Faculdade de Engenharia da Universidade do Porto

Sistema de visão para detecção de pessoas em movimento

Rui Alberto Tavares da Costa

Mestrado Integrado em Engenharia Informática e Computação

Orientador: Jorge Alves da Silva
Proponente: Acronym – Informação e Tecnologia

28 de Julho de 2010

Sistema de visão para detecção de pessoas em movimento

Rui Alberto Tavares da Costa

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo Júri:
Presidente: João António Correia Lopes

Vogal Externo: António José Ribeiro Neves

Orientador: Jorge Alves da Silva

---

28 de Julho de 2010

Resumo

Na actualidade, a vigilância e o controlo de acessos são factores muito importantes que proporcionam a salvaguarda de locais, objectos e até vidas. Os meios utilizados até agora, como os vigilantes, são dispendiosos e, por vezes, falíveis. Neste sentido a evolução tecnológica tem proporcionado sistemas que podem complementar e auxiliar as funções dos vigilantes a um custo baixo e de um modo eficaz.

Os objectivos da presente tese são o estudo de metodologias de detecção de pessoas e consequente implementação de um sistema capaz de detectar pessoas em movimento em ambiente interior não controlado. O sistema efectua a detecção de pessoas em movimento, através da análise de sequências de imagens provenientes de um vídeo ou de uma câmara que capta imagens em tempo real. A questão do ambiente interior não controlado refere-se ao facto do cenário a filmar ser uma sala onde ocorrem situações imprevisíveis como, por exemplo, variações de iluminação. É de salientar que questões inerentes à câmara, como a calibração e lentes, não são objecto de estudo da presente tese de mestrado.

O sistema desenvolvido é constituído por duas partes principais: uma que faz a subtracção de fundo e outra responsável pela detecção de pessoas. A subtracção de fundo consiste na remoção do cenário que permanece estático, de modo a obter somente todos os alvos em movimento. Por outro lado a detecção de pessoas tem como propósito verificar se tais alvos em movimento correspondem a pessoas. Para a subtração do fundo foram implementados e testados diversos algoritmos: diferença de *frames*, método unimodal e método das misturas Gaussianas. Para a detecção de pessoas foram conjugados os resultados de métodos de análise de cor e detecção de cabeças com base na forma.

Como resultado final, nas imagens do vídeo, os alvos em movimento que foram conotados como pessoas são assinalados através de um rectângulo envolvente, de cor verde. Rectângulos de cor amarela assinalam alvos em movimento quando apenas se encontra uma característica humana, ao passo que os rectângulos de cor vermelha representam alvos que não têm nenhuma característica humana.

Palavras-chave: visão por computador, subtracção de fundo, detecção de pessoas.

Abstract

Nowadays, surveillance and access control are very important in order to provide the protection of places, objects and even lives. The ways used today, for example guards, are expensive and sometimes unreliable. In this sense the technological development has provided other ways that help to meet the same targets of securities, with a lower cost and more efficiently.
The purposes of this thesis are the study of methodologies and consequent implementation of a system for detecting people in motion in an uncontrolled indoor environment. The system performs the detection of moving people through the analysis of image sequences from a video or a camera that captures images at real time. The issue of uncontrolled indoor environment refers to the fact that the scenario is a room where unpredictable situations can occur, such as, lighting variations. It is important to notice that issues related to the camera, such as calibration and lens, are not subject of study on this thesis.
The developed system consists of two main parts: one that makes the background subtraction and another responsible for detecting people. Background subtraction is the removal of the scenario that remains static in the video, in order to get only the moving targets. Furthermore, the people detection is concerned about verifying if such moving targets correspond to people. For background subtraction, several algorithms were implemented and tested: frame difference, unimodal method and Gaussian mixture method. For the identification of individuals the results of methods of colour analysis and detection of heads based on the shape were combined.
As a final result, in the images of the video, the moving targets that have been considered as people are marked by a green bounding rectangle. Yellow rectangles point to moving targets when there is only one human characteristic, whereas the red rectangles represent targets that have no human characteristics.

Keywords: computer vision, background subtraction, people detection.

Agradecimentos

Os agradecimentos vão para os meus pais que se esforçaram para que eu estudasse e conseguisse tirar um curso, e que depositaram a confiança em mim em todo o percurso que fiz.

À Soraia pelo apoio que me deu durante o curso.

Aos meus amigos que me ajudaram nas dificuldades que foram aparecendo.

Ao professor Jorge Alves Silva pela orientação durante a tese de mestrado.

Aos colaboradores da Acronym – I.T. por me ajudarem e pelo voto de confiança em especial o Engenheiro Carlos Silva pela orientação e conhecimentos transmitidos.

Rui Alberto Tavares da Costa

Conteúdo

Lista de Figuras

Lista de Tabelas

Abreviaturas e Símbolos

BLOB – *Binary Large Object*
RGB – Red, Green, Blue - espaço de cores
HSV – Hue, Saturation, Value - espaço de cores
YCrCb – espaço de cores, Y representa a luminância, Cr é a diferença entre a componente vermelha e a luminância e Cb é a diferença entre a componente azul e a luminância
YUV – espaço de cores em que Y representa a luminância enquanto U e V representam a cor

Capítulo 1

Introdução

No presente capítulo é apresentada uma breve descrição do projecto em questão, fornecendo informação acerca do tema e das razões que conduziram à sua realização. É ainda descrita a estrutura do relatório.

A presente dissertação demonstra o estudo efectuado durante o desenvolvimento de um sistema de visão por computador que tem como principal funcionalidade a detecção de pessoas. Neste sentido, o sistema terá que discriminar as regiões das imagens que correspondem a pessoas, sendo tais imagens provenientes de um vídeo ou de uma câmara que capta as imagens em tempo real. O ambiente onde o sistema será utilizado será um ambiente não controlado. Designa-se por ambiente não controlado um ambiente cujas condições são imprevisíveis, como por exemplo, ambientes onde há variação de iluminação, onde pode ocorrer a existência de objectos que se mexem, para além de pessoas, entre outras condições.

1.1 Motivação

A visão por computador é uma área científica onde é pretendido simular a visão humana com o recurso a computadores e a dispositivos de aquisição de imagens.

Muitas aplicações têm como base a visão por computador. Entre elas pode-se referir a vigilância, detecção de anomalias e reconhecimento de gestos. A vigilância, que é a área da presente aplicação é de grande importância para garantir a segurança e controlo de acessos em locais. A realização de vigilância, através de vigilantes humanos, torna-se dispendiosa, abrindo a oportunidade de criar métodos automáticos que executem a mesma tarefa de modo mais económico. O facto de envolver pessoas pode atribuir questões inerentes à imperfeição humana, como o cansaço e consequente falta de qualidade na vigilância. A empresa proponente deste projecto, Acronym – Informação e Tecnologia, tem como um dos seus alvos de negócio o controlo de acessos, pelo que o estudo e desenvolvimento de técnicas e sistemas de vigilância baseados em visão por computador é de extrema importância para a sua implantação no mercado.

1.2 Objectivos

O principal objectivo deste projecto de dissertação é a criação de um sistema capaz de detectar pessoas em movimento numa sequência de imagens captadas em tempo real. Convém notar que a detecção de pessoas terá que ser efectuada à medida que o sistema recebe as imagens, ou seja, em tempo real. Este sistema terá de ser implementado tendo em conta que a aplicação final será inserida num espaço interior cujas condições são imprevisíveis, por isso denominado "ambiente não controlado". Desta forma, é necessário que o sistema seja adaptativo no sentido de detectar pessoas independentemente das variações ocorridas no ambiente envolvente.

É de referir que o sistema terá que ser robusto para que seja o mais impermeável possível face a problemas e a situações indesejadas que possam ocorrer, como por exemplo as alterações de iluminação ou movimentos que não são de interesse e que possam prejudicar a detecção de pessoas.

1.3 Sistema de visão para detecção de pessoas

A detecção de pessoas é concretizada em duas fases: a **subtracção de fundo** e a **detecção de pessoas**. O presente relatório está igualmente dividido de acordo com estas duas fases.

De modo sucinto, a subtracção de fundo consiste na análise de uma sequência de imagens de vídeo e na identificação de zonas das imagens que representam objectos em movimento, ou seja, a aplicação, ao receber sucessivas imagens, terá que discriminar os pixeis que representam objectos em movimento dos pixeis que representam o fundo estático. O fundo estático representa o conjunto de objectos que ao longo da sequência de imagens, permanecem imóveis. Para a implementação desta fase foram testados três métodos: método da diferença de quadros (*frames*), método unimodal e método das misturas gaussianas.

Por outro lado, a detecção de pessoas propriamente dita, refere-se à identificação de zonas das imagens que representam pessoas em movimento. Esta fase é feita através da análise de cor e forma das regiões das imagens que apresentam movimento, que foram previamente identificadas. Juntando estas duas fases é possível implementar a detecção de pessoas em ambiente não controlado.

Um importante factor a ter em conta é o tempo de processamento que o sistema leva a detectar pessoas no vídeo capturado. O sistema será utilizado em tempo real, ou seja, a sequência de imagens que entram, terão de ser imediatamente processadas de modo a mostrar o resultado final da detecção de pessoas. Para tal, os métodos utilizados têm de evitar um elevado tempo de processamento, algo que foi tomado em conta para a escolha dos métodos.

1.3.1. Subtracção de fundo

Os pixeis que representam objectos imóveis são aqueles que durante a sucessão de imagens não sofreram variações de intensidade, pertencendo assim ao fundo estático. Desta forma, todos os pixeis presentes nas mesmas coordenadas (x;y) cujos valores de intensidade variam ao longo do vídeo, representam objectos em movimento. Separando os pixeis que representam objectos estáticos dos que representam objectos em movimento, é concretizada a subtracção de fundo. As Figuras 16 e 17 mostram duas imagens da mesma sequência onde estão assinalados dois pixeis, um (A) em que não há movimento, outro (B) cuja intensidade sofreu variação devido ao aparecimento de uma mão.

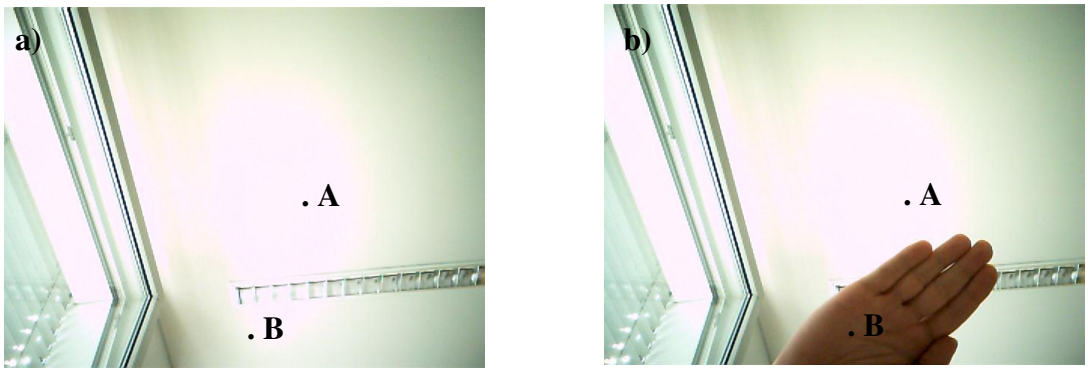

Figura 1 – Exemplos do cenário: a) Imagem do tecto; b) Imagem do tecto com uma mão.

Tal como é ilustrado nas Figuras 1a e 1b, o pixel **A** localizado na mesma coordenada (x,y) de ambas as imagens não sofreu variação de intensidade de uma imagem para a outra, ao passo que no pixel **B** não acontece o mesmo pois da tonalidade branca do tecto passou a conter o tom de pele da mão, havendo assim alteração da intensidade. Neste sentido o pixel A é um pixel pertencente ao fundo, ou seja, aponta para um objecto estático e o pixel B representa um alvo em movimento, que neste caso é a mão.

O resultado da subtracção de fundo é representado por uma imagem binária em que a tonalidade branca (255) representa o movimento e a cor preta (0) o fundo estático. Diversos algoritmos executam a subtracção de fundo, parte dos quais são apresentados no Capítulo 2 referente ao estado da arte. No entanto há várias questões que se tem que ter em conta de modo a efectuar uma subtracção de fundo de um modo o mais eficaz possível. Um dos grandes problemas é a variação de iluminação durante a captura de imagens. A variação de iluminação traz uma mudança de intensidade nos pixeis sem que haja uma verdadeira ocorrência de movimento. É necessário que o sistema seja robusto de modo a evitar que a mudança de iluminação seja considerada como movimento. Para além das situações de mudança de iluminação ainda há outros aspectos que podem ser evitados, como por exemplo os movimentos oscilatórios de cortinados.

No presente projecto, a questão da subtracção de fundo foi abordada começando por implementar em primeiro lugar a subtracção directa de *frames*, seguida do método unimodal e finalmente, de modo a atingir o objectivo, o método das misturas Gaussianas.

1.3.2. Detecção de pessoas

Relativamente à detecção de pessoas, igualmente referida no estado da arte, há vários parâmetros que é necessário definir e restringir para que se possam obter resultados satisfatórios. A posição da câmara relativamente ao cenário a filmar, isto é, frontal ou em perspectiva, o próprio cenário onde se vai captar as imagens, por exemplo, uma sala grande ou um corredor pequeno, e a iluminação são alguns dos factores que influenciam decisivamente nos resultados finais da detecção de pessoas. Como se pretende que o sistema em questão esteja inserido num ambiente interior não controlado, as questões acima referidas terão de ser tidas em conta, tornando o sistema flexível e adaptativo face a eventuais condições que dificultem o funcionamento do sistema.

Depois de obter os alvos em movimento é efectuado o processamento necessário para verificar se o tal alvo é uma pessoa ou não. Para tal, os métodos a implementar devem ser robustos de modo a que as condições imprevisíveis do ambiente não sejam prejudiciais para o

resultado final. Por exemplo, quando há pouca iluminação pode levar a que a subtracção de fundo seja mal executada. Consequentemente a detecção de pessoas é prejudicada pois pode não ter informação suficiente para a verificar se o objecto em movimento é realmente uma pessoa.

Por outro lado, há algumas características que, mesmo em condições adversas, permitem a identificação de pessoas. O ser humano é um ser vivo que se desloca em posição vertical, e o facto de ser bípede coloca a cabeça no topo do corpo. Ao receber a zona da imagem referente ao alvo em movimento e analisando-a de cima para baixo, a cabeça, normalmente, é a primeira parte do corpo que aparece, salvo situações em que a pessoa está com o(s) braço(s) levantado(s). Outra característica é que a cabeça tem normalmente a forma elíptica. Neste sentido todos os alvos em movimento que obedecem às descrições anteriores, são conotados como sendo pessoas. Os métodos da análise de cor e detecção de cabeças foram os escolhidos para serem implementados e ambos serão descritos no Capítulo 3.

A ocorrência de oclusão de uma pessoa em movimento pode ser factor para que ela não seja identificada, por exemplo, quando ocorre oclusão parcial ou total da cabeça. No entanto é impossível prever todas as situações e abrangê-las.

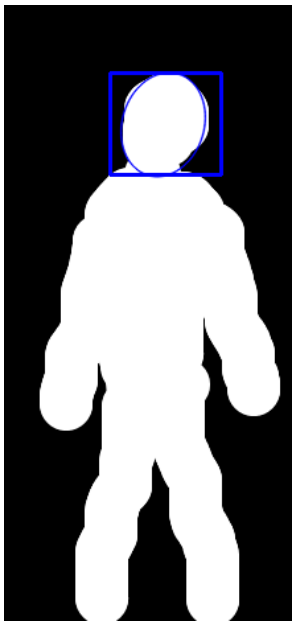

Figura 2 – Detecção de cabeças após binarização da imagem.

A Figura 2 representa a imagem de uma pessoa proveniente da fase de subtracção de fundo que contém na zona da cabeça um quadrado e uma elipse para delinear a forma elíptica da cabeça. Em condições de ambiente normais, isto é, que não prejudiquem o resultado final, a imagem é uma boa representação do resultado produzido pela detecção de pessoas.

1.4 Estrutura do relatório

O presente documento encontra-se dividido em seis capítulos onde estão detalhados todos os factores envolvidos no desenvolvimento deste projecto. Existe ainda um conjunto de anexos que fornecem complementaridade aos temas que vão sendo abordados.

Após o presente capítulo introdutório, no Capítulo 2, Estado da arte, são analisados métodos existentes que possam ser úteis para o trabalho desenvolvido. No Capítulo 3, implementação, encontram-se descritas as tecnologias utilizadas assim como o modo como foram implementados os métodos. Depois, o capítulo referente aos resultados experimentais descreve os resultados obtidos e finalmente, no Capítulo 5 estão descritas as conclusões finais sobre o trabalho, limitações deste e alguns caminhos futuros de modo a melhorar o sistema em questão.

Capítulo 2

Estado da arte

Actualmente há muitas aplicações que necessitam das funcionalidades que a visão por computador disponibiliza, de modo a melhorar as suas capacidades. Algumas dessas aplicações são a vigilância para controlo de acessos e segurança (por exemplo, nos terminais de aeroportos para a detecção e seguimento de potenciais terroristas) ou a detecção de situações de pré-colisão entre veículos.

Uma das aplicações que envolve visão por computador, e que é o tema central focado na presente dissertação, é a detecção de pessoas numa sala em ambiente não controlado. A detecção de pessoas tem como base o processamento e interpretação de sequência de vídeo, de modo a identificar regiões que representam pessoas.

Neste capítulo optou-se por dividir a análise do estado da arte em duas áreas, dada a complexidade que cada uma representa. Assim será descrito, em primeira instância, o estado da arte da **subtracção do fundo** cujo objectivo é obter somente os objectos em movimento, e terminada esta análise, será descrito o estado da arte acerca de métodos que procedem à **detecção da presença humana**.

2.1 Subtracção do fundo

A subtracção do fundo envolve a separação entre os pixeis que representam objectos/pessoas em movimento e os pixeis que representam objectos estáticos de uma imagem. Em geral, o processo de separação é feito por análise da variação de intensidade de cada pixel ao longo da sequência de imagens, sendo o factor indicativo da ocorrência de movimento a existência ou ausência de variação.

As regiões que representam alvos em movimento serão objecto de posterior processamento, restando a região estática, ou fundo, que será desprezada. Desta forma, é possível reduzir a quantidade de informação a processar e é evitado o processamento de informação susceptível de conduzir a falhas no sistema, como por exemplo desenhos numa parede com o formato de pessoas.

A subtracção de fundo é de grande importância para o sistema em questão, no sentido que se esta não for robusta, o passo seguinte, a detecção de pessoas pode falhar.

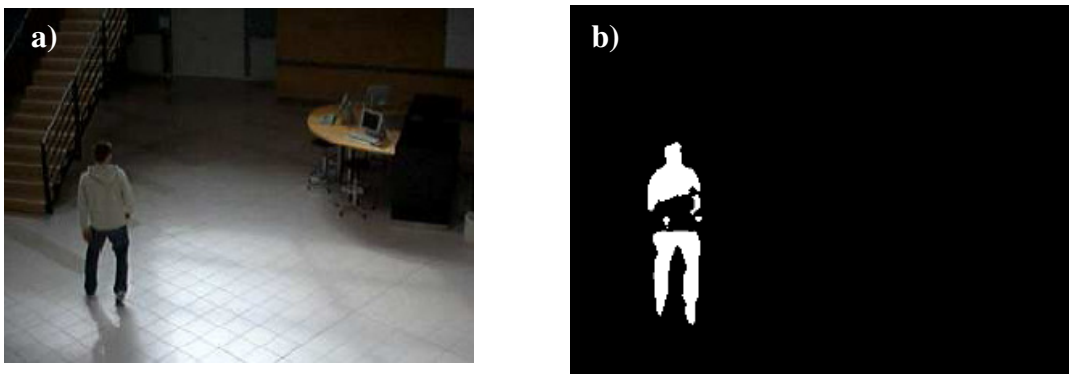

Figura 3 – Exemplo de execução de subtracção de fundo[2]: a) imagem original; b) imagem resultante da subtracção de fundo efectuada na imagem da esquerda.

A subtracção do fundo é uma fase que requer cuidado, uma vez que, como é evidenciado na Figura 3b, pode não devolver o objecto de interesse de forma correcta, neste caso, a zona inferior do tronco é considerada como fundo ao contrário do restante corpo. Desta forma é crucial a escolha do método que produza melhores resultados. Os métodos que processam a subtracção do fundo divergem em termos de complexidade e resultados. Nesta secção serão analisados cinco métodos: diferença de *frames* [1], unimodal [2-4], *kernel estimation* [1,2,5], *codebook* [2,4,6], e misturas Gaussianas [1,7-10].

2.1.1 Método de diferença de *frames*

O presente método consiste em subtrair directamente os valores das intensidades de cada pixel em duas imagens consecutivas. No caso do resultado dessa subtracção ser maior que um limiar (*threshold*), então é considerado que esse pixel corresponde a um alvo em movimento, pois houve uma variação significativa do valor de intensidade. Por outro lado, caso esteja abaixo ou igual ao mesmo limiar, significa que a variação da intensidade não foi demasiado expressiva para que o pixel seja considerado como pertencente a um alvo em movimento. A seguinte condição explica o que foi referido.

$$\text{Se } |frame_i - frame_{i-1}| > \text{limiar , então houve movimento}$$

Este método é de simples execução, uma vez que representa a subtracção directa de valores, mas não é robusto, sendo muito susceptível a pequenas oscilações e alterações de iluminação. O limiar escolhido também influencia o funcionamento do sistema, uma vez que um valor muito alto pode levar a que alguns objectos em movimento sejam ignorados, sendo considerados como fundo, ao passo que um valor muito baixo conduz a que pequenas variações de intensidade que não são devidas a movimento sejam consideradas como movimento.

2.1.2 Método unimodal

O método unimodal é utilizado para a subtracção do fundo e tem como base o princípio que os valores das intensidades de cada pixel do fundo, numa sequência de imagens, seguem uma distribuição unimodal, como por exemplo, uma distribuição Gaussiana.

A ideia base do método pressupõe que as intensidades de cada pixel do fundo são aproximadamente constantes, sofrendo ligeiras variações devido a questões de iluminação ou até inerentes à câmara. Considera-se que um pixel pertence ao fundo estático se a sua intensidade, ao longo da sequência de imagens, estiver compreendida numa gama de valores que é função da média e do desvio-padrão das intensidades: [Média – Desvio-padrão ; Média + Desvio-padrão]. Qualquer pixel cujos valores de intensidades não seguem a distribuição são considerados como pertencentes a alvos em movimento. Deste modo procede-se à separação de fundo, obtendo os pixeis que seguem uma função unimodal como pertencentes ao fundo estático e os restantes pixeis pertencentes a alvos em movimento [2,3]. Por outras palavras, a função da distribuição Gaussiana representa as intensidades que indicam se um pixel é pertencente ao fundo estático, caso as intensidades deste último coincida com as da distribuição.

As intensidades próximas à moda (valor de intensidade mais repetido) são também frequentes embora um pouco menos, e à medida que se vai afastando da moda, a sua ocorrência é cada vez menor [4]. Quando o valor da intensidade de um pixel sair daquela gama, considera-se que esse pixel não pertence ao fundo estático mas a um alvo em movimento.

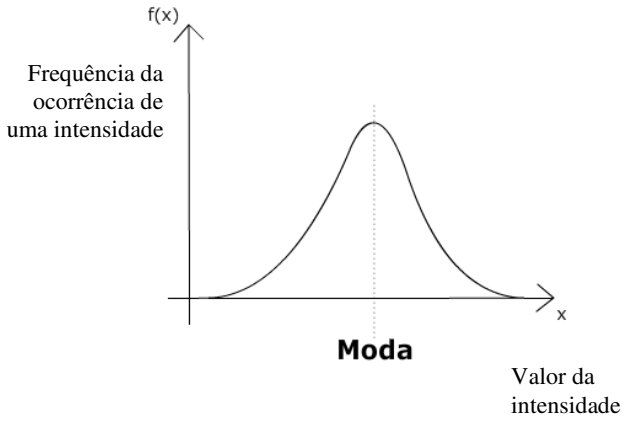

Figura 4 - Distribuição unimodal que representa os valores de intensidade de um pixel permitidos para o fundo estático.

Este método permite efectuar a extracção do fundo, mas requer que este último seja extremamente estático, o que não acontece quando há grandes variações de iluminação ou outras situações que ocorrem em ambiente não controlado, por exemplo, o movimento de cortinados provocado pelo vento. Em suma, este método não é bom para ambientes não controlados [4].

2.1.3 Método de *Kernel estimation*

O método de *Kernel estimation* baseia-se na estimação da função de densidade de probabilidade para o valor de intensidade de um pixel. A informação da função de densidade de probabilidade é formada a partir do histograma das imagens mais recentes. Todos os pixeis que seguem e reforçam essa mesma função de densidade de probabilidade, ou seja, todos os valores de intensidades de cada pixel que são similares aos que já estão previamente guardados, são designados como pertencentes ao fundo. Por outro lado, sempre que a intensidade do pixel é diferente do que se tem obtido, (não segue a função densidade de probabilidade) significa que o pixel representa um objecto em movimento.

No entanto, o cálculo da função de densidade de probabilidade, a partir do histograma de intensidades para cada pixel, leva a um grande consumo de memória e de tempo [1,2]. Mesmo assim, este método consegue ser mais eficaz que o método anterior, em ambientes não controlados, devido à robustez que apresenta face a variações de iluminação, uma vez que dá mais relevo à informação temporalmente mais recente [5].

2.1.4 Método do *Codebook*

O método designado por *codebook* consiste no cálculo das intensidades permitidas para que cada pixel seja considerado como fundo. Esse processo é efectuado mediante o cálculo do raio e do comprimento de um ou mais cilindros (cada cilindro é designado por *codeword*), cujo volume engloba todas as intensidades referentes a intensidades do fundo estático. Um *codebook* é o conjunto das *codewords* de cada pixel. Uma *codeword* representa um volume (cilindro) dentro do cubo do espaço de cores RGB, em que as intensidades do pixel incluídas dentro desse volume são as permitidas para que o mesmo pixel seja considerado como fundo [2,4]. Por outras palavras, aquando da entrada do valor da intensidade de um pixel, é analisado o seu *codebook* e caso os valores de entrada estejam inseridos dentro de um dos *codewords*, então o pixel é considerado como pertencente ao fundo, caso contrário é considerado como pertencente a um alvo em movimento. [4]

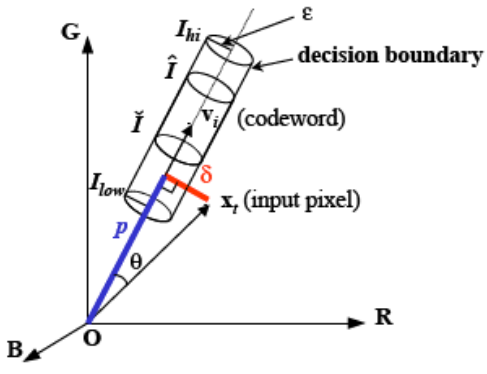

Figura 5 - Representação do *codebook* que contém as intensidades permitidas para que o pixel seja considerado como fundo estático[2].

Ao contrário do que acontece aos métodos anteriormente analisados, o método do *codebook* não carece de cálculos probabilísticos o que reduz o tempo de processamento. No entanto, este método exige que haja um período de treino de modo a construir as diversas *codewords* para cada pixel [4].

2.1.5 Método das misturas Gaussianas

O método das misturas Gaussianas tem como base o método unimodal, mas, em vez de considerar a existência de uma única distribuição Gaussiana, considera que podem existir várias (método multimodal) podendo o seu número, *k*, variar de aplicação para aplicação [6,7].

A ideia base das misturas Gaussianas parte do princípio que apenas uma distribuição Gaussiana é incapaz de se adaptar a variações que não representam movimento, por exemplo, alteração de iluminação ou outras pequenas oscilações de objectos, como cortinados ou árvores. Desta forma, o recurso a mais distribuições Gaussianas é apropriado, uma vez que as pequenas variações iriam incidir sobre essas mesmas distribuições, ficando assim conotadas como pertencentes ao fundo estático e não a alvos em movimento, ao contrário do que acontece no método unimodal.

O facto de se considerar a existência de mais do que uma moda, dependendo do valor *k* a utilizar, permite que cada pixel possa ser classificado como pertencente a uma das distribuições Gaussianas. O método vai actualizando os pesos dados a cada distribuição Gaussiana à medida que vai recebendo cada imagem do vídeo, e de modo similar ao método unimodal, cada pixel tende a seguir uma das distribuições. O facto do valor de intensidade de cada pixel poder seguir várias distribuições torna possível uma rápida adaptação a mudanças e, consequentemente, uma melhor subtracção do fundo estático.

A atribuição de pesos às distribuições, permite que o método seja adaptativo face a transições que ocorram durante a sequência de imagens. Um objecto que está em movimento é representado por pixeis que não pertencem a nenhuma das distribuições Gaussianas. Mas à medida que o objecto se torna estático o peso de uma intensidade de um pixel desse objecto começará a crescer, uma vez que a frequência com que essa intensidade ocorre, cresce ao longo do tempo, até que chega a um ponto em que esse pixel é conotado como pertencente ao fundo estático. As misturas Gaussianas permitem a adaptação a mudanças do estado de um objecto muito facilmente, o que é benéfico para o sistema que foi desenvolvido, ou seja, é robusto face a ambientes não controlados [8].

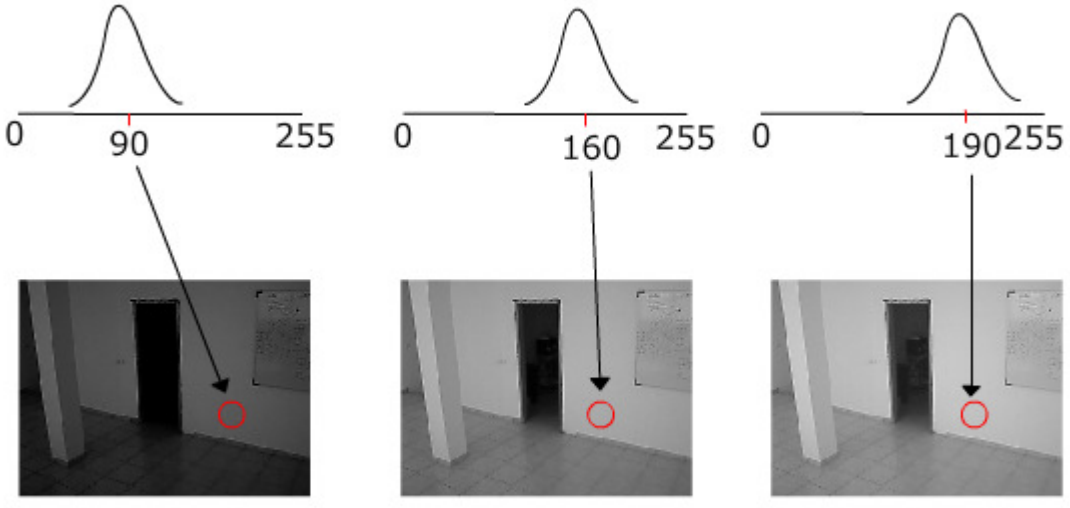

Figura 6 - Representação de misturas Gaussianas num pixel ao longo de 3 imagens de um vídeo.

A Figura 6 apresenta o mesmo pixel ao longo do tempo, estando este assinalado nas três imagens com um círculo vermelho. O valor 90 é a média da distribuição Gaussiana do pixel quando a sala está mais escura, 160 corresponde ao mesmo valor na imagem do meio, e 190 é a média da distribuição Gaussiana do pixel quando a sala está mais iluminada. Tal como foi referido, o método unimodal não é robusto face a alterações de iluminação, uma vez que são necessárias, no caso da Figura 5, pelo menos 3 distribuições Gaussianas para o mesmo pixel, para que este não seja conotado como sendo pertencente a um alvo em movimento. Como o método unimodal só contém uma distribuição Gaussiana, as mudanças de intensidade provocadas pelas alterações de iluminação, não seriam abrangidas pela única distribuição.

A metade de cima da Figura 6 mostra as distribuições Gaussianas do pixel em questão, evidenciando a diferença entre elas, provocadas pela alteração de iluminação da sala (médias de 90, 160 e 190). Havendo várias distribuições, o sistema atribui uma distribuição Gaussiana a cada mudança mais significativa de iluminação, permitindo assim uma adaptação a variações de iluminação repentinas. Quando aparece um novo objecto, e estando este em movimento, é verificado se pertence a uma distribuição Gaussiana que represente o fundo, e caso não esteja, então é conotado como movimento. Em vez de alterações de iluminação, podem ocorrer movimentos oscilatórios (por exemplo de árvores), mas como há várias distribuições Gaussianas, essas mesmas variações de intensidade podem ser abrangidas pelas distribuições.

O número de distribuições Gaussianas é definido inicialmente pelo programador. Quanto maior for este valor, mais processamento exige. No entanto, se não for suficientemente grande, a subtracção do fundo pode falhar, tal como acontece com o método unimodal. Esse valor pode ser actualizado ao longo do tempo, como acontece na aplicação apresentada em [9], o que permite uma redução no tempo de processamento quando não é necessário um número grande de distribuições Gaussianas, e também proporciona uma maior adaptação ao cenário para uma melhor subtracção de fundo estático.

O método das misturas Gaussianas permite melhorar o desempenho ao aplicar múltiplas distribuições Gaussianas, face ao método que aplica somente uma (unimodal). Desta forma, problemas relacionados com variações de iluminação, ou com movimentos esporádicos podem, em algumas ocasiões, ser evitados.

2.2 Detecção de pessoas

A segunda componente do presente sistema de visão consiste na detecção de pessoas propriamente dita. Tal como foi referido anteriormente, esta fase refere-se à identificação das zonas da imagem que representam pessoas em movimento. Da fase anterior provêm somente as partes da imagem onde figuram alvos em movimento e, desta forma, a presente fase terá de indicar se a zona em questão é ou não uma pessoa.

O objectivo de identificar as pessoas numa imagem pode ser alcançado por diversos métodos, sendo os mais relevantes, e que são alvo de estudo neste relatório, a análise de cor [11-13], o método Pfinder [14-17], detecção de cabeças [15,18], detecção do esqueleto [19,20] e o método de reconhecimento de acções humanas [21]. Há outros métodos de identificação da presença humana, como por exemplo a detecção de caras ou de olhos [22,23], mas estes métodos requerem que a aquisição de imagens se faça em condições específicas, como por exemplo as pessoas estarem de frente para a câmara e muito próximas desta. Portanto a detecção de caras e olhos não foi alvo de estudo neste trabalho.

2.2.1 Análise de cor

A detecção de pessoas com base na análise de cor consiste em detectar, nas imagens, pixeis que têm uma cor semelhante à cor da pele humana. Ao analisar uma imagem, se houver pixeis cujas intensidades sejam semelhantes à cor de pele, há uma grande probabilidade desses pixeis representarem uma pessoa, daí que este método é utilizado para a detecção de pessoas. A análise de cor pode ser implementada via imposição de restrições que limitam as componentes de intensidade dos pixeis a determinados intervalos. No entanto há outras maneiras de implementar a análise de cor, como por exemplo, através de fases de treino e cálculos probabilísticos. Embora mais eficazes, estas vias pautam-se por exigir muitos recursos, o que tornaria o sistema ainda mais lento e inadequado para um sistema a ser usado em tempo real.

Relativamente à análise de cor por via de restrições, foram estudados três métodos diferentes referenciados em [11-13]. No primeiro artigo [11], começam por ser isolados os pixeis cujas componentes vermelha (R) e verde (G) da imagem representada no espaço RGB estão numa determinada gama de valores: (89 <= R <= 140) e (71 <= G <= 89), numa escala de 0 a 255. Se esses pixeis formarem um aglomerado de pixeis conexos (BLOB) com um número mínimo de 300 pixeis, então esse BLOB é considerado uma região de interesse. O facto de se considerarem apenas BLOBs com mais de 300 pixeis permite eliminar pequenas regiões que embora possam ter uma cor semelhante à do corpo humano não têm interesse para o processamento. Para além da pequena dimensão, esta restrição permite eliminar o ruído provocado por pixeis isolados que estão dentro da gama referida. É importante remover o ruído pois evita que a aplicação desperdice recursos para o seu processamento.

Com os aglomerados de pixeis (*clusters*) formados, são desenhadas caixas envolventes *(bounding boxes)* que limitam esses mesmos aglomerados. Determinadas as caixas envolventes, são calculadas as percentagens que os aglomerados de pixeis ocupam dentro destas e caso sejam maiores que 65 % da área então considera-se que a zona corresponde à cabeça.

A detecção de pessoas através deste método não dá resultados eficientes, uma vez que só é aplicável a situações bastante particulares. O facto de restringir a pesquisa dos pixeis à gama indicada pode conduzir a erros. Tal como foi referido anterioremente, o método restringe apenas o valor das componentes vermelha e verde, não impondo qualquer restrição ao valor da componente azul.

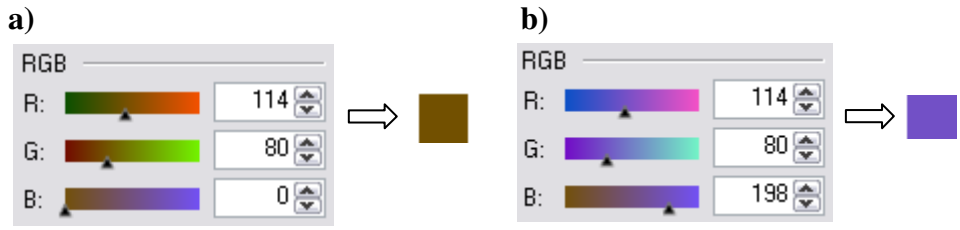

Figura 7 – Influência da componente azul (B) na cor final: a) Componente azul a 0; b) Componente azul igual a 198.

As Figuras 7a e 7b representam o exemplo de duas cores que o método aceita para o tom de pele (valor de 114, intermédio entre 89 e 140 na componente vermelha, e 80, valor entre 71 e 89 na componente verde, e duas intensidades diferentes para a componente azul). Como se pode depreender da Figura 7b, o método permite que o tom de pele tenha uma tonalidade roxa. Isto pode acarretar problemas como o facto de o método abranger alvos (de tonalidade roxa) que não correspondem a pessoas, podendo dar origem a falsos positivos (detectar pessoas quando não existem).

Por outro lado em [12], o sistema de cores utilizado é o HSV (*Hue, Saturation, Value*), ao invés do RGB. Neste caso, o método apresentado restringe os valores das componentes matiz (*hue*) e de saturação (*saturation*). Em [12], para que um pixel seja conotado como pertencente ao fundo, a sua componente *Hue* tem de ter um valor compreendido entre 6 e 38. De modo a remover o ruído causado por pixeis isolados que estão dentro dessa gama, é efectuada uma maximização seguida de uma minimização numa vizinhança de 5x5 de cada pixel no sentido de considerar tom de pele os espaços que não foram considerados como tal e que estão cercados por pixeis com tons de pele. Depois ainda é efectuada uma suavização com uma matriz 3x3. As Figuras 8a e 8b apresentam a imagem original e o resultado obtido através deste método.

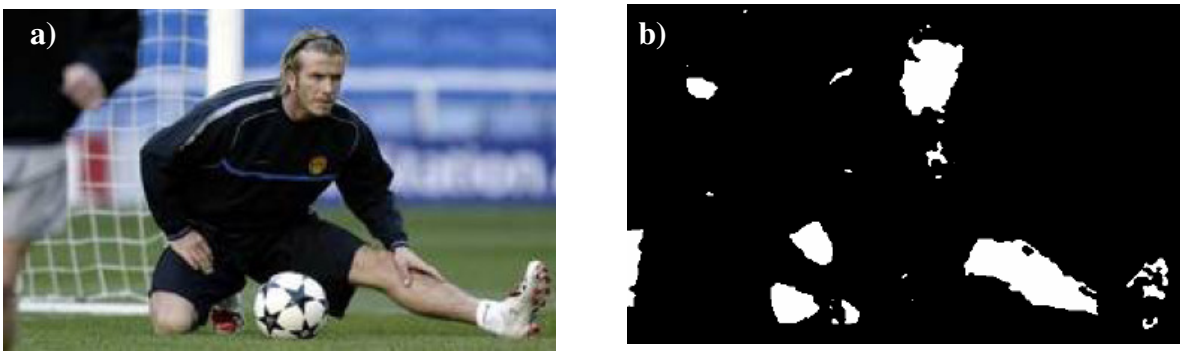

Figura 8 – Aplicação do método da análise de cor [12]: a) Imagem original; b) Resultado final.

No entanto em [13] a abordagem é diferente da anterior. A ideia base é a utilização do espaço RGB em vez do HSV. Este facto permite ao sistema evitar a conversão das intensidades de cada pixel, o que acelera o processo. De modo a detectar a cor de pele, o método referido usa as três componentes RGB e considera como sendo correspondentes a pele humana os pixeis cujas componentes obedecem às seguintes restrições:

$$R > 95, G > 40, B > 20 \quad (2.1)$$
$$\max\{R, G,B\} - \min\{R, G,B\} > 15 \quad (2.2)$$
$$|R - G| > 15 \quad (2.3)$$
$$R > G, R > B \quad (2.4)$$

(2.1) – Limiares para as components “Red”, “Green” e “Blue”
(2.2) – As components não podem ter valores próximos, eliminação de tons cinzento
(2.3) – A componente “Red” e a “Green” não podem estar próximas
(2.4) – A componente “Red” tem de ser a que tem maior valor de intensidade

Os autores referem que se obtêm resultados satisfatórios desde que a iluminação seja suficientemente intensa. Para tentar evitar esta restrição, procederam à detecção do tom de pele após converterem as imagens para um dos espaços de cores YCrCb, ou YUV, uma vez que estes espaços de cores, separam a luminância das componentes relativas à cor (cromaticidade). No entanto, a mesma referência apresenta experiências que demonstram que esta conversão não é uma boa abordagem, devido ao alto número de falhas comparativamente com a utilização do espaço de cores RGB [13].

Da análise destes três artigos, conclui-se que a análise de cor pode ser um bom meio para distinguir entre objectos e pessoas. É evidente que qualquer um destes métodos é falível em situações em não é visível a pele de nenhuma parte do corpo.

2.2.2 Método Pfinder

Este método [14, 15] engloba não só a detecção de pessoas como também a subtracção do fundo. Em primeira instância, o método subtrai o fundo estático, aplicando o método unimodal. Ao capturar uma imagem e receber a sua informação, o método cria um modelo do fundo estático, dando origem a uma média e desvio-padrão associado a cada pixel. Em semelhança ao método unimodal, para cada pixel, sempre que a intensidade de um pixel se desviar da gama $[\mu-\sigma, \mu+\sigma]$, então é considerado como pertencente a um objecto em movimento.

Depois de segmentados os alvos em movimento, o método utiliza as silhuetas dos mesmos para encontrar as partes do corpo, como a cabeça, mãos, pernas, tronco [16, 17]. Caso encontre, então aplica um BLOB à respectiva parte do corpo. Se o conjunto de BLOBs formar um modelo semelhante a uma pessoa, então o alvo é conotado como sendo uma pessoa [14].

Figura 9 - Aplicação de BLOBs às diversas partes do corpo humano [17].

Este método é robusto em relação a oclusões no sentido que este cria e elimina os BLOBs referentes a partes do corpo à medida que ocorre o aparecimento e a oclusão das mesmas. No entanto, segundo as fontes [14,16], por utilizar o método unimodal, este método tem restrições relativamente a mudanças repentinas no ambiente, como a variação da iluminação. Outra lacuna deste método é o facto de ser capaz de captar somente uma única pessoa.

2.2.3 Detecção de cabeças

Um método possível para detectar cabeças é baseado na detecção de regiões das imagens que correspondem à forma da cabeça humana juntamente com os ombros (uma forma do tipo da letra grega Ómega - Ω) [18]. Este pode ser implementado de duas formas: detecção de cabeças nas fronteiras entre alvos em movimento e o fundo, ou por análise de intensidade. A primeira, a detecção de cabeças nos limites dos alvos em movimento, procura somente o formato da cabeça e ombros nas zonas de fronteira, extraídas após a subtracção do fundo. Esta via conduz a identificar pessoas cujas projecções das suas cabeças estão juntas, ou fazem fronteira ao fundo anteriormente subtraído.

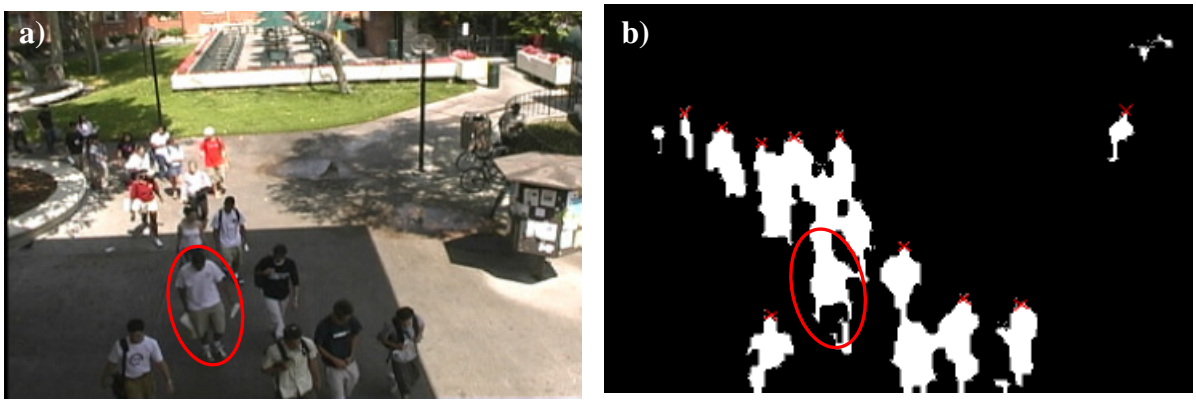

Figura 10 – Detecção de cabeças nas fronteiras entre alvos em movimento e o fundo[18]: a) Imagem original; b) Detecção de cabeças após subtracção de fundo.

O método de detecção de cabeças na fronteira analisa cada alvo de cima para baixo e verifica se este contém a forma de um ómega. O método pressupõe que as cabeças estão sempre no topo dos BLOBs, esperando assim que a forma de um ómega esteja nessa mesma posição. Na Figura 10b estão apresentadas com uma cruz vermelha as zonas dos BLOBs que contêm a forma de ómega. Por esta via não é possível detectar pessoas (neste caso, cabeças) quando a cabeça de uma pessoa não está na fronteira com o fundo estático. Na Figura 10b está assinalada com uma elipse vermelha uma pessoa cuja cabeça não está na fronteira com o fundo estático, o que resulta na não detecção dessa mesma pessoa (igualmente assinalado com um círculo vermelho na Figura 10a).

Por outro lado, a detecção de cabeças pela intensidade, em vez de procurar nas zonas de fronteira, segue um método de detecção de arestas (método de Canny) nas regiões da imagem que representam movimento (Figura 11a). A detecção de arestas consiste na localização das zonas onde há variações acentuadas de intensidade, ou seja, a contornos dos objectos. O método procura contornos em forma de Ω e sempre que encontrar algum, considera que se trata de uma pessoa (Figura 11b). Este método consegue colmatar a lacuna do método anterior, pois é capaz de detectar pessoas mesmo quando as suas cabeças não estão na fronteira com o fundo estático. No

entanto, pode acontecer que ocorram várias formas do tipo Ω que não correspondem a pessoas, originando assim falsos positivos.

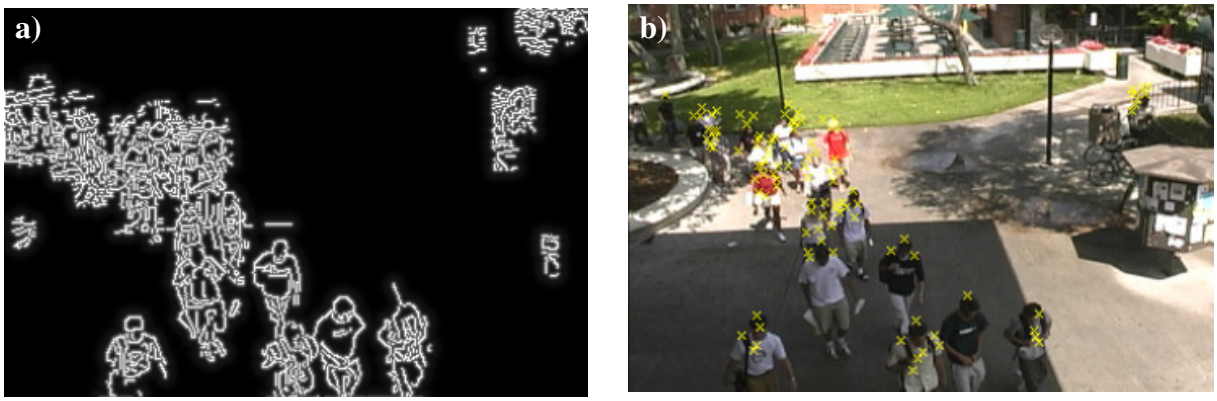

Figura 11 - Detecção de cabeças por via por análise de intensidade (arestas) [18]: a) Aplicação do método de detecção de arestas a uma imagem; b) Detecção de cabeças após a análise da Figura 9a.

Ambos os métodos ainda recorrem à contagem do número de pixeis pertencentes a um alvo em movimento que estão abaixo da zona da cabeça, detectada através de Ω, verificando assim se o objecto em questão tem uma "altura" (número de pixeis suficientes) para que seja considerado uma pessoa.

Na detecção humana através da detecção de cabeças, o problema reside no facto de haver oclusão da cabeça da pessoa. De resto, demonstra ser um método eficiente para a detecção de pessoas na ocorrência de multidões, ao contrário do método anterior.

2.2.4 Detecção de esqueleto

Este método tem como base o facto do corpo humano conter articulações e o movimento humano poder ser representado somente pelo movimento do esqueleto. Desta forma, este método processa a imagem de modo a identificar articulações e verifica se de facto correspondem a uma pessoa [19].

O número de articulações extraídas da imagem depende da aplicação. Por exemplo, no projecto Natal da Microsoft™ [20], cujo objectivo é seguir os gestos das pessoas, são extraídas 48 articulações, existentes na cabeça, mão, pulso, pés, ombros entre outras partes. No entanto em [19] é apresentado um exemplo em que se usam 16 articulações (Figura 12). Sendo conhecida a estrutura anatómica humana, é possível detectar pessoas com base nas articulações. Este método é mais usado para o reconhecimento e identificação de movimentos das pessoas do que para a detecção da sua presença. Este método também recorre a sistemas de visão por computador de modo a que uma pessoa possa interagir com a aplicação somente através de gestos e movimentos, sem o uso de qualquer comando.

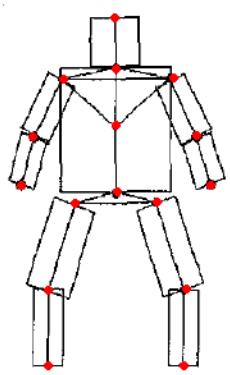

Figura 12 - Representação do esqueleto humano com 16 articulações [19].

2.2.5 Reconhecimento de acções humanas

O método de reconhecimento de acções humanas tem como ideia base o facto das acções humanas envolverem movimentos repetitivos que seguem modelos de poses de pessoas previamente guardados. Desta forma, o método guarda um ou vários modelos que representam cada acção humana e que depois são comparados com cada imagem de um alvo em movimento [21]. Este método baseia-se no facto de, por exemplo, o movimento de uma pessoa a caminhar, poder ser representado por três modelos: um modelo em que as pernas estão abertas, outro modelo em que as pernas estão fechadas e um modelo intermédio em que as pernas estão numa posição intermédia entre abertas e fechadas. Sempre que num vídeo surge um objecto em movimento, cada imagem da sequência é comparada com cada modelo guardado, de modo a verificar se corresponde a algum dos modelos. Se tal ocorrer, significa que o objecto é uma pessoa.

Este método requer que a base de dados que guarda as posições e movimentos das pessoas seja muito grande, envolvendo cada movimento e dentro de cada movimento, as diferentes perspectivas. Tal como o método anteriormente referido, este método é mais utilizado para reconhecimento de movimentos humanos, ao invés da detecção de pessoas.

2.3. Vantagens e desvantagens dos métodos analisados

Terminada a análise do estado da arte de ambas as componentes do sistema em questão, subtracção de fundo e detecção de pessoas, conclui-se que os métodos mais adequados são as misturas gaussianas para a subtracção de fundo e uma junção do método da análise cor com o método de detecção de cabeças para a detecção de pessoas.

As misturas gaussianas proporcionam uma maior adaptação às mudanças de iluminação que ocorram, permitindo desta maneira uma melhor subtracção de fundo, comparativamente com o método unimodal e com o método da diferença de *frames*. Relativamente ao métodos de *Kernel Estimation* e de *codebook*, o facto das misturas Gaussianas atribuirem pesos às distribuições Gaussianas conferem ao sistema a possibilidade de ser robusto face a alterações de iluminação e a outros movimentos sem interesse como o movimento oscilatório de cortinados, ou folhas de papel provocado pelo vento.

Por outro lado, na detecção de pessoas, os dois métodos escolhidos são a análise de cor e a detecção de cabeças nas fronteiras entre alvos em movimento e o fundo. Esta escolha é fundamentada pelo facto do método da análise de cor gastar poucos recursos, isto é, somente é

necessário analisar os valores dos pixeis e verificar se esses valores estão dentro de uma gama de valores permitidos. Mas como pode haver objectos em movimento com o tom de pele humana que não são pessoas, reforça-se o método com a detecção de cabeças, de modo a verificar se o alvo em movimento tem uma parte que corresponde ao formato de uma cabeça. Já o método Pfinder, é limitado no sentido de detectar apenas uma pessoa. Os métodos de detecção de esqueleto e de acções humanas, para além de serem mais utilizados para o reconhecimento de acções (algo que não é objectivo da presente dissertação) exigem a comparação de um ou vários modelos previamente guardados, o que implica um grande armazenamento de tais modelos.

Capítulo 3

Implementação

Neste capítulo descreve-se o modo como o sistema em questão foi desenvolvido. Primeiramente são apresentadas as ferramentas usadas para a elaboração do trabalho e de seguida, são explicados os passos efectuados durante a implementação do sistema, as escolhas efectuadas e respectivas justificações.

3.1 Bibliotecas e linguagens usadas

De modo a implementar o sistema em questão é necessária a utilização de ferramentas que auxiliem o seu desenvolvimento. De seguida serão apresentadas as tecnologias utilizadas no presente projecto, acompanhadas com as razões pelas quais se optou para cada uma das suas aplicações.

3.1.1 OpenCV

A ferramenta OpenCV, Open Computer Vision Library, é uma biblioteca criada pela Intel que é direccionada para o desenvolvimento de aplicações na área de visão por computador. Esta biblioteca contém funções já implementadas que auxiliam nos projectos do programador. As linguagens de programação compatíveis com esta biblioteca são C/C++, Python e Octave.

Na dissertação, a versão utilizada é a 2.0 que foi lançada em Setembro 2009, sendo a mais recente até à data de início dos trabalhos conducentes a esta dissertação.

Dentro do próprio OpenCV ainda foi utilizada a biblioteca cvBlobsLib, sendo esta uma ferramenta muito útil para a detecção e manipulação de BLOBs.

3.1.2 C/C++

A linguagem de programação utilizada neste trabalho de dissertação é C/C++. A escolha desta linguagem baseou-se pelo facto de ser uma das linguagens de programação compatível com o OpenCV.

3.2 Subtracção do fundo

A subtracção do fundo foi a primeira parte do sistema a ser implementada, sendo esta essencial para o bom funcionamento da parte seguinte, a detecção de pessoas. Durante o desenvolvimento do sistema, foram desenvolvidos três algoritmos que são descritos seguidamente. De modo a optimizar os algoritmos, é necessário que haja algumas preocupações, como por exemplo a declaração de variáveis fora de ciclos, que o número de ciclos seja o mínimo possível, minimizar a chamada de funções. As imagens captadas estão no espaço de cores RGB, e são convertidas em níveis de cinzento, para que os cálculos efectuados no restante código utilizem somente uma componente, em vez de três. Tudo isto foi tido em conta na subtracção do fundo de modo a proporcionar uma melhor eficácia na utilização de recursos.

Como resultado final da subtracção do fundo, obtém-se uma imagem binária onde a cor branca representa movimento e a cor preta o fundo estático.

3.2.1 Método da diferença de *frames*

Um dos métodos para separar regiões de uma sequência de vídeo que apresentam movimento das regiões que representam objectos estáticos é calcular a diferença entre as intensidades dos pixeis na mesma posição em imagens consecutivas.

Este método consiste na subtracção de um valor de intensidade de um pixel de uma imagem numa posição (x,y) por um outro valor de um pixel na mesma posição da imagem seguinte do vídeo. Caso o resultado da diferença não seja igual a zero é porque houve uma mudança de intensidade de uma imagem para a outra no mesmo local, o que representa movimento. Na realidade, a diferença não se pode cingir ao valor zero, isto para dar uma margem a pequenas variações de intensidade que possam ocorrer e que não representam propriamente movimento. O nível de similaridade entre pixeis é definido pelo programador, através de um limiar (*threshold*).

A biblioteca do OpenCV fornece algoritmos que permitem o cálculo das diferenças entre duas imagens (`cvAbsDiff`), onde as imagens têm de ter dimensões iguais. A função coloca as diferenças de intensidades de todos os pixeis (em valor absoluto) numa imagem, e para que haja uma margem para pequenas variações de intensidade, essas diferenças são comparadas com um limiar. Se algum valor da diferença de intensidades for superior ao limiar, então é considerado como sendo movimento, caso contrário, é fundo estático. O resultado das diferenças é depois apresentado numa janela.

O método referido não apresenta resultados suficientemente satisfatórios para o projecto, uma vez que há problemas inerentes como alterações de iluminação que provocam maus resultados, mas é um assunto que será demonstrado mais à frente. O sistema a implementar deve ser suficientemente robusto de modo a não permitir a identificação de movimento quando a iluminação é alterada ou outras condições que, em suma, não representam movimento.

3.2.2 Método Unimodal

O método unimodal foi o método que foi implementado depois da subtracção directa de imagens. Este método é utilizado para a detecção de movimento e serve de base para a implementação do método das misturas Gaussianas. Tal como foi referido anteriormente, o método unimodal tem como princípio o pressuposto que as intensidades dos pixeis do fundo seguem, ao longo do tempo, uma distribuição Gaussiana, ou seja, os valores das intensidades estão compreendidos numa gama [μ-σ, μ+σ] em que μ e σ representam, respectivamente, a média e o desvio-padrão da intensidade de cada pixel.

De modo sucinto, o algoritmo em questão começa pelo cálculo da média e do desvio-padrão para cada pixel através da sucessiva captura de imagens. Esta fase é designada por fase de treino e a sua duração (número de imagens captadas) é imposta pelo programador. Após esta fase, passa-se para a fase de detecção de movimento em que um ciclo está sempre a capturar imagens, e a cada pixel será comparado o seu valor da intensidade face aos valores da média e desvio-padrão. Caso o novo valor do pixel esteja inserido no intervalo já referido anteriormente, então coloca-se esse mesmo pixel a preto numa imagem auxiliar, e caso contrário, o pixel da imagem auxiliar fica a branco, pois é considerado movimento. Para que seja retirado algum ruído da imagem auxiliar causado por pixeis isolados, procede-se a duas maximizações seguidas de três minimizações nessa mesma imagem. Depois os valores das médias e desvios-padrão de cada pixel são actualizados com diferentes taxas consoante o pixel seja considerado como fundo ou movimento. Se o pixel em causa for fundo, então a taxa de actualização dá tanto relevo ao pixel que entrou como aos valores da média e desvio-padrão que já estão. Mas se o pixel for movimento, então a taxa dá mais importância aos valores da média e desvio-padrão que já estão anteriormente do que ao pixel que entrou. Isto justifica-se pelo facto da distribuição Gaussiana de cada pixel representar os valores permitidos para o fundo estático. Logo terá que haver um registo mais expressivo sobre o fundo estático, desprezando assim a relevância dos valores de intensidade que representam alvos em movimento. Portanto, a partir do conhecimento do algoritmo, a implementação tem a seguinte composição:

- Fase de treino
- Fase de detecção de movimento

Durante a fase de adaptação ao OpenCV, foi constatado que as imagens capturadas inicialmente não correspondem à realidade, tendo todos os valores a 0. Isto deve-se ao facto da câmara ainda estar a iniciar o processo de aquisição de imagem. Desta forma, aproveitou-se o facto da câmara ainda estar a iniciar para obter as dimensões da imagem com que o programa irá processar e que serão utilizadas durante o algoritmo. Na implementação, optou-se por deixar os valores das dimensões variáveis mediante a câmara utilizada.

Após a definição das dimensões, são definidas as estruturas `IplImage` do OpenCV que serão necessárias para o algoritmo. Todas estas definições e declarações foram colocadas na parte inicial do código para tornar o método mais optimizado, uma vez que se fossem colocadas em ciclos, iria tornar o processo mais pesado e lento. Terminada esta parte, é a fase de treino que se segue.

3.2.2.1 Fase de treino

A fase de treino consiste em dois ciclos. O primeiro ciclo consiste num número de iterações (valor este que é variável mediante opção do programador e depende o quanto ele deseja que os valores iniciais da média e desvio-padrão sejam mais próximos da realidade). Nesse ciclo,

a cada pixel são efectuados os cálculos auxiliares para a média e desvio-padrão referentes às imagens recebidas durante esta fase. Depois de terminar as iterações anteriores, são percorridos uma vez todos os pixeis de modo a calcular os valores reais da média e do desvio-padrão. Os cálculos auxiliares são equações que não têm divisões nem raízes quadradas e que não representam os valores reais da média e desvio-padrão. O cálculo dos valores reais do desvio-padrão e da média só é efectuado no final da fase de treino. Até lá, apenas são acumulados os valores necessários para esse cálculo.

A decisão de ter os cálculos auxiliares e só depois os cálculos reais dos valores da média e do desvio-padrão justifica-se no sentido dos cálculos reais serem pesados em termos de processamento durante várias iterações. Por exemplo, no cálculo do desvio-padrão teria de haver uma raiz quadrada e na média uma divisão em cada pixel a cada iteração executada. Sendo assim, no final do treino, todos os pixeis são percorridos apenas uma vez, para serem calculados os verdadeiros valores da média e do desvio-padrão, sendo somente nesta fase aplicada a raiz quadrada e a divisão.

3.2.2.2 Fase de detecção de movimento

A fase de detecção de movimento contém um ciclo, no início da fase, que funciona de forma semelhante ao ciclo da fase de treino, onde são captadas imagens provenientes da fonte (vídeo ou câmara) em RGB e são convertidas para níveis de cinzento, por razões já indicadas. De seguida, é percorrido cada pixel da imagem adquirida e é verificado se a sua intensidade está na gama $[\mu-\sigma, \mu+\sigma]$. Caso esteja dentro dessa gama, significa que pouco ou nada se alterou face aos valores medidos na fase de treino, ou seja pertence ao fundo, e coloca-se um pixel a preto numa imagem auxiliar na posição do pixel actual. Caso contrário, é marcado como pixel em movimento, colocando um pixel a branco na imagem auxiliar.

Após processar todos os pixeis da imagem adquirida obtém-se uma imagem binária com pixeis a preto e branco, esta imagem é a imagem auxiliar. A esta imagem são feitas duas dilatações (maximizações) com uma matriz 3x3 seguida de três erosões (minimizações) igualmente com uma matriz 3x3. Este processo é efectuado para que o ruído provocado por pixeis isolados que representam movimento no meio de pixeis que representam objectos estáticos e também, por outro lado, que representam objectos estáticos entre pixeis que representam movimento, todos eles obtenham os valores predominantes nas suas vizinhaças. Passada a fase de eliminação de ruído, obtém-se a imagem binária com ruído atenuado que representa alvos em movimento a cor branca e o fundo estático a preto.

Por fim, ainda é efectuada uma nova passagem por todos os pixeis na imagem auxiliar e são feitos os refinamentos aos valores da média e desvio-padrão, em que caso se trate de um pixel preto (representando fundo), então é dada tanta relevância ao pixel em questão como aos valores da média e desvio-padrão existentes. E caso seja um pixel que represente movimento, o novo valor é praticamente desprezado, isto para que os valores do fundo (média e desvio-padrão calculados anteriormente) imperem face aos valores do movimento. Estes novos valores refinados das médias e desvios-padrão são necessários para que nas iterações seguintes haja uma maior adaptação a algumas variações que possam ocorrer. Assim, no caso de haver uma variação lenta da iluminação que permita a adaptação dos valores indicados, é possível que o sistema fique robusto no sentido de não detectar movimento onde não existe. Os cálculos dos novos valores das médias e desvios-padrão são efectuados de acordo com as seguintes equações [24, 25]:

Média: $\mu(t) = (1-\alpha) \times \mu(t-1) + \alpha I(t)$ (3.1)

Desvio-padrão: $\sigma(t) = \sqrt{(1-\alpha) \times \sigma(t-1)^2 + \alpha (I(t) - \mu)^2}$ (3.2)

Legenda:	**μ – Média**
	t - Tempo
	α – Taxa de aprendizagem (relevância)
	I – Novo valor da intensidade
	σ – Desvio-padrão

Todo este processo é efectuado sobre todos os pixeis da imagem adquirida, e repetido à medida que novas aquisições de imagens são efectuadas, terminando quando o programador fechar o processo. A tabela seguinte mostra esquematicamente os passos envolvidos numa só iteração, sendo eles repetidos à medida que o sistema adquira novas imagens.

Tabela 1 – Esquema que representa o pseudo-código do método unimodal.

Entrada: Sequência de imagens provenientes de um vídeo **Saída**: Imagem binária que contém o fundo a cor preta e os objectos em movimento a branco

for (cada pixel da imagem){
  if (o valor do pixel estiver contido na gama da distribuição Gaussiana do pixel em causa)
  Colocar pixel a branco na imagem binária
  else
  Colocar pixel a preto na imagem binária
}
 	
2 x Dilatação da imagem binária
3 x Erosão da imagem binária
 	
for (cada pixel da imagem){
  Actualizar os valores das médias, desvios-padrão mediante se pertencem ao fundo ou a um objecto em movimento
}

Desta forma o método unimodal foi implementado. O passo seguinte foi a implementação do método das misturas Gaussianas.

3.2.3 Método das misturas Gaussianas

O método das misturas Gaussianas é semelhante ao método unimodal com a diferença de em vez de utilizar apenas uma distribuição Gaussiana, é aplicado um conjunto de distribuições Gaussianas, por exemplo 3, 4, 5 ou mais distribuições para cada pixel. O facto de haver mais distribuições Gaussianas permite ao sistema ser mais robusto a alterações do ambiente, uma vez que este não é controlado. No projecto em questão implementou-se 3 distribuições Gaussianas para cada pixel, uma vez que um aumento deste valor iria trazer um maior gasto em termos de processamento. No entanto o número de distribuições deve ser escolhido mediante o cenário onde o sistema irá ser aplicado. Por exemplo, num ambiente onde há muitas alterações de iluminação e movimentos oscilatórios, é necessária a existência de um elevado número de distribuições para que os valores dessas variações estejam abrangidos pelas distribuições. Por outro lado, um ambiente muito estático não requer muitas distribuições.

À semelhança do método unimodal, tem de haver uma adaptação dos valores das diferentes distribuições à medida que entram os novos valores de intensidade provenientes das novas imagens. Para isso utilizar-se-ia um processo denominado por *Expectation-Maximization* [26-28], mas como este processo consome muitos recursos (o que impossibilitaria o facto de transpôr o sistema para tempo real), optou-se por efectuar uma adaptação do processo que [9] igualmente efectuam. Essas adaptações resumem-se a uma simplificação dos cálculos inerentes ao método. Nas misturas Gaussianas, a parte de *Expectation* refere-se ao cálculo da distribuição Gaussiana que deve ser eleita para ser comparada com o valor de entrada (pode ser utilizado a distância de Mahalanobis, por exemplo, de modo a verificar qual é a distribuição mais próxima do valor de entrada). Já, a parte de *Maximization* refere-se à reavaliação dos valores das médias e desvios-padrão da distribuição escolhida na parte de *Expectation*. Esta reavaliação é efectuada mediante cálculos que serão apresentados mais à frente. Uma das adaptações efectuadas de modo a evitar recursos computacionais foi o cálculo da distribuição mais próxima do valor de entrada, ou seja, em vez de utilizar a distância de Mahalanobis, utilizou-se uma equação simples para minimizar a complexidade dos cálculos.

Para além das várias distribuições que distinguem este método do unimodal, há ainda a atribuição da componente de peso a cada distribuição de cada pixel. Esta componente de peso representa a relevância de cada distribuição a qual será tida em consideração para comparar os valores da média e desvio-padrão das distribuições com os valores provenientes das imagens captadas. Note-se que o peso de cada distribuição Gaussiana é influenciado pela proximidade dos seus valores da média e desvio-padrão com os novos valores das intensidades dos pixeis provenientes das novas imagens. Quantas mais vezes os valores de uma determinada distribuição ficarem mais próximos dos valores das imagens relativamente aos valores das outras distribuições, maior será o seu peso.

Tal como o método unimodal, o método das misturas Gaussianas também está dividido em duas fases: fase de treino e fase de detecção de movimento.

3.2.3.1 Fase de treino

A fase de treino consiste num processo iterativo em que os valores da média, desvio-padrão e peso de cada uma das distribuições Gaussianas de cada pixel (3 distribuições, neste caso) são calculados. Esses cálculos têm com base os valores das intensidades dos pixeis das imagens captadas. Por tudo isto, a fase de treino deste método é mais pesada em termos de recursos do que a mesma fase no método unimodal. A aprendizagem é feita inicializando todos os valores das

médias e dos desvios-padrão a 128 (para que a média esteja a meio da gama dos níveis de cinzento, 0-255, e para que o desvio-padrão abranja toda a gama de intensidades), e o peso com o valor 1 na primeira distribuição e 0 nas duas restantes. É de referir que a soma dos pesos das distribuições terá de ser igual a 1. Com estes valores e após várias iterações (o programador define o número de iterações, consoante o desejo de obter valores refinados), os valores das médias, desvios-padrão e pesos vão-se alterando até começarem a estabilizar em valores próximos da realidade.

À medida que é feita a aquisição de imagens, como os valores da média são de 128 e do desvio-padrão igualmente 128, então inicialmente todos os valores das novas imagens estão dentro da gama de todas as distribuições Gaussianas. No entanto, como o peso da primeira distribuição Gaussiana é igual a 1, é a esta distribuição que os novos valores das imagens irão pertencer, sendo os seus valores da média e desvio-padrão alterados em conformidade. A aprendizagem das distribuições Gaussianas é feita, tal como foi referido, com base nos novos valores provenientes das imagens captadas, sendo efectuados os seguintes cálculos [24]:

$$\text{Peso: } \omega(t) = \frac{(\mathrm{N} \times \omega(t-1) + \mathrm{E})}{\mathrm{N}+1} \tag{3.3}$$

Legenda: ω **– Peso**
N – Número de iterações
E – 1 ou 0, se for a distribuição escolhida (1) ou não (0)

Os valores da média e do desvio-padrão são calculados de igual modo no método unimodal e estão representados em 3.1 e 3.2 respectivamente.

O valor do peso de cada distribuição aumenta à medida que há um maior número de intensidades que estão dentro ou próximas da gama [μ-σ, μ+σ] da respectiva distribuição. Tal como referido anteriormente, a primeira distribuição de cada pixel tem o seu valor do peso inicializado a 1 e as outras distribuições a 0, mas como ao longo do tempo os valores da média e desvio-padrão vão sendo restringidos pelos cálculos apresentados, diminuindo assim a amplitude da sua gama [μ-σ, μ+σ]. Essa diminuição do intervalo traduz-se no facto das intensidades dos pixeis das novas imagens ficarem mais próximos dos valores das outras distribuições, passando assim para a distribuição seguinte. Desta forma, o peso da primeira distribuição irá descer, e o valor do peso da distribuição seguinte subir. Esses novos valores de intensidade que irão afluir às distribuições seguintes (a segunda distribuição e depois na terceira), influenciam consequentemente os valores da média, desvio-padrão das respectivas distribuições.

O facto das novas intensidades não pertencerem à primeira distribuição, mas sim às seguintes é o que proporciona a robustez ao algoritmo das misturas Gaussianas, uma vez que as mudanças de intensidade não ficam imediatamente conotadas como sendo movimento, mas sim remetidas para outras distribuições Gaussianas que representam igualmente fundo estático. Quanto maior o número de distribuições que o sistema contém, maior robustez proporciona relativamente a alterações de intensidade que não são de interesse. No entanto é necessário que na fase de treino haja tais alterações de intensidade que permitam o preenchimento nas distribuições seguintes, de modo a que haja maior robustez na identificação do fundo estático. Por outras palavras, caso não ocorram variações do fundo estático, só os valores pertencentes à primeira distribuição é que são preenchidos, tornando o método presente igual ao método unimodal.

No final da fase de treino é verificado para cada pixel qual é a distribuição Gaussiana com maior peso, sendo essa conotada como sendo a distribuição principal, ou seja, a distribuição Gaussiana cujos valores servirão para serem comparados com os valores dos pixeis das imagens novas.

3.2.3.2 Fase de detecção de movimento

Após terminar a fase de treino é efectuada a fase de detecção de movimento que é semelhante à correspondente fase no método unimodal, com a excepção do cálculo dos pesos. A cada iteração o sistema adquire uma nova imagem e para cada pixel é determinada qual é a distribuição principal (a que tiver maior peso) para comparar os valores da média e desvio-padrão desta com os valores da intensidade recebidos do pixel em questão. Se a intensidade estiver dentro da gama da distribuição com maior peso, então o pixel em questão é conotado como pertencente ao fundo, caso contrário é conotado como sendo pertencente a um objecto em movimento. Posteriormente é verificado qual é a distribuição cujos valores sejam os mais próximos ao valor da intensidade do pixel e é a essa distribuição que são efectuadas as actualizações dos seus valores (média, desvio-padrão e peso). Tal como acontece no método unimodal, há dois tipos de actualizações dos valores, caso o pixel em questão seja conotado como pixel estático ou em movimento. No final do ciclo verifica-se qual é a distribuição com maior peso, tornando-se essa na distribuição principal.

A seguir está representado o esquema do algoritmo das misturas Gaussianas aplicado no sistema presente relativo a apenas uma iteração.

Tabela 2 - Esquema que representa o pseudo-código do método das misturas Gaussianas.

Entrada: Imagem proveniente de um vídeo **Saída**: Imagem binária que contém o fundo a cor preta e os objectos em movimento a branco

Aquisição de imagem e **Conversão** para níveis de cinzento
 	
for (cada pixel da imagem recebida){
  if (o valor do pixel estiver contido na gama da distribuição Gaussiana principal)
  Colocar pixel a branco na imagem binária
  else
  Colocar pixel a preto na imagem binária
}
 	
2 x Dilatação da imagem binária
3 x Erosão da imagem binária
 	
for (cada pixel da imagem binária){
  Verificar qual a distribuição Gaussiana mais próxima do valor do pixel
 	
  Actualizar os valores das médias, desvios-padrão e pesos, caso pertençam ao fundo ou a um objecto em movimento
}

3.3 Detecção de pessoas

A segunda fase do projecto consistiu no desenvolvimento da detecção de pessoas propriamente dita. Baseia-se na utilização dos alvos em movimento provenientes da fase da subtracção de fundo, e consequente verificação se tais alvos representam pessoas. Tendo em conta este objectivo, foram implementados dois métodos, um baseado em análise de cor, por forma a detectar a existência de pele humana, outro baseado na detecção de cabeças, que serão de seguida explicados.

3.3.1 Análise de cor

Para detectar a existência de pele humana nos alvos em movimento, foram implementados dois métodos baseados na análise de cor do alvo. Os espaços de cor usados para representar as imagens são diferentes consoante o método: HSV no primeiro e RGB no segundo.

O primeiro método tem como princípio-base o facto de, em condições favoráveis de iluminação, a gama da componente Hue em HSV estar compreendida entre 6 e 38, valores estes obtidos através de experiências em [11]. O resultado deste método é uma imagem binária em que os pixeis que supostamente representam pele humana (cuja Hue está dentro da gama de valores referida) são assinalados a branco e os restantes pixeis são assinalados a preto.

O segundo método diferencia-se no espaço de cores, que neste caso é RGB, e nas restrições. Estas últimas podem ser agrupadas em três. O primeiro grupo de restrições impõe que a componente vermelha seja maior que 95, a componente verde maior que 40 e a componente azul maior que 20. O segundo grupo impõe que pelo menos 2 dos 3 valores das componentes não estejam próximos ($\max\{R, G, B\} - \min\{R, G, B\} > 15$) e finalmente, que a componente vermelha tenha o maior valor das três componentes, e ainda que a diferença entre essa e a componente verde tenha um valor superior a 15. É de salientar que se optimizou o código apresentado em [11], em que se eliminou uma condição ("$|R - G| > 15$" e "$R > G$" podem ser condensadas em "R - G > 15"). Estas restrições foram igualmente sugeridas mediante resultados obtidos em [12]. Efectuadas estas restrições, é apresentado o resultado da mesma maneira que o método anterior.

3.3.2 Detecção de cabeças

O método da detecção de cabeças baseia-se na análise da imagem e na verificação da forma dos alvos de interesse, de modo e certificar-se da semelhança com uma cabeça. Desta forma, é necessário que o programa receba a informação de cada BLOB que representa movimento e a partir daí procurar por formas semelhantes a cabeças.

Segundo [29], há três métodos para localizar cabeças de uma imagem, estando dois deles incluídos no sistema. Os dois métodos que foram implementados são o método que verifica a forma do sigma (Ω) e o método que verifica a forma elíptica. Tal como foi referido no Capítulo 2, a parte superior do corpo humano (cabeça e ombros) tem, aproximadamente a forma da letra grega sigma. Por outro lado, a cabeça humana tem uma forma aproximadamente elíptica. Daí a implementação destes dois métodos.

O modo como se implementou o primeiro método (detecção da forma de sigma) baseia-se no facto de que a cabeça está sempre no topo do alvo em movimento. Para além deste facto, ainda é defendido por [29] que existe um aumento, seguido de um decréscimo no gráfico que contém o

somatório de pixeis do BLOB no eixo vertical. Este comportamento simula a letra grega referida, e é demonstrado na Figura 13.

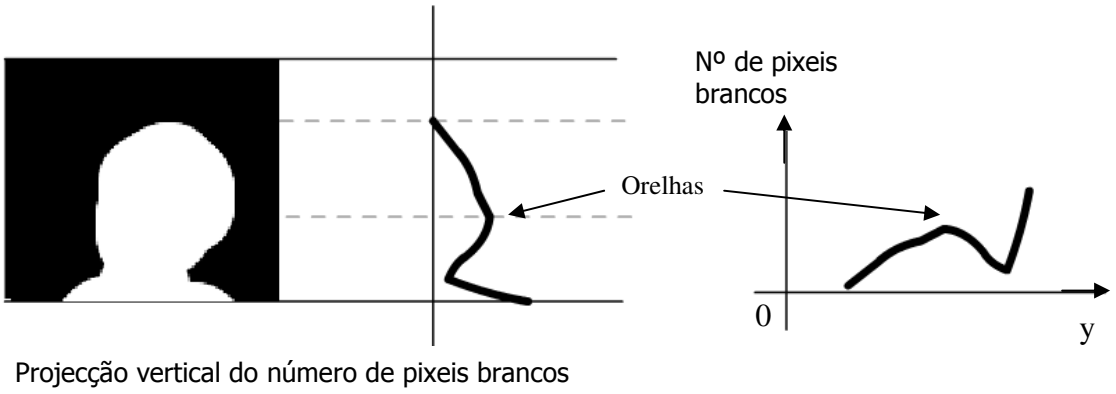

Figura 13 – Demonstração da forma de sigma da cabeça. Os gráficos representam o número de pixeis a branco na horizontal e ponto onde se encontram as orelhas.

Como é perceptível pela imagem anterior, o aumento corresponde à zona da cabeça entre o topo desta e as orelhas, onde a cabeça atinge o máximo da sua largura, e que também representa a metade de cima do sigma. Por outro lado, o decréscimo representa a parte entre as orelhas e o pescoço, em que a quantidade de pixeis vai diminuindo até encontrar uma nova subida, a zona dos ombros. Esta fase de decréscimo também assemelha-se à metade de baixo da forma da letra sigma.

Passando à descrição do código implementado, para cada BLOB em movimento, o algoritmo percorre cada linha de pixeis, calculando o somatório do número de pixeis que existe em cada linha. Enquanto o somatório é calculado, é verificado se o seu valor é maior que máximo valor do somatório até então registado. Em caso afirmativo, o valor máximo toma o valor do somatório da linha actual, o que significa que ainda se está na fase de crescimento, ou seja entre topo da cabeça e as orelhas. Em caso negativo, significa que se atingiu um máximo local, o que representa a zona das orelhas, a parte mais larga da cabeça e consequentemente contém mais pixeis. A partir deste momento é expectável que haja o decréscimo referido, em que a quantidade de pixeis em cada linha iria diminuir progressivamente. No entanto, no programa esta condição não existe, não restringindo assim que haja o decréscimo, isto devido ao facto da cabeça não seguir estritamente a forma do sigma, pois pode conter barba, cabelo comprido, o que torna a parte da cabeça abaixo das orelhas diferente da metade de baixo do sigma. Desta forma, o algoritmo implementado, procura um outro máximo, ainda maior que a largura da cabeça, assinalando assim os ombros, isto porque a envergadura dos ombros é maior que a largura da cabeça, mesmo se a pessoa se encontrar de perfil. Neste sentido, caso o BLOB corresponda aos pontos acima referidos, então é verificado ainda se corresponde à segunda aproximação, explicada de seguida.

O segundo método de detecção de cabeças, de modo a certificar se o objecto em questão é realmente uma pessoa, baseia-se na verificação da forma elíptica da zona da cabeça resultante da detecção efectuada pelo método anterior. Usando apenas o primeiro método, um alvo que tenha uma forma como a que está na Figura 14 poderia ser considerado uma cabeça.

Figura 14 – Figura com a forma de um sigma, no entanto a figura não representa uma cabeça de uma pessoa.

De modo a impedir que tal aconteça, é verificado se a zona do sigma tem a forma elíptica. Para tal é criada uma elipse preenchida, de cor branca, com as mesmas medidas do rectângulo formado pela zona envolvente do sigma e procede-se a um método designado por *template matching*. Este método caracteriza-se por procurar uma imagem (*template*, neste caso a elipse) numa outra imagem (a zona do sigma) para verificar a existência da primeira. De modo a verificar a existência da elipse na zona da cabeça é verificado quantos pixeis brancos da elipse correspondem a pixeis brancos da zona da cabeça proveniente da imagem binária da subtracção de fundo. O número de pixeis que correspondem sobre o número de pixeis que não é comparado com um limiar e caso seja superior, então é porque de facto a zona que provém da primeira aproximação tem a forma elíptica sendo muito provavelmente uma cabeça de uma pessoa. É de salientar que as dimensões da imagem *template* tem de ser igual ou menor que a imagem onde procurar e, devido à função do OpenCV responsável por esta função, ambas têm de estar em níveis de cinzento.

3.3.3 Junção de resultados de detecção de pessoas

De modo a propocionar uma maior robustez na detecção de pessoas, junta-se os dois algoritmos acima referidos (análise de cor e detecção de cabeças), uma vez que caso a detecção de cabeças falhe por via das condições já assinaladas, a análise de cor possa corrigir essas falhas. No entanto o facto de juntar estes dois algoritmos por disjunção (um dos algoritmos pode dar um resultado falso, e o outro é obrigado a retornar verdadeiro) permite ao sistema abranger mais alvos, assinalando assim objectos como sendo pessoas, isto é, falsos positivos. Por outro lado, se a junção dos algoritmos for pela via da intersecção (ambos os algoritmos têm que dar resultados positivos), dá origem à não detecção de pessoas quando elas realmente estão lá, ou seja, falsos negativos. Ambas as vias produzem falhas, mas a junção pela via da intersecção é a melhor opção no sentido de ser mais provável detectar apenas pessoas, desprezando os objectos que tenham a cor de pele ou a forma de uma cabeça humana.

A implementação inclui as duas fases, a subtracção de fundo e a detecção de pessoas e os respectivos algoritmos que foram explicados no presente capítulo. A implementação efectuada inclui um menu onde existe a opção do utilizador escolher qual dos métodos da subtracção de fundo a executar. Ainda há outros parâmetros (como os limiares e as áreas mínimas dos alvos) que o utilizador pode impor, parâmetros estes que influenciam os resultados finais e cujas influências serão demonstradas no capítulo seguinte.

3.4 Identificação dos alvos em movimento

Sempre que numa imagem é detectado um alvo em movimento, ele é assinalado por um rectângulo envolvente e por um número de identificação. A cor do rectângulo depende de o alvo apresentar alguma característica que o identifique como sendo uma pessoa, como se explica na Secção 4.2. O número é usado para indexar uma tabela onde é apresentada informação sobre os alvos detectados, nomeadamente, o seu centro de massa.

Para a identificação de BLOBs usou-se uma biblioteca designada por **cvBlobsLib**. Esta biblioteca contém diversas funções para seleccionar os BLOBs de interesse, por exemplo, somente incluir aqueles que contenham um número de pixeis acima de um determinado valor, ou colorir BLOBs, identificá-los, entre outras. Para cada BLOB é necessário que sejam calculados os seus limites espaciais mínimos e máximos, horizontalmente e verticalmente. Com estes dados é possível criar um rectângulo que englobe o alvo inteiro. Ainda é efectuada a contagem de BLOBs de modo a colocar um número em cima, necessário para a sua identificação. No sistema, caso o alvo em movimento seja uma pessoa, ou seja, tenha pixeis com a cor de pele e no topo do BLOB haja a forma de uma cabeça, então o rectângulo será desenhado a verde. Caso apresente apenas uma das duas características acima referidas, o rectângulo envolvente será desenhado a amarelo, ao passo que caso nenhuma das características esteja presente, o objecto terá o seu rectângulo a vermelho. Para que a identificação dos BLOBs seja bem representada, ao lado da janela onde são apresentados os resultados da detecção de pessoas está uma outra janela com informação relativa aos objectos, neste caso, o seu centro de massa.

3.5 Implementação final

Finalizando a descrição da implementação do sistema, é apresentada uma imagem que resume, de forma esquemática, o sistema desenvolvido.

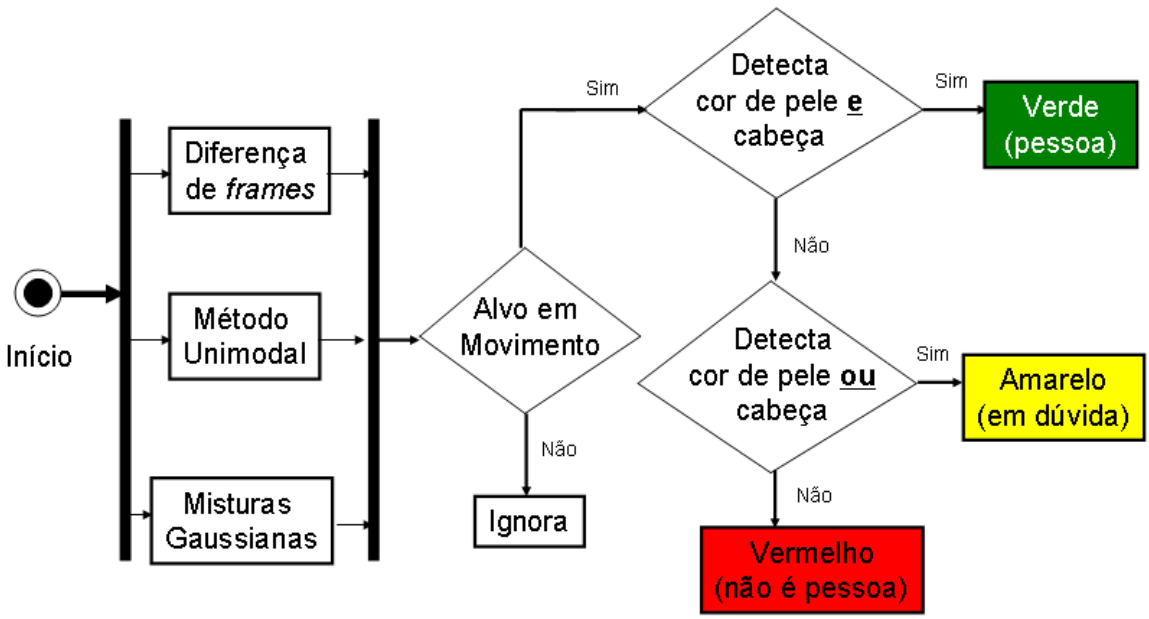

Figura 15 – Esquema do sistema implementado.

Como é perceptível através da Figura 15, no início da aplicação foi implementado um menu onde é pedido ao utilizador para escolher qual dos três métodos de subtracção de fundo será aplicado. Depois, após obter os alvos em movimento (desprezando o fundo estático), são a estes que se aplicam os métodos de análise de cor e de detecção de cabeças, de modo a verificar se os alvos em movimento são de facto pessoas ou não. Aos alvos que forem conotados como pessoas são aplicados rectângulos verdes. No entanto, se somente um dos métodos de detecção de pessoas retornar verdadeiro, ou seja, se somente detectar o tom de pele humano, ou detectar uma cabeça humana, então aplica um rectângulo amarelo à volta do alvo em movimento. Caso o sistema não consiga detectar tom de pele humana nem a forma da cabeça, então o alvo não tem nenhuma característica humana, colocando assim um rectângulo vermelho à volta do alvo em movimento.

Capítulo 4

Resultados experimentais

A aplicação desenvolvida processa uma sequência de imagens previamente gravadas ou directamente provenientes de uma câmara, e assinala os alvos em movimento com rectângulos cuja cor depende de o alvo ser uma pessoa ou não. No início da aplicação é dada ao utilizador a possibilidade de escolher qual a entrada de vídeo a processar (vídeo pré-gravado ou câmara) e o tipo de subtracção de fundo que deseja que a aplicação execute. De seguida, o utilizador deve fornecer os valores de diversos parâmetros que controlam o funcionamento da aplicação. Como por exemplo os valores de alguns limiares referidos no capítulo anterior, assim como o tamanho mínimo (em número de pixeis) que os alvos em movimento devem ter.

Neste capítulo serão apresentados os resultados obtidos em cada uma das partes que compõem o sistema, e será analisado o modo como esses resultados são influenciados pela escolha dos parâmetros da aplicação.

4.1 Subtracção do fundo

Os três métodos implementados apresentam diferentes resultados, o que influencia a sua robustez e consequente utilidade num produto final. O resultado genérico dos três métodos implementados, obtido após a subtracção do fundo é uma sequência de imagens binárias, em que a cor branca representa movimento e a cor preta o fundo estático.

De seguida são apresentados os resultados obtidos por cada método e a influência de alguns parâmetros nesses resultados.

4.1.1 Diferença de *frames*

Na diferença de *frames*, o resultado obtido provêm da diferença directa entre dois *frames* da sequência recebida. Caso essa diferença seja maior que o limiar (*threshold*) indicado pelo utilizador, significa que há movimento, caso seja inferior então é fundo estático. Como foi referido anteriormente, a subtracção do fundo por este método não é robusta, no sentido de se basear somente na subtracção das intensidades dos pixeis na mesma posição em imagens

consecutivas, sem usar qualquer informação acerca do fundo. Problemas como a sobreposição ou o arrastamento do alvo em movimento, causados pela escolha da imagem anterior a subtrair, tornam o resultado pouco robusto.

O resultado é muito variável consoante o limiar imposto pelo utilizador. Limiares muito altos levarão a que partes dos alvos, relevantes para a fase da detecção de pessoas, sejam ignoradas, ao passo que limiares baixos levarão à detecção de movimentos que não tem interesse para a aplicação e também falsos movimentos (mudanças de intensidade que não representam movimento). As Figuras 16a e 16b mostram as imagens utilizadas na diferença, em baixo as Figuras 17a e 17b mostram o resultado da diferença comparado com limiares diferentes.

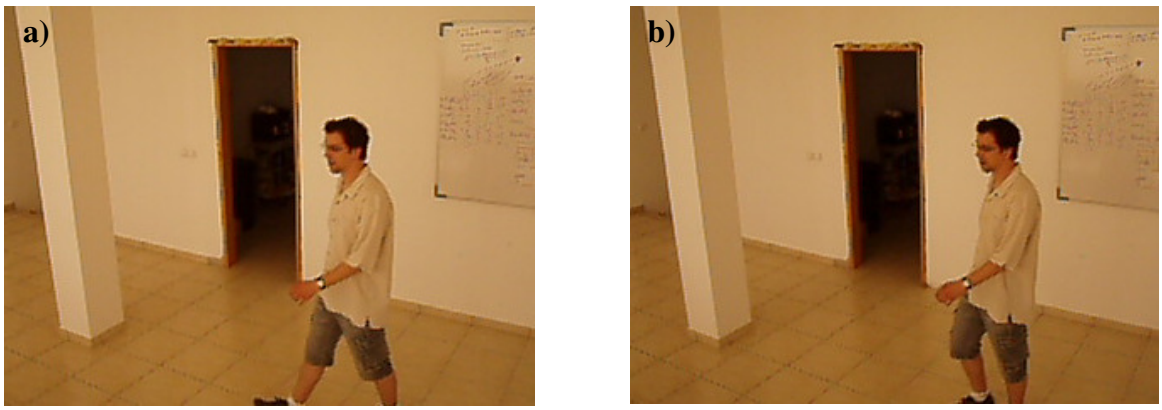

Figura 16 – Imagens utilizadas no método de diferença de *frames*: a) Imagem mais recente, primeiro operando da diferença b) Penúltima imagem, segundo operando da diferença.

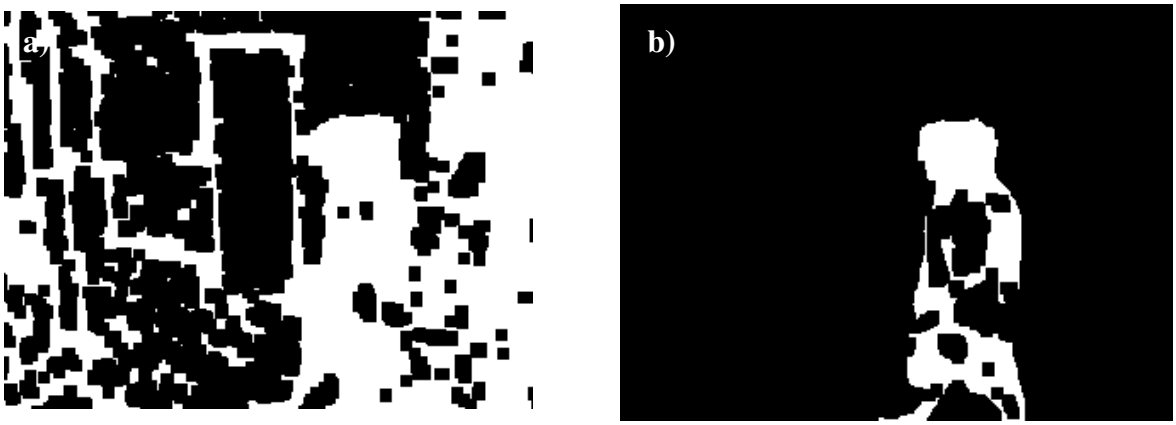

Figura 17 – Resultado da diferença entre as Figuras 16a e 16b: a) Utilizando o limiar igual a 10; b) Utilizando o limiar igual a 30.

Como se pode ver na Figura 17a, quando o limiar é muito baixo (10, neste caso), o método em questão considerou objectos que estão parados como estando em movimento, já que pequenas oscilações de intensidade são imediatamente conotadas como movimento. Por outro lado, na Figura 17b, o que acontece é o contrário, causado pelo facto do valor do limiar tolerar uma elevada diferença entre intensidades, corroendo a imagem do alvo em movimento.

O problema que torna o método pouco robusto é o facto de acarretar situações de sobreposição dos alvos em movimento ou de desfasamento. Ou se opta pela utilização da segunda imagem mais recente e há sobreposição (Figura 18a), ou imagens um pouco mais antigas e ocorre o arrastamento do alvo em movimento (Figura 18b). No primeiro caso, as intensidades das imagens são semelhantes, pois o movimento do alvo de uma imagem para a outra foi quase nulo. Esta semelhança provoca que o interior dos alvos em movimento fique muito "corroído", pois nessa zona as intensidades são semelhantes. Uma solução para evitar este efeito seria subtrair a imagem mais recente à terceira ou quarta mais recente, em vez da segunda mais recente, mas isso iria trazer um maior arrastamento dos alvos em movimento, pois a não sobreposição dos alvos de imagens consecutivas é substituída pelo afastamento dos mesmos. As Figuras 18a e 18b mostram a subtracção de imagens entre a mais recente e a anterior e entre a mais recente e a quarta mais recente.

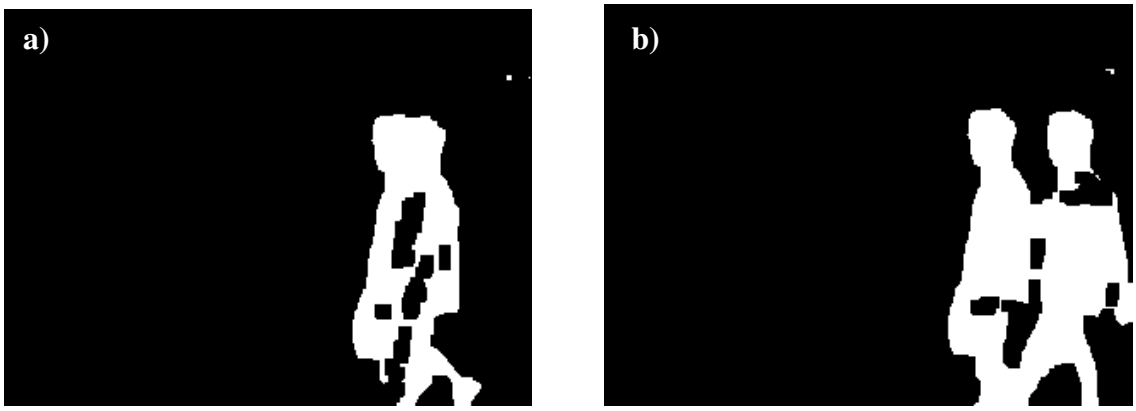

Figura 18 – Método em questão utilizando diferentes *frames*: a) Imagem mais recente e a segunda mais recente (sobreposição); b) Imagem mais recente e a quarta mais recente (arrastamento).

No entanto, há um método para minimizar os efeitos negativos da sobreposição que ocorre no caso da Figura 18a, o *floodfill*. Este método muda a intensidade dos aglomerados de pixeis pretos que estão rodeados de pixeis brancos, para a intensidade branca. Desta forma, as zonas do interior dos alvos que estão a preto ficam preenchidas a branco, não havendo assim a corrosão do interior (ver Figuras 19a e 19b).

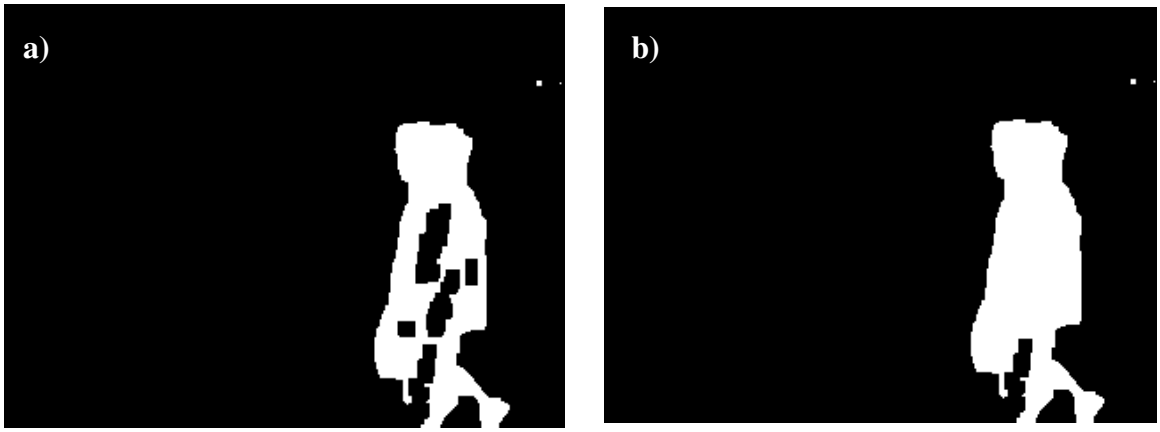

Figura 19 – Influência do método *FloodFill* aplicado ao resultado final do método da diferença de *frames*: a) Sem *FloodFill*; b) Com *FloodFill*.

A câmara é outro factor que influencia o resultado deste método. Por vezes a câmara faz uma adaptação à iluminação existente no cenário que está a filmar, o que leva a variações de intensidade entre *frames* consecutivos, variações essas que não resultam da existência de movimento (Figura 21).

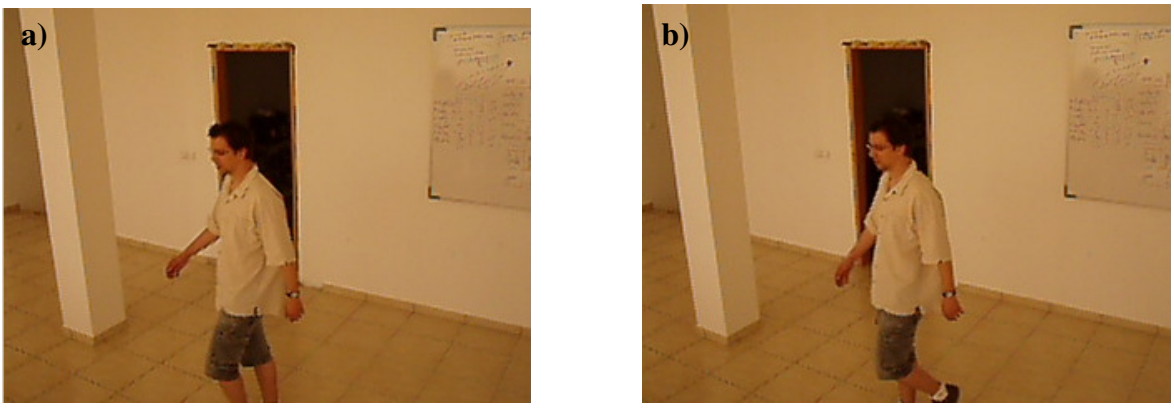

Figura 20 – Aplicação do método da diferença de *frames*: a) Imagem mais recente e mais iluminada b) Penúltima imagem menos iluminada.

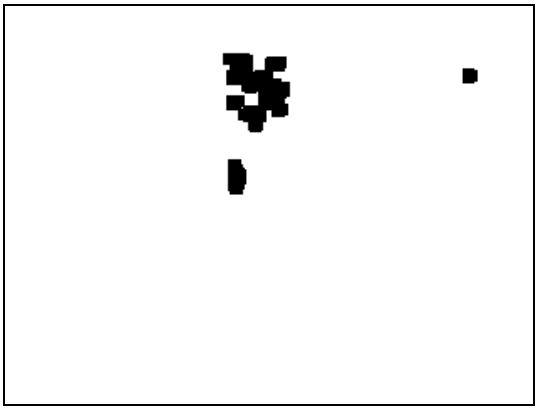

Figura 21 – Resultado da diferença das imagens das Figuras 20a e 20b.

Como é possível verificar na Figura 21 a adaptação à luminosidade ambiente tornou todas as intensidades da imagem mais recente muito diferentes das intensidades da imagem anterior, excepto numa zona a preto que representa uma entrada de uma porta. Havendo essa grande diferença de intensidades causada pela adaptação da câmara, o sistema interpreta que houve movimento, o que na realidade não aconteceu. Isto deveu-se ao facto da Figura 20a ser mais iluminada do que a Figura 20b, o que levou ao sistema a considerar que houve movimento por todo o cenário.

Para que a aplicação desenvolvida possa ser utilizada com câmaras que captam imagens em tempo real, o tempo de processamento é um factor crucial. Uma das vantagens do método da diferença de *frames* é que consegue fazer a subtracção de fundo num tempo médio de 4 milissegundos, o que é muito aceitável para a inclusão num sistema em tempo real.

4.1.2 Unimodal

O método unimodal apresenta melhores resultados que o método anterior, sendo o conhecimento e aprendizagem dos valores respeitantes ao fundo estático factores importantes para uma correcta subtracção de fundo. Estando os valores de intensidades do fundo compreendidos na gama [μ-σ, μ+σ] de cada pixel, o método verifica se cada valor de intensidade de cada pixel recebida proveniente de uma nova imagem está contido nessa mesma gama.

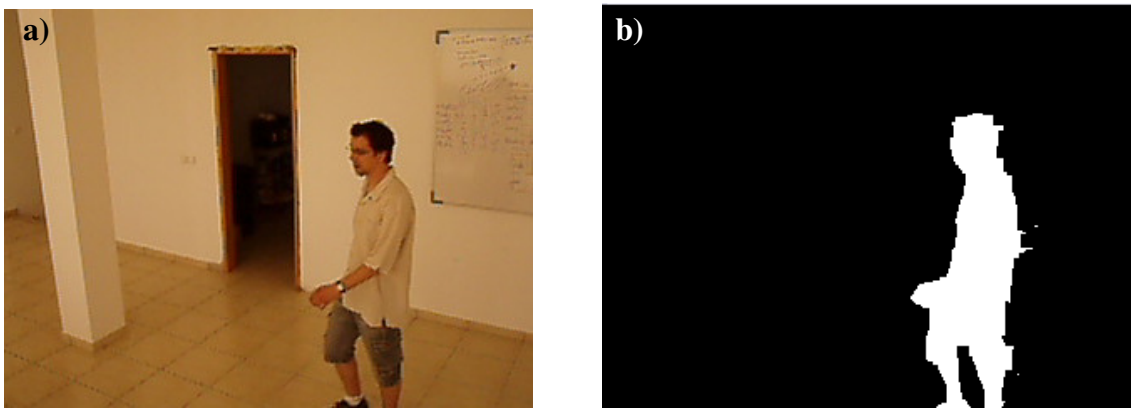

Figura 22 – Aplicação do método unimodal: a) Imagem original; b) Resultado final.

A Figura 22b representa uma pessoa em movimento, sendo essa pessoa representada a branco, e o restante cenário a preto. Quando a intensidade de um pixel permanece inalterada ao longo do tempo, o valor do desvio-padrão, σ, fica muito pequeno, o que faz com que pequenas variações de intensidade sejam consideradas como movimento. Para evitar que isto aconteça, no arranque da aplicação, é pedido ao utilizador que indique um limiar, σ_T, que representa o valor mínimo que σ pode tomar. Este limiar permite que quando o desvio-padrão for demasiado pequeno, o sistema passe a usar o limiar em vez do desvio-padrão, para que mudanças ligeiras de intensidade não sejam conotadas como movimento, ficando o cálculo da gama como sendo [$\mu - \max(\sigma, \sigma_T)$, $\mu + \max(\sigma, \sigma_T)$].

À semelhança do que acontece no método da diferença de *frames*, um valor de σ_T muito alto fará com que alvos em movimento importantes sejam considerados como pertencentes ao fundo estático e vice-versa. As Figuras 23a e 23b representam a influência do limiar imposto pelo utilizador no resultado, mostrando o mesmo momento do vídeo nas duas imagens.

Figura 23 – Método unimodal com diferentes σ_T: a) σ_T igual a 5; b) σ_T igual a 30.

O método unimodal não é robusto relativamente a alterações de iluminação do cenário, pois quando estas ocorrem as intensidades do fundo podem não estar contidas nas gamas calculadas. No entanto, o método adapta-se ao cenário pois o método unimodal molda os valores das médias e desvios-padrão de cada pixel mediante a conjugação das novas intensidades com taxas de aprendizagem (α). As taxas de aprendizagem são valores que atribuem relevância aos valores da média e desvio-padrão ou aos novos valores das intensidades, consoante a classificação do pixel (se pertencente ao fundo ou a um alvo em movimento). Caso as novas intensidades representem movimento, essa aprendizagem é menor, ao passo que se for considerado como fundo o valor α dá tanta relevância aos novos valores de intensidade como aos valores da média e do desvio-padrão. Este processo é efectuado deste modo para que o sistema guarde para cada pixel os valores relativos ao fundo estático do próprio pixel. No sistema, quando o pixel é considerado fundo, a taxa de aprendizagem fica com o valor de 0,5. Desta forma os novos valores da média e do desvio-padrão são:

$$\text{Média: } \mu(t) = (1-0{,}5)\times\mu(t-1) + 0{,}5\times I(t) \tag{4.1}$$

$$\text{Desvio-padrão: } \sigma(t) = \sqrt{(1-0{,}5)\times\sigma(t-1)^2 + 0{,}5\times(I(t)-\mu)^2} \tag{4.2}$$

Por outro lado, quando o pixel for considerado como movimento, a aplicação tem que minimizar a sua relevância de modo a preservar os valores que tem, relativos ao fundo estático. Nesse sentido os valores são actualizados de acordo com as fórmulas seguintes:

$$\text{Média: } \mu(t) = (1-0{,}005)\times\mu(t-1) + 0{,}005\times I(t) \tag{4.3}$$

$$\text{Desvio-padrão: } \sigma(t) = \sqrt{(1-0{,}005)\times\sigma(t-1)^2 + 0{,}005\times(I(t)-\mu)^2} \tag{4.4}$$

A taxa de aprendizagem pode influenciar bastante o resultado do método, no sentido em que uma taxa de aprendizagem elevada, quando o pixel é considerado como pertencente a um alvo em movimento, irá incluir muito rapidamente como fundo os alvos em movimentos que param, ao passo que uma taxa de aprendizagem muito baixa irá incluir muito lentamente como fundo estático alvos que estiveram em movimento mas que passaram a estar parados. Em [30], após várias experiências o valor tomado como taxa de aprendizagem foi de 0,0006.

O valor da taxa de aprendizagem quando a intensidade do pixel é avaliada como fundo estático já não irá influenciar muito no resultado final, uma vez que o valor da nova intensidade é muito próximo dos valores que estão guardados, alterando pouco o resultado das médias e dos desvios-padrão.

Outros factores relevantes para os resultados são as operações morfológicas de erosão e dilatação (minimização e maximização, respectivamente) aplicadas às imagens. Estas operações permitem a atenuação do ruído na imagem final, mas por vezes retiram partes do alvo em movimento que são importantes para futuro processamento, como por exemplo, da detecção de cabeças. Através de experiências realizadas, concluiu-se que o que produz melhores resultados são 2 dilatações, seguidas de 3 erosões com uma matriz 3x3. A opção de aplicar primeiramente as dilatações tem como objectivo eliminar os pontos pretos (pixeis considerados como fundo

estático) envoltos de pixeis brancos, que representam movimento, ou seja, para tirar o ruído dentro dos alvos em movimento. No entanto isto leva a um aumento do ruído no fundo estático, isto é, pixeis isolados considerados como movimento sofrem dilatação, expandindo-se. Assim são executadas três erosões para eliminar esse mesmo ruído do fundo, evitando a expansão dos pixeis pretos dentro dos alvos em movimento, pois estes já foram eliminados, à partida, nas duas dilatações anteriores.

Em relação ao tempo de execução, o método unimodal exige mais tempo de processamento que o método anterior, rondando os 30 milissegundos por imagem.

4.1.3 Misturas Gaussianas

A lógica e funcionamento das misturas Gaussianas são semelhantes ao método unimodal, consequentemente os resultados são similares.

A fase de treino deste método exige mais tempo que a mesma fase no método unimodal, devido à necessidade de determinação dos valores das médias e desvios-padrão das múltiplas distribuições Gaussianas. Inicialmente, como a primeira distribuição tem mais peso do que as restantes, todos os valores de intensidades irão influenciar somente a primeira distribuição, mas à medida que há pequenos movimentos, as intensidades que a representam vão para as outras distribuições Gaussianas, e isso consome recursos.

O método das misturas Gaussianas é tanto mais robusto quanto mais favorável for a fase de treino. Por exemplo, num cenário fechado onde durante a fase de treino não se acendeu uma lâmpada, mas depois, na fase de visualização, tal acontece, ocorre o facto de não haver no sistema uma distribuição Gaussiana que contenha os valores das médias e dos desvios-padrão referentes ao mesmo cenário com a lâmpada acesa. Neste caso, acontece o mesmo que no método unimodal, o sistema considera que tudo é movimento pois as intensidades dos pixeis saíram da gama de intensidades permitidas para o fundo. Mas ao contrário do que acontece com o método unimodal, o facto de haver mais distribuições Gaussianas permite que essa informação da variação das intensidades fique registada numa das distribuições, podendo assim o sistema ser robusto quando voltar a haver mudanças de iluminação no cenário. Portanto, quanto mais situações que não representam movimento relevante (mudança de iluminação, pequenos movimentos oscilatórios) ocorrerem na fase de treino, melhor será a aprendizagem e mais robusto ficará o sistema, pois as várias distribuições Gaussianas ficam com essa informação.

Outro factor de grande relevo para o bom funcionamento deste método é o número de distribuições Gaussianas. Quantas mais distribuições existirem, mais robusto fica o sistema, uma vez que irá permitir uma maior abrangência de intensidades consideradas como fundo, mas por outro lado, exige mais processamento.

Nas Figuras 24 a 28 são apresentados os resultados da fase de treino do método em estudo.

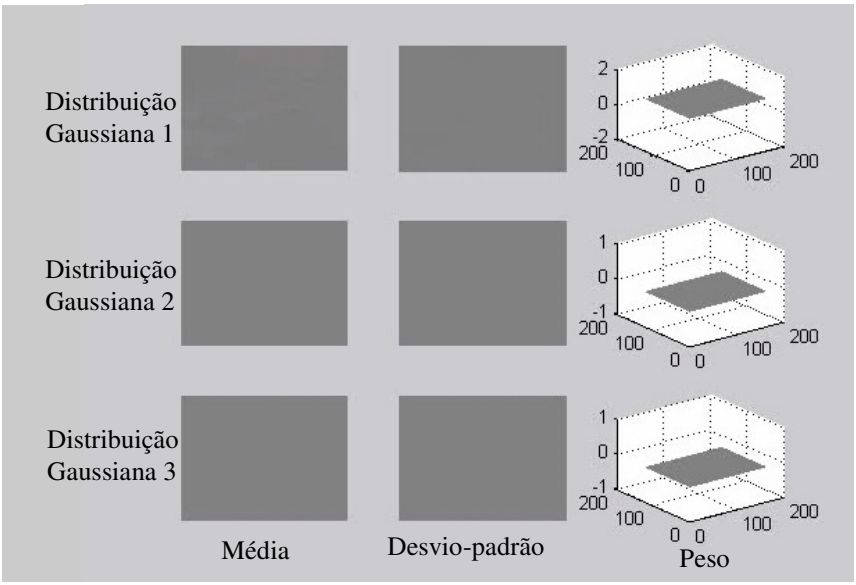

Figura 24 – Fase de treino com 3 distribuições na fase inicial. O peso da primeira distribuição Gaussiana a 1, e das restantes a 0. Os valores das médias, e desvios-padrão das distribuições Gaussianas são 128.

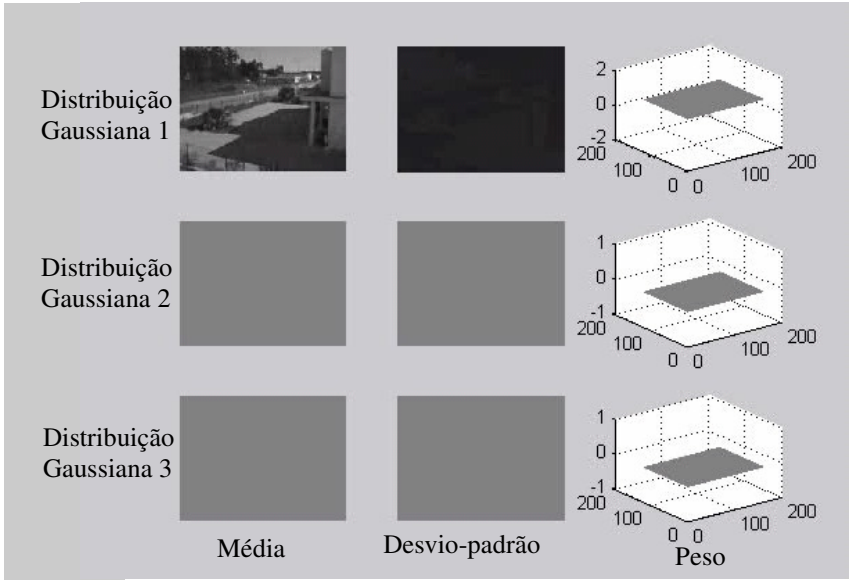

Figura 25 – Fase de treino, após algum tempo, os valores das médias, desvios-padrão das primeiras distribuições Gaussianas dos pixeis já foram actualizados.

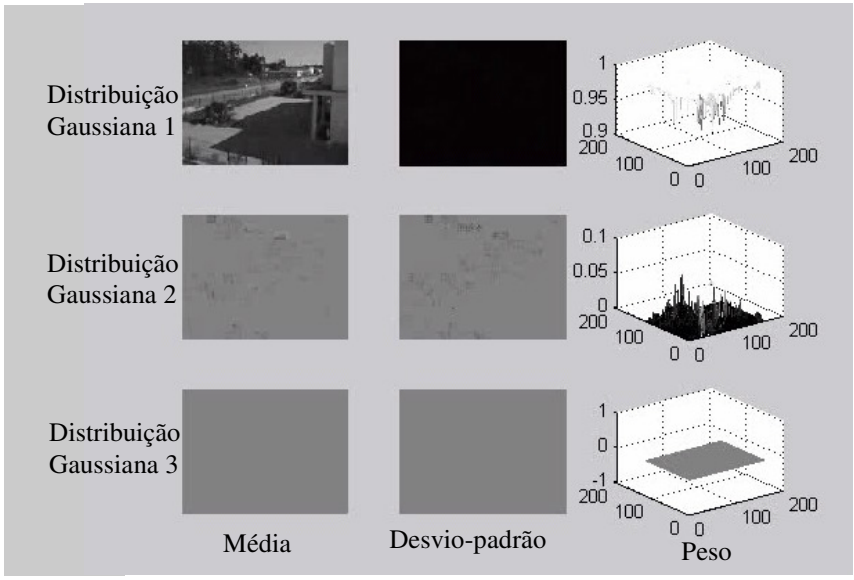

Figura 26 – Fase de treino, na altura quando as intensidades já variaram bastante e já não estão contempladas na primeira distribuição Gaussiana, sendo inseridas na distribuição seguinte (segunda), o peso da primeira Gaussiana deixa de ser 1 em todos os pixeis. A zona da imagem onde há mais movimento (árvores) é a primeira a entrar na segunda distribuição.

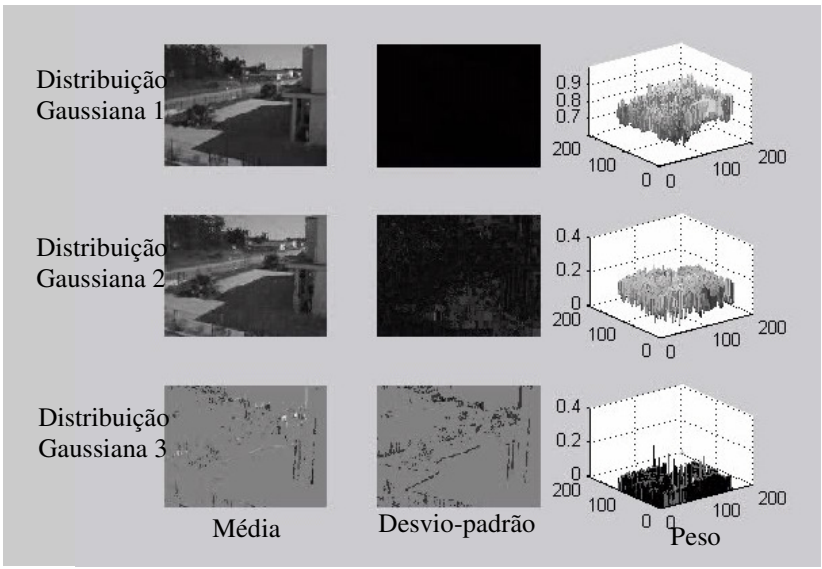

Figura 27 – Continuação dos movimentos, sendo estes captados pela terceira distribuição Gaussiana, pois a segunda já não os abrange (zona das árvores).

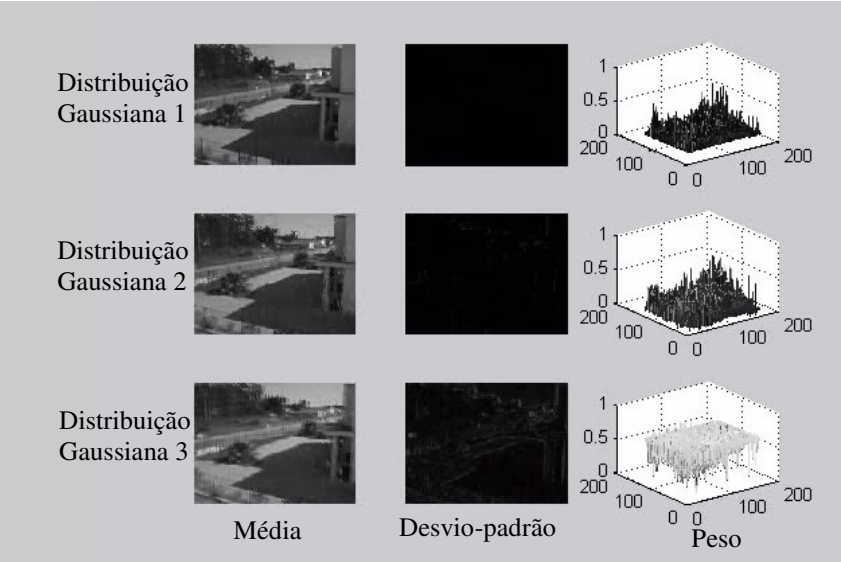

Figura 28 – Situação final da fase de treino, onde as 3 distribuições Gaussianas têm os respectivos valores das médias e desvios-padrão que abrangem as intensidades consideradas como fundo estático. Estando as 3 distribuições preenchidas, significa que para cada pixel há 3 representações do fundo, abrangendo assim movimentos das árvores e mudança de iluminação provocada por nuvens ou movimento do Sol.

É de referir que o ambiente que se utilizou para o exemplo não é interior, esta escolha fundamenta-se pelo facto das variações de iluminação provocadas por nuvens e movimentos de árvores proporcionarem um bom caso de teste para o preenchimento de todas as distribuições durante a fase de treino. Após a fase de treino, a subtracção de fundo é efectuada ignorando as variações de iluminação e os movimentos oscilatórios, resultando assim na obtenção de alvos em movimento que são realmente relevantes para a fase seguinte da detecção de pessoas. Tal como foi referido, caso a fase de treino abranja as variações de iluminação e as pequenas oscilações, haverá um melhor refinamento na obtenção dos alvos em movimento, tornando assim o sistema mais robusto que o mesmo sistema baseado no método unimodal. Mas caso não ocorram essas situações, mas ocorram depois, devido à constante aprendizagem, haverá distribuições Gaussianas que abranjerão essas mesmas alterações.

Como a lógica do método das misturas Gaussianas e do método unimodal têm o mesmo princípio-base, os seus resultados variam de modo semelhante, consoante os valores seleccionados para a taxa de aprendizagem e σ_T, tal como foi explicado na Secção 4.1.2.

Em relação ao tempo de execução, o método das misturas Gaussianas necessita de 340 milissegundos por imagem para detectar os alvos em movimento, algo que restringe a detecção de movimento a todos os *frames*. Este tempo de execução é elevado, e é causado pelas várias verificações no sentido de escolher a melhor distribuição. No entanto os resultados são os melhores entre os métodos até agora implementados.

4.2 Detecção de pessoas

Para a parte da detecção de pessoas foram implementados dois algoritmos: a análise de cor e a detecção de cabeças. Os algoritmos representam abordagens diferentes para o mesmo

propósito da detecção de pessoas, sendo a junção dos resultados de ambos um factor que traz maior robustez ao sistema.

De modo semelhante aos métodos de subtracção de fundo, os métodos para a detecção de pessoas também dependem de alguns parâmetros cujos valores influenciam bastante o resultado final, como se descreve a seguir.

4.2.1 Análise de cor

A detecção de pixeis, cujas intensidades são similares ao tom de pele humana, depende dos limites que são colocados que correspondem ao tom de pele. Quando a intensidade de um pixel é analisada de modo a verificar se corresponde ao tom de pele, esta é comparada com valores que o programador definiu como sendo pertencentes ao tom de pele.

Na aplicação desenvolvida foram efectuadas duas abordagens. A primeira necessitava da conversão para a escala de cores HSV e restringia a componente *Hue* entre os valores 6 e 38 [11]. Já a segunda abordagem utilizava a escala de cores RGB e aplicava as restrições que já foram mencionadas no estado da arte na Secção 2.2.1.

Na Figura 29 é apresentada a imagem original e as Figuras 30a e 30b demonstram o resultado do processamento de detecção do tom de pele, em ambos os métodos implementados.

Figura 29 – Imagem original.

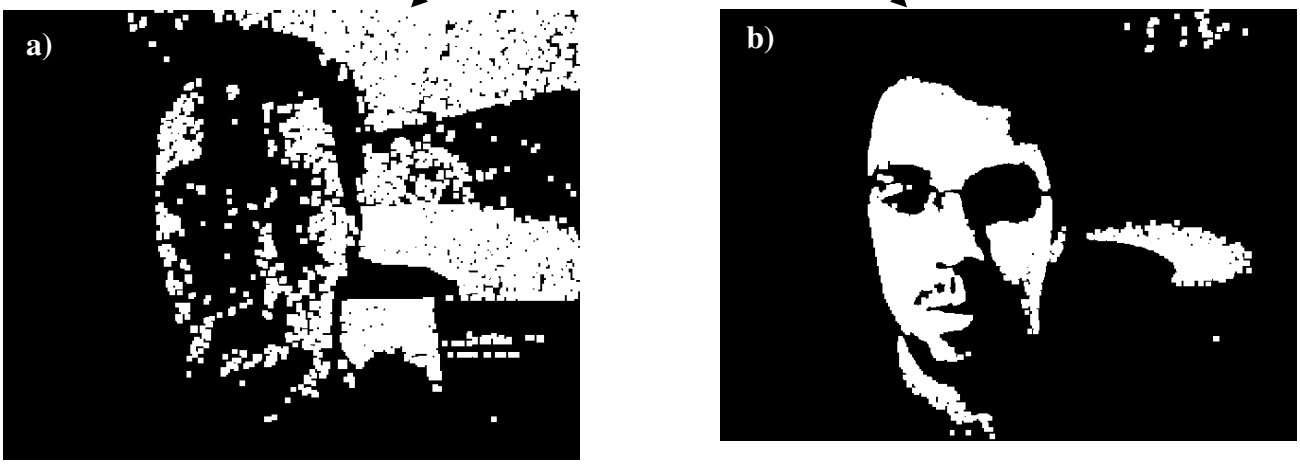

Figura 30 – Resultados finais da aplicação de dois métodos de análise de cor: a) Imagem resultante da primeira abordagem (utilização da escala HSV); b) Imagem resultante da segunda abordagem (utilização da escala RGB).

Como é possível constatar nas Figuras 30a e 30b, os resultados provenientes da segunda implementação são melhores que a primeira, sendo a segunda a abordagem utilizada no sistema. A primeira implementação para além de abranger uma gama alargada de intensidades que vai para além da gama de intensidades representativa do tom de pele, ainda tem o problema de não conseguir abranger as intensidades do tom de pele quando o cenário é muito iluminado. Já a segunda abordagem tem valores que delimitam as intensidades do tom de pele melhor que a primeira abordagem. É de salientar que no sistema implementado, a análise de cor é feita somente aos BLOBs resultantes da fase da subtracção de fundo, desprezando assim pixeis do fundo estático com intensidades do tom de pele (tal como aconteceu na Figura 30b na zona da parede de trás).

4.2.2 Detecção de cabeças

Na detecção de cabeças foram implementados dois métodos: detecção da forma de sigma e verificação da forma elíptica. Em primeira instância é verificado se o topo do alvo em movimento tem a forma de sigma. De seguida, caso esta forma seja detectada, é verificado se essa mesma zona tem a forma elíptica.

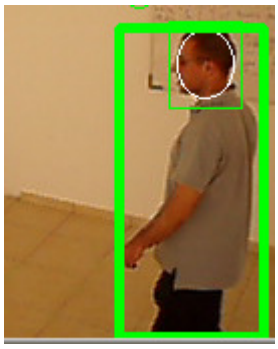

Figura 31 – Detecção da cabeça. O quadrado verde representa a zona do sigma e a forma elíptica é representada pela elipse branca.

Os resultados da detecção de cabeças são influenciados pelos resultados provenientes da fase de segmentação de fundo. Para a detecção da forma do sigma, no caso da segmentação de fundo retirar muitos pixeis que representam a cabeça, pode resultar na não detecção da cabeça, isto porque pode não haver os máximos locais referentes à zona das orelhas e da zona dos ombros.

Figura 32 – Situação em que a detecção de cabeças falhou devido às intensidades do cabelo se assimilarem ao fundo estático.

Pode ocorrer a situação em que o topo do alvo em movimento contenha a forma de um sigma, mas que a forma não seja elíptica. Para isso é feito um *template matching* da zona referente à zona semelhante ao sigma (imagem em escala de cinzento proveniente da subtracção de fundo) com uma imagem de uma elipse de cor branca. O resultado do *template matching* é comparado com um limiar de 0,7 e caso o valor seja igual ou superior, significa que a zona tem a forma elíptica. O valor de limiar referido provém de experiências elaboradas no sentido de descobrir qual o melhor valor, sendo 0,7 o indicado.

Figura 33 – Boa detecção da forma elíptica.

Figura 34 – Má detecção da forma elíptica provocada pela proximidade de intensidade do cabelo com a intensidade da entrada da porta.

O valor do limiar inserido pelo programador influencia o resultado final, sendo um valor baixo muito abrangente (falsos positivos), e um valor muito grande ignora situações onde realmente existem cabeças de pessoas (falsos negativos).

A verificação da existência de uma forma elíptica só é feita caso a forma de sigma tenha sido previamente detectada, No entanto se a forma elíptica não se verificar o alvo em movimento ainda é conotado como sendo pessoa (isto se também tiver pixeis do tom de pele), pois a forma da cabeça pode não ser exactamente elíptica.

No resultado final, caso os dois algoritmos da detecção de pessoas (análise de tom de pele e detecção de cabeças) retornem resultados positivos na detecção, o alvo em movimento é representado por um rectângulo verde envolvente. Ao alvo em movimento é adicionado um número identificador do mesmo. Esse número é colocado no canto superior esquerdo do alvo e consta nas figuras seguintes com o valor 0. Caso apenas um dos algoritmos retornar positivo, o rectângulo envolvente é desenhado a amarelo. Finalmente, na situação em que nenhum dos métodos detecte características humanas, ou seja, o BLOB não contém pixeis com o tom de pele

nem a forma da cabeça, então significa que não é uma pessoa e o rectângulo envolvente é desenhado a vermelho.

Figura 35 - Resultado final da implementação desenvolvida.

Figura 36 - Resultado final da implementação desenvolvida.

As Figuras 35 e 36 representam resultados finais dos métodos implementados responsáveis pela detecção de pessoas em conjunto com a fase da subtracção de fundo. No anexo A estão outros resultados finais obtidos pela implementação desenvolvida.

Capítulo 5

Conclusões e perspectivas de trabalho futuro

No presente capítulo são apresentadas as conclusões relativas ao trabalho desenvolvido no âmbito desta dissertação. São também apresentadas as limitações do sistema para detecção de pessoas desenvolvido e possíveis melhoramentos.

5.1 Conclusões do projecto

Concluído o projecto, verifica-se que o objectivo principal foi cumprido. Foi desenvolvida uma aplicação que permite realizar a detecção de pessoas, em tempo real, numa sequência de imagens provenientes de um vídeo pré-gravado ou, directamente, de uma câmara.

A detecção de pessoas consiste em duas fases principais: a segmentação das imagens adquiridas, tendo em vista a separação do fundo estático dos alvos em movimento que poderão ou não ser pessoas, e a detecção de pessoas propriamente dita que consiste em verificar quais dos alvos em movimento correspondem, efectivamente, a pessoas. Para a subtracção do fundo implementou-se os métodos pretendidos, podendo estes ser escolhidos pelo utilizador. O método que apresenta melhores resultados é o das misturas Gaussianas, por ser robusto a mudanças de iluminação e a pequenos movimentos oscilatórios de árvores ou de objectos como cortinados ou bandeiras. O método unimodal necessita de um fundo muito estático para que seja feita uma boa subtracção do fundo, no entanto a adaptação é concretizada de modo rápido. Já o método da diferença de *frames* é o que apresenta os piores resultados.

Por outro lado, na detecção de pessoas propriamente dita, os resultados dos métodos implementados são muito dependentes das condições de iluminação e do resultado proveniente da fase anterior da subtracção de fundo.

Aos utilizadores da aplicação desenvolvida é dada a possibilidade de, através de alguns parâmetros da aplicação, tendo em conta alguns factores externos como, por exemplo, a

iluminação ambiente, controlar a os resultados obtidos, por forma a serem tão próximos quanto possível dos resultados desejados.

5.2 Limitações

O sistema desenvolvido apresenta, por enquanto, algumas limitações, uma vez que o problema da detecção de pessoas numa sequência de imagens é uma questão complexa.

Na parte de subtracção de fundo, cada método tem as suas limitações, como a susceptibilidade a variações de iluminação, no caso da diferença de *frames* e do método unimodal. Por outro lado, o método das misturas Gaussianas, embora seja mais robusto quanto a este aspecto, consome muitos recursos, nomeadamente tempo de processamento.

Os métodos usados para detectar quais os alvos em movimento que correspondem a pessoas também apresentam algumas limitações. O método de detecção por análise de cor, tendo em vista a detecção de regiões com a cor da pele humana, necessita que as condições de iluminação do cenário sejam favoráveis. Por outro lado, o método baseado na detecção de regiões com a forma de uma cabeça poderá falhar em situações em que a subtracção do fundo não produziu bons resultados. É evidente que, em situações em que as pessoas estiverem muito longe da câmara usada para captar as imagens, qualquer um destes métodos falhará devido às dimensões reduzidas que a região da cabeça terá nas imagens. Estas situações, que podem ser resolvidas com recurso à aquisição de imagens com ampliação (*zoom*), não faziam parte dos requisitos do problema.

Neste trabalho não foram considerados outros problemas que podem limitar a qualidade do resultado final, como sejam a ocorrência de oclusões e a existência de muitas pessoas numa sequência de imagens, como acontece na presença de uma multidão, em que a segmentação das imagens, por forma a separar as pessoas umas das outras é, por si só, um problema muito difícil resolução.

5.3 Perspectivas de trabalho futuro

Não obstante o objectivo inicial ter sido cumprido, é possível melhorar o sistema desenvolvido, tendo em vista a produção de melhores resultados. A fase da detecção de pessoas é a que mais pode ser melhorada no sentido de ser mais eficaz. Por exemplo, a análise de cor poderia ter as suas restrições adaptáveis ao cenário de modo a que não seja prejudicado pelos extremos da iluminação (muito fraca ou muito forte). Há métodos de análise de tom de pele que são capazes de aprender, de modo a identificar os pixeis com tom de pele em várias situações, independentemente do cenário [31, 32]. No entanto é preciso salientar que tais processos são muito demorados e exigem muito processamento, o que iria tornar impossível a execução da aplicação em tempo real.

Outro factor que poderia melhorar o desempenho do sistema seria a detecção de sombras [33] e subsequente eliminação. Na aplicação desenvolvida não há este factor pois optou-se por converter a imagem em escala de cinzento, impedindo assim a aplicação de filtros de detecção de sombras que utilizam a escala de cores de 3 canais.

De modo a tornar a aplicação mais robusta, poderiam ser implementados métodos de seguimento dos alvos em movimento, por exemplo, recorrendo a filtros de Kalman [34], para que haja uma previsão do movimento. A situação de duas ou mais pessoas se cruzarem no mesmo

ponto também não está contemplada na aplicação, sendo a questão de multidões um factor a melhorar futuramente.

Referências

[1] Piccardi M., *Background subtraction techniques: a review*, University of Technology, Sydney, 2004

[2] Hansen D. M., Mortensen B. K., Duizer P. T., Andersen J. R., Moeslund T. B., *Automatic Annotation of Humans in Surveillance Video Recordings*, 1-142, Aalborg, 2006

[3] Haiminen N., Gionis A., *Unimodal Segmentation of Sequences*, 1-8, IEEE Computer Society, Helsinki, 2004

[4] Chalidabhongse T. H., Kim K., Harwood D., Davis L., *A Perturbation Method for Evaluating Background Subtraction Algorithms*, 1-7, IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2003

[5] Mittal A., Paragios N., *Motion-Based Background Subtraction using Adaptive Kernel Density Estimation*, 1-8, Princeton, Champs-sur-Marne, 2004

[6] McIvora A., Zangb Q., Klette R., *The Background Subtraction Problem for Video Surveillance Systems*, 1-14, Auckland, 2000

[7] Cheng J., Yang J., Zhou Y., Cui Y., *Flexible background mixture models for foreground segmentation*, 1-10, Shanghai, 2006

[8] Liem M., *Constructing A Hybrid Algorithm for Tracking and Following People using a Robotic Dog*, 1-89, Universiteit van Amsterdam, Amsterdão, 2008

[9] Zivkovic Z., *Improved Adaptive Gaussian Mixture Model for Background Subtraction*, 1-4, Amsterdão, 2004

[10] Stauffer C., Grimson W.E.L, *Adaptive background mixture models for real-time tracking*, 1-7, Vol. 2, *IEEE Computer Society Conference*, Cambridge, 1999

[11] Zhuo X. L., *Real-Time People Detection*, http://www.sccs.swarthmore.edu/users/01/xianglan/e27/realtime.html, consultado em 28 de Dezembro de 2009

[12] Oliveira V.A., Conci A., *Skin Detection using HSV color space*, 1-2, SIBGRAPI 2009 - XXIInd Brazilian Symposium on Computer Graphics and Image, Niterói, 2009

[13] Kovac J., Peer P., Solina F., *Human skin color clustering for face detection*, EUROCON2003, 144–148, Turku, 2003

[14] Wren C. R., Azarbayejani A., Darrell T., Pentland A. P., *Pfinder: Real-Time Tracking of the Human Body*, IEEE Transactions on pattern analysis and machine intelligence, vol. 29, Nº 7, 1997

[15] Patil R., Rybski P. E., Kanade T., Veloso M. M., *People Detection and Tracking in High Resolution Panoramic Video Mosaic*, 1-6, Pittsburgh, 2004

[16] Pacheco J. N. D., *Tracking People and Activities in Video Recordings of Classroom Presentations*, 1-114, Instituto Superior Técnico de Lisboa, 2009

[17] Haritaoglu I., Harwood D., Davis L. S., *W4: Real-Time Surveillance of People and Their Activities*, 1-22, IEEE Transactions on pattern analysis and machine intelligence, vol. 22, Nº. 8, 2000

[18] Zhao T., Nevatia R., *Bayesian Human Segmentation in Crowded Situations*, 1-8, Los Angeles, 2003

[19] Zhuang Y., Liu X., Pan Y., *Video Motion Capture Using Feature Tracking and Skeleton Reconstruction*, 1-5, University of Hangzhou, Zhejiang, 1999

[20] Kipman A., http://www.eurogamer.net/articles/e3-post-natal-discussion-interview, consultado em 22 de Janeiro de 2010

[21] Dedeoğlu Y., *Human Action Recognition Using Gaussian Mixture Model based Background Segmentation*, 1-9, Ankara

[22] Schneiderman H., *Feature-Centric Evaluation for Efficient Cascaded Object Detection*, 1-8, Pittsburgh, 2004

[23] Ferreira E. H. C., *Detecção e Correção Automáticas de Olhos Vermelhos*, 1-9, Curitiba, 2008

[24] Stauffer C., Grimson W.E.L, *Adaptive background mixture models for real-time tracking*, 1-7, Cambridge, 1999

[25] McIvor A. M., *Background Subtraction Techniques*, In Proc. of Image and Vision Computing, 1-6, Auckland, 2000

[26] Stensmo M., Sejnowski T. J., *A Mixture Model Diagnosis System*, 1-22, SanDiego, 1994

[27] Paalanen P., Kämäräinen J., Ilonen J., Kälviäinen H., *Feature Representation and Discrimination Based on Gaussian Mixture Model Probability Densities. Practices and Algorithms,* 1-25, Lappeenranta, 2005

[28] Reynolds D. A., Quatieri T.F., Dunn. R.B., *Gaussian Mixture Models*, 1-5, Lexington, 2000

[29] Chen Y., *Human Head Detection and Tracking*, 1-8, Ohio

[30] Harville M., Gordon G., Woodfill J., *Adaptive Video Background Modeling Using Color And Depth*, International Conference on Image Processing, 1-4, Palo Alto, 2001

[31] Zhu Q., Cheng K., Wu C., *A Unified Adaptive Approach to Accurate Skin Detection*, 1-4, Santa Barbara, 2004

[32] Jones M. J., Rehg J. M., *Statistical Color Models with Application to Skin Detection*, International Journal of Computer Vision, 1-23, Cambridge, 1999

[33] Horprasert T., Harwood D., Davis L. S.*, A Statistical Approach for Real-time Robust*, 1-19, Maryland, 1999

[34] Welch G., Bishop G., *An Introduction to the Kalman Filter*, 1-16, Department of Computer Science University of North Carolina at Chapel Hill, 2006

Anexos

ANEXO A Resultados finais da implementação

Figura 37 - Resultado final da implementação com uma pessoa.

Figura 38 - Resultado final da implementação com uma pessoa.

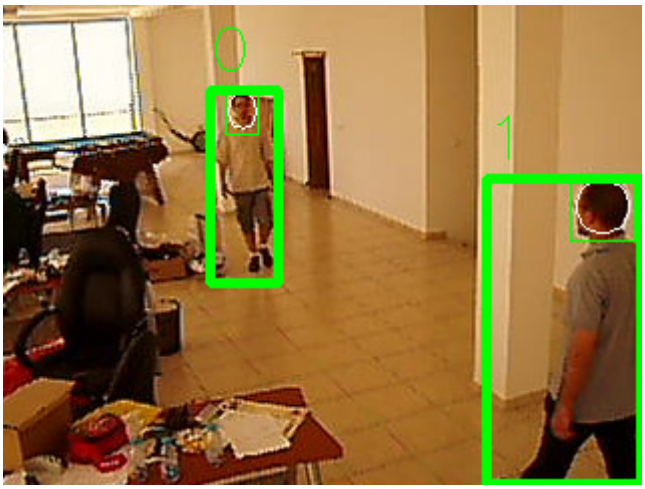

Figura 39 - Resultado final da implementação com duas pessoas. O identificador de cada alvo no canto superior esquerdo está a 0 no alvo do fundo e a 1 no alvo mais próximo da câmara.

Figura 40 - Resultado final da implementação com duas pessoas.

Figura 41 - Resultado final da implementação com uma pessoa.

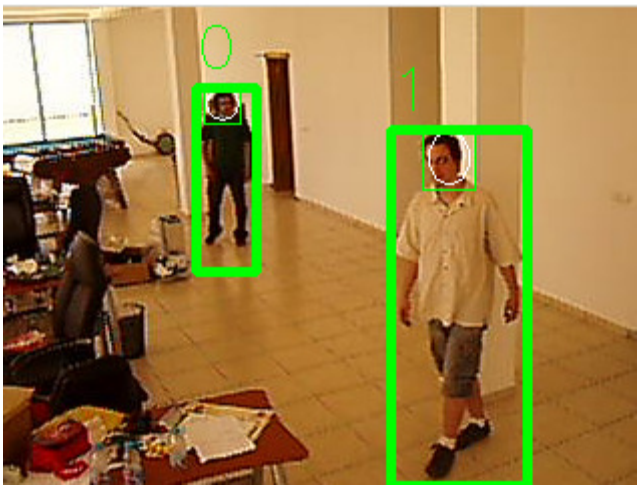

Figura 42 - Resultado final da implementação com duas pessoas.

Figura 43 - Resultado final da implementação com duas pessoas que se cruzam, originando a junção das mesmas num só BLOB.

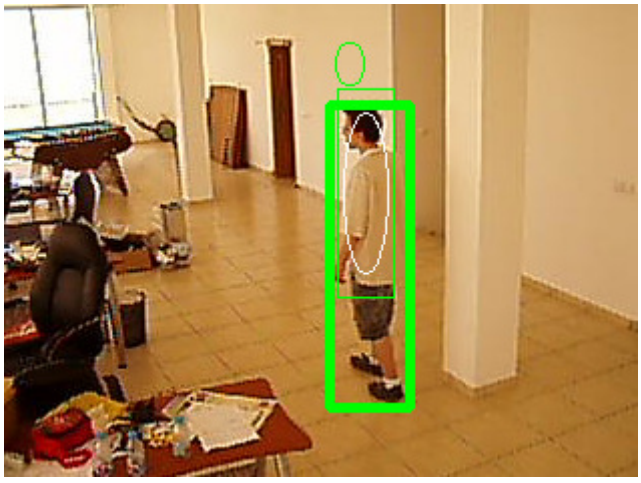

Figura 44 - Resultado final da implementação com problema na detecção de cabeças.

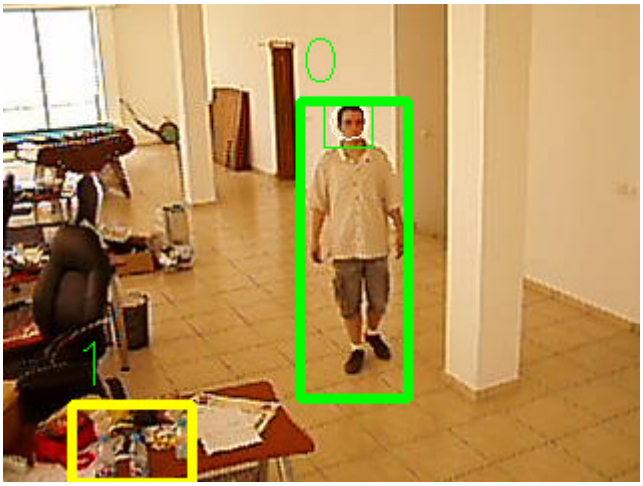

Figura 45 - Resultado final da implementação onde detectou um objecto em movimento em cima da secretária.

Figura 46 - Resultado final da implementação com problema na detecção de cabeças, detectando somente a cor de pele.