MSE (*Mean Squared Error*)) de cada sequência. O PETS2009 também fez uma comparação da precisão do método de alguns autores calculando o erro médio por quadro para cada sequência, como mostra a Figura 2.27. Apesar dessa comparação ter sido feita, não foi disponibilizado o *Ground Truth* oficial da base. Ao analisar a Figura, percebemos que Chan e Albiol obtiveram, no geral, os melhores resultados entre os que foram comparados.

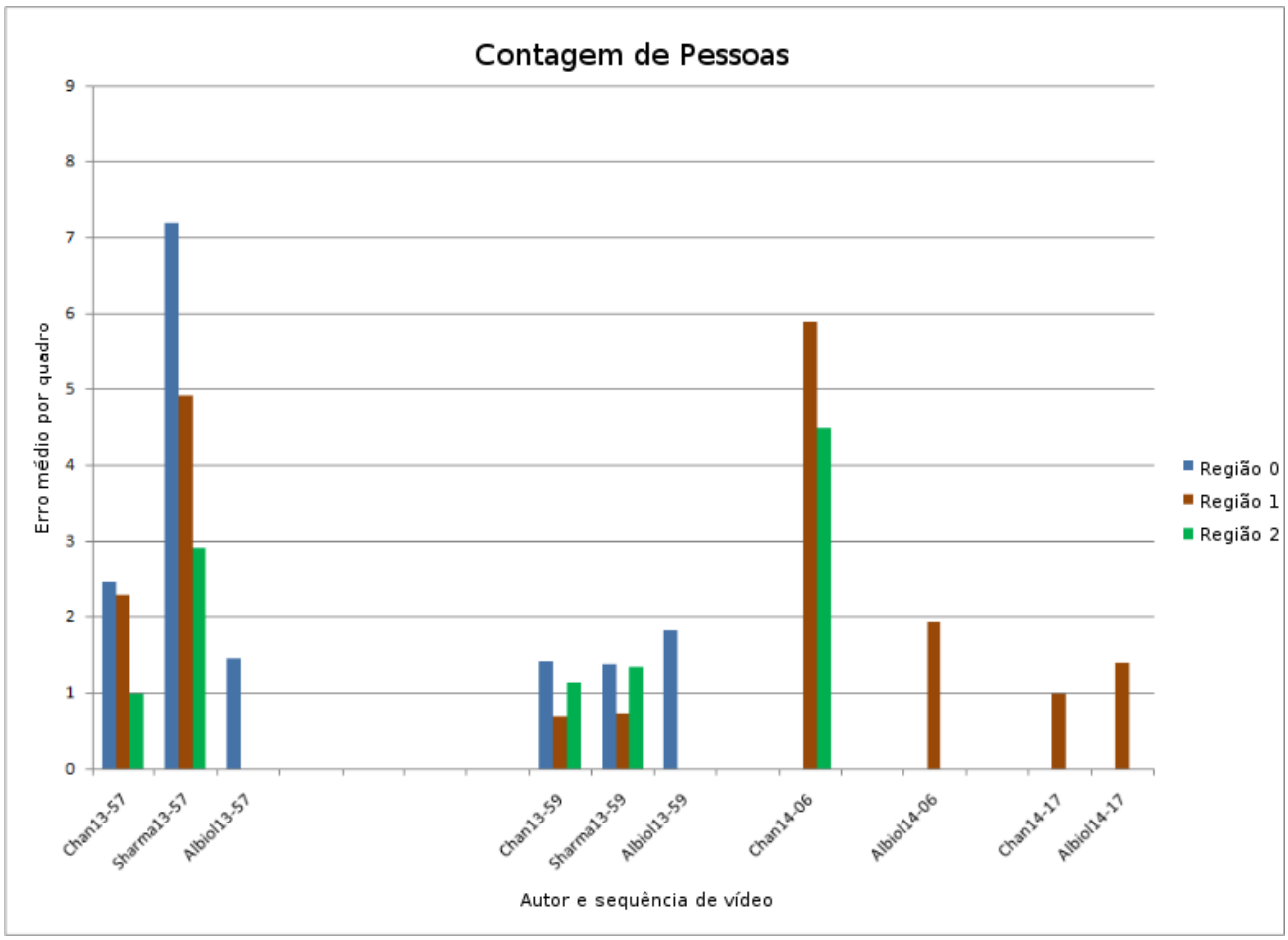


Figura 2.27: Comparação dos resultados do PETS2009.

A última seção deste capítulo compõe a síntese a que chegamos.

2.5 Síntese

Tendo como base todos os artigos apresentados, podemos perceber que não há nenhuma abordagem para contagem de pessoas em sequências de vídeo que lide com todas as adversidades encontradas num sistema de vigilância, principalmente com as oclusões. Além disso, nenhum dos trabalhos estudados utiliza informações de mais de uma câmera da mesma cena e, por isso, não realiza a contagem da cena, mas da visão da única câmera empregada.

Assim, falta um método que trate as oclusões diretamente e que faça contagem do número de pessoas da cena, ao invés de na imagem.

Na próxima subseção, serão apresentadas algumas técnicas candidatas a compor o sistema proposto.

2.5.1 Técnicas Candidatas de Contagem de Pessoas

As abordagens candidatas e seus principais aspectos são apresentados nas subseções a seguir.

2.5.1.1 Sharma et al.

O trabalho desenvolvido por Sharma et al. [SHN09] não utiliza um modelo de fundo para realizar subtração do fundo e detectar as pessoas. Ao invés disso, um algoritmo treinado com características da silhueta dos indivíduos, chamadas de edgelets (Figura 2.2), é utilizado. Dessa forma, os problemas gerados pela variação do fundo não têm grande influência neste método. Além disso, o método é capaz de detectar pessoas estáticas na cena. Entretanto, assim como a detecção, o rastreamento é baseado em aprendizagem, o que cria a necessidade de um treinamento *a priori*.

2.5.1.2 Sidla et al.

Sidla et al. [SLBS06] apresenta um método que seleciona a região de interesse através da movimentação no quadros e busca, utilizando ASM (*Active Shape Model*) [CTCG95] por uma característica chamada Omega Shape, exemplificada na Figura 2.28, que representa o contorno da cabeça, pescoço e ombros das pessoas. Assim, também é possível detectar pessoas paradas.

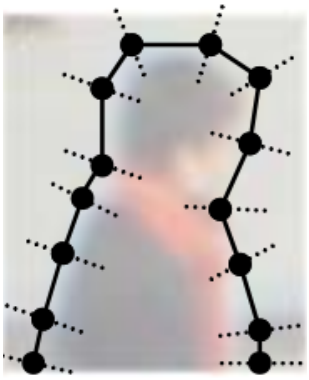


Figura 2.28: Exemplo de Omega Shape [SLBS06].

2.5.1.3 Albiol et al.

O método de Albiol et al. [ASAM09] é probabilístico e não utiliza modelo de fundo, o que é bom por causa da variação de iluminação do ambiente. Os *corner points* são calculados e, a partir do treinamento feito anteriormente, o número de pessoas é estimado. Como a diferenciação entre *corners* dinâmicos e estáticos é feita baseada na movimentação de um quadro para o outro, pessoas paradas não podem ser detectadas.

2.5.1.4 Chan et al.

A abordagem apresentada no trabalho de Chan et al. [CMV09] detecta multidões baseada nas suas texturas e extrai 30 características. Uma normalização de perspectiva, como é mostrado na Figura 2.29, é utilizada para dar pesos diferentes para as características dependendo da distância que elas estão da câmera. A detecção é baseada num treinamento sobre o modelo de texturas.

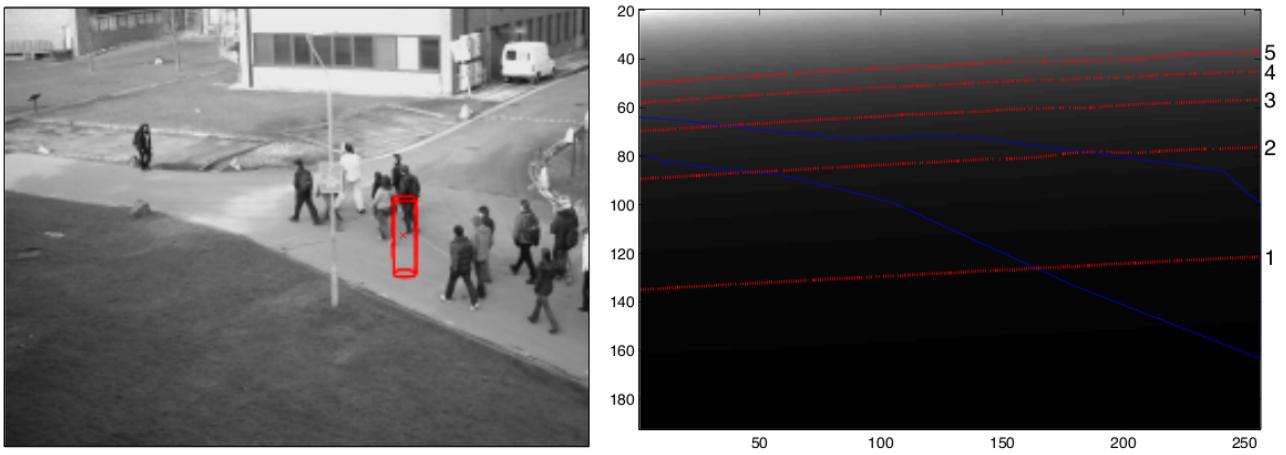


Figura 2.29: A linha azul na imagem da direita representa o caminho onde as pessoas trafegam e as vermelhas os limiares entre os pesos. Neste exemplo, foi criado um objeto (cilindro vermelho) para mostrar em que região as características extraídas dele estão [CMV09].

Capítulo 3

Método Proposto

Como percebemos ao analisar todos os trabalhos apresentados no capítulo anterior, não há um método de contagem de pessoas que realize a contagem sobre a cena, somente sobre as imagens das visões individualmente. Além disso, esses métodos são sempre bastante influenciados por oclusões.

Para definir o sistema proposto neste trabalho, algumas técnicas foram selecionadas e apresentadas na síntese da revisão bibliográfica. Como existem duas abordagens bem distintas, foi definido que dois métodos seriam implementados, um com abordagem indireta – fazendo análise probabilística das multidões – e outro com abordagem direta – contando cada pessoa na cena individualmente. Fazendo isso, é possível comparar o desempenho deles.

O método da abordagem indireta escolhido foi baseado no descrito por Albiol et al. [ASAM09] com pequenas alterações, pois obteve um bom resultado, como mostrado na Figura 2.27. Já o método da abordagem direta escolhido foi uma modificação do método proposto por Sidla et al. [SLBS06]. Ao invés de utilizar ASM para detectar as cabeças, isto será feito através de um classificador SVM, pela facilidade de realizar o treinamento. Além disso, experimentos mostraram que a taxa de classificação correta do SVM é maior em comparação aos resultados de classificadores KNN e RNA (*Rede Neural Artificial*). Em um cenário em que ocorrem diversas oclusões, a chance de uma cabeça ser oclusa é menor do que a de características ao longo do corpo, como usa Sharma et al. [SHN09], por possuir uma área menor.

Para facilitar o problema, foi determinado que seriam usadas somente duas visões das cenas. Entretanto, os métodos desenvolvidos podem ser adaptados para funcionar com mais câmeras.

A primeira seção deste capítulo apresenta a forma de correspondência entre as duas câmeras que será utilizada, ou seja a transformação homográfica, e como foi feito para calculá-la. Em seguida, são apresentados os dois métodos implementados e, finalmente, a última seção trata sobre a forma de avaliação destes métodos.

3.1 Transformação Homográfica

A utilização de duas câmeras baseia-se na hipótese de que é possível diminuir a incidência de oclusões, já que um objeto ocluso em uma visão pode não estar ocluso na outra, e de que elas maximizam as informações da cena.

Para fazer a junção das informações das diferentes visões das sequências de vídeo através de projeção de perspectiva no plano (planta baixa) será feita uma transformação homográfica, que é utilizada em diversos trabalhos, como [SVF09] e [VDBP+09].

Para calcular as matrizes homográficas, é necessário calcular correspondências de pontos do plano de origem, a imagem de uma visão, para o plano de destino, a planta baixa. Essas correspondências foram geradas através das informações de calibração das câmeras disponibilizadas na base PETS2009. Isso é feito transformando pontos 2D da imagem da visão (medidos em pixels) em pontos 3D da cena ou do mundo real (medidos em milímetros). Para calcular um ponto 3D correspondente a um 2D, é necessário definir pelo menos uma de suas coordenadas. Dessa forma, a coordenada Z é definida como 0, pois o plano de destino é o chão, e assim é possível calcular as coordenadas X e Y do ponto da cena. Com um conjunto de n pontos $\{p_1, p_2, \ldots, p_n\}$ no plano de origem e o conjunto de pontos correspondente $\{p'_1, p'_2, \ldots, p'_n\}$, calculamos a homografia H utilizando a fórmula [HZ04]:

$$p'_i = Hp_i \tag{3.1}$$

Considerando os pontos dos planos como coordenadas homogêneas, ou seja $p_i = [x_{p_i}, y_{p_i}, z_{p_i}]$, a relação entre os pontos e a homografia pode ser reescrita como:

$$\begin{bmatrix} x'_{p_i} \\ y'_{p_i} \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_{p_i} \\ y_{p_i} \\ 1 \end{bmatrix} \tag{3.2}$$

Com essa correspondência entre pontos da imagem e pontos da cena, podemos criar uma nova imagem, no caso a planta baixa, ao ignorar a coordenada de altura Z, criar uma proporção de milímetros para pixels e centralizar o ponto de origem da cena na imagem de planta baixa. Para que os pontos onde se encontram as câmeras de ambas as visões ficassem dentro da imagem da planta baixa, foi escolhida a proporção de um pixel para cada 50mm no mundo real e um tamanho de imagem de 600 pixels de altura por 600 pixels de largura. Fazendo isso para as duas visões, é possível criar duas matrizes homográficas que transformam imagens das visões em imagens na planta baixa.

3.2 Métodos Implementados

Nesta seção serão apresentadas duas subseções que comentam as duas abordagens escolhidas para a contagem de pessoas.

3.2.1 *Corner Points* em Duas Visões

A abordagem indireta que foi implementada foi baseada no método dos *corner points* de Albiol et al. [ASAM09], conforme explicado no início do capítulo.

Ela segue o diagrama de blocos apresentado na Figura 3.1.

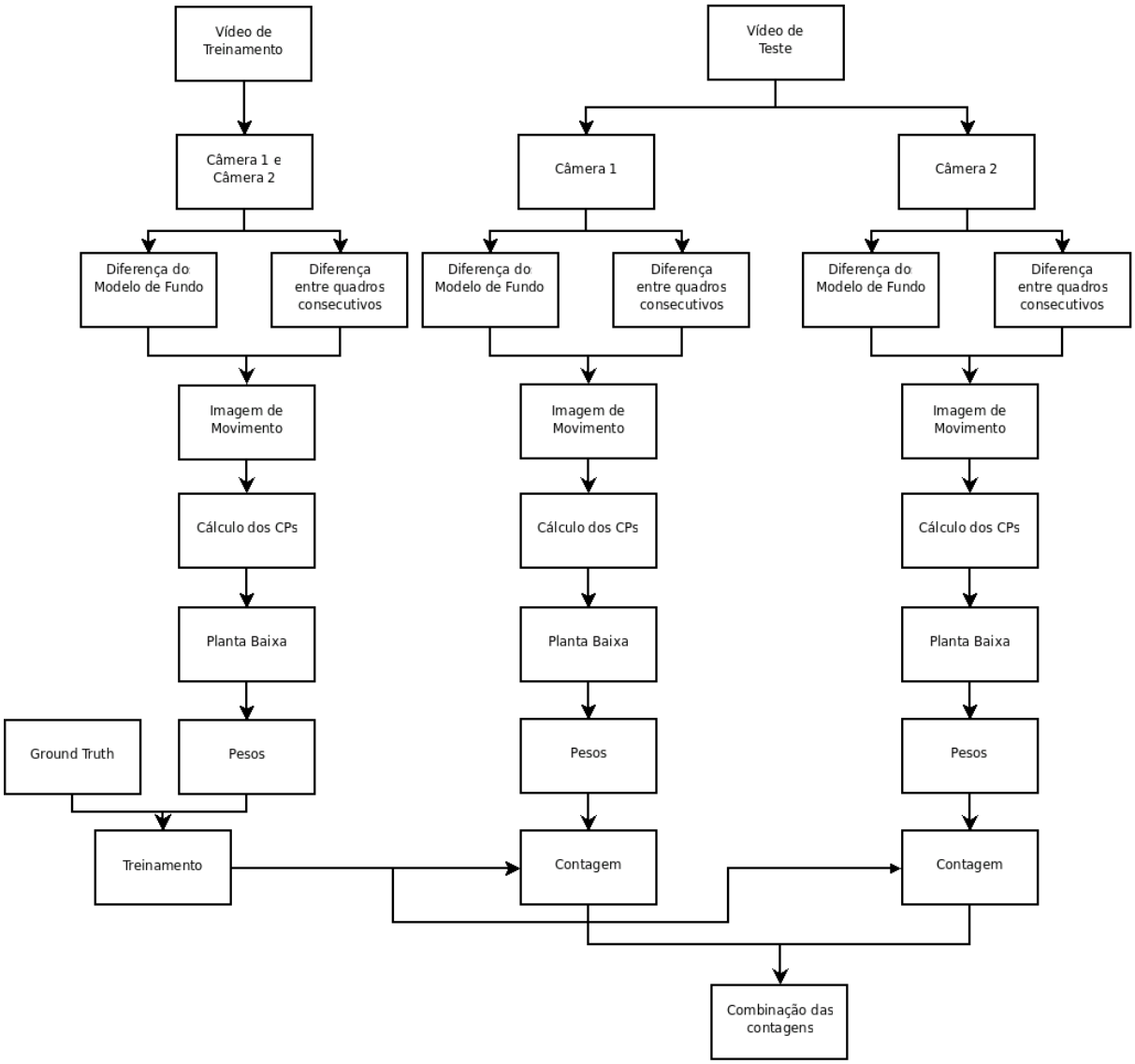


Figura 3.1: Diagrama de blocos de como funciona o método de *Corner Points* em Duas Visões.

Basicamente, para cada visão de um vídeo de treinamento são calculadas as regiões onde existe movimento na cena, diferentemente do método original. Para isso, um modelo de fundo simples (imagem média da base de fundo S0 da base de vídeos PETS2009) é computado. Juntamente com a diferença entre o modelo e as imagens do vídeo, é calculada a diferença entre

o quadro atual e o anterior. Essas duas imagens são combinadas usando um operador AND. Dessa forma, conseguimos eliminar grande parte do ruído causado por diferença de iluminação e continuar com a maior parte dos *blobs* das pessoas. Em seguida, são utilizados os operadores morfológicos de erosão e dilatação – que retira ruído menor e suaviza as bordas dos *blobs* – e um algoritmo de rotulação – que retira ruído que a erosão e dilatação não conseguiram remover por serem grandes demais, mas não grandes o suficiente para serem informação sobre as pessoas na cena, como por exemplo um carro movimentando-se no fundo do vídeo. Ou seja, dados o quadro atual do vídeo $I_{atual}(i,j)$, o quadro anterior $I_{anterior}(i,j)$ e o modelo de fundo $F(i,j)$, obtemos a imagem de movimento $M(i,j)$ da seguinte forma (onde $\wedge$ corresponde à operação binária AND):

$$M(i,j) = (F(i,j) \wedge (I_{atua}l(i,j) - I_{anterior}(i,j))) \tag{3.3}$$

Enquanto, o método original calculava os pontos de interesse na imagem inteira, no método proposto são calculados os *corner points* somente nessa região de movimento, economizando processamento. Para computar os *corners*, pode-se escolher usar uma máscara (região quadrada) ou um *radius* (região circular). A máscara passa pela imagem sendo incrementada pela metade do seu tamanho, ou seja, uma máscara 5 por 5 centrada num pixel X será incrementada para X+2 na próxima iteração. Dentro desta máscara, somente o *corner* com maior magnitude será considerado. A forma com que a máscara se desloca faz com que possam haver *corners* vizinhos. Já ao utilizar o *radius*, este é centrado em cada *corner* detectado e se há algum outro *corner* com magnitude menor que a do *corner* central nessa região, ele será ignorado. Isso garante que a distância mínima entre os *corners* seja do tamanho do *radius*. Para cada pixel (i,j) da imagem atual $I(i,j)$, computamos os gradientes horizontal $G_X(i,j)$ e vertical $G_Y(i,j)$ utilizando o filtro de Sobel [GW02] e a sua magnitude $G(i,j)$ e a matriz de covariância $\sum(i,j)$ (onde $*$ denota a operação de convolução 2D) como:

$$G(i,j) = \sqrt{G_X^2(i,j) + G_Y^2(i,j)} \tag{3.4}$$

$$\sum(i,j) = \begin{bmatrix} G_X^2(i,j) * k(i,j) & (G_X(i,j)G_Y(i,j)) * k(i,j) \\ (G_X(i,j)G_Y(i,j)) * k(i,j) & G_Y^2(i,j) * k(i,j) \end{bmatrix} \tag{3.5}$$

Sendo $A_{max}(i,j)$ o maior autovalor e $A_{min}(i,j)$ o menor autovalor da matriz de covariância, criamos a função discriminante $D(i,j)$:

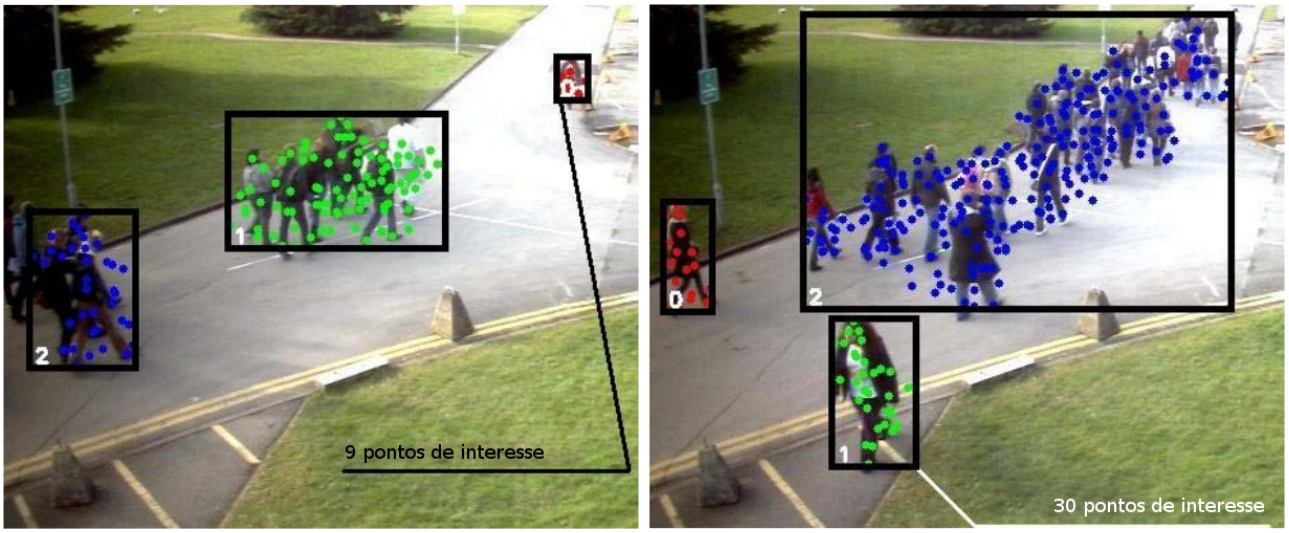
$$D(i,j) = A_{min}(i,j)/A_{max}(i,j) \tag{3.6}$$

Finalmente, se denominarmos a imagem de movimento como $M(i,j)$ e a máscara ou *radius* como $MR(i,j)$, se $M(i,j)$ indica um pixel com movimento, o ponto (i,j) será um *corner*

se obedecer as seguintes restrições e for um máximo local de $MR(i,j)$ (K_D=0,3 e K_G=30):

$$D(i,j) > K_D \wedge G(i,j) > K_G \tag{3.7}$$

Um ponto negativo do trabalho desenvolvido por Albiol et al. [ASAM09] é que ele não considera a perspectiva, ou seja, a distância a que os pontos estão da câmera. Isso é algo que gera erros na contagem, pois uma pessoa longe da câmera gera menos pontos do que quando ela está próxima, como foi citado em [CFP+10] e como mostra a Figura 3.2. Por este motivo, uma melhoria proposta por este trabalho, e inspirada no método de Chan et al. [CMV09], é dar pesos para cada ponto de acordo com a distância que eles estão da câmera.



(a) (b)

Figura 3.2: (a) Pessoa longe da câmera e com 9 pontos de interesse e (b) pessoa próxima a câmera e com 30 pontos de interesse. [CFP+10].

Foram criadas duas abordagens, nas quais os pesos eram dados em quatro regiões e 16 regiões da imagem (como mostra a Figura 3.3) e somente para a Visão 2, que é a que apresenta maior variação do tamanho das pessoas. Entretanto, percebeu-se que era necessário criar uma forma mais verossímil de aplicar os pesos e que a Visão 1 também é afetada por essa variação. Assim, foi criada uma nova forma de calcular os pesos baseada em transformações homográficas, utilizadas nos trabalhos [SVF09] e [VDBP+09]. A forma de calcular os pesos da abordagem de quatro regiões foi empírica, enquanto que a de 16 regiões foi conforme explicado adiante.

Duas matrizes homográficas – da Visão 1 para a planta baixa e da Visão 2 para a planta baixa – foram calculadas. Essas matrizes transformam qualquer ponto de uma das visões num ponto que está no plano do chão da cena e, assim, é possível calcular a distância de qualquer ponto até as câmeras. O ponto $P(i,j)$ no plano da imagem é transformado para o ponto $P'(i,j)$ na planta baixa através da multiplicação pela matriz homográfica H:

$$P'(i,j) = P(i,j)H \tag{3.8}$$

(a) (b)

Figura 3.3: (a) 4 regiões e (b) 16 regiões.

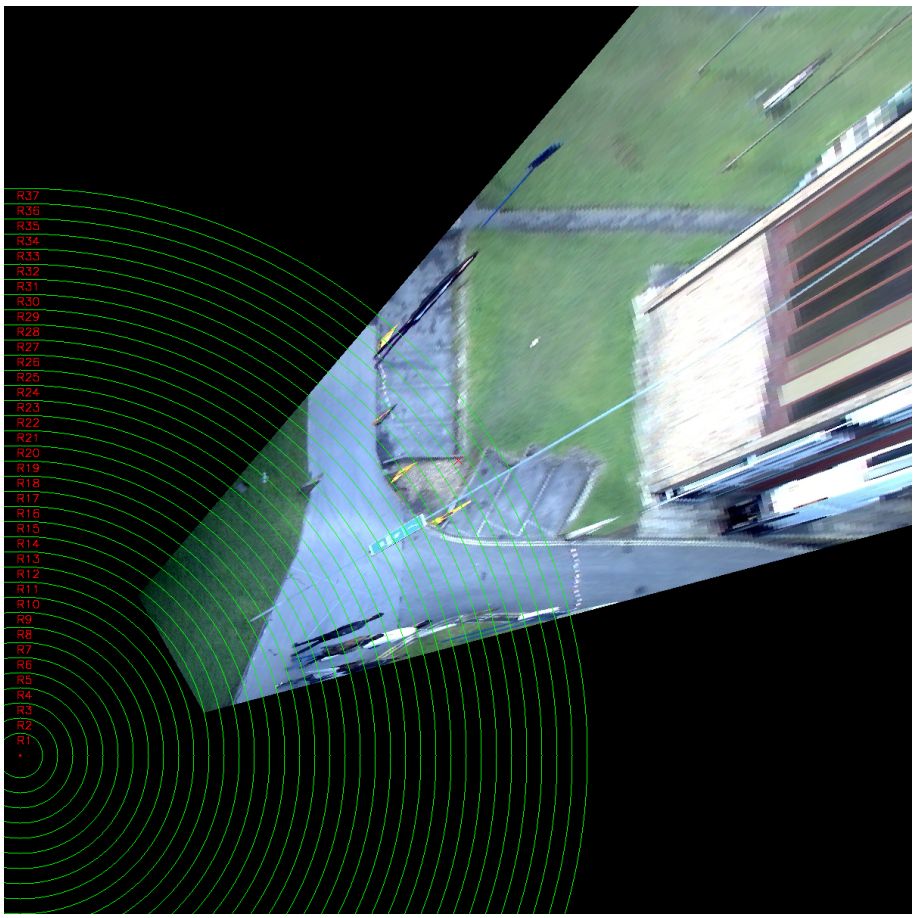
Tendo como calcular a distância de qualquer *corner point* à câmera, criamos 37 regiões circulares com raio aumentando de 20 em 20 pixels (o equivalente a um metro no mundo real), apresentada na Figura 3.4. O tamanho das regiões deve-se ao fato de que um metro é um espaço suficiente para uma pessoa ocupar. Além disso, o número de regiões é 37, pois, utilizando 20 pixels para cada região, esse foi o número de regiões necessárias para cobrir todas as áreas pelas quais as pessoas passam em ambas visões.

Para determinar os pesos de cada região, foi selecionado um trecho de um vídeo no qual é possível calcular a altura de uma pessoa específica em ambas visões, desde que ela entrou na cena até ela sair. A cada uma das 37 regiões é atribuída a média das alturas calculadas que estavam no seu intervalo. Para determinar em qual região a pessoa está, é anotado o pixel do seu pé. Então, é feita uma relação de proporcionalidade entre as alturas médias das regiões e os seus pesos. A região onde se encontra a origem do sistema (coordenadas X=0, Y=0 e Z=0 no mundo) recebe um peso inicial, definido como 1, enquanto que as outras recebem pesos proporcionais comparando as alturas calculadas. Por exemplo, na Visão 1 da abordagem de 37 regiões, a região 35 contém a origem do sistema e, portanto, tem peso igual a 1. Se nela a altura média da pessoa era 85 pixels e na região 27 era 100 pixels, então é feita a proporção **inversa** (inversa pois regiões mais distantes têm pesos maiores e alturas menores que regiões mais próximas):

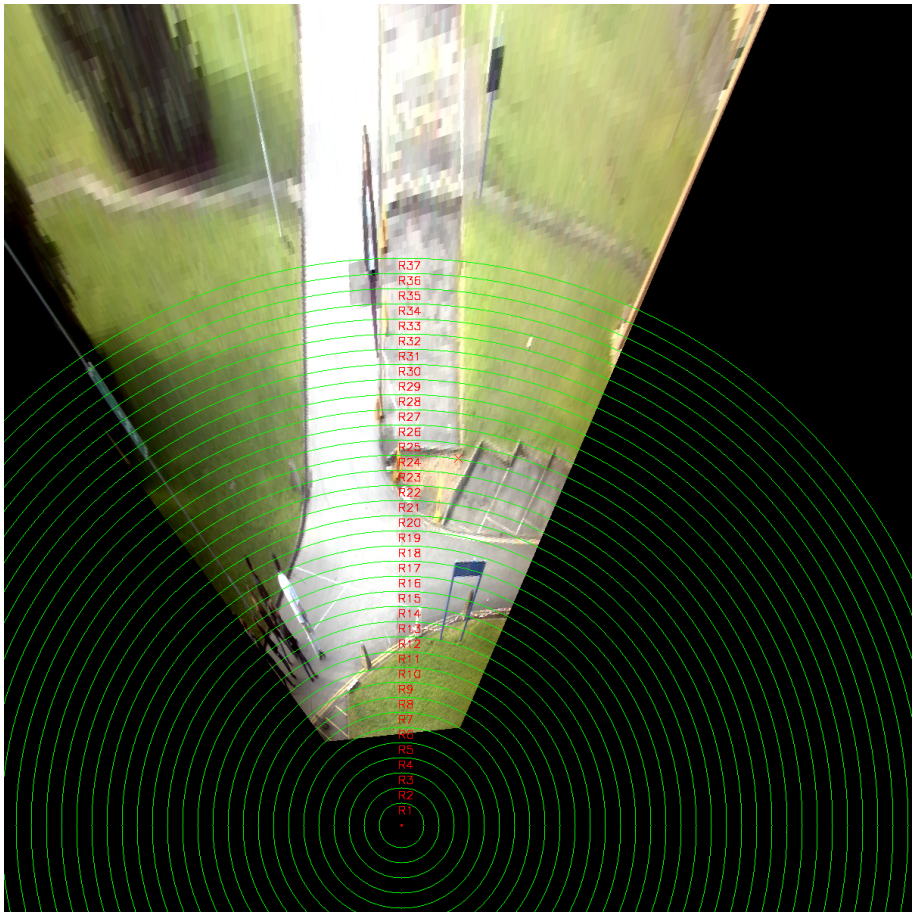
$$\frac{1}{X} = \frac{100}{85}$$

$$X = 0.85$$

Dessa forma foram determinados os pesos de cada uma das regiões. Agora basta aplicar a homografia aos *corner points* e calcular os seus pesos de acordo com a região em que eles estão.



(a)



(b)

Figura 3.4: 37 regiões circulares na planta baixa para (a) Visão 1 e (b) Visão 2.

Entretanto, a homografia é uma transformação de planos e os pontos de interesse das pessoas não estão somente no plano do chão. Por isso, ao aplicar as transformações homográficas a localização dos *corners* são projetadas num local errado, como mostra a Figura 3.5.

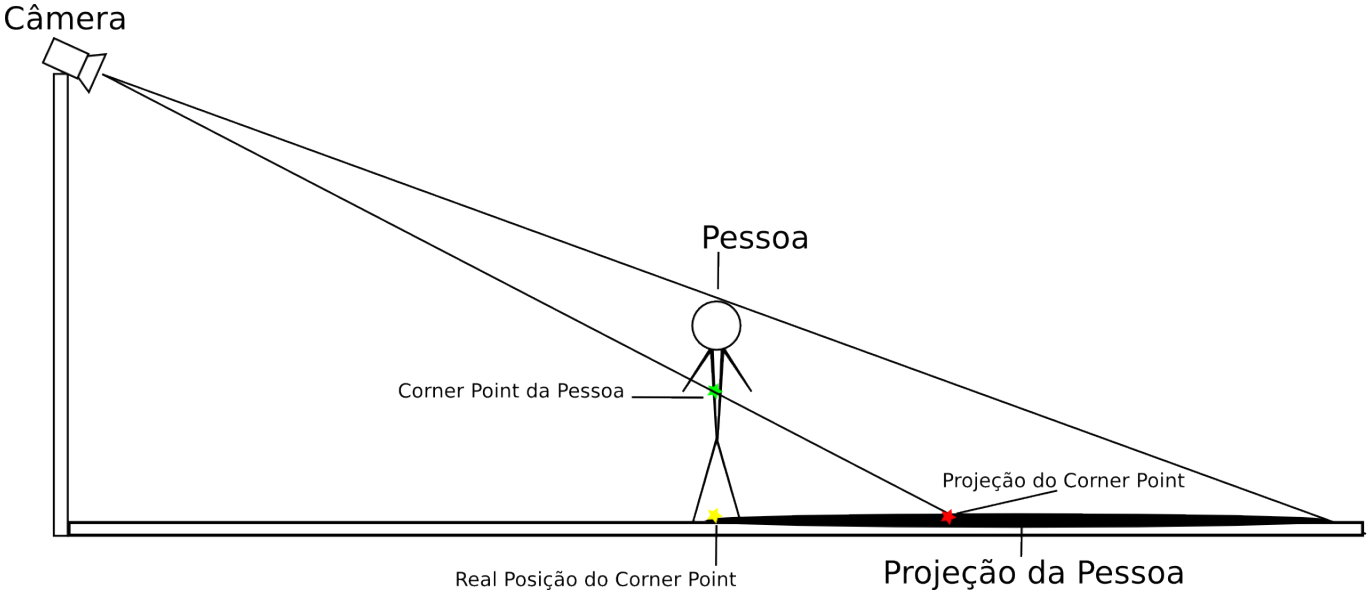


Figura 3.5: Diagrama mostrando como os *corner points* que não estão no chão sofrem alteração na localização da sua projeção.

Para solucionar esse problema, foi proposta uma solução que calcula o tamanho médio das projeções de uma pessoa para cada região. Ou seja, assumimos que todas as pessoas têm a mesma altura e que em uma determinada região sua projeção terá um certo número de pixels. Com isso, para cada *corner* projetado, criamos uma linha entre ele e o ponto no qual encontra-se a câmera que o gerou. Esse *corner* terá sua posição modificada percorrendo essa semi-reta em direção à câmera enquanto duas restrições ocorrerem: o máximo que o *corner* pode ser modificado nessa linha é metade da altura média da projeção da região em que o *corner* está; e ele só pode ser modificado enquanto o novo ponto em que ele for alocado ainda for um ponto em que houve movimentação. Com esse simples algoritmo conseguimos estimar a posição real do *corner* e assim o peso certo será atribuido a ele.

Dados os pesos a todos os pontos, é realizado o seu somatório. Em seguida, é feita a razão entre essa soma e o número de pessoas neste quadro de acordo com o *Ground Truth* da visão. Dessa forma, temos um número médio de *corners* por pessoa deste quadro. Somando o número médio de *corners* por pessoa de todos os quadros do vídeo de treinamento e dividindo o resultado pelo número de quadros, obtemos o número médio de *corners* por pessoa da visão.

Feito esse treinamento para todas as visões, obtemos o número médio de *corners* por pessoas de cada visão. São repetidos, então, todos os procedimentos para um vídeo de teste. A diferença é que nesse vídeo de teste dividimos a soma dos pesos dos pontos de interesse de um quadro pelo número médio de *corners* por pessoa da visão calculada no treinamento. Isso nos resulta na estimativa do número de pessoas que aparecem neste quadro desta visão do vídeo.

Para combinar as informações das duas visões e dar a estimativa final, é calculado o valor máximo, o valor mínimo e valor médio entre as contagens obtidas de cada visão. Esses foram os três testes feitos na combinação das informações.

3.2.2 Detecção de Cabeças em Duas Visões

Inspirado no método do *Omega Shape* de Sidla et al. [SLBS06], decidimos detectar a cabeça das pessoas por ser uma característica que tem uma chance de oclusão menor do que características ao longo do corpo como os *Edgelets* de Sharma et al. [SHN09]. Entretanto, optamos por não utilizar ASM e testar a detecção das cabeças utilizando um classificador SVM e um *Adaboost Perceptron* com *Haar Features*, por eles possuírem uma fase de treinamento mais simples. O método segue o diagrama de blocos da Figura 3.6.

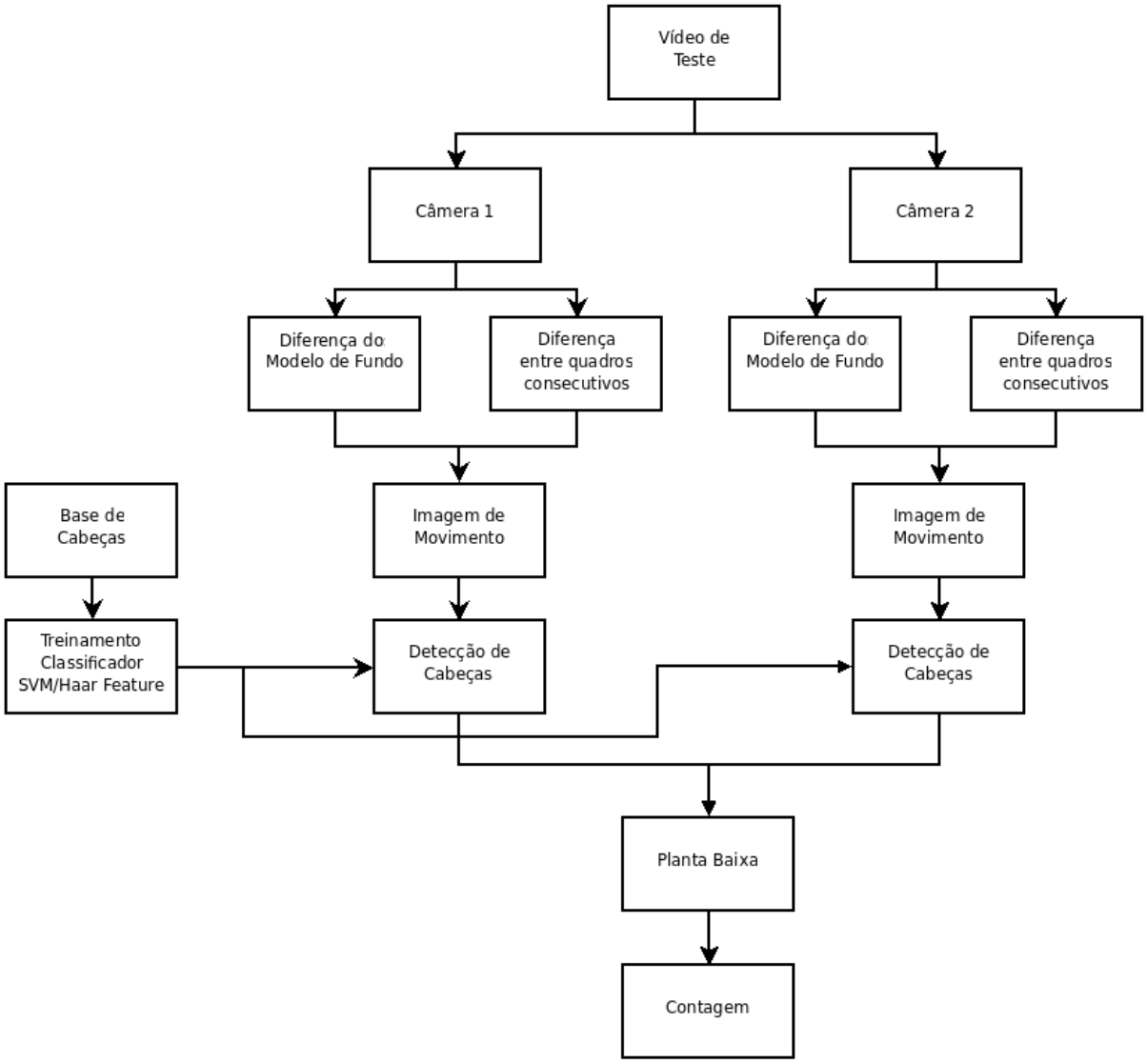


Figura 3.6: Diagrama de blocos de como funciona o método de Detecção de Cabeças em Duas Visões.

O classificador *Adaboost Perceptron* da visão 1 foi treinado com uma base de 2569 ima-

gens de cabeças e 2569 imagens de não-cabeças (imagens com as mesmas dimensões das amostras de cabeça, só que de pernas, bolsas, grama, etc.), além de 3019 imagens negativas (imagens de cenas onde não ha nenhuma cabeça, com dimensões variadas), como mostrado na Figura 3.7 disponibilizadas por um tutorial [refb], nas quais não há cabeças. As dimensões das imagens de cabeça e não-cabeça são de 9 pixels de altura por 9 pixels de largura. Este tamanho foi escolhido pois as cabeças dos vídeos testados variam de 9 por 9 até 25 por 25. A base gerada foi retirada do vídeo S1_L1_Time13-57_view001 da base PETS2009 utilizando um procedimento semi-automático que mostrava as imagens passadas como parâmetro e deixava que o usuário escolhesse um tamanho de janela entre 9 por 9 e 25 por 25. A imagem era rotulada como cabeça ou como não-cabeça pelo usuário. Se as janelas eram diferente de 9 por 9, eram redimensionadas para esta medida. A visão 2 teve seu classificador *Adaboost Perceptron* treinado em uma base de 2358 imagens de cabeças e 853 imagens de não-cabeças – treinadas das mesma forma que as anteriores –, além das mesmas 3019 imagens negativas citadas anteriormente. A base da visão 2 foi retirada do vídeo S1_L1_Time13-57_view002 da base PETS2009. Foram testados classificadores *Adaboost Perceptron* treinados com a taxa de *false alarm* de 0,4 e 0,45.

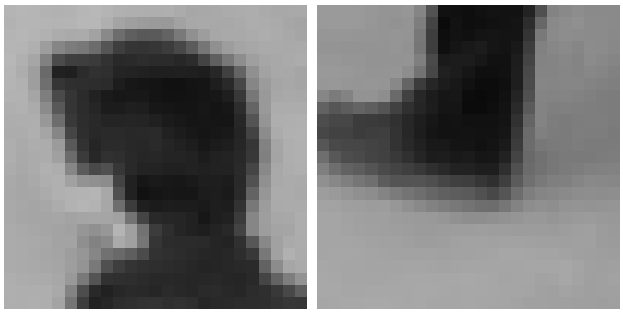
Figura 3.7: Exemplo de imagem negativa da base [refb].

Para o classificador SVM foi utilizada a biblioteca *libsvm* [CL11] e foram utilizadas as mesmas bases de cabeça das visões 1 e 2. Como não era possível utilizar imagens negativas, foi criado um procedimento para aumentar a base de não-cabeças. Além das 2569, para visão 1, e 853, para visão 2, geradas, foram utilizadas mais 11384, para visão 1, e 13368, para visão 2. Essas imagens extras foram colocadas para diminuir a falsa detecção deste classificador. Elas foram geradas da seguinte forma: para cada quadro do vídeo foi detectado onde havia movimento. Foram retiradas manualmente as cabeças da imagem de movimento. Dessas imagens, procurou-se pixels com movimento iterando a busca de 10 em 10. Ao redor desses pixels era aberta e salva uma imagem 9 por 9. O classificador C-SVC foi treinado com o *kernel* de função

de base radial ($exp(-gamma) * |u - v|^2$) após realizar validação cruzada 10-fold para descobrir os melhores valores de C (parâmetro de custo) e gamma (parâmetro para calcular o *kernel* do SVM) para cada base.

Os valores de C e gamma para a visão 1 foram 2 e $0,03125$, respectivamente, enquanto que para a visão 2 foram 8 e $0,03125$. O treino foi feito utilizando um subconjunto das visões 1 e 2 com 1700 e 500 imagens, respectivamente, e testado com 3438 e 1140 imagens. Os resultados foram de 95.4043% e 89.9123%.

A Figura 3.8 mostra um exemplo de imagens de cabeças e não-cabeças, com dimensões de 25 pixels de altura por 25 pixels de largura, retiradas do vídeo S1_L1_Time13-57_view001 da base PETS2009.



(a) (b)

Figura 3.8: (a) Imagem de cabeça e (b) imagem de não-cabeça utilizadas para treinar o classificador SVM.

Assim como no método dos *corner points*, uma imagem de movimento utilizando modelo de fundo e diferença de quadros consecutivos é computada. Para o classificador SVM, nas imagens de ambas visões, ao redor de todo pixel em que houve movimento é aberta uma janela que varia de 9 por 9 até 25 por 25. Essa imagens são redimensionadas para 9 por 9, para ficar do mesmo tamanho que o classificador trata, e um vetor de características (as características são os pixels da imagem, ou seja, numa imagem 9 por 9 existem 81 características), no qual cada característica é um pixel, é extraído delas. Esse vetor é passado para ser classificado pelo SVM treinado. Este processo gera diversas cabeças umas sobre as outras e, por isso, foi criado um algoritmo que agrupa cabeças sobrepostas. Já para o classificador *Adaboost Perceptron*, a imagem inteira é passada para detecção das cabeças de tamanho 9 por 9 até 25 por 25. Em seguida, os pontos que estão em região de movimento são considerados cabeças e os outros ignorados.

As cabeças são projetadas na planta baixa utilizando as transformações homográficas explicadas anteriormente. Da mesma forma que no outro método, existe o problema dos pontos serem projetados no lugar errado. Para resolver isso, a solução adotada é similar. Os pontos que identificam as cabeças têm a sua posição modificada ao longo da semi-reta entre eles e a câmera correspondente. Entretanto, nesse caso a única restrição que acontece é que ele deve percorrer exatamente a altura da projeção de uma pessoa com altura média. Essa projeção é

calculada utilizando proporção de triângulos, como mostra a Figura 3.9. Essa técnica não pode ser utilizada nos método dos *corner points*, pois eles estão espalhados pelo corpo, enquanto que as características deste método sempre estão na cabeça. Dessa forma, as posições dos pontos são corrigidas.

$$\frac{hC}{D} = \frac{hP}{d}$$

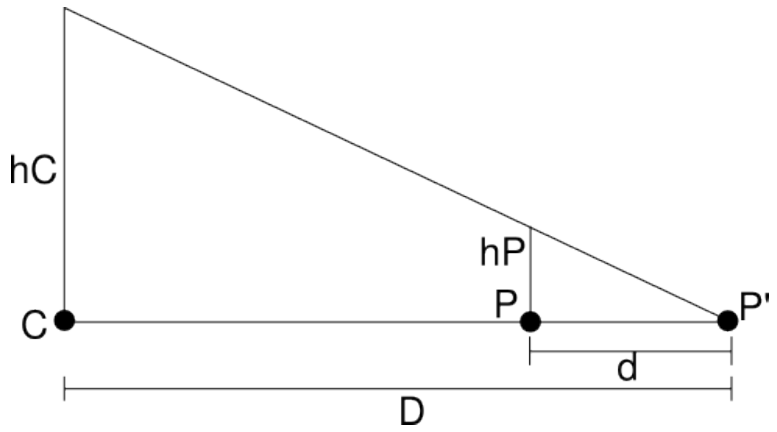


Figura 3.9: Calcula-se a projeção da pessoa (d – a distância entre P e P') utilizando a fórmula acima, onde hC é a altura da câmera que está no ponto C, hP é a altura da pessoa no ponto P e D é a distância entre C e P'.

Com as posições corretas das cabeças, é possível criar correspondências entre as detecções das duas visões. Para encontrar a correspondência de um determinado ponto, abre-se uma máscara que varia de 1 por 1 até 37 por 37 (valor escolhido empiricamente) e se existir um ponto na outra visão nessa região, é criado um ponto correspondente na posição média entre eles. Assumindo que o detector encontra 100% das cabeças na imagem, há três situações em que um ponto não tem correspondência na outra visão:

- Há uma oclusão da cabeça correspondente na outra visão
- Esse ponto é uma falsa detecção
- Esse ponto está numa região da cena que somente uma das visões abrange.

O último caso é mais fácil de ser resolvido, já que basta verificar se ele se encontra em uma dessas regiões. Se for o caso, ele é somado na contagem mesmo não tendo correspondente. Entretanto, é possível que falsos positivos tenham sua posição modificada para uma área dessas, o que acarretaria uma falsa contagem. Definir se ocorreu uma falsa detecção ou oclusão é ainda mais complicado. Para ajudar nessa situação, é útil ter informação temporal do quadros. Se é contada uma pessoa a mais em uma das visões e isso não ocorreu perto de um ponto de entrada, é mais provável que aconteceu uma falsa detecção. Entretanto, há casos de pessoas oclusas de ambas visões durante muito tempo. Isso poderia ser resolvido ao perceber que essa detecção

extra ocorreu em diversos quadros seguidos. Se esse evento ocorre perto da borda é mais seguro dizer que uma nova pessoa entrou no ambiente. Agora, se uma pessoa perdeu sua detecção em uma das câmeras e ela não está perto da borda da cena, é possível que tenha acontecido uma oclusão.

Percebe-se que é complicado decidir o que fazer para corresponder pontos das duas visões. Dessa forma, neste trabalho, pontos sem correspondência, independentemente da região onde estão, são tratados simplesmente como falsas detecções, o que aumenta muito o erro da contagem. Esses problemas serão estudados mais a fundo futuramente.

3.3 Avaliação

Para realizar a avaliação dos métodos propostos, é preciso uma base de dados de vídeos que possua duas visões diferentes da mesma cena. Além disso, é necessário que a base tenha mais de um vídeo, já que é feito treinamento dos algoritmos. A base PETS2009 atende a esses requisitos.

A avaliação dos métodos propostos será feita comparando as estimativas feitas com o *Ground Truth* gerado manualmente (*Ground Truth* é uma medida que nos possibilita avaliar o método, verificando quão perto os resultados chegaram dele. É considerado a contagem correta, seja na visão ou na cena). Assim, será possível calcular o erro médio por quadro, ou seja, a soma das diferenças entre a contagem realizada e o número de pessoas real da cena para todos os quadros do vídeo, dividido pelo número de quadros. Avaliando dessa forma, saberemos o desempenho dos métodos.

Como a quantidade de testes realizados é muito grande, no próximo capítulo só serão apresentados os melhores resultados. Todas as tabelas de dados podem ser conferidas no apêndice.

O próximo capítulo explica os experimentos realizados e os resultados atingidos.

Capítulo 4

Experimentos

Como foi descrito na metodologia, foram definidas duas abordagens que utilizam duas câmeras para realizar a contagem de pessoas de uma cena capturada em sequências de vídeo. Os resultados aqui apresentados mostram a combinação dos melhores parâmetros testados, como foi descrito no capítulo anterior.

Na primeira seção, é apresentada com mais detalhes a forma de avaliação do desempenho dos métodos propostos. A segunda mostra um resumo das tabelas geradas com os melhores resultados da técnica dos *corner points* e uma comparação com o método do qual ela originou. A última seção contém os resultados dos testes feitos para o método de detecção de cabeça.

4.1 Avaliação de Desempenho

A base de vídeos do PETS2009 foi utilizada para testar o desempenho dos métodos propostos, utilizando duas das diversas visões disponibilizadas das mesmas sequências, como mostra a Figura 4.1. As sequências de vídeos dos conjuntos L1 (pessoas andando, multidão de densidade média e tempo nublado) da fonte de dados S1 serão utilizadas para treinamento dos algoritmos e para avaliar o desempenho. Como existem duas sequências de vídeo no conjunto L1, Time13-57 e Time13-59, ao utilizar a primeira para treinamento, os testes são realizados nela mesma e na outra sequência e vice-versa.

Foi realizada uma contagem manual das pessoas de cada quadro de cada uma dessas sequências para ser utilizada como *Ground Truth*. Essa contagem foi feita de duas formas: contagem do número de pessoas na visão – para treinamento do algoritmo dos *corner points* – e contagem do número de pessoas na cena, ou seja, que estão na área comum entre a Visão 1 e a Visão 2 – utilizada para comparar os resultados da contagem feita pelos métodos e definir o erro médio por quadro.

Além disso, quando os testes com imagens das duas visões combinadas começaram, notou-se que muitas vezes os quadros colocados como correspondentes nas duas visões, na

(a) (b)

Figura 4.1: (a) view-001 e (b) view-002 da mesma cena.

realidade, não eram do mesmo instante. Isso fica bastante claro na Figura 4.2. Isso pode ter acontecido porque as câmeras não estavam sincronizadas. No fim, elas capturam em torno de 7 quadros por segundos, mas em algum momento da captura, por exemplo, uma delas pegou 13 imagens em um segundo e apenas uma no segundo segundo, enquanto a outra câmera capturou 7 imagens em cada segundo. Assim a média fica 7 quadros por segundo para ambas câmeras, mas há um intervalo muito grande entre os quadros. Por isso, foi necessário realizar modificações na base de imagens, sincronizando os quadros manualmente. Muitos deles não tinham quadros correspondentes na outra visão, o que fez com que ficássemos com somente 129 quadros dos 221 originais do vídeo S1_L1_Time13-57 e 154 dos 241 no vídeo S1_L1_Time13-59.

(a) (b)

Figura 4.2: (a) Quadro 0 da Visão 1 e (b) Quadro 0 da Visão 2 do vídeo S1_L1_Time13-59. É fácil perceber como a dessincronia é grande nessa parte do vídeo: o rapaz está pisando a linha amarela na Visão 2, enquanto na Visão 1 já está mais pra frente.

Para o método dos *corner points* foram feitos diversos testes modificando diversas variáveis:

- Vídeo de treino: S1_L1_Time13-57 e S1_L1_Time13-59. Quando treinado em um vídeo, o resultado foi gerado testando no outro e também no próprio vídeo de treinamento.
- Número de regiões: sem regiões (ou 1 região), 4 regiões, 16 regiões e 37 regiões (que é o método final).
- Forma de calcular os *corners*: máscara ou *radius*.
- Tamanho da Máscara ou *Radius*: 3x3, 5x5 ou 7x7. Anteriormente também foram testados 1x1, 9x9 e 11x11, mas não forneceram resultados satisfatórios, o que os deixou de fora desses testes.
- No caso de 37 regiões, a altura média das pessoas: 1600, 1650, 1700, 1750 ou 1800 milímetros.

Para o método da Detecção de Cabeças, o treinamento do SVM foi feito usando a base citada na metodologia. Os testes foram realizados nos vídeo S1_L1_Time13-57 e S1_L1_Time13-59.

4.2 *Corner Points* em Duas Visões

Nos resultados abaixo, são apresentados os menores erros médios por quadro (medido em pessoas) para a cena, para a Visão 1 e para a Visão 2, tanto do método proposto quanto do método de Albiol, cujos resultados são os que possuem número de regiões igual a um. Além disso, são informados os seus desvios padrões e parâmetros com os quais esses resultados foram atingidos (Máscara ou *Radius*, Tamanho e, no caso dos resultados para o método proposto, a altura média das pessoas). Nos vídeos testados, a multidão varia de 6 a 42 pessoas.

A Tabela 4.1 mostra os melhores resultados obtidos pelo nosso método e os melhores resultados do método de Albiol et al. [ASAM09] ao treinar com o vídeo S1_L1_13-57 e avaliar sobre o vídeo S1_L1_13-59.

Treino 13_57							
	Altura	Número de Regiões	Visão	Máscara ou Radius	Tamanho	Combinação	Erro Médio por Quadro
CENA	NA	1	V1	Máscara	7x7	NA	3.3922 ±3.3553
	NA	1	V2	Radius	3x3	NA	2.9935 ±4.9577
	1800	37	NA	Máscara	5x5	Valor Médio	2.0326 ±3.1884
V1 Visão	NA	1	V1	Máscara	5x5	NA	1.8693 ±3.6386
	1750	37	V1	Máscara	5x5	NA	1.5098 ±3.0306
V2 Visão	NA	1	V2	Radius	7x7	NA	3.2418 ±5.4656
	1600	37	V2	Radius	7x7	NA	2.6144 ±4.9732

Tabela 4.1: Melhores resultados do método *Corner Points* em Duas Câmeras ao treinar com o vídeo Time13-57.

A Tabela 4.2 mostra os melhores resultados obtidos pelo nosso método e os melhores resultados do método de Albiol et al. [ASAM09] ao treinar com o vídeo S1_L1_13-59 e testar

sobre o vídeo S1_L1_13-57.

Treino 13_59							
	Altura	Número de Regiões	Visão	Máscara ou Radius	Tamanho	Combinação	Erro Médio por Quadro
CENA	NA	1	V1	Radius	7x7	NA	4.3906±4.2629
	NA	1	V2	Máscara	5x5	NA	5.0938±4.3122
	1800	37	NA	Máscara	7x7	Valor Máximo	2.3437±3.4692
V1 Visão	NA	1	V1	Máscara	7x7	NA	1.5156±2.5507
	1650 ou 1700	37	V1	Radius	5x5	NA	1.0469±1.5041
V2 Visão	NA	1	V2	Máscara	5x5	NA	2.8359±3.7039
	1600	37	V2	Radius	5x5	NA	1.9922±3.5264

Tabela 4.2: Melhores resultados do método *Corner Points* em Duas Câmeras ao treinar com o vídeo Time13-59.

Analisando os resultados atingidos por este método, percebemos que a inclusão de perspectiva melhora a contagem. Isso deve-se principalmente aos pesos aplicados aos *corner points*, pois os benefícios da utilização de duas visões em relação à oclusão não é explorada com a forma de junção de informações que é utilizada no método proposto. Dessa forma, a transformação homográfica e as posteriores correções dos pontos na planta baixa são ideais para a fusão de informação das câmeras, mas necessitam de uma política específica de uso dessa informação para que diminuam a ocorrência das oclusões. Os erros médios por quadro atingidos demonstram que o método possui um bom resultado, já que eles ficam em torno de dois em sequências de vídeo com 6 a 42 pessoas. Além disso, através da comparação dos erros médios por quadro e desvios padrões atingidos pelo método proposto com os obtidos pelo método de Albiol, percebemos que o método proposto melhora o método no qual se inspirou.

4.3 Detecção de Cabeças em Duas Visões

Nos resultados abaixo, são apresentados os erros médio por quadro (medido em pessoas) para a cena, para a Visão 1 e para a Visão 2 utilizando o classificador SVM e o *Adaboost Perceptron*. Além disso, são informados os parâmetros com os quais esses resultados foram atingidos (altura média das pessoas e, para o *Adaboost Perceptron*, taxa de *false alarm*).

A Tabela 4.3 mostra os resultados obtidos pelo método desenvolvido para o vídeo S1_L1_13-57 utilizando o classificador SVM.

A Tabela 4.4 mostra os resultados obtidos pelo método desenvolvido para o vídeo S1_L1_13-59 utilizando o classificador SVM.

A Tabela 4.5 mostra os resultados obtidos pelo método desenvolvido para o vídeo S1_L1_13-57 utilizando o classificador *Adaboost Perceptron*.

Teste 13_57				
		Visão 1	Visão 2	Cena
Altura Média	1600	3.046875	3.835938	21.273438
	1650			21.562500
	1700			21.992188
	1750			22.531250
	1800			23.035488

Tabela 4.3: Erro médio por quadro do método Detecção de Cabeças em Duas Visões com SVM para o vídeo S1_L1_13-57.

Teste 13_59				
		Visão 1	Visão 2	Cena
Altura Média	1600	3.117646	3.241830	16.143791
	1650			16.588236
	1700			16.856209
	1750			17.156862
	1800			17.588236

Tabela 4.4: Erro médio por quadro do método Detecção de Cabeças em Duas Visões com SVM para o vídeo S1_L1_13-59.

Teste 13_57					
		False Alarm	Visão 1	Visão 2	Cena
Altura Média	1600	0.40	8.195312	7.960938	21.828125
	1650	0.40			22.078125
	1700	0.40			22.468750
	1750	0.40			22.960938
	1800	0.40			23.609375
	1600	0.45	3.859375	14.046875	19.398438
	1650	0.45			19.820312
	1700	0.45			20.421875
	1750	0.45			21.242188
	1800	0.45			22.000000

Tabela 4.5: Erro médio por quadro do método Detecção de Cabeças em Duas Visões com *Adaboost Perceptron* para o vídeo S1_L1_13-57.

A Tabela 4.6 mostra os resultados obtidos pelo método desenvolvido para o vídeo S1_L1_13-59 utilizando o classificador *Adaboost Perceptron*.

Teste 13_59					
		False Alarm	Visão 1	Visão 2	Cena
Altura Média	1600	0.40	11.588235	11.535948	18.856209
	1650	0.40			18.856209
	1700	0.40			18.836601
	1750	0.40			18.869282
	1800	0.40			18.882353
	1600	0.45	6.980392	4.542484	17.732027
	1650	0.45			17.875816
	1700	0.45			17.921568
	1750	0.45			17.882353
	1800	0.45			18.078432

Tabela 4.6: Erro médio por quadro do método Detecção de Cabeças em Duas Visões com *Adaboost Perceptron* para o vídeo S1_L1_13-59.

Os resultados desse método são claramente ruins. Mesmo os resultados das visões do classificador SVM, que ficaram um pouco acima de três, não são bons pois existem diversos falsos positivos e falsos negativos que se compensam. Já o classificador *Adaboost Perceptron* deixa de detectar diversas cabeças, fazendo com que o erro aumente bastante. Além disso, como a correspondência entre as cabeças nas duas visões implementada ignora cabeças que estão em somente uma das visões e não consegue diferenciar falsos positivos de oclusões, os resultados da cena são ainda piores que a das visões individualmente.

Capítulo 5

Conclusão

Um dos problemas que sistemas de segurança CFTV têm de resolver é contagem do número de pessoas em multidão em sequências de vídeo. Como visto na revisão bibliográfica, os métodos existentes que visam solucionar esse problema realizam a contagem de pessoas na imagem da única visão que utilizam e não na cena em que ela está inserida. Isso faz com que essa contagem não represente corretamente o cenário, onde pode haver pessoas que não são contadas por estarem fora do campo de visão da câmera. Além disso, a oclusão é um obstáculo difícil de ser resolvido e presente em todos os métodos.

Tendo em vista estes problemas, foram desenvolvidos dois métodos, um com abordagem direta e outro com abordagem indireta, para contagem de pessoas em multidão utilizando duas câmeras. Dessa forma, são minimizadas as oclusões e é possível realizar a contagem para a cena e não somente uma imagem dela.

Acerca do método proposto de *Corner Points* em Duas Visões, chegamos à conclusão de que a inclusão da noção de perspectiva melhora a contagem, conforme mostram as tabelas 4.1 e 4.2, já que quanto mais longe o objeto está da câmera, menos pontos de interesse ele terá, mas esses pontos devem ter um peso maior com relação aos pontos mais próximos à câmera para manter a proporção de pontos por pessoa. Concluímos também, ao analisar os resultados dos experimentos, que essa melhoria afeta positivamente tanto a contagem feita sobre a visão quanto a feita sobre a cena.

Ao analisar os resultados dos experimentos com o método de detecção de cabeça em duas câmeras, concluímos que ainda há muito a ser melhorado para que ele obtenha resultados bons tanto para as visões quanto para a cena. Esses resultados seriam taxa de erro médio por quadro abaixo de três e boa taxa de detecção de cabeças para os classificadores desenvolvidos.

Como obtivemos resultados positivos ao utilizar duas visões da cena, chegamos à conclusão de que a técnica para fusão das informações – a transformação homográfica e os algoritmos para corrigir a posição dos pontos – possui boa acurácia.

Com base nos resultados do método dos *corner points*, confirmamos a hipótese de que é possível melhorar a contagem ao utilizar mais de uma câmera. Entretanto, como a combinação

das contagens não considera oclusões, a hipótese de que podemos diminuir a ocorrência delas não foi confirmada ainda.

Como trabalho futuro, existem alguns pontos que devem ser modificados para melhorar os resultados dos métodos propostos.

Primeiramente, os classificadores testados ainda possuem falhas ao criar falsos positivos e deixar de detectar algumas cabeças não oclusas. Melhorá-los ajudaria bastante no resultado final do método de detecção de cabeças. Além disso, algo que traria uma melhoria ainda maior é a modificação da forma de corresponder as cabeças. A inclusão de informação temporal, sobre a qual foi falado durante a descrição do método de detecção de cabeças, deve ajudar a tratar oclusões, falsos positivos e pontos sem correspondentes, elevando a taxa de acerto da contagem.

Outro ponto que poderia ser modificado é o ajuste dos pontos na planta baixa. Atualmente, é assumido um tamanho médio das pessoas. Entretanto, sabemos que a altura das pessoas varia bastante. A criação de uma forma de ajustar os pontos da homografia sem utilizar a altura média diminuiria a diferença entre a posição real dos pontos e a posição estimada, melhorando a contagem.

Além disso, durante este trabalho percebeu-se que a posição das câmeras influenciam bastant os métodos, principalmente se ele possuir transformações homográficas. Algumas conclusões preliminares surgiram, mas elas ainda necessitam de comprovação. Primeiramente, a altura das câmeras deve ser favorável, ou seja, elas devem seguir algumas restrições:

- A distância das câmeras do solo não deve ser muito elevada, senão as projeções das pessoas serão muito pequenas.
- A distância das câmeras do solo não deve ser muito pequena, senão as projeções das pessoas serão muito grandes.

Dessa forma, quando essa distância aproxima-se da altura das pessoas, as projeções geradas tendem ao infinito. Na base PETS2009, a altura das câmeras era boa (entre 5 e 7 metros). Além disso, o ideal é que as câmeras não capturem todo o trajeto por onde as pessoas passam, como ocorre com a Visão 2 da base PETS2009. Nessas situações, há momentos em que as pessoas ainda estão sendo observadas na cena, mas estão muito longe para que suas características sejam extraídas. Por isso, o ideal é posicionar as câmeras como mostra a Figura 5.1.

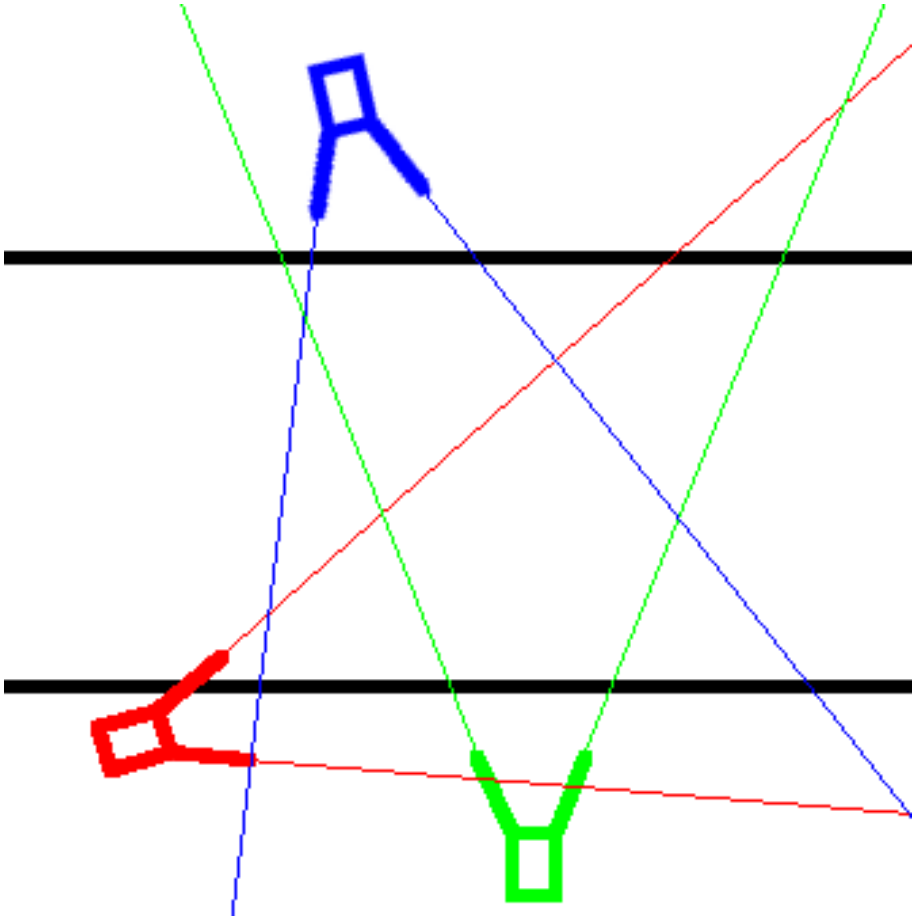


Figura 5.1: As câmeras verde e azul estão posicionadas de forma ideal, enquanto a em vermelho pode gerar problemas para a contagem por acompanhar o trajeto preto.

Referências Bibliográficas

[ASAM09] A. Albiol, M.J. Silla, A. Albiol, and J.M. Mossi. Video analysis using corner motion statistics. In *IEEE International Workshop Performance Evaluation of Tracking and Surveillance*, pages 31–37, Miami, USA, 2009. IEEE Press.

[Avi07] S. Avidan. Ensemble tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:261–271, 2007.

[BETVG08] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, V.110:346–359, 2008.

[CFP+10] D. Conte, P. Foggia, G. Percannella, F. Tufano, and M. Vento. A method for counting people in crowded scenes. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 225–232, Washington, DC, USA, 2010. IEEE Press.

[CGP02] R. Cucchiara, C. Grana, and A. Prati. Detecting moving objects and their shadows - an evaluation with the PETS2002 dataset. In *IEEE International Workshop Performance Evaluation of Tracking and Surveillance*, pages 18–25, Copenhagen, Denmark, 2002. IEEE Press.

[CGS02] R.T. Collins, R. Gross, and J. Shi. Silhouette-based human identification from body shape and gait. In *Proceedings of IEEE Conference on Face and Gesture Recognition*, pages 351–356, Pittsburgh, USA, 2002. IEEE Press.

[CL11] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[CLV08] A.B. Chan, Z.-S. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, 2008.

[CMV09] A.B. Chan, M. Morrow, and N. Vasconcelos. Analysis of crowded scenes using holistic properties. In *IEEE International Workshop Performance Evaluation of Tracking and Surveillance*, pages 101–106, Miami, Florida, 2009. IEEE Press.

[CTCG95] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, V.61:38–59, 1995.

[CV08] A.B. Chan and N. Vasconcelos. Modelling, clustering and segmenting video with mixtures of dynamic textures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, V.30(5):909–926, 2008.

[DSG09] K. Dimitropoulos, T. Semertzidis, and N. Grammalidis. Video and signal based surveillance for airport applications. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 170–175, Genova, Italy, 2009. IEEE Press.

[DT05] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 886–893, Washington, DC, USA, 2005. IEEE Press.

[FBLF08] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, V.30:pp.267–282, 2008.

[GW02] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Prentice Hall, Boston, Massachusetts, 2002.

[HF01] I. Haritaoglu and M. Flickner. Detection and tracking of shopping groups in stores. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 431–438, San Jose, USA, 2001. IEEE Press.

[HSS04] L. Havasi, Z. SzlÁvik, and T. SzirÁnyi. Pedestrian detection using derived third-order symmetry of legs: A novel method of motion-based information extraction from video image-sequences. In *International Conference on Computer Vision*, pages 733–739, Warsaw, Poland, 2004.

[HZ04] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, USA, 2 edition, 2004.

[KCCK02] J. Kim, K. Choi, B. Choi, and S. Ko. Real-time vision-based people counting system for the security door. In *IEEE International Technical Conference On*

Circuits Systems Computers and Communications, Phuket, Thailand, 2002. IEEE Press.

[LDT03] L. Lee, G. Dalley, and K. Tieu. Learning pedestrian models for silhouette refinement. In *IEEE Conference on Computer Vision*, pages 663–670, Nice, France, 2003. IEEE Press.

[MAT10] D. Merad, K.E. Aziz, and N. Thome. Fast people counting using head detection from skeleton graph. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 151–156, Washington, DC, USA, 2010. IEEE Press.

[MMR02] L. Marcenaro, L. Marchesotti, and C.S. Regazzoni. Tracking and counting multiple interacting people in indoor scenes. In *IEEE International Workshop Performance Evaluation of Tracking and Surveillance*, pages 56–61, Copenhagen, Denmark, 2002. IEEE Press.

[PCMT01] A. Prati, R. Cucchiara, I. Mikic, and M.M. Trivedi. Analysis and detection of shadows in video streams: A comparative evaluation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2:571–576, 2001.

[PEP98] C. Papageorgiou, T. Evgeniou, and T. Poggio. A trainable pedestrian detection system. In *Proceedings of Intelligent Vehicles*, pages 241–246, 1998.

[PES10] M. PÄtzold, R.H. Evangelio, and T. Sikora. Counting people in crowded environments by fusion of shape and motion information. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 157–164, Washington, DC, USA, 2010. IEEE Press.

[RDFS10] D. Ryan, S. Denman, C. Fookes, and S. Sridharan. Crowd counting using group tracking and local features. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 218–224, Washington, DC, USA, 2010. IEEE Press.

[refa] Caviar database. Internet. Acessado em Outubro/2009.

[refb] Tutorial: Opencv haartraining (rapid object detection with a cascade of boosted classifiers based on haar-like features) – negative database. Internet. Acessado em Junho/2011.

[RTG98] Y. Rubner, C. Tomasi, and L.J. Guibas. A metric for distributions with applications to image databases. In *IEEE International Conference on Computer Vision*, pages 59–66, Washington, DC, USA, 1998. IEEE Press.

[SA85] S. Suzuki and K. Abe. Topological structural analysis of digital binary images by border following. *Computer Vision, Graphics and Image Processing*, V.30:pp.32–46, 1985.

[Sas10] M.A. Sasse. Not seeing the crime for the cameras? *Communications of the ACM*, 53(2):pp.22–25, 2010.

[SHN09] P.K. Sharma, C. Huang, and R. Nevatia. Evaluation of people tracking, counting and density estimation in crowded environments. In *IEEE International Workshop Performance Evaluation of Tracking and Surveillance*, pages 39–46, Miami, USA, 2009. IEEE Press.

[SLBS06] O. Sidla, Y. Lypetskyy, N. Brändle, and S. Seer. Pedestrian detection and tracking for counting applications in crowded situations. In *IEEE International Conference on Video and Signal Based Surveillance*, pages 70–76, Sydney, Australia, 2006. IEEE Press.

[SOK06] A.F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, Santa Barbara, USA, 2006.

[SVF09] L. Snidaro, I. Visentini, and G.L. Foresti. Multi-sensor multi-cue fusion for object detection in video surveillance. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 364–369, Washington, DC, USA, 2009. IEEE Press.

[SWG00] C. Stauffer, Eric W., and L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, V.22:pp.747–757, 2000.

[TCB04] D. Tsishkou, L. Chen, and E. Bovbel. Semi-automatic face segmentation for face detection in video, 2004. Internacional Conference on Intelligent Access to Multimedia Documents on the Internet.

[Tek95] A.M. Tekalp. *Digital Video Processing*. Prentice Hall, 1995.

[TK91] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical report, Carnegie Mellon University, 1991.

[VDBP+09] S. Verstockt, S. De Bruyne, C. Poppe, P. Lambert, and R. Van de Walle. Multi-view object localization in h.264/avc compressed domain. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 370–374, Washington, DC, USA, 2009. IEEE Press.

[VJ07] J.D. Valle Jr. Contagem Automática de Pessoas em Cenas de Vídeo Usando Visão Computacional. Master's thesis, Pontifícia Universidade Católica do Paraná – PUCPR, 2007.

[VJS03] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *IEEE Conference on Computer Vision*, pages 734–741, Nice, France, 2003. IEEE Press.

[WHH+09] Z. Wu, N.I. Hristov, T.L. Hedrick, T.H. Kunz, and M. Betke. Tracking a large number of objects from multiple views. In *International Conference on Computer Vision*, pages 1546–1553, Kyoto, Japan, 2009.

[YJS06] A. Yilmar, O. Javed, and M. Shah. Object tracking: A survey. *ACM Computing Surveys*, 38, 2006.

[ZDC09] X. Zhao, E. DellandrÉa, and L. Chen. A people counting system based on face detection and tracking in a video. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 67–72, Washington, DC, USA, 2009. IEEE Press.

Apêndice A

Resultados dos Testes do Método de *Corner Points* em Duas Visões

Esta seção apresenta todos os resultados obtidos nos testes descritos na seção sobre Avaliação do Desempenho na Metodologia. Os resultados que têm número de regiões igual a 1 correspondem aos resultados da implementação da abordagem de Albiol et al. Os resultados que têm número de regiões 4 e 16 foram testes intermediários até chegar ao método final, que tem 37 regiões. Nas tabelas a seguir, estão destacados os melhores resultados, apresentados como erro médio por quadro, da seguinte forma:

- Para a implementação de Albiol et al. [ASAM09]:
  - Vermelho: melhor resultado da Visão 1 usando *Ground Truth* da cena.
  - Verde: melhor resultado da Visão 2 usando *Ground Truth* da cena.
  - Amarelo: melhor resultado da Visão 1 usando *Ground Truth* da visão.
  - Azul: melhor resultado da Visão 2 usando *Ground Truth* da visão.
- Para o método proposto, de 37 regiões:
  - Vermelho: melhor resultado da combinação da Visão 1 com Visão 2 usando *Ground Truth* da cena.
  - Amarelo: melhor resultado da Visão 1 usando *Ground Truth* da visão.
  - Azul: melhor resultado da Visão 2 usando *Ground Truth* da visão.

Dessa forma, as tabelas apresentadas no capítulo de Resultados contêm esses dados destacados.

Treino 13_57								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	47	1,4922	4,8906	1,8954	3,4575
V2	NA	1	NA	26	2,4375	4,4609	3,6078	3,2353
V1+V2	NA	1 – 1	Valor máximo	47 – 26	-	3,8203	-	3,1830
			Valor mínimo		-	5,5312	-	3,5098
			Valor médio		-	4,7968	-	2,6928
V2	NA	4	NA	43	4,0547	4,4219	5,6797	4,9281
V1+V2	NA	1 – 4	Valor máximo	47 – 43	-	2,7343	-	4,9607
			Valor mínimo		-	6,5781	-	3,4248
			Valor médio		-	4,1796	-	2,4771
V2	NA	16	NA	48	2,2344	3,7109	3,0327	2,5294
V1+V2	NA	1 – 16	Valor máximo	47 – 48	-	2,9140	-	2,5490
			Valor mínimo		-	5,6875	-	3,4379
			Valor médio		-	4,3593	-	2,3137
V1	1600	37	NA	34	1,1016	4,5469	1,6405	3,0065
V2	1600	37	NA	18	2,0234	3,0469	3,2418	2,7124
V1+V2	1600	37 – 37	Valor máximo	34 – 18	-	2,8203	-	2,6535
			Valor mínimo		-	4,7734	-	3,0653
			Valor médio		-	3,8125	-	2,1372
V1	1650	37	NA	34	1,1172	4,5625	1,6275	3,0458
V2	1650	37	NA	18	2,0703	3,0938	3,2222	2,6928
V1+V2	1650	37 – 37	Valor máximo	34 – 18	-	2,8750	-	2,6339
			Valor mínimo		-	4,7812	-	3,1045
			Valor médio		-	3,8515	-	2,1176
V1	1700	37	NA	34	1,1250	4,5703	1,6078	3,0784
V2	1700	37	NA	18	2,0391	3,0313	3,2680	2,6863
V1+V2	1700	37 – 37	Valor máximo	34 – 18	-	2,8125	-	2,6209
			Valor mínimo		-	4,7890	-	3,1437
			Valor médio		-	3,8359	-	2,1307
V1	1750	37	NA	33	1,1250	3,9922	1,8758	2,8889
V2	1750	37	NA	18	2,0859	3,0469	3,2941	2,6993
V1+V2	1750	37 – 37	Valor máximo	33 – 18	-	2,6406	-	2,6339
			Valor mínimo		-	4,3984	-	2,9542
			Valor médio		-	3,5312	-	2,1633
V1	1800	37	NA	33	1,1328	4,0156	1,8562	2,9085
V2	1800	37	NA	18	2,1719	2,9922	3,3529	2,7059
V1+V2	1800	37 – 37	Valor máximo	33 – 18	-	2,5703	-	2,6405
			Valor mínimo		-	4,4375	-	2,9738
			Valor médio		-	3,4921	-	2,1045

Tabela A.1: Tabela de Resultados: Máscara 3x3. Treinamento S1_L1_13-57.

Treino 13_59								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	49	1,8125	5,5234	1,7974	3,8170
V2	NA	1	NA	29	3,6719	6,3672	3,1176	3,1765
V1+V2	NA	1 – 1	Valor máximo	49 – 29	-	5,0468	-	2,8169
			Valor mínimo		-	6,8437	-	4,1764
			Valor médio		-	6,1328	-	3,1568
V2	NA	4	NA	57	6,7656	9,3047	3,6078	3,9150
V1+V2	NA	1 – 4	Valor máximo	49 – 57	-	4,7109	-	2,8888
			Valor mínimo		-	10,1171	-	4,8431
			Valor médio		-	7,5703	-	3,2352
V2	NA	16	NA	53	3,3438	5,8047	2,1765	2,3137
V1+V2	NA	1 – 16	Valor máximo	49 – 53	-	4,2968	-	2,2287
			Valor mínimo		-	7,0312	-	3,9019
			Valor médio		-	5,8046	-	2,7973
V1	1600	37	NA	36	1,9297	5,7656	1,3268	3,6732
V2	1600	37	NA	19	2,1016	3,9063	2,6144	2,2810
V1+V2	1600	37 – 37	Valor máximo	36 – 19	-	3,6796	-	2,2026
			Valor mínimo		-	5,9921	-	3,7516
			Valor médio		-	4,9453	-	2,4771
V1	1650	37	NA	36	1,9063	5,7734	1,3529	3,6863
V2	1650	37	NA	19	2,1172	3,9219	2,6209	2,2484
V1+V2	1650	37 – 37	Valor máximo	36 – 19	-	3,6953	-	2,1633
			Valor mínimo		-	6,0000	-	3,7712
			Valor médio		-	4,9609	-	2,4771
V1	1700	37	NA	36	1,9453	5,8125	1,3660	3,7124
V2	1700	37	NA	20	2,6484	4,8906	2,2026	2,1569
V1+V2	1700	37 – 37	Valor máximo	36 – 20	-	4,3906	-	2,0522
			Valor mínimo		-	6,3125	-	3,8169
			Valor médio		-	5,5703	-	2,7450
V1	1750	37	NA	36	1,9609	5,8281	1,3595	3,7320
V2	1750	37	NA	20	2,7266	4,9688	2,2353	2,1895
V1+V2	1750	37 – 37	Valor máximo	36 – 20	-	4,3906	-	2,0915
			Valor mínimo		-	6,4062	-	3,8300
			Valor médio		-	5,6093	-	2,7320
V1	1800	37	NA	36	1,9766	5,8438	1,3595	3,7451
V2	1800	37	NA	20	2,7109	4,9219	2,2876	2,1765
V1+V2	1800	37 – 37	Valor máximo	36 – 20	-	4,3437	-	2,0849
			Valor mínimo		-	6,4218	-	3,8366
			Valor médio		-	5,5859	-	2,7124

Tabela A.2: Tabela de Resultados: Máscara 3x3. Treinamento S1_L1_13-59.

Treino 13_57								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	47	1,6641	4,7344	2,1699	3,5621
V2	NA	1	NA	26	2,6094	4,7734	3,3791	2,9935
V1+V2	NA	1 – 1	Valor máximo	47 – 26	-	3,9609	-	2,9346
			Valor mínimo		-	5,5468	-	3,6209
			Valor médio		-	4,8750	-	2,7516
V2	NA	4	NA	42	3,7734	4,1406	5,5948	4,8039
V1+V2	NA	1 – 4	Valor máximo	47 – 42	-	2,4453	-	4,8627
			Valor mínimo		-	6,4296	-	3,5032
			Valor médio		-	3,8906	-	2,3856
V2	NA	16	NA	47	2,1172	3,5938	3,1503	2,5686
V1+V2	NA	1 – 16	Valor máximo	47 – 47	-	2,7812	-	2,5816
			Valor mínimo		-	5,5468	-	3,5490
			Valor médio		-	4,1718	-	2,3202
V1	1600	37	NA	35	1,0234	4,5625	1,6601	3,1961
V2	1600	37	NA	18	1,9688	2,8672	3,3268	2,7059
V1+V2	1600	37 – 37	Valor máximo	35 – 18	-	2,6953	-	2,6470
			Valor mínimo		-	4,7343	-	3,2549
			Valor médio		-	3,7109	-	2,2156
V1	1650	37	NA	35	1,0313	4,5547	1,6405	3,2026
V2	1650	37	NA	18	1,9453	2,8750	3,3529	2,7059
V1+V2	1650	37 – 37	Valor máximo	35 – 18	-	2,6953	-	2,6470
			Valor mínimo		-	4,7343	-	3,2614
			Valor médio		-	3,7500	-	2,1830
V1	1700	37	NA	35	1,0547	4,5781	1,6471	3,2353
V2	1700	37	NA	18	2,0000	2,8672	3,3922	2,7059
V1+V2	1700	37 – 37	Valor máximo	35 – 18	-	2,6562	-	2,6470
			Valor mínimo		-	4,7890	-	3,2941
			Valor médio		-	3,7734	-	2,1699
V1	1750	37	NA	34	0,9766	4,0469	1,8105	3,0196
V2	1750	37	NA	18	2,0078	2,8594	3,3856	2,6993
V1+V2	1750	37 – 37	Valor máximo	34 – 18	-	2,5156	-	2,6339
			Valor mínimo		-	4,3906	-	3,0849
			Valor médio		-	3,5546	-	2,1437
V1	1800	37	NA	34	0,9609	4,0625	1,8301	3,0392
V2	1800	37	NA	18	1,9844	2,8359	3,4510	2,7255
V1+V2	1800	37 – 37	Valor máximo	34 – 18	-	2,4453	-	2,6535
			Valor mínimo		-	4,4531	-	3,1111
			Valor médio		-	3,5312	-	2,1437

Tabela A.3: Tabela de Resultados: *Radius* 3x3. Treinamento S1_L1_13-57.

Treino 13_59								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	49	1,8125	5,3359	2,0131	3,8758
V2	NA	1	NA	28	3,3906	5,9922	2,9804	3,0000
V1+V2	NA	1 – 1	Valor máximo	49 – 28	-	4,8203	-	2,7712
			Valor mínimo		-	6,5078	-	4,1045
			Valor médio		-	5,7968	-	3,0588
V2	NA	4	NA	56	6,7266	9,3125	3,3595	3,6928
V1+V2	NA	1 – 4	Valor máximo	49 – 56	-	4,5390	-	2,7385
			Valor mínimo		-	10,1093	-	4,8300
			Valor médio		-	7,3515	-	3,2483
V2	NA	16	NA	52	3,0938	5,6172	2,2288	2,2222
V1+V2	NA	1 – 16	Valor máximo	49 – 52	-	4,0937	-	2,2091
			Valor mínimo		-	6,8593	-	3,8888
			Valor médio		-	5,5859	-	2,8431
V1	1600	37	NA	36	1,3516	5,1094	1,5490	3,5033
V2	1600	37	NA	20	2,3438	4,6953	2,2288	2,1438
V1+V2	1600	37 – 37	Valor máximo	36 – 20	-	4,0468	-	2,0196
			Valor mínimo		-	5,7578	-	3,6274
			Valor médio		-	5,1718	-	2,6405
V1	1650	37	NA	36	1,3594	5,1172	1,5294	3,4967
V2	1650	37	NA	20	2,3828	4,7188	2,2288	2,1307
V1+V2	1650	37 – 37	Valor máximo	36 – 20	-	4,0625	-	2,0065
			Valor mínimo		-	5,7734	-	3,6209
			Valor médio		-	5,1640	-	2,6405
V1	1700	37	NA	36	1,3594	5,1172	1,5425	3,5359
V2	1700	37	NA	20	2,4219	4,7422	2,2353	2,1111
V1+V2	1700	37 – 37	Valor máximo	36 – 20	-	4,0468	-	2,0065
			Valor mínimo		-	5,8125	-	3,6405
			Valor médio		-	5,1953	-	2,6078
V1	1750	37	NA	36	1,3672	5,1250	1,5359	3,5294
V2	1750	37	NA	20	2,5000	4,8047	2,2549	2,1176
V1+V2	1750	37 – 37	Valor máximo	36 – 20	-	4,0390	-	2,0130
			Valor mínimo		-	5,8906	-	3,6339
			Valor médio		-	5,2031	-	2,5882
V1	1800	37	NA	36	1,3828	5,1406	1,5294	3,5359
V2	1800	37	NA	20	2,5547	4,8281	2,2941	2,1176
V1+V2	1800	37 – 37	Valor máximo	36 – 20	-	4,0234	-	2,0196
			Valor mínimo		-	5,9453	-	3,6339
			Valor médio		-	5,2031	-	2,5751

Tabela A.4: Tabela de Resultados: *Radius* 3x3. Treinamento S1_L1_13-59.

Treino 13_57								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	25	1,4609	4,8125	1,8693	3,6013
V2	NA	1	NA	13	2,6641	4,1094	4,0261	3,5621
V1+V2	NA	1 – 1	Valor máximo	25 – 13	-	3,8437	-	3,4967
			Valor mínimo		-	5,0781	-	3,6666
			Valor médio		-	4,5390	-	2,7843
V2	NA	4	NA	22	3,5469	3,9297	5,5229	4,8105
V1+V2	NA	1 – 4	Valor máximo	25 – 22	-	2,6718	-	4,8496
			Valor mínimo		-	6,0703	-	3,5620
			Valor médio		-	4,0000	-	2,5359
V2	NA	16	NA	25	2,1406	3,7578	2,8366	2,4510
V1+V2	NA	1 – 16	Valor máximo	25 – 25	-	3,1718	-	2,5032
			Valor mínimo		-	5,3984	-	3,5490
			Valor médio		-	4,3515	-	2,4771
V1	1600	37	NA	18	1,1328	4,2656	1,5359	3,1634
V2	1600	37	NA	9	2,5859	2,6250	3,6601	3,0000
V1+V2	1600	37 – 37	Valor máximo	18 – 9	-	2,5546	-	2,9281
			Valor mínimo		-	4,3359	-	3,2352
			Valor médio		-	3,4062	-	2,1437
V1	1650	37	NA	18	1,1406	4,3203	1,5229	3,1765
V2	1650	37	NA	9	2,5234	2,6094	3,6471	2,9869
V1+V2	1650	37 – 37	Valor máximo	18 – 9	-	2,5390	-	2,9150
			Valor mínimo		-	4,3906	-	3,2483
			Valor médio		-	3,4375	-	2,1307
V1	1700	37	NA	18	1,1406	4,3516	1,5163	3,1830
V2	1700	37	NA	9	2,4844	2,5391	3,6536	2,9673
V1+V2	1700	37 – 37	Valor máximo	18 – 9	-	2,4765	-	2,9019
			Valor mínimo		-	4,4140	-	3,2483
			Valor médio		-	3,3906	-	2,1241
V1	1750	37	NA	18	1,1406	4,3672	1,5098	3,2026
V2	1750	37	NA	9	2,4766	2,4688	3,6601	2,9608
V1+V2	1750	37 – 37	Valor máximo	18 – 9	-	2,4062	-	2,9084
			Valor mínimo		-	4,4296	-	3,2549
			Valor médio		-	3,3984	-	2,1307
V1	1800	37	NA	17	1,4531	3,3047	2,0784	2,7255
V2	1800	37	NA	9	2,4766	2,4688	3,7386	3,0131
V1+V2	1800	37 – 37	Valor máximo	17 – 9	-	2,2734	-	2,9673
			Valor mínimo		-	3,5000	-	2,7712
			Valor médio		-	2,8593	-	2,0326

Tabela A.5: Tabela de Resultados: Máscara 5x5. Treinamento S1_L1_13-57.

Treino 13_59								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	26	1,7813	5,4297	1,7778	3,9281
V2	NA	1	NA	14	2,8359	5,0938	3,3529	3,1503
V1+V2	NA	1 – 1	Valor máximo	26 – 14	-	4,6562	-	3,0392
			Valor mínimo		-	5,8671	-	4,0392
			Valor médio		-	5,4140	-	2,9607
V2	NA	4	NA	29	6,3828	8,9688	3,5817	3,8627
V1+V2	NA	1 – 4	Valor máximo	26 – 29	-	4,7500	-	2,9803
			Valor mínimo		-	9,6484	-	4,8104
			Valor médio		-	7,2734	-	3,2941
V2	NA	16	NA	27	2,8516	5,3125	2,2614	2,3072
V1+V2	NA	1 – 16	Valor máximo	26 – 27	-	4,2578	-	2,2941
			Valor mínimo		-	6,4843	-	3,9411
			Valor médio		-	5,5234	-	2,8496
V1	1600	37	NA	18	1,1328	4,2656	1,5359	3,1634
V2	1600	37	NA	10	2,1797	4,1250	2,3922	2,2810
V1+V2	1600	37 – 37	Valor máximo	18 – 10	-	3,5546	-	2,1699
			Valor mínimo		-	4,8359	-	3,2745
			Valor médio		-	4,3359	-	2,4640
V1	1650	37	NA	18	1,1406	4,3203	1,5229	3,1765
V2	1650	37	NA	10	2,1797	4,1094	2,4052	2,2288
V1+V2	1650	37 – 37	Valor máximo	18 – 10	-	3,5859	-	2,1241
			Valor mínimo		-	4,8437	-	3,2810
			Valor médio		-	4,3437	-	2,4640
V1	1700	37	NA	18	1,1406	4,3516	1,5163	3,1830
V2	1700	37	NA	10	2,2109	4,1250	2,4118	2,2092
V1+V2	1700	37 – 37	Valor máximo	18 – 10	-	3,5937	-	2,1176
			Valor mínimo		-	4,8828	-	3,2745
			Valor médio		-	4,3906	-	2,4248
V1	1750	37	NA	18	1,1406	4,3672	1,5098	3,2026
V2	1750	37	NA	10	2,2656	4,1641	2,4118	2,2222
V1+V2	1750	37 – 37	Valor máximo	18 – 10	-	3,5937	-	2,1437
			Valor mínimo		-	4,9375	-	3,2810
			Valor médio		-	4,4218	-	2,4509
V1	1800	37	NA	18	1,1328	4,3906	1,5163	3,2353
V2	1800	37	NA	10	2,2344	4,1328	2,4183	2,1765
V1+V2	1800	37 – 37	Valor máximo	18 – 10	-	3,5859	-	2,1241
			Valor mínimo		-	4,9375	-	3,2875
			Valor médio		-	4,4218	-	2,3856

Tabela A.6: Tabela de Resultados: Máscara 5x5. Treinamento S1_L1_13-59.

Treino 13_57								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	31	1,6406	4,8828	2,0131	3,7712
V2	NA	1	NA	16	2,8047	4,3906	3,6471	3,2092
V1+V2	NA	1 – 1	Valor máximo	31 – 16	-	4,0546	-	3,1307
			Valor mínimo		-	5,2187	-	3,8496
			Valor médio		-	4,7343	-	2,8104
V2	NA	4	NA	27	3,2813	3,9453	4,8562	4,1830
V1+V2	NA	1 – 4	Valor máximo	31 – 27	-	2,5546	-	4,1960
			Valor mínimo		-	6,2734	-	3,7581
			Valor médio		-	4,1093	-	2,3660
V2	NA	16	NA	30	2,0391	3,4688	2,9020	2,5033
V1+V2	NA	1 – 16	Valor máximo	31 – 30	-	2,9921	-	2,5098
			Valor mínimo		-	5,3593	-	3,7647
			Valor médio		-	4,2578	-	2,5359
V1	1600	37	NA	22	1,0703	3,7344	1,9608	2,9477
V2	1600	37	NA	12	1,9922	3,2656	2,7647	2,4183
V1+V2	1600	37 – 37	Valor máximo	22 – 12	-	2,9218	-	2,3006
			Valor mínimo		-	4,0781	-	3,0653
			Valor médio		-	3,6015	-	2,3333
V1	1650	37	NA	22	1,0781	3,7422	1,9542	2,9281
V2	1650	37	NA	12	2,0234	3,3125	2,8105	2,4379
V1+V2	1650	37 – 37	Valor máximo	22 – 12	-	2,9531	-	2,3202
			Valor mínimo		-	4,1015	-	3,0457
			Valor médio		-	3,6484	-	2,3006
V1	1700	37	NA	22	1,0703	3,7656	1,9542	2,9412
V2	1700	37	NA	12	2,0625	3,3047	2,8235	2,3987
V1+V2	1700	37 – 37	Valor máximo	22 – 12	-	2,9296	-	2,2941
			Valor mínimo		-	4,1406	-	3,0457
			Valor médio		-	3,6562	-	2,2614
V1	1750	37	NA	22	1,0625	3,7734	1,9150	2,9673
V2	1750	37	NA	12	2,0234	3,2813	2,8235	2,4118
V1+V2	1750	37 – 37	Valor máximo	22 – 12	-	2,9062	-	2,3006
			Valor mínimo		-	4,1484	-	3,0784
			Valor médio		-	3,6562	-	2,2483
V1	1800	37	NA	22	1,0781	3,7891	1,8889	2,9935
V2	1800	37	NA	12	2,0156	3,2891	2,7974	2,3333
V1+V2	1800	37 – 37	Valor máximo	22 – 12	-	2,8906	-	2,2418
			Valor mínimo		-	4,1875	-	3,0849
			Valor médio		-	3,6640	-	2,1960

Tabela A.7: Tabela de Resultados: *Radius* 5x5. Treinamento S1_L1_13-57.

Treino 13_59								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	31	1,6406	4,8828	2,0131	3,7712
V2	NA	1	NA	17	2,9688	5,1953	3,1242	3,0000
V1+V2	NA	1 – 1	Valor máximo	31 – 17	-	4,5078	-	2,8888
			Valor mínimo		-	5,5703	-	3,8823
			Valor médio		-	5,1484	-	3,0130
V2	NA	4	NA	34	5,7422	8,2813	3,2614	3,5294
V1+V2	NA	1 – 4	Valor máximo	31 – 34	-	4,0859	-	2,7189
			Valor mínimo		-	9,0781	-	4,5816
			Valor médio		-	6,5390	-	3,0522
V2	NA	16	NA	32	2,3906	4,6641	2,3725	2,2745
V1+V2	NA	1 – 16	Valor máximo	31 – 32	-	3,6328	-	2,2549
			Valor mínimo		-	5,9140	-	3,7908
			Valor médio		-	4,8515	-	2,7908
V1	1600	37	NA	23	1,0547	4,4688	1,6078	3,3137
V2	1600	37	NA	12	1,9922	3,2656	2,7647	2,4183
V1+V2	1600	37 – 37	Valor máximo	23 – 12	-	3,0703	-	2,3137
			Valor mínimo		-	4,6640	-	3,4183
			Valor médio		-	3,9375	-	2,4248
V1	1650	37	NA	23	1,0469	4,4922	1,5948	3,3268
V2	1650	37	NA	12	2,0234	3,3125	2,8105	2,4379
V1+V2	1650	37 – 37	Valor máximo	23 – 12	-	3,1250	-	2,3333
			Valor mínimo		-	4,6796	-	3,4313
			Valor médio		-	4,0156	-	2,3921
V1	1700	37	NA	23	1,0469	4,4922	1,5686	3,3268
V2	1700	37	NA	12	2,0625	3,3047	2,8235	2,3987
V1+V2	1700	37 – 37	Valor máximo	23 – 12	-	3,0937	-	2,3006
			Valor mínimo		-	4,7031	-	3,4248
			Valor médio		-	3,9843	-	2,3725
V1	1750	37	NA	23	1,0781	4,5391	1,5621	3,3333
V2	1750	37	NA	12	2,0234	3,2813	2,8235	2,4118
V1+V2	1750	37 – 37	Valor máximo	23 – 12	-	3,0703	-	2,3071
			Valor mínimo		-	4,7500	-	3,4379
			Valor médio		-	4,0000	-	2,3660
V1	1800	37	NA	23	1,0781	4,5703	1,5490	3,3725
V2	1800	37	NA	12	2,0156	3,2891	2,7974	2,3333
V1+V2	1800	37 – 37	Valor máximo	23 – 12	-	3,0546	-	2,2352
			Valor mínimo		-	4,8046	-	3,4705
			Valor médio		-	4,0312	-	2,3464

Tabela A.8: Tabela de Resultados: *Radius* 5x5. Treinamento S1_L1_13-59.

Treino 13_57								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	14	1,5156	4,6328	2,0784	3,3922
V2	NA	1	NA	7	3,2500	3,5547	4,7516	4,1961
V1+V2	NA	1 – 1	Valor máximo	14 – 7	-	3,4531	-	4,0915
			Valor mínimo		-	4,7343	-	3,4967
			Valor médio		-	4,0234	-	2,8300
V2	NA	4	NA	12	3,3203	3,3125	6,1765	5,3987
V1+V2	NA	1 – 4	Valor máximo	14 – 12	-	2,5625	-	5,4313
			Valor mínimo		-	5,3828	-	3,3594
			Valor médio		-	3,4531	-	2,8104
V2	NA	16	NA	13	2,3594	2,4141	3,9608	3,2614
V1+V2	NA	1 – 16	Valor máximo	14 – 13	-	2,3984	-	3,2810
			Valor mínimo		-	4,6484	-	3,3725
			Valor médio		-	3,4140	-	2,3986
V1	1600	37	NA	10	1,3359	3,9844	1,9216	2,8170
V2	1600	37	NA	5	2,9453	2,4844	3,9542	3,2549
V1+V2	1600	37 – 37	Valor máximo	10 – 5	-	2,4843	-	3,1960
			Valor mínimo		-	3,9843	-	2,8758
			Valor médio		-	3,0156	-	2,1437
V1	1650	37	NA	10	1,3359	4,0000	1,9020	2,8366
V2	1650	37	NA	5	2,8828	2,4688	4,0131	3,3268
V1+V2	1650	37 – 37	Valor máximo	10 – 5	-	2,4453	-	3,2679
			Valor mínimo		-	4,0234	-	2,8954
			Valor médio		-	3,0390	-	2,1437
V1	1700	37	NA	10	1,3438	4,0078	1,9085	2,8431
V2	1700	37	NA	5	2,8828	2,4844	4,0261	3,3137
V1+V2	1700	37 – 37	Valor máximo	10 – 5	-	2,4765	-	3,2549
			Valor mínimo		-	4,0156	-	2,9019
			Valor médio		-	3,0078	-	2,1307
V1	1750	37	NA	10	1,3359	4,0313	1,8693	2,8693
V2	1750	37	NA	5	2,8750	2,4453	4,0784	3,3529
V1+V2	1750	37 – 37	Valor máximo	10 – 5	-	2,4375	-	3,3006
			Valor mínimo		-	4,0390	-	2,9215
			Valor médio		-	3,0156	-	2,1241
V1	1800	37	NA	10	1,3516	4,0469	1,8562	2,8824
V2	1800	37	NA	5	2,8281	2,3359	4,0915	3,3660
V1+V2	1800	37 – 37	Valor máximo	10 – 5	-	2,3437	-	3,3137
			Valor mínimo		-	4,0390	-	2,9346
			Valor médio		-	2,9531	-	2,0915

Tabela A.9: Tabela de Resultados: Máscara 7x7. Treinamento S1_L1_13-57.

Treino 13_59								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	14	1,5156	4,6328	2,0784	3,3922
V2	NA	1	NA	8	2,8672	5,0938	3,3791	3,2810
V1+V2	NA	1 – 1	Valor máximo	14 – 8	-	4,2968	-	3,0588
			Valor mínimo		-	5,4296	-	3,6143
			Valor médio		-	5,0000	-	2,9215
V2	NA	4	NA	16	6,0000	8,5234	3,5621	3,7647
V1+V2	NA	1 – 4	Valor máximo	14 – 16	-	4,1796	-	2,8496
			Valor mínimo		-	8,9765	-	4,3071
			Valor médio		-	6,3984	-	2,8431
V2	NA	16	NA	15	2,6172	4,9219	2,4314	2,4379
V1+V2	NA	1 – 16	Valor máximo	14 – 15	-	3,8125	-	2,4117
			Valor mínimo		-	5,7421	-	3,4183
			Valor médio		-	4,7890	-	2,5882
V1	1600	37	NA	10	1,3359	3,9844	1,9216	2,8170
V2	1600	37	NA	5	2,9453	2,4844	3,9542	3,2549
V1+V2	1600	37 – 37	Valor máximo	10 – 5	-	2,4843	-	3,1960
			Valor mínimo		-	3,9843	-	2,8758
			Valor médio		-	3,0156	-	2,1437
V1	1650	37	NA	10	1,3359	4,0000	1,9020	2,8366
V2	1650	37	NA	5	2,8828	2,4688	4,0131	3,3268
V1+V2	1650	37 – 37	Valor máximo	10 – 5	-	2,4453	-	3,2679
			Valor mínimo		-	4,0234	-	2,8954
			Valor médio		-	3,0390	-	2,1437
V1	1700	37	NA	10	1,3438	4,0078	1,9085	2,8431
V2	1700	37	NA	5	2,8828	2,4844	4,0261	3,3137
V1+V2	1700	37 – 37	Valor máximo	10 – 5	-	2,4765	-	3,2549
			Valor mínimo		-	4,0156	-	2,9019
			Valor médio		-	3,0078	-	2,1307
V1	1750	37	NA	10	1,3359	4,0313	1,8693	2,8693
V2	1750	37	NA	5	2,8750	2,4453	4,0784	3,3529
V1+V2	1750	37 – 37	Valor máximo	10 – 5	-	2,4375	-	3,3006
			Valor mínimo		-	4,0390	-	2,9215
			Valor médio		-	3,0156	-	2,1241
V1	1800	37	NA	10	1,3516	4,0469	1,8562	2,8824
V2	1800	37	NA	5	2,8281	2,3359	4,0915	3,3660
V1+V2	1800	37 – 37	Valor máximo	10 – 5	-	2,3437	-	3,3137
			Valor mínimo		-	4,0390	-	2,9346
			Valor médio		-	2,9531	-	2,0915

Tabela A.10: Tabela de Resultados: Máscara 7x7. Treinamento S1_L1_13-59.

Treino 13_57								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	20	1,8359	4,3906	2,2680	3,5948
V2	NA	1	NA	11	3,0781	5,2266	3,2418	3,0261
V1+V2	NA	1 – 1	Valor máximo	20 – 11	-	4,4375	-	2,8888
			Valor mínimo		-	5,1796	-	3,7320
			Valor médio		-	4,9062	-	3,0196
V2	NA	4	NA	17	2,7578	3,1563	5,1046	4,3268
V1+V2	NA	1 – 4	Valor máximo	20 – 17	-	2,3593	-	4,3529
			Valor mínimo		-	5,1875	-	3,5686
			Valor médio		-	3,3671	-	2,5098
V2	NA	16	NA	20	2,0156	3,9297	2,5556	2,2745
V1+V2	NA	1 – 16	Valor máximo	20 – 20	-	3,3671	-	2,3398
			Valor mínimo		-	4,9531	-	3,5294
			Valor médio		-	4,1875	-	2,6470
V1	1600	37	NA	15	1,1641	3,6719	1,8301	2,9085
V2	1600	37	NA	8	2,1563	3,3984	2,6144	2,2941
V1+V2	1600	37 – 37	Valor máximo	15 – 8	-	3,0546	-	2,1960
			Valor mínimo		-	4,0156	-	3,0065
			Valor médio		-	3,6250	-	2,3137
V1	1650	37	NA	15	1,1406	3,6953	1,8170	2,9216
V2	1650	37	NA	8	2,1172	3,4375	2,6797	2,2941
V1+V2	1650	37 – 37	Valor máximo	15 – 8	-	3,0859	-	2,1895
			Valor mínimo		-	4,0468	-	3,0261
			Valor médio		-	3,6406	-	2,2810
V1	1700	37	NA	15	1,1328	3,7031	1,7974	2,9673
V2	1700	37	NA	8	2,1016	3,4375	2,6667	2,2157
V1+V2	1700	37 – 37	Valor máximo	15 – 8	-	3,0859	-	2,1372
			Valor mínimo		-	4,0546	-	3,0457
			Valor médio		-	3,6484	-	2,2614
V1	1750	37	NA	15	1,0938	3,7422	1,7909	2,9739
V2	1750	37	NA	8	2,1328	3,3594	2,6405	2,1895
V1+V2	1750	37 – 37	Valor máximo	15 – 8	-	3,0312	-	2,1176
			Valor mínimo		-	4,0703	-	3,0457
			Valor médio		-	3,6250	-	2,2679
V1	1800	37	NA	15	1,1016	3,7500	1,7974	2,9935
V2	1800	37	NA	8	2,1328	3,3750	2,6667	2,2549
V1+V2	1800	37 – 37	Valor máximo	15 – 8	-	3,0312	-	2,1830
			Valor mínimo		-	4,0937	-	3,0653
			Valor médio		-	3,6171	-	2,2549

Tabela A.11: Tabela de Resultados: *Radius* 7x7. Treinamento S1_L1_13-57.

Treino 13_59								
	Altura	Número de Regiões	Combinação	Num Cps/Pessoa Medio	Erro 13_57 visao	Erro 13_57 cena	Erro 13_59 visao	Erro 13_59 cena
V1	NA	1	NA	20	1,8359	4,3906	2,2680	3,5948
V2	NA	1	NA	11	3,0781	5,2266	3,2418	3,0261
V1+V2	NA	1 – 1	Valor máximo	20 – 11	-	4,4375	-	2,8888
			Valor mínimo		-	5,1796	-	3,7320
			Valor médio		-	4,9062	-	3,0196
V2	NA	4	NA	21	4,7344	7,2266	3,1438	3,3333
V1+V2	NA	1 – 4	Valor máximo	20 – 21	-	3,6718	-	2,8235
			Valor mínimo		-	7,9453	-	4,1045
			Valor médio		-	5,5390	-	2,7581
V2	NA	16	NA	20	2,0156	3,9297	2,5556	2,2745
V1+V2	NA	1 – 16	Valor máximo	20 – 20	-	3,3671	-	2,3398
			Valor mínimo		-	4,9531	-	3,5294
			Valor médio		-	4,1875	-	2,6470
V1	1600	37	NA	15	1,1641	3,6719	1,8301	2,9085
V2	1600	37	NA	8	2,1563	3,3984	2,6144	2,2941
V1+V2	1600	37 – 37	Valor máximo	15 – 8	-	3,0546	-	2,1960
			Valor mínimo		-	4,0156	-	3,0065
			Valor médio		-	3,6250	-	2,3137
V1	1650	37	NA	15	1,1406	3,6953	1,8170	2,9216
V2	1650	37	NA	8	2,1172	3,4375	2,6797	2,2941
V1+V2	1650	37 – 37	Valor máximo	15 – 8	-	3,0859	-	2,1895
			Valor mínimo		-	4,0468	-	3,0261
			Valor médio		-	3,6406	-	2,2810
V1	1700	37	NA	15	1,1328	3,7031	1,7974	2,9673
V2	1700	37	NA	8	2,1016	3,4375	2,6667	2,2157
V1+V2	1700	37 – 37	Valor máximo	15 – 8	-	3,0859	-	2,1372
			Valor mínimo		-	4,0546	-	3,0457
			Valor médio		-	3,6484	-	2,2614
V1	1750	37	NA	15	1,0938	3,7422	1,7909	2,9739
V2	1750	37	NA	8	2,1328	3,3594	2,6405	2,1895
V1+V2	1750	37 – 37	Valor máximo	15 – 8	-	3,0312	-	2,1176
			Valor mínimo		-	4,0703	-	3,0457
			Valor médio		-	3,6250	-	2,2679
V1	1800	37	NA	15	1,1016	3,7500	1,7974	2,9935
V2	1800	37	NA	8	2,1328	3,3750	2,6667	2,2549
V1+V2	1800	37 – 37	Valor máximo	15 – 8	-	3,0312	-	2,1830
			Valor mínimo		-	4,0937	-	3,0653
			Valor médio		-	3,6171	-	2,2549

Tabela A.12: Tabela de Resultados: *Radius* 7x7. Treinamento S1_L1_13-59.