UNIVERSIDADE ESTADUAL DO CEARÁ

CENTRO DE CIÊNCIAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO

ÉRIC GUSTAVO REIS DE SENA

SISTEMA DE CONTAGEM DE PESSOAS UTILIZANDO PROCESSAMENTO INTELIGENTE DE IMAGENS

FORTALEZA – CEARÁ

2016

ÉRIC GUSTAVO REIS DE SENA

SISTEMA DE CONTAGEM DE PESSOAS UTILIZANDO PROCESSAMENTO INTELIGENTE DE IMAGENS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências e Tecnologia da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Orientador: Prof. Dr Marcial Porto Fernandez

Co-Orientador: Prof. Dr José Everardo Bessa Maia

FORTALEZA – CEARÁ

2016

Dados Internacionais de Catalogação na Publicação

Universidade Estadual do Ceará

Sistema de Bibliotecas

Sena, Eric Gustavo Reis de .
Sistema de contagem de pessoas utilizando processamento inteligente de imagens [recurso eletrônico] / Eric Gustavo Reis de Sena. - 2016.
1 CD-ROM: il.; 4 ¾ pol.

CD-ROM contendo o arquivo no formato PDF do trabalho acadêmico com 88 folhas, acondicionado em caixa de DVD Slim (19 x 14 cm x 7 mm).

Dissertação (mestrado acadêmico) - Universidade Estadual do Ceará, Centro de Ciências e Tecnologia, Mestrado Acadêmico em Ciência da Computação, Fortaleza, 2016.
Área de concentração: Ciência da Computação.
Orientação: Prof. Dr. Marcial Porto Fernandez.
Coorientação: Prof. Dr. José Everardo Bessa Maia.

1. Visão Computacional. 2. Contagem de pessoas. 3. Câmeras de Circuitos Fechados de TV. I. Título.

Mestrado Acadêmico em Ciência da Computação UECE MACC

ATA DA NONAGÉSIMA PRIMEIRA DEFESA PÚBLICA DE DISSERTAÇÃO DE MESTRADO

Ao vigésimo oitavo dia do mês de agosto de dois mil e dezesseis, no miniauditório do prédio de Pesquisa e Pós-Graduação em Computação, do Mestrado Acadêmico em Ciência da Computação – MACC, realizou-se a sessão pública de defesa da dissertação de **ERIC GUSTAVO REIS DE SENA,** aluno regularmente matriculado no Mestrado Acadêmico em Ciência da Computação–MACC, intitulada: **"SISTEMA DE CONTAGEM DE PESSOAS UTILIZANDO PROCERSSAMENTO INTELIGENTE DE IMAGENS ".** A Banca Examinadora reuniu-se no horário de 15:00h às 16:30 horas, sendo constituída pelos Professores Doutores **Marcial Porto Fernandez (Orientador/UECE); Leonardo Sampaio Rocha (UECE); e Rafael Bráz Azevedo Farias(UFC).** Inicialmente o mestrando expôs seu trabalho e a seguir foi submetido à arguição pelos membros da Banca, dispondo cada membro de tempo para tal. Finalmente a Banca reuniu-se em separado e concluiu por considerar o mestrando APROVADO, por sua dissertação e sua defesa pública. Eu, **Professor Dr. Marcial Porto Fernandez**, Orientador e Presidente da Banca, lavrei a presente Ata que será assinada por mim e demais membros da Banca. Fortaleza, 30 de setembro de 2016.

Prof. Dr. Marcial Porto Fernandez
(Orientador – UECE)

Prof. Dr. Leonardo Sampaio Rocha
(UECE)

Prof. Dr. Rafael Bráz Azevedo Farias
(UFC)

AGRADECIMENTOS

Aos meus pais, Sebastião Carlos de Sena e Maria Helena Reis de Sena e à Patrícia Veríssimo de Oliveira, mãe da minha amada filha Ágatha Sena, pelo eterno incentivo e carinho.

Ao bolsista voluntário Washington Luiz de Lima Praxedes Filho que foi vital para a realização das atividades.

Aos meus amigos Alexandre Galvão Patriota, José Eudes Pires Rodrigues, Nilton Antônio de Oliveira Júnior, Camila Cunha e Emanuelly Oliveira, que de uma forma ou de outra me estimularam e apoiaram.

Aos meus colegas por suas sugestões, em especial Altino Dantas Basílio Neto e Aratã Saraiva.

A todos os professores que me fizeram crescer ao longo do curso, em especial aos professores Gustavo Augusto Lima de Campos, Mariela Inés Cortés e Marcial Porto Fernandez pelas oportunidades.

Agradeço à Federação das Indústrias do Estado do Ceará pela concessão de bolsa de estudo.

A todos aqueles que de alguma forma fizeram parte da minha vida e me transfomaram no que sou hoje.

“Não temos medo de admitir que não sabemos, não precisamos ter vergonha disso. A única vergonha é fingir que temos todas as respostas.”

(Neil deGrasse Tyson)

RESUMO

O monitoramento visual inteligente, com ênfase para o campo da contagem automática do fluxo de pessoas, consiste em estimar a posição e o deslocamento de indivíduos em uma área estratégica. Além de ser de uma enorme relevância, é algo bastante desafiador para os sistemas de vigilância contemporâneos, visto que viabiliza a extração de estatísticas essenciais que possibilitam tomada de decisões específicas. Este trabalho apresenta um método de contagem de pessoas baseado em combinações de técnicas de visão computacional e meta-heurísticas. O método proposto faz esta contagem por meio da análise de imagens capturadas por uma câmera de vídeo fixa, utilizando uma modelagem adaptativa do *background*, em que os *pixels* de *foreground* são classificados e agrupados em regiões conexas no formato de imagens binárias, o que permite a diferenciação e a associação dos objetos em cena. Sequencialmente é agregado um conjunto de técnicas de matemática morfológica que corrige distorções na imagem binária. Desse modo, torna-se viável a aplicação de uma técnica de análise de bolha para extrações de métricas, essenciais para a metodologia de contagem de pessoas baseada em um classificador supervisionado, o k-vizinhos mais próximo. Viabiliza-se assim, também, a determinação do posicionamento e da direção de deslocamento quadro a quadro das pessoas, obtidos por meio do filtro preditivo de Kalman combinado com o algoritmo húngaro, que resolve os problemas de atribuições das detecções. Além disso, foi implementada uma estrutura que armazena as direções de deslocamento das pessoas baseada em uma máquina de estados para fim de contagem por direção. A metodologia foi validada pelo desenvolvimento de um protótipo, sendo realizados experimentos que demostraram que o sistema proposto é eficiente em cenários de baixa intensidade de luminosidade, isto é, sem oscilações súbitas de claridade.

Palavras-chave: Visão Computacional. Contagem de Pessoas. Câmeras de Circuitos Fechados de TV.

ABSTRACT

The intelligent visual monitoring, with emphasis on the automatic counting of course the flow of people is to estimate the person flow in a particular strategic area, in addition to be of paramount importance, it is quite challenging to contemporary surveillance systems, as enables the extraction of essential statistics for monitoring people to specific decision-making. This paper presents a method of counting people based on combinations of computer vision techniques and meta-heuristics. The proposed method estimates the flow of people through the image stream analysis captured by a still video camera, using an adaptive modeling background, where the pixels of foreground are classified and grouped into related areas in binary image format, which allows for differentiation and association of the objects in the scene, it is sequentially added one morphological mathematical technique set that corrects distortion in the binary image. Thus, it enables the implementation of a blob analysis technique for metric extraction essential for people counting methodology based classifier, the k-Nearest Neighbors, as well as the determination of the position and direction a frame by frame offset of persons obtained through predictive Kalman filter combined with Hungarian algorithm to solve the problems assignments of probes, moreover, a structure which stores the offset directions people based on the state machine was implemented in order to count per direction. The methodology was validated by the development of a prototype, which performed experiments have shown that the proposed system is effective in low intensity light scenarios, i.e. without sudden fluctuations in brightness.

Keywords: Computer Vision. People Counting. Closed-Circuit Television Camera

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ALGORITMOS

LISTA DE ABREVIATURAS

AVI	Audio Video Interleaved
EM	Expectation-Maximization
GMM	Gaussian Mixture Model
JPEG	Joint Photographic Expert Group
k-NN	k-Nearest Neighbors
LRS	Learning, Recognition, and Surveillance
MATLAB	Matrix Laboratory
MPP	Modelo de Processo Pixel
PETS2009	Performance Evaluation of Tracking and Surveillance
RGB	Red, Green and Blue
ROI	Region Of Interest
SVH	Sistema Visual Humano

SUMÁRIO

1 INTRODUÇÃO

O desenvolvimento de sistemas complexos de processamento de vídeos que viabilizam a contagem de pessoas de forma automatizada tem-se tornado possível devido aos atuais avanços das tecnologias dos computadores, das técnicas de visão computacional e das facilidades e da disponibilidade de imagens a baixo custo, sejam elas obtidas por meio de câmeras de segurança ou de outros dispositivos de captura de vídeos.

A contagem de pessoas caracteriza-se como um problema extremamente desafiador, devido à complexidade envolvida no processo, e é essencialmente importante, pois possibilita a coleta de dados estatísticos que são imprescindíveis para obtenção de indicadores para tomada de decisões estratégicas que auxiliarão nas operações de gerenciamento e de planejamento. Entre os diversos exemplos de sua aplicação, existem: a medição da eficácia do marketing em relação às vendas ou à divulgação, o monitoramento de uma área em um shopping para justificar-se o preço do aluguel com base no volume de pedestres, a otimização do fluxo de pessoas em eventos públicos, o controle de entrada e de saída em edifícios no caso de uma evacuação de emergência, a gestão de vigilância pública ou privada, etc.

As técnicas de análise de imagens se destacam pela capacidade de: diferenciar pessoas de outros objetos, contabilizar o número de pessoas em grupos e facilidade de se adaptar uma câmera ao computador em ambientes internos ou externos e não exigindo um infra-estrutura especial como nos outros métodos (SILVA, 2008).

O meio de captura de imagens pode ser realizada com apenas uma câmera (ambiente 2D) ou com a combinação de várias câmeras (ambiente 3D). De modo geral, a estimação da quantidade de pessoas pode ser realizada sob uma determinada imagem estática ou uma sequência de imagens.

Entre as principais características que um sistema de contagem de pessoas deve apresentar são: execução em tempo real, adquirir independência da necessidade de um ambiente controlado, ou seja , um sistema que se adapte as mudanças dinâmicas das condições ambientais (sombra, variação de iluminação etc), auto-ajuste de layout de passagem a fim de setorizar a região monitorada (frequentemente zoneados por uma ou mais linhas virtuais que podem compor uma região fechada para que o sistema conte o número de pessoas entrando ou saindo da região, ou ainda permanecendo no seu interior), solucionar problemas relacionados a densidade do fluxo de pessoas, uma vez que os cenários com intenso fluxo de indivíduos acarretam uma aglomeração da multidão, e consequentemente podem causar um grande número de oclusões

dinâmicas (objetos ou pessoas em movimentos que sobrepõem uns aos outros ao longo do tempo) e adequá-se as situações de oclusões estáticas devido as estruturas do cenário ou objetos fixos que podem encobrir o objeto-alvo.

Além de todos os fatores ambientais ou fenômenos de atuação (ações dos objetos na cena) não controlados, existem também outros, não menos importantes tais como, a qualidade da imagem, o tipo de lente da câmera e a capacidade do sistema de utilizar dados de múltiplos sensores (câmeras) ou de apenas uma câmera, que é o caso do modelo adotado neste trabalho.

Na literatura, a contagem de pessoas tem sido categorizada em dois métodos, sendo o primeiro fundamentado na detecção, e o segundo, em mapa. Em Hou e Pang (2011) é abordado o método detecção, em que o número de pessoas em uma cena é determinado pela detecção de pessoas uma a uma e apontando suas posições. Essa detecção pode se suceder de dois modos: investigando os padrões de delineamento físico das pessoas e suas características, buscando, por exemplo, padrões redondos ou ovais para representar uma cabeça, formatos retangulares para os braços e assim por diante, ou ainda identificando pessoas diretamente de uma imagem. Já o método em mapa explora a ligação entre algumas características da imagem, como a análise de texturas e de *pixels* que estão presentes no primeiro plano na imagem, e o número de pessoas. Os dois métodos apresentam suas vantagens e desvantagens. Segundo Maddalena et al. (2014), admite-se que pode ser encontrada uma maneira de mesclar os dois modelos para o benefício da contagem de pessoas.

O presente trabalho se propõe a quantificar o tráfego de pedestres considerando suas direções de deslocamento, esse método se baseia em contagem por meio de algoritmo de classificação de padrões. O cenário é simulado em ambiente parcialmente controlado, em que, é aplicado o método de detecção por meio de câmeras (ambiente 2D) posicionadas obliquamente em diferentes posições de uma mesma perspectiva.

Deste modo, o sistema proposto neste trabalho consistiu na associação de técnicas aplicadas por Silva (2008) e Valle (2007). Em Silva (2008) se baseio na técnica de subtração do plano de fundo da cena para detecção de objetos móveis em cenas de vídeos capturado por uma câmera, aliada a técnicas que possibilitam a identificação de pessoas utilizando o conhecimento a-priori sobre métricas corporais para auxiliar no processo de identificação, assim como, é utilizando o filtro preditivo com a finalidade rastrear as pessoas dentro de uma região de interesse no campo visual da câmera, em que a região de interesse é setorizada em sub-regiões representadas por linhas virtuais que auxiliam nas definições das políticas de contagens das

pessoas ao deixarem a cena. Já em Valle (2007) duas abordagens foram propostas, a primeira é baseada em dois limiares, tais como a largura média dos objetos que contém apenas uma pessoa, e a área média superior desses objetos que normalmente engloba as cabeças das pessoas contidas em um objeto, que são comparadas posteriormente com a largura e a área de cada novo objeto com estes limiares, decidindo se nele contêm uma, duas ou três pessoas, enquanto a segunda abordagem utiliza um classificador previamente treinado para, dado um objeto, decidir se ele contêm uma, duas, ou três pessoas. Para tal fim, é aplicado um esquema de zoneamento do objeto e características como área e largura são extraídas de sua região superior.

Além da reprodução e combinação parcial das técnicas citadas, foi agregado a esse trabalho uma metodologia de máquina de estado com o objetivo de se definir as regras de contagens multidirecional, onde a região de captura das cenas sob o campo de visão da câmera foram setorizados por linhas virtuais, de modo formar sub-regiões que servirão para indicar as mudanças de localizações das pessoas entre essas sub-regiões, e assim, obterem-se as contagens por direções, ou seja, as contagens para cada direção são realizadas independente umas das outras, e estas são feitas nos momentos que os indivíduos abandonarem o campo de visão da câmera.

2 REFERENCIAL TEÓRICO

Neste capítulo, é introduzida uma breve revisão bibliográfica das principais técnicas de análise de vídeo com o objetivo de estimar a quantidade de pessoas presentes em uma determinada região de interesse na cena. Ele será dividida em três blocos: segmentação do movimento, rastreamento e contagem de pessoas.

2.1 SEGMENTAÇÃO DO MOVIMENTO

A segmentação de movimento consiste em separar os objetos presentes em uma sequência de imagens considerando as suas trajetórias no tempo e espaço, permitindo a análise do padrão do movimento. A priori não há uma maneira específica para diferenciar os métodos de segmentação de movimento e, na literatura, ocorre sua subdivisão em seis grupos: diferença de imagem, teoria estatística, fluxo óptico, *wavelet*, *layers* e fatoração.

Lipton et al. (1998) abordou a técnica de diferença temporal, que é um método para segmentar objetos em movimento em uma sequência de imagens verificando a diferença de dois ou três *frames* subsequentes. A limitação dessa técnica dá-se na incorporação de objetos que não se movimentam como parte do modelo do *background*, porém ela é tolerante à variação de iluminação.

Em Haritaoglu et al. (1998), para separação do primeiro plano da imagem de fundo, são empregadas as técnicas de filtro de mediana e limiarização, que são sobrepostas aos valores dos *pixels* ao longo de alguns instantes no vídeo. Analisa-se o número de vezes em que um determinado *pixel* foi classificado como pertencente ao *background* e ao *foreground*, e o número de *frames* desde a última vez em que um *pixel* recebeu a classificação de um *foreground*. À proporção que um *pixel* for classificado muitas vezes como *background* ou *foreground* e isso não for compatível com sua última classificação, esta será atualizada.

Stauffer e Grimson (1999) abordou um modelo estatístico multimodal para segmentação capaz de incorporar movimentos contínuos. O modelo consiste em um conjunto de funções de probabilidade que representam a informação visual de cada *pixel*, isto é, efetua-se a modelagem de cada *pixel* como sendo a combinação de várias distribuições Gaussianas. Então, conforme os novos valores para cada *pixel* em sua posição específica vão se atualizando, o modelo busca comparar esse atual valor com todas as Gaussianas do modelo do background. Caso alguma distribuição correspondente seja encontrada, os parâmetros das distribuições são

atualizados. Caso contrário, uma nova Gaussiana é obtida, em que o valor da média é igual ao valor do *pixel* e a sua variância é maior. As vantagens e desvantagens dessa técnica são similares à técnica de diferença temporal, porém a GMM exige um maior esforço computacional.

A subtração de fundo é um método que busca separar os *pixels* em dois grupos: o *foreground* e o *background*. Isso é feito mediante a identificação das alterações de *pixel* a *pixel* de cada sequência de *frame* do vídeo, tendo como referência um modelo estático do *background* para classificá-los Snidaro et al. (2005) e Björgvinsson (2006). A principal limitação desses métodos deriva da dependência de um modelo de *background* estático. Essa característica faz com que seja um modelo extremante sensível a mudança de iluminação, embora tenha como vantagem ter a capacidade de segmentar objetos que não estão em movimento na cena.

2.2 RASTREAMENTO

Com os resultados das segmentações obtidas na etapa anterior, segue-se a etapa de rastreamento dos objetos do primeiro plano. O rastreamento de um objeto-alvo em vídeo consiste em segui-lo continuamente ao longo do tempo, de modo a prever sua trajetória enquanto ele permanecer dentro do campo de monitoramento. Geralmente identificam-se os componentes correlatos por meio de uma função fundamentada em distância ou em similaridade, equiparando as características extraídas a priori dos objetos, ou seja, em cenas anteriores, com os presentes objetos detectados em cenas posteriores.

Existem diversos métodos de rastreamento, porém o critério da escolha do método que se deve adotar é relativo intrinsecamente à complexidade quanto à capacidade de lidar com multiobjetos, oclusões, ruídos adversos, oscilações da intensidade de luminosidade, etc.

Segundo Amer (2005), os algoritmos de rastreamento de objetos podem ser divididos entre aqueles que rastreiam partes específicas de um objeto e os que rastreiam objetos por completo. O primeiro aborda modelo 2D ou 3D dos objetos, enquanto o segundo aborda o casamento de característica e predição de posição dos objetos.

Conforme Elgammal et al. (2000), a aplicação do modelo 2D e 3D para rastreamento das partes de pessoas, assim como das pessoas, é um trabalho complexo, em razão do número de parâmetros gerados pela combinação de um modelo de imagem autêntica com os modelos abordados. Ele sugere rastrear objetos empregando características visuais de baixo nível como forma, cor, linhas e pontos, pois esses são computacionalmente mais eficientes.

Haritaoglu et al. (2000) compôs um cânone de movimento apoiado nas grandezas de velocidade e aceleração com a finalidade de ponderar os posicionamentos dos objetos no *frame* corrente em relação aos *frames* subsequentes. Assim, de modo recursivo, uma nova posição deve ser estimada para cada objeto dos *frames* seguintes para que os objetos ainda não identificados sejam relacionados quanto às suas posições relativas, de tal modo que, aqueles que estiverem mais próximos sejam agregados.

Kim et al. (2002) abordou o rastreamento de objetos comparando as características do *frame* anterior com as do *frames* atual por meio de *bounding box* que circunvizinha os objetos, de modo a obter os atributos de medidas como fecho convexo, área e ponto central. Em alguns casos, ao longo da dinâmica, os centros das *bounding boxes* podem fundir-se, situação que acarreta na junção de vários objetos na mesma *bounding box*, produzindo consequentemente uma segmentação em que os objetos se tornam meramente um, com seus deslocamentos preditos, averiguando suas velocidades e direções até o instante presente, procedendo dessa forma de modo correto com a oclusão de objetos.

Amer (2005) propôs um método de rastreamento de objetos baseado no casamento de características de tamanho, aspecto e deslocamento. A correspondência entre as características de objetos é com frequência realizada empregando equações de restrição, distância ou similaridade. Ou seja, pela equiparação das grandezas adquiridas por meio das características apuradas com limites de antemão especificadas. O autor justifica a sua abordagem pelo fato de que os algoritmos fundamentados em estimativa de movimento ou predição de posição não se curvam facilmente em cenários complexos, com grande densidade de objetos e, portanto, com maior quantidade de oclusões.

De acordo com o trabalho descrito por Snidaro et al. (2005), o rastreamento é efetuado com a abordagem do filtro de Kalman, cujos parâmetros são obtidos por meio dos *bounding boxes* que circunvizinham os objetos, a fim de se obter as características como a média dos valores de cor, as coordenadas da *bounding box*, as coordenadas do centróide, etc. Essa abordagem se fundamenta na propriedade de recursividade, em razão de projetar um estado de um processo dinâmico que dispensa a memorização de todos os estados decorridos, garantindo assim uma boa estimativa inicial do local em que cada objeto pode ser encontrado na imagem seguinte.

Wang et al. (2014) apresenta um sistema de detecção de objetos em movimento chamado *Flux Tensor* com modelos de *Split Gauss* que explora as vantagens do método de movimento baseado na formulação tensor espaço-temporal. Esse sistema híbrido pode lidar com desafios como as sombras, mudanças de iluminação, fundo dinâmico e objetos parados e removidos.

2.3 CONTAGEM DE PESSOAS

Existem muitas tecnologias que são usadas para contar pessoas por meio de dispositivos, tais como lazer, imagem térmica, raios infravermelhos, visão computacional e wifi. A seguir serão citadas as mais relevantes que utilizam as técnicas de visão computacional.

Schofield et al. (1997) propôs um método de redes neurais com o intuito de discriminar e reconhecer o *background* improgressivo e os elementos do *foreground* em movimento advindos de cenas de vídeo. O sistema apresentado teve como objetivo contar de modo automático as pessoas que permanecessem diante de um elevador aguardando sua chegada. O intento é que se possa aprimorar o desempenho do serviço do elevador.

Kettnaker e Zabih (1999) desenvolveram um sistema de contagem de pessoas fundamentado em um modelo estatístico de aparência, construída para cada pessoa em conjunto com algumas restrições topológicas do ambiente. Essas restrições se devem pelo fato do método utilizar múltiplas câmeras para monitorar diferentes lugares do mesmo ambiente, de forma que seja capaz de localizar nas câmeras subsequentes o indivíduo. Isso significa que, de antemão, as câmeras não sobrepõem a mesma área de captura, significando que uma pessoa é capturada uma vez em cada câmera de forma sequencial.

Haritaoglu et al. (2000) associou dois métodos fundamentados na geometria do molde do objeto para a contagem de quantidade de pessoas, tendo como referência as cabeças dos indivíduos. Assim, uma análise é efetuada de forma local, onde a silhueta da parte superior do objeto é checada, a fim de verificar se os *pixels* que compõem o formato da região superior são correspondentes aos da curvatura de uma cabeça. Uma segunda análise, de forma global, é obtida por meio da projeção do histograma vertical do objeto, em que os picos relevantes são empregados com o objetivo de filtrar os resultados extraídos da análise feita localmente da abordagem anterior. Posteriormente, os contornos similares a uma cabeça são mantidos, posto que os picos obtidos pelo método antecedente que não alcançarem um limiar de altura devem ser excluídos. A contagem é obtida da investigação do eixo do tronco: caso seja completamente

traçada uma linha do chão ao pico por dentro da silhueta segmentada, uma pessoa é contabilizada.

Em Kim et al. (2002), é exposto um sistema de contagem de pessoas que localiza e rastreia pessoas em movimento, utilizando-se de uma única câmera imóvel e ortogonalmente posicionada em relação ao piso, com a finalidade de diminuir o problema de oclusão. Na etapa de segmentação, o autor abordou a subtração de imagem com atualização dinâmica, seguido de binarização e aplicações de operações morfológicas. Posteriormente dispõe-se do processo de estimação *Convex Hull Approximation* que avalia a região fronteiriça retangular em volta das pessoas contidas na imagem. Assim, nos subsequentes *frames*, são estimadas as áreas dos retângulos e é checado se há alguma pessoa, possibilitando, desse modo, a garantia de que quando alguns indivíduos forem sobrepostos parcialmente, os rastreadores não os percam. A peculiaridade do sistema está na sua potencialidade em distinguir pessoas dos demais objetos em cena, como também, computar genuinamente o total de pessoas em grupos.

Em Bhuvaneshwar e Mirchandani (2004), é proposto um sistema de detecção de pedestres em um cruzamento, em tempo real, para controle adaptativo do sinal de trânsito, usando uma câmera fixa. Foram abordadas a subtração de imagem, seguida de binarização, aplicações de operações morfológicas e a técnica de histograma para extração de sombras dos objetos. Posteriormente, os pedestres são identificados e os demais objetos são suprimidos, em seguida são comparados quanto à altura e área de cada um dos retângulos, que são produzidos de forma a envolverem os objetos percebidos em cena. Assim, com os parâmetros extraídos dessas medidas de grandezas, estima-se a quantidade de pedestres.

Kong et al. (2005) expôs um sistema que deve ser a priori treinado por meio de redes neurais. A espinha dorsal do sistema se dá com o seguinte embasamento: primeiramente com a aplicação de técnica de subtração do *background*, e posteriormente com a localização dos objetos e o histograma do perímetro em torno de cada objeto localizado. Com o sistema treinado, é feita a contagem de pessoas dentro de uma região predefinida no campo de visão da câmera.

Snidaro et al. (2005) construiu um contador de pessoas baseado em rastreador. Optou-se pela instalação de câmeras por cima do recinto, de forma que os objetos seriam contabilizados no momento em que fossem interceptados após passar por uma linha virtual de contagem dentro campo de visão da câmera. O motivo principal pela abordagem de câmeras sobre o ambiente é o fato de assim ser possível evitar falhas provocadas por oclusões. A metodologia de contagem de pessoas se fundamenta na área em que está contido o objeto, ou seja, é analisada a área preenchida pelo objeto que contém uma pessoa com a área de um outro objeto que adentra no

campo de visão da câmera, pois a área que contém uma pessoa é aproximadamente constante, e desse modo é estimado o número de pessoas contidas.

Sidla et al. (2006) utilizou um algoritmo multiestágios para detectar uma ampla margem de bordas nas imagens, denominado algoritmo (CANNY, 1986). A abordagem desse algoritmo se justifica pela aplicação em ambientes superlotados, de forma a fazer a discriminação de atributos obtidos da região superior para representar uma pessoa (cabeça e ombro). Posteriormente, os dados das regiões superiores discriminadas são comparados com os dados de um modelo obtido de forma manual.

Em Jeon e Rybski (2006), é proposto um sistema que determina o número de pessoas presentes em uma sala. Em um dado intervalo de tempo as posições das faces das pessoas são localizadas por um detector de faces, que se baseia em estatísticas que determinam o número de pessoas presentes em sala. Experimentos mostraram que a presença de ruído na detecção das faces resulta em falsas faces, e mesmo uma imagem de face real com características diferentes pode ser confundida com um ruído, o que torna as análises das imagens muito difíceis.

Björgvinsson (2006) utilizou câmeras posicionadas verticalmente sobre o ambiente, em saídas e entradas de lojas, para contar quantas pessoas passaram por elas. Quando um objeto passa pela área de fluxo de contagem, ele é rastreado até sair de cena. Se desaparecer através da área de entrada, o contador de entrada é incrementado. Se o contrário ocorrer, ou seja, se desaparecer através da área de saída, o contador de saída é incrementado. Um objeto pode ser dividido em dois, quando sua largura ultrapassa a largura máxima que ele pode ter, tornando assim a contagem mais precisa.

Em Valle (2007) foram propostos dois métodos para a contagem automática de pessoas utilizando técnica de visão computacional, sendo a primeira abordagem baseada em dois limiares: largura média das bolhas que contém apenas uma pessoa e representação da área média da região superior dessas bolhas, que normalmente engloba as cabeças das pessoas contidas em uma bolha, e compara cada nova bolha com esses limiares. Depois, decide se nele estão contidas uma, duas ou três pessoas. Já a segunda abordagem utiliza um classificador *k*-vizinhos mais próximos previamente treinado para uma dada bolha, a fim de decidir se ela contém uma, duas, ou três pessoas. Para isso, é adotado um esquema de zoneamento da bolha e características como área e largura são extraídas de sua região superior. A abordagem apresentou limitações na segmentação em ambientes superlotados, segmentando mais de uma pessoa em uma única bolha. Mostrou também erros causados por oclusões. Quanto ao rastreador, este cometeu vários

erros em ambientes de muito movimento. Outra inconveniência foi a necessidade de calibração manual do sistema para o ambiente.

Em Teixeira e Savvides (2007), é proposto um sistema para localização e contagem de pessoas em áreas internas, baseado no movimento e nas dimensões de um indivíduo. Um histograma mostra o movimento de objetos que foram detectados utilizando o método da diferença de *frames* da cena. O sistema é resistente a flutuações de intensidade dos *pixels*, à variações graduais de iluminação e ao reposicionamento de objetos na cena. O sistema de contagem de pessoas foi implementado com múltiplas câmeras ligadas em rede e calibradas de acordo com a área de captura de cada câmera. Alterações abruptas de iluminação podem causar detecções falsas, mas que desaparecem nos *frames* seguintes.

Silva (2008) apresentou métodos de análise de imagens para a detecção de objetos móveis na imagem capturada por uma câmera de vídeo fixa utilizando-se de uma técnica de subtração do plano de fundo da imagem do quadro corrente da cena associada às técnicas de remoção de sombras e ruídos, que permitem a identificação da localização e o reconhecimento de pessoas. Propôs-se uma abordagem de rastreamento por meio de filtro preditivo em que os indivíduos seriam contados ao deixarem o ambiente. O sistema evidenciou ser efetivo nos experimentos empreendidos.

Zaccariotto (2010) propôs o desenvolvimento de um protótipo que utiliza um método de reconhecimento de padrões em vídeo, no qual é realizada a identificação de pessoas em um ambiente controlado. Na etapa de segmentação, é utilizado o conceito de movimento que se baseia na utilização de uma imagem de referência do ambiente. Após esta etapa, é realizada a extração de característica, e, em seguida, gerada a base de conhecimento mediante a classificação supervisionada. Na etapa da classificação, é utilizado o algoritmo dos k-vizinhos mais próximos.

No trabalho de Yoshinaga et al. (2010), foi proposto um sistema para contagem do número de pessoas em tempo real. O sistema calculou quantos pedestres havia e onde estavam nas sequências de vídeo pelos seguintes procedimentos: no primeiro momento, as regiões candidatas são segmentadas em *blobs* de acordo com a subtração de fundo; no segundo momento, um conjunto de características é extraído a partir de cada *blob* e de uma rede neuronal, que calcula o número de pedestres que corresponde a cada conjunto de características. Para realizar o processamento em tempo real, foram utilizados apenas recursos simples e válidos, e um fundo de modelagem adaptativa foi utilizado para estimar a densidade Parzen, que realiza a detecção de objetos nas imagens de entrada. Apresentou uma precisão superior a 80%.

Em Mukherjee e Das (2013b), foi apresentado um robusto e inovador modelo, O Modelo de Ômega, para detecção e contagem de seres humanos na cena. O modelo proposto emprega um conjunto de quatro descritores distintos para identificar as características únicas das regiões de cabeça, pescoço e ombros de uma pessoa. Esta assinatura única cabeça-pescoço-ombro, dada pelo Modelo Ômega, explora desafios tais como: as variações entre tamanho e forma das regiões da cabeça, pescoço e ombro para alcançar uma detecção robusta de seres humanos, mesmo sob oclusão parcial; e alterações dinâmicas do fundo e das condições de iluminação. Os autores comentaram como trabalho futuro a implementação do Modelo Ômega estendida para a aplicação em vídeo.

O trabalho de Sivabalakrishnan e Shanthi (2015) apresentou um novo método para segmentação de pessoas, rastreamento e contagem. Propôs-se uma abordagem Fuzzy baseada em combinação de várias heurísticas em aplicações de processamento de imagem. Uma vez que apresentou uma abordagem eficiente, confiável e automática para segmentação, rastreamento e contagem de pessoas, passou a ser usado em sistemas de vigilância. Para lidar com os desafios de extração de objetos em ambientes dinâmicos, desenvolveu-se um sistema de inferência baseado em lógica Fuzzy para identificação de pessoas em rastreamento. O método de contagem de pessoas proposto é simples, mas eficiente, e alcançou um bom desempenho em tempo real.

No trabalho de Al-Zaydi et al. (Aug. 2016), foi apresentado um sistema de contagem de multidões adaptável para aplicações em vigilância de vídeo. O método proposto é composto de um par de modelos do processo Gaussiano colaborativo com diferentes núcleos, que são concebidos para contar as pessoas, levando em conta o nível de oclusão. O nível de oclusão é medido e comparado com um limiar predefinido para seleção do modelo de regressão para cada quadro. Além disso, o método proposto identifica dinamicamente a melhor combinação de características de contagem de pessoas. Os resultados mostram que o método proposto oferece uma precisão mais elevada quando comparada com o estado da arte dos métodos referidos na literatura aberta.

3 OBJETIVOS

3.1 GERAL

Desenvolver um sistema protótipo especialista, capaz de estimar o fluxo de pessoas por direção em cenas de vídeos utilizando contagem baseada em um classificador de padrão supervisionado.

3.2 ESPECÍFICOS

(a) Reconhecer e localizar os objetos em cena;
(b) Construir um modelo de contador baseado em um algoritmo de classificação previamente treinado;
(c) Contar pessoas levando em consideração as direções finais do deslocamentos;
(d) Testar o método aplicando a câmeras em diferentes ângulos de uma mesma cena;
(e) Avaliar a eficácia do método por meio da análise estatística de desempenho.

4 FUNDAMENTAÇÃO TEÓRICA

Nesta divisão serão expostas as tarefas desenvolvidas de processamento e análise de imagens de maneira abrangente. Portanto, serão revisados diversos conceitos básicos e essenciais, que auxiliarão na compressão das particularidades dos métodos de maior complexidade, buscando, à medida que for pertinente, comentar cada um do ponto de vista matemático.

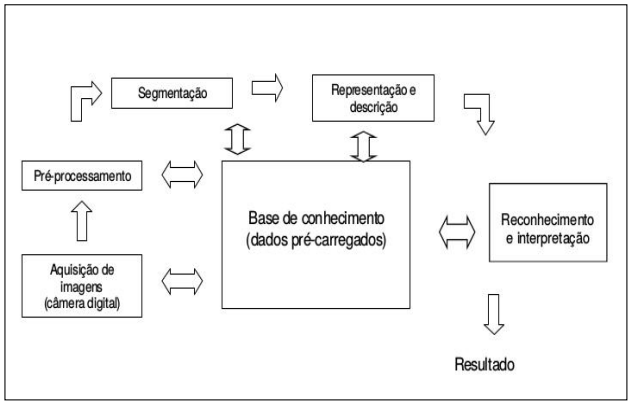
Como passo inicial, é importante tratar resumidamente alguns conceitos básicos relevantes à imagem digital e suas terminologias. Nas abordagens seguintes, os conceitos mais complexos são introduzidos gradativamente a fim de auxiliar na compreensão e nas tarefas de implementações dos modelos matemáticos.

4.1 PROCESSAMENTO DE IMAGEM DIGITAL

De acordo com (GONZALEZ; WOODS, 2007), “o processamento de imagens é dividido nas seguintes etapas: aquisição da imagem, pré-processamento, segmentação, representação e/ou descrição e reconhecimento e/ou interpretação". A aquisição da imagem é feita por meio de um dispositivo de entrada – hardware, como câmeras digitais, por exemplo. O pré-processamento visa deixar a imagem mais “limpa”, corrigir possíveis falhas na aquisição e ainda realçar características importantes, como bordas e vértices.

As etapas fundamentais para o processamento de imagens digitais, segundo (GONZALEZ; WOODS, 2007) resultam na Figura 1.

Figura 1 – Método para o processamento de imagens digitais



Fonte**:** Adaptado de Gonzalez e Woods(2007).

A etapa de segmentação é responsável por agrupar os *pixels* pertencentes a um mesmo objeto ou região de interesse. A segmentação de imagens é frequentemente abordada

quando se pretende localizar objetos e formas em imagens. A resolução de um problema de segmentação de imagem é baseada em um conjunto sequencial de técnicas adaptadas ao domínio do problema. Na etapa de representação e descrição, busca-se extrair características de interesse que possibilitem a descrição e o agrupamento em classes de objetos. Já a etapa de reconhecimento procura atribuir um rótulo ao objeto, ancorada nas informações de uma base pré-determinada (GONZALEZ; WOODS, 2007).

4.1.1 Definição de Imagem digital

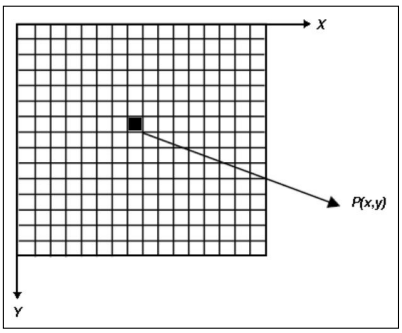
O início da construção de uma imagem digital dar-se-á a partir da imagem analógica capturada por um dispositivo capaz de submetê-la à digitalização, isto é, discretizar os valores das coordenadas espaciais e os valores das amplitudes dos níveis de cores, esses processos são denominados respectivamente por amostragem e quantização.

A representação de um modelo matemático da imagem monocromática digital auxilia na fácil generalização do conceito para imagem multicromática. Assim, uma imagem monocromática pode ser descrita formalmente como uma função bidimensional de intensidade luminosa, representada por uma matriz *f(x,y)* de tamanho *N x M*, exibida em 4.1. Onde cada elemento da matriz representa um pixel (*Picture and Element*) da imagem convertido de uma imagem analógica, isto é, um pixel equivale à atribuição de um valor discreto para representar uma cor ou intensidade, de forma a associá-lo a um par de coordenadas espaciais positivas, denotado por x e y.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix} \tag{4.1}$$

Uma exposição gráfica dessa matriz pode ser vista na Figura 2, em que, por convenção, a origem da imagem está situada no canto superior esquerdo da figura.

Figura 2 – A figura mostra uma imagem e como a representamos nos eixos x e y no plano cartesiano.



Fonte**:** Computação gráfica - Teoria e prática, volume 2, AZEVEDO, Eduardo; CONCI, Aura; LETA, Fabiana - 2008.

4.1.2 Modelo de Imagens

Segundo (PEDRINI; SCHWARTZ, 2008), uma importante propriedade física, que pode ser melhor compreendida por meio de um modelo físico para a intensidade de uma cena sob observação, pode ser expressa em termos de componentes, como a intensidade de luz incidente e refletida pelo objetos presentes na cena. Esses componentes são chamados de *iluminância*[1] (medida em $lumen/m^2$ ou lux) e *reflectância* (medida em $watt/sr\ m^2$), respectivamente, e são simbolizadas por $\imath(x,y)$ e $\varsigma(x,y)$.

A função *f(x,y)* pode ser representada como:

$$f(x,y) = \imath(x,y)\varsigma(x,y), \tag{4.2}$$

em que os limites teóricos são: $0 < \imath(x,y) < \infty$ e $0<\varsigma(x,y)<1$, com a característica de $\imath(x,y)$ ser determinada pela fonte de luz, enquanto $\varsigma(x,y)$ é determinada pelas características dos objetos na cena, que são invariante a mudanças de iluminação, composição de cena ou geometria.

Todos esses conceitos mencionados podem ser estendidos à imagem multiespectral, na qual a imagem colorida pode ser representada pelas combinações das cores primárias (R, *red*), verde (G, *green*) e azul (B, *blue*).

[1] refere-se somente às luzes branco e preto, enquanto valor das cores refere-se a crominância, que é composta pela matiz e saturação.

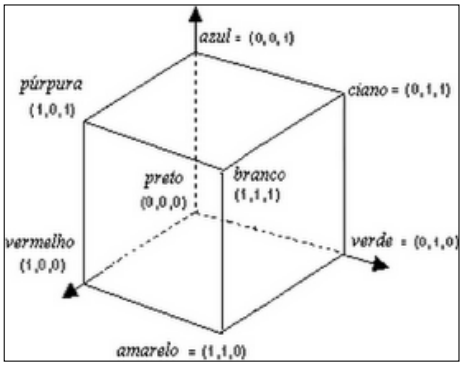
4.1.3 Modelo de Cores

As cores podem ser definidas como percepções visuais a partir de um sinal luminoso. Elas são cruciais na análise de imagens. Do ponto de vista físico, cor é uma representação da luz: radiação eletromagnética detentora de energia (partículas denominadas fótons) que incide sob um objeto e seus pigmentos refletem a sua cor. O seres humanos capturam e interpretam as radiações com intervalo de comprimento de onda entre 300 e 830 nanômetros.

Em um dispositivo digital, a cor é reproduzida através de modelos representativos que medem quantitativamente os fatores envolvidos em uma determinada escala. Um modelo de cor é tridimensional quando cada ponto corresponde a uma cor diferente. No presente trabalho, será abordado somente o modelo RGB. Embora existam diversos outros modelos de cores, estes não serão considerados, valendo ressaltar que nenhum modelo de uma cor é necessariamente superior a outro. Frequentemente, a escolha de um modelo de cor é imposta por fatores externos.

O sistema RGB tem como componentes do espaço de cor as cores primárias vermelho, azul e verde. Dependendo do sistema de aplicação, as componentes do modelo em termo de escala podem variar entre 0 a 255 (para imagens de 8 *bits*) ou 0 a 1 (cubo normalizado).

Figura 3 – Representação cartesiana do espaço RGB de cores normalizado.



Fonte: <http://gasparbrogueira.web44.net/metodologia.html>.

Na Figura 3, é possível perceber que os pontos com iguais intensidades nas três coordenadas, ou seja, aqueles que fazem parte da diagonal do cubo que tem início na origem e termina no ponto máximo do cubo, são equivalentes aos tons de cinza.

A maioria dos dispositivos utiliza a escala RGB, os quais possuem sensores que simulam a ação dos cones no sistema visual humano, responsáveis pela detecção de luz. Diante disso, (TKALCIC et al., 2003) estabeleceu que os valores dos componentes RGB são dados

pelas somas da sensibilidade dos sensores com a incidência de luz, de acordo com a equação abaixo:

$$\begin{aligned} R &= \int_{300}^{830} S(\lambda)R(\lambda)\,d\lambda \\ G &= \int_{300}^{830} S(\lambda)G(\lambda)\,d\lambda \\ B &= \int_{300}^{830} S(\lambda)B(\lambda)\,d\lambda, \end{aligned} \tag{4.3}$$

em que $S(\lambda)$ é o grau de incidência luminosa capturada pelo dispositivo e $R(\lambda)$, $G(\lambda)$ e $B(\lambda)$ são funções de sensibilidade de cada sensor de cor, sendo λ a variável de integração (comprimento de onda perceptível). Note que os intervalos de integração são baseados na capacidade de captura de comprimento de onda do ser humano.

A partir da equação acima, verifica-se que os valores dos pontos na escala RGB dependem da sensibilidade do sensor em distinguir cores. Portanto, esse modelo de cor é dependente do dispositivo.

4.2 SEGMENTAÇÃO

O processo de segmentação subdivide uma imagem em regiões e objetos que a compõem. A precisão desta etapa determina o sucesso ou o fracasso final dos procedimentos de análise computacional (GONZALEZ; WOODS, 2009). Em outras palavras, a segmentação é a etapa de identificação dos objetos contidos na imagem, que é fundamentada na mensuração de dissimilaridade (regiões com grandes variações) e similaridade (*pixels* similares são unidos para formar o objeto).

4.2.1 Limiarização

As diversas técnicas de segmentação têm como objetivo segmentar uma imagem em dois grupos fundamentais para a execução das etapas posteriores. O primeiro grupo é denominado *foreground* (primeiro plano), que é composto por uma coleção de objetos de interesse em uma imagem, enquanto o segundo grupo é denominado *background* (segundo plano), que é a composição da imagem sem a presença desses objetos, em outras palavras, é tudo aquilo que define o plano de fundo no qual os objetos estão enquadrados. A remoção do background garante que, em uma imagem, seja apenas selecionado e considerado o objeto de

interesse, que então o sistema identificará e comparará com os traços de interesse (BOURIDANE, 2009).

Estes dois grupos são obtidos por meio de uma imagem limiarizada, isto é, de acordo com a técnica da limiarização, que consiste em atribuir um valor ou rótulo fixo para todos os *pixels* de mesmo grupo, estabelecendo assim um limiar de acordo com as características dos objetos que se quer isolar do plano de fundo.

Seja $L(x,y)$ uma imagem limiarizada no intervalo $[0,T]$, defina-se então que:

$$L(x,y) = \begin{cases} T & se\ f(x,y) > T \\ f(x,y) & se\ f(x,y) \leq T. \end{cases} \tag{4.4}$$

A imagem L(x,y) permite diferentes combinações de truncamento, sendo que a limiarização mais clássica é a binária, na qual há somente dois valores possíveis para cada *pixel*: o 0 para representar a cor preta e o 1, para a cor branca.

4.2.2 Subtração de Imagem

A técnica de subtração de imagem é um modelo denominado não-adaptativo e não estatístico, indicada para ambientes extremamente controlados, uma vez que é sensível às mudanças de ambientes.

O método de subtração consiste na subtração entre uma imagem estática do ambiente observado e a imagem de cada frame consecutivo com uma posterior limiarização da diferença e, assim, gera uma imagem binária, na qual os objetos contidos em si são separados do fundo da imagem.

Em (HEIKKILÄ; SILVÉN, 2004) é apresentado um modelo de subtração ponderada, podendo ser resumidamente representado pelas equações abaixo:

$$\begin{aligned} &| I_t(i,j) - B_t(i,j) | > \tau \\ &B_{t+1}(i,j) = \alpha \times I_t(i,j) + (\alpha - 1) \times B_t(i,j), \end{aligned} \tag{4.5}$$

em que $I_t(i,j)$ é o *frame* de entrada corrente, $B_t(i,j)$ é a imagem correspondente ao modelo corrente de fundo de imagem e τ é uma constante definida experimentalmente. Note que, os valores de $B_t(i,j)$ são atualizados a cada segmentação utilizando a soma ponderada.

Os modelos baseados em subtração de fundo de imagem foram muito utilizados quando, no passado, barreiras computacionais impostas pelo *hardware* disponível limitavam a

complexidade dos sistemas de processamento de vídeo em tempo real (STAUFFER; GRIMSON, 1999).

Nos dias atuais, diante do maior poder computacional oferecido, a maioria das pesquisas em análise de movimento humano faz uso de modelos mais robustos e complexos. Nesse contexto, a próxima abordagem será dos métodos baseados em modelos adaptativos estatísticos de detecção de movimento.

4.3 MODELO DE PROCESSO PIXEL

Na abordagem estatística, é apresentado o modelo de processo *pixel* mencionado em Friedman e Russell (1997) e Stauffer e Grimson (1999). Esse modelo se baseia na estimação por meio de uma mistura de distribuições Gaussianas, em que a combinação de várias distribuições possa representar cada *pixel* em sua específica posição de modo-temporal visando, assim, de forma dinâmica e adaptativa, obter a reclassificação de cada *pixel* como parte ou não da estrutura que compõe o *background*.

4.3.1 Princípio

Considerando um processo que se desenvolve no tempo em torno da média, ou seja, um processo estacionário, tal que $X_{i,j}$ é uma variável aleatória que representa uma série temporal dos valores observados de um *pixel* particular, em uma específica posição i e j no frame, cuja descrição é dada por:

$$X_{i,j} = \{X_1, ..., X_t\} = \{I(x_R, x_G, x_B, l) : 1 \leq l \leq t\} \tag{4.6}$$

designa-se I como uma sequência de imagem tal que cada *pixel* caracterizado no espaço de cor RGB é denotado por (x_R, y_G, z_B).

Define-se que a função de probabilidade do valor da observação do corrente *pixel* $X_t \in R^p$ em algum instante é dada pela Equação (4.7) (pode ser interpretada como a probabilidade de se observar uma intensidade particular ou cor do *pixel*), nas quais os parâmetros associados ao tempo são: K (número de distribuições), $\omega_{i,t}$ (o coeficiente da i-enésima componente da mistura de Gaussianas denotado como peso), η (representado na Equação (4.8), trata-se de uma função de densidade Gaussiana com média $\mu_{i,t}$ e matriz de covariância $\Sigma_{i,t}$).

$$P(X_t) = \sum_{i=1}^{k} \omega_{i,t} \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \tag{4.7}$$

$$\eta(X_t, \mu_{i,t}, \Sigma_{i,t}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_{i,t}|}} e^{-\frac{1}{2}(X_t - \mu_{i,t})^T \Sigma_{i,t}{}^{-1} (X_i - \mu_{i,t})}. \tag{4.8}$$

Por razões computacionais, Stauffer e Grimson (1999) assumiram que os componentes do espaço de cor RGB são independentes e têm as mesmas variâncias σ^2, logo assume-se independência para o produto de normais univariadas. A suposição permite evitar uma inversão de matriz dispendiosa, em detrimento de alguma precisão, assim a matriz de covariância é da forma:

$$\Sigma_{i,t} = \sigma^2{}_{i,t} I, \tag{4.9}$$

em que I é uma matriz diagonal principal no espaço tridimensional.

4.3.2 Significados dos Parâmetros

Os parâmetros iniciais das distribuições envolvidas para descrever o modelo de processo *pixel* precisam ser estimados. Nessa conjuntura a determinação dos estimadores para estes parâmetros é de extrema importância, uma vez que a atribuição de valores iniciais para esses parâmetros afetam a precisão da extração do fundo.

Neste contexto, Zang e Klette (2006) analisaram em detalhe o impacto de diferentes valores dos parâmetros iniciais. Assim, essa análise será referência de como escolher adequadamente os valores dos parâmetros iniciais, de modo que, quando necessário, serão atribuidos limites razoáveis que garantam melhores resultados.

Em suma, os valores do *pixel* podem ser modelados com múltiplas Gaussianas, e para melhor entendimento de como representá-los, é necessário compreender a função de densidade Gaussiana Multivariada e seus parâmetros.

A função de densidade Gaussiana Multivariada descrita é determinada pelo vetor de média $\mu_{i,t}$ que representa os valores de *pixel* em cada pilha RGB e por sua matriz de covariâncias $\Sigma_{i,t}$ que generaliza a noção de variância para múltiplas bandas, isto é, representa a medida de variação de intensidade de cada banda espectral e entre cada par de bandas.

Outro importante elemento estrutural do modelo são as componentes, referenciadas como o peso $\omega_{i,t}$ de cada Gaussiana, isto é, representam a contribuição relativa de cada Gaussiana na modelagem dos *pixels*. Portanto, quão maior esse valor, mais importância relativa à função de densidade Gaussiana tem para os valores dos *pixels*. Zang e Klette (2006) propôs parâmetros de misturas de Gaussianas no módulo de análise, em que K denota o número de componentes $\omega_{i,t}$

em um modelo de mistura Gaussiana. Definiu que para as cenas simples em ambientes fechados, um valor pequeno de K é suficiente, talvez K = 2 ou para as cenas em ambientes externos, é necessário um valor maior para K, usualmente de 3, 4, ou 5.

4.3.3 Modelo de Gaussiano Inicial

Estabeleceu-se uma distribuição multiváriavel Gaussiana, de acordo com Morellas et al. (2003) e Lee (2004), que propuseram o uso do algortimo EM (*Expectation-Maximization*) como algoritmo para estimar os parâmetros iniciais da distribuição Gaussiana multivariável. Esse método agrega mais precisão para estimar a distribuição inicial dos valores de *pixel* de imagem e formação de um modelo mais ajustado a essa fase. Mas o inconveniente dessa abordagem é que ela exige a necessidade de coletar dados por algum tempo, de modo a gerar informações estatísticas, desperdiçando, assim, os recursos do computador com armazenamento de dados.

Stauffer e Grimson (1999) generalizou a abordagem, propondo uma modelagem de mistura de K distribuição gaussianas, em que o histórico de cada *pixel* em um intervalo de tempo, como já descrito na Equação (4.6), seria temporariamente armazenado. E, assim, um algorítimo *k*-Médias *online* é considerado para obtenção dos parâmetros iniciais dessas Gaussianas. Desse modo, cada novo valor observado para cada *pixel* é verificado em relação à distribuição das K Gaussianas, até que a adequação é localizada, segundo o critério da Equação (4.10) e posteriormente o modelo é atualizado.

Deve-se atualizar os parâmetros com o objetivo de adaptar a distribuição às mudanças do *background*, estabelecendo assim novas distribuições Gaussianas. Isto se justifica pelo fato de existirem diversos fatores que causam mudanças nas cenas ao longo do tempo, como a variação de iluminação.

A vantagem desse modelo é que qualquer objeto pode ser incorporado e reincorporado ao *background*, uma vez que o modelo mantém o *background* existente, assim os *pixels* originais permanecem na mistura até que um novo *pixel* observado seja mais provável.

4.3.4 Critério de Correspondência

Cada novo valor de *pixels* X_t é verificado em relação à distribuição das K distribuições Gaussianas existentes, até que seja encontrada uma correspondência dado pela restrição definida na Equação (4.10). Se nenhuma das K distribuições corresponder ao critério para o valor do *pixel* corrente, a distribuição menos provável é substituída por uma distribuição com o valor atual

do *pixels* igual ao seu valor médio $\mu_{i,t+1} = X_{t+1}$, uma variância $\sigma^2_{i,t+1}$ inicialmente elevada e um peso $\omega_{i,t+1}$ baixo. Caso contrário seus parâmetros são atualizados conforme na equação (4.14)

$$d_i(\mu_{i,t}, X_{t+1}) = \sqrt{(X_{t+1} - \mu_{i,t})^t \Sigma^{-1}_{i,t} (X_{t+1} - \mu_{i,t})} < \kappa\sigma_{i,t}. \tag{4.10}$$

4.3.5 Atualização dos Pesos

A atualização dos pesos $\omega_{i,t}$ tem como objetivo ajustar o nível de significância para todas as funções de Gaussianas dentro do modelo, isto é, o quanto cada função de distribuição de Gaussiana que compõe o modelo representa para estimação dos valores de *pixels*. A atualização de $\omega_{i,t}$ é representada na equação abaixo:

$$\omega_{i,t+1} = (1 - \alpha) \times \omega_{i,t} + \alpha \times \mathbf{1}_i(X_{t+1}), \tag{4.11}$$

$$\mathbf{1}_i(X_{t+1}) = \begin{cases} 1, & se\ d_i(\mu_{i,t}, X_{t+1}) < \kappa\sigma_{i,t} \\ 0, & se\ \text{caso contrário}, \end{cases} \tag{4.12}$$

em que $\mathbf{1}_i(X_{t+1})$ indica a correspondência da i-ésima Gaussiana para o pixel X_{t+1} e α a taxa constante de aprendizagem, onde $\omega_{i,t+1}$ está sujeito a uma restrição:

$$\sum_{i=1}^{k} \omega_{i,t+1} = 1. \tag{4.13}$$

4.3.6 Atualização da Média e Variância

Para atualização dos parâmetros (média e matriz de covariância), devem-se conceituar os *pixels* seguindo o critério de correspondência mencionado na Equação (4.10). O novo *pixel* será associado à distribuição Gaussiana que tiver maior correspondências e assim a Gaussiana associada tem seus parâmetros atualizados a fim de corrigir o modelo inicial, mantendo para as demais Gaussianas a média e matriz de covariâncias originais, tal que os seus respectivos pesos $\omega_{i,t}$ são decrementados segundo uma taxa de aprendizagem α, visto na Equação (4.11). Caso esse novo *pixel* não tenha nenhuma associação com as Gaussianas do modelo, a Gaussiana com menor peso $\omega_{i,t}$ tem seus parâmetros substituídos.

No caso dos critérios dados na Equação (4.12) indicarem correspondência, os parâmetros do i-ésimo componente são atualizados da seguinte forma:

$$\begin{aligned} \mu_{i,t+1} &= (1-\rho)\mu_{i,t} + \rho X_{t+1} \\ \sigma^2_{i,t+1} &= (1-\rho)\sigma^2_{i,t} + \rho(X_{t+1} - \mu_{i,t+1})(X_{t+1} - \mu_{i,t+1})^T, \end{aligned} \tag{4.14}$$

em que $\rho = \alpha \times P(\mathbf{X}_{t+1} \mid \mu_{i,t}, \Sigma_{i,t})$, tal que, $P(\mathbf{X}_{t+1} \mid \mu_{i,t}, \Sigma_{i,t})$ é a função de probabilidade do valor da observação do corrente *pixel* com seus respectivos parâmetros, α é o parâmetro de aprendizado predefinido, $\sigma^2_{i,t+1}$ é a variância da i-ésima gaussiana na mistura tempo t+1, $\mu_{i,t+1}$ é a média do *pixel* no tempo t+1 e $X_t + 1$ é o *pixel* no tempo t+1.

Já no caso das distribuições sem correspondência os únicos parâmetros a serem atualizado são:

$$\begin{aligned} \mu_{i,t+1} &= \mu_{i,t} \\ \sigma^2_{i,t+1} &= \sigma^2_{i,t}. \end{aligned} \tag{4.15}$$

Se X_{t+1} corresponde a nenhuma das distribuições K, então a distribuição menos provável (ou seja, a distribuição com o menor peso) é substituída por uma distribuição em que o valor de corrente X_{t+1} atua como média. E para o desvio é escolhido um valor empírico "alto", enquanto para o peso a priori é escolhido um valor empírico "baixo" (STAUFFER; GRIMSON, 1999).

4.3.7 Modelo de Estimação do Background

Depois que a inicialização dos parâmetros é obtida pelo processo *pixels*, a primeira detecção do *foreground* pode ser feita e então os parâmetros são atualizados.

Zang e Klette (2006) afirmaram que a estimação do *background* é resolvida com a especificação das distribuições Gaussianas, com a que tem mais suporte de evidência em uma menor variância, uma vez que o objeto em movimento tem variância maior do que um *pixel* do *background*. Então, a fim de representar o *background*, primeiramente, é adotado o critério de taxa $r_i = \omega_{i,t}/\|\Sigma_{i,t}\|$ para efetuar a ordenação decrescente das K distribuições Gaussianas. Assim a distribuição do fundo permanece no topo com a variância menor pela aplicação de um limiar T.

Conforme Bouwmans et al. (2008) esta ordenação supõe que o *pixel* do *background* corresponde ao maior peso com uma pequena variância devido ao fato de o *background* ser mais presente do que objetos em movimentos, como também ter valor praticamente constante.

A Equação (4.16) escreve formalmente que, dadas as K-Gaussianas ordenadas conforme o critério e taxa já mencionados, são selecionadas as B primeiras distribuições Gaussianas que superam um certo limite T para representarem uma provável distribuição do *background* e as outras distribuições restantes são consideradas para representar a distribuição do *foreground*. Assim, cada *pixel* será rotulado como pertencente ao *background* ou *foreground* da seguinte forma: todos os *pixels* X_t que não tiverem correspondência conforme o critério na (4.10), com nenhuma componente serão rotulados como *foreground*; caso contrário, serão rotulados como *background*. Por conseguinte, são feitas as atualizações dos parâmetros conforme a representação na Equação (4.14).

$$B = \text{argmin}_{\text{b}} \left(\frac{\sum_{i=1}^{b} \omega_{i,t}}{\sum_{i=1}^{k} \omega_{i,t}} > T \right). \tag{4.16}$$

Segundo Mukherjee e Das (2013a), o GMM é adaptativo, pois pode incorporar as mudanças lentas de iluminação e a remoção e adição de objetos para o *background*. Quanto maior o valor de T na Equação (4.16), maior é a probabilidade de um fundo multimodal.

4.4 MORFOLOGIA MATEMÁTICA

A morfologia matemática surgiu com a colaboração de Serra (1994), que estabeleceu seus conceitos básicos e suas ferramentas, com a finalidade de estudar a estrutura geométrica da imagem.

A linguagem utilizada na morfologia matemática é a teoria dos conjuntos. Conjuntos, neste contexto, representam objetos em uma imagem. Por exemplo, o conjunto de todos os *pixels* pretos em uma imagem binária é uma descrição morfológica completa da imagem. Em imagens binárias, os conjuntos em questão são membros do espaço dos inteiros bidimensionais - 2D - Z^2, em que cada elemento do conjunto é uma tupla, cujas coordenadas são as coordenadas de um pixel preto na imagem. Imagens em tons de cinza podem ser representadas como conjuntos, cujos componentes estão em Z^3. Neste caso, dois componentes de cada elemento do conjunto referem-se às coordenadas do *pixel*, e o terceiro corresponde ao valor do seu nível de cinza. Conjuntos em espaços dimensionais ainda mais elevados podem conter outros atributos da imagem, como cor ou componentes que variam com o tempo (GONZALEZ, 2002).

Os filtros morfológicos são aplicados para corrigir e diminuir ao máximo os erros de segmentação e ruídos na morfologia do objeto, completando pequenos buracos e eliminando

regiões isoladas. Os filtros morfológicos mais utilizados são os filtros de Dilatação e Erosão. Ambos atuam nas bordas internas e externas dos objetos (SILVA, 2008).

A característica principal dos filtros morfológicos é trabalhar com um elemento estruturante e com as características da sua vizinhança percorrendo uma imagem binária. A morfologia do objeto pode ser moldada realizando duas operações na imagem: a erosão e a dilatação. Gonzalez e Woods (2009) define simplificadamente a dilatação como a expansão dos componentes de uma imagem, enquanto a erosão é definida como uma redução desses componentes.

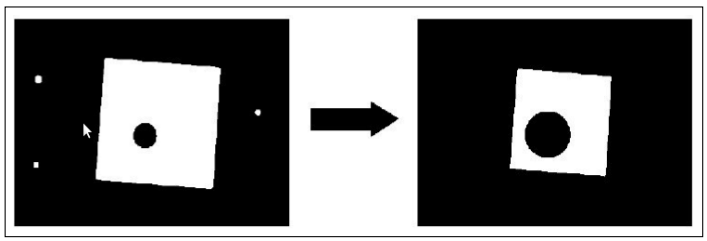
Frequentemente, operações primitivas de processamento de imagem são adotadas a fim de amenizar os erros morfológicos dos objetos, ou seja, são aplicadas algumas sequências de filtros de operações morfológicas, tais como o de “abertura”, “fechamento” e “preenchimentos" binários, de modo a preencher pequenas aberturas ou suprir as regiões isoladas de poucos *pixels*.

Contudo, nas técnicas em que são utilizadas limiarizações para obtenção de uma imagem binária, o resultante da segmentação de movimento ainda pode conter falsos positivos objetos em virtude da tenacidade de ruídos nos *frames* limiarizados L_n, pois as limiarizações podem eliminar grande parte dos falsos positivos de imagens, mas poderá também eliminar objetos-alvo verdadeiros em cenas.

4.4.1 Erosão

O efeito do operador de erosão em uma imagem é o desgaste (redução) gradual das regiões de borda de *pixels* de *foreground*. Assim, sua principal função é eliminar pequenos ruídos nas imagens a fim de destacar apenas elementos importantes na imagem. Como pode ser visto na Figura 4, aplicando-se a erosão, os objetos contidos no primeiro plano se tornarão pequenos, enquanto “buracos” presentes nesses objetos se tornarão maiores.

Figura 4 – Erosão de uma imagem binária com um elemento estruturante tipo disco

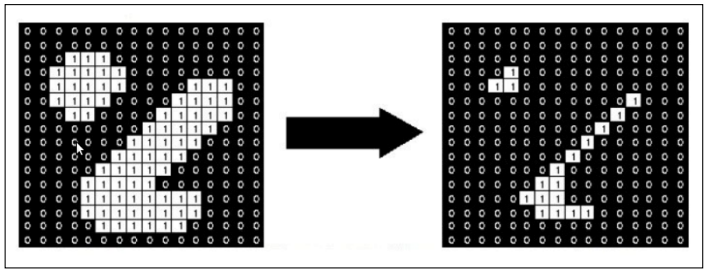


Fonte**:** Elaborado por Lefloch (2007).

Cada operação morfológica utiliza um elemento de estrutura para fazer a erosão e a dilatação de uma imagem. Esses elementos possuem um formato específico, podendo este ser um retângulo, um círculo, um losango, entre outros tipos de formas geométricas. Ao se definir o tipo de elemento estrutural que se deseja aplicar à erosão ou à dilatação na imagem, os *pixels* vão se lapidando um por um até atingir o formato pretendido.

Na Figura 5 é usado um elemento estrutural em forma de um quadrado, sendo ele uma matriz quadrada de tamanho 3×3, conforme representado na Equação (4.18). O funcionamento da erosão se dá da seguinte maneira: o algoritmo busca cada *pixel* contido no primeiro plano da imagem, que são os *pixels* denotados pela cor branca, e aplica em cada um o elemento estrutural definido, de uma forma que o centro do elemento estrutural, denotado pelo quadrado vermelho na Figura 6, coincida com o *pixel* analisado no momento. O teste se dá quando é verificado se o elemento estrutural está contido em primeiro plano da imagem. Se o elemento estiver contido, o *pixel* analisado continuará em primeiro plano; do contrário, o *pixel* passará a ser agora parte do plano de fundo da imagem.

Figura 5 – Erosão de uma imagem binária com um elemento estruturante quadrado 3×3



Fonte**:** Elaborado por Lefloch (2007).

Simbolicamente descrita pela Equação (4.17):

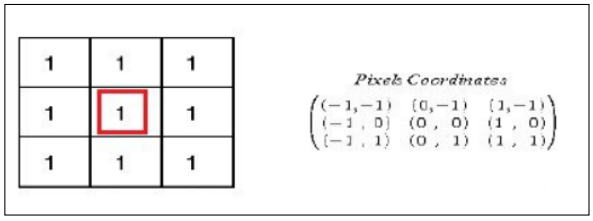
$$L_n^e = L_n \ominus B, \tag{4.17}$$

em que ($\ominus$) indica a operação de erosão e B o elemento estruturante (máscara de covolução 3×3).

4.4.2 Dilatação

A dilatação é uma operação morfológica oposta à erosão. Ela consiste em expandir os *pixels* que delimitam os objetos em primeiro plano na imagem. Como pode ser demostrado na

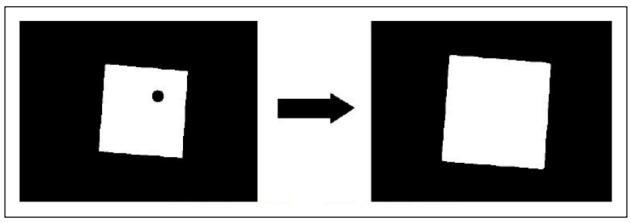
Figura 6 – Elemento estrutural em forma de quadrado 3×3



Fonte**:** Elaborado por Lefloch (2007).

Figura 7, aplicando-se a dilatação, os objetos contidos no primeiro plano se tornarão maiores, enquanto que os buracos presentes nesses objetos se tornarão menores, podendo até mesmo desaparecer da imagem.

Figura 7 – Imagem binária de uma dilatação morfológica



Fonte**:** Elaborado por Lefloch (2007).

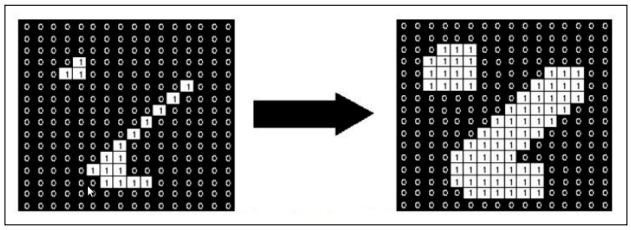
Simbolicamente descrita como:

$$L_n^d = L_n \oplus B, \tag{4.18}$$

em que ($\oplus$) indica a operação de dilatação e B, o elemento estruturante (máscara de convolução 3×3).

O funcionamento da dilatação apresenta a mesma ideia do funcionamento da erosão. A diferença, no caso, é que em vez de o algoritmo aplicar o elemento estrutural em cada *pixel* do primeiro plano, ele aplica nos *pixels* contidos no plano de fundo da imagem, que são os *pixels* representados pela cor preta na Figura 8. Sendo assim, o algoritmo identifica cada *pixel* contido no plano de fundo da imagem e aplica em cada um deles o elemento estrutural definido na Figura 7, de forma que o centro do elemento estrutural coincida com o *pixel* analisado no momento. O teste se dá quando é verificado se o elemento estrutural contém algum *pixel* presente no primeiro plano. Se o elemento estrutural contiver algum *pixel* do primeiro plano, então o *pixel* analisado fará parte do *foreground* da imagem, do contrário, o *pixel* continuará como parte do plano de fundo.

Figura 8 – Dilatação de uma imagem binária com um elemento estruturante quadrado 3×3

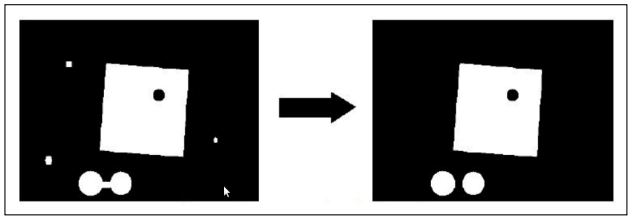


Fonte**:** Elaborado por Lefloch (2007).

4.4.3 Abertura Morfológica

A abertura morfológica é uma combinação das duas operações anteriores, a erosão e a dilatação, como representado na Equação (4.19). A ideia dessa técnica consiste em primeiro realizar a erosão e logo em seguida aplicar a dilatação, a fim de se eliminar os ruídos que estão contidos na imagem, ou seja, trata-se de uma dilatação da erosão de uma imagem. Na Figura 9, pode ser visto o resultado final após a realização desta técnica, nas quais os ruídos que estavam presentes anteriormente na imagem acabaram desaparecendo.

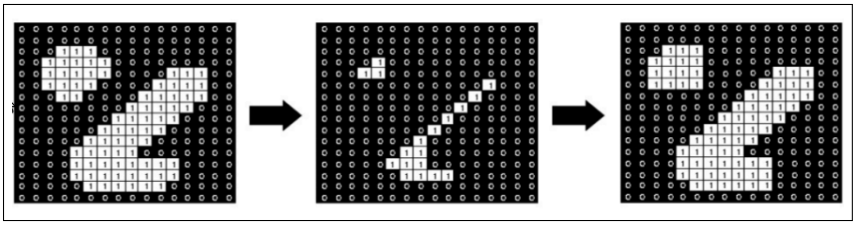
Figura 9 – Abertura morfológica em uma imagem



Fonte**:** Elaborado por Lefloch (2007).

Na Figura 10, é visto como se dá o funcionamento dessa operação. O algoritmo primeiro aplicará o elemento estrutural nos *pixels* em primeiro plano, realizando a erosão e separando quais *pixels* continuam em primeiro plano e quais passam a pertencer ao plano de fundo. Terminada a operação, é aplicada a dilatação na erosão obtida. O algoritmo identificará só os *pixels* do plano de fundo para análise e aplicará o elemento estrutural neles, polindo o objeto da imagem de uma forma que os *pixels* defeituosos sejam eliminados e os ruídos existentes na imagem desapareçam. Com essa nova filtragem, o resultado que se obtém é um conjunto dos *pixels* dos objetos em movimentos segmentados. Todavia, neste resultado há possibilidade de existirem regiões de um mesmo objeto desconectadas.

Figura 10 – Imagem binária de uma abertura morfológica



Fonte**:** Elaborado por Lefloch (2007).

Simbolicamente descrita como:

$$L_n^a = ((L_n \ominus B) \oplus B) \circ B, \tag{4.19}$$

em que ($\circ$) indica a operação de abertura e B, o elemento estruturante (máscara de convolução 3x3).

4.4.4 Tratando Objetos Desconectados

Em alguns casos, as imagens segmentadas apresentam falsos objetos-alvo, devido ao baixo contraste entre o fundo e o objeto em movimento, situação que pode levar a não segmentação de algumas partes dele, fazendo com que um único objeto tenha mais partes desconectadas. Nestes segmentos devem ser agrupadas ou eliminadas as regiões consideradas pequenas, de forma que represente cada objeto por apenas um único conjunto de *pixels* conectados, tomando cuidados para não modificar o seu tamanho. (DEDEOGLU, 2004) aplicou com êxito sucessivas operações de dilatação e erosão para eliminar falhas de segmentação provenientes de ruído, fechando regiões abertas e unindo partes separadas.

Valle (2007) sugeriu duas operações morfológicas com o objetivo de conectar as partes para se obter um único objeto. Assim, os passos referentes as essas operações são apresentadas da seguinte forma:

- Primeiro - aplica-se a operação de dilatação para conectar os segmentos mais próximos, cuja operação é descrita na Equação de adição de Minkowski, que segue abaixo:

$$L_n^{''} = L_n^{'} \oplus B, \tag{4.20}$$

na qual ($\oplus$) é a operação de dilatação aplicada à imagem $L_n^{'}$, usando elemento estruturante B (máscara de convolução 3x3)

- Segundo - aplica-se a erosão para que as características de forma e tamanho dos objetos sejam mantidas cuja operação é descrita na equação de Minkowski, que se segue abaixo:

$$L_n''' = L_n'' \ominus B. \tag{4.21}$$

Apesar da variedade de estratégias adotadas para minimizar o ruído de segmentação, a utilização delas não trará, necessariamente, bons resultados em qualquer situação. A eliminação de regiões pequenas, por exemplo, depende da definição de um limiar para exclusão de tais regiões. Um certo limiar poderá eliminar grande parte dos falsos positivos de imagens de ambiente interno, mas poderá também eliminar objetos-alvo verdadeiros em cenas de ambientes externos, em que a tomada da cena geralmente abrange uma área maior.

4.5 ANÁLISE DE BOLHA

Segundo Liu et al. (2007), a detecção de bolha no espaço de cor RGB se dá pela análise de fatores que levam o sistema visual humano (SVH) a identificar um objeto em uma imagem. O primeiro deles, denominado Mapa de Saliência da Região, é responsável por analisar a saliência da região do objeto, a partir da medida de contraste e da posição dos *pixels* que envolvem a região de interesse. Os contrastes de cada ponto são obtidos pela diferença Gaussiana de cores, dada pela Equação (4.22), em um espaço de cores não normalizado.

$$FD_{i,j} = (1 - e^{\frac{-d_{i,j}}{2\sigma^2}}) \times 255, \tag{4.22}$$

onde $d_{i,j}$ é a distância euclidiana entre as cores i e j. Além disso, deve-se considerar os outros fatores que permitem ao SVH identificar regiões de interesse, a saber:

Área: Intuitivamente, regiões com áreas maiores têm maior efeito sobre outras. As regiões de interesse são geralmente regiões com uma área grande em relação à imagem. Assim o parâmetro de área θ_1 pode ser definido como na Equação (4.23).

$$\theta_1 = \frac{A_i}{\text{Área da imagem}} \tag{4.23}$$

Efeito central: é o fator que está relacionado com a tendência de usuários atentarem para regiões centrais. Liu et al. (2007) definiu o fator de efeito central como uma função

densidade Gaussiana:

$$\theta_2 = 1 - e^{-\frac{P_i^2}{2\sigma^2}}, \tag{4.24}$$

em que P_i é a distância normalizada relativa da região em relação ao centro da imagem e σ, a saliência de regiões periféricas.

Diferença contextual: mede o fator de adjacência entre duas regiões baseado no número de *pixels* que as separam. Esse fator pode ser medido na Equação (4.25)

$$\theta_3 = 1 + \frac{V_{i,j}}{Vizinho_i}, \tag{4.25}$$

em que $V_{i,j}$ é o numero de *pixels* vizinhos entre duas regiões e $Vizinho_i$ é o numero total de *pixels* da região i.

Efeito global:A saliência de uma região inclui tanto o efeito global como a diferença contextual. As regiões vizinhas mais próximas afetam mais a atenção do sistema visual que as mais distantes (LIU et al., 2007). Dessa forma, essa propriedade é medida por uma função de densidade Gaussiana da distância espacial entre duas regiões. O valor numérico desse fator é dado pela Equação (4.26).

$$\theta_4 = 1 - e^{-\frac{SD_{i,j}^2}{2\sigma^2}}, \tag{4.26}$$

em que $SD_{i,j}$ é a distância normalizada entre as regiões i e j.

De posse do valor de todos os fatores que permitem ao sistema visual humano identificar regiões de interesse, a saliência geral de uma dada região i é calculada como na Equação (4.27).

$$S_i = \theta_2 \times \sum_{i=1}^{n} (FD_{i,j} \times \theta_1 \times \theta_4 \times \theta_3), \tag{4.27}$$

Caso haja mais de uma região de interesse na imagem, uma pessoa dedicará sua atenção a um de cada vez, baseado na ordem decrescente de saliência.

4.6 RASTREAMENTO

Rastreamento de pessoas em vídeo é o processo de localização de uma pessoa em movimento (ou grupos de pessoas) ao longo do tempo usando uma câmera. Sendo que seu principal objetivo é relacionar as ocorrências da mesma pessoa ao longo do percurso realizado, evitando que a mesma pessoa seja contada mais de uma vez.

Os propósitos de manter os objetos sob rastreamento são: necessidade de estabelecer a correspondência entre objetos conforme as medidas disponíveis ao longo do tempo, estimar a dinâmica, não completamente observável, do objeto e manter uma predição/estimação em situações de "morte" do objeto, ou seja, quando o objeto não é diretamente observável.

As dificuldades encontradas no rastreamento são: os múltiplos alvos (deve fazer várias associações), os alarmes falsos (detecções de falsos positivos), detecção de incerteza (oclusões, falhas de sensores, etc) e ambiguidades (várias medições na porta).

4.6.1 Filtro de Kalman

O filtro de Kalman foi desenvolvido em 1960 por R. E. Kalman, como uma solução recursiva para o problema de filtragem linear de dados discretos. É um modelo baseado em processo estocástico linear, que é capaz de lidar eficientemente com a tarefa de rastreamento de objetos em movimento em uma conjuntura de sistemas lineares dinâmicos e, desse modo, permitir uma inferência exata.

Além disso, é um modelo Bayesiano congênere a um modelo de Markov oculto, porém o espaço de estado das variáveis latentes é contínua e todas as variáveis latentes e observadas seguem uma distribuição Gaussiana. É importante salientar que, embora estes filtros sejam modelados em uma cadeia de Markov, os ruídos não necessariamente seguem uma distribuição Gaussiana e, no entanto, aplica-se esta hipótese, porque ela simplifica consideravelmente a matemática para derivar o filtro, e é normalmente uma aproximação válida, e, portanto, ela é parametrizada por média e covariância.

O filtro de Kalman exige pouco requerimento computacional, possuindo uma elegante propriedade de recursividade, em razão de projetar um estado de um processo dinâmico que dispensa a memorização de todos estados decorridos, assim é caracterizado como ótimo estimador para sistemas lineares unidimensionais, em que os erros seguem uma distribuição Gaussiana.

O procedimento em geral do método do filtro de Kalman é realizar as medições das observações através do acompanhamento (visual), posteriormente se obtêm as predições das medições a partir das *tracks* preditas, o que produz uma área no espaço do sensor onde se espera uma observação. Essa área, denominada de *validation gates*, pode ser utilizada para restringir as pesquisas, verificando se uma medida está situada nessa região, e se estiver, então ela será um candidata válida para um emparelhamento ou correspondência dos pares de objetos combinados.

A seguir é descrito formalmente o filtro de Kalman. A priori assume-se que o verdadeiro estado no tempo t para ser com base no estado no momento t-1 conforme a equação:

$$x_t = A_t x_{t-1} + B_t u_t + w_t, \tag{4.28}$$

em que: x_t é o vetor de estado contendo os termos de interesse para o sistema (posição, velocidade) no tempo t

u_t é o vetor que contém qualquer controle de entrada (ângulo de direção, ajuste de aceleração, força de travagem)

A_t é a matriz de transição de estado que se aplica o efeito de cada parâmetro de estado do sistema no instante t-1 no estado do sistema no momento t (por exemplo, a posição e velocidade no instante t-1 ambas afetam a posição no momento t)

B_t é a matriz de controle de entrada que se aplica o efeito de cada um deles. O parâmetro de entrada de controle do vetor u_t do vetor de estado (por exemplo, aplica-se o efeito da posição do acelerador sob a velocidade do sistema e posição)

w_t é o vetor contendo os termos do ruído processo para cada parâmetro no vetor de estado. O processo de ruído é assumido seguir uma distribuição normal multivariada com médias zero e matriz de covariâncias Q_t, sendo que $w_t \sim \mathcal{N}(0, Q_t)$.

No tempo t, as medições do sistema são efetuadas sob uma determinada observação ou medição y_t do estado de acordo com o modelo:

$$y_t = H_t x_{t-1} + v_t, \tag{4.29}$$

em que: y_t é o vetor de medições no tempo t

H_t é a matriz de transformação que mapeia os parâmetros do vetor de estado para o domínio de medição

v_t é o vetor contendo os termos de ruído de medição para cada observação no vetor de medição. Como o ruído do processo assumido médias zero e matriz covariâncias R_t representada por $v_t \sim \mathcal{N}(0, R_t)$

O verdadeiro estado do sistema x_t pode não ser observado diretamente e o filtro de Kalman providência um algoritmo para determinar uma estimativa $\hat{x}_t$, combinando com o modelo do sistema e as medições ruidosas de determinados parâmetros ou funções lineares de parâmetros. As estimativas dos parâmetros de interesse no vetor de estado são, portanto, agora fornecidas pela funções de probabilidade, em vez de valores discretos. O filtro de Kalman é baseado na distribuição Gaussiana, denotada como $\mathcal{N}_p(\mu_t, P_{t|/t-1})$. Em que $P_{t|t-1} = E[(x_t - \hat{x}_{t|t-1})(x_t - \hat{x}_{t|t-1})^T]$ é a matriz de covariância (associada com a predição $\hat{x}_t$ de valor desconhecido de x_t), onde os termos da diagonal principal estão associados com os termos correspondentes ao vetor de estado, enquanto os valores fora da diagonal principal fornecem a covariância entre os termos do vetor de estados.

As equações do filtro de Kalman permitem calcular de forma recursiva $\hat{x}_t$ associando com o conhecimento a priori. Desta forma, o ponto de partida para a execução propriamente dita das etapas de predição e correção conforme nas Equações (4.30) e (4.31).

$$\hat{x}_{t|t-1} = F_t \hat{x}_{t-1|t-1} + B_t u_t, \tag{4.30}$$

$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^T + Q_t. \tag{4.31}$$

A variância associada à predição $\hat{x}_t$ de um desconhecido x_t dado por $P_{t/t-1}$ é obtida pela diferença entre a Equação (4.28) e (4.30).

$x_t - \hat{x}_{t|t-1} = F(x_t - \hat{x}_{t|t-1}) + w_t \Rightarrow P_{t|t-1} = E[(F(x_t - \hat{x}_{t|t-1}) + w_t) \times (F(x_t - \hat{x}_{t|t-1} + w_t)^T)] = FE[(x_t - \hat{x}_{t|t-1}) \times (x_t - \hat{x}_{t|t-1}))^T] \times F^T + FE[x_t - \hat{x}_{t|t-1}) \times w_t^T] + E[w_t x_{t-1} - \hat{x}_{t|t-1}^T] F^T + E[w_t w_t^T]$.

note que os erros de estimação e o processo de ruídos são não correlacionados.

$E[(x_t - \hat{x}_{tvt-1})w_t^T] = E[w_t x_t - \hat{x}_{t|t-1})^T] = 0 \Rightarrow P_{t|t-1} = FE[(x_t - \hat{x}_{t|t-1}) \times (x_t - \hat{x}_{t|t-1}))^T] \times F^T + E[w_t w_t^T] \Rightarrow P_{t|t-1} = FP_{t-1|t-1}F^T + Q_t$.

As equações de atualização de medição são dado por:

$$\hat{x}_{t|t} = \hat{x}_{t-1|t-1} + K_t(z_t - H_t\hat{x}_{t|t-1}), \tag{4.32}$$

$$P_{t|t} = P_{t|t-1} - K_t H_t P_{t|t-1}, \tag{4.33}$$

em que $K_t = P_{t|t-1}H_t^T(H_t P_{t|t-1}H_t + R_t)^{-1}$ e z_t é uma observação (ou medição) do estado real no tempo t.

Em outras palavras, o algoritmo funciona fazendo a média de uma previsão da posição do sistema com uma nova medição, utilizando uma média atribuído a partir da covariância das medições. O resultado da média ponderada é uma nova estimativa de estado que se encontra em algum lugar entre o estado avaliado e medido. Este processo é repetido a cada intervalo de tempo, com a nova previsão e a sua covariância correspondente alimentação de volta para a previsão utilizadas na iteração seguinte. Portanto, o filtro de Kalman é um algoritmo recursivo que requer apenas a última estimativa em vez de toda a história da estimativa para prever o novo estado, e esse procedimento consiste em duas fases principais: a predição da posição ou velocidade dos objetos de escopo e as correções ou atualizações dessas variáveis.

As vantagens na utilização do filtro de Kalman estão na redução das áreas de análise, uma vez que se dispõe de uma boa estimativa inicial do local em que cada elemento pode ser encontrado no momento seguinte; menor custo computacional no método de correspondência; cálculo de estimativas de posição e velocidade, com as respectivas medidas de incerteza. Segundo CORREIA (1995) a desvantagem é a complexidade do cálculo das matrizes de covariância a exigir maior capacidade de armazenamento para essas matrizes. Uma descrição completa e abundante minúcias do filtro de Kalman pode ser obtida na composição de Maybeck et al. (1982).

4.6.2 Associação de Dados

Conforme Grisetti e Arras (2009), a associação de dados é o processo de associar medições incertas com as *tracks* conhecidas. Este problema engloba uma série de questões relacionadas: criação de *tracks*, manutenção e eliminação, único ou múltiplos sensores, detecção de alvos único ou múltiplos etc.

No contexto desse trabalho as variáveis de interesse (estados) que que caracterizam o comportamento do sistema podem estar sujeitas as: incertezas de medição e modelo, pertubações por ruídos ou outras interferências durante a medição, podem ser não observáveis (precisam ser estimadas) e sofrerem oclusões etc.

Desse modo, para obter uma estimativa ótima (minimização de erro) das variáveis de interesse (estados) do sistema no contexto do acompanhamento visual, optou-se por um procedimento de associação de dados por meio de um método para rastreamento de múltiplos alvos propostos por Welch (2014), que abordou o método de Murty (conforme o algoritmo ilustrado no apêndice A) otimizado para o rastreamento de múltiplos alvos, em que são classificadas todas as atribuições em ordem crescente de custo, esperando-se alcançar, consequentemente, um melhor acoplamento entre a *track* e a medição. Isto é, o método relaciona cada um dos objetos de interesse nos respectivos *frames* subsequentes alimentando o filtro de Kalman, sob uma dada consideração.

4.7 CLASSIFICADOR *k*-NEAREST NEIGHBORS

O k-Nearest Neighbors é o mais elementar algoritmo de *Machine Learning* e corresponde a um dos arquétipos do apredizagem indutivo. É um algoritmo não paramétrico baseado em memória, utilizado em problemas de classificação, capaz de classificar um novo objeto em relação a um conjunto de objetos referenciais pré-agrupados segundo as semelhanças entre eles. Em outras palavras, se tomarmos um problema de classe com valores discretos em um conjunto de dados discreto, cada vizinho vota em uma classe e, assim, o objeto-alvo é classificado na classe mais votada.

De forma geral, segundo Mitchell (1997) o k-Nearest Neighbors constrói aproximações locais da função objetiva, diferente para cada novo dado a ser classificado. Essa característica pode ser vantajosa quando a função objetiva é muito complexa, mas ainda pode ser descrita por uma coleção de aproximações locais de menor complexidade.

A fase de treinamento requer pouco esforço computacional. No entanto, classificar objetos de teste requer a distância desse objeto a todos os objetos de treinamento. Segundo Azevedo et al. (.2 ed.2008), as mais utilizadas são a distância euclidiana e distância Manhattan (ou distância bloco-cidade). Esta última é a soma simples dos componentes horizontais e verticais, enquanto a primeira pode ser calculada através da aplicação do teorema de Pitágoras. Outra comumente usada é a distância de Mahalanobis, que se baseia nas correlações entre variáveis

nas quais distintos padrões podem ser identificados e analisados.

Porém, os cálculos com base em distância para a predição podem ser dispendiosos. Além disso, todos os algoritmos baseados em distância podem ser afetados pela presença de atributos redundantes e irrelevantes. Outro problema com o k-NN está relacionado à dimensionalidade dos exemplos. Com o aumento da dimensionalidade, a distância dos vizinhos mais próximos aproxima-se da distância ao vizinho mais afastado. Uma das formas para reduzir o impacto da dimensionalidade consiste em selecionar um subconjunto de atributos relevantes para o problema tratado (FACELI, 2011).

Em problemas de classificação em que o problema toma valores discretos o processo é definido formalmente conforme o equivalente na Equação (4.34), o que é justificado porque a constante que minimiza a função custo 0-1 é a moda (FACELI, 2011).

$$\hat{f}(x_t) \leftarrow Moda(f(x_1), f(x_2), .., f(x_k)). \tag{4.34}$$

Em problemas de regressão, podem ser utilizadas duas estratégias, dependendo da função de custo usada. Se a função de custo for minimizar o erro quadrático, a média dos valores obtidos para cada um dos k vizinhos deve ser utilizada, o que pode ser formalmente definido na Equação (4.35).

$$\hat{f}(x_t) \leftarrow Média(f(x_1), f(x_2), .., f(x_k)). \tag{4.35}$$

Já no caso em que a função de custo for minimizar o desvio absoluto, deve ser atualizada a mediana em vez da média. Nesse caso, a função passa a ser como na Equação (4.36) .

$$\hat{f}(x_t) \leftarrow Mediana(f(x_1), f(x_2), .., f(x_k)). \tag{4.36}$$

A justificativa para esses procedimentos é que a média é a constante que minimiza o erro quadrático, enquanto a constante que minimiza o desvio absoluto é a mediana (FACELI, 2011).

Em Aha et al. (Jan. 1991), são apresentados dois algoritmos para selecionar os objetos mais relevantes, de forma a reter em memória apenas esses objetos. As versões são *Edit*

k-NN para eliminações e inserções sequenciais, respectivamente representadas pelos Algoritmos 1 e 2, são dois exemplos que armazenam apenas protótipos.

Algoritmo 1: Algoritmo para *Edit k*-NN - Eliminação sequencial

Entrada: Um conjunto de treinamento $D = \{(x_i, y_i), i = 1, 2, ..n\}$

Um conjunto de treinamento $D' = \{(x_i, y_i), i = 1, 2, ..m; m < n\}$

início

  para *cada conjunto de exemplo* (x_i, y_i) **faça**

    (x_i, y_i) é corretamente classificada por $D \mid (x_i, y_i)$ /* Remove (x_i, y_i) de D */;

    $D \leftarrow D \mid (x_i, y_i)$

  fim

fim

D

Algoritmo 2: Algoritmo para *Edit k*-NN - Inserção sequencial

Entrada: Um conjunto de treinamento $D = \{(x_i, y_i), i = 1, 2, ..n\}$

Um conjunto de treinamento $D' = \{(x_i, y_i), i = 1, 2, ..m; m < n\}$ $D \leftarrow \{\}$;

início

  para *cada conjunto de exemplo* (x_i, y_i) **faça**

    (x_i, y_i) é incorretamente classificada por $D' \mid (x_i, y_i)$ /* Acrescenta (x_i, y_i) em D' */;

    $D' \leftarrow D' \cup \{\}(x_i, y_i)\}$

  fim

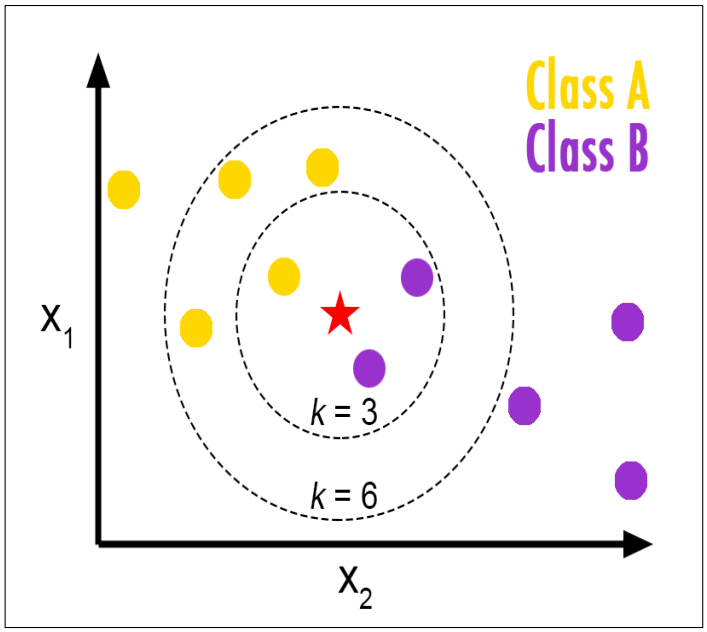
fim

D'

A seguir é apresentado um exemplo ilustrativo para diferentes valores para k para o algoritmo *k*-NN. Defina-se um simples conjunto de treinamento que consiste em duas classes (A e B) conforme a representação na Figura 11, com cinco casos cada, conforme indicado pelos círculos de cores amarela e roxo. Apenas duas características (X_1 e X_2) são usadas para discriminar entre as classes, de forma característica, e o espaço é 2-dimensional. Considere uma observação não marcada, indicada pela estrela vermelha, para classificá-la como A ou B. Assim, o algoritmo *k*-NN atribuirá classe pelo voto da maioria dos k vizinhos mais próximos. Para o caso k = 3 (pequeno círculo), um vizinho é de classe A e dois são da classe B, portanto, classificar a observação não marcada como um membro do B; para k = 6 (círculo grande), no entanto, quatro vizinhos são de Classe A, e apenas dois são da classe B, de modo que a observação não

marcada em vez disso é classificada como um membro de A.

Figura 11 – k-Nearest Neighbors



Fonte**:** http://bdewilde.github.io/blog/blogger/.

4.8 EXTRAÇÃO DAS CARACTERÍSTICAS PARA INSTÂNCIAS

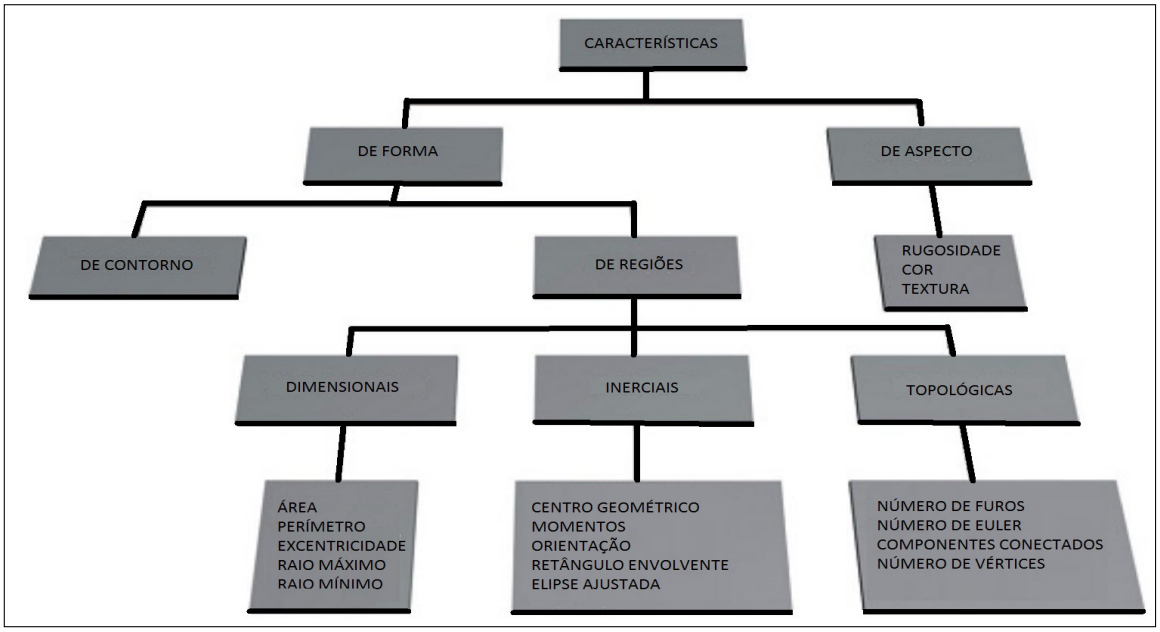
A extração de características ou de medidas dos objetos na imagem é um processo primordial na segmentação de imagem, pois permite a representação das características que os compõem, o que viabiliza a descrição do conteúdo da imagem.

O fundamento básico para escolha da melhor abordagem para representar as características dos objetos depende do domínio do problema, uma vez que, existem outros pormenores problemas práticos como: a presença de ruído, a oclusão, as transformações geométricas, a rotação, translação etc.

Em Azevedo et al. (.2 ed.2008) são apresentados duas principais linhas de abordagens: de forma e de aspecto (ilustrado na Figura 12). Os descritores de aspecto são estruturas importantes que utilizam os *pixels* para classificar e reconhecer objetos em imagens. Normalmente estão associados à impressão de rugosidade e contraste que surge devido a variação tonal ou pelas variações locais em valores de *pixels* que se seguem de modo regular ou estocástica por toda a extensão do objeto ou sobre uma região da imagem. Enquanto os descritores de forma são estruturas adequadas de representação e facilitam o armazenamento e a manipulação dos

objetos segmentados da imagem, além de simplificarem o cálculo de certos descritores de região, os *pixels* localizados no interior da região ou objeto são considerados no cálculo do descritor, em vez de utilizar os *pixels* que formam a borda da região.

Figura 12 – Classe de Características



Fonte**:** Computação gráfica - Teoria e prática, volume 2, AZEVEDO, Eduardo; CONCI, Aura; LETA, Fabiana.

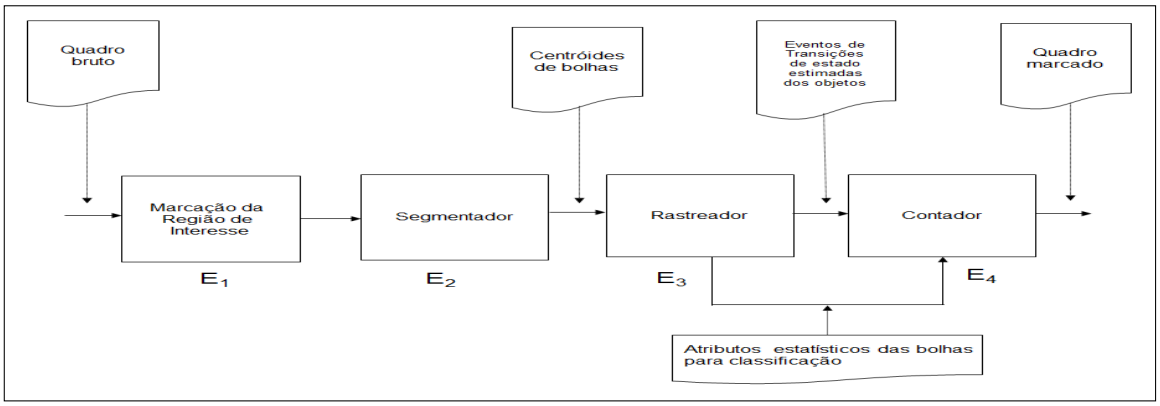
5 METODOLOGIA

Neste capítulo serão descritos detalhes do protótipo desenvolvido. A abordagem proposta advém de uma linha de pesquisa denominada Visão Computacional, que desenvolve um conjunto de métodos, técnicas e tecnologias para a construção de sistemas computacionais artificiais, os quais podem ser capazes de interpretar imagens.

Em concordância com Valle (2007), um sistema de visão completo envolve quatro etapas básicas: captura e pré-processamento de vídeo, segmentação de objetos de interesse, rastreamento de objetos de interesse, contagem de pessoas dentro de uma região delimitada. Seguindo a mesma estrutura, o protótipo dessa pesquisa é apresentado e representado em uma estrutura semelhante, composta por quatro blocos ou etapas, de acordo com a ilustração na Figura 13.

O fluxo sequencial de execução do sistema se inicia após a aquisição da imagem por meio de um dispositivo de entrada, isto é, o processo inicial começa com a ativação de função que constrói um leitor multimídia objeto que pode ler dados de vídeo de arquivo multimídia. Na sequência, dar-se-á a inicialização do sistema desenvolvido de maneira cíclica com a extração dos quadros em nível de RGB (Red, Green, Blue).

Figura 13 – Diagrama de fluxo da metodologia proposta



Fonte**:** Próprio autor

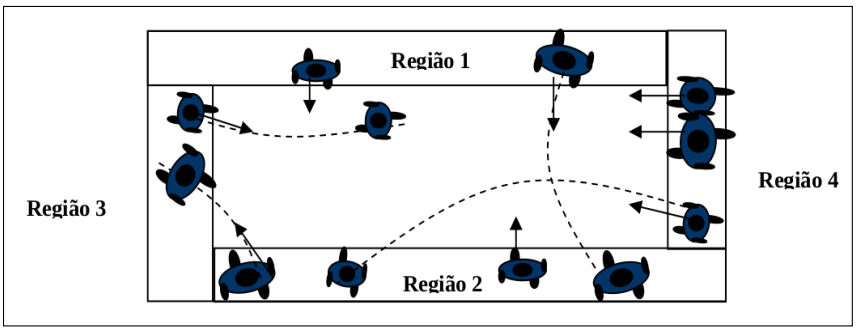
Uma inicialização do sistema começa com a especificação de todos os parâmetros dos métodos abordados para viabilizar o bom funcionamento do sistema. Desta forma, é realizada uma calibração, que consiste em determinar todos os parâmetros, uma vez que o sistema exige a interposição humana para regularização dos valores desses parâmetros com o propósito único de alcançar maior qualidade dos resultados. Assim, esses valores dos parâmetros são ajustados de

acordo com uma pré-análise das variáveis de ambiente de maneira empírica em alguns segundos de vídeo, pois cada ambiente monitorado possui suas próprias peculiaridades.

Desse modo, na etapa E_1, inicia-se o estruturador semiautomático de regiões fronteiriças (considere semiautomático pelo fato de o usuário poder adaptar essas fronteiras de acordo com a estrutura da passagem de circulação dos pedestres), que divide uma área virtualmente em sub-regiões proporcionais no formato de "cata-vento", denominadas de regiões de interesse ou Region Of Interest (ROI), conforme na Figura 13. Isto viabiliza a identificação do sentido de movimento da pessoa por meio das suas transições entre essas zonas de fronteiras.

Segundo Silva (2008), a abordagem fundamenta-se na premissa que nenhuma região é horizontalmente dominante ou verticalmente dominante, evitando que uma pessoa atravesse o campo de visão da câmera sem mudar de região de fronteira. Como a entrada no campo de visão da câmera ocorre somente pelas regiões de fronteira, apenas essas regiões da imagem precisam ser monitoradas em busca de novas pessoas na cena, o que contribui para baixar o custo computacional do processo, sem a redução da confiabilidade.

Figura 14 – Representação do local monitorado por uma câmera de vídeo com quatro regiões de fronteira



Fonte**:** Elaborado por Silva (2008)

A etapa E_2 consiste no processamento sequencial das imagens através de um modelo de processo *pixel*, que consiste em um modelo adaptativo estatístico de detecção de movimento proposto por Stauffer e Grimson (1999) ou modelo processo /textitpixel apresentado na seção 3. Em resumo, é um tipo de modelo que tem seus parâmetros constantemente atualizados e, com isso, torna-se capaz de mensurar as variações de intensidade dos *pixels* e, a partir dessas variações, os *pixels* são rotulados em duas classes: a *background* ou *foreground*. Assim, como produto final, o processo retorna uma imagem binária (máscara de bolhas) para cada quadro processado.

A imagem binária resultante do processo de classificação dos *pixels* pode apresentar alguns erros morfológicos nos objetos contido nela, tais como: ruídos, contornos serrilhados e componentes isolados. Para tentar corrigi-la, são aplicadas operações morfológicas. Frequentemente, são utilizados os operadores de erosão e dilatação (filtros morfológicos) na imagem binária em uma única passagem, com o objetivo principal de obter a fronteira exata do objeto em movimento. O desempenho na fase de segmentação refletirá diretamente na contagem, pois, em casos críticos, a extração de características causa uma propagação sistemática de erros nas heurísticas dos processos sequentes.

Depois de completar o processo de correção da imagem binária, inicia-se a extração de características por meio de técnicas de análises de bolhas. Essa técnica consiste em um conjunto diversificado de métodos de análise das regiões mais consistentes, cujo objetivo é extrair as características físicas dos objetos individualmente mediante descritores de forma ou aspecto. Segundo Mukherjee e Das (2013b), concluiu-se que existem diferentes influências dos descritores como parâmetros e que nenhum deles pode individualmente detectar seres humanos, portanto sugerem que o peso da decisão deve levar em conta o bom desempenho do sistema.

Os descritores neste trabalho são utilizados como instâncias para o classificador padrão abordado, este será descrito no final do processo. Entre as diferentes abordagens de descritores, optou-se pelo descritor de forma. A escolha se baseia no fato do descritores de aspecto não apresentarem um padrão linear, isto é, devido as variações de texturas, cores, rugosidades etc. Desse modo, foram selecionados cinco descritores de forma, tais como: a área, o perímetro, a compacidade, a largura máxima e a largura mínima.

Como estratégia técnica, apenas a parte superior da bolha com as mesmas delimitações relativas que envolve a bolha (*bounding box* - caixa delimitadora que envolve a bolha) é analisada com a finalidade de extrair os descritores já mencionados, podendo ser visualizada na Figura 15 cuja a relação da bounding box é dada pela Equação 5.1 . A detecção da região superior da bolha frequentemente corresponde à região que engloba a cabeça e tronco de uma pessoa adulta, o que minimizará o impacto de colisões por aproximação entre pessoas, como também evita a necessidade de utilização das técnicas de extração de sombra, pois níveis adicionais de processamento exigem mais recursos de memória e tempo de processamento.

$$[x, y, largura \times bw, altura \times bh], \tag{5.1}$$

Figura 15 – Ilustração bounding box da região superior

Fonte**:** Próprio autor

em que, bw e bh são índices de proporcionalidade empíricos e estão no intervalo [0,1].

A etapa E_3 é a de rastreamento das bolhas, em que são utilizados os seus centróides. De maneira geral, para cada objeto que adentra em cada uma das regiões, no primeiro instante que é identificado, é incoado o rastreamento do objeto, que o persegue até chegar na demarcação fronteiriça oposta àquela em que o objeto adentrou seguidamente até que deixe o campo de visão da câmera. Para efeito visual, cada objeto na imagem é delimitado por uma *bounding box* que envolve o objeto-alvo detectado, como por exemplo, um retângulo, que delimitará a área relativa a cada objeto na imagem analisada.

Esta etapa é composta por três subprocessos: filtro preditivo de estimação de posição, algoritmo de associação de *tracks* e analisador de fluxo direcional (estrutura que armazena o sentido do movimento de cada objeto ao atravessar o campo de visão da câmera).

No presente trabalho, foi abordado o mesmo método de rastreamento de Silva (2008), mais especificamente o filtro de Kalman (KALMAN, 1960), que prevê a localização do objeto em cada *frame*, de tal forma que é calculada uma probabilidade de cada detecção ser associada corretamente ao respectivo objeto ao longo dos *frames*.

O filtro de Kalman não se comporta de forma multimodal, ou seja, cada filtro é capaz de representar apenas uma estimativa, desse modo, a cada novo objeto ou pessoa detectada, é necessária a implementação de um novo filtro. Uma vez que, cada objeto ou pessoa está associado a um determinado filtro, surge um segundo problema, que é como associar estes filtros quadro a quadro. Neste sentido, é utilizado um sistema de gestão de rastreamento de múltiplos alvos capazes de associar vários objetos ou pessoas nos subsequentes quadros do vídeo. É utilizado o método para rastreamento de múltiplos alvos proposto por Welch (2014).

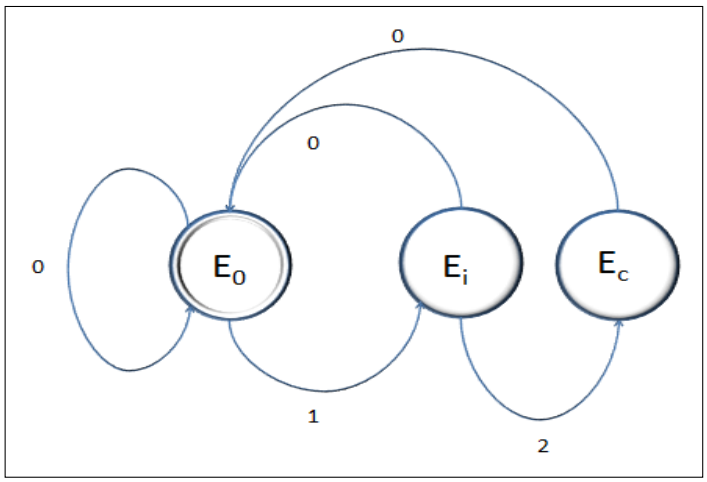
O rastreamento em imagens sequenciais consiste em localizar o objeto-alvo com base na posição (coordenadas (x, y) do centróide do conjunto de *pixels* associado ao objeto-alvo) e velocidade no instante t_i no quadro atual, que é comparado às ocorrências desse suposto objeto-alvo na área estimada no próximo quadro, no instante t_{i+1}. Para fazer com que o rastreamento seja feito de maneira correta, foi criada uma estrutura que armazena características para cada objeto em primeiro plano, sendo essa estrutura chamada *track* ou rastro, assim, quando o objeto fica, por vários *frames*, sem aparecer na cena analisada, então o seu rastro é apagado, sendo criadas novas estruturas *track* para rastrear os novos objetos que possam aparecer na cena. Portanto, após a previsão das novas detecções (em próximo quadro do vídeo) dão-se as medições (que são utilizadas na comparação das medições com as previsões) de modo que os *tracks* possam ser associados, não associados, deletados ou criados novos.

Durante o rastreamento existe um subprocesso em paralelo sendo executado, denominado analisador de fluxo direcional, que memoriza o sentido do fluxo das pessoas dentro da ROI, o que será útil posteriormente para contagem considerando a direção. Esse subprocesso tem papel fundamental na identificação contínua da localização do objeto em trânsito nas diferentes sub-regiões utilizando as coordenadas dos centróides, obtidas pelas estimativas do rastreador de filtro de Kalman a cada *frame*, como também, no controle das regras que operam na informação dinâmica com base no principio de máquina de estado.

Segundo Black (2008), uma máquina de estados é um modelo que correlaciona entradas e saídas. O conceito é concebido como uma máquina abstrata que deve estar em um de seus finitos estados. A máquina está em apenas um estado por vez, que é chamado de estado atual. Um estado armazena informações sobre o passado, isto é, ele reflete as mudanças desde a entrada em um estado, no início do sistema, até o momento presente. Uma transição indica uma mudança de estado e é descrita por uma condição que precisa ser realizada para que a transição ocorra.

O papel fundamental do analisador de fluxo direcional é estimar as transições entre regiões para todos objetos em movimento na região de interesse, em que os estados são as condições sobre as quais se aplicam as regras de contagem. Quando um objeto chega a um determinado estado, é verificada a condição para contar ou não; enquanto as transições são as mudanças de regiões do objeto em trânsito. Essas relações entre estes estados são ilustradas na Figura 16.

Figura 16 – Abstrações de Estados e Transições



Fonte**:** Próprio autor

O fluxo de objeto em cada quadro é tratado como máquina de estado finito, que consiste em três estados: E_0 é estado inicial ou 0, onde ocorre a localização da zona de fronteira da primeira detecção do objeto na cena, ou seja, uma região de fronteira (de entrada) e nenhuma transição para a zona central. E_i é o estado intermediário, em que objeto a partir de uma zona fronteiriça adentra a zona central; realizando a primeira transição e E_c é o estado contável, quando o objeto adentra uma zona de fronteira diferente da região de entrada a partir da zona central, realizando assim a segunda transição.

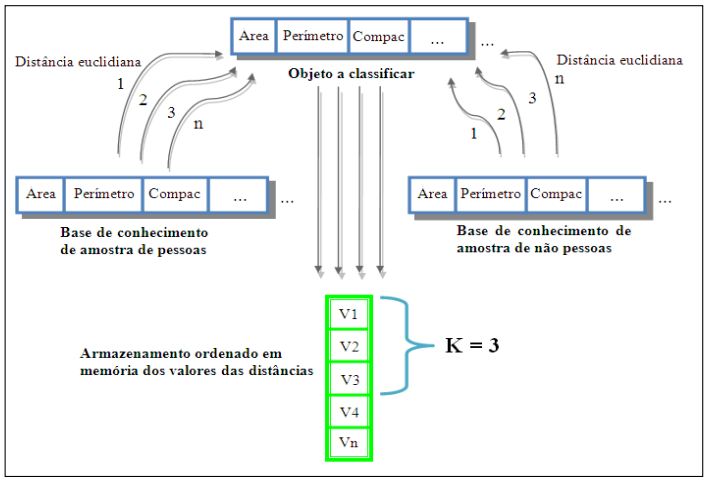
A quarta e última etapa E_4 consiste na contagem de pessoas nas bolhas baseada em classificador padrão não-paramétrico, abordagem esta, utilizada também por Valle Valle (2007), mais especificamente denominada de algoritmo de *k-Nearest Neighbors* (*k*-NN) (FUKUNAGA; NARENDRA, 1975). A escolha se deu por ser um método amplamente conhecido e utilizado, principalmente quando se tem pouco ou nenhum conhecimento antecipado sobre a distribuição dos dados.

O *k*-NN é um método que utiliza métricas de distância. Neste trabalho optou-se pela distância Euclidiana entre as instâncias construídas a partir dos descritores geométricos. Deste modo, a construção do classificador é obtida por meio de um conjunto de treinamento (base de conhecimento) representado por vetores compostos por descritores extraídos de amostras de objetos que, por meio deles, obtêm-se as classes que serão abordadas para classificações do objetos, de modo que os *k* mais próximos vetores de atributos de referência irão rotular cada vetor do conjunto das novas amostras com as possíveis classes, que são dados pela votação (voto da maioria) de cada associado do conjunto amostra.

Segundo Zaccariotto (2010), a decisão de qual classe pertence o objeto analisado depende diretamente do valor definido para *k*, assim, quando se tratar de um resultado binário (pessoa ou não pessoa), o autor utilizou um valor ímpar, dessa forma evitando a possibilidade de empate entre as classes. Para atribuir o valor de k, sugere-se realizar alguns testes informais visando encontrar o melhor valor.

Na fase de treinamento, após as capturas das características das regiões superiores dos objetos que comporão os vetores de características, segue a rotulação dos objetos de modo manual, isto é, sob a perspectiva visual de um ser humano. É analisado se os objetos são pessoas ou não. Deste modo, um objeto pode ser rotulado "0" se for composto por não pessoas e por "1" , "2" ou "3" se for composto respectivamente por uma, duas ou três pessoas. Assim, em resumo, o processo de classificação é dada comparando o novo objeto em cena com a base de conhecimento, conforme a representação na Figura 17.

Figura 17 – Processo *k*-NN



Fonte**:** Elaborado por Zaccariotto (2010)

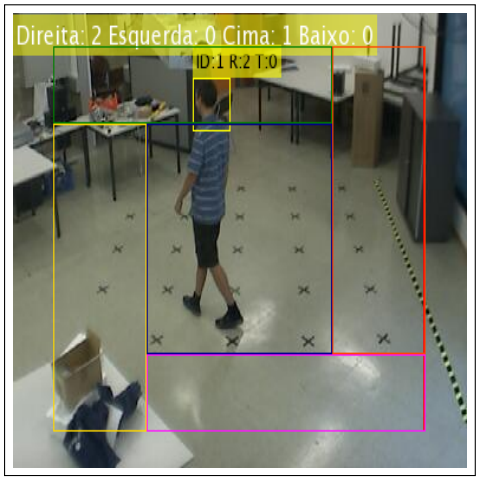
Genericamente, com base na saída do classificador, é estimado o número de pessoas que compuseram a bolha (vale ressaltar que uma bolha pode ser composta por não pessoas, uma pessoa ou grupos de pessoas aglomeradas) e posteriormente o contador é incrementado com valor correspondente ao rótulo da classe. Finalmente, subsequente ao bloco processado, os dados são armazenados em uma estrutura e as informações visuais são geradas.

De modo mais minucioso, o contador é regido por regras de estado, isto é, ele é ativado quando o objeto faz mais duas transições (utilizando o analisador de fluxo direcional) dentro do ROI. O número de pessoas contabilizadas é incrementado quando o classificador, que está contido no contador, realiza a classificação da bolha detectada. Ou seja, a relação

entre o contador e o classificador é dada do seguinte modo: à medida que os novos objetos vão abandonando o ROI, o classificador é excetuado, seguidamente o contador é incrementado com o mesmo valor correspondente ao seu rótulo dado pelo classificador (conforme o pseudocódigo ilustrado em apêndice A). Vale salientar que, os objetos-alvos que entrarem e retornarem pela mesma região que entraram, estes não serão computados por não terem completados as transições segunda a regência da máquina de estado apresentada.

Na Figura 18 é ilustrado o exibidor de resultados de contagem de pessoas ao longo do tempo na região de interesse. Como se pode verificar, o fluxo das pessoas é categorizado em 4 direções sob a perspectiva do usuário que assiste à cena, sendo elas: direita, esquerda , baixo e cima.

Figura 18 – Ilustração do exibidor de contagem



Fonte: Instituto de Computação Gráfica e Visão na Universidade Tecnológica de Graz

Para realizar a avaliação dos métodos, utilizou-se uma base de dados de vídeos que possuem duas visões diferentes da mesma cena (com exceção de um vídeo, que tem apenas um visão da cena e difere o ambiente dos demais), por isso é imprescindível que as bases tenham mais de um vídeo, uma vez que também é realizado o treinamento do algoritmo de reconhecimento padrão. A avaliação dos métodos será realizada efetuando a comparação das estimativas alcançadas com o *Ground Truth* gerado de forma manual.

6 EXPERIMENTOS

Neste capítulo serão apresentados as configurações das ferramentas utilizadas, o critério de avaliação, os detalhes da base de dados, os métodos, os procedimentos e os resultados experimentais, e por fim, as considerações.

6.1 AMBIENTE E BASES DE DADOS

Para a realização dos experimentos, a ferramenta foi executada em um computador com a seguintes configurações: processador Intel Core i7-3630QM CPU @ 2.40GHz x 8, *kernel* de 6MB de cache, 8GB de memória RAM DDR3 1600 MHz, com sistema operacional Linux Ubuntu 14.04 LTS-64bits. A implementação do sistema do modelo experimental foi totalmente desenvolvido em uma linguagem de programação iterativa de quarta geração denominada MATLAB (*matrix laboratory*) 2013a *Student Version*.

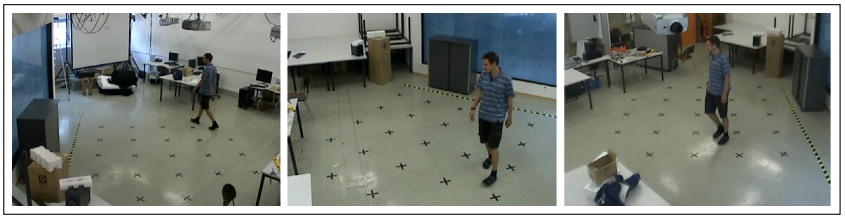
Neste trabalho, para a análise quantitativa dos métodos foi utilizado o *Multi-Camera Datasets*, um conjuntos de amostras testes de vídeos da base de dados disponibilizado via internet pelo grupo LRS (*Learning, Recognition, and Surveillance*) do Instituto de Computação Gráfica e Visão na Universidade Tecnológica de Graz (2008), ambos filmados com variadas visões da mesma sequência e ângulos de filmagem oblíquo e, cujo formato dos vídeos é AVI em RGB.

Os vídeos da LRS foram capturados em um ambiente de laboratório com três distintas visões de uma mesma sequência e ambiente. Em cada vídeo uma ou mais pessoas circulam pela cena simulando a entrada e saídas de pessoas pela região de interesse previamente ajustadas. As sequências utilizadas foram todos vídeos categorizados na fonte de origem como: *Easy Data Set* (EDS) com apenas uma pessoa e *Medium Data Set* (MDS) com três pessoas, conforme nas Figuras 19, 20 , 21 e 22. Cada vídeo de teste com 2589 quadros, a uma taxa de 30 quadros por segundos e resolução de 384×288. A abordagem limitou a detectar apenas a região superior das bolhas que frequentemente inclui a região do ombro à cabeça das pessoas.

Figura 19 – Vídeos de testes EDS-1 (Visão_001,Visão_002 e Visão_003)

Fonte**:** Vídeo de referência do *LRS* da Universidade Tecnológica de Graz.

Figura 20 – Vídeos de testes EDS-2 (Visão_004,Visão_005 e Visão_006)



Fonte**:** Vídeo de referência do *LRS* da Universidade Tecnológica de Graz.

Figura 21 – Vídeos de testes EDS-3 (Visão_007,Visão_008 e Visão_009)

Fonte**:** Vídeo de referência do *LRS* da Universidade Tecnológica de Graz.

Figura 22 – Vídeos de testes MDS (Visão_010,Visão_011 e Visão_012)

Fonte**:** Vídeo de referência do *LRS* da Universidade Tecnológica de Graz.

Além dos vídeos citados, também foi utilizado um vídeo de demonstração da companhia Mathworks capturado em um ambiente real com uma única visão sob uma iluminação artificial, com 431 quadros numa taxa de 30 quadros por segundos e resolução 360 x 640, conforme a ilustração na Figura 23 .

Figura 23 – Vídeo Atrium (Visão_013)

Fonte**:** Vídeo de referência do *demo* da companhia Mathworks

6.2 MEDIDAS DE AVALIAÇÃO

Para avaliação do desempenho em termo de poder de predição do classificador e contador foi empregado a matriz de confusão, cujo aspecto da acurácia do modelo gerado será a medida dada pela taxa de acerto médio, especificidade e sensibilidade. Além disso, para validação e estimação da acurácia do teste no contexto de treinamento do classificador foi utilizado o método *r-fold cross-validation*.

Matriz de confusão: por simplicidade, seja um problema com duas classes. usualmente, uma classe é denotada positiva(+) e a outra negativa(-). Tem-se então a matriz de confusão, conforme ilustrada na Tabela 1, em que:

- VP corresponde ao número de verdadeiros positivos, ou seja, o número de exemplos da classe positiva classificados corretamente;
- VN corresponde ao número de verdadeiros negativos, ou seja, o número de exemplos da classe negativa classificadas corretamente.
- FP corresponde ao número de falsos positivos, ou seja, o número de exemplos cuja classe verdadeira é negativa , mas que foram classificados incorretamente como pertencendo à classe positiva.
- FN corresponde ao número de falsos negativos, ou seja, números de exemplos pertencentes originalmente à classe positiva que foram incorretamente preditos como classe negativa.

Tabela 1 – Matriz de confusão

	+	-
+	VP	FN
-	FP	VN

Fonte:Próprio autor.

As medidas posteriores podem ser facilmente generalizadas para problemas com mais de duas classes.Todas estas medidas indicam resultados entre 0 e 1, pelo que quanto mais próximo de 1 as medidas forem, melhor é a avaliação.

- *Taxa de acerto ou acurácia total*: calculada pela soma dos valores da diagonal principal da matriz, dividida pela soma dos valores de todos elementos da matriz.

$$ac(\hat{f}) = \frac{VP+VN}{n} \tag{6.1}$$

- *Sensibilidade ou revocação*: corresponde à taxa de acerto na classe positiva. Também aqueles preditos como positivos por $\hat{f}$.

$$sens(\hat{f}) = \frac{VP}{VP+FN} \tag{6.2}$$

- *Especificidade*: corresponde à taxa de acerto na classe negativa.

$$esp(\hat{f}) = \frac{VN}{VP+FP} \tag{6.3}$$

Método de validação cruzada: consiste em dividir aleatoriamente os dados da amostra teste em r subconjuntos com a proporção de exemplo de cada classe semelhante à proporção contida no conjunto de dados total. Os objetos de r-1 partições são utilizados no treinamento de um preditor, que é então testado na partição restante. Esse processo é repetido r vezes, utilizando em cada ciclo uma partição diferente para teste. O desempenho final do preditor é dado pela média dos desempenhos observados sobre cada conjunto de teste.

6.3 PROCEDIMENTOS EXPERIMENTAIS

Nesta seção são apresentadas as questões relativas ao desenho da experiência e à avaliação do modelo preditivo. A extração das características do objetos de interesses na imagem depende da calibração dos métodos sequenciais na etapa de segmentação (algoritmos de misturas gaussianas, etc), que consiste na combinação empírica dos parâmetros das técnicas utilizados, ou seja, um ajuste correto desses parâmetros possibilita a obtenção de uma melhor precisão da estimação das métricas abordadas, em consequência, uma melhor acurácia para o modelo de classificador.

Uma base de treinamento foi gerada a partir base de conjunto vídeo *Hard Data Set* (HDS), seguidamente é feito a seleção de uma amostra aleatória simples sob o universo de bolhas detectadas nos vídeos de treinos para posterior treinamento do algoritmo de classificação. Foram

coletados as partes superiores da bolha, que confere uma região pré-demarcada retangular que coincida com a região do ombro de uma pessoa ou mais pessoas até o ápice de suas cabeças na posições ereta habituais. Desta forma, foram coletados 742 amostras de imagens da região superior de pessoas em diferentes ângulos, subdivididas igualmente em três classes rotuladas por "1" , "2" ou "3" , isto é, se contiverem respectivamente uma, duas pessoas ou três na bolha extraída. Além disso, foram coletados 176 objetos aleatórios compostos por não pessoas (animais, pequenos objetos, mobílias, figuras geométricas e ruídos aleatórias de imagens, etc), sendo assim criado uma terceira classe de objetos com rotulo "0".

O algoritmo *k*-NN pressupõe que o conjunto de treinamento é composto pelas variáveis descritivas e pela sua classificação; o *k*-NN então utiliza tais variáveis para classificar um novo item (SU, 2011). Assim, para a classificação do número de pessoas contidas nas bolhas é feito uma pré-avaliação estatística para a seleção do melhor parâmetro *k* para classificador, sendo utilizados as métricas de 5 descritores de formas, obtidas a partir das análises de bolhas, cujas medidas selecionadas de cada objeto definem a estrutura do vetor de características, que serão as instâncias para o algoritmo *k-NN*. Com base na distância entre essas instâncias, os *k*-vizinhos mais próximos são identificados e, com base em um *k* escolhido, uma nova observação é atribuída à classe com maior número de observações em *k*.

Após o treinamento do classificador *k*-NN, uma avaliação estatística foi gerada a partir de um teste sob alguns segundos do vídeo de treinamento que foi resguardado no sentido de não extraído amostras para essa eventual validação. Deste modo, decidiu-se o melhor valor para o parâmetro *k* do classificador por meio da validação cruzada. Os resultados das classificações dadas são comparados com a classe rotulada à partir da análise visual humana que define as classes em que os objetos pertencem. Essa análise visual é efetuada sob as amostras de bolhas coletadas e é elaborada do seguinte modo: cada elemento da amostra de bolhas é estratificado visualmente em subgrupos significativos (estratos) e anexada ao seu vetor de característica correspondente.

A avaliação estatística para a escolha do valor *k* pode ser resumida nas seguintes tarefas:

- Tarefa 1 - Um vídeo de treinamento é gerado, e para cada objeto detectado, é capturado e comprimido em uma imagem do tipo JPEG (Joint Photographic Expert Group), sendo nomeado com a uma chave de identificação única que associa as medidas métricas obtidas pelo analisador de bolha (descritores de formas) salvo em um arquivo texto;

- Tarefa 2 - A partir de uma análise sob perspectiva subjetiva visual de um ser humano, cria-se um conjunto de treinamento, isto é, realizam-se a definição das classes e escolha da etiquetagem dos objetos capturados;
- Tarefa 3 - Dado o conjunto de treinamento com suas medidas padronizados para *z-scores* (evita que uma dimensão se sobreponha em relação às outras e que o aprendizado fique estagnado), aplica-se a técnica *r-fold cross-validation* para estimar o desempenho do algoritmo de aprendizado de máquina *k*-NN;
- Tarefa 4 - Simulam-se diferentes valores ímpares para *k*, e escolhe o melhor parâmetro *k* para realização das etapas seguinte (a cada a nova instância a ser classificada deve-se padronizar para *z-scores* relativo a amostra de treinamento).

  O escopo da normalização é para minimizar os problemas oriundos do uso de unidades e dispersões distintas entre as variáveis, uma vez que, isto previne que uma dimensão se justaponha em relação às outras, precavendo desta maneira uma possível estagnação do aprendizagem. A técnica de normalização utilizada neste trabalho foi a *Z-score* (6.4), em que, os dados são normalizados no entorno da média e do desvio padrão ficando com média igual 0 e variância igual a 1.

$$Z_x = \frac{x - \mu_x}{\sigma_x} \tag{6.4}$$

  onde o Z_x é o valor novo para um determinado número x que será normalizado, μ_x é a média e σ_x é desvio padrão.

  Para que aplicação seja possível, foi feito um pré-processamento dos dados, de forma a obter a média e o desvio padrão do conjunto dos valores para cada uma das características extraídas dos objetos em cena. Posteriormente, para cada novo objeto é feito a normalização dessas novas medidas utilizando as estimativas obtidas.

Com o classificador treinado e os demais parâmateros do sistema ajustado, dar-se-à realização da contagem de pessoas, que é feita calculando a soma acumulada de pessoas que atravessam o campo de visão da câmera pelo sistema, na qual é comparado com o *Ground Truth* (GT), que é gerado manualmente. Assim, a contagem de pessoas por fluxo multidirecionais são realizadas tanto por GT gerado manualmente (analisados manualmente sob perspectiva visual humana), como também, de modo automático pelo sistema (contagem baseada por classificador).

Com a finalidade de avaliar o desempenho do sistema de contagem automática, essas duas contagens são comparadas para obter o número de acertos e erros, uma vez que, estes serão utilizados para gerar as métricas de desempenhos já mencionadas.

6.4 RESULTADOS EXPERIMENTAIS

Nesta seção é feita uma descritiva dos experimentos realizados e a apresentação dos resultados quantitativos para algoritmo de contagem de pessoas. Os resultados apresentados expõem a combinação dos melhores parâmetros para os métodos testados de modo empírico (demonstrado no apêndice B), por isso, apenas os melhores resultados são apresentados conforme apresentado na Tabela 2.

6.4.1 Experimento 1

A cena deste experimento é composto pela Visão_001, Visão_002 e Visão_003. Apresentou rastros bem definidos, sendo repetido o mesmo caminho algumas vezes. Ocorreu apenas um erro de classificação.

6.4.2 Experimento 2

A cena deste experimento é composto pela Visão_004, Visão_005 e Visão_006. Com a região de interesse de visão parametrizadas diferentemente e rastros bem definido, apenas ocorreram erros de classificação.

6.4.3 Experimento 3

A cena deste experimento é composto pela Visão_007, Visão_008 e Visão_009. Ocorreram vários momentos em que pessoa percorria a sub-região de fronteira, permanecendo parte do corpo dentro da sub-região central, contudo sem de fato realizar transições, como também, ocorreu um classificações.

6.4.4 Experimento 4

A cena deste experimento é composto pela Visão_010, Visão_011 e Visão_012. Os indivíduos presentes na cena perambulavam bastante pelas sub-regiões, e em alguns momentos de muito movimento, o rastreador se perdia com facilidade devido as oclusões. Além disso, o rastreador deixava de detectar diversas cabeças, pois o limiar de correspondência entre as cabeças e ombro implementada falha quando ocorria erro de segmentação.

6.4.5 Experimento 5

Na visão_013, ocorreram rastros bem definidos, com poucas oclusões de aproximação nas sub-regiões, o rastreador se comportou razoavelmente bem, contudo o limiar de correspondência entre as cabeças e ombro foi estendido para um limiar maior para que o rastreador conseguisse associar os rastros, essa maior variação do limiar também explica a baixa taxa de acerto do classificador, uma vez que, o conjunto de treinamento não ajustou-se ao novo perfil da cena (devido ao afastamento ou a aproximação da pessoa em relação a câmera essa região pode variar).

6.5 CONCLUSÕES DO CAPÍTULO

Em geral, esses vídeos apresentam duas característica comum: a de um intenso fluxo de pessoas e a de baixa densidade de contagem. Isso se deve pelo fato da regra de abstração de estado, onde as transições só podem ocorrer desde que a pessoa passe da sub-região central para sub-região diferente da qual ela entrou.

Ao analisar os resultados dos experimentos, percebe-se a importância da noção de perspectiva de visão da cena, uma vez que, a posição, o afastamento e a aproximação da câmera alteram as projeções das pessoas, o que influência sistematicamente na precisão e assertividade do métodos abordados.

O rastreador e classificador com bastante frequência sofreram algumas flutações estatísticas. No rastreador foram as oclusões, que interferiram significativamente na qualidade do método de rastreamento, como também, algumas distorções de segmentação, o que fazia em alguns momentos ele perder o alvo. Enquanto para o classificador, o limite proporcional fixado em relação região superior extraída da *bounding box* dos objetos variavam com aproximação e afastamento das pessoas em relação a câmera, o que causava ruídos nas métricas extraídas.

Acerca do método proposto, em geral, alcançou os objetivos propostos, demonstrando nos cenários testados, ser capaz de contabilizar pessoas em fluxo multidirecional, mesmo com o risco de erros. Considere a taxa de acerto do classificador como equivalente taxa de acerto para contagem, têm-se que as taxas de acertos individuais obtidas nos experimentos apresentados ficaram entre 60% e 100%. Tendo em vista a contagem por direção feita sob diferentes visões de uma mesma cena, concluí-se que, o experimento alcançou um bom resultado com taxa de acerto média de 78,16%.

Tabela 2 – Resultados dos vídeos - classificação Real (R) x Sistema (S)

Direção		Direita		Esquerda		Cima		Baixo		Acerto de	Erro de
Vídeo	**Classe**	**R**	**S**	**R**	**S**	**R**	**S**	**R**	**S**	**Classificação**	**Classificação**
Visão_001	**0**	-	-	-	-	-	-	-	-	9	1
	1	3	3	4	4	3	2	-	-		
	2	-	-	-	-	-	1	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_002	**0**	-	-	-	-	-	-	-	-	7	0
	1	4	4	-	-	3	3	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_003	**0**	-	-	-	-	-	-	-	-	3	2
	1	3	2	2	1	-	-	-	-		
	2	-	1	-	1	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_004	**0**	-	1	-	-	-	-	-	-	5	1
	1	4	3	1	1	1	1	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_005	**0**	-	1	-	-	-	-	-	-	6	1
	1	4	3	3	3	-	-	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_006	**0**	-	-	-	-	-	1	-	-	9	1
	1	4	4	2	2	4	3	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_007	**0**	-		-	-	-	1	-	-	3	1
	1	2	2	-	-	2	1	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_008	**0**	-	-	-	-	-	-	-	-	6	0
	1	4	4	1	1	1	1	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_009	**0**	-	-	-	1	-	-	-	-	8	1
	1	4	4	4	3	1	1	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_010	**0**	-	2	-	7	-	-	-	-	18	9
	1	11	9	14	7	-	-	-	-		
	2	2	2	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_011	**0**	-	2	-	4	-	-	-	-	14	6
	1	7	6	9	5	-	-	2	2		
	2	2	1	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_012	**0**	-	2	-	3	-	-	-	-	20	5
	1	11	9	11	8	-	-	-	-		
	2	-	-	3	3	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Visão_013	**0**	-	-	-	3	-	-	-	-	3	3
	1	-	-	5	2	1	1	-	-		
	2	-	-	-	-	-	-	-	-		
	3	-	-	-	-	-	-	-	-		
Total	**0**	-	8	-	18	-	2	1	-	111	31
	1	61	53	56	37	16	12	2	2		
	2	4	4	3	4	-	0		-		
	3	-	-	-	-	-	-	-	-		

Fonte: Próprio autor.

7 CONCLUSÕES E TRABALHOS FUTUROS

Neste estudo, tem como contribuição no sentido da implementação de um sistema a custo reduzido para contagem multidirecional de pessoas usando câmara de vídeo de vigilância fixa.

Em relação ao processamento das imagens, definiu-se um sistema de software que processa um conjunto de técnicas de processamento de imagens. Na versão atual é requerida uma configuração inicial do sistema e uma base de conhecimento vinculada para treinamento do classificador.

Ao analisar os resultados dos experimentos, o método expressou bons resultados, contudo, percebe-se alguns pontos críticos que impedem um melhor desempenho, tais como a segmentação, o limiar para detecção da região superior e o ajuste de perspectiva de visão da cena. Todos esses pontos críticos são influenciados pela posição em relação ao afastamento e a aproximação da câmera, uma vez que essas mudanças alteram as projeções das pessoas, e consequentemente a precisão e assertividade do métodos envolvidos.

O presente trabalho corroboram as conclusões apresentadas por Valle (2007) e Silva (2008) em seus trabalhos, onde resultados da taxa de acerto está entre 80% e 100%, dependendo das condições dos testes. Considerando-se que em seus trabalhos foram empregados campos de visões que possuem condições favoráveis para se realizar a contagem automática de pessoas.

Concluí-se que ainda há muito a ser melhorado para que a contagem obtenha resultados ótimos. É evidente que o sistema atual funciona dentro das limitações impostas, entretanto ele serve como base para a construção de um sistema mais completo, robusto e eficiente. Perante disto, pode-se depreender como consequência de que há necessidade de alindamento dos métodos para que estes possam ser empregados para processamento em ambientes e situações factuais.

Finalmente, existem algumas sugestões para possíveis futuros estudos que devem ser modificados para melhorar os resultados dos métodos propostos tais como: investigar ou propor novos métodos de limites de fronteiras; criar um método auto-ajustável para captura da região superior dos objetos; realizar um estudo semelhante sobre a contagem de pessoas com base em outros modelos; ampliar à amostra para incorporar a contagem de cenas com mais pessoas; verificar se o processo e os motivos desse novos modelos a serem testados ocorrem de modo semelhante em ambientes reais; comparar o modelo de contagem de pessoas em ambiente de cenários dinâmicos com ambientes fechados.

Entre as principais contribuições deste trabalho destaca-se: o desenvolvimento de uma solução técnica que permite a utilização de câmaras de vigilância para a contagem automática multidirecional de pessoas com base em uma infraestrutura de baixo custo, analise da aplicabilidade de combinações de métodos de visão computacional para contar o número de pessoas por direção, calibração de parâmetros dos métodos abordados que melhore a confiabilidade da contagem, as avaliações do impacto dos métodos propostos sob a disposição da câmera em diferentes visões de uma mesma cena e a aplicação de técnica de máquina de estado para definir as regras de contagens.

Um fato importante a ser observado neste sistema é de que o modelo de classificação dos objetos é baseado em amostras de treinamento, desta forma ele torna-se perene e nem sempre assertivo em todas as situações, por isso o modelo precisa ser acompanhado. Nesse sentindo, é sugerido um indicador de aderência Teste de Kolmogorov-Smirnov (KS) (usado para determinar se duas distribuições de probabilidade subjacentes diferem uma da outra) (HAZEWINKEL, 2001) para a avaliação do novos objetos adicionados a amostra de treinamento inicial, com isso, pode-se decidir-se o re-treinamento do modelo será necessário, ou seja, se o modelo está se ajustando ou não aos novos perfis de objetos em cena. Se não, há grande chances de não estar classificando corretamente, ao contrário espera-se que a classificação seja bastante assertiva.

Este índice compara a distribuição da classificação da base de treinamento inicial com o perfil do novos objetos, medindo se existe diferença (possível perfil de mudança da amostra de treinamento) entre a atual amostra teste e a do desenvolvimento do modelo. Neste caso, a semelhança entre as novas amostras é o que se espera. Quanto menor a diferença entre a nova amostra e a de treinamento do modelo, melhor. Ou seja: quanto menor o KS (mais próximo de zero), melhor. É importante avaliar onde está a diferença na classes de pessoas, para avaliar se a amostra nova tem mais uma determinada classe.

É bom salientar que entrada de amostra novas diferente da observada no modelo não necessariamente significa má qualidade do mesmo. É fundamental observar em quais variáveis da instância compostas pelo descritores estão as diferenças representativa, identificando possíveis mudanças de perfil do objetos. O KS representa a máxima diferença entre as distribuição da base de treinamento atual e da base de treinamento inicial. Quanto menor for este valor, menor será a diferença de perfil entre as duas amostras comparadas. Assim, recomenda-se a investigação das causas dessa possível alteração do perfil dos objetos.

GLOSSÁRIO

B

Background: é a parte de uma cena que está por trás de uma figura principal ou objeto, sendo o oposto definido como *foreground*.

Bolha: é um grande objeto binário de um quadro binário, isto é, um componente conectado em uma imagem onde todos os *pixels* da componente têm o mesmo nível de intensidade, também tratado como um possível objeto-alvo em uma imagem. .

F

Foreground: a porção mais próxima de uma cena para o espectador (oposta à do *background*).

G

Ground Truth:é uma medida para avaliar a acurácia do sistema. Nessa abordagem, é realizada uma contagem manual do fluxo de pessoas coma finalidade de avaliar a taxa de acerto, além de outras medidas para mensuração do desempenho do sistema.

M

Máscara binária: também chamada binível, é uma imagem digital na qual há apenas dois valores possíveis para cada *pixel*.

T

Track: é uma estrutura que armazena características para cada objeto em primeiro plano a fim de estabelecer uma correspondência entre bolhas em primeiro plano em dois quadros consecutivos.

REFERÊNCIAS

AHA, D. W.; KIBLER, D.; ALBERT, M. K. Instance-based learning algorithms. **Machine learning**, Boston,MA, v. 6, n. 1, p. 37–66, Jan. 1991.

AL-ZAYDI, Z. Q.; NDZI, D. L.; YANG, Y.; KAMARUDIN, M. L. An adaptive people counting system with dynamic features selection and occlusion handling. **Journal of Visual Communication and Image Representation**, Elsevier, v. 39, p. 218–225, Aug. 2016.

AMER, A. Voting-based simultaneous tracking of multiple video objects. **Circuits and Systems for Video Technology**, IEEE, v. 15, n. 11, p. 1448–1462, 2005.

AZEVEDO, E.; CONCI, A.; LETA, F. **Computação Gráfica:** Teoria e Prática. 2 ed. [S.l.]: Campus, 2008.

BHUVANESHWAR, V.; MIRCHANDANI, P. B. Real-time detection of crossing pedestrians for traffic-adaptive signal control. In: **Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on**. [S.l.], 2004. p. 309–313.

BJöRGVINSSON. **Peocounter-People Counting Software**. Tese (Doutorado em Ciência da Computação) — Chalmers University of Technology, Gothenburg, 2006.

BLACK, P. E. Finite state machine. **Dictionary of Algorithms and Data Structures**, US National Institute of Standards and Technology. Feb, v. 6, 2008.

BOURIDANE, A. **Imaging for Forensics and Security**: From Theory to Practice [S.l.]: Springer, 2009. cap 1. p. 1–10.

BOUWMANS, T.; BAF, F. E.; VACHON, B. Background modeling using mixture of gaussians for foreground detection-a survey. **Recent Patents on Computer Science**, Bentham Science Publishers, v. 1, n. 3, p. 219–237, Nov. 2008.

CANNY, J. A computational approach to edge detection. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, n. 6, p. 679–698, Jan. 1986.

CORREIA, M. V. **Análise de Movimento em Seqüências de Imagens**. Dissertação (Mestrado em eletrotécnica e de computadores) —Faculdade de Engenharia da Universidade do Porto. Portugal, 1995.

DEDEOGLU, Y. **Moving object detection, tracking and classification for smart video surveillance**. Dissertação (Mestrado em ciência) — Bilkent university, 2004.

ELGAMMAL, A.; HARWOOD, D.; DAVIS, L. Non-parametric model for background subtraction. In: SPRINGER. **European conference on computer vision**. [S.l.], 2000. p. 751–767.

FACELI, K. **Inteligência artificial:** uma abordagem de aprendizado de máquina. [S.l.]: Grupo Gen-LTC, 2011.

FRIEDMAN, N.; RUSSELL, S. Image segmentation in video sequences: A probabilistic approach. In: MORGAN KAUFMANN PUBLISHERS INC. **Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence**. [S.l.], 1997. p. 175–181.

FUKUNAGA, K.; NARENDRA, P. M. A branch and bound algorithm for computing k-nearest neighbors. **Computers, IEEE Transactions on**, IEEE, v. 100, n. 7, p. 750–753, Aug. 1975.

GONZALEZ, R. Re: **Digital Image Processing.** [S.l.]**:** Reading, Addison Wesley, 2002.

GONZALEZ, R. C.; WOODS, R. E. **Processamento de imagens digitais.** 3 ed. [S.l.]: Pearson, 2007.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing Using MATLAB**. 2 ed. [S.l.]: Gatesmark Publishing, 2009.

GRISETTI, C. S. G.; ARRAS, W. B. K. **Robotics 2 association.** Disponível em: <http: //ais.informatik.uni-freiburg.de/teaching/ws09/robotics2/pdfs/rob2-11-dataassociation.pdf>. Acesso em: 23 mar.2016.

HARITAOGLU, I.; HARWOOD, D.; DAVIS, L. S. W4s: A real-time system for detecting and tracking people in 2 1/2d. In: SPRINGER. **European Conference on computer vision**. [S.l.], 1998. p. 877–892.

HARITAOGLU, I.; HARWOOD, D.; DAVIS, L. S. W 4: Real-time surveillance of people and their activities. **Pattern Analysis and Machine Intelligence**, IEEE, v. 22, n. 8, p. 809–830, 2000.

HAZEWINKEL, M. Kolmogorov-smirnov test. **Encyclopedia of Mathematics,** Springer, p. 978–1000, 2001.

HEIKKILÄ, J.; SILVÉN, O. A real-time system for monitoring of cyclists and pedestrians. **Image and Vision Computing**, Elsevier, v. 22, n. 7, p. 563–570, 2004.

HOU, Y.-L.; PANG, G. K. People counting and human detection in a challenging situation. In: **Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on**, v. 41, n. 1, p. 24–33, 2011.

JEON, Y.; RYBSKI, P. **Analysis of a spatio-temporal clustering algorithm for counting people in a meeting. Citeseer.** Pittsburgh **,**Jan. 2006.

KALMAN, R. E. A new approach to linear filtering and prediction problems. **Journal of Fluids Engineering**, American Society of Mechanical Engineers, v. 82, n. 1, p. 35–45, 1960.

KETTNAKER, V.; ZABIH, R. Counting people from multiple cameras. In: **Multimedia Computing and Systems, 1999. IEEE International Conference on**. [S.l.], 1999. v. 2, p. 267–271.

KIM, J.-W.; CHOI, K.-S.; CHOI, B.-D.; KO, S.-J. Real-time vision-based people counting system for the security door. In: **International Technical Conference on Circuits/Systems Computers and Communications**. [S.l.: s.n.], 2002. p. 1416–1419.

KONG, D.; GRAY, D.; TAO, H. Counting pedestrians in crowds using viewpoint invariant training. In: **British Machine Vision Conference.** [S.l.], 2005.

LEE, D.-S. Online adaptive gaussian mixture learning for video applications. In: **International Workshop on Statistical Methods in Video Processing**. [S.l.], 2004. p. 105–116.

LEFLOCH, D. Real-time people counting system using a single video camera. In: **Electronic Imaging** [**S.l.**], 2007. p. 1109-1112

LIPTON, A. J.; FUJIYOSHI, H.; PATIL, R. S. Moving target classification and tracking from real-time video. In:**pplications of Computer Vision, 1998. WACV'98. Proceedings., Fourth IEEE Workshop on**. [S.l.], 1998. p. 8–14.

LIU, H.; JIANG, S.; HUANG, Q.; XU, C.; GAO, W. Region-based visual attention analysis with its application in image browsing on small displays. In: **Proceedings of the 15th ACM international conference on Multimedia**. [S.l.], 2007. p. 305–308.

MADDALENA, L.; PETROSINO, A.; RUSSO, F. People counting by learning their appearance in a multi-view camera environment. **Pattern Recognition Letters**, Elsevier, v. 36, p. 125–134, 2014.

MAYBECK, P. S.; D'AZZO, J. J.; HOUPIS, C. H.; KEPLER, H. B.; REHG, V.; JR, C. T. T.; HITZELBERGER, M. R. W.; SHANKLAND, D. G.; HARRINGTON, M. T. C.; FISCHER, W. A. et al. Stochastic models, estimating, and control. **New Directions in Effective Management**, Academic Press, v. 8, p. 12–14, 1982.

MILLER, M. L.; STONE, H. S.; COX, I. J. Optimizing murty's ranked assignment method. **IEEE Transactions on Aerospace and Electronic Systems**, IEEE, v. 33, n. 3, p. 851–862, 1997.

MITCHELL, T. M. **Machine Learning**. 1. ed. New York, USA: McGraw-Hill, Inc., 1997. ISBN 0070428077, 9780070428072.

MORELLAS, V.; PAVLIDIS, I.; TSIAMYRTZIS, P. Deter: detection of events for threat evaluation and recognition. **Machine Vision and Applications**, Springer, v. 15, n. 1, p. 29–45, 2003.

MUKHERJEE, S.; DAS, K. An adaptive gmm approach to background subtraction for application in real time surveillance. **arXiv preprint arXiv:1307.5800**, 2013.

MUKHERJEE, S.; DAS, K. Omega model for human detection and counting for application in smart surveillance system. **arXiv preprint arXiv:1303.0633**, Mar. 2013.

PEDRINI, H.; SCHWARTZ, W. R. **Análise de imagens digitais:** princípios, algoritmos e aplicações. [S.l.]: Thomson Learning, 2008.

SCHOFIELD, A.; STONHAM, T.; MEHTA, P. Automated people counting to aid lift control. **Automation in Construction**, Elsevier, v. 6, n. 5, p. 437–445, 1997.

SERRA, J. The"centre de morphologie mathematique": an overview. **Mathematical Morphology and Its Applications to Image Processing**, Kluwer Academic Publishers, Norwell, MA, p. 369–374, 1994.

SIDLA, O.; LYPETSKYY, Y.; BRANDLE, N.; SEER, S. Pedestrian detection and tracking for counting applications in crowded situations. In: **2006 IEEE International Conference on Video and Signal Based Surveillance**. [S.l.], 2006. p. 70–70.

SILVA, L. S. D. **Sistema Computacional para Contagem Automática de Pessoas Baseado em Análise de Seqüências de Imagens**. Dissertação (Mestrado em Computação) — Centro Federal de Educação Tecnológica de Minas Gerais, 2008.

SIVABALAKRISHNAN, M.; SHANTHI, K. Person counting system using efv segmentation and fuzzy logic. **Procedia Computer Science**, Elsevier, v. 50, p. 572–578, 2015.

SNIDARO, L.; MICHELONI, C.; CHIAVEDALE, C. Video security for ambient intelligence. In: **IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans**, IEEE, v. 35, n. 1, p. 133–144, 2005.

STAUFFER, C.; GRIMSON, W. E. L. Adaptive background mixture models for real-time tracking. In: **Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.** [S.l.], 1999. v. 2.

SU, M.-Y. Using clustering to improve the knn-based classifiers for online anomaly network traffic identification. **Journal of Network and Computer Applications**, Elsevier, v. 34, n. 2, p. 722–730, 2011.

TEIXEIRA, T.; SAVVIDES, A. Lightweight people counting and localizing in indoor spaces using camera sensor nodes. In: **2007 First ACM/IEEE International Conference on Distributed Smart Cameras**. [S.l.], 2007. p. 36–43.

TKALCIC, M.; TASIC, J. F. et al. Colour spaces: perceptual, historical and applicational background. In: **The IEEE Region 8 EUROCON 2003. Computer as a Tool.** [S.l.: s.n.], 2003. p.22-24.

VALLE, J. **Contagem automática de pessoas em cenas de vídeo usando visão computacional**. Dissertação (Mestrado em informática) — Pontifícia Universidade Católica do Paraná, 2007.

WANG, R.; BUNYAK, F.; SEETHARAMAN, G.; PALANIAPPAN, K. Static and moving object detection using flux tensor with split gaussian models. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops**. [S.l.: s.n.], 2014. p. 414–418.

WELCH, G. F. Kalman filter. In: **Computer vision**. [S.l.]: Springer, 2014. p. 435–437.

YOSHINAGA, S.; SHIMADA, A.; TANIGUCHI, R.-i. Real-time people counting using blob descriptor. **Procedia-Social and Behavioral Sciences**, Elsevier, v. 2, n. 1, p. 143–152, 2010.

ZACCARIOTTO, V. L. Detecção de pessoas utilizando vídeo em ambiente controlado. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Curso de Ciência da Computação, Universidade São Francisco, Itatiba, 2010.

ZANG, Q.; KLETTE, R. Parameter analysis for mixture of gaussians. In: **Communication and Information Technology Research Technical Report 188. [S.l.],**2006. p. 188-193.

APÊNDICES

APÊNDICE A – Algoritmo de Contagem de Pessoas Por Direção

Algoritmo 3: Contagem do fluxo direcional

```
    Entrada: Frames do vídeo
[1] Contagem do fluxo por direção
[2] início
[3]     existir frames frame = lerVideo();
[4]     numobj = rastreador.numObj;
[5]     objetos = rastreador.VetorObjetos();
[6]     para i=1 até numObj faça
[7]         obj = objetos[i];
[8]         regiaoAtual = obterRegiao(obj.centroide);
[9]         obj.regiao ≠ regiaoAtual adicionaTransicao(obj, regiaoAtual);
[10]        adicionaRegiao(obj, regiaoAtual);
[11]        obj.transicoes = 2 e obj.regioes[1] ≠ obj.regioes[3] contar = ativarClassificador(obj);
[12]        obj.regioes[3] caso 1 hacer
[13]            */ BAIXO */
[14]            contagem[1] += contar;
[15]        fin
[16]        caso 2 hacer
[17]            */ CIMA */
[18]            contagem[2] += contar;
[19]        fin
[20]        caso 3 hacer
[21]            */ DIREITA */
[22]            contagem[3] += contar;
[23]        fin
[24]        caso 4 hacer
[25]            */ ESQUERDA */
[26]            contagem[4] += contar;
[27]        fin
[28]        resetarFluxo(obj);
[29]    fim
[30] fim
[31] contagem
```

APÊNDICE B – Descritivas dos Parâmetros Utilizados

Na Tabela 3 é exposto uma descrição dos parâmetros do modelo nas respectivas variáveis de estrutura efetiva na elaboração do algorítimo.

Tabela 3 – Parâmetros do modelo GMM

Parâmetros	Descrição
$\sigma_{0,i}$	Desvio padrão inicial de cada Gaussiana.
σ_{th}	Número de vezes que o valor do *pixel* pode exceder o desvio padrão de uma das B Gaussianas para ser abalizado como *pixel* de *foreground*.
$\omega_{0,i}$	Peso inicial de cada Gaussiana.
ω	Coeficiente invariante empregado na atualização do modelo dos pesos das Gaussianas.
T	Percentual ínfimo de *background* no corrente quadro, sendo este empregado no estágio de classificação.
ng	número de combinação de Gaussianas empregado na mistura de Gaussianas.
α	taxa de aprendizagem.

Fonte: Próprio autor.

Parametrização da Região de Interesse

Para parametrização da região de interesse foi implementado uma simples função denomidanda "segROI" que divide a regiões de fronteiras parametrizando a área de interesse a ser monitorada a partir do centro do quadro, em que, dado a resolução do vídeo $w \times h$ define se os parâmetros empíricos (α, β) para ajustar a proporção de área capturada no quadro, como também a proporções das regiões fronteiriças através dos parâmetros (pw, ph), odendo ser visualizada no conjunto de equações em B.1. Onde, os parâmetros α, β, pw e ph estão entre o intervalo [0,1].

$$\begin{aligned} y &= (h - h \times \alpha)/2 \\ x &= (w - w \times \beta)/2 \\ ROI &= [x, y, w - 2 \times x, h - 2 \times y] \\ segROI(pw, ph, ROI) \end{aligned} \tag{B.1}$$

Tabela 4 – Parâmetros para definição da Região de Interesse - I

Proporção	Visão_001	Visão_002	Visão_003
da Área Capturada (α, β)	(0,82;0,85)	(0,82;0,85)	(0,82;0,99)
da Regiões de Fronteiras (pw,ph)	(0,25;0,25)	(0,25;0,25)	(0,25;0,16)

Fonte: Próprio autor.

Tabela 5 – Parâmetros para definição da Região de Interesse - II

Proporção	Visão_004	Visão_005	Visão_006
da Área Capturada (α, β)	(0,82;0,85)	(0,80;0,99)	(0,60;0,99)
da Regiões de Fronteiras (pw,ph)	(0,25;0,20)	(0,30;0,15)	(0,15;0,10)

Fonte: Próprio autor.

Tabela 6 – Parâmetros para definição da Região de Interesse - III

Proporção	Visão_007	Visão_008	Visão_009
da Área Capturada (α, β)	(0,70;0,80)	(0,70;0,95)	(0,40;0,99)
da Regiões de Fronteiras (pw,ph)	(0,30;0,20)	(0,15;0,15)	(0,15;0,00)

Fonte: Próprio autor.

Tabela 7 – Parâmetros para definição da Região de Interesse - IV

Proporção	Visão_010	Visão_011	Visão_012
da Área Capturada (α, β)	(0,60;0,99)	(0,90;0,99)	(0,90;0,99)
da Regiões de Fronteiras (pw,ph)	(0,30;0,00)	(0,20;0,00)	(0,30;0,00)

Fonte: Próprio autor.

Tabela 8 – Parâmetros para definição da Região de Interesse - V

Proporção	Visão_013
da Área Capturada (α, β)	(0,7;0,7)
da Regiões de Fronteiras (pw,ph)	(0,50;0,89)

Fonte: Próprio autor.

Parametrização do *k*-NN

Foi realizado uma análise para diferentes valores *k* do modelo *k*-NN, essa análise tem como objetivo melhorar a assertividade da classificação das bolhas, e consequentemente pré-avaliar o desempenho do método de contagem.

A partir da associação das detecções a cada *frame*, as quais foram manualmente classificadas, para que o método de vizinhos mais próximos, por meio distância Euclidiana entre os vetores de característica, estabeleça a classificação de novos objetos baseado na suas similaridades métricas. Seguidamente foram realizados alguns testes com o vídeo, obtendo-se o resultado (como mostra na Tabela 9), que visa a analisar o seu desempenho por meio das estimativas médias da validação cruzada de 10-*fold* executada cem vezes realizada para um conjunto de amostra com 903 objetos.

Tabela 9 – Validação Cruzada

Avaliação	**Vizinhos**				
10-Folder	**1**	**3**	**5**	**7**	**9**
Taxa de acerto	0,7387	0,6025	0,5773	0,5665	0,5659
Sensibilidade	0,9051	0,9387	0,8832	0,6734	0,6050
Especificidade	0,7920	0,7019	0,6285	0,6134	0,6001

Fonte: Próprio autor.

Para os vídeos referentes à detecção de pessoas, a avaliação dos resultados obtidos por meio da validação cruzada de 10-fold indicaram que a análise realizada para k = 1 (uma vizinhança) alcançam o melhor desempenho para todos vídeos analisados. Uma vez que, a taxa de acerto apresentou um alto grau de concordância entre o resultado de uma medição e o valor verdadeiro mensurado. Também verifica-se que a sensibilidade e a especificidade nos vídeos testes apontam que existe uma grande quantidade de falsos negativos (um resultado positivo é mais provável de ser verdadeiro positivo do que falso-positivo) e quando o objeto for julgado negativo é bastante confiável a sua classificação (um resultado negativo é mais provável de ser um verdadeiro do que falso negativo do que um falso-negativo).

Parametrização do Modelo GMM

Parâmetros ajustado empiricamente, os demais parâmetros foram utilizado o *default* do MATLAB.

Tabela 10 – Parâmetros do modelo GMM

Parâmetros	α	**ng**	**nº de frames treinamento**	T
Valor	0,0007	5	70	3,5

Fonte: Próprio autor.

Parametrização das Operações Morfológicas

- Abertura com elemento retangular 3×3;
- Fechamento com elemento retangular 15×15;
- Preenchimento de fundo a partir da borda da imagem.

Parametrização do Filtro de Kalman

Foi utilizando método padrão disponível no MATLAB com ajuste apenas na matriz de custo de atribuição do algoritmo de Murty para valor igual a 200.

Parametrização de Controle das Tracks

- Invisível (*) por: 50
- Limiar de idade (**) menor que: 16
- Limiar de visibilidade (***) menor que: 0,7

* Margem mínima de *frames* para consecutivas ausência de atribuição.

** É um contador linear atribuída a mesma *track* sequencialmente.

*** Razão entre o total *track* detectada e atribuída pela idade.

Parametrização da região superior dos objetos por meio da Bounding box pré-ajustada

Para caracterização dos limiares da região superior dos objetos detectados que correspondente a cabeça e ombro foram utilizados para os vídeos EDS e MDS o padrão bw = 0,99 e bh = 0,20, enquanto para o vídeo Atrium, o padrão foi bw = 0,5 e 0,89. Esses ajustes são parametrizações da "bounding box"seguindo a expressão $[x, y, largura \times bw, altura \times bh]$.

ANEXO

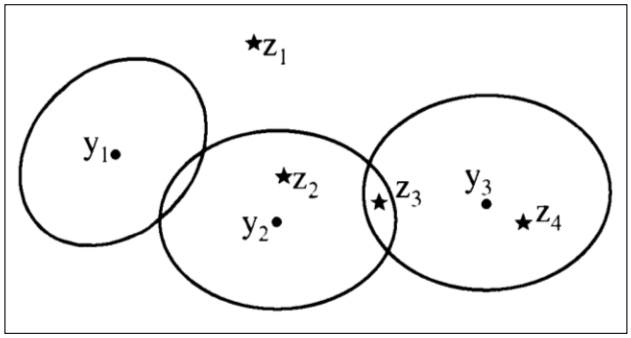
ANEXO A – Método de Murty para Rastreamento de Múltiplos Alvos

O Algoritmo Munkres (também conhecido como algoritmo Húngaro) é um algoritmo eficiente para resolver o problema da atribuição em tempo polinomial. O algoritmo tem muitas aplicações na otimização combinatória, como no problema do caixeiro viajante, contudo no contexto desse trabalho é apresentado como uma solução para o problema de associação de dados em tempo real.

ALGORITMO MURTY

Nessa seção será apresentado uma breve descrição de método. O método de Murty encontra as k melhores solução para o problema de alocação linar, P_0. O problema é expressado como um grafo bipartido, representado como uma lista de triple $\langle y, z, l \rangle$. Em uma aplicação de rastreamento multi-alvo, cada y representa um alvo hipotético, cada z representa um medida, e l é a uma função de log-verossimilhança a qual z é uma medida de y.

Figura 24 – Exemplo de rastreamento na qual quatro novas medições, Z_1, Z_2, Z_3, Z_4 são validados para três tracks, Y_1, Y_2, Y_3



Fonte**:** Elaborado por Miller et al. (1997).

Nas ilustrações nas Figuras 24 e 25 nas quais existem as *tracks* y_1, y_2 e y_3 que estão sendo seguidas e quatro medidas, z_1, z_2, z_3 e z_4 são observadas. As elipses em torno de cada *tracks* representar convencional regiões de validação. Assim, nenhuma medição pode ser associado com *track* y_1; duas medições, z_2 e z_3 poderia ser associado com *track* y_2 e semelhante medições z_3 e z_4 caírem dentro do volume de validação da *track* y_3. A medição z_1 não valida a qualquer existente *track* e pode, portanto, ser um falso alarme ou início de uma nova *track*. Estas associações possíveis pode ser representado como um grafo bipartido cujos nós fonte representam medições e cujos sorvedouros representam *tracks*. A adjacência matriz de um

Figura 25 – Representação da configuração do grafo bipartido

	$f1$	$f2$	$f3$	$f4$	$y1$	$y2$	$y3$	$n1$	$n2$	$n3$	$n4$
$z1$	2.3	.	.	.	.	.	.	2.5	.	.	.
$z2$	.	2.3	.	.	.	0.9	.	.	2.5	.	.
$z3$	.	.	2.3	.	.	1.6	1.7	.	.	2.5	.
$z4$	.	.	.	2.3	.	.	2.1	.	.	.	2.5
$u1$	0.0	0.0	0.0	0.0	3.0	.	.	0.0	0.0	0.0	0.0
$u2$	0.0	0.0	0.0	0.0	.	3.0	.	0.0	0.0	0.0	0.0
$u3$	0.0	0.0	0.0	0.0	.	.	3.0	0.0	0.0	0.0	0.0
$u4$	0.0	0.0	0.0	0.0	.	.	.	0.0	0.0	0.0	0.0
$u5$	0.0	0.0	0.0	0.0	.	.	.	0.0	0.0	0.0	0.0
$u6$	0.0	0.0	0.0	0.0	.	.	.	0.0	0.0	0.0	0.0
$u7$	0.0	0.0	0.0	0.0	.	.	.	0.0	0.0	.0.0	0.0

Elaborado por Miller et al. (1997).

gráfico que representa os resultados na Figura 24 aparece na Figura 25. Os valores na matriz de adjacência representar o log de verossimilhança negativa de cada tarefa e, para este exemplo, são um tanto arbitrárias.

As quatro primeiras linhas desta matriz representam as quatro medições, z_1 através z_4, e os próximos sete filas umas fileiras de enchimento. Há uma linha "*dummy*"para cada *track* atual (y_1 através y_3), mais uma linha para cada potencialmente nova *track*, cada um dos quais poderia crescer a partir de uma corrente de medição z_1, z_2, z_3 ou z_4. Assim, há sete linhas de imitação rotulados u_1 através u_7 nesta matriz.

A matriz tem uma coluna para cada possível alarme falso, representado pelas colunas de f_1 a f_4, mais uma coluna para cada faixa atual e potencialmente nova *track*. As *tracks* atuais têm rótulos y_1 a y_4 e os potencialmente novas faixas têm rótulos n_1 através n_4. Há uma coluna de alarme falso e potencialmente uma nova faixa para cada linha de medição.

Uma entrada diferente de zero na matriz de adjacência é o logaritmo negativo da probabilidade de a associação correspondente. Assim, a entrada na célula 2.3 (Z_1, f_1) corresponde à probabilidade de e $e^{-2.3}$ de uma medição z_1 ser um falso alarme. As linhas fictícios são destinados a modelar situações em que nenhuma medida corresponde a um alvo atual. Por exemplo, a entrada na célula 3.0 (u_1, y_1) representa a probabilidade de $e^{-3.0}$ que corresponde nenhuma medição para controlar y_1. Neste exemplo, as outras entradas para a coluna y_1 estão vazios, de modo que o resultado mais provável para y_1 é que ele não irá ser associada com qualquer um de entre z_1 através z_4. Ao atribuir-lo para u_1, podemos evitar que seja atribuído a uma medição. Uma ou mais das novas faixas n_1 a n_4, não pode ser associado com uma medição. Se assim for, um nó de nova-*track* deve ser atribuído a um nó de origem "*dummy*". Esse é o papel reservado para nós de origem u_4 através u_7 neste exemplo.

Algoritmo 4: Encontrar a melhor solução, S_0, para P_0. Deixe C_0 = o custo de S_0, e deixe U_0 e V_0 serem as variáveis duais associadas com S_0.

Entrada: inicializar uma fila de prioridade de problema/parcial-solução para com estrutura $< P_0, S_0, U_0, V_0, C_0 >$. O topo de estrutura desta fila será sempre a estrutura com um limite menor custo (C_0 neste caso)

[1] melhores soluções **início**
[2] Limpar a lista de soluções para ser devolvido
[3] **para** *i = 1 até k, ou até a fila de prioridade está vazio* **faça**
[4] o topo da estrutura na fila no conter uma solução completa Tome o topo da estrutura $< P, s, u, v, c >$, fora da fila.
[5] Encontre a melhor solução completa, S, para P, através da aplicação de um aumento de JV para s, com u e v como variáveis duais. Deixe U e V serem o resultado da variável dual, e C ser o custo de S.
[6] S existecolocar P sobre a fila de prioridade. Tome a estrutura do topo, $< P, S, U, V, C >$, fora da fila de prioridade.
[7] Adicione S para lista de soluções a ser retornada.
[8] **para** $m = 1$ *até n, onde n é o número de z's em P, deixe* $minY[m]$*=-1 (*$minY[m]$ *eventualmente será índice do y nó que está conectado a* z_m *por um arco com menor slack. Aqui ele está sendo definido como um valor inválido)* **faça**
[9] Deixe $deadY = -1$ (deadY indicará qual y tem sido removido a partir de P)
[10] S não está vazio Deixe $highSlack = -INFINITY$
[11] **para** *m = 1 até n* **faça**
[12] $minY[m] = deadY$ (sempre verdade a primeira vez) $minSlack[m] = INFINITY$ **para** *todas triples,* $< y_j, z_m, l >$ *que estão em P mas não está em S* **faça**
[13] Deixe $slack = 1 - U[m] - V[j]$.
[14] $slack < minSlack[m]$ $minSlack[m] = slack$
[15] $minY[m] = j$.
[16] **fim**
[17] $minSlack[m] > highSlack$ Deixe $highSlack = miniSlack[m]$.
[18] Deixe $highZ = m$.
[19] **fim**
[20] Deixe $m = highZ$.
[21] Encontre a triple , $< y_j, z_m, l >$ que contém z_m em S.
[22] Deixe $P^{'} = P, s^{'} = S, u^{'} = U e v^{'} = V$
[23] Remova $< y_j, z_m, l >$ de $P^{'}$
[24] Remova $< y_j, z_m, l >$ de $s^{'}$
[25] deadY = j
[26] Deixe $c^{'} = C + highSlack$
[27] Substituía $< P^{'}, s^{'}, u^{'}, v^{'} >$ sobre a fila de prioridade
[28] A partir de P, remova todas triples $< y_h, z_m, l >$, $h \neq j$, e todas triples $< y_j, z_h, l >$, $h \neq i$
[29] Remove $< y_j, z_m, l >$ a partir de S.
[30] **fim**
[31] **fim**
[32] **fim**
[33] melhor solução