Faculdade de Engenharia da Universidade do Porto

Contribuições para deteção de pessoas em espaços complexos

Nuno Miguel Ferreira Lopes Conde Pires

VERSÃO FINAL

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Eletrotécnica e de Computadores
Major Telecomunicações

Orientador: Pedro Carvalho (Doutor)
Co-orientador: Luís Corte-Real (Professor Doutor)

Fevereiro 2014

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

MIEEC - MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES | **2013/2014**

A Dissertação intitulada

"Contribuições para Detecção de Pessoas em Espaços Complexos"

foi aprovada em provas realizadas em 07-02-2014

o júri

Presidente Professora Doutora Maria Teresa Magalhães da Silva Pinto de Andrade
Professora Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

Professor Doutor José Manuel Castro Torres
Professor Associado do da Faculdade de Ciências e Tecnologia da Universidade Fernando Pessoa

Professor Doutor Luís António Pereira de Meneses Corte-Real
Professor Associado do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

Doutor Pedro Carvalho
Investigador INESC - TEC

O autor declara que a presente dissertação (ou relatório de projeto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extratos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são corretamente citados.

Autor - Nuno Miguel Ferreira Lopes Conde Pires

Faculdade de Engenharia da Universidade do Porto

Resumo

A deteção de pessoas é um assunto em constante estudo e evolução, na área de processamento de imagem e visão, com aplicação em diversas áreas, como a robótica, vigilância e segurança, ou mesmo na saúde. Mais especificamente no âmbito da videovigilância, a deteção e seguimento de pessoas é essencial, onde o processamento de imagem tem um papel fundamental, permitindo tornar os sistemas de vigilância cada vez mais autónomos e robustos.

Motivados pelo facto de existir atualmente na literatura, um conjunto de direções distintas no âmbito da deteção de indivíduos, com métricas de avaliação e conjuntos de dados variados, é complicado efetuar uma comparação direta entre elas. Torna-se assim difícil responder a questões, tais como, "Qual é a melhor abordagem?"," Quais os principais fatores que perturbam a performance dos algoritmos?", ou até "Qual o ponto de evolução atual desta área de investigação?".

Este trabalho foca-se na tentativa de responder a este tipo de questões, trazendo 4 contribuições principais. A primeira consiste num estudo e avaliação de várias técnicas e metodologias regularmente utilizadas para este propósito. Foram então implementados e testados 4 detetores presentes na literatura, com características distintas, com o objetivo de identificar e compreender quais os seus pontos fortes e limitações, de acordo com as experiências realizadas.

A segunda contribuição está relacionada com a implementação de melhorias numa das métricas de avaliação utilizadas, com o objetivo de as tornar mais fiáveis e discriminativas.

A terceira contribuição consiste no estudo da possibilidade de adaptar um detetor responsável por detetar sinais de trânsito, na deteção de pessoas, motivado pela semelhança existente entre a fisionomia do ser humano e dos sinais de trânsito, que revela resultados promissores.

Por fim, a última contribuição consiste na implementação de um filtro de pós-processamento, aplicado a imagens térmicas, com o objetivo de remover falsas deteções provocadas por reflexões presentes na imagem. Este filtro é aplicável a qualquer algoritmo, revelando provocar um impacto muito positivo nas suas performances, reduzindo o número de

falsos positivos sem perder a capacidade de deteção e simultaneamente reduzindo a área das regiões detetadas.

Abstract

People detection is a subject in constant study and progress in the area of image processing and vision, with application in various areas such as robotics, surveillance and security, or even health. More specifically in the context of video surveillance, detection and tracking of people is essential, where image processing plays a key role , giving the possibility to create even more autonomous and robust surveillance systems.

Motivated by the fact that currently exists in the literature, a set of different directions within the detection of individuals with varied evaluation metrics and data sets, is complicated to make a direct comparison between them. It thus becomes difficult to answer questions such as, "What is the best approach?", "What are the main factors affecting the performance of the algorithms?", or even "What is the current point of development of this area of research?".

This work focuses on the attempt to answer such questions, bringing four main contributions. The first one is related with an extensive study and evaluation of four state of the art detectors, with distinct specifications, in order to identify the strengths and limitations of each one, regarding several experiments.

The second contribution is related with the implementation of improvements in a performance evaluation method, in order to make the generated metrics, more reliable and discriminative. The third contribution embraces the study of the possibility of applying a traffic sign detector in pedestrian detection, motivated from the shape similarity between traffic signs and human beings, which reveals promissory results.

Finally the last contribution is related with the implementation of a post processing filter applied in thermal images, responsible for removing false detections caused by reflections that exist in the images. This filter is applicable to any algorithm, revealing a very positive impact on their performances, reducing the number of false alarms, without losing detection ability, while reducing the area of the detected regions.

Agradecimentos

Serve este espaço para demonstrar a minha gratidão a todos aqueles que contribuíram diretamente ou indiretamente no desenvolvimento desta dissertação.

Pretendo agradecer à instituição INESC por me receber e me dar a oportunidade de trabalhar neste projeto.

Agradeço aos orientadores Professor Doutor Luís Corte-Real e Doutor Pedro Carvalho pela orientação prestada, pelo tempo e paciência despendidos em reuniões e esclarecimento de dúvidas, como também pelo fornecimento de todo o material necessário para o desenvolvimento desta dissertação.

Pretendo também agradecer ao Eng. Iago Landesa por toda a sua disponibilidade e pela ajuda fornecida no desenvolvimento deste trabalho.

Agradeço também ao Eng. Lucian Ciobanu pelo tempo despendido e ajuda proporcionada.

Agradeço aos meus colegas de trabalho pela constante boa disposição que mantiveram no local de trabalho, proporcionando uma maior motivação e pela ajuda também prestada em algumas questões.

Agradeço aos meus pais por estarem sempre comigo, me incentivarem ao estudo, e a quem devo a oportunidade de estudar.

Por fim, agradeço à minha namorada pelo apoio e constante força e motivação que sempre me transmitiu.

Índice

Lista de figuras

Lista de tabelas

Abreviaturas e Símbolos

- Lista de abreviaturas (ordenadas por ordem alfabética)

3D-MC	*3D - Motion Context*
ACF	*Active Contour Feature*
Adaboost	*Adaptive Boosting*
CHOG	*Color Histogram of Oriented Gradients*
CMOS	*Complementary Metal-Oxide-Semiconductor*
CSS	*Color Second order Statistics*
CVML	*Computer Vision Markup Language*
DT	*Distance Transform*
EM	*Expectation-Maximum*
FDF	*Four Directional Feature*
FIR	*Far Infra-Red*
GGP	*Granularity Gradient Partitions*
GMM	*Gaussian Mixture Model*
HIK	*Histogram Intersection Kernel*
HIKSVM	*Histogram Intersection Kernel Support Vector Machine*
HMC	*Harmonic Motion Context*
HOF	*Histogram of Oriented Flows*
HOG	*Histogram of Oriented Gradients*
HOG+M	*Histogram of Oriented Gradients + Magnitude*
HOT	*Histogram of Oriented Templates*
IKSVM	*Intersection Kernel Support Vector Machine*
ISM	*Implicit Shape Model*
KLT	*Kanade Lucas Tomasi*
LBP	*Local Binary Pattern*
LML	*L1-norm Minimization Learning*
LogitBoost	*Logistic Boosting*

MAP	*Maximum a Posteriori*
MCMC	*Markov Chain Monte Carlo*
MCMIBoost	*Multi-Class Multi-Instance Boosting*
MIL	*Multiple Instance Learning*
MIR	*Middle Infra-Red*
mi-SVM	*Multiple Instance Support Vector Machine*
MPLBoost	*Multi-Pose Learning Boosting*
MSO	*Multi Scale Orientation features*
NIR	*Near Infra-Red*
NIT	*New Infra-Red Technology*
NRLBP	*Non-Redundant Local Binary Pattern*
OpenCV	*Open Source Computer Vision*
OSME1C	*OneStopMoveEnter1Cor*
OSME1F	*OneStopMoveEnter1Front*
OSOW2C	*OneShopOneWait2Cor*
OSOW2F	*OneShopOneWait2Front*
R-HOG	*Rectangular-Histogram of Oriented Gradients*
ROI	*Region of Interest*
RBF	*Radial Basis Function*
SIFT	*Scale-Invariant Feature Transform*
SMC	*Sequential Monte Carlo*
SVM	*Support Vector Machine*
TES3C	*TwoEnterShop3Cor*
TES3F	*TwoEnterShop3Front*
V-HOG	*Variable size- Histogram of Oriented Gradients*

- Lista de Símbolos

μm	Micrómetro
mm	Milímetro
K	*Kelvin*
°C	*grau* Celsius
F	*grau Fahrenheit*
W	Potência Radiante $\left[{}^{W}/_{m^2}\right]$
σ	*Constante de Stefan - Boltzmann* $\left[5{,}7 \times 10^{-8}\ {}^{W}/_{K^4 m^2}\right]$
ε	Emissividade
T	Temperatura absoluta *[K]*

Capítulo 1

Introdução

1.1 - Introdução

O tema retratado nesta dissertação é um tópico em constante estudo e evolução no decorrer desta última década, tendo vindo a merecer grande atenção por parte do setor académico e industrial, onde o processamento de imagem e visão tem um papel fundamental, permitindo tornar os sistemas de vigilância cada vez mais autónomos e robustos.

Atualmente, a deteção de pessoas tem um enorme interesse para um vasto leque de aplicações, tais como videovigilância, análise de imagem, na implementação de veículos inteligentes, que permitem a automatização de ações que melhoram a segurança da condução, na animação de personagens utilizada em filmes e jogos, ou até na análise biomecânica de ações relacionadas com desporto ou medicina. Tais aplicações encontram-se em constante melhoramento, a fim de se obter sistemas cada vez mais completos e fiáveis.

Com o recente avanço na tecnologia CMOS, a integração da captura de imagens e vídeos em vários sistemas, torna-se cada vez mais real, devido à possibilidade de se reduzir o tamanho e custo de vários aparelhos e simultaneamente aumentar o seu poder de processamento, tais como câmaras de vídeo, máquinas fotográficas ou até telemóveis.

Esta dissertação tem um foco particular em aplicações direcionadas à supervisão e vigilância autónoma de espaços, sejam eles interiores ou exteriores, tais como aeroportos ou zonas comerciais. Tornar estes sistemas autónomos traz grandes vantagens, como a redução de esforço humano, o que em termos financeiros produz uma diminuição nos custos, ou a prevenção contra erros e ambiguidades provocadas pelo erro humano nesta tarefa.

A tarefa de deteção consiste em determinar se um objeto existente numa certa região da imagem se trata de um humano ou não. Para tal, existem na literatura variadas abordagens com características e estruturas distintas. Uns baseiam-se em métodos de janela deslizante,

analisando a imagem por inteiro e outros em métodos de subtração de fundo, com o objetivo de encontrar as regiões da imagem que merecem maior atenção (regiões de interesse (ROI)). Também diferem no tipo de *features* (características) que são extraídas e como são extraídas da imagem, podendo ser divididas em três subconjuntos: forma, aparência e movimento, embora existam abordagens que integram vários tipos de *features*, colmatando assim as limitações que cada uma delas apresenta. Outro aspeto onde as várias abordagens diferem, está relacionado com o tipo de classificadores que utilizam para analisar a informação.

As falhas provenientes deste processo devem-se a um vasto leque de fatores relacionados com a fisionomia do corpo humano e o ambiente em que está inserido. O ser humano assume grandes variações de postura no seu quotidiano o que dificulta a tarefa de deteção, visto ser necessário cobrir a maior quantidade possível de posturas que o ser humano pode representar na imagem. A diferença de vestuário introduz outro desafio, que dificulta a própria deteção, pois recursos fortes como a cor (usado em deteção de faces) demonstram algumas limitações, devido à grande variedade de cores e texturas que daí resultam. A variação da orientação e do tamanho dos objetos, devido ao posicionamento da câmara provoca grandes desafios aos modelos a serem gerados. Variações na luminosidade da cena, encontradas normalmente em cenários exteriores, podem ajudar ou prejudicar no sucesso da tarefa, dependendo da direção e qualidade da iluminação. Uma das consequências da direção da luminosidade consiste na existência de sombras, que muitas vezes levam o sistema a falhar, originando falsas deteções.

A complexidade do padrão de fundo ou a elevada densidade da cena são outros aspetos que limitam a performance dos algoritmos de deteção. Intensas densidades populacionais ou a presença de certos objetos nas imagens podem provocar ocultações, sendo este um dos maiores desafios na tarefa de deteção. Várias situações de ocultação podem ocorrer, como por exemplo, no caso de existir a presença de objetos na imagem, como mobília, árvores, carros, edifícios, ou mesmo outros indivíduos, que cobrem a pessoa ou partes do corpo da mesma, ou até mesmo a própria postura do indivíduo pode provocar a ocultação de partes do seu corpo.

A eficiência de um algoritmo de deteção depende da sua imunidade em relação a todos os aspetos acima referidos. Para lidar com todos estes problemas, é essencial identificar diversas características humanas usando várias pistas de informação e a utilização de classificadores adequados a cada situação. No entanto, quanto mais intensa e aprofundada for a análise e processamento da informação, mais os sistemas se tornam computacionalmente complexos, provocando assim, o aumento do tempo de processamento, o que exige maior capacidade computacional por parte dos equipamentos usados e dificulta a integração destes algoritmos em sistemas aplicados em tempo real.

1.2 - Objetivos

Com o desenvolvimento desta dissertação pretende-se, para além de outros aspetos, fazer um estudo e levantamento dos vários tópicos relacionados com a deteção automática de indivíduos, com o objetivo de atingir maiores níveis de conhecimento relativamente a esta área de processamento. Pretende-se testar vários algoritmos com características distintas, aplicados em cenários diversificados, com o propósito de analisar a sua performance a vários níveis, bem como identificar as limitações que possuem, perante os vários desafios a que são sujeitos, como será explicado mais à frente.

De forma a produzir resultados mais concretos e fiáveis para a análise dos algoritmos, propôs-se também a implementação de algumas alterações a métricas de avaliação do estado da arte.

Motivados por um trabalho realizado na área de prevenção rodoviária, pretende-se estudar a possibilidade da adaptação de um detetor de sinais de trânsito na deteção de pessoas, motivados pela semelhança existente entre a fisionomia dos sinais e do ser humano.

Ao verificar a pouca utilização de imagens térmicas na deteção automática de pedestres, propôs-se o desenvolvimento de um filtro de pós processamento adaptativo, aplicável a qualquer sistema, que tem como objetivo remover reflexões totais e parciais detetadas incorretamente pelos algoritmos.

1.3 - Estrutura

O documento está dividido em 6 capítulos, sendo o primeiro a introdução, onde é contextualizado o tema em estudo e onde são apresentados os vários desafios a enfrentar nesta dissertação.

O capítulo 2 consiste num levantamento de várias técnicas e metodologias existentes na literatura relativas ao tema em questão, acompanhado de análises e comparações entre as várias abordagens, e discutindo vários aspetos relacionados com a deteção de objetos, mais especificamente, assuntos relativos às várias *features* e classificadores normalmente utlizados para este efeito e o impacto que provocam na performance dos algoritmos.

O capítulo 3 descreve os testes realizados e o resultado das avaliações efetuados a quatro algoritmos distintos existentes na literatura, com o intuito de analisar as suas performances em várias situações, identificando os pontos fortes e limitações de cada uma das abordagens. Também se encontram descritas neste ponto, as alterações propostas às métricas de avaliação aplicadas aos resultados dos algoritmos utilizados.

No capítulo 4 são abordados assuntos relacionados com a possível adaptação de um detetor utilizado na deteção de sinais de trânsito, na deteção de pedestres, visto que existem semelhanças entre a fisionomia dos sinais de trânsito e o ser humano.

O capítulo 5 é dedicado ao estudo e análise da informação proveniente de imagens termográficas, onde é proposto um filtro de pós-processamento para melhorar o resultado final dos algoritmos neste tipo de ambientes. Neste ponto, a arquitetura, funcionamento e o impacto deste filtro encontram-se descritos e analisados de uma forma pormenorizada. O filtro é aplicado aos algoritmos utilizados no capítulo 3, com o objetivo de demonstrar que este é possível integrar em qualquer algoritmo de deteção, melhorando assim a sua performance.

Por fim, o capítulo 6 apresenta as conclusões finais que se podem retirar desta dissertação, bem como possíveis trabalhos futuros que possibilitem o melhoramento dos trabalhos realizados neste projeto.

Capítulo 2

Estado da Arte

Atualmente, a deteção automática de pessoas é um assunto em constante estudo e desenvolvimento, com o fim de se obter sistemas cada vez mais completos e fiáveis. Várias abordagens são utilizadas, umas com melhores resultados nuns aspetos e piores noutros, visto que existem diversos fatores que influenciam a performance destes sistemas, logo um bom levantamento das mais variadas técnicas é essencial para o sucesso desta dissertação. Tais fatores incluem, variações de postura, luminosidade, vestuário, resoluções de imagem ou até ocultações de indivíduos, que dificultam o sucesso do algoritmo.

À partida, na tarefa de deteção, não se conhece a localização, tamanho, orientação ou postura de um indivíduo na imagem. Assim, várias abordagens aplicam uma janela deslizante na imagem de entrada, em diversificadas localizações e escalas, janela esta que irá ser classificada como contendo um humano ou não. Outras abordagens consistem na interpretação de um indivíduo como sendo um conjunto de partes humanas; as janelas são classificadas como contendo uma parte humana ou não.

É frequentemente utilizado um método para diminuir a área de pesquisa na imagem, denominado subtração de fundo. Esta abordagem baseia-se na ideia de que se existir um modelo de fundo, pode-se subtrair uma imagem de referência do respetivo modelo, sendo as regiões que restam, as que devem ser processadas. Estas áreas resultantes são normalmente denominadas de regiões de primeiro plano ou de interesse. Como esta metodologia se baseia num modelo de fundo, este tipo de técnicas aplicam-se maioritariamente em situações em que são utilizadas câmaras estacionárias. Para além disso, sistemas que utilizem subtração de fundo necessitam de um modelo de fundo para cada situação em particular, obrigando assim à criação de um modelo de fundo associado a cada cenário.

De forma a lidar com a ocultação de indivíduos, é comum explorar-se a deteção de partes do corpo em vez de corpo inteiro, visto que se um indivíduo se encontra parcialmente oculto, apresenta apenas uma porção do corpo disponível para análise. No entanto, o sucesso deste

tipo de abordagens está tipicamente limitado pela qualidade dos métodos de subtração de fundo e/ou segmentação, que utilizam.

Pode-se concluir, que o sucesso de um algoritmo de deteção depende da qualidade da classificação que é efetuada às várias regiões da imagem, classificação essa que depende da sua arquitetura e do poder discriminativo das *features* que utiliza. Existem na literatura vários artigos que se focam na comparação e análise de vários tipos de *features* e classificadores e as várias combinações que podem existir entre eles [Dollar2012] [Gavrila2009] [Munder2006].

Os estudos e desenvolvimentos relativos à deteção automática de pedestres estão maioritariamente focados na aplicação em imagens RGB, embora também existam abordagens que se focam noutros tipos de informação, como estéreo ou infravermelhos. Motivado pela pouca insistência na utilização e estudo da deteção de pessoas em imagens térmicas e pela oportunidade de estudar a possibilidade de adaptar técnicas e metodologias utilizadas frequentemente noutras fontes de informação, foi dedicada uma parte desta dissertação ao estudo do comportamento de várias abordagens que utilizam este tipo de imagens e à possibilidade de adaptar técnicas utilizadas em imagens RGB, de forma a encarar os desafios que este tipo de imagens proporcionam, como é o caso das reflexões, que existem de forma acentuada neste tipo de imagens e que provocam um impacto bastante negativo na performance dos algoritmos, principalmente originando falsas deteções.

O resto do capítulo está dividido em dois grandes conjuntos: Deteção em Imagens RGB e Estratégias e desafios em imagens térmicas. No primeiro foram discutidas várias *features* regularmente utilizadas para caracterizar a informação, e aspetos relacionados com a classificação dos objetos. No segundo foram analisadas algumas abordagens que utilizam informação proveniente de imagens térmicas e desafios que estas proporcionam.

2.1 - Deteção em imagens RGB

Existe na literatura, um leque diversificado de *features* utilizado na deteção de objetos, particularmente de humanos, podendo ser utilizadas individualmente, ou combinando informação proveniente de várias. São criadas através de informação de baixo nível, como por exemplo, a partir de informação de textura, cantos, cor ou até mesmo de movimento. Foi decidido dividir este conjunto em três subconjuntos: forma, aparência e movimento.

2.1.1 - Deteção baseada em *features* de forma

Para definir a forma de um objeto, são regularmente utilizadas *features* baseadas em contornos, devido ao facto de a forma da silhueta humana poder ser representada de uma forma fiável através de um conjunto de contornos (figura 2.1). No entanto, abordagens baseadas em informação deste tipo têm as suas limitações, visto que em situações de pouco contraste na imagem ou de muita complexidade, torna-se difícil extrair os contornos de uma forma precisa.

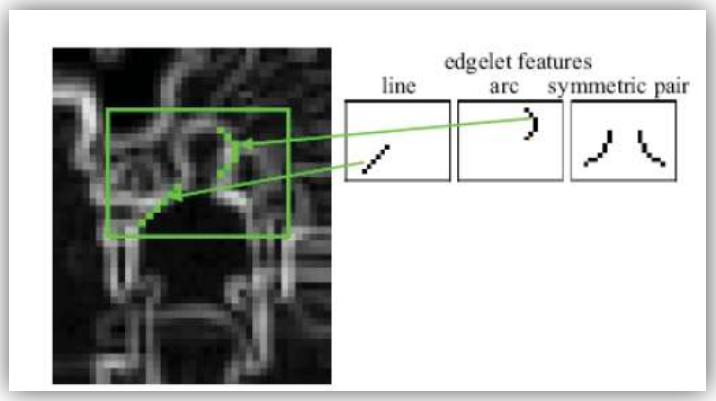


Figura 2.1 – Exemplo ilustrativo de uma *feature* de forma, as *Edgelet features* **[Nevatia2005]**.

Para descrever essas formas, podem ser utilizados contornos binários para modelar a forma humana completa [Nguyen2009], ou para modelar partes do corpo individualmente [Andriluka2009] [Lin2007] [Lin2010] [2005Nevatia]. A tarefa de deteção é então concluída através de *template matching*. Este tipo de abordagens não é muito fiável em situações em que o plano de fundo é complexo.

Zhao propôs em [Zhao2003] um algoritmo de deteção aplicável a câmaras estacionárias que utiliza subtração de fundo numa estrutura Bayesiana, cuja segmentação origina regiões de primeiro plano onde são utilizados modelos de forma humana para interpretar a informação aí presente. A solução passa pelo número de objetos humanos identificados e respetivos parâmetros associados maximizarem a estimação MAP (*Maximum a Posteriori*), probabilidade esta que caracteriza a eficiência da reconstrução da imagem de primeiro plano a partir das probabilidades da imagem. É utilizado MCMC (*Markov Chain Monte Carlo*) com saltos e difusões dinâmicas com o intuito de alcançar a melhor solução atravessando todo o espaço de solução. O MCMC é uma ferramenta utilizada para provar uma distribuição probabilística. Este método tem como objetivos ter uma boa performance na deteção em cenários densos onde ocorrem várias situações de ocultação.

Em [2005Nevatia] é adotada a deteção de partes do corpo, com o objetivo de lidar com situações de ocultação e tornar o algoritmo mais tolerante a variações do posicionamento das câmaras. Este método é dividido em dois níveis, um responsável pela deteção das partes e outro pela combinação das várias partes detetadas. Relativamente ao primeiro, são utilizados detetores treinados para um conjunto de *features* de forma, as *edgelet* (figura 2.1). Estas *features* consistem numa cadeia de contornos. Os detetores são treinados através de um melhoramento da abordagem de *boosting* implementada em [Viola2001]. No segundo nível é definida uma função de probabilidades conjuntas da imagem para tentar resolver o problema das ocultações. A forma humana é aproximada através de uma elipse 2D e a visibilidade das partes do corpo é calculada de acordo com a profundidade relativa dos objetos. Por fim, o problema da deteção de múltiplos indivíduos é formulado através da estimação MAP.

Um dos descritores baseados no contexto de forma com resultados mais significativos, e consequentemente dos mais utilizados na literatura, é o descritor HOG [Dalal2005]. Consiste na construção de histogramas de gradientes orientados, que contam a ocorrência das orientações dos gradientes em regiões específicas da imagem, como se pode verificar na figura 2.2. O HOG difere do SIFT, no aspeto em que o HOG é calculado numa densa rede de células uniformemente espaçadas e utiliza sobreposição de contrastes locais normalizados com o intuito de melhorar a precisão da descrição.

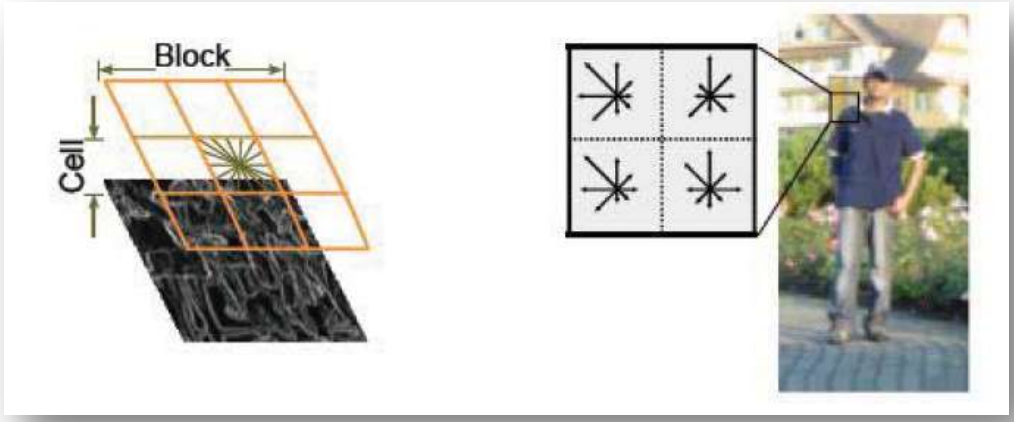


Figura 0- (a) Descritor HOG, (b) um exemplo da sua utilização para codificar uma região **[Nguyen2012]**.

Vários algoritmos baseados no HOG ou variantes deste foram concebidos [Begard2008] [Ye2010] [Maji2008] [Tang2010] [Xu2010] [hang2010] [Choo2010] [Bansal2010].

Em [Begard2008] foi aplicada uma variante do HOG, o HOG+M, que complementa o HOG especificamente no facto de que cada histograma é composto por nove *orientation bins* de magnitudes normalizadas, mais um, referente à magnitude do gradiente. São analisados e propostos dois métodos baseados no algoritmo *AdaBoost*, com vista a deteção em tempo real. São utilizados descritores locais *gradient-based* (HOG+M), que são combinados numa estrutura

em cascata. Um otimiza o uso de cada descritor selecionado, para minimizar as operações efetuadas na imagem de forma a acelerar o processo de deteção. O outro associa a cada a cada descritor um *weak learner* mais poderoso construído a partir do HOG+M completo, com o objetivo de assegurar que os classificadores produzidos são os melhores possíveis. Este método de seleção avalia localmente os histogramas e fundamenta o processo de seleção na comparação dos classificadores resultantes. Visto que os dois métodos propostos são complementares, ao serem integrados produzem melhores resultados que o algoritmo *Adaboost* clássico (figura 2.3).

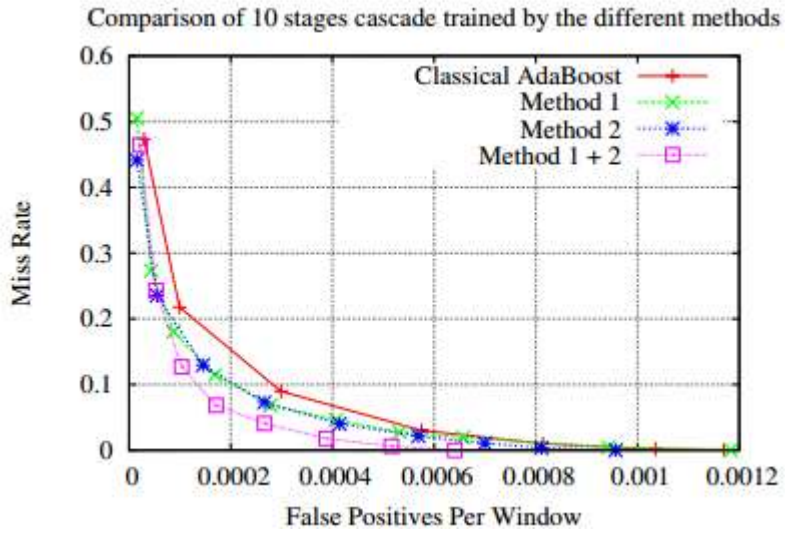


Figura 2.2 – Comparação dos resultados dos detetores treinados através do *Adaboost* clássico, métodо1, método 2 e combinação dos dois **[Begard2008]**.

AdaBoost é um algoritmo meta-heuristico que auxilia na aprendizagem de outros algoritmos, potenciando assim a performance dos mesmos. Trata-se de um algoritmo de *machine learning*, adaptativo na perspetiva de que classificações subsequentes são atualizadas a favor das instâncias mal classificadas pelos classificadores anteriores. O *Adaboost* pode ser visto como a minimização de uma função convexa de perda, nomeadamente uma perda exponencial, segundo um conjunto de funções convexas. O *Adaboost* clássico chama um classificador "fraco" repetidamente, e em cada iteração atualiza uma distribuição de pesos que caracterizam a importância da informação. Em cada iteração os pesos dos exemplos classificados incorretamente aumentam, e os pesos dos exemplos classificados corretamente diminuem, para que o classificador se foque nos exemplos que despoletaram classificações corretas até ao momento.

Em [Maji2008] foi proposto uma nova *feature* baseada no HOG, embora com uma estrutura mais simples e com uma dimensão mais baixa. Esta *feature* é mais simples pelo facto de não incluir sobreposição de células, peso gaussiano das respostas em cada célula,

nem normalizações da imagem, o que proporciona *features* com dimensões muito mais pequenas. Esta *feature* combinada com IKSVM (*Intersection Kernel Support Vector Machine*), produz melhores resultados que o método proposto por *Dalal*, que utiliza o HOG com SVM linear (*Support Vector Machine*).

O tipo de classificadores mais utilizado na literatura é o SVM. Caracterizam-se como sendo modelos de aprendizagem supervisionados, com algoritmos de aprendizagem associados, que analisam informação, reconhecendo padrões usados para classificação e análises de regressão. O SVM básico utiliza um conjunto de informação de entrada e prevê, para cada *input*, que classe (em duas classes existentes) forma o *output*, tornando o SVM um classificador linear não probabilístico. Tendo então um conjunto de exemplos de treino, um algoritmo de aprendizagem constrói um modelo que define novos exemplos como pertencendo a uma das categorias. Um modelo SVM é então uma representação dos exemplos através de pontos no espaço, mapeados de forma a que os exemplos de cada categoria estejam separados por uma fronteira de decisão, o mais acentuada possível. Assim, os novos exemplos são mapeados no mesmo espaço e a categoria a que pertencem é prevista de acordo com o lado da fronteira em que estão inseridos. Os SVM lineares são regularmente utilizados devido à sua robusta performance e elevada velocidade de processamento. É também possível utilizar SVM para efetuar classificações não lineares, utilizando o "*kernell trick*", que mapeia implicitamente os *inputs* em espaços de *features* de grandes dimensões. Esta variante traz algumas vantagens, no entanto, o tempo necessário para classificar um exemplo, é linear em relação ao número de *support vectors*, daí ter sido criada uma exceção denominada HIK (*Histogram* Intersection Kernel), que é computada em tempo logarítmico.

Choo et al em [Choo2010] propuseram a utilização do HOG num algoritmo de deteção que utiliza segmentação da imagem e máscaras virtuais. A segmentação permite ao algoritmo focar-se apenas nas regiões de interesse, por outro lado o uso de máscaras virtuais, reduz significativamente o ruído e algum processamento desnecessário, o que resulta numa diminuição do tempo de processamento. O conceito de máscara virtual permite uma diminuição do número de regiões de interesse a ser codificadas, pois este consiste em impor um certo valor de *threshold* a cada região, dependendo da sua localização na imagem, rejeitando aquelas que não atingem um certo tamanho, diminuindo assim a carga computacional e tempo de processamento. A segmentação é efetuada através de um método de subtração de fundo, com o objetivo de se encontrar e definir as regiões de interesse onde se encontram realmente indivíduos em movimento, evitando assim processamento desnecessário. Nesta implementação são propostas alguma simplificações ao HOG, de forma a se alcançar maior velocidade de processamento, embora se perca consequentemente um pouco de precisão na performance do algoritmo.

Em [Zhang2010] [Xu2010], é proposto o método LML (*L1-norm Minimization Learning*), que integra a seleção de *features* com a construção de um classificador linear para deteção

de pedestres. O método de aprendizagem *L1-norm Minimisation* permite obter a dispersão do espaço das *features* de uma forma efetiva. Para além disso, para minimizar o número de *weak classifiers* na cascata de classificadores, foi aplicada programação inteira como caso especial do *L1-norm minimisation* no espaço inteiro. São exploradas duas variantes do HOG, respetivamente o R-HOG e o V-HOG, que diferem do HOG relativamente à geometria dos blocos. Ficou provado que a utilização deste método de aprendizagem permite atingir melhor performance tanto em velocidade de processamento, como em precisão nos cenários testados.

Na figura 2.4 encontra-se ilustrado o gráfico que demonstra o impacto deste tipo de aprendizagem na performance dos algoritmos.

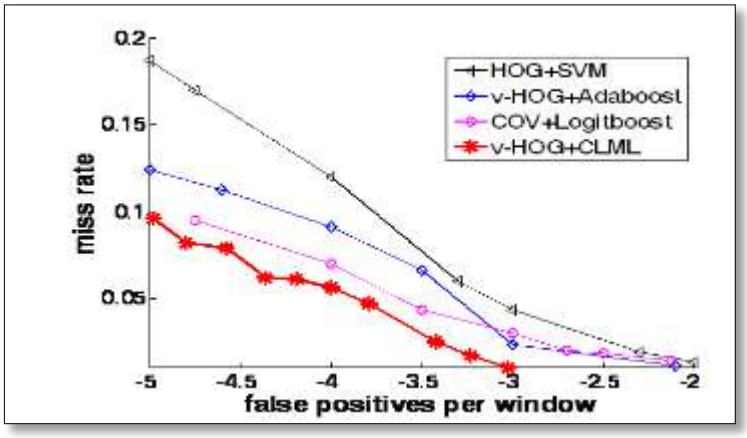


Figura 2.3 – Comparação dos resultados do método desenvolvido, com outros existentes na literatura **[Xu2010]**.

Ye et al desenvolveram em [Ye2010] um tipo de *features* baseado no HOG, as MSO (*Multi Scale Orientation features*). MSO distingue-se do HOG pelo facto de as suas *features* poderem ser finas e/ou grosseiras. Enquanto as MSO são *multi-escala* tanto em relação às unidades de *features* como à orientação dos pixéis, o HOG exige um tamanho de blocos fixo. Por fim, a localização espacial das unidades MSO são determinadas pela seleção de *features*, enquanto no HOG estas são fixas. Na figura 2.5, encontra-se ilustrada a *feature* MSO e na figura 2.6 encontram-se descritos os resultados relativos à performance deste método em comparação com outros descritos na literatura.

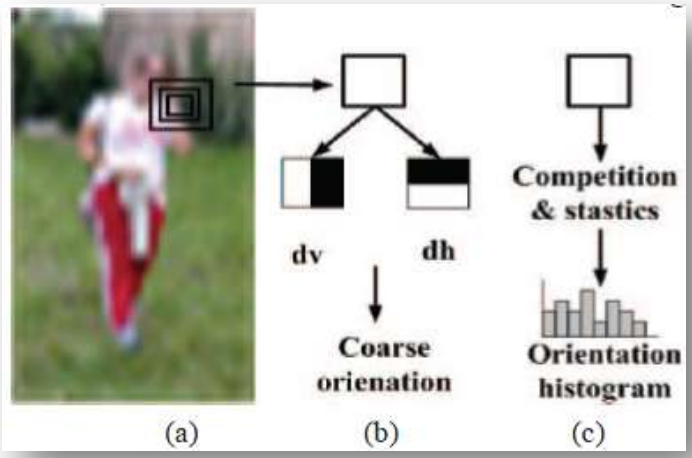


Figura 2.4 - MSO *features*, (a) Unidades *Multi-scale* de uma região, (b) *feature* grosseira, (c) *feature* fina **[Ye2010]**.

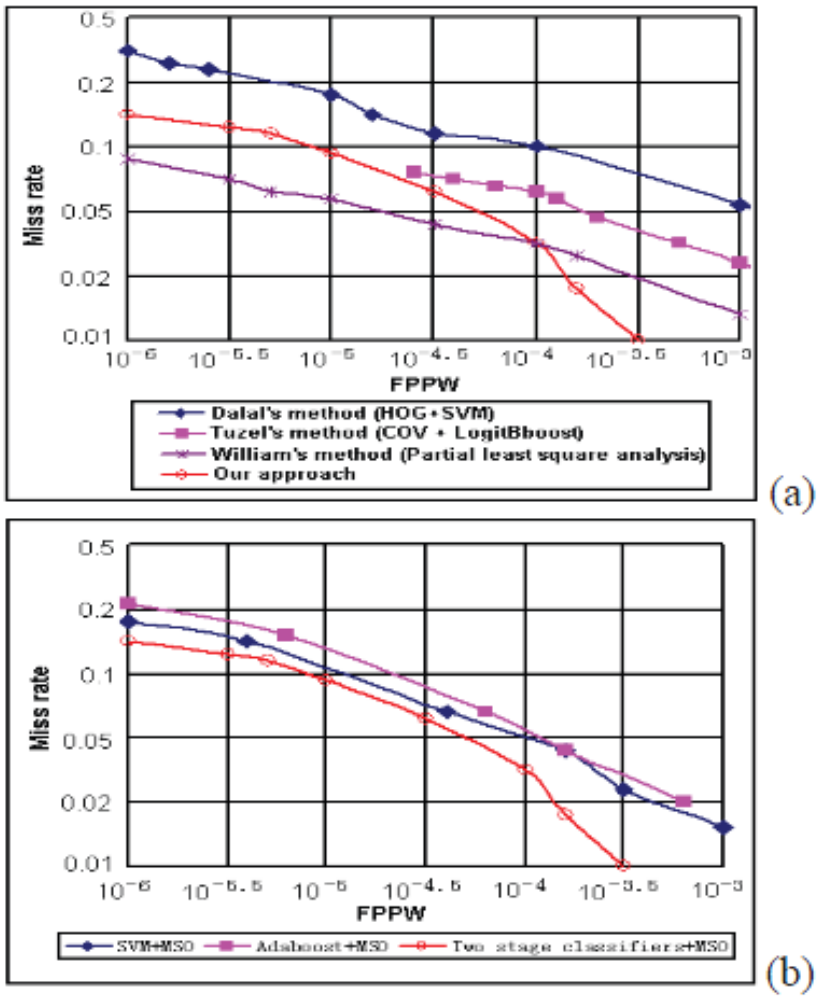


Figura 2.5- (a) Comparação da performance da abordagem desenvolvida com outros métodos, (b) Comparação da performance do classificador utilizado com outros existentes **[Ye2010]**.

Em [Tang2010] é implementada uma abordagem diferente baseada no descritor HOG, denominada HOT (*Histogram of Oriented Templates*). O método HOT consiste na definição de vários *templates* (figura 2.7) para cada píxel da imagem, *templates* estes constituídos pelo píxel em questão e dois dos seus pixéis vizinhos, dependendo do *template* em questão (figura 2.8). Se a intensidade e gradientes destes três pixéis satisfizerem uma função pré definida, o píxel central é associado ao *template* em questão. Os resultados provenientes dos histogramas dos pixéis em conjunto com os vários templates construídos são então utilizados num conjunto de funções, que combinadas constituem os recursos para a deteção. Em comparação com o método HOG, o HOT utiliza, para além da informação dos gradientes e das intensidades, a relação entre três pixéis em vez de um só. Os resultados relativos à performance deste método encontram-se ilustrados na figura 2.9, que permitem concluir que, de facto, o método HOT é mais eficiente que o HOG, e para além de uma melhor performance, é um recurso que trabalha em paralelo o que possibilita o funcionamento desta aplicação em tempo real.

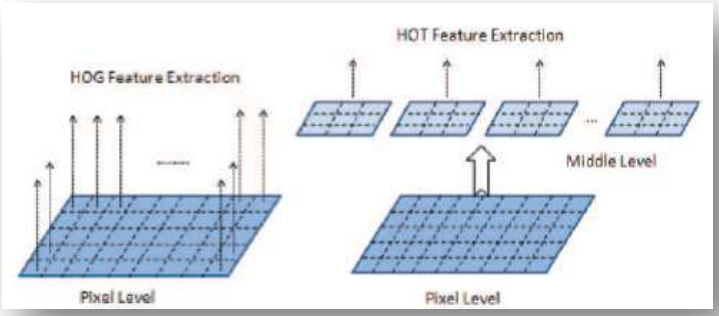


Figura 2.7 - Extração de *features* de diferentes níveis **[Tang2010]**.

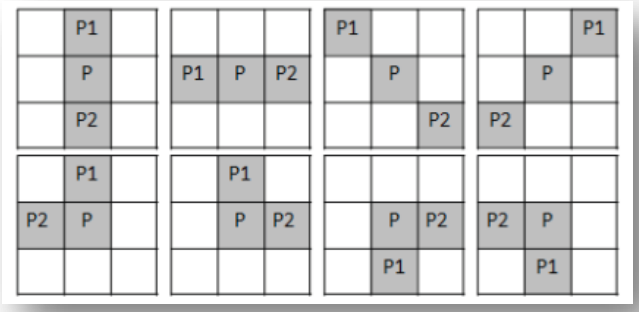


Figura 2.8 - *Templates* utilizados para cálculo das *features* **[Tang2010]**.

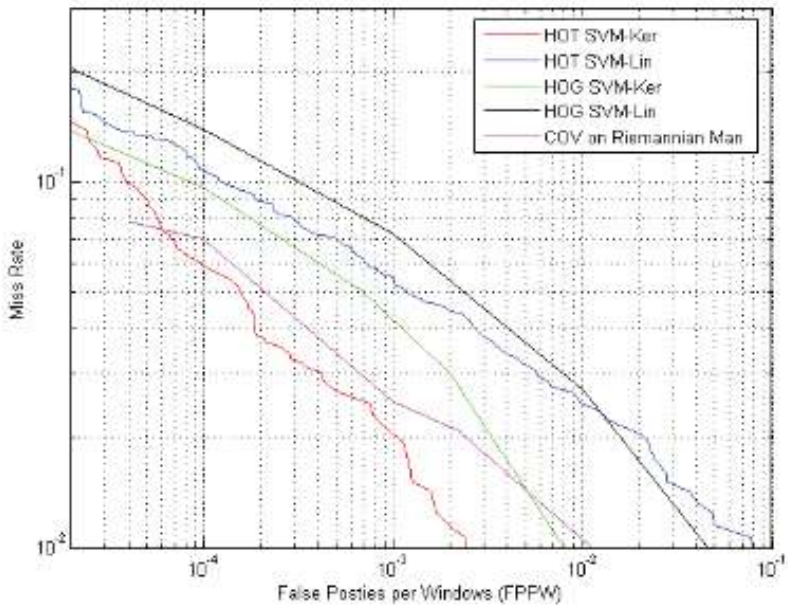


Figura 2.9 - Comparação do método desenvolvido com o HOG [Dalal2005] e o COV [Tuzel2008] **[Tang2010]**.

Liu et al em [Liu2009] introduziram um conceito inovador, que consiste num descritor de partição de gradientes granular e sintonizável, cuja abordagem se encontra descrita na figura 2.10. O conceito de granularidade é usado para definir a incerteza espacial e angular dos segmentos de linha no espaço *Hough*. Esta incerteza é então projetada na imagem. Ao se alterar o parâmetro de granularidade, o nível de incerteza pode ser controlado, podendo assim dar origem a uma família de descritores com propriedades de representação versátil. Então, os descritores GGP (*Granularity Gradient Partitions*) finamente granulados representam informação acerca da geometria do objeto (o mesmo que *Edgelet*). Por outro lado, descritores grosseiramente granulados fornecem informação relativa à representação estatística do objeto (o mesmo que o HOG). Adicionalmente, a posição, força, orientação e distribuição dos gradientes passam a ser incorporados num descritor unificado, com o objetivo de melhorar ainda mais a representação do GGP. Resultados provenientes de testes aplicados a esta abordagem permitem concluir que, embora a estrutura do GGP seja muito simples (apenas segmentos de linhas), no que toca à representação de pessoas, o descritor revela resultados dignos de comparação com os métodos já existentes (Figura 2.11), como o método proposto em [Tuzel2007] [Dalal2005].

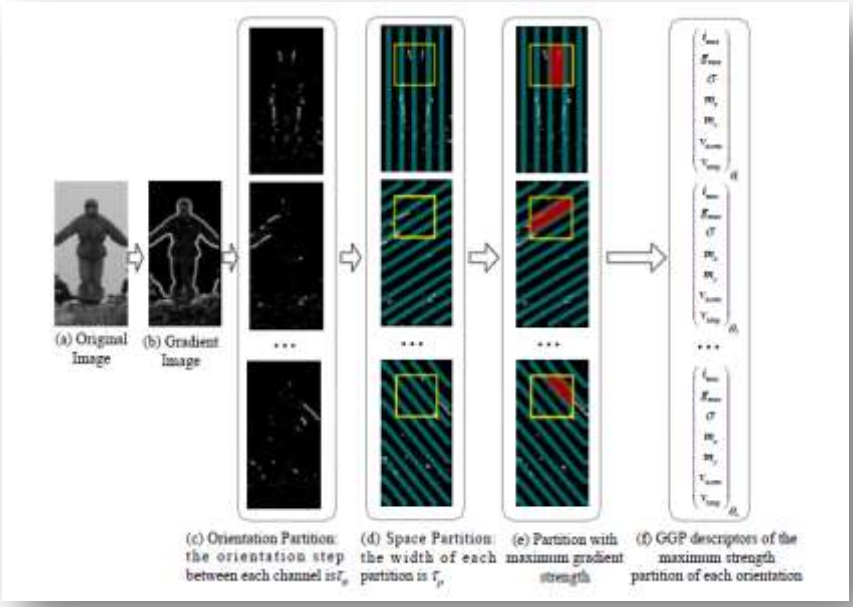


Figura 2.10 - Arquitetura da abordagem desenvolvida em **[Liu2009]**.

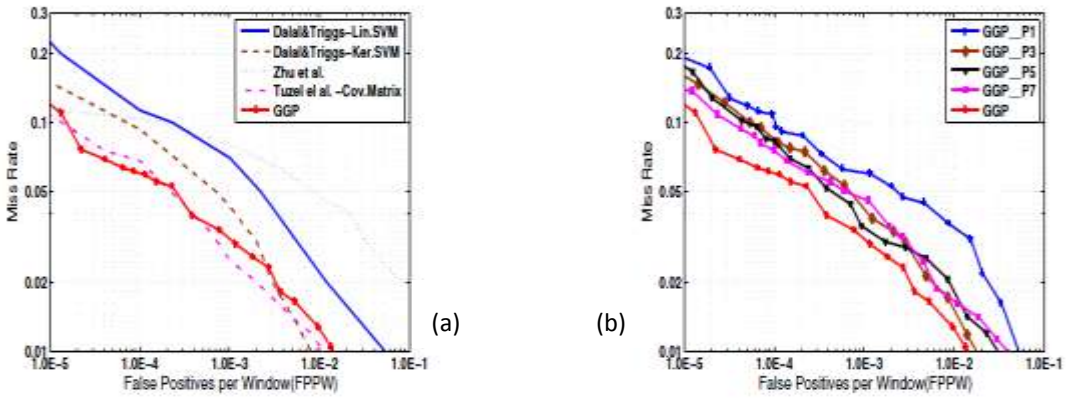


Figura 2.11 - (a) Comparação do método desenvolvido em [Liu2009] com outras abordagens, entre elas [Dalal2005] e [Tuzel2008], relativamente ao *dataset* da INRIA, (b) Avaliação do método utilizando definições de granularidade diferentes **[Liu2009]**.

Gao et al em [Gao2009] introduziram a *Adaptive Contour Feature* (ACF) ilustrada na figura 2.12, que deriva das amplitudes e orientações dos contornos num espaço de escala de regiões da imagem, denominados grânulos. Uma cadeia de *patches* quadrangulares (grânulos) num espaço granular orientado, descrevem os contornos de uma forma mais discriminativa que o HOG. Os resultados obtidos encontram-se ilustrados e comparados com outras *features* na figura 2.13.

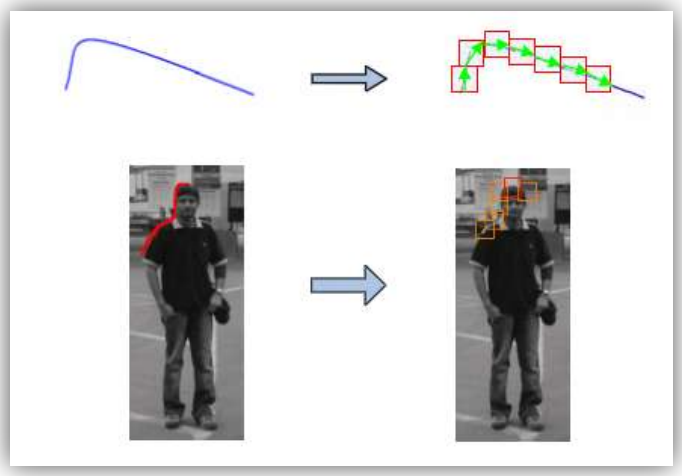
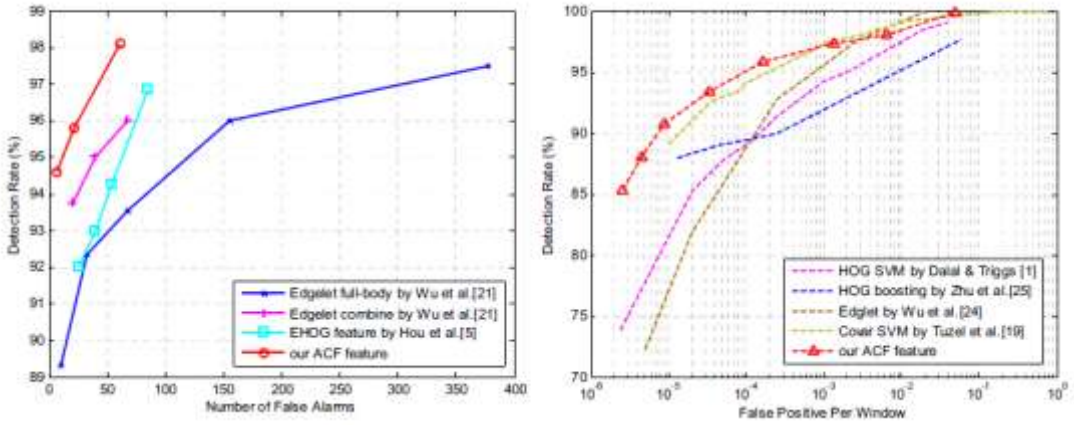


Figura 2.12 - Descrição de contornos através de *Adaptive Contour Feature* **[Gao2009]**.



(a) (b)

Figura 2.13 - (a) Comparação das curvas ROC relativas ao *dataset* USC, (b) comparação das curvas Taxa de Deteção/FPPW relativas ao *dataset* INRIA **[Gao2009]**.

Em [Tuzel2007], foram introduzidas as matrizes de covariância de *features* por *Tuzel et al*, utilizadas como descritores, sendo mais tarde em [Tuzel2008] [Odobez2008] [Tosato2010], implementados melhoramentos a este tipo de abordagem. As regiões de interesse (regiões onde possivelmente se encontra um indivíduo) são representadas pela matriz de covariâncias das características da imagem, como a localização espacial, intensidade, amplitude das derivadas de alta ordem, ou orientação de contornos. Uma pessoa é então caracterizada por várias matrizes de covariâncias resultantes da sobreposição das regiões como se pode verificar na figura 2.14.

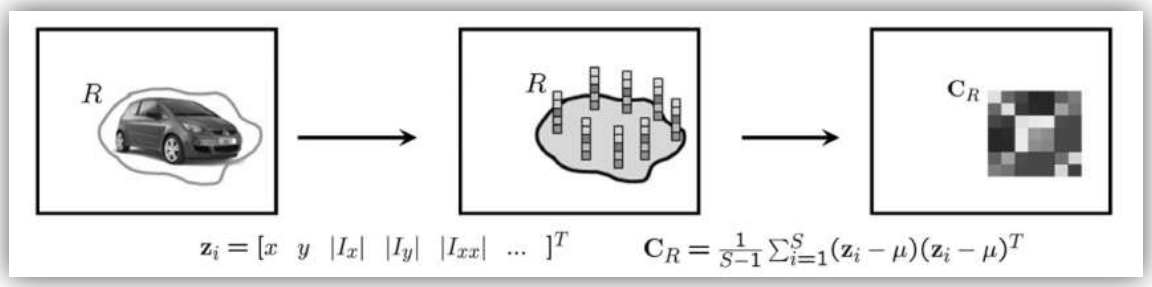


Figura 2.14 - Descritor de covariâncias. A região R é representada através da matriz de covariâncias C_R **[Tuzel2008]**.

Técnicas clássicas de aprendizagem não são adequadas para treinar este tipo de descritores, pois estas matrizes não se baseiam em espaços vetoriais, daí as matrizes de covariância não singulares serem formuladas a partir de um espaço Riemmaniano conectado (figura 2.15).

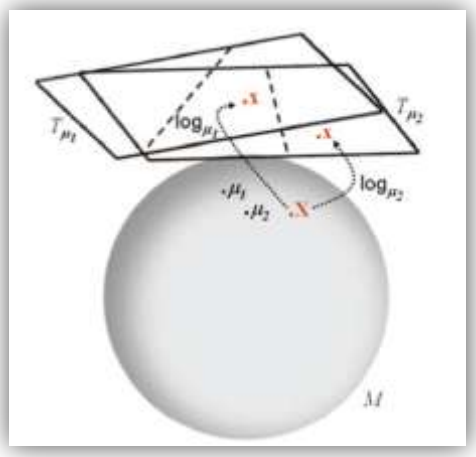


Figura 2.15 - Classificação no Espaço Riemmaniano **[Tuzel2008]**.

Em [Tuzel2007] [Tuzel2008] é então desenvolvida uma abordagem baseada na aplicação de um classificador *LogitBoost* (figura 2.16) a todas as janelas da imagem, utilizando como descritores as *features* de covariância. A grande contribuição deste artigo baseia-se numa nova abordagem para classificar pontos num espaço Riemmaniano através da incorporação da informação a priori sobre a geometria do espaço. Esta abordagem proporciona bons resultados relativos à tarefa da deteção como se pode verificar na figura 2.17, daí servir de ponto de partida para outras abordagens.

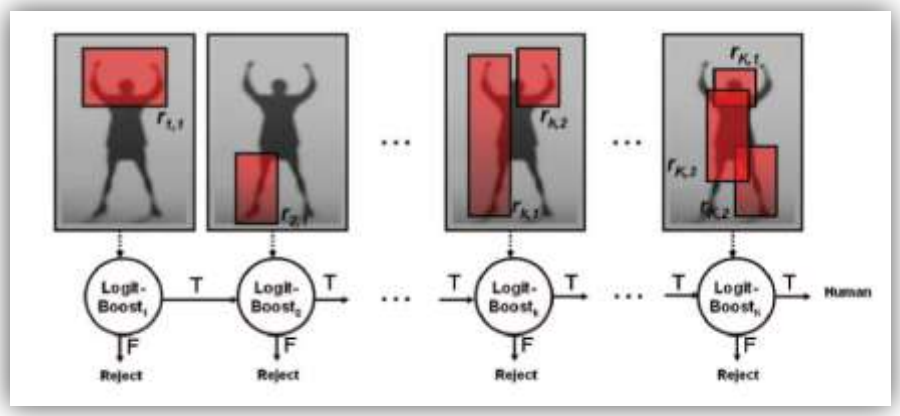


Figura 2.16 - Cascata de classificadores *LogitBoost* **[Tuzel2008]**.

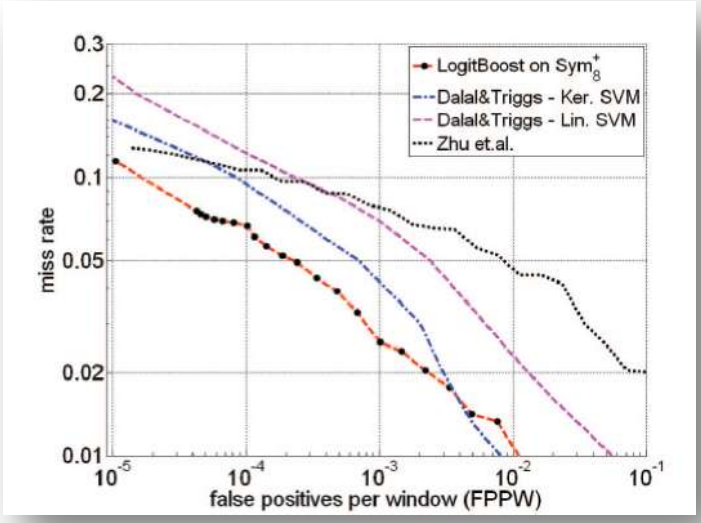


Figura 2.17 - Comparação do método desenvolvido em [Tuzel2008] com o HOG [Dalal2005] relativamente ao *dataset* da INRIA **[Tuzel2008]**.

LogitBoost é uma variante do algoritmo de *boosting Adaboost*, que é caracterizado como sendo uma otimização convexa, que consiste em minimizar uma perda logística. *LogitBoost* aprende iterativamente um conjunto de *weak classifiers* minimizando a verossimilhança probabilística binomial negativa da informação de treino, através de iterações Newtonianas. Em cada iteração a minimização é atingida, resolvendo-se um problema de regressão pesada de mínimos quadrados.

Tosato et al em [Tosato2010], propuseram uma alteração a [Tuzel2008], que torna o algoritmo mais simples, diminuindo o seu custo computacional, mantendo a performance. Visto que em [Tuzel2008] é implementado um algoritmo "ganancioso", o que implica que em cada *boosting iteration* é escolhido o *patch* mais discriminativo, ou seja, o que revela melhor performance na classificação, tal obriga a que exista um grande número de descritores de covariâncias a serem processados. Este facto motivou *Tosato et al* a seguir outra direção,

consistindo em informar o classificador quais são as áreas mais discriminativas do corpo humano, obrigando-o a se concentrar nestas. Este conceito resulta num classificador mais "forte" para cada parte do corpo, formando assim uma cascata de um número mais reduzido de classificadores "fortes" (figura 2.18), em vez de muitos classificadores "fracos". Os resultados obtidos encontram-se ilustrados e comparados com outros métodos presentes na literatura na figura 2.19.

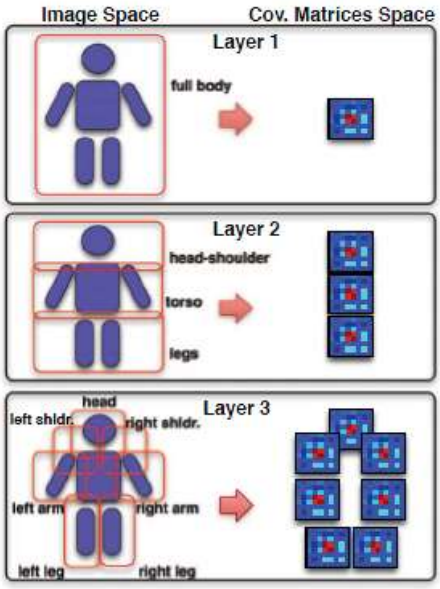


Figura 2.18 - Modelo humano baseado em partes. O corpo humano é dividido hierarquicamente em onze partes distintas, cada uma delas descrita por um descritor de matrizes de covariância **[Tosato2010]**.

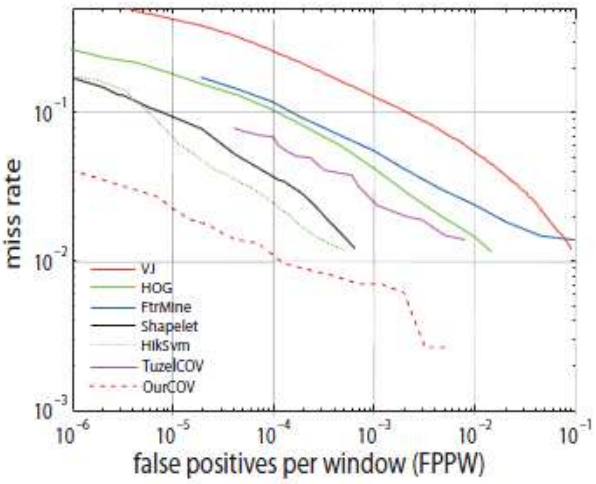


Figura 2.19 - Comparação do método desenvolvido, com outros presentes no estado da arte **[Tosato2010]**.

Motivados pelo desafio de detetar indivíduos em situações bastante complexas em imagens com pequena resolução *Li et al* [Li2009], propuseram um método baseado em EM (*Expectation Maximum*). Como os métodos baseados na segmentação de *blobs* não necessitam de grandes resoluções, foi decidido aplicar este tipo de metodologia. Numa fase inicial, são extraídas da imagem original *features* KLT (*Kanade Lucas Tomasi*). Uma máscara é obtida a partir da imagem original. Através de uma etapa de filtragem efetuada utilizando as máscaras de primeiro plano, maior parte dos *feature points* presentes na imagem de fundo são removidos, restando os pontos relativos aos indivíduos.

Em [Lin2007] [Lin2010] foi explorada uma abordagem Bayesiana de segmentação e deteção caracterizada por combinar templates globais da forma humana com outros, locais, discriminativos de várias partes do corpo, como se pode verificar na figura 2.20. É aplicado um método de *template matching* aos vários *templates* de partes, de forma a gerar um conjunto de possíveis hipóteses, otimizando-o segundo uma *framework* Bayesiana MAP, através de reavaliações de probabilidades globais e de análises de ocultações. É estudada a performance do algoritmo na tarefa de deteção com e sem uma etapa de subtração de fundo, demonstrando melhores resultados quando esta é efetuada.

Conclui-se também, que o modelo da árvore de *templates* de parte proposto (Figura 2.20) captura bem as articulações do corpo, gerando uma deteção robusta e eficiente.

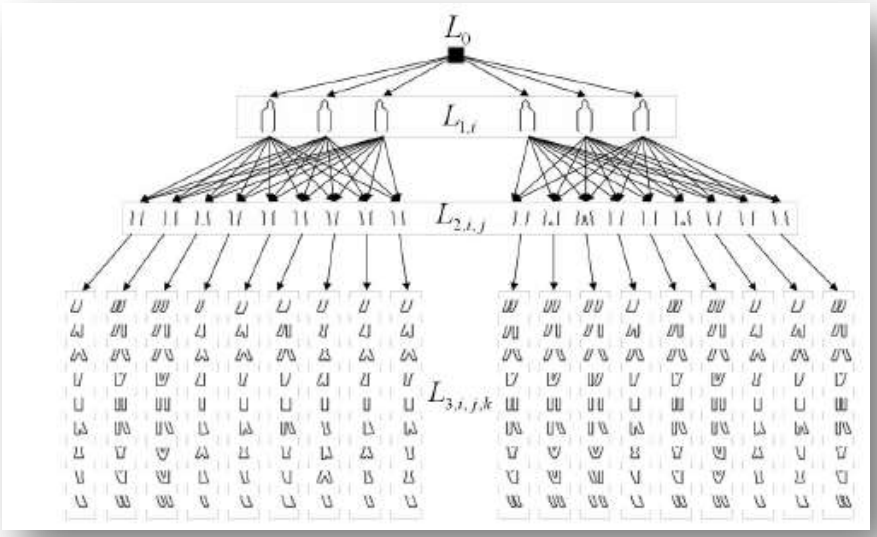


Figura 2.20- Árvore de *templates* de partes **[Lin2007].**

2.1.2 -Deteção baseada em *features* de aparência

Ao se identificar as limitações existentes na utilização das características de forma, surgiu a necessidade de se identificar outros tipos de *features* com outras características discriminativas, surgindo assim as *features* de aparência, baseadas por exemplo na intensidade ou textura da imagem.

Uma das features de aparência mais usuais são as *Haar Wavelets* (figura2.21), utilizadas em imagens *monocromáticas*, para descrever os objetos a partir da sua textura, codificando as diferenças orientadas de intensidades entre regiões adjacentes, [Papageorgiou2000], [Viola2001], [Viola2003]. Em [Dollar2007], foram utilizadas as Haar generalizadas (figura 2.22), que consistem numa extensão das *Haar Wavelets* originais com configurações mais diversificadas.

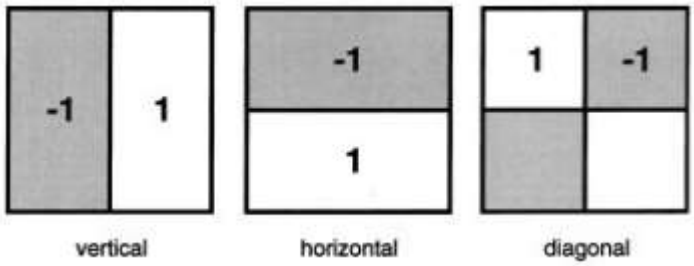


Figura 2.21 - Representação dos três tipos de Haar *wavelets* 2D: vertical, horizontal e diagonal **[Papageorgiou2000]**.

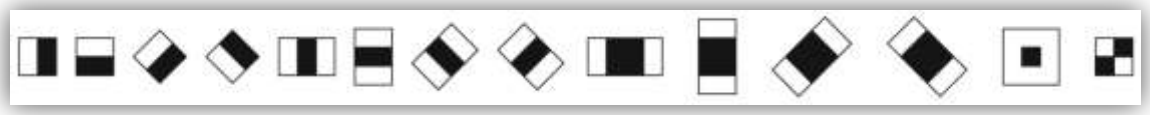


Figura 2.22 - Representação das Haar *wavelets* generalizadas **[Gavrila2009]**.

Avanços recentes na área de machine learning foram também aproveitadas para lidar com os desafios que a deteção de indivíduos proporciona. Por exemplo, para adaptar a classificação à articulação do corpo humano, em [Pang2008] foi aplicado o *Logistic Multiple Instance Boosting* (LMIB). Utiliza como *features* as *Haar Wavelets* [Viola2003]. Para cada janela de deteção, onde esteja presente um indivíduo, são gerados conjuntos de instâncias, que serão classificados positivamente, caso pelo menos uma das suas instâncias seja classificada como positiva e vice-versa. Na figura 2.23 encontra-se descrita performance deste método, comparativamente com outros referidos na literatura.

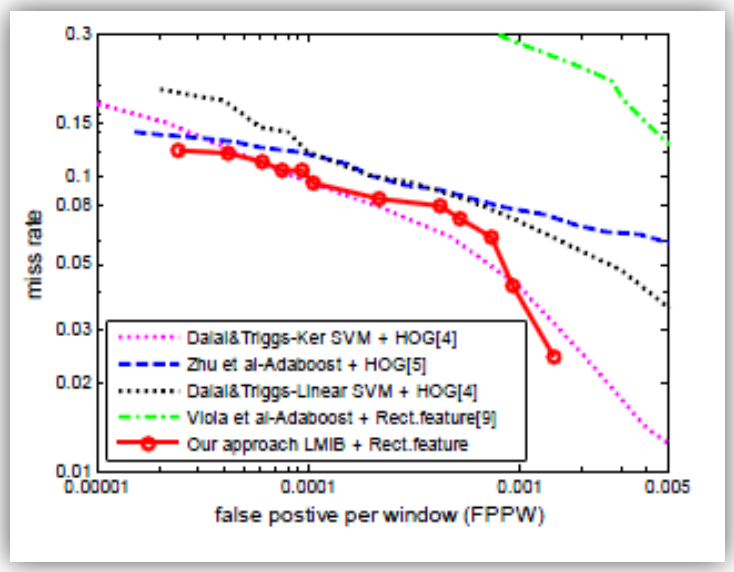


Figura 2.23 - Comparação da performance do algoritmo proposto em [Pang2008], com outros métodos referidos na literatura, mais especificamente [Dalal2005] e [Viola2003] **[Pang2008]**.

Num período mais recente, em [Yadong2008] foram introduzidos os *Local Binary Patterns* (LBP) para descrever o modelo humano, como se pode observar na figura 2.24. Até então os LBP eram apenas utilizados para classificação de texturas. O aparecimento deste tipo de *features* surgiu com a necessidade de se criar um tipo de características fiáveis em termos de mudanças de luminosidade, devido ao seu poder discriminativo, e pelo facto de possuírem uma complexidade computacional bastante pequena, o que resulta num processo de deteção com complexidade e tempos de processamento consideravelmente mais baixos daí serem regularmente utilizadas [Yadong2008] [Wang2009] [Nguyen2010] [Zeng2010].

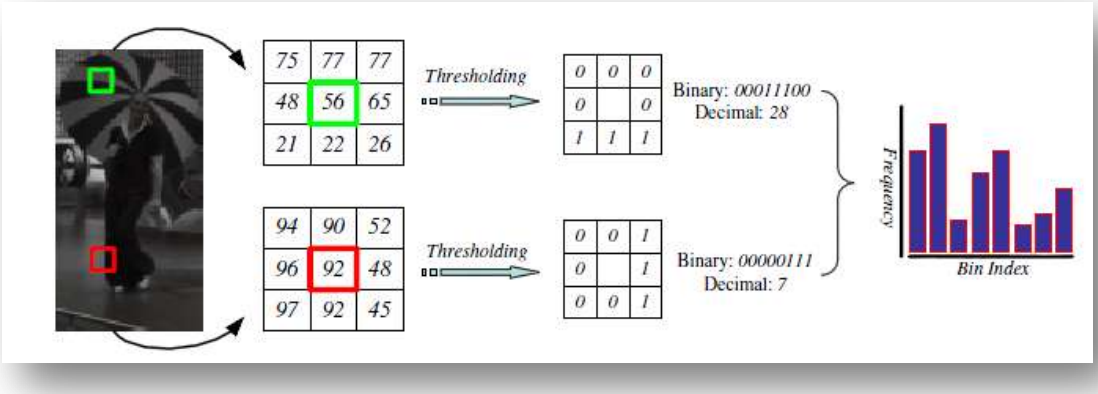


Figura 2.24 - Ilustração do LBP **[Yadong2008]**.

Nguyen et al construíram em [Nguyen2010] uma variante do descritor LBP, o NRLBP (*Non-Redundant Local Binary Pattern*), utilizada para codificar a aparência local dos indivíduos, como se pode observar na figura 2.25. O NRLBP foi criado com o intuito de melhorar principalmente dois tipos de limitações que o LBP tradicional encontra, relacionados com o poder de armazenamento e discriminativo. Relacionado com o primeiro aspeto, o LBP original necessita de um número elevado de *bins* de histogramas, com o NRLBP o número de *bins* é reduzido para metade. Relativamente ao segundo, o LBP original é sensível a mudanças relativas entre o plano de fundo e o de primeiro plano. Depende rigorosamente das intensidades de locais particulares, que no caso de pessoas pode variar muito, principalmente devido à grande variedade de vestuário existente. Resumidamente o NRLBP assume que o código LBP e o seu complemento são equivalentes. A comparação da performance deste método com outros presentes na literatura encontra-se descrita nas figuras 2.26 e 2.27.

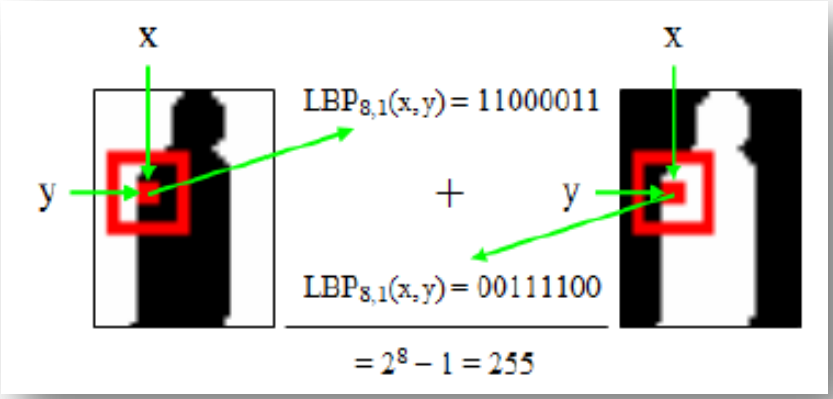


Figura 2.25 – Representação do mesmo indivíduo com contrastes diferentes entre o plano de fundo e a imagem de primeiro plano. De notar que os códigos LBP são complementares**. [Nguyen2010]**.

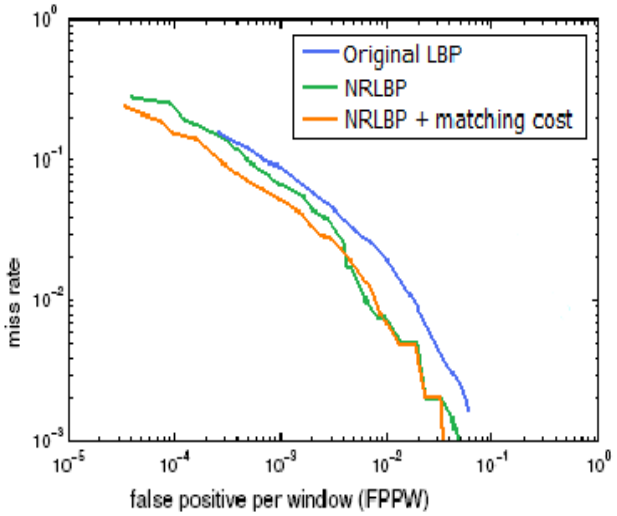


Figura 2.26 – Comparação da performance do método quando utilizadas apenas *features* de aparência e quando são integradas *features* de forma **[Nguyen2010]**.

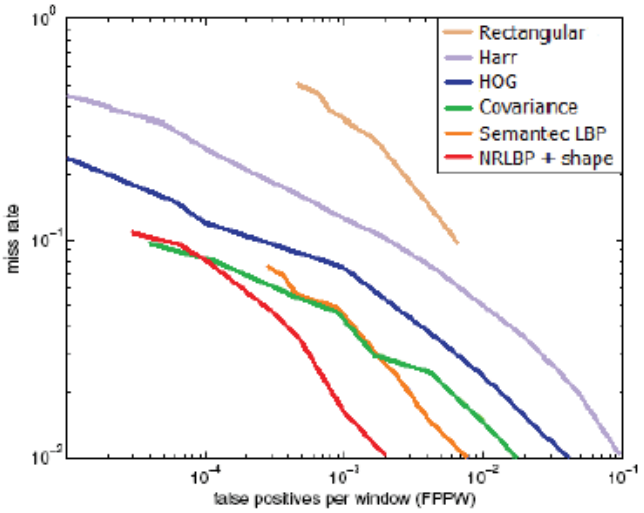


Figura 2.27 – Comparação do detetor desenvolvido com outros existentes no estado da arte **[Nguyen2010]**.

Em [Ott2009] uma variante do HOG foi introduzida, o CHOG. Esta nova abordagem consiste em codificar janelas de recursos específicos que se adaptam às características locais da imagem. A característica usada para diferenciar o primeiro plano do plano de fundo é a cor. Esta técnica destaca-se do HOG pelo facto de usar estimativas locais das estatísticas das imagens de primeiro plano e de fundo diretamente no descritor. Desta forma os contornos das regiões de primeiro plano são amplificadas e as variações de sombreamento e textura são reduzidos (figura 2.28). Esta segmentação suave é efetuada de uma forma computacionalmente eficiente através de projeções lineares locais, tornando assim esta abordagem adequada para um esquema de janela deslizante. Este método incorpora as pistas de segmentação diretamente no processo de extração e aprendizagem, estando aí uma das suas grandes vantagens. Os resultados relativos à performance deste método encontram-se descritos na figura 2.29.

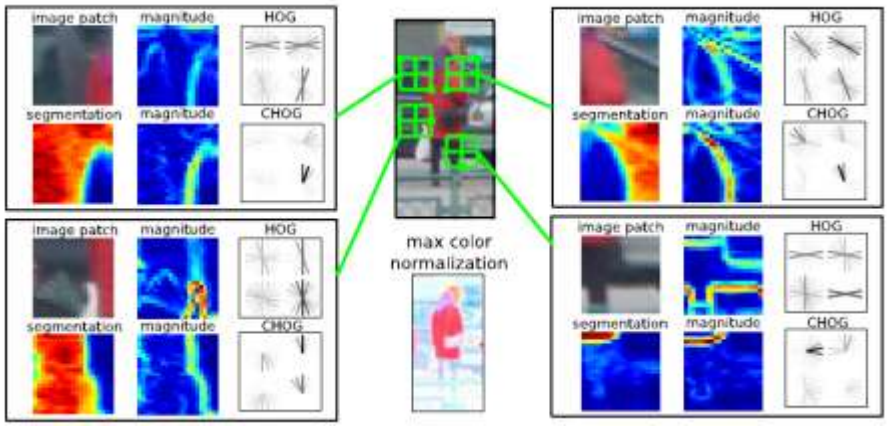


Figura 2.28 – Vista global da extração de *features* **[Ott2009]**.

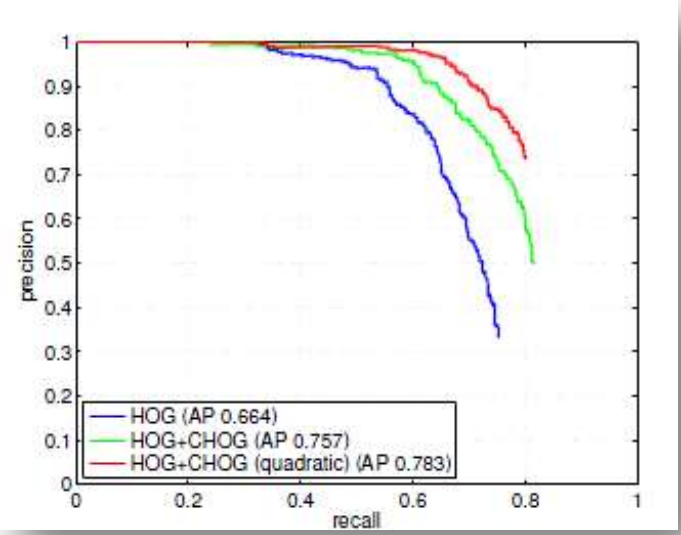


Figura 2.29 – Comparação de *Precision*/*Recall* com e sem a integração do CHOG **[Ott2009]**.

2.1.3 -Deteção baseada em *features* de movimento

A informação de movimento existente nas sequências de vídeo, também pode ser explorada, para melhorar o poder discriminativo dos descritores. O movimento pode ser utilizado como ferramenta para diferenciar um objeto de outro, o que permite melhorar a descrição de um objeto. O ser humano normalmente efetua movimentos cíclicos, que podem ser transformados em padrões.

Existem muitos métodos desenvolvidos, que utilizam informação de movimento, descrevendo-o através de diferenças temporais [Viola2003] [Nguyen2011] [Haga2004] ou mesmo fluxos óticos [Wojek2009] [Tang2009] [Dalal2006].

Em [Viola2003] são utilizadas *features* retangulares (figura 2.30), que são retiradas da subtração de imagens de *frames* consecutivas, com o objetivo de codificar padrões de movimento.

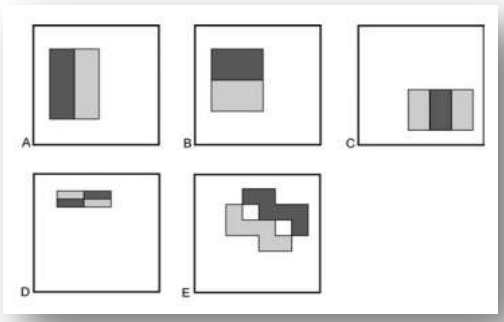


Figura 2.30 – *Features* retangulares que codificam padrões de movimento **[Viola2003].**

Em [Haga2004] é proposto mais um método que efetua a deteção de pessoas baseada em informação de movimento local de cada região. Os padrões de movimento são descritos utilizando informação proveniente da diferença entre *frames* consecutivos A partir do movimento local existente em cada região são retiradas três características fundamentais: a singularidade temporal e espacial da imagem em movimento e a continuidade temporal do movimento. Através destas três variáveis o algoritmo é capaz de calcular de uma forma precisa a velocidade da região modificada, o que possibilita uma redução do erro de seguimento de cada indivíduo ao longo das várias *frames*. Este método não se baseia em modelos de forma humana. O foco deste sistema está no facto de utilizar as três propriedades de movimento referidas acima, com o objetivo de descrever de uma forma mais eficiente as diferenças de fundo e de movimento dos indivíduos, conseguindo assim efetuar uma deteção mais eficiente. Um dos aspetos menos positivos presentes neste artigo está relacionado com o facto de não conseguir detetar pessoas que não estejam em movimento.

Em [Moeslund2004], é implementado um melhoramento ao algoritmo SMC (*Sequential Monte Carlo*), que é regularmente utilizado na literatura para capturar o movimento de objetos. O SMC é então aplicado para capturar o movimento de cadeias articuladas como braços ou pernas. Concretamente é utilizada a posição da mão como informação auxiliar. É introduzida informação auxiliar proveniente da *frame* correspondente, com o objetivo de corrigir as partículas que foram previstas. Comparando esta abordagem com o SMC *standard*, conclui-se que as partículas obtidas são mais corretas, devido ao facto de que a função de importância introduzida ser mais precisa que a utilizada no SMC *standard*.

Em [Grahn2005] é estudada a eficiência computacional relativa ao processo de classificação humana, através de informação de movimento. São aplicados filtros lineares de diferenças espácio-temporais, para representar o movimento presente na imagem, como proposto em [Viola2003]; devido à sua linearidade, as respostas destes filtros podem ser codificadas de forma eficiente. Os padrões de movimento são posteriormente classificados através de SVM, que provou ser eficiente em casos de espaços de *features* não lineares de grandes dimensões. Por fim, é adotada uma arquitetura em cascata, para explorar o facto de que maior parte das janelas são facilmente classificadas, enquanto as restantes são mais complicadas de classificar.

Em relação à utilização de fluxos óticos, em [Dalal2006] foram concebidos histogramas de fluxos (HOF), baseados em fluxos diferenciais, que foram usados para descriminar não só movimentos nos limites dos objetos, como também padrões de movimentos no interior dos mesmos.

Em [Su2009] foi desenvolvido um método de segmentação de sequências de vídeo baseado em informação de movimento. Inicialmente é aplicada uma operação de subtração de três *frames* consecutivos, para determinar áreas onde exista movimento legítimo, e de seguida é aplicado um método de subtração de fundo adaptativo para extrair a região em movimento

completa. Por fim é aplicada uma janela deslizante, baseado num *threshold* adaptativo, com o intuito de determinar as fronteiras dos segmentos.

Em [Borges2012] são introduzidas estatísticas de movimento no processo de deteção. São analisados comportamentos cíclicos na trajetória dos *blobls* e a relação entre o tamanho do *blob* e a sua posição vertical, devido ao facto de que a altura de um *blob* diminui à medida que este se afasta da câmara. Ao contrário de outras abordagens, em [Borges2012] não são analisados padrões de movimento locais, como de pernas ou braços, pois o sistema foca-se na análise de padrões de movimento do *blob* como um todo. No entanto, antes de ser efetuada a análise de movimento é necessária uma etapa de subtração de fundo, para segmentar as regiões de interesse a serem analisadas. Estas duas vertentes de informação são então utilizadas em conjunto através de um classificador de *Bayes*. A escolha deste classificador advém do facto de este fornecer um erro médio mínimo para os padrões gaussianos distribuídos.

2.2 - Deteção baseada em combinações de *features*

Existem várias abordagens na literatura que tiram partido da ideia de combinar várias *features* diferentes, de forma a melhorar a performance dos algoritmos de deteção. De facto, torna-se interessante explorar a informação proveniente de vários tipos de *features*, visto que a informação a ser codificada se torna mais diversificada e abrangente, no entanto a capacidade computacional exigida aumenta, bem como os tempos de processamento. De facto, cada *feature* possui as suas limitações, logo torna-se interessante explorar a ideia de combinar *features* diferentes, de forma a colmatar as suas limitações.

Em [Viola2003] são utilizadas *features* retangulares, que são retiradas da subtração de imagens de *frames* consecutivas, com o objetivo de codificar padrões de movimento (figura 2.30). Esta técnica consiste em combinar informação de intensidade com informação de movimento, com o fim de construir um detetor único que analisa duas *frames* consecutivas de uma sequência. Para tirar partido da informação vinda destas características, o detetor é treinado para reconhecer padrões através do *AdaBoost* em cascata, com o objetivo de diminuir o tempo despendido no processo de classificação. O detetor ao ser aplicado à imagem procura um padrão de intensidades que define o objeto. O objeto é então seguido ao longo de vários *frames* com a intenção de se reconhecer padrões de movimento ou outras pistas que auxiliem a deteção. O método descrito em [Viola2003], foi elaborado com o intuito de desenvolver uma representação de movimento eficiente e que proporcione boas performances mesmo com baixas resoluções e em condições adversas, como chuva ou neve.

Leibe et al em [Leibe2005], exploraram também a vantagem da utilização de diversas features, bem como a integração de informação local e global. Este artigo contribui com a proposta de um novo esquema de integração de pistas, baseado nas hipotéticas segmentações

efetuadas na imagem de entrada. É combinada a informação local proveniente da amostragem de *features* de aparência (uma extensão de escala-invariante do ISM), com pistas globais acerca da silhueta dos objetos, o que faz com que o *Chamfer matching* atinja níveis mais elevados de robustez a variações de escalas, à complexidade da imagem e a ocultações parciais. *Chamfer matching* consiste em procurar locais na imagem, onde existe a melhor correspondência possível, entre um conjunto treinado de *templates* de forma. As formas dos objetos são comparadas usando uma DT (*Distance Transform*), que calcula para cada píxel a distância para o feature píxel mais próximo.

Em [Beleznai2009] *Beleznai et al*, construíram um sistema de deteção de pedestres que combina pistas de movimento genéricas e um detetor baseado em aparência. É utilizado um método de *clustering* denominado, *scale adaptive mean shift*, para segmentar as regiões de interesse extraídas da imagem através de informação de movimento, por via de um método de subtração de fundo adaptativo. O detetor de aparência é então aplicado às regiões de interesse segmentadas, com o objetivo de verificar se existe algum indivíduo nessa área. Para isso são extraídas *features* HOG que irão ser verificadas através de um classificador *AdaBoost*, terminando assim o processo de deteção.

Uma técnica baseada num detetor Bayesiano é proposto em [Bischof2009]. Este detetor utiliza pistas relativas à forma e movimento, com o objetivo de obter uma solução MAP para aplicações em cenários com câmaras estacionárias. [Bischof2009] fornece duas contribuições principais. Primeiro, foi introduzido um conceito baseado na integração de contornos em imagens integrais, que demonstrou aumentar significativamente a velocidade de processamento na etapa de *template matching* e análise de ocultações. De seguida, um descritor de forma é estimado não parametricamente, gerando hipóteses fiáveis de possíveis deteções em situações de ocultação. A combinação de informação local e global demostrou bons resultados em situações de grande complexidade.

Em [Nguyen2011], a informação de movimento local é codificada usando o NRLBP. A informação de movimento é retirada dos contornos do objeto, para codificar o movimento de partes do corpo, como pernas ou braços e são codificadas através da diferença entre duas *frames* consecutivas, como se pode observar na figura 2.31. É importante retirar a informação proveniente dos contornos, pois para além de serem uma ferramenta poderosa na descrição da forma de um objeto, informações vindas do plano de fundo ou do interior das regiões de interesse são irrelevantes. Os resultados obtidos encontram-se descritos na figura 2.32.

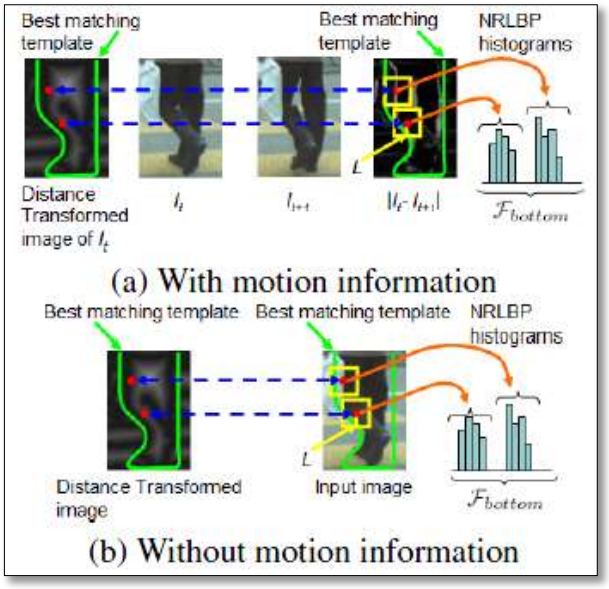


Figura 2.31 – Formação do vetor de *features* **[Nguyen2011]**.

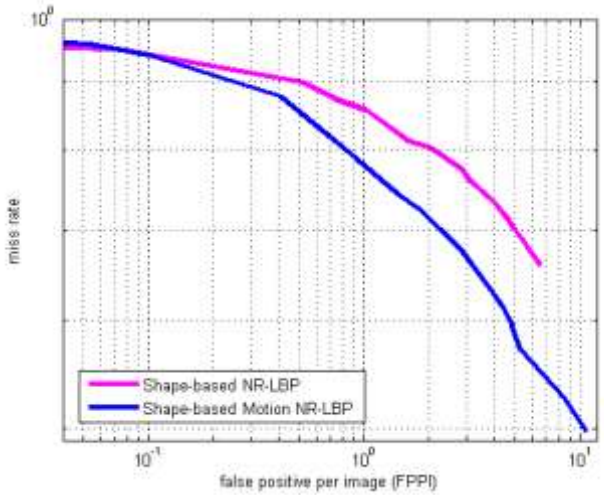


Figura 2.32 – Avaliação e comparação da performance da abordagem descrita em [Nguyen2011] (com informação de movimento) com o método implementado em [Nguyen2010] (sem informação de movimento) **[Nguyen2011]**.

Em [Junqiu2012] são exploradas features de aparência (para deteção de pedestres e sombras) e movimento (para filtrar os resultados de deteção). É utilizado um método de subtração de fundo para se obter as regiões de primeiro plano e de sombras, através de um modelo que é criado usando distribuições Gaussianas múltiplas para cada píxel. Foram também criados modelos das diferentes partes do corpo humano, que integrados formam *templates* de forma humana. Através da informação retirada das regiões de primeiro plano e de sombra é efetuada deteção de partes através de *template matching*, baseado na função da distância de *Chamfer*. Resultados provenientes da deteção são então filtrados utilizando informação de movimento. Este método produz uma baixa taxa de falsos positivos, devido à

integração de vários tipos de *features*, e tem bastante sucesso no que toca a ocultações, visto que é utilizada informação relativa ao posicionamento e forma das sombras (Figura 2.33) com o objetivo de reconhecer pedestres mesmo que obstruídos.

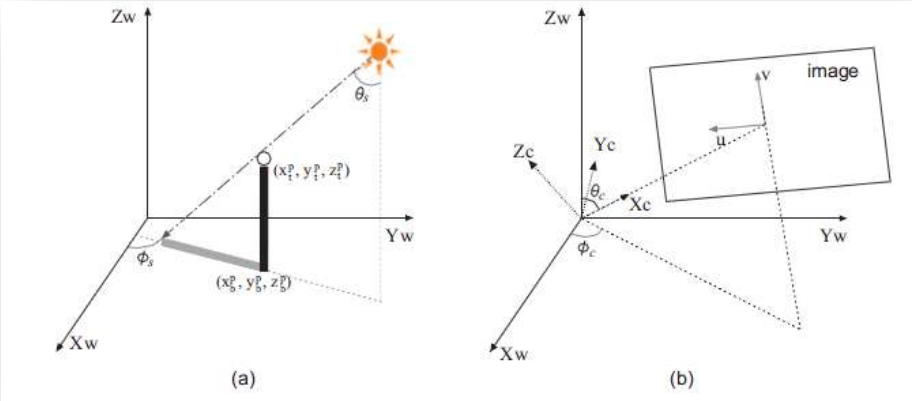


Figura 2.33 – (a) Representação do sistema de coordenadas 3D da câmara, tendo em consideração a posição do Sol. (b) Representação da influência da posição do Sol e do tamanho do objeto, na posição e comprimento da sua sombra **[Junqiu2012]**.

Em [Wojek2009] foi estudada a integração de informação de movimento (HOF (*Histograms of Flow*)), com outros tipos de *features* como o HOG ou Haar *wavelets* para melhorar a performance dos algoritmos. Foi também analisada a combinação de várias características com diferentes classificadores, como o HIKSVM, MPLBoost, Adaboost ou SVM Linear, que revelou ser uma escolha decisiva para se atingir a máxima performance. Os resultados encontram-se ilustrados na figura 2.34.

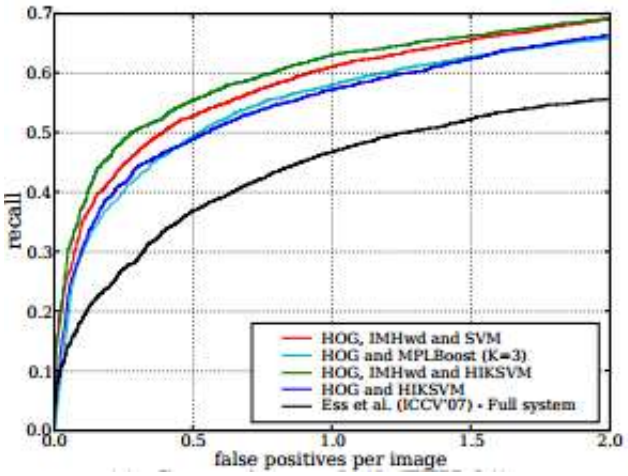


Figura 2.34 – Resultados relativos às abordagens desenvolvidas em **[Wojek2009]**.

Tang et al propuseram em [Tang2009], combinar *features* de aparência e movimento a serem utilizadas na tarefa de deteção. O fluxo ótico é calculado através da diferença entre *frames* consecutivas. A informação de movimento (uma adaptação do HOT para movimento) extraída, representa não só movimentos globais referentes aos limites delineados pelo corpo humano, como também padrões de movimento locais causadas pelos membros. A informação de movimento é combinada com o HOT, constituindo assim a *feature* final a ser utilizada na deteção. Como já referido anteriormente o HOT representa informação de textura e de gradiente. Como classificador é utilizado o SVM nesta experiência. Uma visão global do método encontra-se na figura 2.35.

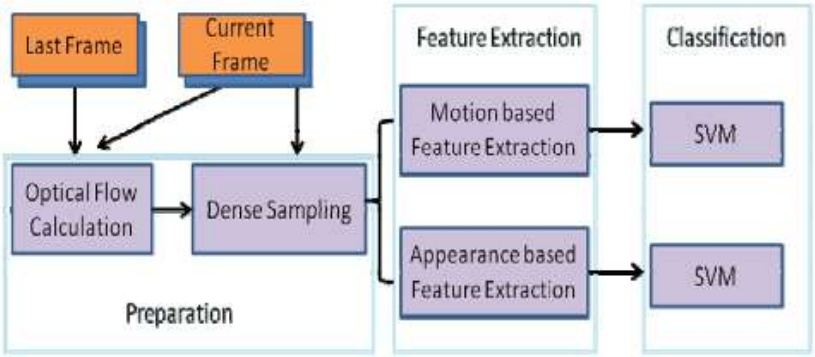


Figura 2.35 - Vista geral da abordagem implementada em **[Tang2009]**.

Em [Wang2009] *Wang et al* exploraram a combinação de dois tipos de *features*, o HOG e o LBP em conjunto com dois tipos de detetores que são treinados usando SVM linear, sendo um deles um detetor global aplicado nas janelas de decisão e um detetor de partes responsável por regiões locais. A integração do HOG com o LBP foi desenvolvida com o principal intuito de atingir bons resultados em situações de ocultação parcial de indivíduos. Para cada janela de decisão são construídos mapas de probabilidade de ocultações através da resposta de cada bloco HOG. Cada mapa é então segmentado utilizando uma abordagem *Mean-Shift*. Por sua vez a região segmentada da janela com o maior número de respostas negativas é definido como uma região onde existe ocultação. De seguida são aplicados detetores de partes às regiões da janela onde não existe ocultação, com o objetivo de detetar a presença de um indivíduo. Esta abordagem revela uma performance robusta, no que toca à deteção em ambientes em que ocorram várias ocultações, daí se tornar objeto de melhoramento em vários artigos como [Zeng2010].

Outro exemplo da utilização do HOG em simultâneo com outra *feature* encontra-se em [Zeng2010], onde mais uma vez é integrado com o LBP, visto que a informação de contornos conjugada com informação de textura revela ter um poder discriminativo elevado. Esta integração tem por objetivo melhorar a tarefa de deteção, especialmente no que toca a obstruções parciais de indivíduos. Neste caso, o objetivo consiste em diminuir o tempo de

processamento da abordagem proposta em [Wang2009]. Para atingir tal objetivo foi utilizado uma estrutura de dois estados, baseada numa cascata de classificadores que vão filtrando a informação, começando por um conjunto de classificadores mais grosseiros, que vão sendo cada vez mais refinados. De forma a melhorar o retorno de cada estado alterou-se o mi-SVM (*multiple instance Support Vector Machines*) para treinar o HOG e o LBP. O MIL foi aplicado com o objetivo de diminuir o tempo de processamento do algoritmo e obter um *recall* elevado em cada estado de uma cascata de classificadores SVM, originando o mi-SVM, que produz resultados semelhantes com um tempo de processamento dez vezes inferior, mantendo os níveis de precisão (Figura 2.36).

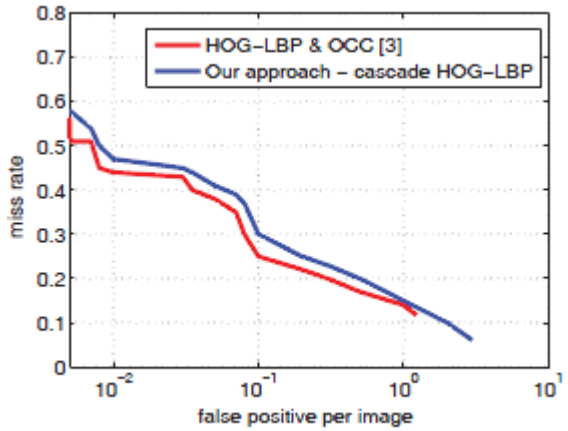


Figura 2.36 – Comparação da performance da cascata de rejetores proposta com o método desenvolvido em [Wang2009] **[Zeng2010]**.

O trabalho desenvolvido em [Walk2010], explora a introdução das estatísticas de segunda ordem da cor (CSS), como *feature* adicional a várias outras *features*, como o HOG [Dalal2005], HOF [Wojek2009] e o LBP [Yadong2008], que demonstrou ser uma boa opção a tomar, visto melhorar consideravelmente a performance dos algoritmos como se pode verificar nos gráficos ilustrados na figura 2.37.

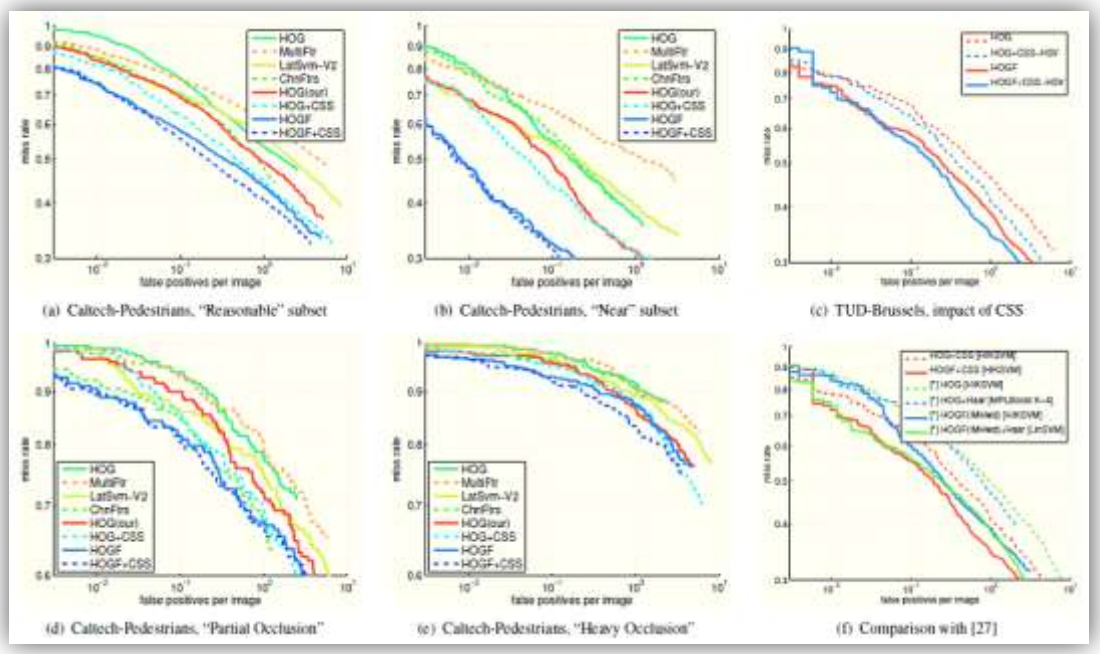


Figura 2.37 - Avaliação e comparação do impacto na performance de vários métodos do estado da arte, da integração do CSS **[Walk2010]**.

Em [Bansal2010] é utilizado estéreo como ferramenta para a deteção, em conjunto com informação de contorno e aparência. São usados mapas de profundidade para gerar deteções através de *template matching* de estruturas 3D humanas. No que toca à classificação, são treinados vários classificadores para classes individuais (diferentes tipos de objetos), sendo codificadas *features* invariantes 3D, que são combinadas por meio de uma cadeia de *Markov*, com o intuito de melhorar as relações espaciais 3D. Para aperfeiçoar as deteções provenientes do estéreo, múltiplos classificadores baseados no descritor HOG são melhorados através de *template matching* com templates de contorno de vários fragmentos do corpo humano 2D, o que provoca um aumento da precisão do algoritmo.

Enzweiler et al em [Enzweiler2010], integraram fluxos óticos com informação de intensidade e profundidade (*stereo*), com o objetivo de tratar situações de ocultação parcial de indivíduos. Foram motivados pela possibilidade de tirar partido do facto de que, nos limites de ocultações parciais de objetos existem descontinuidades locais de movimento e profundidade. O sistema baseia-se numa fase de segmentação responsável por extrair da imagem, regiões com informação de movimento e profundidade. Através dos resultados da segmentação, vários parâmetros relacionados com as áreas obstruídas, são definidos, que integrados com um conjunto de classificadores *component-based* treinados em *features* de profundidade e movimento, concluem o processo de deteção.

2.3 - Estratégias e desafios em imagens termográficas

Esta secção destina-se ao estudo e análise de aspetos relacionados com a deteção de pessoas em imagens térmicas.

Em [Haritaoglu2000], é proposto um método a ser integrado num computador *low-cost* utilizado num sistema de vigilância em tempo real, que processa fontes de vídeo monocromáticas estacionárias que podem ser visíveis ou de infravermelhos, visto que este sistema foi construído com o intuito de ser aplicado em vigilâncias noturnas, daí não tirar partido de características como a cor. Constrói modelos dinâmicos baseados no movimento e aparência dos indivíduos com o fim de se obter bons resultados mesmo em situações de ocultação. Este sistema aprende a modelar imagens de fundo estatisticamente para detetar objetos de primeiro plano mesmo quando o modelo de fundo não é estacionário. Depois de um modelo de fundo ser criado, os pixéis mais relevantes são agrupados em *blobs* que posteriormente irão ser codificados, sendo feita uma análise de silhueta, onde a partir de modelos de silhueta estáticos e de análises periódicas efetuadas a cada *blob* classificando-o numa de três categorias: Pessoa sozinha; Grupo de pessoas; Outros objetos.

Em [Bertozzi2005], é mais uma vez implementado um sistema de deteção de pedestres em imagens estéreo de infravermelhos. O sistema é caracterizado por incorporar três abordagens diferentes, deteção de zonas quentes, deteção baseada em contornos, e computação *v-disparity* [Labayrade2002], como se pode verificar na figura 2.38. Estéreo é também utilizado para calcular a distância e tamanho dos objetos. É efetuada uma validação final do objeto através de características térmicas e outras baseadas na morfologia da cabeça humana. A primeira abordagem está encarregue de detetar zonas quentes, enquanto as outras duas se concentram na deteção de objetos frios que podem potencialmente também ser pedestres. O algoritmo agrupa os objetos detetados com coordenadas semelhantes, criando assim uma lista de áreas de interesse, áreas estas que têm de ter um tamanho e um *aspect-ratio* específico. As áreas selecionadas são depois filtradas utilizando modelos de cabeça que codificam características morfológicas e térmicas.

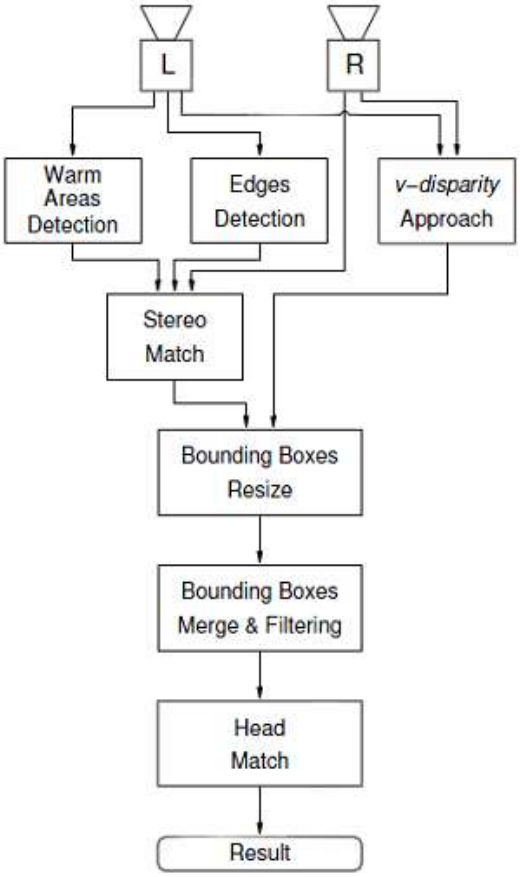


Figura 2.38 – Arquitetura da abordagem implementada em **[Bertozzi2005].**

Em [Kumar2006] é proposto um algoritmo que funde informação proveniente de imagens naturais e térmicas, com o objetivo de colmatar as limitações que estas possuem, de forma a aumentar a performance do sistema. A estratégia consiste numa etapa de deteção que utiliza informação de gradiente juntamente com uma etapa de subtração de fundo. O esquema ilustrado na figura 2.39 representa a arquitetura geral da abordagem.

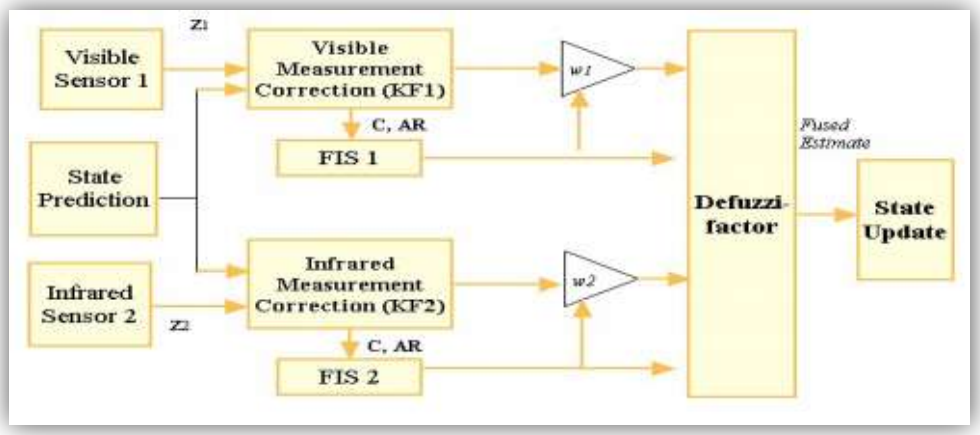


Figura 2.39 – Arquitetura da abordagem desenvolvida em **[Kumar2006]**.

Jin Yun et al desenvolveram em [Yun2007] um sistema de deteção aplicável a imagens de infravermelhos. Consiste inicialmente numa etapa de segmentação, que é feita através de um método denominado *K-Means clustering* [Kanungo2002], que demonstrou ser bastante robusto. Foi também proposto um esquema de extração de features baseado no *histogram of maximal oriented energy map*, usando como filtros para seleção de orientações, *Log-Gabor wavelets*. O classificador utilizado é o SVM.

Em [Soga2008] é combinada informação de forma e intensidade, com o intuito de se efetuar deteção de pessoas em sistemas de infravermelhos. É utlizada uma câmara NIT (*New Infra-Red Technology*) monocular, num ambiente de fraca luminosidade, onde os objetos a serem detetados se encontram distantes da câmara. O método consiste em, selecionar numa primeira fase regiões de interesse, através da extração de regiões brilhantes da imagem, e a informação de forma é posteriormente utilizada para verificação. Mais especificamente, o seletor de regiões de interesse é implementado através de um *boosted cascade* modificado combinado com restrições dinâmicas de perspetiva (figura 2.40), que elimina objetos não humanos. Por fim as restantes regiões são classificadas através de SVM com uma FDF (*Four Diretional Feature*).

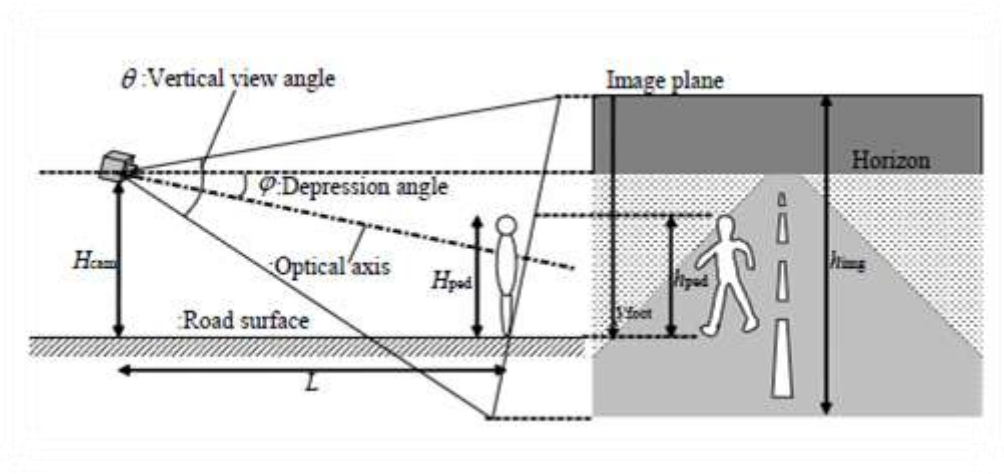


Figura 2.40 – Representação da geometria aplicada aos objetos **[Soga2008]**.

Em [Krotosky2008], é estudado e implementado um método aplicável a sistemas de vigilância multimodais e multi perspectiva. O sistema consiste na incorporação de duas câmaras "normais" e duas de infravermelhos. A estratégia baseia-se na combinação probabilística de um classificador SVM treinado com *features* HOG, extraídas das fontes de informação, com um detetor baseado em disparidade, que se foca na relação entre o tamanho e profundidade dos indivíduos na imagem. A estrutura da abordagem encontra-se ilustrada na figura 2.41.

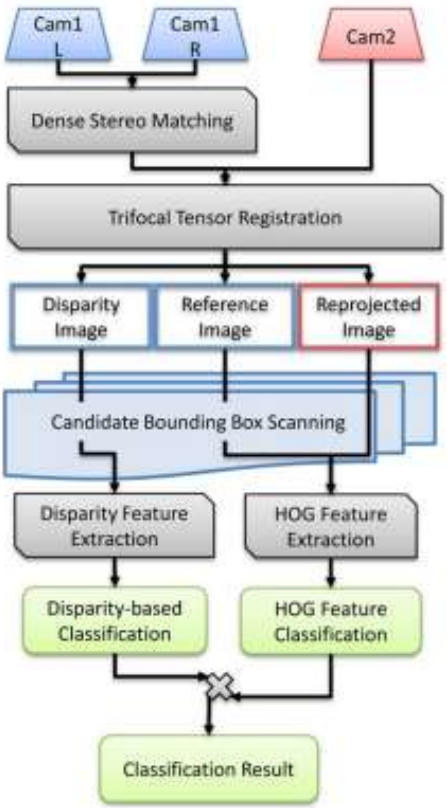


Figura 2.41 – Estrutura da abordagem desenvolvida em **[Krotosky2008]**.

Embora não sejam exploradas imagens térmicas, em [Karaman2009] é atacado o problema que sombras e reflexões provocam na performance dos algoritmos. Foi proposta uma etapa de segmentação, com o objetivo de detetar a região fronteiriça entre o individuo e a sua sombra, eliminando assim as sombras existentes nas regiões segmentadas. O processo consiste em aplicar um método iterativo de espelhamento de regiões da imagem, onde é medida a correlação entre a região espelhada e o seu complemento na imagem original. A janela que tiver maior correlação corresponde á janela onde a linha de fronteira está localizada, devido à simetria existente entre os indivíduos e respetivas sombras. A abordagem desenvolvida encontra-se ilustrada na figura 2.42.

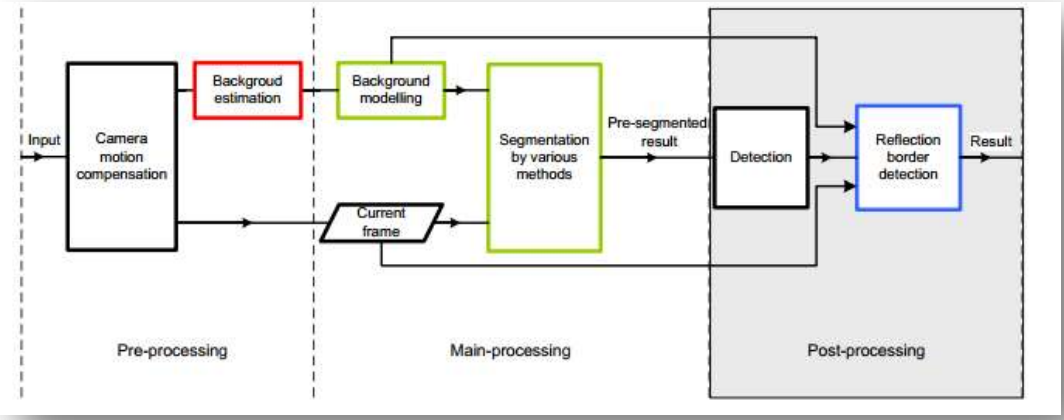


Figura 2.42 – Visão geral da abordagem desenvolvida em **[Karaman2009]**.

Em [Fernandez2010] é desenvolvido um sistema de deteção aplicável a camaras térmicas. O sistema num robô utilizado em tarefas de vigilância. A estratégia consiste em alternar de uma forma inteligente, entre subtração de imagens e *optical flow*, dependendo se a plataforma está ou não em movimento. A abordagem começa com uma fase de análise estática para a deteção de possíveis candidatos. Esta análise consiste inicialmente numa etapa de uniformização das imagens capturadas em tons de cinza, seguido de uma fase de eliminação de zonas incandescentes e de restrição das zonas restantes através de valores de área mínimos que estas devem cumprir. De seguida é aplicada uma análise dinâmica através de *optical flow* ou subtração de imagens, dependendo do estado da plataforma.

Foi desenvolvido em [Zhu2011], um estudo comparativo acerca da possibilidade de aplicação de vários modelos de descrição em imagens térmicas (HOG, SIFT e SURF), e o impacto que produzem num algoritmo de seguimento. Foram desenvolvidos e testados alguns métodos de segmentação de imagem com características distintas, com o objetivo de concluir qual o que revela melhores resultados, quando aplicados a este tipo de imagens. Os resultados revelaram que os métodos baseados em histogramas de gradientes orientados são os que se comportam melhor neste tipo de imagens.

2.4 - Conclusão

As abordagens existentes na literatura são bastante diversificadas, embora aquelas que revelam melhores resultados, possuem vários elementos em comum. De uma forma geral, os detetores adotam sistemas baseados em métodos de janela deslizante multi-escala, que englobam a extração de *features* e classificação binária.

Com o objetivo de diminuir a área de processamento das imagens e consequentemente o tempo de processamento dos algoritmos, são frequentemente aplicadas etapas de subtração de fundo, que consistem na ideia de que se existir um modelo de fundo específico para cada cenário, se subtrair a uma imagem de referência o respetivo modelo, restam as áreas de interesse que devem ser processadas, dai ser frequentemente aplicado em sistemas com o objetivo da deteção em tempo real. A aplicação deste tipo de métodos também permite reduzir o número de falsos alarmes pois restringe significativamente a informação a ser processada. No entanto a subtração de fundo também tem as suas limitações, como o facto da necessidade de existir um modelo de fundo associado a cada cenário, o que obriga à criação de um modelo de fundo para cada situação específica. Outro senão está relacionado com o facto de este tipo de métodos apenas poder ser aplicado em câmaras estacionárias, embora existam abordagens que estudem a possibilidade de criar modelos de fundo adaptativos, de forma a ser possível adaptar subtração de fundo em câmaras móveis.

Em termos de *features* utilizadas, a literatura revela uma grande aposta no descritor HOG, ou variantes deste, devido ao sucesso que demonstram comparativamente com as abordagens restantes. Existem outras *features* (estáticas e de movimento), que também proporcionam bons resultados, como as HAAR *wavelets*, LBP ou até matrizes de covariância. As variantes resultantes destas *features* estão frequentemente relacionadas com simplificações destas, diminuindo as suas dimensões, com o objetivo de diminuir o tempo de processamento e complexidade computacional sem custos significativos relativamente ao seu poder discriminativo, ou com melhoramentos no seu poder discriminativo sem aumentos significativas na complexidade computacional e tempo de processamento.

Visto que cada *feature* possui as suas limitações, verifica-se que a integração de várias *features* distintas na mesma plataforma provocam um impacto muito positivo na performance dos algoritmos, à custa, de um aumento da complexidade computacional, sendo assim frequente seguir esta direção.

Para combater este problema, artigos mais recentes, focam-se no melhoramento da eficiência e eficácia dos classificadores utilizados. Os classificadores explorados com maior regularidade são SVM e o Adaboost, devido ao sucesso que revelam neste tipo de tarefas. Recentemente têm aparecido propostas de variantes destes, como o HIKSVM ou o *MPLBoost*,

que demonstram resultados promissores, relativamente à eficiência e eficácia da classificação, reduzindo simultaneamente o custo computacional.

Maior partes das abordagens focam-se em analisar informação proveniente de imagens RGB, embora também sejam explorados outros tipos de informação como o estéreo ou os infravermelhos, que são explorados noutro tipo de cenários em que as imagens RGB não podem ser utilizadas, como em cenários sem luminosidade ou onde fatores climatéricos degradam a qualidade da informação. Por outro lado, se o ambiente em questão estiver exposto à luz solar, trabalhar na gama de infravermelhos torna-se complicado, pois as câmaras captam as imagens através da medição da radiação térmica existente. Os sistemas baseados neste tipo de informação utilizam frequentemente informação de intensidade com objetivo de extrair regiões brilhantes, visto que se parte do princípio que as pessoas são os objetos mais brilhantes na imagem pelo facto de emitirem energia. De facto verifica-se que as abordagens baseadas em informação proveniente de imagens térmicas utilizam ferramentas semelhantes às utilizadas em imagens RGB, como a subtração de fundo, segmentação, *features* de forma, como o HOG, visto que os contornos humanos se encontram bem definidos neste tipo de imagens, combinações de diferentes *features* ou restrições de perspetiva a partir de um conhecimento prévio do cenário. Um dos maiores problemas existentes na utilização deste tipo de informação, está relacionado com a variabilidade da capacidade de absorção e transmissão da energia que os elementos constituintes do cenário possuem, que provocam a existência de reflexões nas imagens, que se refletem através de falsos alarmes. Existem vários métodos desenvolvidos com o objetivo de enfrentar o desafio provocado pelas reflexões, maioritariamente baseados no conhecimento prévio do cenário, através de etapas de filtragem baseadas em valores de *threshold*, e definindo restrições espaciais e/ou de perspetiva.

Capítulo 3

Análise experimental de algoritmos de deteção

Neste capítulo serão discutidos aspetos relacionados com a implementação, testes e avaliações de quatro algoritmos distintos para deteção de pessoas. As diferenças entre os algoritmos estão relacionadas com as *features* que são extraídas, bem como os classificadores que as analisam, ou até aspetos relacionados com a arquitetura dos mesmos. Pretende-se estudar o seu comportamento relativamente à deteção e à robustez a vários fatores que influenciam a performance dos algoritmos, tais como, variações de postura dos indivíduos, de luminosidade das imagens, diferentes resoluções e também de posicionamento da câmara. É esperado que através da análise dos resultados gerados por cada um dos algoritmos, seja possível avaliar de uma forma precisa a capacidade de deteção dos algoritmos e identificar as suas limitações relativamente aos fatores acima descritos.

Este capítulo encontra-se dividido em 5 subcapítulos, começando por uma introdução aos dados de teste que foram selecionados para testar os algoritmos.

No ponto 3.2 são apresentados os quatro algoritmos que foram sujeitos às várias análises efetuadas.

O subcapítulo 3.3 consiste na descrição das métricas de avaliação aplicadas aos resultados obtidos e as alterações que lhes foram introduzidas, com o objetivo de obter resultados mais discriminativos, permitindo assim tirar conclusões mais concretas.

No ponto 3.4 encontram-se descritos e analisados os vários testes e os resultados daí provenientes relativamente a cada um dos algoritmos, com o objetivo de responder às questões levantadas acerca da performance e limitações de cada das abordagens selecionadas para teste.

O capítulo termina com o ponto 3.5, que consiste numa discussão geral dos resultados gerados, com o objetivo de tirar conclusões acerca da capacidade de deteção e limitações de cada um dos algoritmos, perante os cenários testados.

3.1 - Dados de Teste

Com o objetivo de serem analisadas várias situações distintas, com características diversas, foram escolhidos dois conjuntos de dados diferentes, algumas sequências do projeto CAVIAR, e um conjunto de sequências capturado nas instalações do INESC Porto. Foi atribuído o nome INESC a este conjunto, de forma a evitar confusões, sempre que forem abordados assuntos relativos a este conjunto de dados ao longo do documento.

Foram usadas seis sequências de vídeo do conjunto de dados do CAVIAR, três com vista de corredor (OSOW2C,OSME1C,TES3C) e outras três com vista frontal (OSOW2F,OSME1F,TES3F), com o objetivo de se observar e analisar, para além de outros aspetos, o impacto da posição da câmara e da postura dos indivíduos, na performance dos algoritmos. A opção de utilizar sequências de vídeo do CAVIAR foi fortemente motivado pelo facto de já existir informação de referência gerada para todas as sequências e também por ser uma ferramenta regularmente utilizada na literatura, facilitando assim a comparação de aspetos relativos à performance dos vários algoritmos.

As imagens existentes no conjunto de dados do CAVIAR têm uma resolução baixa de 384x288, e incluem variadas situações de ocultação entre pessoas e outras provocadas pelos próprios limites da imagem.

A figura 3.1 consiste num conjunto de *frames* ilustrativas de cada uma das vistas do CAVIAR.

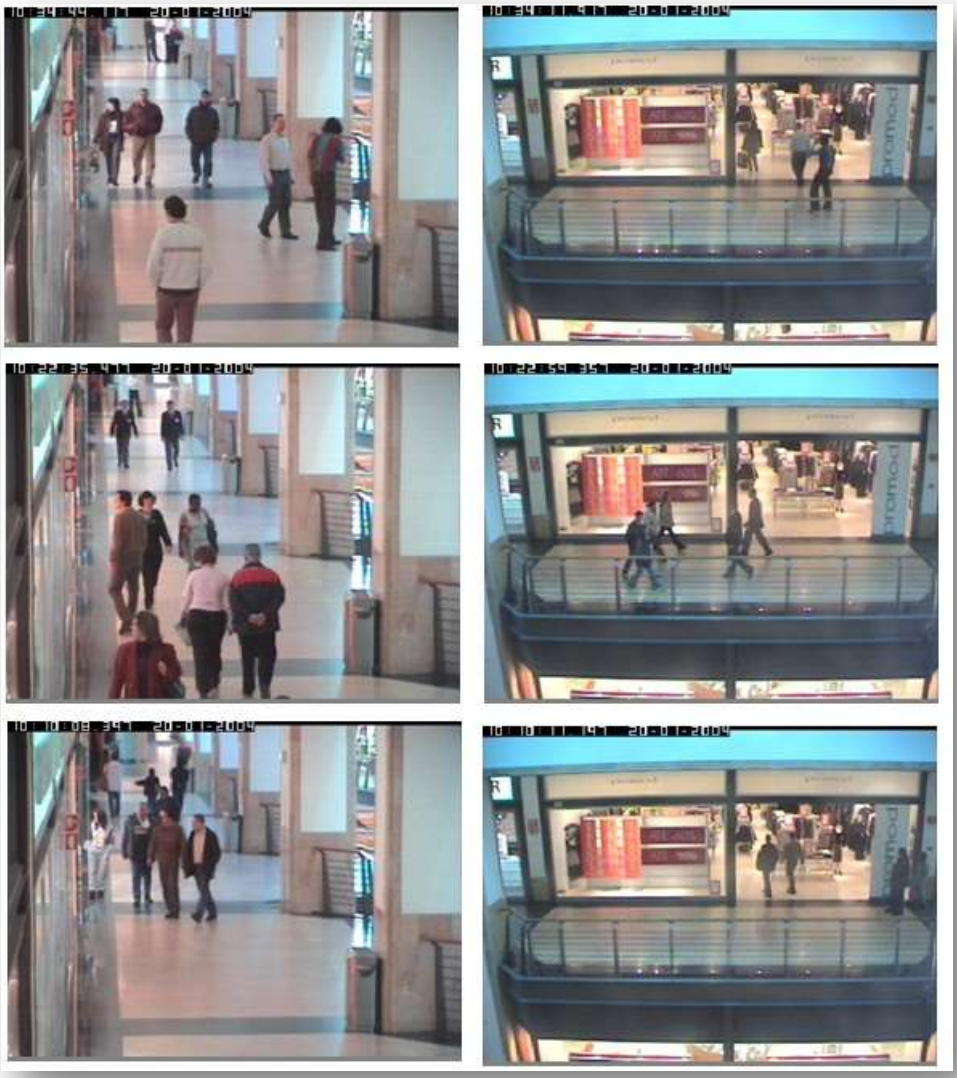


Figura 3.1 – *Frames* exemplo do conjunto de dados do CAVIAR. À esquerda encontra-se imagens relativas às sequências com vista do corredor e à direita com vista frontal.

A tabela 3.1, representa o número de *frames* de cada sequência do conjunto de dados do CAVIAR, utilizadas nos testes.

Tabela 3.1 – Número de *frames* de cada sequência do *dataset* do CAVIAR.

Sequência de vídeo	N° de *frames*
OneShopOneWait2cor (OSOW2C)	1462
OneShopOneWait2front (OSOW2F)	1462
OneStopMoveEnter1cor (OSME1C)	1590
OneStopMoveEnter1front (OSME1F)	1590
TwoEnterShop3cor (TES3C)	1149
TwoEnterShop3front (TES3F)	1075
Total	**8328**

O conjunto de dados do INESC foi introduzido, devido ao facto de as suas sequências possuírem vários tipos de ocultações de indivíduos (integrais ou parciais), tanto por objetos, como paredes, mesas ou pilares, como também devido a outras pessoas existentes nas imagens. Outro factor que contribuiu para a decisão de utilizar este conjunto está relacionado com a possibilidade de analisar a capacidade de adaptação dos vários algoritmos a escalas e resoluções diferentes, visto que neste conjunto de dados, as pessoas se encontram relativamente próximas da câmara, logo o tamanho da região delimitadora do corpo de cada indivíduo é maior do que nas sequências do CAVIAR. Para além disso, as imagens existentes no conjunto do INESC têm uma resolução elevada, 1580×889.

O conjunto é constituído por quatro sequências de vídeo, onde se encontram presentes cinco pessoas, em localizações e posturas variadas. De uma forma geral, nas sequências CAM2-1 e CAM2-2 os indivíduos presentes na imagem encontram-se muito mais próximos da câmara, originando várias situações de ocultação provocadas pelos limites da imagem. Encontram-se ilustradas *frames* exemplo destas sequências na figura 3.2.

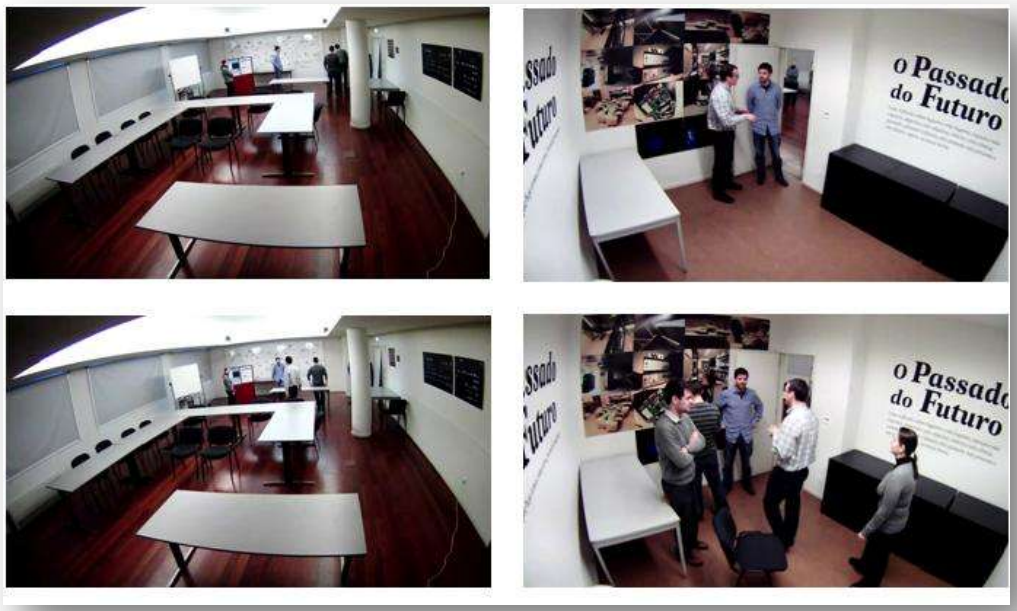


Figura 3.2 – Imagens exemplo do conjunto de dados do INESC em cada um dos cenários (à esquerda CAM1 e à direita CAM2).

A tabela 3.2 representa o número de *frames* de cada sequência do conjunto de dados do INESC.

Tabela 3.2 – Número de *frames* de cada sequência do *dataset* do INESC.

Sequência de Vídeo	Nº de *frames*
CAM1-1	2504
CAM1-2	1392
CAM2-1	1345
CAM2-2	1006
Total	**6247**

Como não existia informação de referência para este conjunto de dados, foi gerada toda a informação necessária manualmente, definindo *bounding boxes* nas regiões da imagem onde se encontram as pessoas e gerando um ficheiro CVML semelhante ao utilizado no CAVIAR, com a informação relativa à localização de cada região da imagem onde se encontra cada indivíduo.

Os ficheiros CVML possuem uma estrutura específica e muito intuitiva. Cada ficheiro CVML está associado a uma sequência de vídeo, onde a informação acerca da localização de cada objeto se encontra agrupada por *frame*. A informação de cada objeto é constituída por quatro valores distintos:

- A coordenada x do ponto central da região;
- A coordenada y do ponto central da região;
- Comprimento da região;
- Largura da região;

De forma a compreender melhor a estrutura destes ficheiros, na figura 3.3 encontra-se ilustrado um excerto de um ficheiro CVML.

```
<?xml version="1.0" encoding="UTF-8"?>
<dataset name="TwoEnterShop3cor">
      <frame number="0">
            <objectlist>
                  <object id="0">
                        <orientation>90</orientation>
                        <box xc="208" yc="103" w="72" h="120">
                  </object>
            </objectlist>
      </frame>
      <frame number="1">
            <objectlist>
                  <object id="0">
                        <orientation>90</orientation>
                        <box xc="100" yc="120" w="72" h="120">
                  </object>
                  <object id="0">
                        <orientation>90</orientation>
                        <box xc="214" yc="109" w="72" h="120">
                  </object>
            </objectlist>
      </frame>
</dataset>
```

Figura 3.3 - Exemplo de um excerto de um ficheiro CVML.

3.2 - Algoritmos

Os algoritmos utlizados encontram-se descritos abaixo e são os seguintes:

- Uma implementação do HOG existente nas bibliotecas do *OpenCV*, baseada no artigo [Dalal2005];
- Um detetor que utiliza Haar *wavelets* como objeto de um classificador *Adaboost*, existente nas bibliotecas do *OpenCV*, baseado em [Viola2003];
- Uma integração do HOG com o LBP [Wang2009];
- O algoritmo desenvolvido em [Odobez2008], que utiliza como características, matrizes de covariância, como objeto de uma cascata de classificadores *LogitBoost*.

3.2.1 - HOG

O algoritmo HOG utilizado é uma implementação existente nas bibliotecas do *OpenCV*, baseado no artigo [Dalal2005]. Utiliza como *features* histogramas de gradientes orientados (HOG) e como classificador, SVM linear (*Support Vector Machine*).

O descritor HOG consiste na construção de histogramas de gradientes orientados, que contam a ocorrência das orientações dos gradientes em regiões específicas da imagem. O HOG é calculado numa densa rede de células uniformemente espaçadas e utiliza sobreposição de contrastes locais normalizados com o intuito de melhorar a precisão da descrição. O descritor consiste na avaliação de histogramas locais normalizados das orientações dos gradientes da imagem construídos numa rede densa de células. *Dalal et al* basearam-se no facto de que a aparência e forma dos objetos podem ser bem caracterizadas através de distribuições das intensidades dos gradientes locais. O método consiste então na divisão da janela da imagem em células, sendo para cada uma delas acumulado um histograma das direções dos gradientes, relativos aos pixéis da célula correspondente. De forma a tornar o descritor mais robusto a variações de luminosidade ou sombras, foi inserida uma etapa de normalização de contraste das respostas locais antes de serem utilizadas. Esta etapa consiste na acumulação de uma medida baseada na energia dos histogramas locais sobre uma região maior, denominada de bloco, usando esses resultados para normalizar todas as células contidas em cada bloco. Para finalizar o método, é aplicado um classificador SVM linear aos descritores HOG construídos.

A figura 3.4 ilustra a abordagem acima descrita.

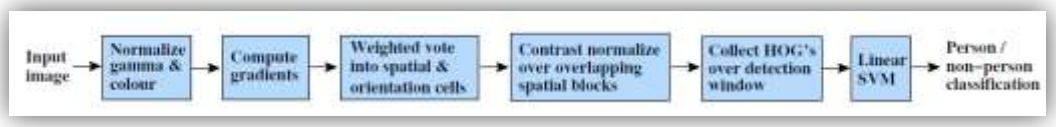


Figura 3.4 - Representação global da arquitetura do algoritmo HOG **[Dalal2005]**.

3.2.2 - IDIAP

O algoritmo IDIAP, [Odobez2008], é baseado em [Tuzel2007], procurando diminuir o seu tempo de processamento e inserindo informação temporal na tarefa de deteção, através da integração de um método de subtração de fundo. Utiliza como descritor para o classificador, matrizes de covariância entre características, nas várias regiões da imagem. O classificador consiste numa cascata de "*Weak classifiers*" *LogitBoost*.

As regiões de interesse (regiões onde possivelmente se encontra um indivíduo) são representadas pela matriz de covariâncias das características da imagem, como a localização espacial, intensidade, amplitude das derivadas de alta ordem e orientação de contornos. Uma pessoa é então caracterizada por várias matrizes de covariâncias resultantes da sobreposição

das regiões. Como já referido no capítulo 2, técnicas clássicas de aprendizagem não são adequadas para treinar este tipo de descritores, pois estas matrizes não se baseiam em espaços vetoriais, daí as matrizes de covariância serem formuladas a partir de um espaço Riemmaniano conectado. A grande contribuição deste artigo baseia-se numa nova abordagem para classificar pontos num espaço Riemmaniano, através da incorporação de informação à priori sobre a geometria do espaço. Como o processo de mapeamento das covariâncias para cada classificador é lento para *features* de grandes dimensões espaciais, optaram por construir classificadores baseados em subconjuntos das *features* totais da imagem, o que significa dividir a matriz de covariâncias completa em submatrizes, permitindo assim explorar a correlação entre pequenos grupos de *features* para cada classificador. À medida que o número destes pequenos conjuntos aumenta, foi implementado um método de seleção, responsável por escolher o conjunto que resulta na melhor performance na fase de treino do *LogitBoost*. Estas submatrizes são então combinadas com as médias das *features* das imagens na mesma janela, o que provoca uma rejeição mais rápida das regiões. Esta abordagem proporciona resultados relativos à deteção muito positivos, daí ser regularmente analisada na literatura.

3.2.3 - MUD

O algoritmo MUD [Wang2009] combina dois tipos de características, o HOG (*Histograms of Oriented Gradients*) e LBP (*Local Binary Patterns*) em conjunto com dois tipos de detetores que são treinados usando SVM linear, sendo um deles um detetor global aplicado nas janelas de decisão e um detetor de partes responsável por regiões locais. Foi estudada a combinação do HOG e LBP com o principal objetivo de atingir bons resultados em situações de ocultação parcial de indivíduos. Para cada janela de decisão são construídos mapas de probabilidade de ocultações através da resposta de cada bloco HOG. Cada mapa é então segmentado utilizando uma abordagem *Mean-Shift*. Por sua vez a região segmentada da janela com o maior número de respostas negativas é definido como uma região onde existe ocultação. De seguida são aplicados detetores de partes às regiões da janela onde não existe ocultação, com o objetivo de detetar a presença de um indivíduo. Esta abordagem revela uma performance robusta, no que toca à deteção em ambientes em que ocorram várias ocultações, daí se tornar objeto de melhoramento em vários artigos.

A abordagem implementada por *Wang et al* encontra-se ilustrada na figura 3.5.

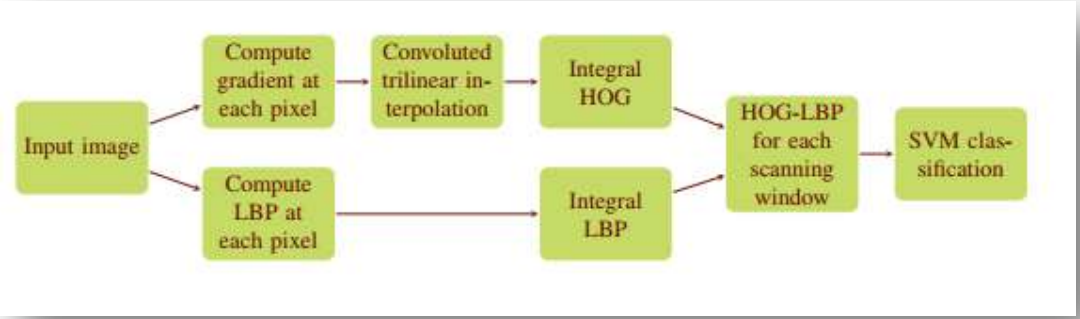


Figura 3.5 - Representação global da arquitetura do algoritmo MUD **[Wang2009]**.

3.2.4 - HAAR

O algoritmo HAAR é uma implementação existente nas bibliotecas do *OpenCV*, baseado no artigo [Viola2001]. Utiliza Haar *wavelets* como objeto de uma cascata de classificadores *Adaboost*. O algoritmo utiliza imagens integrais, como forma de representação das imagens, o que possibilita uma avaliação das *features* mais rápida em várias escalas. De forma a selecionar um conjunto mais pequeno de *features* (as mais relevantes) foi aplicado uma pequena modificação ao *Adaboost*, que define que cada *weak classifier* depende apenas de uma única *feature*. Combinar sucessivamente classificadores cada vez mais complexos numa estrutura em cascata, aumenta drasticamente a velocidade do algoritmo, obrigando-o a focar-se nas regiões de maior interesse.

3.3 - Métricas de Avaliação

Como já foi referido anteriormente, a informação de referência está relacionada com as características das caixas delimitadoras da região onde se encontra um indivíduo. O mesmo acontece com a informação presente nos *outputs* dos algoritmos de deteção. São então aplicadas várias estratégias de avaliação com base nesta informação.

Nesta plataforma são utilizadas para avaliação dos algoritmos, uma versão modificada das métricas *frame based* [Bashir2006], e as métricas *Partition Distance* [Cardoso2009].

3.3.1 - Métricas *Frame Based*

Esta métrica consiste na verificação da posição do centróide de cada região detetada, para verificar se este se encontra no interior de alguma região presente no ficheiro de referência, como se pode verificar na figura 3.6.

Caso o centróide se encontre no interior da região, assume-se que se encontrou um indivíduo.

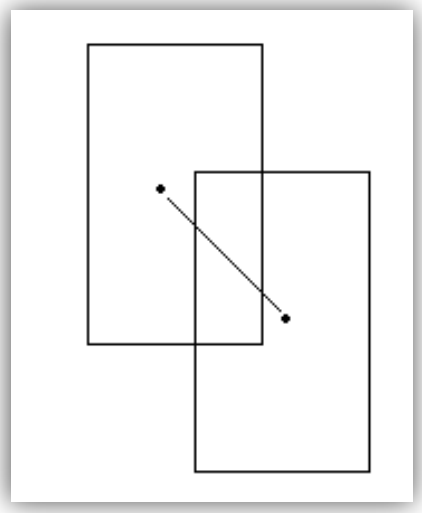


Figura 3.6 – Posição e distância entre os centróide de duas regiões (métricas *frame-based*).

As métricas *frame-based* medem a performance dos algoritmos, individualmente, em cada *frame* de cada sequência de vídeo, sem ter em consideração a conservação de identidade dos indivíduos ao longo da sequência.

Cada *frame* é então testada individualmente, comparando-se a informação presente nos ficheiros de *output* dos algoritmos com a existente nos de referência.

3.3.2 - Versão Original

Originalmente em [Bashir2006], através da análise da posição do centróide de todas as regiões detetadas, são calculadas diversas quantidades descritas abaixo, que serão utilizadas na construção de estatísticas globais, relativas à performance dos algoritmos.

As quantidades calculadas são as seguintes:

- ***True Negative* (Verdadeiro Negativo), TN**: Número de *frames* onde tanto o *output* do algoritmo como o ficheiro de referência, consideram que não existe nenhuma pessoa presente.

- ***True Positive* (Verdadeiro Positivo), TP**: Número de *frames* onde o *output* do algoritmo considera que a região de pelo menos um dos objetos possui o seu centróide no interior de alguma das regiões presentes na informação de referência.

- ***False Negative* (Falso Negativo), FN**: Número de *frames* onde no ficheiro de referência existe pelo menos um objeto, e no *ouput* do algoritmo ou não existe nenhum ou então os que existem, o seu centróide não se encontra dentro de nenhuma região presente no ficheiro de referência.

- ***False Positive* (Falso Positivo), FP**: Número de *frames*, onde no *output* do algoritmo existe pelo menos um objeto, quando no ficheiro de referência não existe nenhum, ou então os que existem não incluem nenhum centróide no seu interior.

- **Total *Ground truth*, TG**: Número total de *frames* do ficheiro de referência.

- **Total *Frames*, TF**: Número total de *frames* da sequência.

Depois de calculadas estas quantidades, algumas das estatísticas propostas são:

$$False\ Alarm\ Rate\ (FAR) = \frac{FP}{TP + FP} \tag{3.1}$$

$$Detection\ Rate\ (DR) = \frac{TP}{TP + FN} \tag{3.2}$$

$$False\ Negative\ Rate\ (FNR) = \frac{FN}{FN + TP} \tag{3.3}$$

3.3.2.1 - Limitações

Ao analisar as métricas desenvolvidas em [Bashir2006], conclui-se que existem algumas limitações no poder discriminativo deste tipo de métricas. Embora seja possível tirar algumas conclusões através da aplicação destas métricas tal como foram definidas pelo autor, estas refletem de uma forma pouco exigente a performance do algoritmo, pois as estatísticas que daí resultam são baseadas nas quantidades acima referidas. Estas quantidades refletem o número total de *frames* de uma sequência (e não o número total de objetos) que respeitam as restrições impostas por elas. Conclui-se então que uma *frame* ou é classificada como TP,TN,FP ou FN. Este tipo de classificação não tem um poder discriminativo muito pormenorizado, visto que estas quantidades não refletem o número de objetos. De forma a compreender melhor estas limitações, tomemos como exemplo a quantidade TP (*True Positive*): Originalmente, TP consiste no número de *frames* da sequência onde, o *output* do algoritmo considera que a região de pelo menos um dos objetos possui o seu centróide no interior de alguma das regiões presentes na informação de referência; basta que uma das

regiões em cada *frame*, possua o centróide no interior de qualquer região existente no ficheiro de referência, para que o algoritmo revele uma taxa de deteção de 100%, mesmo que existam outros objetos por detetar, como se pode verificar através da análise da tabela 3.3, que representa os resultados destas métricas aplicadas a uma *frame* exemplo (figura 3.7).

Tabela 3.3 – Resultados das métricas *frame-based* apicadas a uma *frame* exemplo.

	***Frame* exemplo**
False Alarm Rate (FAR) (%)	0
Detection Rate (DR) (%)	100
False Negative Rate (FNR) (%)	0

Figura 3.7 – *Output* de uma *frame* exemplo relativa à tabela 3.3.

Motivado por estas limitações, decidiu-se proceder a um conjunto de alterações de forma a contar e classificar os objetos (pessoas), e não as *frames*, obtendo-se assim uma análise mais fiável e detalhada dos resultados.

3.3.3 - Versão Modificada

Foram implementadas alterações relativas às estatísticas globais que resultam destas métricas, inspirado pelo facto de que estas não descrevem de uma forma muito exigente a performance do algoritmo. Estas estatísticas são construídas a partir das diversas quantidades já referidas anteriormente, e foi precisamente no cálculo destas quantidades que as alterações foram implementadas, tornando-as orientadas ao objeto, quando originalmente são orientadas à *frame*, ou seja, são contados os objetos e não as *frames* que cumprem com

os requisitos de cada quantidade. A figura 3.8 descreve de uma forma geral, a representação da abordagem.

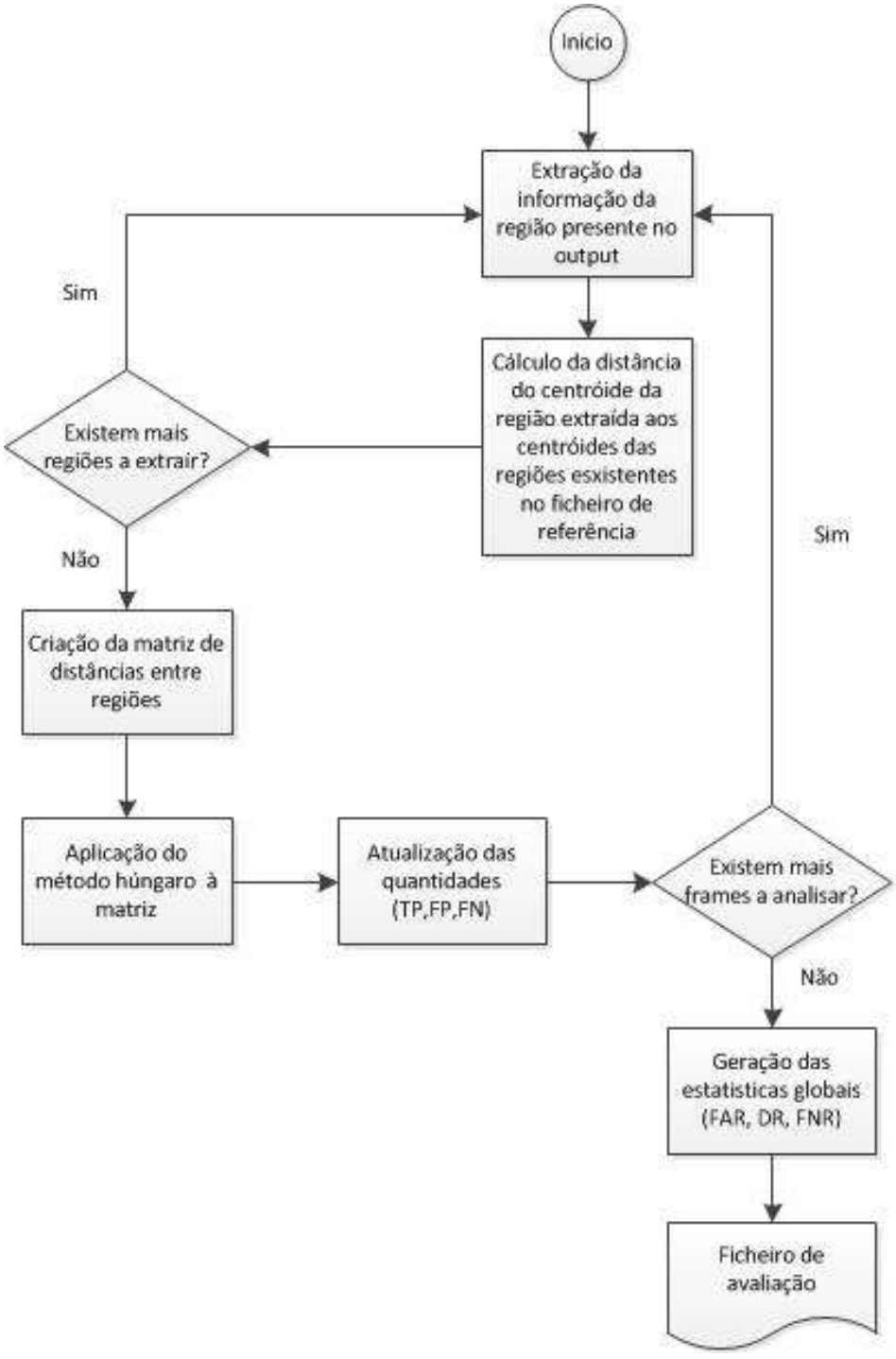


Figura 3.8 – Esquema ilustrativo da abordagem implementada.

Foi então proposto que as quantidades revelem não o número de *frames*, mas sim o número de objetos que cumprem com os requisitos de cada quantidade, como já referido anteriormente. Desta forma, as quantidades foram redefinidas como:

- ***True Negative*, TN**: Número de *frames* onde tanto o *output* do algoritmo como o ficheiro de referência, consideram que não existe nenhuma pessoa presente.
- ***True Positive*, TP**: Número de objetos presentes no *output* do algoritmo, cujo centróide se encontra no interior de alguma região presente no ficheiro de referência
- ***False Negative*, FN**: Número de objetos existentes no ficheiro de referência, que no *output* do algoritmo não existem ou então os que existem, o seu centróide não se encontra dentro de nenhuma região presente no ficheiro de referência.
- ***False Positive*, FP**: Números de objetos existentes no *output* do algoritmo, quando no ficheiro de referência não existe nenhum, ou então os que existem não incluem nenhum centróide no seu interior.
- **Total *Ground truth*, TG**: Número total de objetos presentes no ficheiro de referência.
- **Total *Frames*, TF**: Número total de *frames* da sequência.

Para tal, foi aplicado às várias regiões de cada *frame* o método húngaro, que tem como objetivo resolver o problema de afetação de cada região existente no output do algoritmo, à região correspondente no ficheiro de referência, tornando possível o *matching* das regiões. A escolha da utilização do método húngaro foi motivada pela sua adequação a este tipo de desafio, pois o matching das regiões pode ser formulado como um problema de otimização combinatória. O método é aplicado *frame* a *frame*, onde para cada região existente no *output* do algoritmo, cujo centróide se encontre no interior de uma região presente no ficheiro de referência, são calculadas as distâncias entre eles, sendo criada uma matriz de distâncias. O método húngaro é então aplicado à matriz, com o objetivo de associar cada região presente no *output* à região correspondente do ficheiro de referência, partindo do princípio que para cada região do ficheiro de referência, pode existir apenas uma região associada. Para uma melhor compreensão deste método considere-se uma *frame* exemplo (figura 3.9):

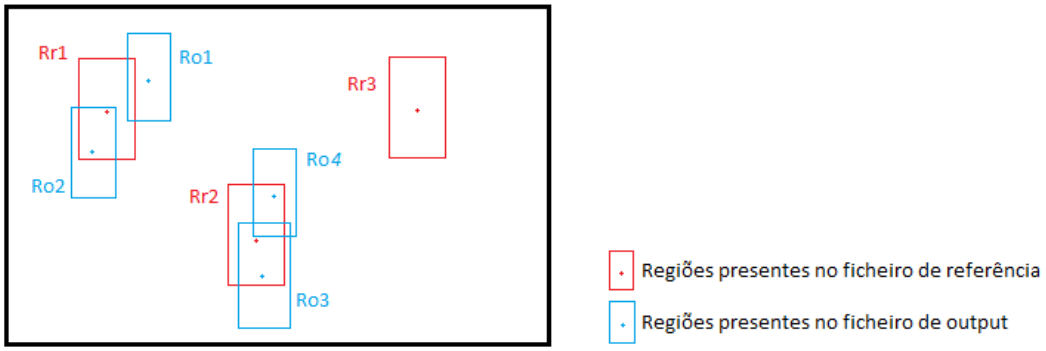


Figura 3.9 – *Frame* exemplo.

Como se pode verificar, na figura 3.9, existem três regiões presentes no ficheiro de referência (Rr1, Rr2 e Rr3) e quatro no ficheiro de *output* (Ro1, Ro2, Ro3 e Ro4). A matriz de distâncias associada a esta *frame* exemplo encontra-se descrita na tabela 3.4:

Tabela 3.4 – Exemplo da matriz de distâncias relativa à *frame* exemplo presente na figura 3.9.

	Rr1	Rr2	Rr3
Ro1	-	-	-
Ro2	3	-	-
Ro3	-	2	-
Ro4	-	5	-

Ao aplicar o método húngaro à matriz, obtém-se os seguintes resultados descritos na tabela 3.5:

Tabela 3.5 – Representação da resolução do problema de afetação de regiões através do método húngaro.

Rr1	**3 (Ro2)**
Rr2	**2 (Ro3)**
Rr3	**-**

Figura 3.10 – Grafo representativo do problema de afetação de regiões.

Após a associação das regiões são então atualizadas as quantidades TP,FP e FN da seguinte forma:

Se uma região presente no ficheiro de *output* tiver sido associada a uma região existente no ficheiro de referência, considera-se que foi encontrado um verdadeiro positivo (TP), caso contrário considera-se que foi encontrado um falso positivo (FP). Por outro lado, cada região existente no ficheiro de referência que não tenha sido associada a nenhuma região é considerada como sendo um falso negativo (FN). Assume-se então que Ro2 e Ro3 são TP, Ro1 e Ro4 são FP e Rr3 é FN.

Depois de calculadas estas quantidades, o método repete-se para todas as *frames* da sequência e no final as estatísticas seguintes são geradas:

$$False\ Alarm\ Rate\ (FAR) = \frac{FP}{TP + FP} \tag{3.4}$$

$$Detection\ Rate\ (DR) = \frac{TP}{TP + FN} \tag{3.5}$$

$$False\ Negative\ Rate\ (FNR) = \frac{FN}{FN + TP} \tag{3.6}$$

Estas estatísticas refletem agora a percentagem de objetos em cada sequência, que não foram detetados (FNR), que foram detetados corretamente (DR) ou incorretamente (FAR), obtendo-se assim uma métrica mais discriminativa da performance do algoritmo.

Na tabela 3.6 estão descritas as diferenças entre as estatísticas calculadas através da versão original e a modificada, relativas a uma figura exemplo (figura 3.11), com o intuito de se verificar de uma forma mais pormenorizada, o impacto das alterações efetuadas às métricas aplicadas. Na figura 3.11 estão presentes oito indivíduos, e cinco regiões detetadas.

Figura 3.11 – *Frame* Exemplo do conjunto de dados do CAVIAR.

Tabela 3.6 - Resultados das métricas *frame-based* relativamente à *frame* presente na figura 3.11, com e sem as alterações efetuadas.

	Versão Original	Versão modificada
False Alarm Rate (FAR) (%)	0	20
Detection Rate (DR) (%)	100	50
False Negative Rate (FNR) (%)	0	50

Como era esperado, ao aplicar a versão modificada, a taxa de deteção diminui significativamente, enquanto a taxa de falsos alarmes e falsos negativos sofrem um aumento relevante. Esta discrepância de valores vem comprovar que realmente com a introdução das alterações efetuadas, o poder discriminativo destas métricas é maior. Conclui-se então, como estava previsto, que a versão original não toma em consideração todos os objetos presentes em cada imagem, e que as alterações efetuadas proporcionam os resultados esperados, ultrapassando com sucesso as limitações identificadas, gerando assim estatísticas que revelam resultados mais próximos da realidade, o que permitirá tirar conclusões mais fiáveis e pormenorizadas acerca da performance dos algoritmos analisados.

3.3.4 - *Partition distance metrics*

Para além das métricas *Frame Based*, também são utilizadas as métricas *Partition Distance*, mais especificamente as *Symmetric partition distance,* definidas em [Cardoso2009] com o objetivo de se poder tirar outro tipo de conclusões visto que este tipo de métricas permite-nos ter uma visão mais concreta do contributo de cada *frame* e objeto para o resultado da performance do algoritmo, em termos de precisão relativamente à área das regiões detetadas. Através desta ferramenta, torna-se possível analisar a semelhança entre as regiões originais e as detetadas, permitindo assim compreender a performance do algoritmo ao longo da sequência, ao contrário das métricas *frame based*, que se limitam caracterizar a performance de uma forma global.

Tendo duas partições A e B de um espaço S, a medida *Symemtric partition-distance*, consiste no número mínimo de elementos que têm de ser eliminados de S, para que apenas reste os elementos comuns às duas partições (figura 3.12).

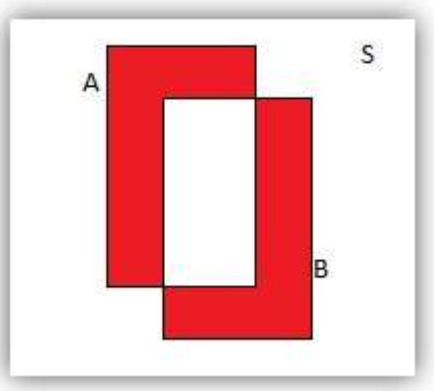


Figura 3.12- Exemplo da segmentação de partições através das métricas *Partition Distance*. Cada partição está associada a uma região.

Partições são segmentações correspondentes a diferentes pessoas.

O processo de segmentação reparte a imagem em diferentes regiões com características homogéneas, através de descontinuidades ou similaridades entre os componentes da imagem. A segmentação espacial de uma imagem corresponde à partição em várias regiões constituintes da imagem, partindo do princípio de todas as regiões estão desmembradas e todos os píxeis pertencem a uma região.

O problema de avaliação de segmentações consiste em encontrar uma descrição precisa da relação entre duas partições através de desigualdades.

Enquanto através das métricas *frame based* conseguimos concluir se o centróide de uma dada região se encontra dentro do perímetro de outra, com esta métrica, conseguimos medir de uma forma mais precisa a semelhança entre as regiões, através do erro existente entre elas e a contribuição de cada *frame* para o resultado final.

Ao utilizar as duas métricas em simultâneo, consegue-se obter informação mais precisa e discriminativa acerca da performance na tarefa de deteção dos vários algoritmos.

3.4 - Teste e Discussão de resultados

Nesta secção encontram-se descritos e analisados os resultados obtidos nos testes e avaliações aplicados aos vários algoritmos, com o objetivo de avaliar a sua capacidade de deteção e limitações relativas às várias situações analisadas, como o impacto que situações de ocultação, variações de postura, iluminação e resolução, provocam na performance dos algoritmos.

Foi construída uma plataforma de testes, cuja estrutura se encontra descrita na figura 3.13. É constituída por um *script* que invoca os algoritmos a serem testados, que recebem como *inputs* os vídeos/imagens a serem analisados. Os algoritmos geram um ficheiro CVML (*Computer Vision Markup Language*), [List2004], com os resultados de deteção para cada uma das sequências. De seguida, são aplicadas as métricas de avaliação escolhidas aos resultados gerados, que em conjunto com a informação de referência, irão produzir os ficheiros de avaliação, que permitirão a avaliação da performance dos vários algoritmos em cada uma das sequências de vídeo utilizadas. Como as métricas *Partition Distance* recebem como *input*, máscaras de segmentação dos resultados da deteção, antes de serem aplicadas, é utilizado um programa que lê os *outputs* dos algoritmos e gera automaticamente as máscaras de segmentação respetivas.

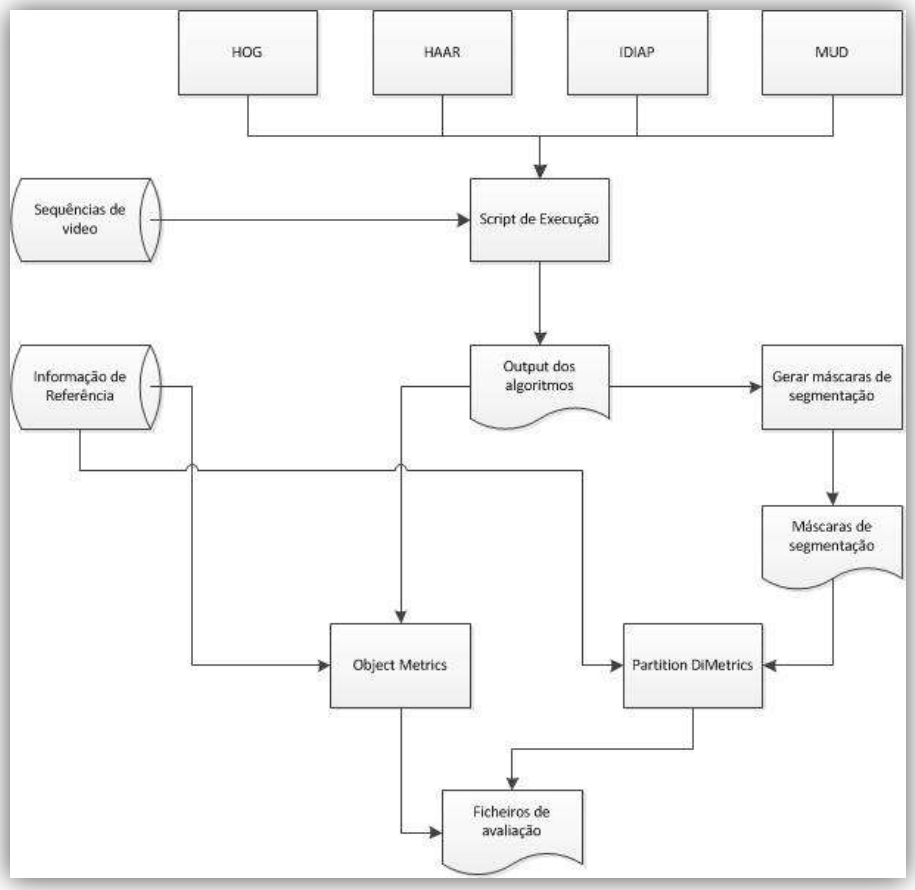


Figura 3.13 – Esquema da plataforma de testes implementada.

3.4.1 - Sumário das avaliações e testes efetuados no Dataset do CAVIAR

Tabela 3.7 – Resultados em percentagem das métricas *frame-based* relativas a cada um dos algoritmos, nas sequências do CAVIAR com vista de corredor.

	OSOW2C				OSME1C				TES3C			
	HOG	**IDIAP**	**HAAR**	**MUD**	**HOG**	**IDIAP**	**HAAR**	**MUD**	**HOG**	**IDIAP**	**HAAR**	**MUD**
False Alarm Rate (FAR) (%)	10,68	18,53	8,95	**0,46**	6,36	26,65	5,76	**0,47**	19,55	20	17,74	**0,79**
Detection Rate (DR) (%)	38,12	48,21	38,31	**60,52**	28,53	43,18	31,79	**50,55**	15,24	22,29	15,49	**23,70**
False Negative Rate (FNR) (%)	61,88	51,79	61,69	**39,48**	71,47	56,81	68,21	**49,45**	84,76	77,71	84,51	**76,30**

- **Vista corredor:**

Através da análise da tabela 3.7, conclui-se, no que diz respeito às sequências do CAVIAR com vista do corredor, que o algoritmo que produz taxas de deteção mais elevadas é o MUD com uma taxa de deteção média de 44,9%, seguido do IDIAP com 37,89%, e os restantes dois métodos (HOG e HAAR) apresentam os resultados mais fracos, 27,3% e 28,5% respetivamente.

Relativamente às taxas de falso alarme, todos os algoritmos possuem valores relativamente baixos, exceto o IDIAP que apresenta uma taxa média de 21,72%. O MUD revela os melhores resultados, 0,5%. O HOG e o HAAR apresentam taxas 12,2% e 10,8% respetivamente.

Consequentemente, no que toca a falsos negativos, o MUD destaca-se obviamente mais uma vez dos restantes, apresentando uma taxa de 55%, por sua vez o IDIAP revela uma taxa de 62,10%, o HAAR 71,5% e o HOG 72,7%.

Após uma análise cuidada destes resultados, conclui-se, como era esperado, que o algoritmo que revela globalmente a melhor performance no conjunto de dados do CAVIAR (vista corredor), é o MUD, visto que possui simultaneamente as taxas de deteção mais elevadas e as taxas de falso alarme e falsos negativos mais baixas, revelando assim ser o método com maior precisão e capacidade de deteção. Os níveis de performance deste algoritmo vão de encontro às expectativas, devido principalmente ao facto de integrar dois tipos de *features* diferentes, o HOG e o LBP, que se complementam muito bem, como já foi explicado no capítulo 2. Para além deste aspeto, o facto de possuir um módulo responsável pela análise de situações de ocultação, também contribui para o sucesso do algoritmo.

O IDIAP revela resultados um pouco inferiores no que diz respeito às taxas de deteção e consequentemente valores superiores no que toca a falsos negativos, embora para atingir tal objetivo, revele valores de falsos positivos, na casa dos 21%, mostrando não ter tanta precisão (tabela 3.7).

O HAAR e o HOG apresentam os resultados menos promissores, demonstrando ter uma performance geral semelhante, embora o HAAR revele valores ligeiramente melhores, pois revela simultaneamente taxas de deteção ligeiramente superiores e taxas de falsos alarmes ligeiramente inferiores (tabela 3.7). Ambos os métodos produzem taxas de deteção baixas e consequentemente taxas de falsos negativos elevadas, no entanto revelam poucos falsos alarmes, revelando assim níveis de precisão aceitáveis.

Os principais motivos que limitam a performance dos algoritmos estão relacionados com, as dimensões reduzidas das imagens e o número elevado de situações de ocultação existentes.

Tabela 3.8 - Resultados em percentagem das métricas *frame-based* relativas a cada um dos algoritmos, nas sequências do CAVIAR com vista frontal.

	OSOW2F				**OSME1F**				**TES3F**			
	HOG	**IDIAP**	**HAAR**	**MUD**	**HOG**	**IDIAP**	**HAAR**	**MUD**	**HOG**	**IDIAP**	**HAAR**	**MUD**
False Alarm Rate (FAR) (%)	80,32	**2,15**	6,22	4,10	92,19	8,55	10,45	**9,03**	94,51	18,94	5,09	**4,51**
Detection Rate (DR) (%)	12,78	7,61	27,10	**29,63**	3,36	7,28	9,52	**12,34**	2,31	6,65	**28,34**	11,50
False Negative Rate (FNR) (%)	87,22	92,39	72,90	**70,37**	96,64	92,72	90,48	**87,66**	97,69	93,35	**71,66**	88,50

- **Vista frontal:**

Através da análise da tabela 3.8, conclui-se, no que diz respeito às sequências com vista frontal, que os algoritmos que produzem taxas de deteção média mais elevadas são o HAAR e o MUD, que rondam os 21,7% e 17,8% respetivamente, e os restantes dois métodos (IDIAP e HOG) apresentam os resultados menos promissores, 7,18% e 6,1% respetivamente.

Relativamente às taxas de falso alarme, o MUD revela mais uma vez os melhores resultados, com uma taxa de 5,9%, seguido do HAAR com 7,3%.

O IDIAP apresenta uma taxa de 9,88%. O HOG demonstra ter os resultados mais fracos, 89%.

Consequentemente, no que diz respeito a falsos negativos, o HAAR apresenta uma taxa média de 78,3%., por sua vez o MUD revela uma taxa de 82,1%, o IDIAP 92,82% e o HOG 93,8%.

Após uma análise cuidada destes resultados, conclui-se que a performance dos algoritmos no conjunto de dados do CAVIAR (vista frontal), reduz substancialmente de uma forma global, embora o HAAR e o MUD revelem os resultados menos negativos (tabela 3.8), visto que possuem simultaneamente as taxas de deteção mais elevadas e taxas de falsos alarmes e falsos negativos mais baixas, revelando assim serem os métodos com maior precisão e capacidade de deteção, neste conjunto de dados.

O IDIAP e o HOG, apresentam resultados inferiores aos restantes, relativamente à sua performance nestes cenários, revelando uma fraca capacidade de deteção e um taxa de falsos alarmes significativamente elevada para o número de deteções corretas que produz, essencialmente no caso do HOG, em que os valores atingem o patamar dos 90% (tabela 3.8), revelando pouca precisão neste tipo de cenários, devido essencialmente à postura e orientação dos indivíduos nas imagens.

Conclui-se então, que no conjunto de dados do CAVIAR, vista frontal, nenhum dos algoritmos apresenta resultados promissores, o que significa que o impacto da vista é significativo. O facto de os indivíduos aparecerem com uma postura e orientação diferente na imagem, visto que aparecem de lado e não de frente também prejudica o sucesso dos algoritmos. No entanto existem outros fatores que também contribuem para o insucesso dos algoritmos, como é o caso da resolução das imagens, pois neste conjunto de dados, as imagens possuem uma resolução de 384 X 288 e a câmara encontra-se bastante afastada, o que origina regiões de interesse com tamanhos muito reduzidos.

3.4.2 - Sumário das avaliações e testes efetuados no Dataset do INESC

Tabela 3.9 - Resultados em percentagem das métricas *frame-based* relativas a cada um dos algoritmos, nas sequências do INESC (CAM1-1 e CAM1-2).

	CAM1-1				CAM1-2			
	HOG	**IDIAP**	**HAAR**	**MUD**	**HOG**	**IDIAP**	**HAAR**	**MUD**
False Alarm Rate (FAR) (%)	53,24	63,40	**35,54**	60,58	**23,68**	56,41	51,48	50,20
Detection Rate (DR) (%)	**58,96**	**58,90**	23,22	7,22	56,09	**65,62**	30,84	7,30
False Negative Rate (FNR) (%)	**41,04**	**41,10**	76,78	92,78	43,91	**34,38**	69,16	92,70

Tabela 3.10 - Resultados em percentagem das métricas *frame-based* relativas a cada um dos algoritmos, nas sequências do INESC (CAM2-1 e CAM2-2).

	CAM2-1				CAM2-2			
	HOG	**IDIAP**	**HAAR**	**MUD**	**HOG**	**IDIAP**	**HAAR**	**MUD**
False Alarm Rate (FAR) (%)	88,08	93,81	83,02	**70,21**	77,99	89,11	65,00	**56,68**
Detection Rate (DR) (%)	35,25	**58,53**	9,01	14,81	55,38	**83,48**	15,91	15,69
False Negative Rate (FNR) (%)	64,75	**41,47**	90,99	85,19	44,62	**16,52**	84,09	84,31

Através da análise das tabelas 3.9 e 3.10, conclui-se, no que diz respeito às sequências do INESC, que o algoritmo que produz taxas de deteção mais elevadas é o IDIAP com uma taxa de deteção média que ronda os 66,6%, seguido do HOG com 51,4%, e os restantes dois métodos (HAAR e MUD) apresentam os resultados mais fracos, 19,7% e 11,3% respetivamente.

Relativamente às taxas de falso alarme, todos os algoritmos revelam valores elevados, principalmente o IDIAP que apresenta uma taxa média de 75,7%. O HAAR revela os melhores resultados, 58,8%. O MUD e o HOG apresentam 59,4% e 60,7% respetivamente. Como as imagens deste conjunto de dados possuem uma resolução elevada (1580 X 889), os algoritmos necessitam de utilizar um número elevado de janelas de pesquisa, e como a informação presente nas imagens tem um nível de complexidade acentuado, torna-se compreensível que as taxas de falsos alarmes sejam superiores às obtidas nas sequências do CAVIAR.

Consequentemente, no que toca a falsos negativos, o IDIAP destaca-se mais uma vez dos restantes, apresentando uma taxa de 33,4%, embora para atingir este valor, revele um

número elevado de falsos positivos, como já referido anteriormente. Por sua vez o HOG revela uma taxa de 48,6%, o HAAR 80,2% e o MUD 88,7%.

Após uma análise cuidada destes resultados, conclui-se que o algoritmo que revela globalmente a melhor performance no conjunto de dados do INESC, é o IDIAP, visto que possui simultaneamente as taxas de deteção mais elevadas e consequentemente, as taxas de falsos negativos mais baixas, embora para atingir tal objetivo, revele valores de falsos positivos elevados, mostrando não ter grande precisão.

O HAAR e o MUD são os métodos produzem taxas de deteção baixas e consequentemente taxas de falsos negativos elevados, e consequentemente são os que revelam menos falsos alarmes, devido ao fraco poder de deteção neste tipo de cenários.

3.4.3 - Análise de Ocultações

Com o objetivo de analisar a capacidade dos algoritmos em lidar com situações em que existem ocultações de pessoas (totais ou parciais), e verificar o impacto que estas provocam na performance dos algoritmos, foram selecionados dois segmentos específicos da sequência de vídeo OSME1C, pertencente ao conjunto de dados do CAVIAR, onde se encontram várias situações de ocultação de indivíduos, provocadas pela presença de outras pessoas ou objetos na imagem ou pelos limites da própria imagem, como se pode verificar na figura 3.14, onde estão ilustradas *frames* exemplo de cada uma dos segmentos selecionadas. Na tabela 3.11 estão descritas os segmentos selecionadas para análise.

Tabela 3.11 – Representação dos segmentos selecionadas para análise de ocultações.

	***Frame* Inicial**	***Frame* Final**
	OSME1C	
Segmento 1	250	300
Segmento 2	840	890

Figura 3.14 – *Frames* exemplo de cada um dos segmentos selecionados.

Tabela 3.12 – Resultados em percentagem das métricas *frame-based* relativas a cada um dos algoritmos, em cada um dos segmentos selecionados.

	Zona 1				Zona 2			
	HOG	**IDIAP**	**HAAR**	**MUD**	**HOG**	**IDIAP**	**HAAR**	**MUD**
False Alarm Rate (FAR) (%)	0,59	32,84	5,44	**0**	**0**	30,36	3,43	2,39
Detection Rate (DR) (%)	34,99	55,18	28,78	**63,56**	38,24	53,18	33,14	**56,08**
False Negative Rate (FNR) (%)	65,01	44,81	71,22	**36,44**	61,76	46,81	66,86	**43,92**

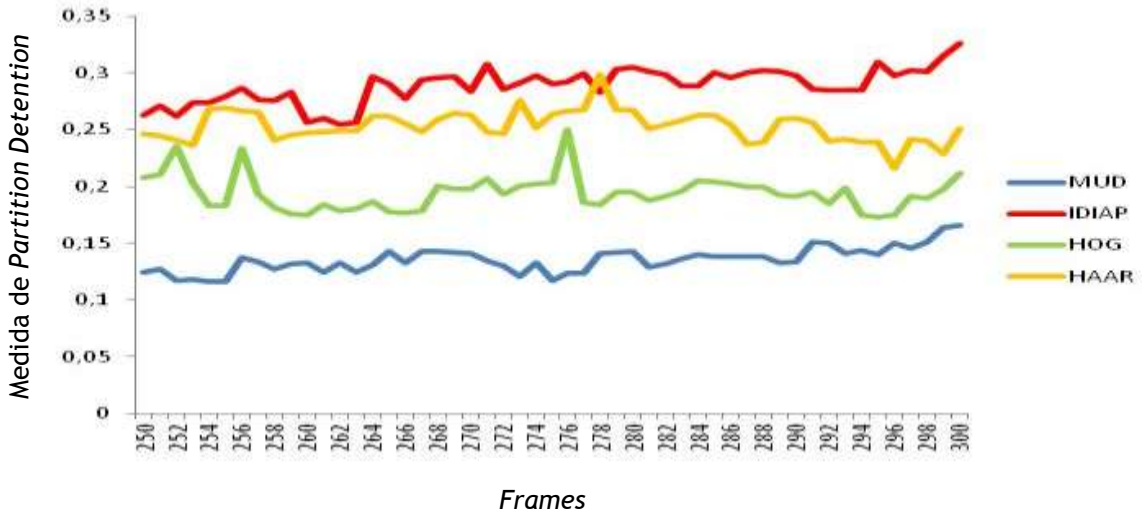


Figura 3.15 – Resultado das métricas *Partition Distance* relativas ao segmento 1.

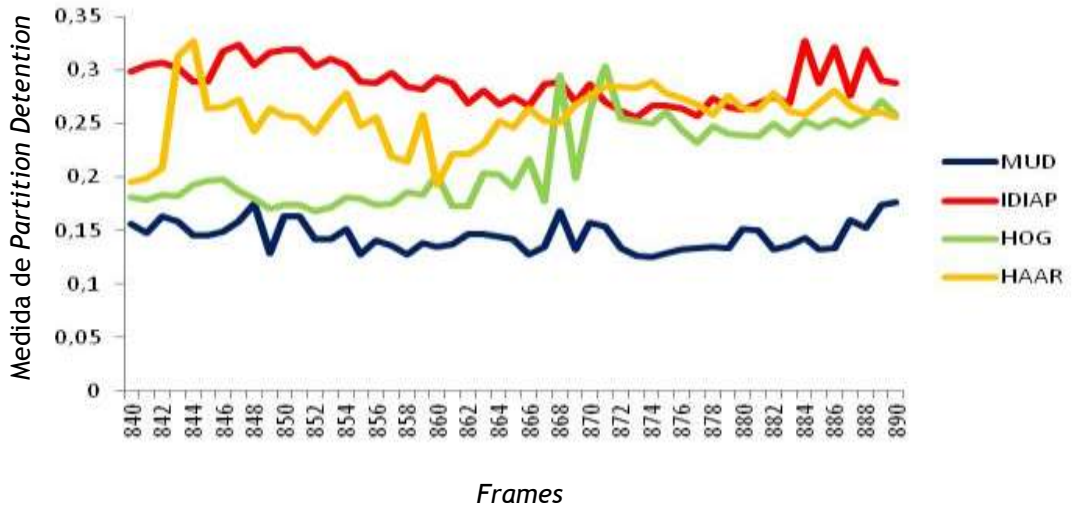


Figura 3.16 - Resultado das métricas *Partition Distance* relativas ao segmento 2.

Relativamente aos segmentos 1 e 2 (pertencentes à sequência de vídeo OSME1C do conjunto de dados do CAVIAR), conclui-se que o algoritmo que revela maior precisão relativamente à área das regiões detetadas, é o MUD, seguido do HOG e por fim o HAAR e o IDIAP, como se pode verificar através da análise dos gráficos presentes nas figuras 3.15 e 3.16, que refletem o erro entre a área das regiões detetadas e as existentes no ficheiro de referência.

O MUD demonstra ser simultaneamente o algoritmo que de uma forma geral, possui menor erro ao longo dos segmentos selecionados, rondando os 15%, e maior capacidade deteção (tabela 3.12), como era esperado, devido ao facto de possuir uma etapa de processamento responsável por tratar situações onde ocorrem ocultações totais ou parciais de indivíduos, como já foi referido anteriormente.

Os restantes algoritmos revelam resultados um pouco inferiores, embora o IDIAP demonstre maior capacidade de deteção que o HOG e o HAAR (tabela 3.12), revela respetivamente um erro 10% e 5% superior, relativamente à área das regiões detetadas (gráficos 3.15 e 3.16),o que reflete uma baixa precisão. O IDIAP embora seja o algoritmo que apresenta os segundos melhores resultados relativos à capacidade de deteção (tabela 3.12), é simultaneamente aquele que revela um erro maior (gráficos 3.15 e 3.16),à volta dos 30%. Este fenómeno está relacionado com facto de este algoritmo gerar um número elevado de falsos positivos (a rondar os 30%), revelando assim ser um algoritmo com uma precisão menor que os restantes. Este número elevado de falsos alarmes está relacionado com o aumento da complexidade da informação provocada pela proximidade entre os indivíduos que este tipo de situações proporciona.

O HOG e o HAAR revelam capacidades de deteção semelhantes embora os resultados do HOG superem ligeiramente os do HAAR, como se verifica através da análise da tabela 3.12. De

facto, através da análise dos gráficos 3.15 e 3.16, verifica-se que o HAAR revela de uma forma geral erros ligeiramente superiores ao HOG, principalmente devido à menor capacidade de deteção e precisão no tamanho das regiões detetadas que demonstra. Um dos pontos fortes destes algoritmos está relacionado com o número insignificante de falsos positivos que revelam mesmo neste tipo de situações, onde a grande proximidade entre os vários intervenientes na imagem, provoca normalmente um ligeiro aumento de falsos alarmes. De facto, os picos que se verificam nas *frames* 276, 869 e 871 relativamente ao HOG e nas 278, 844 e 874 relativamente ao HAAR, são justificados pela presença dos raros falsos positivos gerados pelos algoritmos ou por regiões de tamanhos pouco precisos, como se pode verificar na figura 3.17.

Figura 3.17 – Resultados da deteção do HOG relativamente às *frames* 276, 869 e 871 na linha de cima; Resultados da deteção do HAAR relativamente às *frames* 278, 844 e 874 na linha de baixo.

Exemplos que ilustram a precisão e capacidade de deteção destes algoritmos neste tipo de situações encontram-se na figura 3.18.

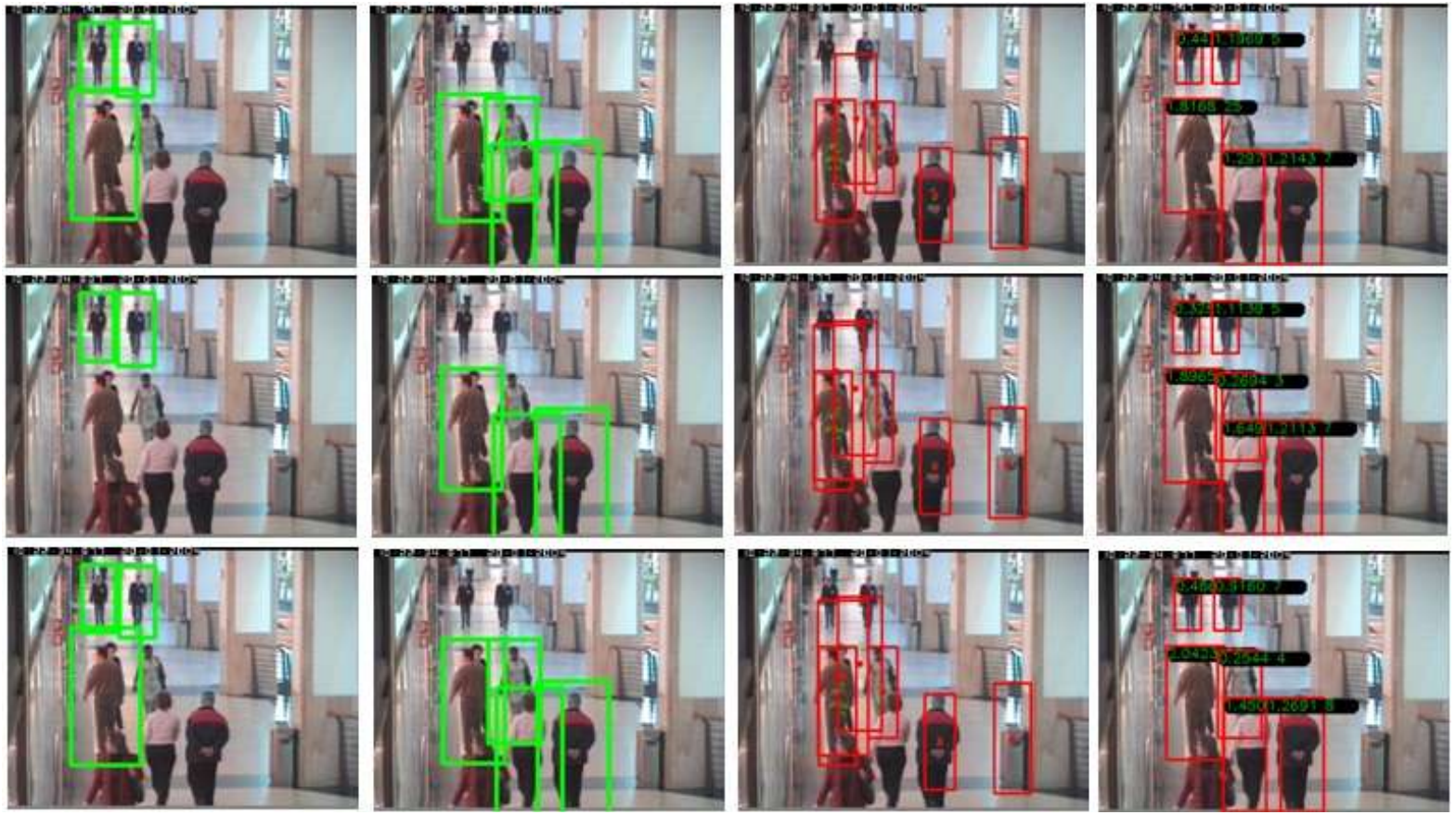


Figura 3.18 – Exemplos de deteção nos segmentos selecionados relativamente a todos os algoritmos. Coluna 1 – HAAR; Coluna2 – HOG; Coluna3 – IDIAP; Coluna4 - MUD

3.4.4 - Análise do impacto da distância da câmara ao objeto

Nesta secção são analisados aspetos relacionados com impacto que a distância dos objetos em relação à câmara, provoca na performance dos algoritmos. Como já referido anteriormente, as imagens do conjunto de dados do CAVIAR possuem uma resolução de 384 x 288. Como se trata de uma resolução baixa, regiões relativas a indivíduos que apareçam afastados da câmara, possuem um tamanho reduzido, provocando assim dificuldades na tarefa de deteção. Portanto, para avaliar o impacto que este aspeto provoca na tarefa de deteção em cada um dos algoritmos, foi selecionada um segmento da sequência de vídeo TES3C do conjunto de dados do CAVIAR, em que os indivíduos presentes na imagem se encontram de distantes da câmara. O segmento é constituído por 140 *frames*, que vai desde a *frame* 790 à 930. Na figura 3.19, encontram-se ilustradas a *frame* inicial e final do segmento selecionado.

Figura 3.19 – Representação da *frame* inicial e final do segmento selecionado.

***Tabela 3.13** – Representação do segmento escolhido para análise.*

	***Frame* Inicial**	***Frame* Final**
TES3C	790	930

Tabela 3.14 - Resultados em percentagem das métricas *frame-based* relativas a cada um dos algoritmos, relativamente ao segmento selecionado.

	HOG	**IDIAP**	**HAAR**	**MUD**
False Alarm Rate (FAR)(%)	23,93	66,60	21,01	**1,77**
Detection Rate (DR) (%)	13,59	0,35	14,35	**16,95**
False Negative Rate (FNR) (%)	86,41	99,65	85,65	**83,05**

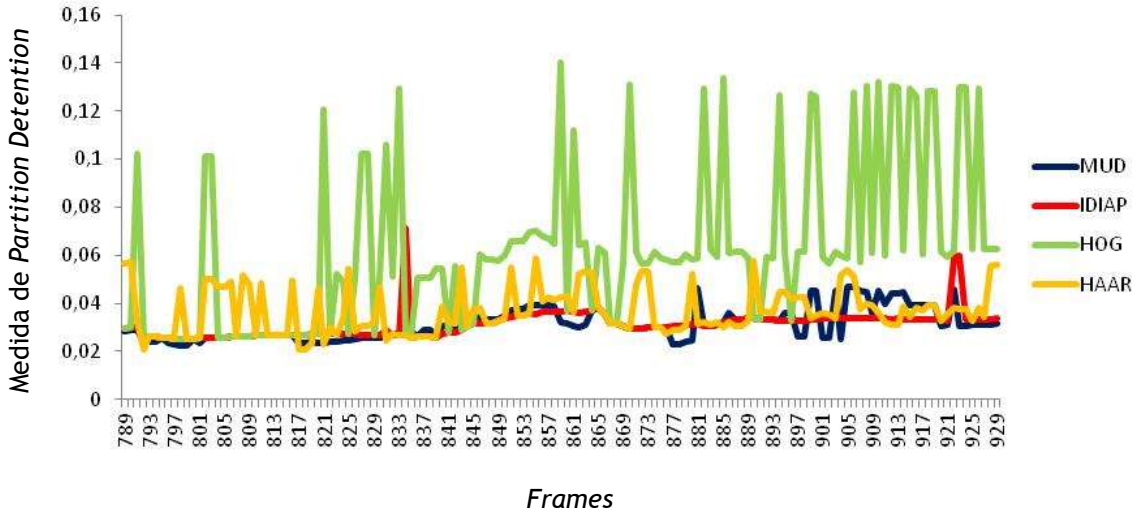


Figura 3.20 - Resultado das métricas *Partition Distance* relativas ao segmento selecionado.

Através da análise do gráfico presente na figura 3.20 e dos resultados presentes na tabela 3.14, verifica-se que embora todos os algoritmos apresentem resultados pouco satisfatórios, aquele que apresenta maior capacidade de deteção e precisão relativamente às regiões detetadas nesta situação concreta, é o MUD, seguido do HAAR e do HOG e por fim o IDIAP. Como já referido anteriormente, o gráfico reflete o erro entre a área das regiões detetadas e as presentes no ficheiro de referência.

Relativamente ao MUD, conclui-se que é o algoritmo que possui maior capacidade de detetar individuos em situações em que estes se encontram afastados da câmara, onde consequentemente as regiões têm dimensões reduzidas. Para além da sua capacidade de deteção neste tipo de situações, o MUD revela grande precisão relativamente à área das regiões detetadas, visto que estas se assemelham às dimensões reais, presentes no ficheiro de referência. Estes dois aspetos, em conjunto com o número reduzido de falsos positivos que o algoritmo gera (tabela 3.14), permitem atingir valores de erro mínimos, como se verifica através da análise do gráfico presente na figura 3.20.

Relativamente ao HAAR e ao HOG, os algoritmos possuem uma capacidade de deteção semelhante neste tipo de situações, atingindo resultados próximos do MUD (tabela 3.14). Ao analisar o gráfico presente na figura 3.20, conclui-se que o HAAR embora possua uma capacidade de deteção semelhante ao HOG (tabela 3.14), revela maior precisão, devido ao número ligeiramente inferior de falsos positivos que gera e à maior semelhança relativamente à área das regiões detetadas, daí demonstrar de uma forma geral um erro inferior. Verifca-se a influência da existência de falsos alarmes e de regiões de tamanhos pouco precisos, através dos vários picos presentes ao longo do gráfico presente na figura 3.20, ilustrados na figura 3.21. Os fatores que contribuem para a existência de discrepâncias entre o tamanho das regiões detetadas e as existentes no ficheiro de

referência, estão relacionados com a perturbação que a existência de sombas e a proximidade entre individuos, provoca no sucesso dos algoritmos, como se pode verificar na figura 3.21.

Figura 3.21 – *Frames* ilustrativas da deteção relativamente aos picos analisados. Na linha de cima encontram-se os *outputs* relativos ao HAAR e na de baixo os relativos ao HOG.

Em relação ao IDIAP, conclui-se que o algoritmo apresenta grandes dificuldades na tarefa de deteção neste tipo de situações, verificando-se que não é capaz de detetar praticamente nenhum individuo na zona selecionada, como se pode verificar através da análise da tabela 3.14. Nas poucas regiões detetadas verifica-se a influência das sombras, relativamente ao tamanho das regiões detetadas, originando regiões com áreas ligeiramente maiores que as presentes no ficheiro de referência. Os picos presentes nas *frames* 834 e 922, devem-se à perturbação causada pela maior existência de falsos alarmes, como se pode verificar na figura 3.22.

Figura 3.22 – Da esquerda para a direita; *Output* da *frame* 834 e 922 do segmento selecionado.

Na figura 3.23 encontram-se ilustrados alguns exemplos de deteção destes algoritmos, relativos à zona selecionada, exceto o IDIAP por não exibir resultados relevantes nesta situação em concreto.

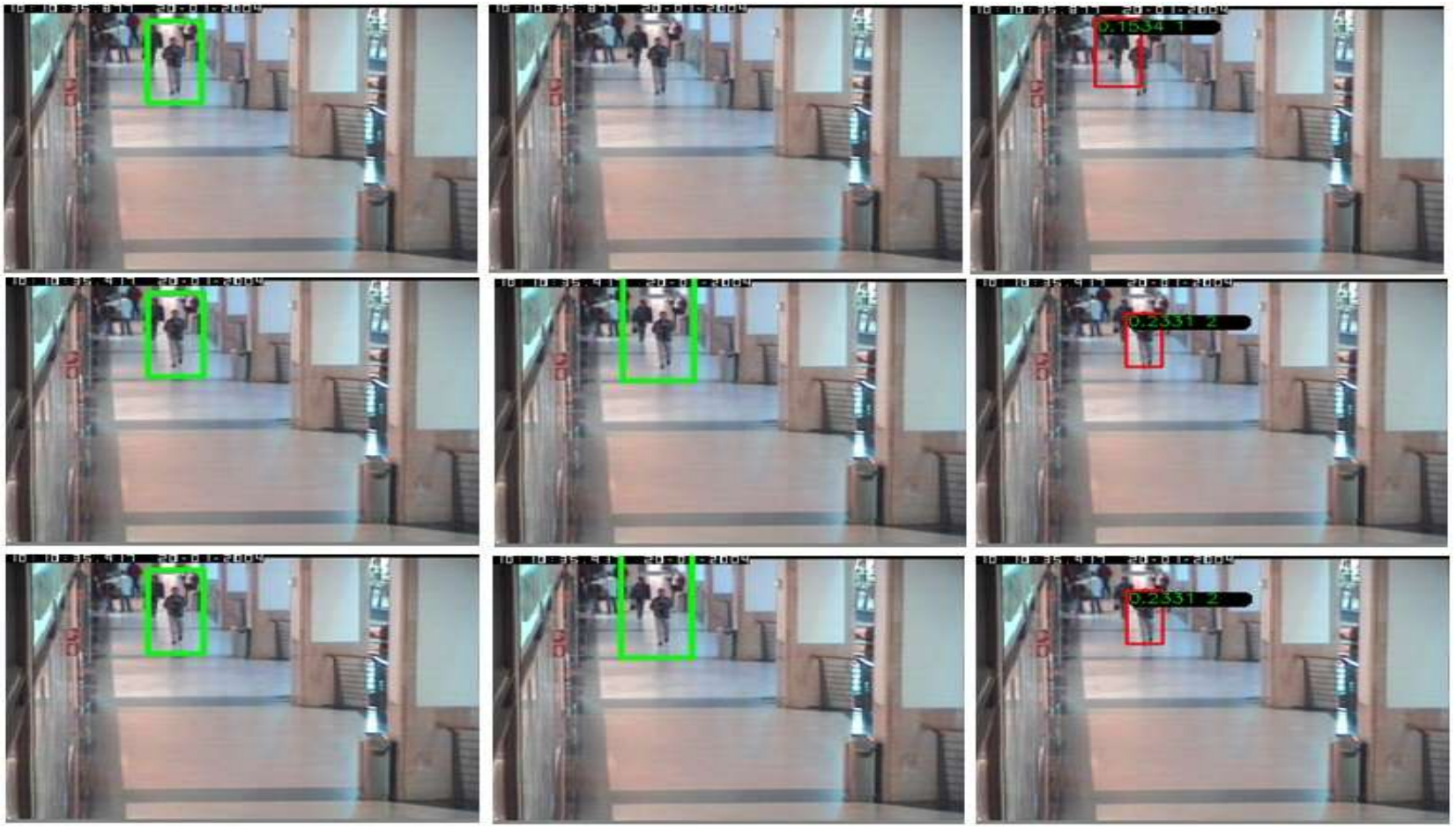


Figura 3.23 – Exemplos ilustrativos dos outputs dos algoritmos relativamente ao segmento selecionado; Coluna 1- HAAR; Coluna2 – HOG; Coluna3- MUD.

3.4.5 - Análise do impacto da resolução

Nesta secção são estudados e analisados aspetos relacionados com impacto que a resolução das imagens provoca na performance dos algoritmos. Para tal, foi selecionada como objeto de estudo, a sequência de vídeo CAM1-1 do conjunto de dados do INESC. Como já referido anteriormente, as imagens deste conjunto de dados possuem uma resolução elevada (1580 x 889), o que implica a utilização de um número elevado de janelas de pesquisa, provocando assim maiores desafios ao processo de deteção. O segmento selecionado para análise encontra-se descrita na tabela 3.15 e *frames* exemplo encontram-se ilustradas na figura 3.24.

É também aqui analisado o impacto que a resolução das imagens provoca no tempo de processamento dos algoritmos. Para isso são comparados os tempos de processamento relativos a uma sequência de vídeo do CAVIAR (OSME1C), cuja resolução das imagens é de 384 x 288 e uma sequência de vídeo do conjunto de dados do INESC (CAM1-1), cuja resolução das imagens é de 1580 x 889.

Figura 3.24 – *Frames* exemplo do segmento selecionado.

Tabela 3.15 – Representação do segmento selecionado para análise.

	***Frame* Inicial**	***Frame* Final**
CAM1-1	17	80

Tabela 3.16 – Resultados em percentagem das métricas *frame-based* relativas a cada um dos algoritmos, relativamente ao segmento selecionado.

	HOG	**IDIAP**	**HAAR**	**MUD**
False Alarm Rate (FAR) (%)	55,36	61,35	**33,62**	52,47
Detection Rate (DR) (%)	***59,18***	***60,40***	21,31	6,84
False Negative Rate (FNR) (%)	***40,82***	***39,60***	78,70	93,16

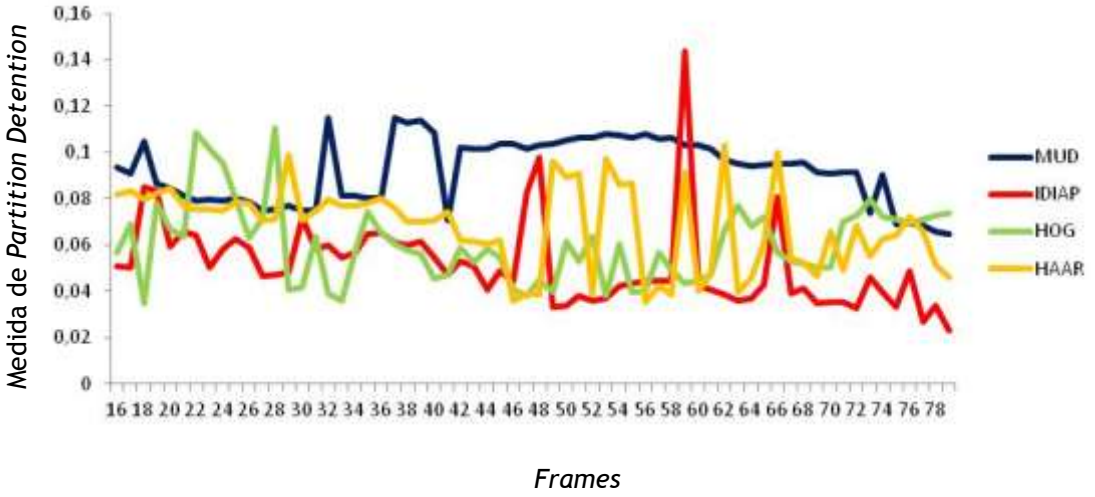


Figura 3.25 - Resultado das métricas *Partition Distance* relativas ao segmento selecionado.

Tabela 3.17 - Tabela ilustrativa dos tempos médios de processamento por *frame* de cada um dos algoritmos, relativamente à sequência OSME1C do CAVIAR e CAM1-1 do INESC.

	OSME1C	CAM1-1
HOG	0,39s	5,28s
HAAR	0,09s	1,03s
IDIAP	2,36s	5,92s
MUD	5,41s	69,01s

Através da análise da tabela 3.16, conclui-se que os algoritmos que revelam melhores resultados neste cenário concreto são o HOG e o IDIAP, seguidos do HAAR e por fim o MUD.

Relativamente ao HOG e ao IDIAP, verifica-se que a capacidade de deteção destes algoritmos aumenta substancialmente, relativamente aos resultados obtidos no conjunto de dados do CAVIAR, onde a resolução é muito inferior (384 x 288), embora para atingir estes níveis de performance, produzam um número superior de falsos positivos. Este fenómeno ocorre maioritariamente devido ao elevado número de janelas de pesquisa necessárias para analisar este tipo de imagens e à grande complexidade do cenário (maior que a existente no conjunto de dados do CAVIAR). Ao analisar o gráfico presente na figura 3.25, conclui-se que o HOG proporciona níveis de precisão semelhantes ao IDIAP, demonstrando erros de uma forma geral semelhantes, ao longo do segmento selecionado. Visto que a capacidade de deteção destes algoritmos é semelhante (tabela 3.16), a razão pela qual o IDIAP revela um erro ligeiramente superior, está relacionada com o número ligeiramente superior de falsos positivos que gera e com o tamanho, por vezes pouco preciso, das regiões detetadas. Verifica-se a influência destes fatores através da análise dos picos existentes nas *frames* 23 e 28 (relativamente ao HOG) e nas *frames* 47 e 59 (relativamente ao IDIAP), onde uma maior

existência de falsos alarmes e regiões pouco precisas, degradam a qualidade dos resultados obtidos, como se encontra ilustrado na figura 3.26.

Figura 3.26 - Resultados da deteção do HOG relativamente às *frames* 23 e 28 na linha de cima; Resultados da deteção do IDIAP relativamente às *frames* 47 e 59 na linha de baixo.

Relativamente aos tempos de processamento, verifica-se que as imagens presentes no conjunto de dados do CAVIAR têm uma resolução cerca de 12 vezes inferior às existentes no conjunto de dados do INESC. De facto, verifica-se que o tempo médio de processamento por *frame* aumenta em cerca de 12 vezes relativamente ao HOG, embora no que diz respeito ao IDIAP, se verifique um aumento para cerca de apenas o triplo. Relativamente ao IDIAP, a principal razão pela qual não se verifica um aumento tão acentuado no que diz respeito ao tempo médio de processamento quando a resolução aumenta, deve-se principalmente à etapa de subtração de fundo que o algoritmo possui, que reduz substancialmente a quantidade de informação a ser processada.

Conclui-se então, que o aumento da resolução provoca um impacto positivo nestes algoritmos, originando taxas de deteção elevadas, embora o número de falsos positivos seja também elevado.

Relativamente ao HAAR, verifica-se que a sua capacidade de deteção mantém-se de uma forma geral, quando aplicado a este tipo de resolução, originando taxas de deteção semelhantes, comparativamente com o acontece no conjunto de dados do CAVIAR. O mesmo não acontece em relação ao número de falsos positivos, que aumenta drasticamente neste tipo de cenários em que a resolução e complexidade do plano de fundo são elevadas. O fraco poder de precisão do algoritmo mantém-se, originando regiões com dimensões que se afastam

das existentes na informação de referência, como se pode verificar na figura 3.27, onde se encontram ilustrados alguns exemplos de deteção deste algoritmo, relativos ao segmento selecionado. No que diz respeito ao tempo médio de processamento por *frame*, este aumenta em cerca de 12 vezes quando aplicado às imagens do INESC.

Relativamente ao MUD, verifica-se que a sua capacidade de deteção diminui drasticamente de uma forma geral, quando aplicado a este tipo de resolução, originando taxas de deteção muito inferiores, comparativamente com o acontece no conjunto de dados do CAVIAR. Era de esperar que a performance do algoritmo aumentasse com o aumento da resolução, no entanto este fenómeno não se verifica, devido ao facto de não se ter acesso ao código fonte do algoritmo, e assim sendo torna-se impossível ajustar alguns parâmetros de forma a tornar este algoritmo mais compatível com este tipo de resoluções. O mesmo não acontece em relação ao número de falsos positivos, que aumenta drasticamente neste tipo de cenários em que a resolução e complexidade do plano de fundo são elevadas. Embora a capacidade de deteção sofra um impacto negativo acentuado, o poder de precisão do algoritmo mantém-se originando regiões muito próximas das existentes na informação de referência, como se pode verificar na figura 3.27, onde se encontram ilustrados alguns exemplos de deteção deste algoritmo, relativos ao segmento selecionado. Relativamente ao tempo médio de processamento por *frame*, este aumenta também em cerca de 12 vezes quando aplicado às imagens do INESC.

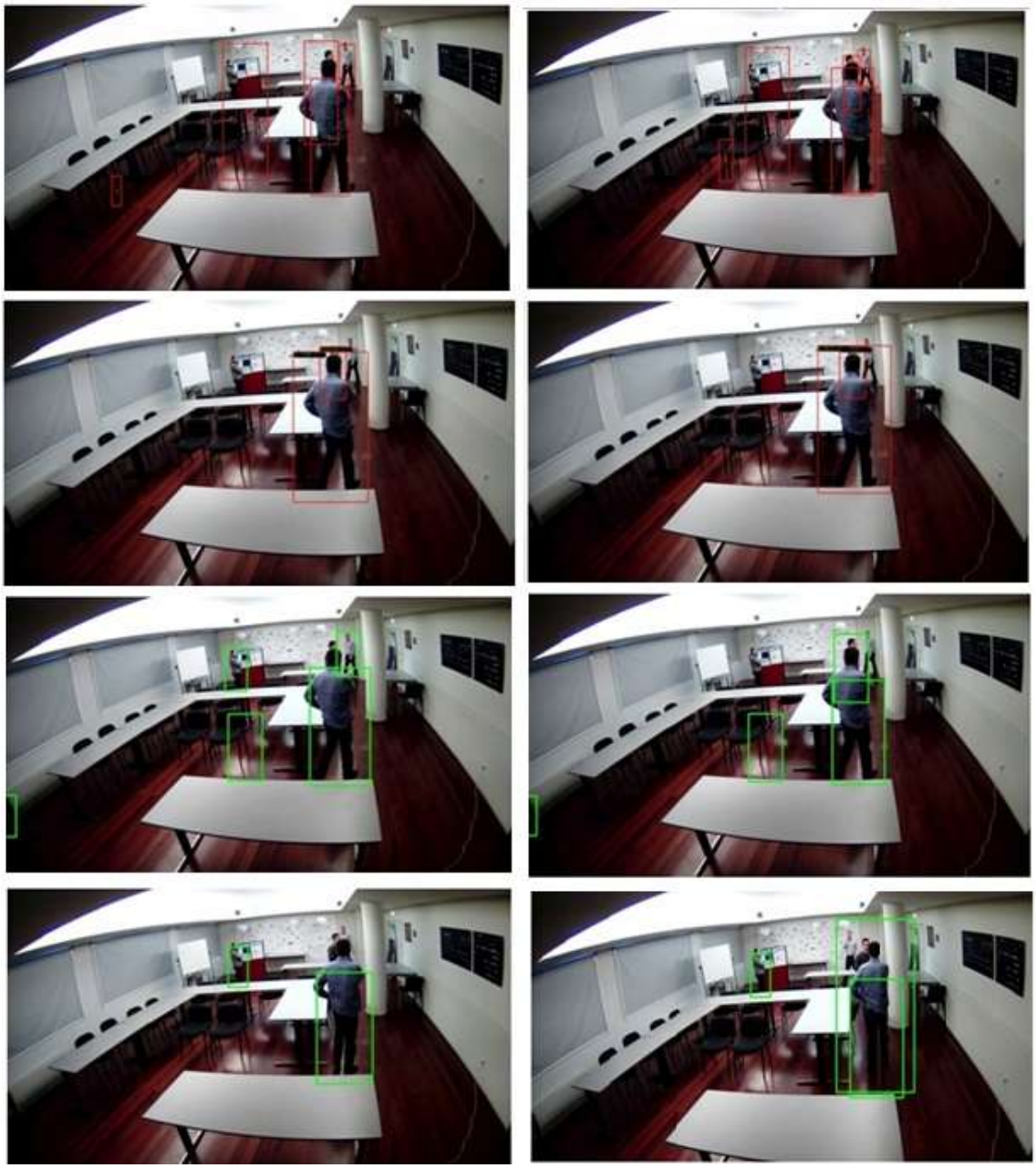


Figura 3.27 - Exemplos ilustrativos dos *outputs* dos algoritmos relativamente ao segmento selecionado; Linha 1- IDIAP; Linha2 – MUD; Linha3 - HOG; Linha4 – MUD.

3.5 - Conclusão

O algoritmo MUD apresenta os melhores resultados no conjunto de dados do CAVIAR com vista do corredor, devido ao facto de ser um algoritmo mais completo que os restantes, pois combina informação de cor (HOG) com textura (LBP), tornando assim a informação extraída das imagens mais completa e informativa. Relativamente à vista frontal os níveis de performance baixam significativamente, embora seja o algoritmo, que a par do HAAR, apresentam os melhores resultados neste cenário. Os principais fatores responsáveis por este decréscimo na performance devem-se essencialmente à postura e orientação dos indivíduos na imagem, que aparecem de forma transversal e a distância do objeto à câmara. A performance do algoritmo é fraca no conjunto de dados do INESC, não só devido ao facto de o ambiente ser mais complexo do que os do CAVIAR, com diversas situações de ocultação, como também devido à grande resolução das imagens, que originam regiões de interesse de grandes dimensões. O MUD é um algoritmo com precisão elevada, visto que gera baixas taxas de falsos positivos, sem prejudicar a tarefa de deteção. Um dos maiores contributos deste algoritmo está relacionado com o facto de possuir um módulo responsável por tratar situações de ocultação, o que lhe permite detetar pessoas em situações em que o corpo do indivíduo não se encontra totalmente visível, daí ser o algoritmo que apresenta melhores resultados neste tipo de situações. Conclui-se então, que o MUD possui a capacidade de lidar com situações de ocultação e de detetar indivíduos distantes da câmara (com resoluções pequenas), no entanto demonstrou ser sensível no que toca a grandes resoluções, demonstrando resultados pouco promissores neste tipo de situações, devido à impossibilidade de ajustar alguns parâmetros do algoritmo a resoluções superiores, pois não se teve acesso ao código fonte da aplicação. O MUD é o algoritmo mais lento relativamente à tarefa de deteção, revelando um tempo médio de processamento por *frame* de 5,4s no que diz respeito às imagens com resolução de 384 x 288. Verifica-se ainda que o tempo de processamento aumenta proporcionalmente com a resolução, visto que revela um tempo de processamento cerca de 12 vezes superior relativamente às imagens do conjunto de dados do INESC, onde a resolução é cerca de 12 vezes maior (1580 x 889).

O IDIAP consegue manter uma boa performance nos dois conjuntos de dados (CAVIAR e INESC), embora esta reduza significativamente no conjunto de dados do CAVIAR com vista frontal, devido aos fatores descritos anteriormente. Um dos pontos fracos deste algoritmo está relacionado com o facto atingir boas taxas de deteção à custa de um número elevado de falsos positivos, concluindo-se assim que a precisão do algoritmo é fraca.

Relativamente a situações de ocultação o IDIAP revela boa capacidade de detetar indivíduos neste tipo de situações, visto que consegue detetar indivíduos parcialmente ocultos, embora revele um número elevado de falsos positivos e em alguns casos o tamanho

das regiões detetadas não se aproxima dos desejados, devido à grande proximidade dos indivíduos nas imagens neste tipo de situações, que tornam mais difícil a extração da informação. No que toca à capacidade de deteção de indivíduos em escalas reduzidas (secção 3.4.4), verifica-se que o algoritmo demonstra grandes dificuldades neste tipo de situações, não conseguindo detetar praticamente nenhum individuo nesta situação. No que diz respeito à sua performance quando aplicado a imagens de grande resolução, esta aumenta significativamente, como era esperado, devido à maior e melhor quantidade de informação a ser analisada, embora consequentemente o número de falsos positivos seja elevado, devido ao número elevado de janelas de pesquisa a serem analisadas. Conclui-se então, que o IDIAP de uma forma geral apresenta bons resultados, embora seja sensível a alterações de postura e orientação dos indivíduos e a baixas resoluções, que implicam deteção em pequenas escalas. É de realçar relativamente ao tempo de processamento do algoritmo, que embora seja superior ao do HOG ou do HAAR em imagens com baixa resolução, à medida que a resolução aumenta o seu tempo tem um crescimento mais ténue que os restantes. Isto deve-se à etapa de subtração de fundo que o algoritmo executa, que permite reduzir significativamente a informação a ser processada, poupando assim processamento desnecessário.

O algoritmo HOG apresenta resultados pouco promissores na tarefa de deteção relativamente ao conjunto de dados CAVIAR, principalmente nas sequências com vista frontal, onde o facto de os indivíduos se encontrarem longe da câmara e se encontrarem na imagem com uma postura lateral e não frontal, prejudicam muito a performance do algoritmo, embora os resultados poderiam ser melhorados, se fossem ajustados alguns parâmetros do programa, de forma torná-lo mais compatível com resoluções inferiores. Em relação à capacidade de deteção em situações onde ocorrem ocultações de indivíduos, este algoritmo produz bons resultados. Relativamente à capacidade de deteção do algoritmo em situações em que os indivíduos se encontram distantes da câmara, proporcionando assim regiões de tamanho reduzido, conclui-se que esta reduz substancialmente. Esta limitação deve-se à sua necessidade de um suporte espacial maior, o que limita a utilização deste algoritmo em escalas mais reduzidas.

Os resultados indicam que o HOG embora não apresente os resultados esperados no conjunto de dados do CAVIAR, a sua performance aumenta substancialmente, à medida que a resolução das imagens aumenta, embora a sua precisão decresça significativamente, originando um número muito maior de falsos positivos.

O algoritmo HAAR apresenta resultados um pouco abaixo das expectativas relativamente à performance no processo de deteção, tanto no conjunto de dados do INESC como do CAVIAR, embora mantenha a sua capacidade de deteção nas sequências com vista frontal, produzindo resultados semelhantes. É um algoritmo que gera poucos falsos alarmes, embora este número aumente nas sequências do conjunto de dados do INESC (pelos motivos anteriormente descritos), mantendo de uma forma geral a sua capacidade de deteção. O HAAR revelou ser o

algoritmo que tem menor capacidade em lidar com situações onde ocorrem variadas ocultações, revelando pouca capacidade deteção neste tipo de situações e o tamanho das regiões detetadas, é, de uma forma geral superior ao desejado, devido à grande proximidade dos indivíduos nas imagens neste tipo de situações. Conclui-se então, que o HAAR embora não revele, de uma forma global os melhores resultados relativamente aos testes efetuados, revela uma certa imunidade aos desafios analisados, visto que tem a capacidade de manter minimamente a sua performance em todos os cenários.

Através do trabalho realizado neste capítulo torna-se clara a dificuldade em definir de uma forma geral, qual o algoritmo que revela melhores resultados na tarefa de deteção, devido à grande variedade de desafios que a deteção automática de pessoas proporciona. Verifica-se que aspetos como a resolução das imagens, a distância dos indivíduos à câmara, a complexidade do cenário ou até a variedade de posturas que os pedestres exibem nas imagens, provocam impactos negativos em todos os algoritmos. Ficou também demonstrado, que resoluções de imagem superiores, provocam um impacto positivo na performance dos algoritmos (exceto no MUD), no que toca à capacidade deteção, embora provoque de uma forma geral um aumento acentuado de falsos alarmes, reduzindo assim a sua precisão.

Conclui-se então que se o tempo de processamento não estiver em questão, o algoritmo MUD é o mais indicado para imagens com resoluções menores. Relativamente a imagens com resoluções superiores a escolha recai para o IDIAP, pois é o algoritmo que revela melhores resultados neste tipo de cenários. Outra razão pela qual a escolha recai sobre este algoritmo, deve-se ao facto de este algoritmo possuir uma etapa de subtração de fundo que reduz substancialmente a informação a ser processada, daí se verificar que o tempo médio de processamento por *frame* aumenta apenas para um pouco mais do dobro quando aplicado a resoluções cerca de 12 vezes superiores.

Se por outro lado o tempo de processamento for crucial (para por exemplo ser aplicado em sistemas em tempo real), as escolhas recairiam sobre o HOG para resoluções superiores e o HAAR escalas mais reduzidas, visto serem aqueles que demonstram melhor relação performance vs. Tempo de processamento, conseguindo atingir taxas de deteção aceitáveis num curto espaço de tempo.

Capítulo 4

Contribuições para a adaptação de um detetor de sinais de trânsito para deteção de pessoas

Nesta secção, são estudados e analisados aspetos relacionados com a possibilidade de adaptar um detetor utilizado na deteção de sinais de trânsito [Landesa2010], para a deteção automática de pedestres, a partir daqui denominado "detetor". A ideia de estudar esta possibilidade foi motivada pela semelhança existente entre esta abordagem e a desenvolvida por *Viola and Jones* em [Viola2004] que demonstrou resultados positivos na deteção de pedestres em [Viola2003], visto que a arquitetura do sistema se mantém inalterada, sem alterações no algoritmo Adaboost nem no *design* em cascata. A diferença reside nas features que utiliza, visto que enquanto as Haar-like consistem na codificação da média de intensidades entre duas regiões retangulares adjacentes, as features desenvolvidas em [Landesa2010] (quantum features) codificam a média da diferença de intensidades entre duas regiões que não necessitam ser adjacentes, como se pode verificar na figura 4.1.

Figura 4.1 - Representação da diferença entre as *features* exploradas em [Landesa2010] e [Viola2004] respetivamente.

É de realçar também que este detetor combina informação de dois tipos de *features* diferentes, que é uma opção frequentemente utilizada na literatura, demonstrando ser uma escolha acertada a fazer, pois combinando features diferentes obtém-se um conjunto de informação mais discriminativa, colmatando-se assim as suas limitações. A escolha de aplicar este detetor na deteção de pessoas, está também relacionado com o facto de os dois tipos de features utlizados serem baseadas no LBP e nas features *Haar-like*, que são regularmente utilizadas na literatura e serviram de ponto de partida para várias abordagens, que são respetivamente o LCP e as *quantum features*. Tornou-se também interessante explorar esta adaptação pelo facto de ser utilizada informação de contornos, que já mostrou fornecer informação rica e fiável para a deteção de pessoas.

Com o objetivo de adaptar este detetor na deteção de pessoas foi proposta uma etapa de pré-processamento, com o objetivo de realçar não só os contrastes globais das imagens, como também de regiões locais, de forma a tornar a informação mais discriminativa.

Foi também proposta uma etapa de treino onde foram gerados milhares de amostras positivas e negativas que serviram para treinar os classificadores utilizados.

Este capítulo é constituído por uma descrição dos vários aspetos relacionados com o detetor, como a sua arquitetura e as características que utiliza. De seguida são apresentados as contribuições feitas e discutidos os testes efetuados, com o objetivo de concluir se o detetor é aplicável à deteção de pedestres.

4.1 - Detetor Base

O detetor implementado em [Landesa2010], é aplicado em imagens em escalas de cinza e o processo de deteção é dividido em duas partes distintas. A primeira consiste numa pré seleção rápida de formas que se baseia numa variante do descritor de textura LBP, denominado LCP (*Local Contour Patterns*), que se encarrega de rejeitar maior parte das janelas de pesquisa, e de pré classificar as restantes numa de três categorias (circular ou dois tipos de triangulares), visto que praticamente todos os sinais de trânsito existentes na União Europeia são circulares, triangulares ou retangulares, cujos contornos podem ser rapidamente construídos a partir de um conjunto restrito de combinações de estruturas lineares, como se pode verificar na figura 4.2.

Figura 4.2 - Representação das janelas de pesquisa necessárias para definir as formas relativas a cada uma das categorias, através de combinações de estruturas lineares **[Landesa2010]**.

De seguida, um detetor baseado no *Adaboost* com uma estrutura em cascata, que utiliza um conjunto de características de textura mais simples, denominadas *quantum features*, analisa as regiões pré selecionadas na etapa anterior com o objetivo de efetuar uma classificação final das regiões, numa das três categorias de sinais existentes.

4.1.1 -LCP

Com o objetivo de tornar possível a utilização deste detetor em aplicações em tempo real, foi decidido criar uma etapa responsável por efetuar uma pré-filtragem rápida de regiões, baseada em características de forma, e uma pré-seleção das regiões que não são rejeitadas, dividindo-as em três categorias (regiões circulares, e dois tipos de triangulares). O processo consiste em aplicar o detetor *Canny* à imagem, de forma a extrair os contornos da imagem. O LCP consiste numa medida aplicada a imagens binárias, com o objetivo de localizar estruturas geométricas locais, sendo caracterizada pela formulação (4.1), em que P_b consiste no valor binário do píxel em questão e N corresponde ao comprimento da região quadrangular em torno do píxel central.

$$LCP_{B,N} = \sum_{b=0}^{B-1} P_b 2^b \tag{4.1}$$

A figura 4.3 ilustra algumas estruturas simples codificadas pelo $LCP_{8,3}$.

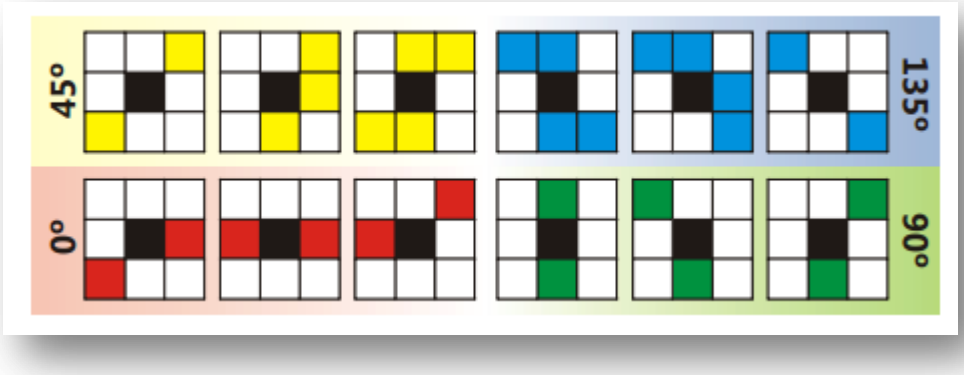


Figura 4.3 – Exemplos de estruturas lineares codificadas através do $LCP_{8,3}$ [Landesa2010].

Cada píxel existente na imagem de contornos é descrito através de uma *codeword* específica. A maior parte das *codewords* são descartadas de imediato, pois para descrever sinais de trânsito apenas se necessita de estruturas lineares com ângulos específicos; como se pode observar na figura 4.3. São então utilizadas as *codewords* que representam quatro conjuntos de direções diferentes.

São então construídos histogramas de quatro *bins*, cada um deles relativo a cada uma das direções definidas, de forma a contar o número de ocorrências de cada uma delas na região.

4.1.2 - *Quantum Features*

Depois da maior parte das janelas terem sido rejeitadas através da etapa de processamento anterior, um detetor baseado no *Adaboost* com uma estrutura em cascata, que utiliza um conjunto de características de textura mais simples, denominadas *quantum features*, analisa as regiões pré selecionadas na etapa anterior com o objetivo de efetuar uma classificação final das regiões. *Quantum features* são baseadas nas *features Haar-like*, que consistem na comparação de valores de intensidade entre pixéis em resolução nativa (tamanho das imagens positivas e negativas utilizadas para treinar o classificador), tendo em consideração todas as combinações de pixéis possíveis dentro de cada região, de forma similar ao que acontece com as *Haar-like features*, com a diferença de que os pixéis não necessitam obrigatoriamente de ser adjacentes, como se pode verificar na figura 4.4.

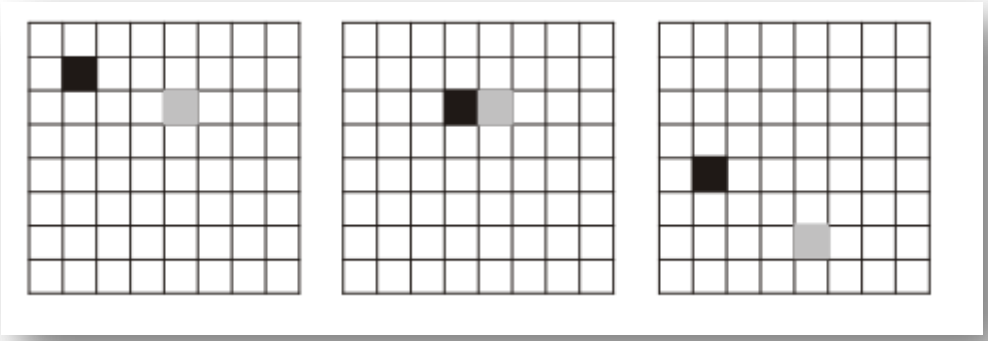


Figura 4.4 – Exemplo ilustrativo das *quantum features* para uma *patch* 8x8 **[Landesa2010]**.

Através da aplicação do conceito de polaridade desenvolvido em [Castro2010], as *quantum features* possuem informação capaz de distinguir se um píxel é mais claro ou escuro, ou até se semelhante ou não em questões de intensidade, relativamente a outro.

Analisando a morfologia dos sinais de transito, este tipo de representação tem uma importância significativa, visto que normalmente os sinais são constituídos por uma borda mais escura que o seu plano de fundo e mais clara que o conteúdo no seu interior. Como se pode observar na figura 4.5.

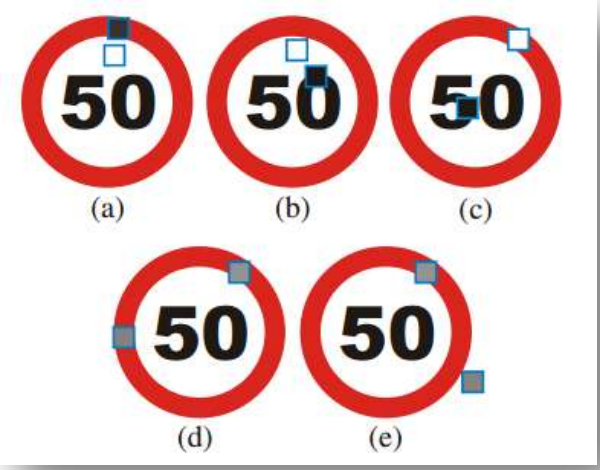


Figura 4.5 – Representação das *quantum features* com *polarity enhancement* **[Landesa2010]**.

4.2 - Proposta

Com o objetivo de adaptar este detetor na deteção de pessoas foi proposta uma etapa de pré-processamento que consiste num conjunto de operações aplicadas à imagem de entrada, com o objetivo de melhorar a performance do detetor.

Foi também proposta uma fase de treino dos detetores através da criação de um conjunto de amostras positivas e outro de negativas (de maior dimensão), com o objetivo de treinar as várias etapas dos classificadores.

4.2.1 - Etapa de pré-processamento

Foi desenvolvida uma etapa de pré-processamento, com o objetivo de realçar não só os contrastes globais das imagens, como também de regiões locais, de forma a tornar a informação mais discriminativa, realçando os objetos e esbatendo sombras presentes nas imagens. A etapa consiste na aplicação de um método de equalização de histogramas, que consiste no ajuste de contrastes através do espalhamento eficaz dos valores de intensidade mais frequentes. Uma das principais vantagens deste método é que é uma técnica relativamente simples facilmente invertível.

Na figura 4.6 encontram-se ilustradas imagens com e sem a aplicação do método, onde se verifica o aumento do contraste entre os indivíduos e o plano de fundo e realçando os contornos da imagem.

Figura 4.6 – Exemplos ilustrativos da aplicação da etapa de pré-processamento proposta; à esquerda *frames* sem a aplicação da etapa de pré-processamento e à direita as mesmas *frames* com a aplicação da etapa.

4.2.2 - Etapa de treino

Foram geradas milhares de imagens com exemplos positivos e negativos para treinar os classificadores, com uma relação de altura x largura de 3 x 1. Foram treinados os dois classificadores, selecionando-se aleatoriamente 5000 exemplos positivos. Relativamente aos exemplos negativos, foram utilizadas 5000 imagens de cada um dos conjuntos de dados utilizados (CAVIAR e INESC) para treinar as primeiras etapas de cada um dos classificadores, sendo as etapas seguintes treinadas através de exemplos negativos com diferentes resoluções. Imagens exemplo dos dois conjuntos encontram-se ilustradas na figura 4.7.

A arquitetura do detetor é semelhante à proposta por *Viola et al* em [Viola2003], com a diferença de utilizar quantum *features* com *polarity enhancement* como descritores para a construção dos "*weak classifiers*".

Figura 4.7 – Na linha de cima encontram-se exemplos de amostras positivas e na linha de baixo exemplos de amostras negativos, que foram utilizadas para treinar os vários estados do detetor.

4.3 - Testes e Avaliações

4.3.1 - Sumário das avaliações e testes efetuados no Dataset do CAVIAR

Na tabela 4.1 encontram-se os resultados relativos à performance geral do detetor nas seis sequências de vídeo do CAVIAR utilizadas no capítulo 3. As métricas aplicadas aos resultados obtidos são as mesmas que são utilizadas em toda a dissertação (*Frame Based* Metrics e *Symmetric Partition Distance Metrics*).

Tabela 4.1 - Resultados em percentagem das métricas *frame-based* relativas ao detetor, nas sequências do CAVIAR.

	OSOW2C	OSOW2F	OSME1C	OSME1F	TES3C	TES3F
False Alarm Rate (FAR) (%)	84,89	84,95	78,64	85,99	85,52	95,80
Detection Rate (DR) (%)	29,45	31,72	25,72	23,76	8,21	7,59
False Negative Rate (FNR) (%)	70,55	68,28	74,28	76,24	91,79	92,41

Conclui-se, no que diz respeito às sequências do CAVIAR com vista do corredor, que o detetor atinge taxas de deteção inferiores aos algoritmos analisados no capitulo 3, e consequentemente taxas de falsos negativos superiores, como se pode verificar na tabela 4.1 e 3.7, embora mantenha a sua capacidade de deteção relativamente à vista frontal, mostrando assim que mudanças na postura e orientação dos indivíduos não afetam a performance do detetor, como se pode verificar através da análise das tabelas 4.1 e 3.8. Para além de manter de uma forma geral a performance em cada uma das vistas, ainda consegue superar em praticamente todas as sequências qualquer um dos algoritmos analisados no capítulo anterior

No entanto para atingir estas taxas de deteção o detetor revela taxas de falso alarme elevadas, sendo esta uma das suas maiores limitações. O fator mais relevante para o número elevado de falsos positivos que o detetor gera, está relacionado com a fraca capacidade de rejeição que a etapa de pré-filtragem baseada no LCP possui, que é responsável pela pré-seleção das regiões que não são rejeitadas, pois as restrições geométricas definidas têm pouca capacidade de rejeição quando aplicadas à forma humana.

4.3.2 - Sumário das avaliações e testes efetuados no Dataset do INEC

Tabela 4.2 - Resultados em percentagem das métricas *frame-based* relativas ao detetor, nas sequências do INESC.

	CAM1-1	CAM2-1	CAM2-2
False Alarm Rate (FAR) (%)	96,67	94,46	95,42
Detection Rate (DR) (%)	79,38	71,15	93,20
False Negative Rate (FNR) (%)	20,62	28,85	6,80

A capacidade de deteção do detetor aumenta drasticamente nas sequências do INESC. Isto acontece devido ao facto de as imagens deste conjunto de dados possuírem resoluções muito maiores do que as do CAVIAR.

Ainda relativamente às taxas de deteção, o detetor revela bons resultados, superando até qualquer um dos algoritmos testados no capítulo 3, embora atinja estes valores à custa de um número muito elevado de falsos positivos. Comparativamente ao IDIAP (tabelas 3.9 e 3.10), que é o algoritmo que apresenta melhores resultados neste conjunto de dados, o detetor consegue superar este algoritmo em termos de taxas de deteção, na ordem dos 15% (valor médio), à custa de uma taxa de falsos positivos ligeiramente superior, na ordem dos 30% (valor médio). Como já referido anteriormente, o principal fator que lev o detetor a revelar taxas de falso alarme tão elevadas, está relacionado com o fraco poder de rejeição da etapa de pré-filtragem baseada no LCP.

4.3.3 - Análise de Ocultações

Com o objetivo de analisar a capacidade do detetor em lidar com situações em que existem ocultações de pessoas (totais ou parciais), e verificar o impacto que estas provocam na performance do algoritmo, procedeu-se de forma análoga à efetuada na secção 3.4.3.

Tabela 4.3 - Resultados em percentagem das métricas *frame-based* relativas ao detetor, relativamente aos segmentos selecionados.

	Segmento 1	Segmento 2
False Alarm Rate (FAR) (%)	76,64	78,64
Detection Rate (DR) (%)	28,55	23,26
False Negative Rate (FNR) (%)	71,34	76,73

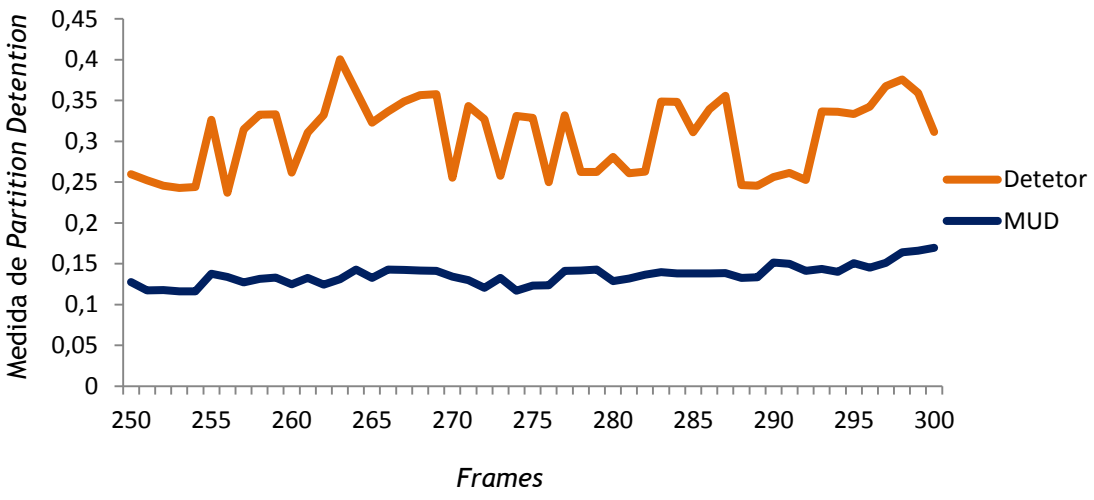


Figura 4.8 - Resultado das métricas *Partition Distance* relativas ao segmento 1.

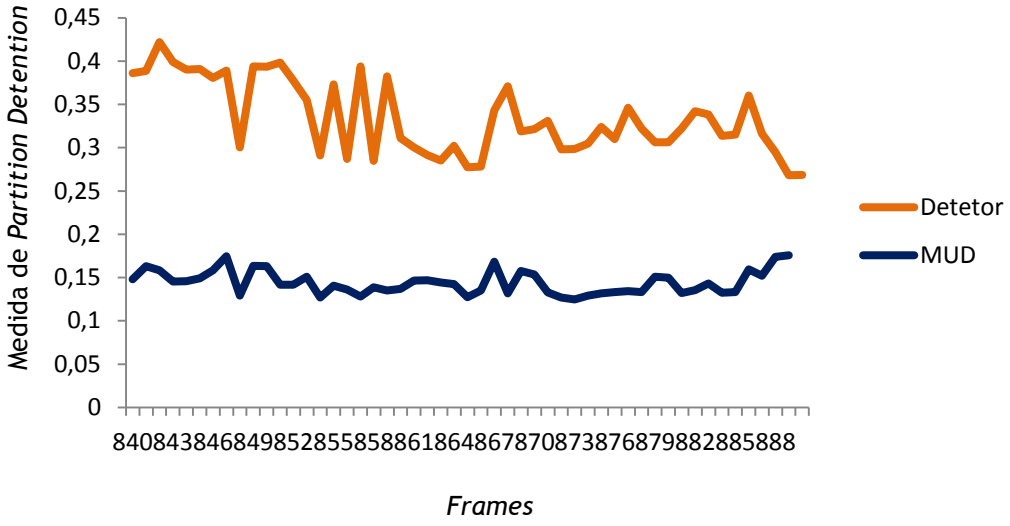


Figura 4.9 - Resultado das métricas *Partition Distance* relativas ao segmento 2.

Como já referido anteriormente, este detetor atinge níveis de performance baixos relativamente à tarefa de deteção, nas sequências de vídeo do CAVIAR. Para se conseguir efetuar uma análise menos confusa acerca dos resultados obtidos, decidiu-se apenas comparar os gráficos relativos aos resultados do detetor e do MUD, que é o algoritmo que melhor se comporta neste tipo de situações. Através da análise dos gráficos presentes nas figuras 4.8 e 4.9 e da tabela 4.3, conclui-se que o detetor tem a capacidade de detetar indivíduos, mesmo quando ocultados parcialmente, embora para atingir esse objetivo, gere um número elevado de falsos positivos em torno da região da imagem onde ocorrem as ocultações e para além disso as regiões por si detetadas são muto diferentes das reais, como se pode observar na figura 4.10, daí revelar erros muito superiores ao MUD, na ordem dos 20%. A razão de o detetor gerar uma grande quantidade de falsos positivos neste tipo de situações, está relacionado com o facto de que várias pessoas agrupadas num curto espaço, tornam a informação aí presente mais complexa, levando o detetor a decidir várias vezes de forma errada, para além da fraca capacidade de rejeição da etapa de pré-filtragem, concluindo-se assim que o detetor demonstra algumas limitações em lidar com este tipo de situações.

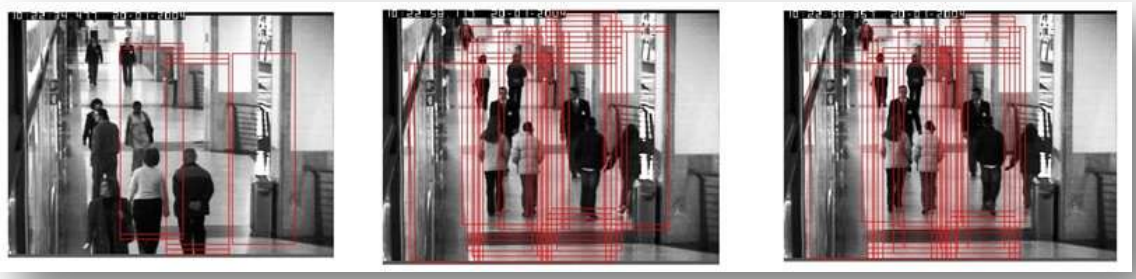


Figura 4.10 - Exemplos de deteção relativos ao detetor nos segmentos selecionados.

4.3.4 - Análise do Impacto da distância da câmara ao objeto

Nesta secção são analisados aspetos relacionados com impacto que a distância da câmara ao objeto provoca na performance do detetor, efetuando as mesmas experiências que foram realizadas na secção 3.4.4 e procedendo ao mesmo tipo de análise.

Tabela 4.4 - Resultados em percentagem das métricas *frame-based* relativas ao detetor no segmento selecionado.

	Detetor
False Alarm Rate (FAR) (%)	88,52
Detection Rate (DR) (%)	5,46
False Negative Rate (FNR) (%)	94,53

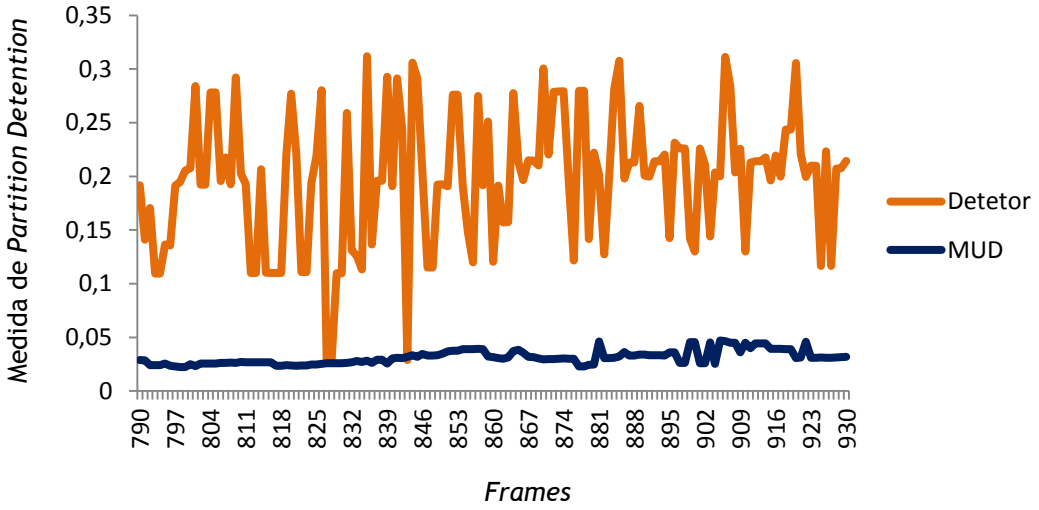


Figura 4.11 - Resultado das métricas *Partition Distance* relativas ao segmento selecionada.

Relativamente à capacidade de detetar indivíduos que se encontrem afastados da câmara, que consequentemente originam áreas reduzidas que delimitam as pessoas, pode-se concluir através da análise dos gráficos presentes nas figuras 4.11 e 4.12 e da tabela 4.4, que o detetor tem grandes limitações, não conseguindo detetar praticamente nenhum individuo neste tipo de situações. Decidiu-se mais uma vez, comparar apenas os gráficos relativos aos resultados do detetor e do MUD, por ser o algoritmo que revela melhores resultados neste tipo de situações. Conclui-se que o impacto provocado por baixas resoluções de imagem é grande, visto que nas sequências do INESC, que têm uma resolução muito maior, o detetor produz resultados muito mais promissores.

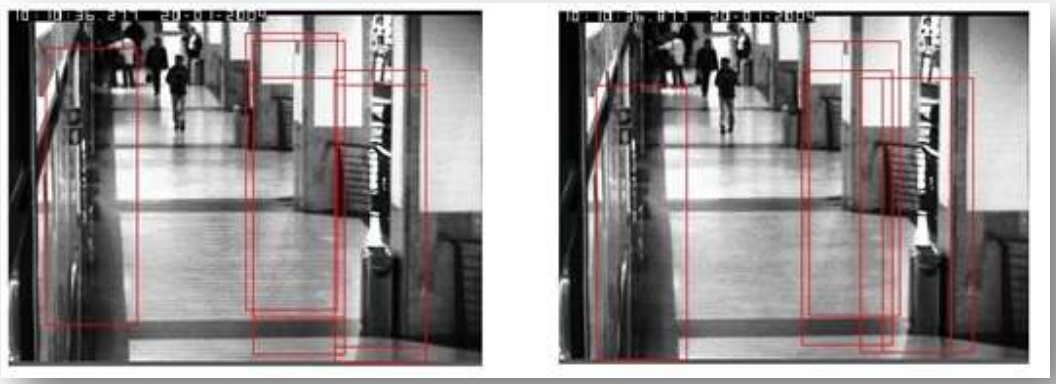


Figura 4.12 - Exemplos de deteção relativos ao detetor, no segmento selecionado.

4.3.5 - Análise do Impacto da Resolução

Nesta secção são estudados e analisados aspetos relacionados com impacto que a resolução das imagens provoca na performance dos algoritmos. Para tal, procedeu-se de forma análoga ao efetuado no capítulo 3.

Tabela 4.5 - Resultados em percentagem das métricas *frame-based* relativas ao detetor no segmento selecionado.

	Detetor
False Alarm Rate (FAR) (%)	94,31
Detection Rate (DR) (%)	81,76
False Negative Rate (FNR) (%)	18,24

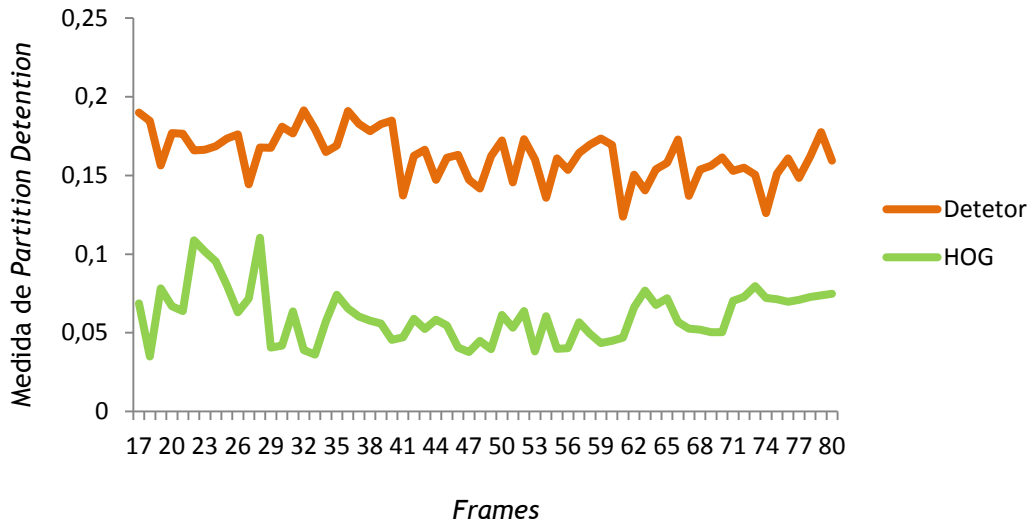


Figura 4.13 - Resultado das métricas *Partition Distance* relativas à zona selecionada.

Tabela 4.6 - Tabela ilustrativa dos tempos médios de processamento por *frame* de cada um dos algoritmos, relativamente à sequência OSME1C do CAVIAR e CAM1-1 do INESC.

	OSME1C	CAM1-1
HOG	0,39s	5,28s
HAAR	0,09s	1,03s
IDIAP	2,36s	5,92s
MUD	5,41s	69,01s
Detetor	20,69s	451,08s

Relativamente à capacidade de deteção deste detetor neste tipo de situações, verifica-se que a sua capacidade de deteção aumenta drasticamente de uma forma geral, quando aplicado a este tipo de resolução, originando taxas de deteção muito superiores, comparativamente com o acontece no conjunto de dados do CAVIAR. O mesmo acontece em relação ao número de falsos positivos, que aumenta drasticamente neste tipo de cenários em que a resolução e complexidade do plano de fundo são elevadas. Embora a capacidade de deteção sofra um impacto positivo acentuado, o poder de precisão do algoritmo mantém-se fraco, originando um número elevado de falsos positivos, como se pode verificar na figura 4.14, onde se encontram ilustrados alguns exemplos de deteção deste algoritmo, relativos ao segmento selecionado. O erro elevado que este detetor revela de uma forma global, deve-se essencialmente ao número elevado de falsos alarmes que revela.

Relativamente aos tempos de processamento por *frame* (tabela 4.6), é de realçar que o detetor revela tempos médios de processamento muito superiores aos algoritmos analisados no capítulo 3, tanto nas imagens do conjunto de dados do CAVIAR e do INESC, embora este aumente significativamente relativamente ao último conjunto, onde a resolução das imagens são muito superiores às do conjunto do CAVIAR, o que resulta num número muito superior de janelas de pesquisa.

O facto de o detetor ter sido implementado em Matlab, também contribui para o aumento dos tempos médios de processamento, pois como os outros algoritmos foram implementados em C++, tiram partido de algumas vantagens que programar neste tipo de linguagem tem comparativamente ao Matlab. Algumas das vantagens são a maior velocidade de execução e o menor consumo de recursos do sistema. No entanto a utilização do Matlab também tem os seus pontos fortes, como a grande capacidade de armazenamento de memória que este disponibiliza.

Figura 4.14 - Exemplos de deteção relativos ao detetor no segmento selecionado.

4.4 - Conclusão

Através da análise dos testes efetuados ao detetor e respetivos resultados, conclui-se que este produz resultados pouco animadores relativamente ao conjunto de dados do CAVIAR, embora mantenha minimamente a sua performance nas duas vistas (corredor e frontal), demonstrando assim que a capacidade de deteção do detetor não é afetada por variações de postura e orientação dos indivíduos.

Relativamente à capacidade de deteção em situações de ocultação, conclui-se que o detetor tem a capacidade de detetar indivíduos, mesmo quando ocultados parcialmente, embora para atingir esse objetivo, gere um número elevado de falsos positivos em torno da região da imagem onde ocorrem as ocultações e para além disso as regiões por si detetadas são muto diferentes das reais daí revelar erros elevados ao longo dos segmentos analisados. Isto deve-se ao facto de que várias pessoas agrupadas num curto espaço tornam a informação

aí presente mais complexa, levando o detetor a decidir várias vezes de forma errada, concluindo-se assim que o detetor demonstra dificuldades em lidar com este tipo de situações.

A capacidade de detetar indivíduos que se encontrem afastados da câmara, originando regiões de interesse reduzidas, é pouco promissora, como se verificou na secção 4.3.4.

Conclui-se ainda como era de esperar que o detetor produz melhores resultados quando aplicado a imagens com resoluções elevadas, visto que a sua performance aumenta drasticamente nas sequências do INESC, superando qualquer um dos algoritmos analisados no capítulo 3.

Uma das maiores limitações deste detetor está relacionada com o elevado número de falsos alarmes que gera, que aumenta com a resolução das imagens e a proximidade entre indivíduos, devido ao maior número de janelas de pesquisa exigido e a uma maior e mais complexa quantidade de informação existente, originando um número superior de falsos positivos. Verificou-se ainda que maior parte das falsas deteções ocorrem em torno dos indivíduos, o que significa que o número de falsos alarmes poderá facilmente ser reduzido, através da criação de uma etapa de pré-processamento baseado no LCP de forma análoga à existente para deteção de sinais de trânsito, mais compatível com a deteção de pedestres.

Outra das limitações do detetor está relacionado com o elevado tempo de processamento despendido na tarefa de deteção, que aumenta com a resolução, como era de esperar.

Conclui-se então que a ideia de adaptar este detetor à deteção de pedestres aparenta ter potencial, embora seja necessária recorrer ao melhoramento de alguns aspetos, de forma a melhorar a sua performance, essencialmente no que diz respeito à pouca robustez que este exibe, relativamente à sua capacidade de rejeição de regiões, à sua precisão no que diz respeito à área das regiões detetadas e ao elevado tempo de processamento exigido para efetuar a deteção.

Capítulo 5

Imagens Térmicas

Designa-se por termografia a técnica de detetar variações de temperatura num objeto através de imagens, medindo a radiação térmica emitida pelo mesmo. A termografia desempenha um papel de grande importância em vários setores de atividade como: na indústria automóvel e aeronáutica na manutenção elétrica e mecânica de equipamentos; no controlo de reatores e torres de refrigeração na indústria química; na engenharia civil para avaliação do isolamento térmico de edifícios, identificação de zonas de infiltração e fugas; na área militar e policial para o combate a crimes em ambientes de total escuridão ou quando os criminosos se encontram dissimulados no ambiente local; permite auxiliar os bombeiros na localização de vítimas em locais de fumo intenso e escuridão; na área da segurança rodoviária no auxílio da visão noturna dos automobilistas. Pelo facto de ser uma técnica não invasiva, a termografia é, também aplicada na área da medicina como um método imagiológico para o diagnóstico de inúmeras doenças, como se pode verificar na figura 2.39.

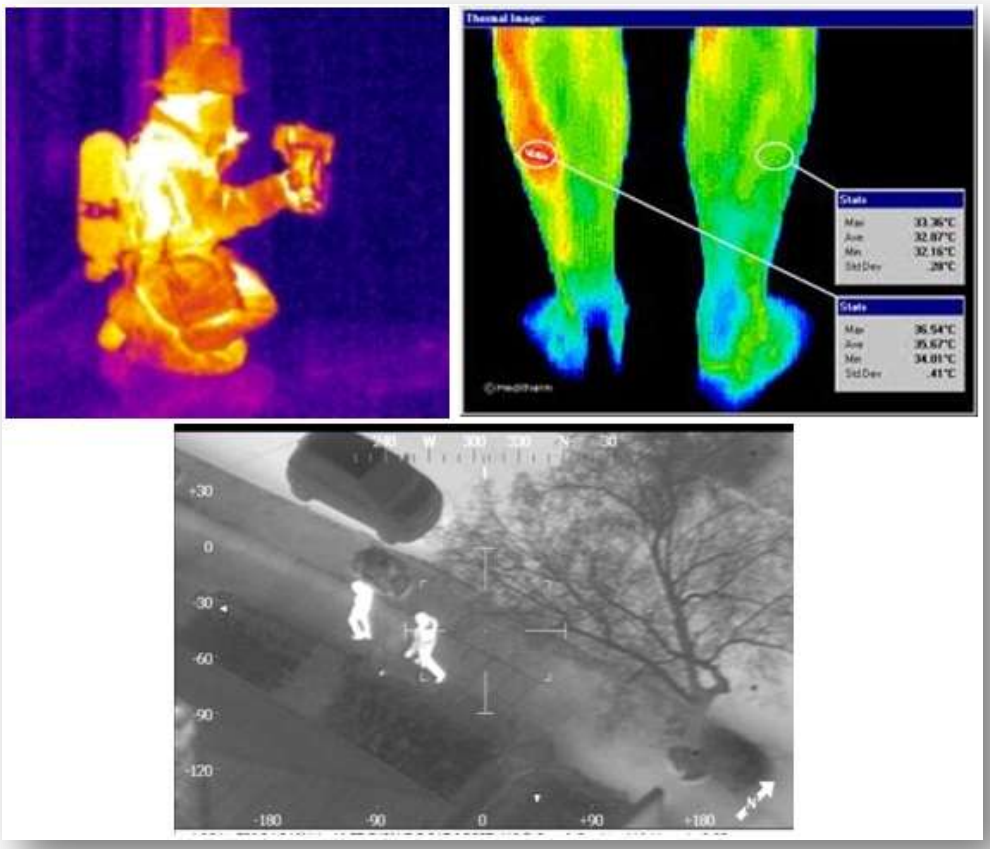


Figura 5.1 – Exemplos de várias aplicações de imagens térmicas em diferentes setores.

A utilização deste tipo de informação traz várias vantagens, principalmente pelo facto de poder fornecer informação mais precisa e fiável em várias situações em que outras fontes de informação não têm possibilidade de o fazer, como por exemplo em ambientes onde não existe iluminação, ou onde fatores climatéricos, como chuva intensa ou nevoeiro, degradam a informação capturada, daí este tipo de informação ser utilizado na tarefa de deteção, como se pode observar em [Yun2007] ou em [Soga2008].

A decisão de explorar a utilização de imagens termográficas está também relacionada com o pouco enfase dado a este tipo de imagens na literatura e pela possibilidade de integrar com sucesso técnicas e metodologias utilizadas em imagens RGB neste tipo de imagens. O facto de ser possível extrair destas imagens de uma forma fiável informação de contornos ou de textura por se encontrarem bem definidos nestas imagens e pelo sucesso demonstrado na utilização deste tipo de informação na deteção em imagens RGB, comprovam essa possibilidade.

No entanto as imagens termográficas também têm as suas limitações. Um dos principais desafios que este tipo de informação enfrenta, está relacionado com a maior ou menor capacidade de absorver e transmitir energia que os elementos constituintes do cenário possuem, originando várias reflexões que podem originar vários falsos alarmes, sendo este o

principal problema a ser analisado quando se utiliza imagens termográficas como fonte de informação. Este capítulo está então dividido em sete subcapítulos, começando com uma breve introdução à termografia e respetivas áreas de aplicação. O subcapítulo 5.1 consiste na descrição de aspetos relacionadas com as características e particularidades das câmaras termográficas. Na secção 5.2 são descritos os dados de teste que foram escolhidos para testar os algoritmos. Os algoritmos escolhidos são os mesmos que foram utilizados no capítulo 3, com o objetivo de analisar a sua capacidade de adaptação a este tipo de imagens. O subcapítulo 5.3 é constituído pela descrição das métricas de avaliação aplicadas aos resultados obtidos. No subcapítulo 5.4 encontra-se descrita a proposta de um filtro de pós-processamento que foi concebido, com o objetivo de remover reflexões existentes nas regiões detetadas, baseado num método de remoção de sombras presentes em imagens RGB implementado em [Karaman2009], melhorando assim a deteção neste tipo de ambientes. A motivação de desenvolver este tipo de filtro advém do facto que as reflexões provocam perturbações no funcionamento do algoritmo, originando falsos positivos e aumentos da área das regiões detetadas. O capítulo termina com os subcapítulos 5.5 e 5.6 que consistem respetivamente na demonstração e análise dos resultados obtidos e respetivas elações que foram retiradas acerca da adaptabilidade dos algoritmos a este tipo de imagens, bem como o impacto da integração do filtro proposto nos mesmos.

5.1 - Câmaras Térmicas

As câmaras termográficas operam na gama de infravermelhos. Como se pode observar na figura 5.2, esta gama encontra-se situada entre a luz visível e as micro-ondas no espectro eletromagnético. Os comprimentos de onda vão desde 1μm até 14μm, estando este intervalo subdividido em três partes: infravermelhos curtos (*Near - Infrared* - NIR) que vai desde 1 μm até 2 μm; infravermelhos médios (*Middle - Infrared* - MIR) que vai dos 2 μm até os 4 μm e infravermelhos longos (*Far- Infrared* - FIR) que vai desde os 4 μm até 1mm.

A zona entre os 4 μm a 1 mm é incomum para fins de geração de imagens térmicas devido à alta absorção espectral da atmosfera nesta faixa.

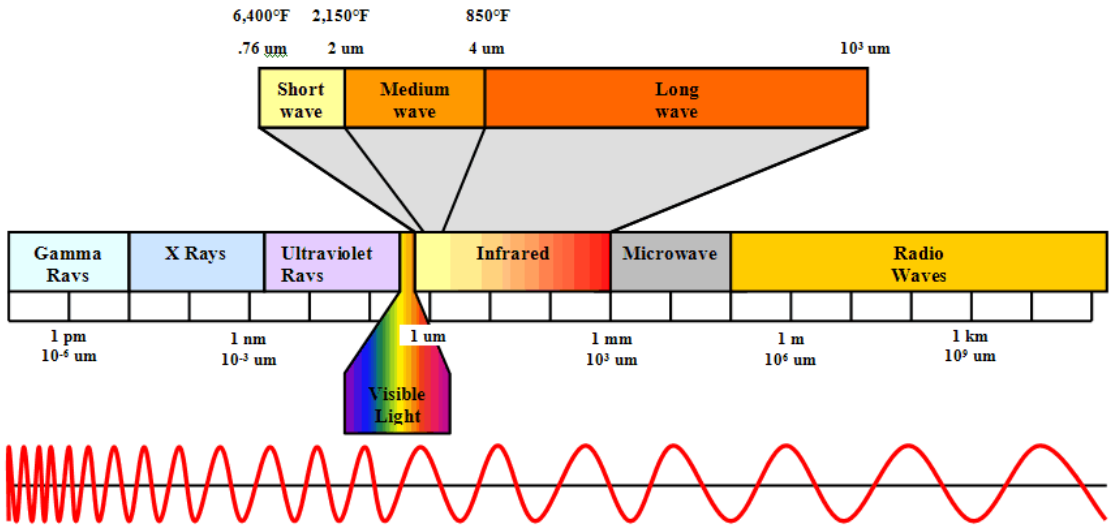


Figura 5.2 – Representação do espetro Eletromagnético.

Os raios infravermelhos apesar de não serem detetados pela visão humana são sentidos sob a forma de calor. Como tal, podem ser utilizados como uma forma de medir o calor irradiado por um objeto. Este pode ser classificado como sendo um corpo negro ou um corpo real. O primeiro é um objeto capaz de absorver toda a radiação que incide sobre ele em qualquer comprimento de onda enquanto o segundo é um objeto capaz de emitir uma determinada parte da energia. O parâmetro que determina a capacidade de emissão de energia é a emissividade (ε). Qualquer objeto (orgânico ou inorgânico) que possua uma temperatura acima do zero absoluto (0 K, -273,15 °C ou -459 F) emite uma certa quantidade de radiação infravermelha em função da sua temperatura. Essa radiação emitida pode incidir sobre a superfície de outro objeto podendo ser refletida, absorvida ou transmitida que será posteriormente captada pela câmara termográfica, *(*figura 5.3).

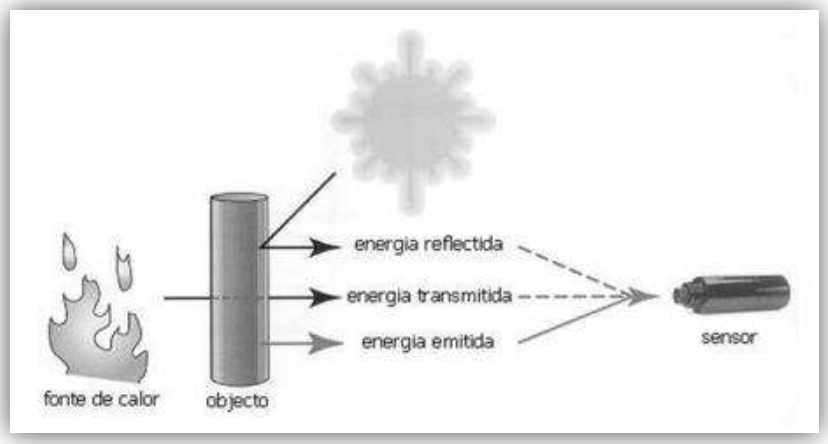


Figura 5.3 – Imagem ilustrativa da reflexão, absorção e transmissão de energia.

O princípio de funcionamento de uma câmara termográfica baseia-se na lei de *Stefan-Boltzmann*. A lei enuncia que a energia radiante total emitida por um corpo negro por unidade de superfície é proporcional à quarta potência da temperatura absoluta. Sendo expressa pela fórmula matemática (5.1):

$$W = \sigma \varepsilon T^4 \tag{5.1}$$

Onde:

W – Potência Radiante $\left[{}^{W}/_{m^2}\right]$;
σ – Constante de *Stefan - Boltzmann* $\left[5{,}7 \times 10^{-8}\ {}^{W}/_{K^4 m^2}\right]$;
ε – Emissividade;
T – Temperatura absoluta [K].

5.2 - Dados de Teste

Foi utilizado um conjunto de imagens térmicas provenientes de três cenários distintos com o objetivo de englobar o maior número possível de fatores e desafios que possam advir da análise de imagens térmicas, principalmente no que toca a reflexões; Estas foram capturadas numa garagem e dois corredores, sendo um deles rodeado de vidros e janelas.

As imagens foram divididas em duas sequências distintas, *Img_thermal1* e *Img_thermal2*, contendo o primeiro as imagens relativas à garagem e o último aos corredores.

Foram também criadas duas outras sequências, *Img_thermal1clean* e *Img_thermal2clean*, onde as reflexões existentes nas imagens foram removidas manualmente. Estas últimas sequências foram criadas com o intuito de se analisar e comparar o impacto das reflexões e reflexões na performance dos algoritmos. *Frames* Ilustram-se exemplos destas sequências na figura 5.4. De notar que foram utilizadas imagens em tons de cinza, no entanto, as imagens estão representadas com uma palete de cores, para auxiliar o leitor.

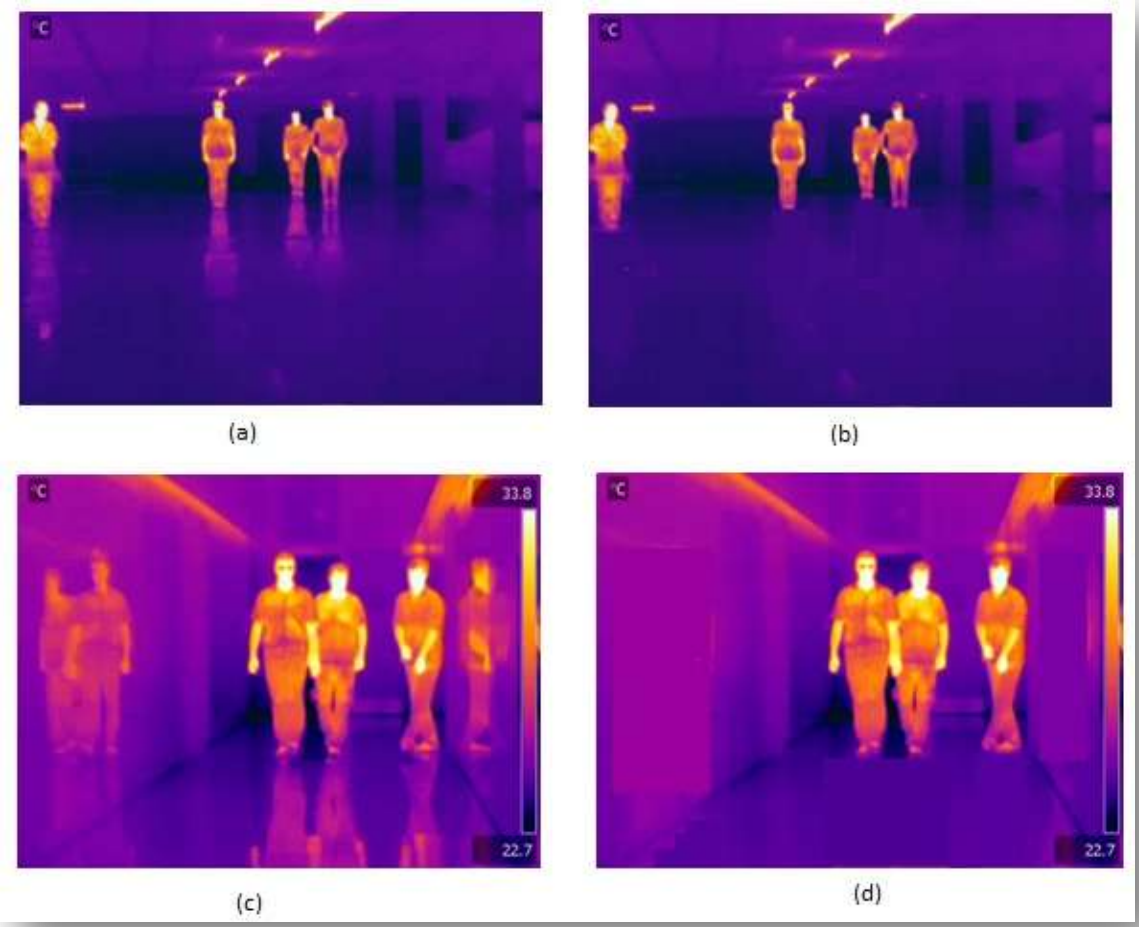


Figura 5.4 – *Frames* exemplo das sequências utilizadas. (a) *Img_Thermal1*, (b) *Img_Thermal1clean*, (c) *Img_Thermal2*, (d) *Img_Thermal2clean*.

5.3 - Métricas de Avaliação

As métricas de avaliação aplicadas são as já referidas e utlizadas anteriormente, a variante das *Object metrics* (*Frame-based*) [Bashir2006] desenvolvidas nesta dissertação, e as *Partition distance metrics (Symmetric Partition distance)* [Cardoso2009]. Através das primeiras é possível verificar o impacto do filtro desenvolvido, em aspetos relacionados com falsas deteções devido a reflexões. Através das últimas torna-se possível analisar o impacto do filtro na remoção de reflexões parciais existentes nas áreas detetadas.

5.4 - Filtro Proposto

Foi proposta a implementação de um filtro de pós-processamento baseado no método desenvolvido em [Karaman2009], aplicado a todos os algoritmos utilizados nesta dissertação, com o objetivo de diminuir, por um lado o número de falsas deteções devido a reflexões, e por outro a área das regiões detetadas, visto que estas regularmente incluem parte de reflexões no pavimento, como se pode verificar na figura 5.5.

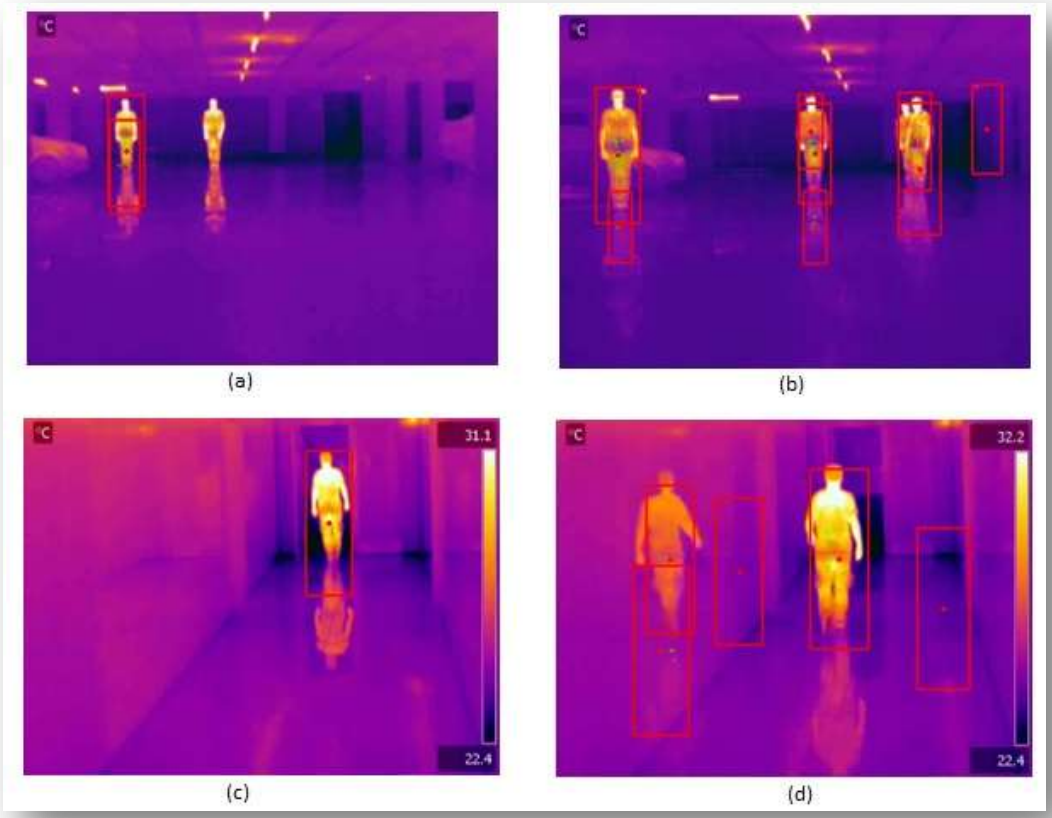


Figura 5.5 – Exemplos do impacto de reflexões totais e parciais na deteção em imagens térmicas.

A motivação de desenvolver este filtro está relacionado com o facto de que as reflexões se encontram bem definidas em imagens térmicas, provocando assim perturbações no funcionamento dos algoritmos, originando falsas deteções e aumentos nas dimensões das regiões detetadas, logo é esperado que a integração de uma etapa de análise e remoção de reflexões provoque um impacto positivo na performance dos algoritmos.

O filtro é constituído por dois blocos distintos, como se pode observar na figura 5.7, sendo o primeiro responsável pela remoção de regiões consideradas como sendo reflexões, ou seja, que não incluem nenhum indivíduo, ou parte dele. O segundo é responsável por diminuir o tamanho das regiões restantes, através da remoção de reflexões parciais que possam existir

junto da parte inferior das regiões, provocadas pelo espelhamento dos pés e pernas no pavimento, como se pode observar na figura 5.6.

Figura 5.6 – Exemplo de uma falsa deteção devido à presença de reflexões parciais provocados pelo espelhamento do corpo no pavimento.

O filtro é aplicado *frame* a frame como etapa de pós processamento, analisando todas as regiões identificadas pelo algoritmo. A figura 5.7 ilustra a arquitetura geral da abordagem.

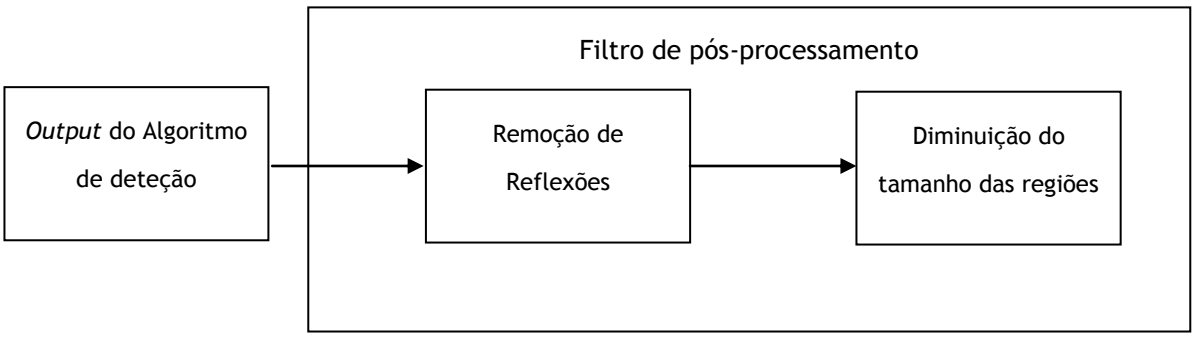


Figura 5.7 – Filtro de pós-processamento.

Na figura 5.8 encontra-se ilustrado um exemplo do efeito da aplicação do filtro numa *frame* exemplo.

Figura 5.8 – Exemplo de deteção sem filtro (a) e com filtro de pós-processamento (b).

O primeiro bloco (Remoção de reflexões) consiste num valor de *threshold (Thresh)* adaptativo aplicado a cada região detetada positivamente pelo algoritmo, caso o valor da região satisfaça o valor de *Thresh* definido, a região é aceite, caso contrário esta é descartada, como se pode verificar no esquema representado na figura 5.9.

É então calculada a soma da intensidade de todos os pixéis de cada região de contornos (figura 5.10 (b)), identificando-se assim aquela que possui a soma da intensidade dos píxeis mais elevada (Max_{soma}), que corresponde à região mais brilhante da imagem. O valor de threshold *Thresh* é calculado através da seguinte fórmula:

$$Thresh = val \times Max_{soma} \tag{5.2}$$

Definiu-se empiricamente que para a sequência *Img_thermal1*, $\text{val} = {}^{1}/_{3}$ e para a *Img_thermal2*, $\text{val} = {}^{2}/_{3}$, de forma a maximizar a performance do filtro, embora seja possivel ajustar facilmente este valor, caso o cenário que esteja a ser analisado assim o exija, visto ser um parâmetro de afinação do filtro. A razão de os valores de *Thresh* diferirem nas duas sequências, advém do facto que na última o ambiente circundante possui maior capacidade de absorção da radiação emitida pelos corpos, devido aos vidros e janelas presentes nas paredes e ao envernizamento do chão, originando reflexões mais intensas, logo com valores de intensidade mais próximos dos reais.

Devido à maior ou menor capacidade de absorção de radiação que o ambiente circundante possui, a média da intensidade dos pixéis existentes numa região que contém uma reflexão, é sempre inferior à de uma região que inclua o próprio objeto, mesmo quando isso não é facilmente visível a olho nu.

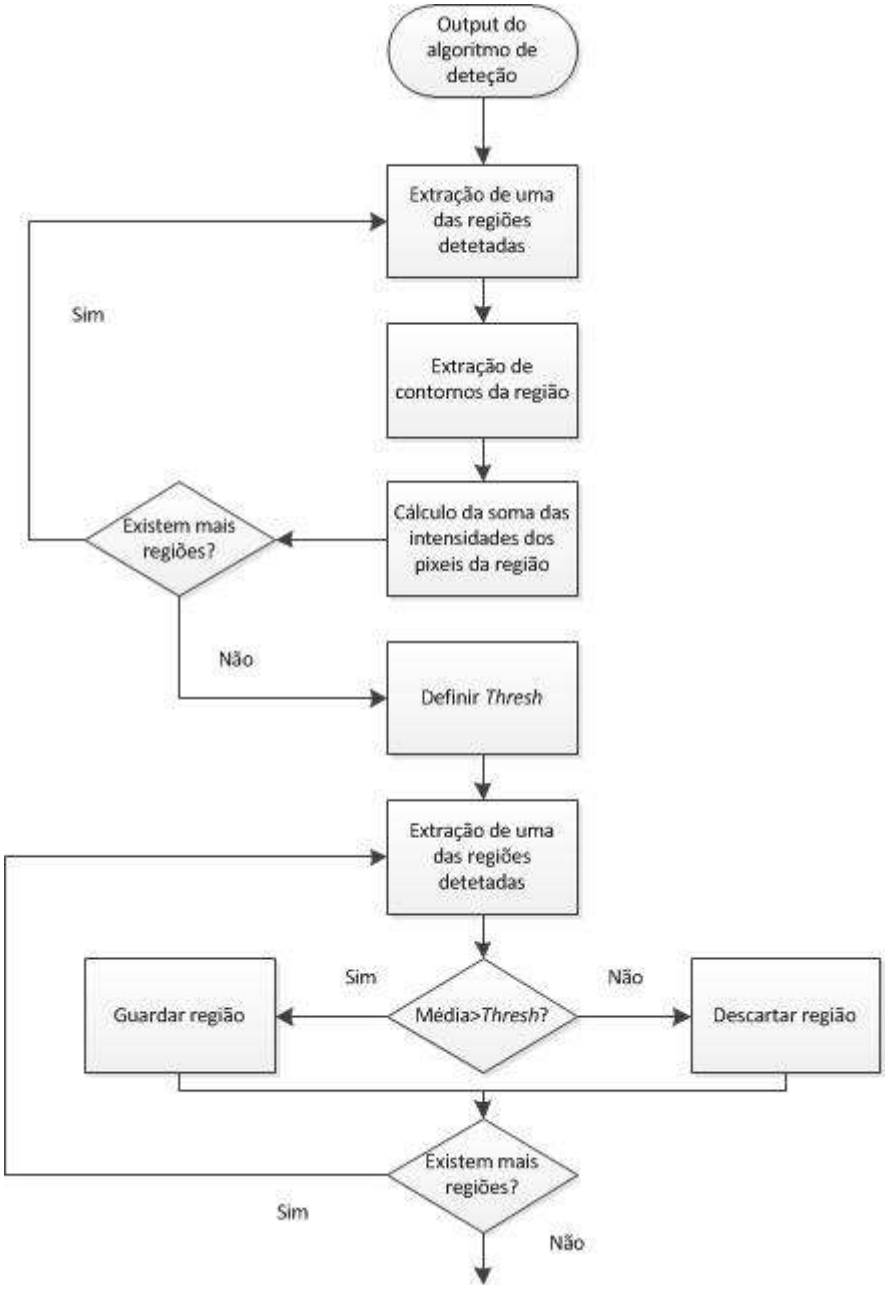


Figura 5.9 – Esquema representativo do bloco "Remoção de Reflexões".

Depois de filtradas as reflexões, as restantes regiões são sujeitas a um método responsável por diminuir o seu tamanho (segundo bloco do filtro). Esta etapa é inspirada no método de remoção de sombras desenvolvido em [Karaman2009], com algumas alterações exigidas pelo facto de serem utilizadas imagens termográficas e não RGB. Consiste num processo iterativo, que tem como objetivo remover a presença de reflexões parciais que possam existir junto da parte inferior das regiões, provocadas pelo espelhamento dos pés e pernas no pavimento, através da identificação da linha de fronteira entre a reflexão e o individuo. Primeiro, é aplicado o detetor de contornos *Canny* existente nas bibliotecas do *OpenCV* a cada região, com o objetivo de extrair a informação de contornos da região, como se pode verificar na figura 5.10.

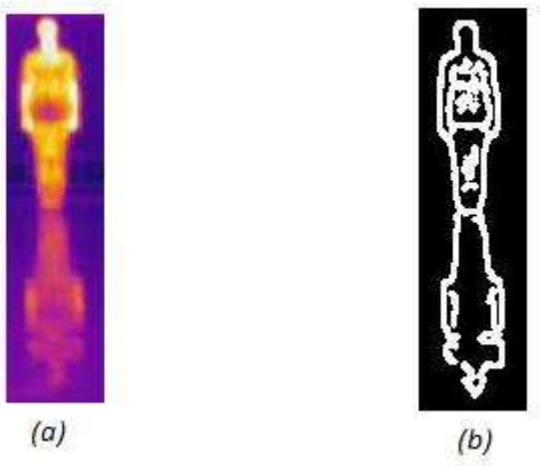


Figura 5.10 – Exemplo de região detetada (a) e extração de informação de contornos aplicada à região detetada, através do detetor *Canny* (b).

O processo consiste em aplicar uma janela deslizante à região de contornos, que a percorre verticalmente, de cima para baixo, de janela em janela, com o objetivo de encontrar a zona fronteiriça entre o indivíduo e a sua reflexão. Este tipo de abordagem segue a lógica de que a zona fronteiriça pode ser identificada através da identificação da janela que apresenta maior semelhança com a janela espelhada correspondente, devido à simetria existente entre o indivíduo e a sua reflexão, como se pode verificar na figura 5.10.

Para se encontrar a fronteira foi desenvolvido um método de espelhamento de cada janela (figura 5.11 b)) de forma a simular a sua reflexão. A região espelhada (figura 5.11 c)) é então comparada com a região original correspondente (figura 5.11 d)), calculando-se a diferença entre a intensidade de cada píxel da região original e o píxel correspondente na região espelhada, de forma a medir a sua diferença (equação 5.3). A esta medida, deu-se o nome Dif. Em cada iteração, a janela atual é espelhada, para posterior comparação. Através de uma análise empírica, definiu-se a altura da janela, hw = 5% do tamanho da região, como se pode verificar na figura 5.11.

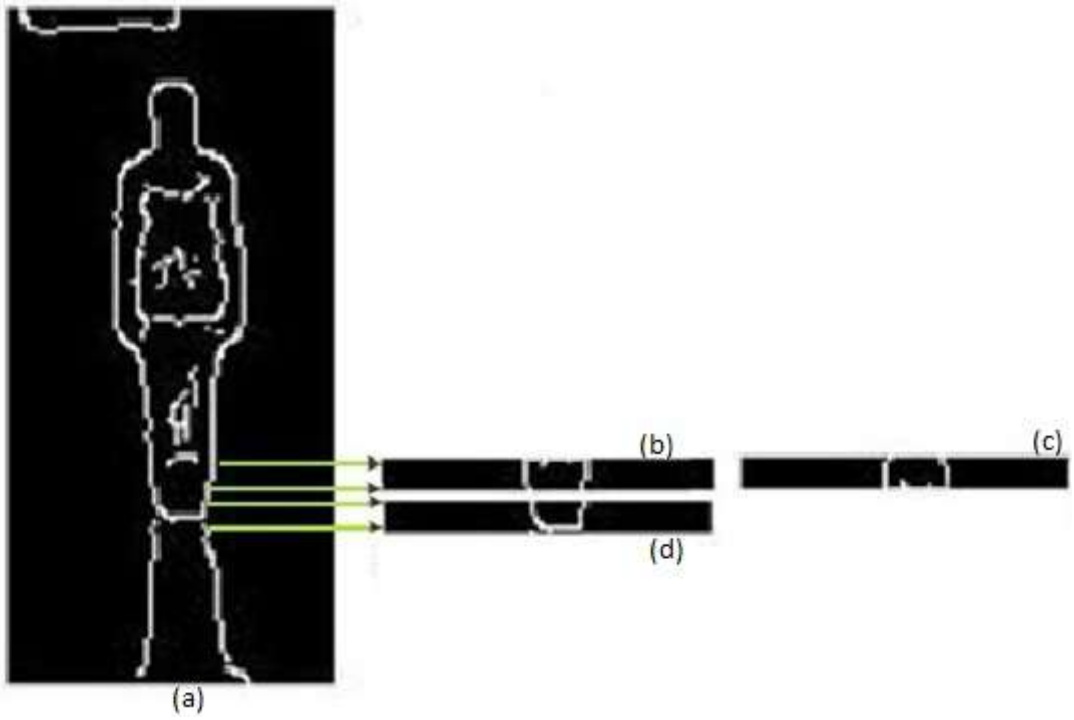


Figura 5.11 – Exemplo de uma região de contornos (a), região original (b), região espelhada (c) e região a ser comparada (d).

De forma a diminuir o número de iterações foi definido como ponto inicial do processo, um quarto da altura do objeto, como se pode visualizar na figura 5.12, diminuindo assim em 25% o tempo de processamento e carga computacional.

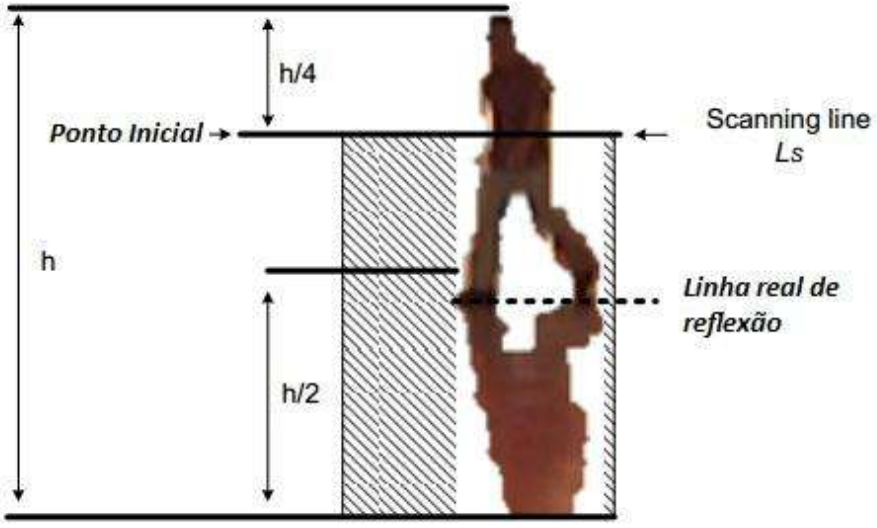


Figura 5.12 – Representação do ponto inicial de processamento (adaptado de [Karaman2009]).

A medida de diferença Dif entre a região espelhada e a região original correspondente é descrita através da fórmula (5.3):

$$Dif = \frac{\sum_{i=1}^{N} \left| Pix_O - Pix_{esp} \right|}{N} \tag{5.3}$$

Onde:

Pix_O – Píxel da região original;

Pix_{esp} – Píxel da região espelhada;

N – Número total de pixéis da janela.

Como já referido, Dif consiste no valor absoluto da diferença entre o valor de cada píxel da região espelhada e o seu complemento na região original correspondente. O número de pixéis em cada janela é N, que pode variar, dependendo da região que está a ser analisada, daí ser necessário normalizar esta medida. Quanto maior for a semelhança entre as regiões, menor é o valor Dif.

É definido dinamicamente um valor de *threshold* $Thresh_{max}$ (5.4), para cada região de contornos (figura 5.11 a)), que consiste na divisão por um factor de 2, do valor da janela que demonstra maior diferença (Dif_{max}) relativamente à região espelhada correspondente, constituindo uma das restrições necessárias para encontrar a linha fronteiriça entre o indivíduo e a sua reflexão. Este valor foi definido empiricamente tendo sido testada a divisão por vários fatores, porém o fator 2 foi o que revelou ser mais adequado, pois verificou-se que não existem zonas onde se encontram reflexões com valores Dif acima deste threshold. Isto significa que o valor Dif correspondente a janelas localizadas em zonas de reflexão nunca é superior a metade da média da janela com maior diferença (Dif_{max}).

$$Thresh_{max} = \frac{Dif_{max}}{2} \tag{5.4}$$

Depois de analisadas todas as janelas, e estarem devidamente definidos os valores Dif_{max} e $Thresh_{max}$, passa-se a uma etapa final de identificação da linha de fronteira entre o individuo e a reflexão. Para tal, são analisados os valores Dif relativos a cada uma das janelas, a partir do último pico acima de Threshmax, até se encontrar a primeira inversão de concavidade que define a janela onde se encontra a linha de fronteira, como está demonstrado na figura 5.13.

Teoricamente a janela com Dif inferior deveria corresponder à zona de fronteira por ser a janela onde existe maior semelhança devido à sua simetria, no entanto, devido ao facto de Dif ser inferior em janelas que incluem partes da reflexão, causado pelo valor da intensidade dos pixéis nestas regiões ser inferior, podem existir picos posteriores com valores Dif inferiores mesmo que a semelhança seja inferior (figura 5.13), logo definir a zona de fronteira não se pode limitar apenas a essa restrição.

Como tal foi definido que, o primeiro pico mínimo abaixo de Threshmax que cumpra com as restrições abaixo descritas, corresponde à janela onde se encontra a região de fronteira entre o indivíduo e a reflexão, como está demonstrado na figura 5.13.

Foram analisadas no mínimo, 25% das janelas existentes.

O ponto de início do processo corresponde à janela relativa ao último pico máximo acima do valor de *threshold* Thresh_{max} Não existe nenhum valor Dif acima de $Thresh_{max}$ nas janelas restantes.

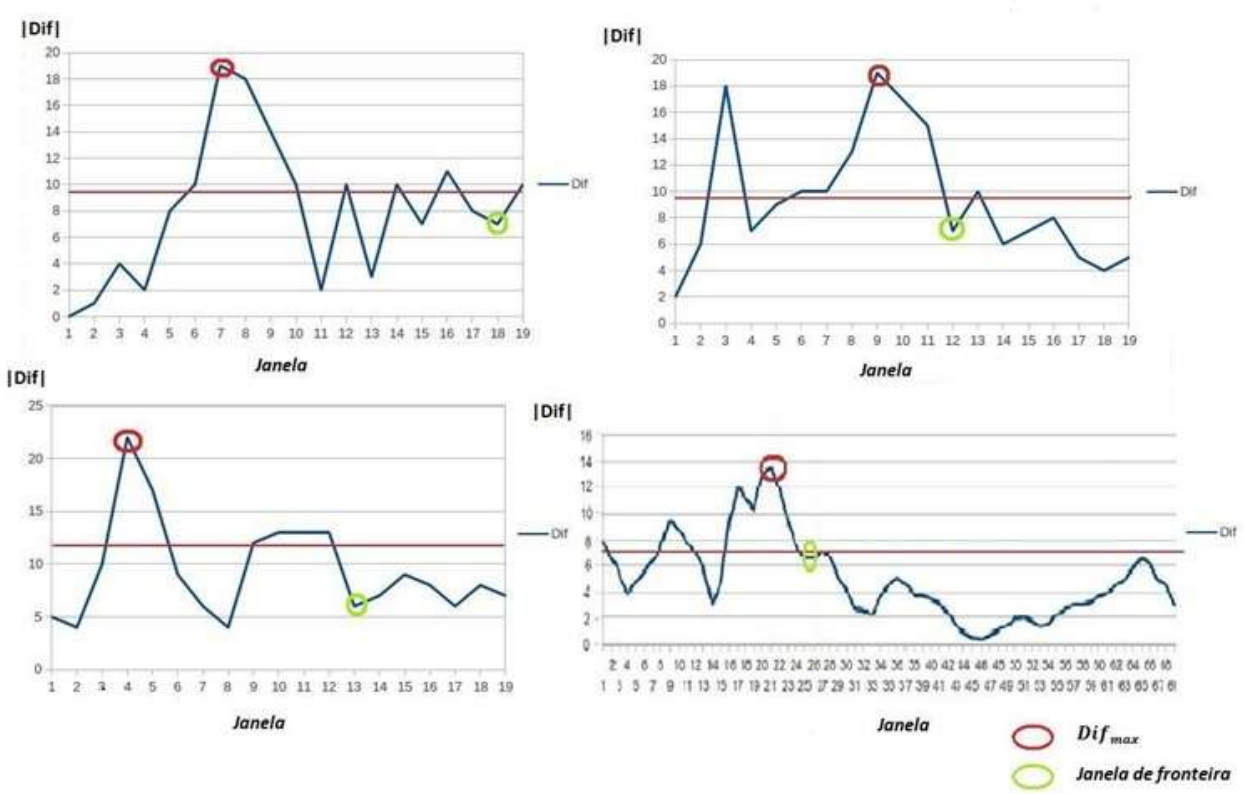


Figura 5.13 – Representação de vários gráficos ilustrativos do valor Dif em cada janela relativamente a quatro regiões detetadas e respetiva localização da linha delimitadora do início da reflexão.

Na figura 5.14 encontra-se representado o esquema ilustrativo do processo iterativo implementado que constitui o segundo bloco (Diminuição do tamanho das regiões).

Figura 5.14 - Esquema representativo do bloco "Diminuição do tamanho das regiões".

5.5 - Testes e Avaliações

Nesta secção encontram-se descritos e analisados os resultados obtidos provenientes dos testes e avaliações aplicados aos vários algoritmos, com o objetivo de avaliar a sua capacidade de deteção quando utilizadas imagens térmicas como fonte de informação, como também verificar se o filtro de pós-processamento desenvolvido provoca o impacto esperado relativamente à performance dos mesmos.

5.5.1 - Resultados HOG

Tabela 5.1 - Resultados em percentagens das métricas *frame-based* relativas ao algoritmo HOG para cada uma das sequências.

	Img_thermal 1	*Img_termal1clean*	*Image_thermal 1* com filtro	*Img_thermal 2*	*Img_thermal 2clean*	*Image_thermal 2* com filtro
False Alarm Rate (FAR) (%)	3,45	3,45	3,45	42,48	20,51	**11,59**
Detection Rate (DR) (%)	82,35	82,35	82,35	**86,67**	82,67	82,43
False Negative Rate (FNR) (%)	17,65	17,65	17,65	**13,33**	17,33	17,57

- *Img_thermal1*:

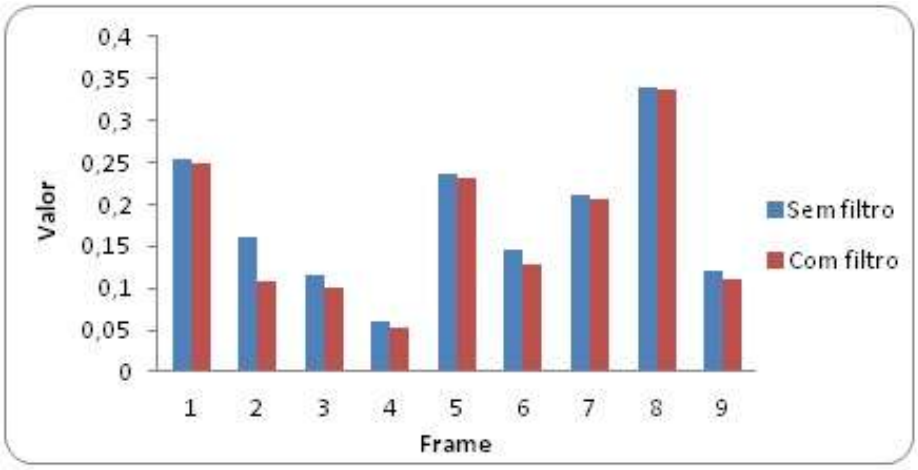


Figura 5.15 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal1*, relativos ao algoritmo HOG.

- *Img_thermal2:*

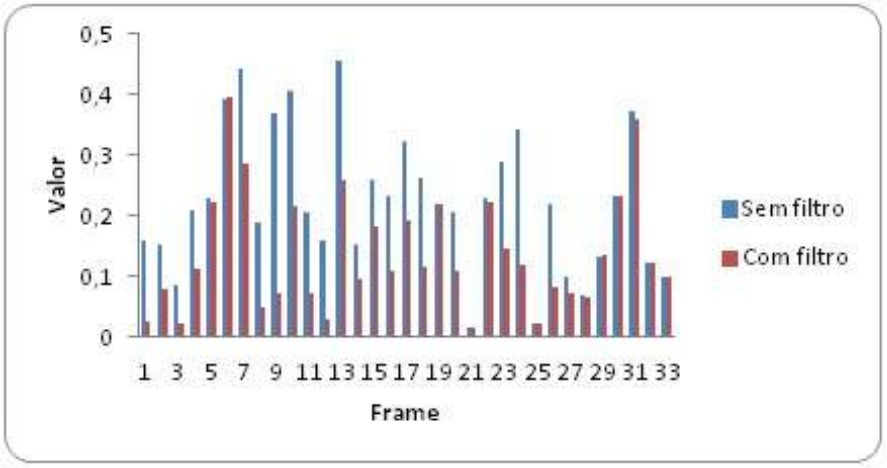


Figura 5.16 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal2*, relativos ao algoritmo HOG.

Através da análise dos resultados presentes na tabela 5.1 e no gráfico da figura 5.15, pode-se concluir que, na sequência *Img_thermal1*, todas as taxas se mantiveram e as áreas das regiões detetadas aproximaram-se de forma pouco significativa dos valores reais, o que significa que de uma forma geral as reflexões existentes foram removidas pelo algoritmo. A razão pela qual não se verifica alterações nas taxas de deteção, falsos positivos, e falsos alarmes, deve-se ao facto de que o próprio algoritmo tem a capacidade de descartar as reflexões existentes nesta sequência, pois as reflexões existentes nas imagens são suaves, devendo-se este aspeto à pouca capacidade de absorção de energia que o pavimento e

paredes possuem. Assim sendo, apenas se verifica o impacto do filtro na diminuição das áreas das regiões onde se encontram pessoas, através da remoção das reflexões parciais aí presentes.

Relativamente à sequência *Img_thermal2*, gráfico da figura 5.16 verifica-se uma diminuição em cerca de 30% no que toca a falsos positivos, à custa de uma diminuição da taxa de deteção na ordem dos 4%, o que revela que o filtro descarta maior parte das reflexões. As áreas das regiões detetadas diminuíram significativamente, aproximando-se bastante das dimensões reais, o que significa que de uma forma geral, as reflexões parciais nelas existentes foram removidas. É de realçar que as imagens da sequência *Img_thermal2* são mais afetadas por reflexões, visto que estas são intensas do que acontece na sequência *Img_thermal1*, devido aos fatores anteriormente explicados, como a existência de vidros ao longo do cenário. Pode-se ainda verificar que ao aplicar o filtro, se consegue obter uma taxa de falsos positivos inferior (-9%) em comparação com *Img_thermal2clean*, onde as reflexões foram manualmente removidas (*Img_thermal2clean*), conservando-se a taxa de deteção.

5.5.2 - Resultados IDIAP

Tabela 5.2 - Resultados das métricas *frame-based* relativas ao algoritmo IDIAP para cada uma das sequências.

	Img_thermal1	*Img_termal 1clean*	*Image_thermal 1* com filtro	*Img_thermal2*	*Img_thermal 2 clean*	*Image_thermal 2* com filtro
False Alarm Rate (FAR) (%)	83,33	73,68	**60,87**	87,14	66,39	**56,99**
Detection Rate (DR) (%)	30,00	**33,33**	30,00	**60,81**	54,05	54,05
False Negative Rate (FNR) (%)	70,00	**66,67**	70,00	**39,19**	45,95	45,95

- *Img_thermal1:*

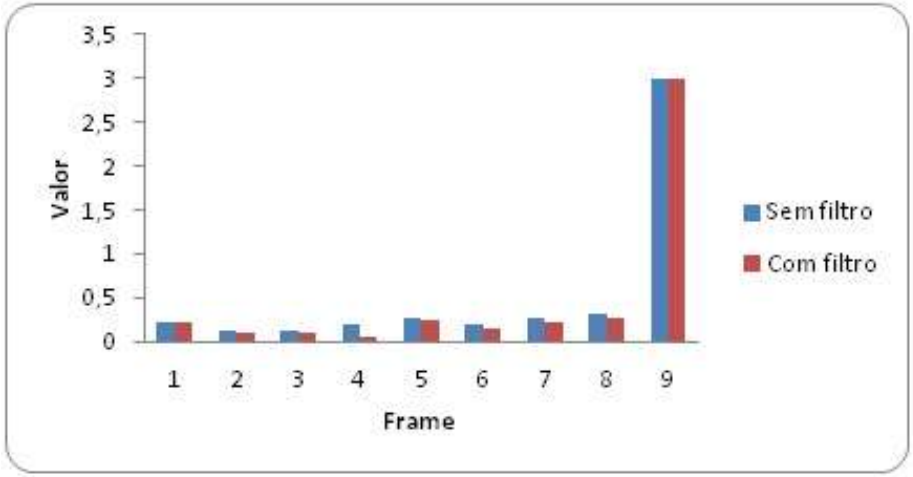


Figura 5.17 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal1*, relativos ao algoritmo IDIAP.

- *Img_thermal2:*

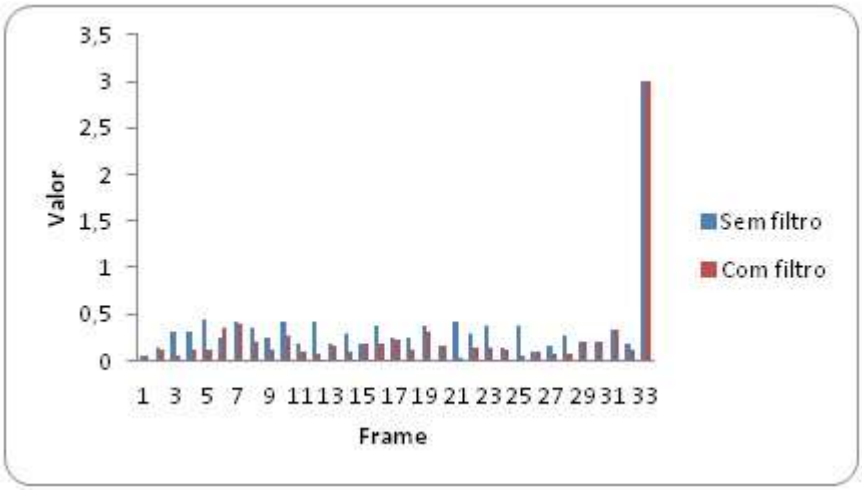


Figura 5.18 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal2*, relativos ao algoritmo IDIAP.

Como já foi discutido e analisado no capítulo 3, este algoritmo atinge as taxas de deteção descritas, à custa de um número elevado de falsos negativos, daí o impacto do filtro neste algoritmo ser acentuado.

Através da análise dos resultados presentes na tabela 5.2 e no gráfico da figura 5.17, pode-se concluir que, na sequência *Img_thermal1*, houve uma diminuição da taxa de falsos positivos em cerca de 23%, sem perdas na capacidade de deteção. As áreas das regiões detetadas diminuíram de uma forma geral, o que significa que as reflexões parciais existentes foram removidas em maior parte das regiões. Verifica-se ainda que a integração do filtro permite atingir, comparativamente com o caso ideal (*Img_thermal1clean*), melhores performances, mais especificamente, uma diminuição da taxa de falsos positivos na ordem dos 13%, à custa de um decréscimo mínimo da taxa de deteção (3%).

Relativamente à sequência *Img_thermal2*, gráfico da figura 5.18, verifica-se uma diminuição em cerca de 31% no que toca a falsos positivos, à custa de uma diminuição da taxa de deteção na ordem dos 6%. As áreas das regiões detetadas diminuíram significativamente, aproximando-se das dimensões reais, o que significa que de uma forma geral, as reflexões parciais nelas existentes foram removidas. Verifica-se ainda que a integração do filtro permite atingir, comparativamente com o caso ideal (*Img_thermal2clean*),uma taxa de falsos positivos inferior na ordem dos 10%, sem perder capacidade de deteção.

5.5.3 - Resultados MUD

Tabela 5.3 - Resultados das métricas *frame-based* relativas ao algoritmo MUD para cada uma das sequências.

	Img_thermal 1	*Img_termal1 clean*	*Image_thermal 1* com filtro	*Img_thermal2*	*Img_thermal2 clean*	*Image_thermal 2* com filtro
False Alarm Rate (FAR) (%)	69,00	**3,85**	69,00	31,52	**0**	14,29
Detection Rate (DR) (%)	**79,41**	73,53	**79,41**	84,00	**85,16**	80,00
False Negative Rate (FNR) (%)	**20,59**	26,47	**20,59**	16,00	**14,86**	20,00

- *Img_thermal1:*

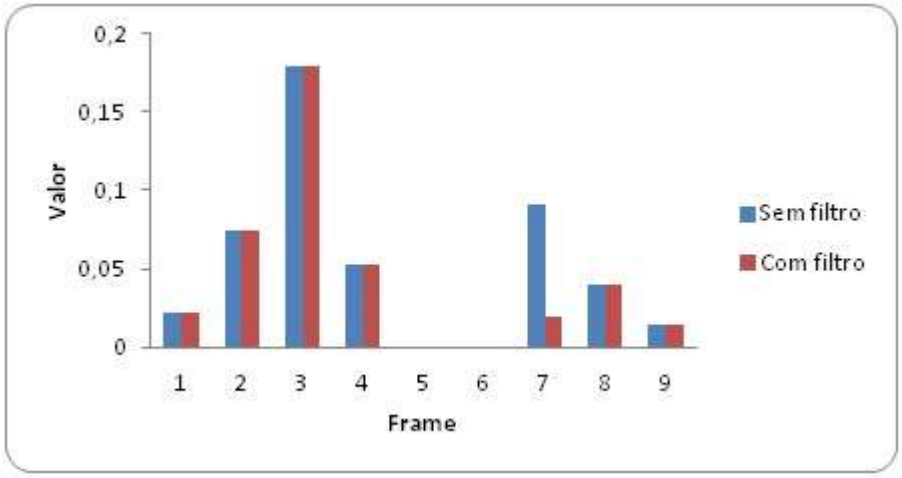


Figura 5.19 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal1*, relativos ao algoritmo MUD.

- *Img_thermal2:*

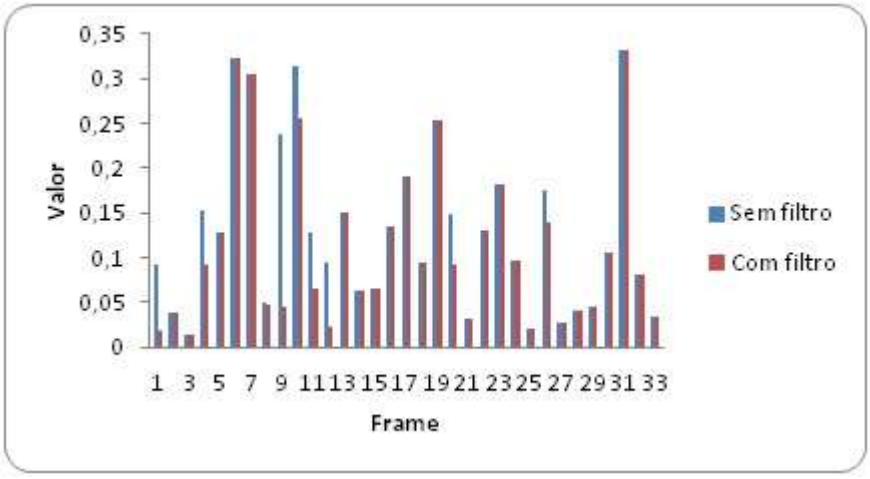


Figura 5.20 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal2*, relativos ao algoritmo MUD.

Através da análise dos resultados presentes na tabela 5.3 e no gráfico da figura 5.19 pode-se concluir que, na sequência *Img_thermal1*, todas as taxas se mantiveram, e as áreas das regiões detetadas são semelhantes com e sem a aplicação do filtro, o que significa que o tamanho das regiões detetadas pelo algoritmo é próximo das dimensões reais, ou seja, as reflexões dos objetos não se encontram incluídas nas regiões detetadas. A razão pela qual não se verifica alterações nas taxas de deteção, falsos positivos, e falsos alarmes, deve-se ao facto de que o próprio algoritmo tem a capacidade de descartar as reflexões existentes nesta sequência, pois as reflexões existentes nas imagens são suaves, devendo-se este aspeto à pouca capacidade de absorção de energia que o pavimento e paredes possuem. Comparativamente com *Img_thermal1clean* verifica-se que a taxa de falsos alarmes é um pouco superior, na ordem dos 3%, no entanto atinge uma taxa de deteção mais elevada, na ordem dos 6%. Isto deve-se ao facto de que ao remover manualmente as reflexões das imagens, diminui-se o ruído, realçando os contornos dos indivíduos e o contraste relativamente ao plano de fundo. Relativamente à sequência *Img_thermal2*, gráfico da figura 5.20, verifica-se uma diminuição em cerca de 27% no que toca a falsos positivos, à custa de uma diminuição da taxa de deteção na ordem dos 4%. As áreas das regiões detetadas diminuíram significativamente em praticamente todas as regiões, aproximando-se das dimensões reais, o que significa que de uma forma geral, as reflexões nelas existentes foram removidas. Comparativamente com o caso ideal (*Img_thermal2clean*), a taxa de falsos alarmes embora seja um pouco maior, aproxima-se do valor desejado, mostrando que a integração do filtro provoca o impacto esperado.

5.5.4 - Resultados HAAR

Tabela 5.4 - Resultados em percentagem das métricas *frame-based* relativas ao algoritmo HAAR para cada uma das sequências.

	Img_thermal1	*Img_termal1 clean*	*Image_thermal 1* com filtro	*Img_thermal2*	*Img_thermal2 clean*	*Image_thermal 2* com filtro
False Alarm Rate (FAR) (%)	57,14	60,00	**55,56**	51,61	**42,86**	44,44
Detection Rate (DR) (%)	**11,76**	**11,76**	8,82	**20,00**	10,67	**20,00**
False Negative Rate (FNR) (%)	**88,24**	**88,24**	91,18	**80,00**	89,33	**80,00**

- *Img_thermal1:*

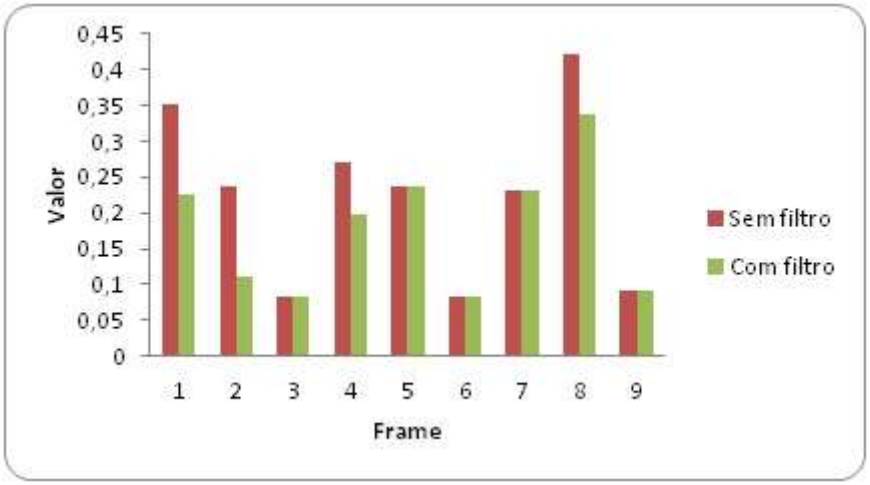


Figura 5.21 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal1*, relativos ao algoritmo HAAR.

- *Img_thermal2:*

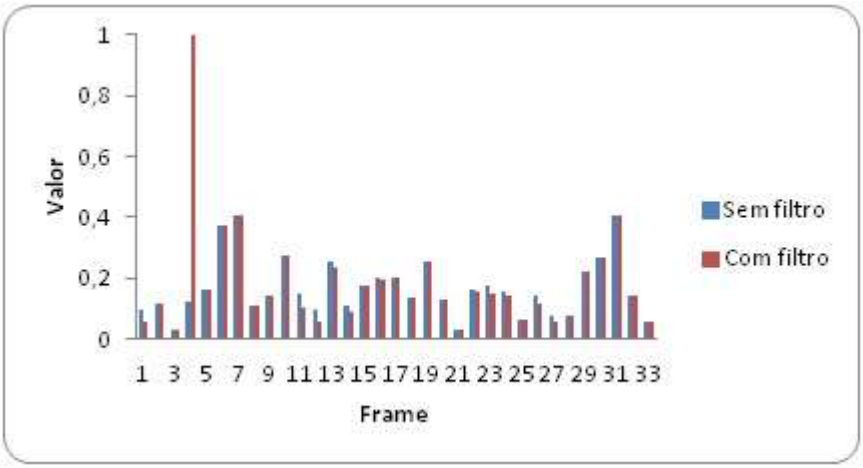


Figura 5.22 - Valor das métricas *partition distance* por *frame* na sequência *Img_thermal2*, relativos ao algoritmo HAAR.

Relativamente ao algoritmo HAAR, embora se consiga verificar o impacto do filtro tanto no que toca a falsos alarmes como nas dimensões das áreas detetadas, optou-se por não aprofundar a análise do impacto do filtro de pós processamento neste algoritmo, visto que a sua performance neste tipo de imagens é fraca, como se pode verificar na tabela 5.4, atingindo taxas de deteção de 11,76% e 20%, relativamente às sequências *Img_thremal1* e *Img_thermal2* respetivamente.

5.6 - Conclusão

Após uma análise das métricas aplicadas aos resultados obtidos conclui-se que, o filtro reduz substancialmente o número de falsos positivos e falsos negativos, devido ao seu poder de remoção das áreas relativas a reflexões, à custa de uma diminuição mínima, ou em alguns casos nula, da taxa de deteção dos algoritmos e reduz com uma precisão aceitável, as reflexões parciais presentes nas áreas detetadas.

Após uma análise cuidada de todos os aspetos realçados anteriormente, é possível concluir que o filtro de pós-processamento, provoca um impacto positivo no resultado final, como era esperado, visto ter como principal objetivo a diminuição de falsos alarmes sem custos relevantes relativamente à capacidade de deteção. Por um lado, diminui substancialmente a taxa de falsos positivos, à custa de uma perda mínima, ou em alguns casos sem perdas, relativamente à taxa de deteção. Este facto demonstra que a etapa de remoção de reflexões baseada em valores de *threshold* de intensidades provoca o impacto esperado. Por outro, reduz significativamente à área das regiões detetadas, aproximando-se assim das dimensões reais das regiões, o que significa que a remoção das reflexões parciais

existentes nas áreas detetadas é efetuada de uma forma geral com sucesso, comprovando assim a possibilidade de adaptar um método de remoção de sombras aplicado em imagens RGB, em imagens térmicas, através da introdução de alterações inspiradas pelas especificidades deste tipo de imagens.

Em alguns casos verifica-se até que com a integração deste filtro, se consegue em algumas ocasiões atingir melhores performances, do que as obtidas em condições ideais, onde as reflexões foram manualmente removidas (*Img_thermal1clean* e *Img_thermal2clean*).

Conclui-se então que este filtro é aplicável a qualquer algoritmo de deteção, sem provocar uma diminuição significativa na sua capacidade de deteção. Como se trata apenas de um filtro de pós-processamento, os impactos significativos só se verificam em aspetos relativos a falsas deteções, com o compromisso de não deteriorar a capacidade de deteção dos algoritmos.

O facto de as dimensões das regiões detetadas não se assemelharem mais às ideais advém do facto de que o filtro apenas reduz as dimensões das regiões em termos de altura e apenas na parte inferior das mesmas, visto que é aí que as reflexões se encontram localizadas.

Capítulo 6

Conclusão e Trabalho Futuro

6.1 - Conclusão

A deteção de pessoas é um tópico em constante evolução, não só devido ao avanço tecnológico que possibilita o desenvolvimento de métodos cada vez mais complexos, como também devido à necessidade de desenvolver *features* mais discriminativas e classificadores cada vez mais robustos e eficientes com menor complexidade computacional.

Nesta dissertação foram estudados, implementados e analisados vários métodos de deteção de pessoas em sequências de vídeo. Foram analisados aspetos relacionados com a arquitetura das abordagens, com as *features* que exploram o impacto que variadas situações desafiantes provocam na sua performance, como por exemplo, variações de postura ou orientação dos indivíduos, de iluminação, de diferentes resoluções de imagem, a capacidade de lidar com situações de ocultação de indivíduos, totais ou parciais.

Foram analisados quatro algoritmos de deteção presentes na literatura, que utilizam *features* e classificadores distintos, com o objetivo de identificar os pontos fortes e limitações que cada um deles possui.

Após a realização desta dissertação conclui-se que talvez perguntas como, "Qual é a melhor abordagem?" não sejam as mais corretas de se fazer, embora, perguntas como, " Quais os principais fatores que perturbam a performance dos algoritmos?", ou até "Qual o ponto de evolução atual desta área de investigação?", já façam mais sentido. De facto torna-se complicado definir qual a melhor abordagem, pois cada uma delas possui os seus prós e contras, que dependem de variados fatores, devendo a solução passar pela integração de diferentes *features* e classificadores robustos e eficientes que possibilitem que o processamento da informação sejam mais rápido e eficiente, sem esquecer o compromisso entre performance e complexidade computacional. Conclui-se então que para se tomar a decisão correta na escolha de uma abordagem, deve-se fazer um estudo do ambiente onde vai ser implementado, podendo-se assim escolher qual a melhor direção a seguir.

Relativamente ao ponto de evolução desta área, encontra-se atualmente bastante evoluída, no entanto ainda existe um longo caminho percorrer até à perfeição, visto ser este o objetivo principal quando se pensa em automatizar a supervisão de espaços.

Os principais contributos desta dissertação podem ser divididos em três pontos principais.

Foram implementadas alterações no funcionamento das métricas *frame based* [Bashir2006] tal como proposto pelo autor, tornando-as orientadas ao objeto e não à *frame*, motivados pelas limitações que estas têm, tornando assim esta métrica mais específica e discriminativa, relativamente aos resultados gerados. Conclui-se que definir o problema de *matching* das regiões como um exercício de otimização combinatória, faz todo o sentido, daí optar por aplicar o método húngaro revelou ser uma escolha acertada.

Outra contribuição está relacionada com o estudo da possibilidade de adaptar um detetor aplicado à deteção de sinais de trânsito, à deteção de pedestres. O trabalho desenvolvido permitiu concluir que, o detetor embora revele várias limitações parece promissora a adaptação a este tipo de situações, se forem implementadas algumas alterações no seu funcionamento, de forma a tornar as características utilizadas mais compatíveis com as especificidades relativas à forma humana. Outro factor que pode contribuir para melhorar a performance deste detetor quando aplicado à deteção de pessoas, está relacionado com a fase de treino, onde se poderia incluir mais amostras negativas, provenientes de várias fontes, de forma a tornar o detetor mais robusto a falsas deteções.

Por fim, o último contributo está relacionado com remoção de reflexões presentes em imagens térmicas. Para tal, foi desenvolvido um filtro de pós-processamento, aplicável a qualquer algoritmo de deteção, que tem como objetivo filtrar regiões detetadas pelos algoritmos que tenham sido provocadas por reflexões existentes nas imagens, e também reduzir o tamanho das regiões restantes, removendo reflexões parciais que aí possam existir. O filtro demonstrou resultados positivos relativamente aos testes efetuados, provando que a sua integração num algoritmo de deteção, provoca o impacto esperado na sua performance, mais especificamente, diminuindo significativamente o número de falsos alarmes gerados, e diminuindo também o tamanho das regiões detetadas corretamente, através da remoção de reflexões parciais que possam existir, sem prejudicar significativamente a capacidade de deteção dos algoritmos.

6.2 - Trabalho Futuro

Relativamente ao detetor utilizado na deteção de sinais de trânsito, existem melhoramentos que podem ser aplicados de forma a tornar o detetor mais eficaz na tarefa de deteção de pedestres. Através do estudo da possibilidade de adaptação do detetor na deteção de pessoas, concluiu-se que a sua maior limitação, para além do elevado tempo despendido na tarefa de deteção, é o número elevado de falsos positivos que revela.

Projetar uma etapa de pré-processamento baseado no LCP de forma análogo à existência para deteção de sinais de trânsito, mais compatível com a deteção de pedestres, ajudaria a reduzir os falsos alarmes, definindo novas estruturas LCP mais adequadas à forma humana. Assim a capacidade de rejeição desta etapa de processamento aumenta, provocando uma maior redução no número de janelas a analisar.

A hipótese de treinar o detetor com um número maior de amostras negativas provenientes de diferentes fontes em diferentes escalas, é também uma opção a tomar, visto que quanto maior for a quantidade de informação fornecida para treinar os classificadores, mais robustos se tornam.

Se o conhecimento prévio do cenário estiver em consideração, pode-se também definir um conjunto de restrições espaciais, que limitem ainda mais a área da imagem a ser processada.

Mais uma direção a seguir no que toca à redução do número de falsos alarmes, passa por integrar um método de subtração de fundo, que permitiria diminuir significativamente a informação a ser processada. A aplicação deste método para além de trazer benefícios em termos de capacidade de deteção, também provocaria uma redução acentuada no número de janelas a ser processadas, resultando assim numa diminuição do tempo de processamento despendido na tarefa de deteção.

O facto de o detetor ter sido implementado em Matlab, também contribui para o elevado tempo de processamento que o algoritmo despende, propondo-se então a implementação deste detetor em C++ (através da biblioteca *OpenCV*), tirando partido de algumas vantagens que programar neste tipo de linguagem tem comparativamente ao Matlab. Algumas das vantagens estão relacionadas com a maior velocidade de execução e o menor consumo de recursos do sistema. De facto o Matlab é compilado em Java que por sua vez é compilado em C. Por outro lado o OpenCV é uma biblioteca escrita em C, sendo assim uma linguagem de nível inferior, que permite velocidades execução significativamente superiores.

Relativamente ao filtro de remoção de reflexões, são propostos alguns melhoramentos futuros. Note-se que a performance do filtro está diretamente relaciona com a qualidade da segmentação efetuada, logo propõe-se o estudo e integração de um método de segmentação

que produza segmentações com maior qualidade que o detetor *Canny*, tornando a localização e remoção das reflexões mais precisa.

Outro melhoramento está relacionado com o facto de o bloco de remoção se reflexões parciais apenas corrigir o tamanho das regiões detetadas na parte inferior da região (devido à remoção de reflexões parciais no pavimento). Propõe-se aplicar um detetor de cabeças a cada região com o objetivo de localizar de uma forma precisa a localização das cabeças dos indivíduos, permitindo assim corrigir o tamanho das regiões detetadas na parte superior das regiões.

Referências

[Andriluka2009] Andriluka, M., S. Roth, et al. (2009). Pictorial structures revisited: People detection and articulated pose estimation. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.

[Bansal2010] Bansal, M., J. Sang-Hack, et al. (2010). A real-time pedestrian detection system based on structure and appearance classification. Robotics and Automation (ICRA), 2010 IEEE International Conference on.

[Bashir2006] Bashir, F.; Porikli, F., Performance Evaluation of Object Detection and Tracking Systems, IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS), June 2006.

[Begard2008] Begard, J., N. Allezard, et al. (2008). Real-time human detection in urban scenes: Local descriptors and classifiers selection with AdaBoost-like algorithms. Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on.

[Beleznai2009] Viertl, N., C. Beleznai, et al. (2009). A pedestrian detection system combining motion detection, spatial grouping and classification. Image and Signal Processing and Analysis, 2009. ISPA 2009. Proceedings of 6th International Symposium on.

[Bertozzi2005] Bertozzi, M., E. Binelli, et al. (2005). Stereo Vision-based approaches for Pedestrian Detection. Computer Vision and Pattern Recognition - Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on.

[Bischof2009] Beleznai, C. and H. Bischof (2009). Fast human detection in crowded scenes by contour integration and local shape estimation. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.

[Borges2012] Borges, P. V. K. (2012). Pedestrian Detection Based on Blob Motion Statistics. Circuits and Systems for Video Technology, IEEE Transactions on **PP**(99): 1-1.

[Castro2010] I. Landesa-Vázquez and J. L. Alba-Castro, The Role of Polarity in Haar-like Features for Face Detection, Accepted for presentation in 20th International Conference on Pattern Recognition, ICPR 2010

[Cardoso2009] J. S. Cardoso, P. Carvalho, L.F. Teixeira and L. Corte-Real, Partition-distance methods for assessing spatial segmentations of images and videos, *Computer Vision and Image Understanding*, July 2009.

[Chen2009] Yu-Ting, C., C. Chu-Song, et al. (2009). Multi-class multi-instance boosting for part-based human detection. Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on.

[Choo2010] Che Yon, C., K. Lee, et al. (2010). Pedestrian detection with image segmentation and virtual mask. Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on.

[Dalal2005] Dalal, N. and B. Triggs (2005). Histograms of oriented gradients for human detection. Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on.

[Dalal2006] Dalal, N., B. Triggs, et al. (2006). Human detection using oriented histograms of flow and appearance. Proceedings of the 9th European conference on Computer Vision - Volume Part II. Graz, Austria, Springer-Verlag**:** 428-441.

[Dollar2007] Dollar, P., T. Zhuowen, et al. (2007). Feature Mining for Image Classification. Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on.

[Dollar2012] Dollar, P., C. Wojek, et al. (2012). "Pedestrian Detection: An Evaluation of the State of the Art." Pattern Analysis and Machine Intelligence, IEEE Transactions on **34**(4): 743-761.

[Enzweiler2010] Enzweiler, M., A. Eigenstetter, et al. (2010). Multi-cue pedestrian classification with partial occlusion handling. Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.

[Fernandez2010] FERNÁNDEZ-CABALLERO, Antonio; CASTILLO, José Carlos; MARTÍNEZ-CANTOS, Javier; MARTINÉZ-TOMÁS, Rafael - Optical flow or image subtraction in human detection from infrared camera on mobile robot. Journal Robotics and Autonomous Systems. Volume 58, Issue 12, December 2010.

[Gao2009] Wei, G., A. Haizhou, et al. (2009). Adaptive Contour Features in oriented granular space for human detection and segmentation. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.

[Gavrila2009] Enzweiler, M. and D. M. Gavrila (2009). Monocular Pedestrian Detection: Survey and Experiments. Pattern Analysis and Machine Intelligence, IEEE Transactions on **31**(12): 2179-2195.

[Grahn2005] Grahn, J. and H. Kjellstron (2005). Using SVM for efficient detection of human motion. Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on.

[Haga2004] Haga, T., K. Sumi, et al. (2004). Human detection in outdoor scene using spatio-temporal motion analysis. Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on.

[Zhang2010] Ran, X., Z. Baochang, et al. (2010). Human detection in images via L1-norm Minimization Learning. Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on.

[Haritaoglu2000] Haritaoglu, I., D. Harwood, et al. (2000). W4: Real-time surveillance of people and their activities. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(8): 809-830.

[Haritaoglu2002] Haritaoglu, I., D. Beymer, et al. (2002). Ghost^{3D}: detecting body posture and parts using stereo. Motion and Video Computing, 2002. Proceedings. Workshop on.

[Holte2011] Holte, M. B., T. B. Moeslund, et al. (2011). 3D human action recognition for multi-view camera systems. 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, 3DIMPVT 2011, May 16, 2011 - May 19, 2011, Hangzhou, China, IEEE Computer Society.

[Junqiu2012] Junqiu, W. and Y. Yagi (2012). Pedestrian detection based on appearance, motion, and shadow information. Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on.

[Kanungo2002] Kanungo, T., D. M. Mount, et al. (2002). "An efficient k-means clustering algorithm: analysis and implementation." Pattern Analysis and Machine Intelligence, IEEE Transactions on **24**(7): 881-892.

[Karaman2009] Karaman, M., L. Goldmann, et al. (2009). Improving object segmentation by reflection detection and removal.

[Krotosky2008] KROTOSKY, Stephen J.; TRIVEDI, Mohan Manubhai - Person surveillance using visual and infrared imagery. IEEE transactions on circuits and systems for video technology, vol. 18, no. 8, August, 2008.

[Kumar2006] KUMAR, Praveen; MITTAL, Ankush, KUMAR, Padam - Fusion of Thermal Infrared and Visible Spectrum Video for Robust Surveillance. Indian Conference on Computer Vision, Graphics & Image Processing - ICVGIP , pp. 528-539, 2006

[Labayrade2002] R. Labayrade, D. Aubert, and J.P Tarel. Real Time Obstacle Detection in Stereo Vision Flat Road Geometry through "V-Disparity" Representation. In Procs. IEEE Intelligent Vehicles Symposium 2002, Paris, France, June 2002

[Leibe2005] Leibe, B., E. Seemann, et al. (2005). Pedestrian detection in crowded scenes. Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on.

[Landesa2010] Landesa, V., x, et al. (2010). Fast real-time multiclass traffic sign detection based on novel shape and texture descriptors. Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on.

[Li2009] Ya-Li, H. and G. K. H. Pang (2009). Human detection in a challenging situation. Image Processing (ICIP), 2009 16th IEEE International Conference on.

[Li2010] Ya-Li, H. and G. K. H. Pang (2010). Human detection in crowded scenes. Image Processing (ICIP), 2010 17th IEEE International Conference on.

[Lin2007] Zhe, L., L. S. Davis, et al. (2007). Hierarchical Part-Template Matching for Human Detection and Segmentation. Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on.

[Lin2010] Zhe, L. and L. S. Davis (2010).Shape-Based Human Detection and Segmentation via Hierarchical Part-Template Matching. Pattern Analysis and Machine Intelligence, IEEE Transactions on **32**(4): 604-618.

[List2004] List, T. and R. B. Fisher (2004). CVML - an XML-based computer vision markup language. Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on.

[Liu2009] Yazhou, L., S. Shiguang, et al. (2009). Granularity-tunable gradients partition (GGP) descriptors for human detection. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.

[Maji2008] Maji, S., A. C. Berg, et al. (2008). Classification using intersection kernel support vector machines is efficient. Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on.

[Moeslund2004] Moeslund, T. B. and E. Granum (2004). Motion capture of articulated chains by applying auxiliary information to the sequential Monte Carlo algorithm. Proceedings of the Fourth IASTED International Conference on Visualization, Imaging, and Image Processing, September 6, 2004 - September 8, 2004, Marbella, Spain, Acta Press.

[Munder2006] Munder, S. and D. M. Gavrila (2006). "An Experimental Study on Pedestrian Classification." Pattern Analysis and Machine Intelligence, IEEE Transactions on **28**(11): 1863-1868.

[Nguyen2009] Nguyen Duc, T., L. Wanqing, et al. (2009). A novel template matching method for human detection. Image Processing (ICIP), 2009 16th IEEE International Conference on.

[Nguyen2010] Duc Thanh, N., L. Wanqing, et al. (2010). Human detection using local shape and Non-Redundant binary patterns. Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on.

[Nguyen2011] Nguyen, D. T., P. Ogunbona, et al. (2011). Human detection with contour-based local motion binary patterns. Image Processing (ICIP), 2011 18th IEEE International Conference on.

[Nguyen2012] Nguyen, Duc Thanh, Human detection from images and videos, Doctor of Philosophy thesis, School of Computer Science and Software Engineering, University of Wollongong, 2012. http://ro.uow.edu.au/theses/3665.

[Odobez2008] Yao, Jian., Odobez, Jean-Marc, "Fast Human Detection from Videos Using Covariance Features", The Eighth International Workshop on Visual Surveillance - VS2008, 2008.

[Ott2009] Ott, P. and M. Everingham (2009). Implicit color segmentation features for pedestrian and object detection. Computer Vision, 2009 IEEE 12th International Conference on.

[Pang2008] Junbiao, P., H. Qingming, et al. (2008). Pedestrian detection via logistic multiple instance boosting. Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on.

[Papageorgiou2000] Papageorgiou, C. and T. Poggio (2000). A Trainable System for Object Detection. Int. J. Comput. Vision **38**(1): 15-33.

[Soga2008] Soga, M., S. Hiratsuka, et al. (2008). Pedestrian Detection for a Near Infrared Imaging System. Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on.

[Su2009] Yuting, S., Q. Rongrong, et al. (2009). Surveillance video sequence segmentation based on moving object detection, Piscataway, NJ, USA, IEEE.

[Tang2009] Shaopeng, T. and S. Goto (2009). Human detection using motion and appearance based feature. Information, Communications and Signal Processing, 2009. ICICS 2009. 7th International Conference on.

[Tang2010] Shaopeng, T. and S. Goto (2010). Histogram of template for human detection. Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on.

[Tosato2010]Tosato, D., M. Farenzena, et al. (2010). Part-based human detection on Riemannian manifolds. Image Processing (ICIP), 2010 17th IEEE International Conference on.

[Tuzel2007] Tuzel, O., F. Porikli, et al. (2007). Human Detection via Classification on Riemannian Manifolds. Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on.

[Tuzel2008] Tuzel, O., F. Porikli, et al. (2008). Pedestrian Detection via Classification on Riemannian Manifolds. Pattern Analysis and Machine Intelligence, IEEE Transactions on **30**(10): 1713-1727.

[Viola2001] Viola, P. and M. Jones (2001). Rapid object detection using a boosted cascade of simple features. Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on.

[Viola2003] Viola, P., M. J. Jones, et al. (2003). Detecting pedestrians using patterns of motion and appearance. Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on.

[Walk2010] Walk, S., N. Majer, et al. (2010). New features and insights for pedestrian detection. Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.

[Wang2009] Xiaoyu, W., T. X. Han, et al. (2009). An HOG-LBP human detector with partial occlusion handling. Computer Vision, 2009 IEEE 12th International Conference on.

[Wojek2009] Wojek, C., S. Walk, et al. (2009). Multi-cue onboard pedestrian detection. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on.

[Xu2010] Ran, X., Z. Baochang, et al. (2010). Cascaded L1-norm Minimization Learning (CLML) classifier for human detection. Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on.

[Yadong2008] Yadong, M., Y. Shuicheng, et al. (2008). Discriminative local binary patterns for human detection in personal album. Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on.

[Yang2010] Jingjing, Y., S. XiaoHong, et al. (2010). Fast Pedestrian Detection Using Slice-Based Motion Analysis. Pervasive Computing Signal Processing and Applications (PCSPA), 2010 First International Conference on.

[Ye2010] Qixiang, Y., J. Jianbin, et al. (2010). Fast pedestrian detection with multi-scale orientation features and two-stage classifiers. Image Processing (ICIP), 2010 17th IEEE International Conference on.

[Yun2007] Ting-Jin, Y., G. Yong-Cai, et al. (2007). Human detection in far-infrared images based on histograms of maximal oriented energy map. Wavelet Analysis and Pattern Recognition, 2007. ICWAPR '07. International Conference on.

[Zeng2010] Chengbin, Z., M. Huadong, et al. (2010). Fast human detection using mi-sVM and a cascade of HOG-LBP features. Image Processing (ICIP), 2010 17th IEEE International Conference on.

[Zhao2003] Zhao, T. and R. Nevatia (2003). Bayesian human segmentation in crowded situations. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 18, 2003 - June 20, 2003, Madison, WI, United states, Institute of Electrical and Electronics Engineers Computer Society.

[Zhu2011] Zhu, T. Deteção e Seguimento de objetos em imagens termográficas: análise experimental de modelos de descrição. INESC Porto, Julho, 2011.