



UNIVERSIDADE DA BEIRA INTERIOR  
Faculdade de Engenharia

# Automated Detection of Abandoned Objects in Surveillance Environments

Rui Miguel Coutinho Silva

Dissertação para obtenção do Grau de Mestre em  
**Engenharia Informática**  
(2º ciclo de estudos)

Orientador: Prof. Doutor Hugo Proença

Covilhã, Outubro de 2016



# Agradecimentos

Nesta página manifesto a minha gratidão a todos os que contribuíram e me motivaram durante a realização deste trabalho.

Ao meu orientador Prof. Dr. Hugo Proença, pela constante disponibilidade para ajudar e pela confiança depositada.

Ao SOCIA-Lab e aos seus membros, pela oportunidade de adquirir conhecimentos na área de Visão por Computador.

Ao Prof. Dr. Miguel Guevara Lopez, por toda a força e incentivo prestado.

Ao Centro de Computação Gráfica (CCG), pela flexibilidade de me conceder algum tempo para a elaboração deste documento.

A todos os meus colegas do CVIG (*Computer Vision Interaction and Graphics*).

Aos meus pais, ao meu avô, e à Fernanda.

Obrigado!



# Índice

|   |     |
|---|-----|
| Agradecimentos .....                                  | iii |
| Índice .....  | v   |
| Lista de Figuras .....                                | vii |
| Lista de Tabelas.....                                 | ix  |
| Acrónimos e Estrangeirismos .....                     | xi  |
| 1. Introdução .....                                   | 1   |
| 1.1. Motivação .....                                  | 1   |
| 1.2. Descrição do problema e objetivos .....          | 1   |
| 1.3. Estrutura do documento .....                     | 2   |
| 2. Estado-da-Arte .....                               | 3   |
| 3. Análise do Método de Referência.....               | 15  |
| 3.1. Método de referência .....                       | 15  |
| 3.2. Subtração do plano de fundo .....                | 17  |
| 3.3. Detecção de objetos estáticos .....              | 19  |
| 3.4. Detecção de objetos abandonados e removidos..... | 21  |
| 3.5. Implementação do método de referência.....       | 25  |
| 3.6. Limitações .....                                 | 29  |
| 4. Método Proposto .....                              | 31  |
| 4.1. Pré-Processamento .....                          | 31  |
| 4.2. Subtração do plano de fundo .....                | 32  |
| 4.3. Remoção de Sombras .....                         | 39  |
| 4.4. Seguimento ( <i>Tracking</i> ).....              | 40  |
| 4.5. Classes dos objetos em movimento .....           | 41  |

|      |  |    |
|------|--|----|
| 4.6. | Deteção de automóveis .....                      | 42 |
| 4.7. | Deteção de pessoas .....                         | 43 |
| 4.8. | Deteção de objetos abandonados .....             | 45 |
| 4.9. | Classificação do tipo de objeto abandonado ..... | 49 |
| 5.   | Resultados e Discussão .....                     | 51 |
| 5.1. | Casos de sucesso .....                           | 51 |
| 5.2. | Falhas no método .....                           | 60 |
| 5.3. | Resultados alcançados .....                      | 64 |
| 6.   | Conclusões e Trabalho Futuro.....                | 65 |
| 6.1. | Conclusões.....                                  | 65 |
| 6.2. | Trabalho futuro .....                            | 65 |
|      | Obras Citadas.....                               | 67 |

# Lista de Figuras

|   |    |
|---|----|
| FIGURA 1- ALARME DE OBJETO ABANDONADO E O SEU PROPRIETÁRIO ASSINALADO. (ADAPTADA DE [1]).   | 4  |
| FIGURA 2 - DETEÇÃO DE BAGAGEM ESTÁTICA; O SEGUIMENTO DO PROPRIETÁRIO É INICIADO; QUANDO O PROPRIETÁRIO SAI DE CENA, O ALARME É DISPARADO. (ADAPTADA DE [3]).  | 4  |
| FIGURA 3 - EM (G) E (H) SÃO VISÍVEIS AS MÁSCARAS DE OBJETOS ABANDONADOS GERADAS PELOS DOIS MÉTODOS PROPOSTOS. (ADAPTADA DE [4]).  | 5  |
| FIGURA 4 - EXEMPLO DAS CARACTERÍSTICAS EXTRAÍDAS DOS OBJETOS. (ADAPTADA DE [6]).  | 5  |
| FIGURA 5 - A RIQUEZA DAS CORES ( $N_{CR}$ ) NA IMAGEM ATUAL (I) E NO <i>BACKGROUND</i> ( $B^1$ ). QUANDO ACONTECE UM EVENTO DE ABANDONO A RIQUEZA DAS CORES EM I É MAIOR DO QUE EM $B^1$ . PELO CONTRÁRIO, QUANDO UM OBJETO É REMOVIDO $B^1$ TEM CORES MAIS RICAS DO QUE I. (ADAPTADA DE [7]).                            | 6  |
| FIGURA 6 - O RESULTADO DA DETEÇÃO DE UM OBJETO ESTÁTICO. (ADAPTADA DE [8]).   | 7  |
| FIGURA 7 - DISTINÇÃO ENTRE OBJETO ABANDONADO E REMOVIDO. (A) ARESTAS DO <i>BLOB</i> . (B) ARESTAS DO <i>BLOB</i> DILATADAS. (C) ARESTAS DETETADAS NA IMAGEM. (D) CONTÉM O RESULTADO DA OPERAÇÃO DE AND BINÁRIO ENTRE (B) E (C). (ADAPTADA DE [9]).  | 7  |
| FIGURA 8 - DISTINÇÃO ENTRE PESSOAS ESTÁTICAS E OBJETOS ABANDONADOS. (ADAPTADA DE [10]).   | 8  |
| FIGURA 9 - EXEMPLO DA DETEÇÃO DE OBJETO ABANDONADO E DETEÇÃO DO PROPRIETÁRIO. (ADAPTADA DE [11]).   | 9  |
| FIGURA 10 - EXEMPLO DA ANÁLISE DA UNIÃO OU SEPARAÇÃO DE <i>BLOBS</i> PARA TENTAR INFERIR SE O OBJETO FOI ABANDONADO OU ROUBADO. (ADAPTADA DE [15]).   | 10 |
| FIGURA 11 - GRÁFICO DO FLUXO GERAL DO SISTEMA. UM OBJETO ABANDONADO TENDE A TER PONTUAÇÕES ELEVADAS DO SEU NÍVEL ESTÁTICO (S), DE DISTINÇÃO COM O <i>BACKGROUND</i> (F), E DE ABANDONO (A). ESTAS PONTUAÇÕES SÃO DEPOIS PASSADAS A UM CLASSIFICADOR QUE PRIORIZA OS ALERTAS PELA SUA IMPORTÂNCIA (I). (ADAPTADA DE [16]). | 11 |
| FIGURA 12 - DIAGRAMA DO MÉTODO PROPOSTO EM [20]. (ADAPTADA DE [20]).  | 12 |
| FIGURA 13 - DETEÇÃO DE OBJETOS ESTÁTICOS UTILIZANDO R-RANSAC. (ADAPTADA DE [21]).   | 13 |
| FIGURA 14 - <i>FRAMEWORK</i> PARA DETEÇÃO DE OBJETOS ABANDONADOS. (ADAPTADA DE [22]).   | 16 |
| FIGURA 15 - MÉTODO DE <i>REGION GROWING</i> PARA DETEÇÃO DE OBJETO ABANDONADO. (ADAPTADA DE [22]).  | 24 |
| FIGURA 16 - EXEMPLO DA DETEÇÃO DE UM OBJETO ABANDONADO. (ADAPTADA DE [22]).   | 25 |
| FIGURA 17 - MODELO DO PLANO DE FUNDO ESTIMADO, IMAGEM DE ENTRADA E RESPECTIVA MÁSCARA DE SEGMENTAÇÃO.   | 26 |
| FIGURA 18 - RESULTADO DA APLICAÇÃO DO DESCRITOR DE SIMILARIDADE LOCAL SOBRE O PLANO DE FUNDO, E MÁSCARA COM AS ARESTAS DESSAS ZONAS.  | 27 |
| FIGURA 19 - RESULTADO DA APLICAÇÃO DO DESCRITOR DE SIMILARIDADE LOCAL SOBRE A IMAGEM DE ENTRADA, E MÁSCARA COM AS ARESTAS DESSAS ZONAS.   | 27 |
| FIGURA 20 - MÁSCARA DA REGIÃO ESTÁTICA E OS SEUS CONTORNOS.   | 28 |
| FIGURA 21 - REGIÃO SEGMENTADA DO PLANO DE FUNDO EM FUNÇÃO DO CENTROIDE DA MÁSCARA DA REGIÃO ESTÁTICA.   | 28 |
| FIGURA 22 - REGIÃO SEGMENTADA DA IMAGEM DE ENTRADA EM FUNÇÃO DO CENTROIDE DA MÁSCARA DA REGIÃO ESTÁTICA.  | 29 |
| FIGURA 23 - EXEMPLO DA APLICAÇÃO DOS MÉTODOS DE BGS NO <i>DATASET</i> CDNET 2012 [37]. (A) <i>GROUND TRUTH</i> PARA A <i>FRAME</i> ATUAL; (B) RESULTADOS SUBSENSE [36]; (C) RESULTADOS GMM [5]. (ADAPTADA DE [36]).   | 33 |
| FIGURA 24 - EXEMPLO DA APLICAÇÃO DOS MÉTODOS DE BGS NO <i>DATASET</i> CDNET 2014 [38]. (A) <i>GROUND TRUTH</i> PARA A <i>FRAME</i> ATUAL; (B) RESULTADOS SUBSENSE [36]; (C) RESULTADOS GMM [5]. (ADAPTADA DE [36]).   | 34 |
| FIGURA 25 - IMAGEM DE ENTRADA.  | 35 |
| FIGURA 26 - MODELO ESTIMADO DO PLANO DE FUNDO.  | 35 |

|  |    |
|--|----|
| FIGURA 27 - SEGMENTAÇÃO DOS PIXELS DETETADOS COMO PRIMEIRO PLANO.....  | 35 |
| FIGURA 28 - ESQUEMA DO SUBSENSE. (ADAPTADA DE [36]). .....   | 36 |
| FIGURA 29 - NO CANTO INFERIOR DIREITO, MÁSCARA RESULTANTE DA APLICAÇÃO DO MÉTODO DE REMOÇÃO DE<br>SOMBRA [39]. .....                           | 39 |
| FIGURA 30 - LINHAS AMARELAS REPRESENTAM A LEITURA REAL DA POSIÇÃO. LINHAS AZUIS REPRESENTAM A<br>ESTIMATIVA DO FILTRO DE <i>KALMAN</i> . ..... | 40 |
| FIGURA 31 - <i>PIXELS</i> DETETADOS COMO PRIMEIRO PLANO PARA O EXEMPLO DA FIGURA 30.....   | 41 |
| FIGURA 32 - EXEMPLO DA DETECÇÃO DE AUTOMÓVEIS. ....  | 42 |
| FIGURA 33 - CLASSIFICADOR DETETA COM SUCESSO O AUTOMÓVEL COM ID 2 DA FIGURA 32.....  | 42 |
| FIGURA 34 - RESULTADO DA DETECÇÃO DE AUTOMÓVEIS NO DATASET AVSS PV MEDIUM. ....  | 43 |
| FIGURA 35 - EXEMPLO DA DETECÇÃO DE PESSOAS. ....   | 44 |
| FIGURA 36 - ALGORITMO HOG - HISTOGRAMS OF ORIENTED GRADIENTS [14]. .....   | 44 |
| FIGURA 37 - EXEMPLO DA DETECÇÃO DE UM GRUPO DE PESSOAS COM O CLASSIFICADOR <i>HEAD AND SHOULDERS</i> . ..                                      | 45 |
| FIGURA 38 - ALARME DE MALA ABANDONADA EM ESTAÇÃO DE METRO - <i>DATASET PETS 2006</i> . ....  | 46 |
| FIGURA 39 - MÁSCARAS DE PRIMEIRO PLANO CAPTURADAS A CADA 30 SEGUNDOS, QUE ORIGINARAM O ALARME DA<br>FIGURA 38. ....                            | 47 |
| FIGURA 40 - MÁSCARA DE OBJETO ABANDONADO OBTIDA APÓS ANÁLISE DAS MÁSCARAS DE PRIMEIRO PLANO DA<br>FIGURA 39. ....                              | 48 |
| FIGURA 41 - RESISTÊNCIA A OCLUSÕES - <i>DATASET PETS 2006</i> . ....   | 48 |
| FIGURA 42 - AS ETAPAS DO PROCESSO R-CNN PARA DETECÇÃO DE OBJETOS UTILIZADO PELO CAFFE [45]. ....   | 49 |
| FIGURA 43 - IMAGENET: SUB-CLASSES DA CLASSE "BAG" (SACO). .....  | 50 |
| FIGURA 44 - DETECÇÃO DE MALA ABANDONADA - <i>DATASET AVSS AB EASY</i> . ....   | 51 |
| FIGURA 45 - MÁSCARAS DE <i>FOREGROUND</i> - <i>DATASET AVSS AB EASY</i> . ....   | 52 |
| FIGURA 46 - MÁSCARA DE OBJETO ABANDONADO - <i>DATASET AVSS AB EASY</i> . .....   | 52 |
| FIGURA 47 - DETECÇÃO DE AUTOMÓVEL ESTACIONADO (ABANDONADO) EM LOCAL PROIBIDO - <i>DATASET AVSS PV<br/>MEDIUM</i> . .....                       | 53 |
| FIGURA 48 - MÁSCARA DE OBJETO ABANDONADO DA FIGURA 47. ....  | 53 |
| FIGURA 49 - DETECÇÃO DE MALA ABANDONADA - <i>DATASET PETS 2007</i> . .....   | 54 |
| FIGURA 50 - FRAME POSTERIOR AO ABANDONO DO OBJETO COM ID 23 - <i>DATASET ABODA VIDEO 01</i> . ....   | 54 |
| FIGURA 51 - DETECÇÃO DE OBJETO ABANDONADO - <i>DATASET ABODA VIDEO 01</i> . ....   | 55 |
| FIGURA 52 - <i>FRAME</i> EM QUE OCORRE O ABANDONO DA MALA - <i>DATASET ABODA VIDEO 02 DATASET</i> . ....                                       | 56 |
| FIGURA 53- ALARME DE OBJETO ABANDONADO - <i>DATASET ABODA VIDEO 02</i> . ....  | 56 |
| FIGURA 54 - MÁSCARAS DE <i>FOREGROUND</i> E MÁSCARA DE OBJETOS ABANDONADOS QUE ORIGINARAM O ALARME DA<br>FIGURA 53. ....                       | 57 |
| FIGURA 55 - DETECÇÃO DE OBJETO ABANDONADO EM IMAGENS EM TONS DE CINZENTO - <i>DATASET ABODA VIDEO<br/>05</i> . ....                            | 57 |
| FIGURA 56 - DETECÇÃO DE OBJETO ABANDONADO EM IMAGENS EM TONS DE CINZENTO - <i>DATASET ABODA VIDEO<br/>05</i> . ....                            | 58 |
| FIGURA 57 - EXEMPLO DE IMAGEM PASSADA À <i>FRAMEWORK CAFFE</i> . .....   | 59 |
| FIGURA 58 - PREDIÇÃO DA REDE NEURONAL PARA A FIGURA 57 .....   | 59 |
| FIGURA 59 - EXEMPLO DE IMAGEM PASSADA À <i>FRAMEWORK CAFFE</i> . .....   | 59 |
| FIGURA 60 - PREDIÇÃO DA REDE NEURONAL PARA A FIGURA 59 .....   | 59 |
| FIGURA 61 - BACKGROUND APRENDE INFORMAÇÃO DEMASIADO RÁPIDO - <i>DATASET ABODA VIDEO 03</i> . .....   | 60 |
| FIGURA 62 - EXEMPLO DE <i>FRAME</i> ANTERIOR À FIGURA 61 - <i>DATASET ABODA VIDEO 03</i> . ....  | 61 |
| FIGURA 63 - DETECÇÃO DE MALA ABANDONADA - <i>DATASET ABODA VIDEO 07</i> . ....   | 61 |
| FIGURA 64 - INFORMAÇÃO DE <i>TRACKING</i> ANTES DO ABANDONO DO OBJETO - <i>DATASET ABODA VIDEO 07</i> . ..                                     | 62 |
| FIGURA 65 - INFORMAÇÃO DE <i>TRACKING</i> APÓS O ABANDONO DO OBJETO - <i>DATASET ABODA VIDEO 07</i> . ..                                       | 62 |
| FIGURA 66 - DUPLO ALARME PARA O MESMO OBJETO - <i>DATASET ABODA VIDEO 08</i> . ....  | 63 |
| FIGURA 67 - DETECÇÃO EFICAZ COM IMAGENS RGB - <i>DATASET ABODA VIDEO 07</i> . .....  | 63 |

# Lista de Tabelas

|  |    |
|--|----|
| TABELA 1 - COMPARAÇÃO ENTRE OS DIFERENTES MÉTODOS DE SUBTRAÇÃO DE PLANO DE FUNDO (BGS) COM AS PONTUAÇÕES GERAIS E POR CATEGORIA PARA O DATASET CDNET 2012 [37]. (ADAPTADA DE [36])... 32 | 32 |
| TABELA 2 - COMPARAÇÃO ENTRE OS DIFERENTES MÉTODOS DE SUBTRAÇÃO DE PLANO DE FUNDO (BGS) COM AS PONTUAÇÕES GERAIS E POR CATEGORIA PARA O DATASET CDNET 2014 [38]. (ADAPTADA DE [36])... 33 | 33 |
| TABELA 3 - RESULTADOS PARA OS DATASETS I-LIDS [23] E ABODA [49]. .....   | 64 |



## Acrónimos e Estrangeirismos

Na escrita deste documento foram empregues alguns acrónimos e estrangeirismos. São termos com uma grande utilização nesta área de investigação e a sua tradução para Português não era possível de forma prática e compacta.

| <b>Termo</b>         | <b>Designação</b>  |
|----------------------|--|
| <b>Background</b>    | Plano de fundo do vídeo, contém as zonas estáticas.  |
| <b>Benckmark</b>     | Resultado da execução de um programa de forma a avaliar o desempenho.  |
| <b>BGS</b>           | Background Subtraction - Subtração do plano de fundo.  |
| <b>Blob</b>          | Mancha numa imagem binária que representa um objeto de interesse.  |
| <b>Dataset</b>       | Conjunto de imagens ou vídeos.   |
| <b>Deep learning</b> | Subárea da aprendizagem automática habitualmente utilizada no reconhecimento da fala, texto e imagem.                |
| <b>Feedback</b>      | Retorno de informação, avaliação do desempenho.  |
| <b>Foreground</b>    | Primeiro plano do vídeo, contém os objetos em movimento.   |
| <b>Frame</b>         | Uma imagem do vídeo, a um determinado momento.   |
| <b>Frame rate</b>    | Frequência a que são geradas as imagens numa câmara ou vídeo.  |
| <b>Framework</b>     | Conjunto de diferentes funcionalidades utilizadas para resolver um problema num domínio específico.                  |
| <b>Ground truth</b>  | Conjunto de dados corretamente anotados que permitem avaliar a precisão da classificação de um determinado algoritmo |
| <b>Inpainting</b>    | Processo que permite reconstruir partes perdidas ou deterioradas em imagens ou vídeos.                               |
| <b>Pixel</b>         | Menor elemento ao qual é possível atribuir uma cor (representa um ponto da imagem).                                  |

|                  |   |
|------------------|---|
| <b>Ranking</b>   | Classificação ordinal de um conjunto de técnicas ou itens.    |
| <b>Sample</b>    | Uma amostra com um determinado valor.                         |
| <b>Threshold</b> | Valor limiar utilizado num determinado algoritmo.             |
| <b>Timestamp</b> | Marca temporal que denota a hora em que ocorreu um evento.    |
| <b>Tracking</b>  | Seguimento do objeto. Trajetória do objeto ao longo do tempo. |

# 1. Introdução

Neste capítulo introdutório apresentamos as motivações que deram origem a este trabalho de dissertação, fazemos a descrição do problema e apresentamos os objetivos que se pretendem alcançar. No final é descrita a estrutura deste documento.

## 1.1. Motivação

Nos últimos anos assistimos a um aumento significativo da criminalidade e de ações terroristas orquestradas em espaços públicos (eventos desportivos, estações de comboio ou metro, e aeroportos). A constante melhoria dos sistemas de segurança e de videovigilância inteligente tornou-se assim um requisito necessário e implícito no nosso modelo de sociedade atual.

São áreas de investigação nas quais se aplica bastante esforço e interesse, e onde as técnicas de visão computacional prestam um contributo essencial.

## 1.2. Descrição do problema e objetivos

Num sistema de videovigilância tradicional um humano é responsável por prestar atenção às imagens de várias câmeras de vigilância, tentando dessa forma detetar eventos que coloquem em causa a segurança do local ou das pessoas.

Como se pretende que a quantidade de locais a observar seja cada vez maior, seria altamente dispendioso aumentar na mesma proporção o número de intervenientes humanos. Além disso, após algumas horas a efetuar esta tarefa rotineira, a atenção e concentração de um observador humano diminui consideravelmente.

Os sistemas de videovigilância inteligentes são assim uma importante ferramenta, pois permitem a análise eficaz e em tempo real de uma elevada quantidade de vídeos, e disparam alertas que podem desencadear uma posterior ação dos responsáveis pela segurança.

O objetivo desta dissertação consiste em identificar as técnicas utilizadas para a detecção automática de objetos abandonados. Iremos escolher o método mais robusto, proceder à sua implementação e apresentar algumas melhorias e novas funcionalidades.

### 1.3. Estrutura do documento

Este documento está dividido em 6 capítulos:

1. **Introdução** - O primeiro capítulo apresenta de forma sucinta o projeto. Contém as motivações e uma breve descrição dos objetivos a alcançar.
2. **Estado-da-arte** - É apresentado um estudo das tecnologias disponíveis e avanços científicos relacionados com o tema em questão.
3. **Análise do Método de Referência** - Com base na análise do Estado-da-arte um dos métodos é selecionado e analisado ao pormenor.
4. **Método Proposto** - Neste capítulo descrevemos a nossa abordagem ao problema, que visa minimizar as limitações identificadas no método de referência.
5. **Resultados e Discussão** - São apresentados os resultados alcançados pelo nosso método.
6. **Conclusões e Trabalho Futuro** - No capítulo final são apresentadas as conclusões do projeto, e identificado trabalho futuro ou melhorias que podem ser implementadas.

## 2. Estado-da-Arte

A videovigilância inteligente preocupa-se em obter de forma automática a descrição do que está a acontecer numa área monitorizada, e de seguida tomar as medidas adequadas com base nessa interpretação.

As premissas e requisitos de um sistema de videovigilância inteligente podem variar dependendo do cenário específico que se encontra sob vigilância (aeroporto, parque de estacionamento, centro da cidade, estrada, sala de aulas, etc) e o que é monitorizado (a cena inteira ou apenas objetos em movimento).

Os principais módulos deste tipo de sistemas envolvem a deteção e seguimento de objetos, a classificação dos objetos e a compreensão da sua atividade.

No caso concreto da deteção automática de objetos abandonados, os métodos que iremos de seguida analisar focam-se em uma ou várias destas categorias de análise de imagem:

- 1) avaliação ao nível do pixel (à base da segmentação: objetos em movimento ou plano de fundo (*background*)).
- 2) avaliação estática do objeto (analisar com recurso à forma do objeto: tipo, tamanho, etc).
- 3) avaliação dinâmica do objeto (avaliar características ao longo do tempo de vida em cena do objeto: velocidade, trajetória).
- 4) avaliação do comportamento (deteção de eventos: uma pessoa juntou-se a um grupo de várias pessoas, etc.).

Em [1] é proposto um sistema para deteção automática de objetos abandonados ou roubados, de forma a auxiliar um operador humano em ambientes vigiados. Para o módulo de *tracking* são utilizadas as características de cor e posição dos *blobs*, enquanto que para o módulo de classificação que permite distinguir entre um objeto ou uma pessoa são utilizados os Momentos Invariantes de Hu (invariantes em relação à escala, rotação e translação). É feita uma associação entre os objetos e as pessoas (proprietário ou ladrão). Um alarme de objeto abandonado acontece quando uma pessoa larga um objeto na cena e a posição de *tracking* do objeto permanece estática por um determinado período de tempo. (Figura 1).



Figura 1- Alarme de objeto abandonado e o seu proprietário assinalado. (Adaptada de [1]).

O método [2] pretende detetar objetos abandonados em condições do mundo real, diferenciando pessoas paradas de objetos abandonados, e é robusto contra oclusões temporárias de objetos potencialmente abandonados. Para classificar os objetos como pessoas em movimento, pessoas paradas ou objetos abandonados utiliza a informação dos *blobs* fornecidos pela máscara de *foreground* e analisa o seu comportamento. Os quatro tipos de comportamentos considerados para a classificação são: a criação do *blob*, a divisão de um *blob* em múltiplos *blobs*, a união de vários *blobs* em um único *blob*, e a velocidade do centróide do *blob*.

A abordagem proposta em [3] faz a relação entre um objeto hipoteticamente abandonado e a pessoa que o abandonou, utilizando a transformada de *Hough* para detetar a parte superior do corpo e a informação da cor dessa zona para a reconhecer ao longo da cena (Figura 2).



Figura 2 - Detecção de bagagem estática; O seguimento do proprietário é iniciado; Quando o proprietário sai de cena, o alarme é disparado. (Adaptada de [3]).

O trabalho [4] apresenta dois métodos com plano de fundo adaptativo para deteção de objetos abandonados ou removidos, ambos com base no Modelo de Mistura de Gaussianas (GMM) [5] (Figura 3).

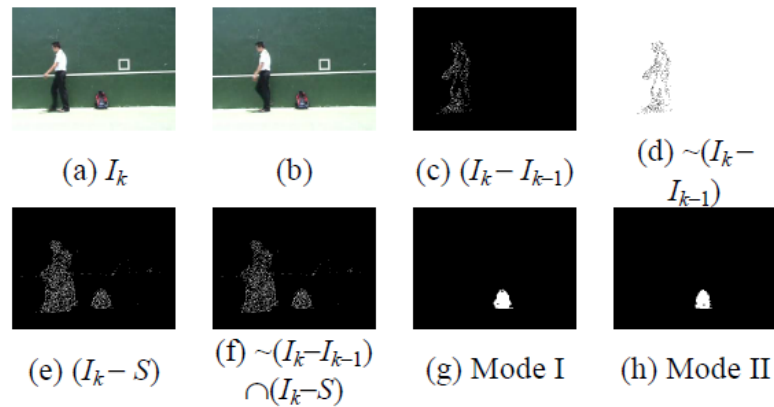


Figura 3 - Em (g) e (h) são visíveis as máscaras de objetos abandonados geradas pelos dois métodos propostos. (Adaptada de [4]).

O cenário concreto dos aeroportos é abordado no método [6], que procura distinguir os objetos abandonados classificando-os como mala, *trolley*, pessoa ou grupo de pessoas, através da extração de algumas características simples (linhas, círculos e cantos) da imagem de cada objeto de interesse (Figura 4) que são depois usadas num classificador Bayesiano.

| Images Features | Uncluttered images | Cluttered images |
|-----------------|--------------------|------------------|
| Lines           |                    |                  |
| Circles         |                    |                  |
| Corners         |                    |                  |

Figura 4 - Exemplo das características extraídas dos objetos. (Adaptada de [6]).

O método [7] tenta fazer a detecção dos objetos abandonados ou removidos recorrendo a uma análise comparativa da riqueza das cores entre as zonas estáticas da imagem com as mesmas zonas no plano de fundo de referência (*background*) (Figura 5).





|                       | $I$   | $N_{CR}$ | $B^1$  | $N_{CR}$ |
|-----------------------|---|----------|--|----------|
| $t=2825$<br>Abandoned |  | 17       |  | 13       |
| $t=5089$<br>Removed   |  | 12       |  | 19       |

Figura 5 - A riqueza das cores ( $N_{CR}$ ) na imagem atual ( $I$ ) e no *background* ( $B^1$ ). Quando acontece um evento de abandono a riqueza das cores em  $I$  é maior do que em  $B^1$ . Pelo contrário, quando um objeto é removido  $B^1$  tem cores mais ricas do que  $I$ . (Adaptada de [7]).

Na técnica proposta em [8] dois modelos de *background* são mantidos a uma diferente velocidade de atualização (*frame rate*).

A Figura 6 (a) mostra o *background* atualizado a cada 200 *frames*, (b) o *background* atualizado a cada 2000 *frames*, e (c) a imagem atual.

As imagens que contêm a máscara de primeiro plano (*foreground*) para cada *background* são mostradas em (d) e (e) respetivamente. (f) contém a máscara com pixels candidatos a regiões estáticas, que são aqueles que permanecem nos *foregrounds* acima de um determinado *threshold* de  $T$  *frames*.

A distância entre os histogramas de cor das regiões de interesse em cada um desses *backgrounds* de referência com a imagem atual permite avaliar se o objeto foi abandonado ou removido.

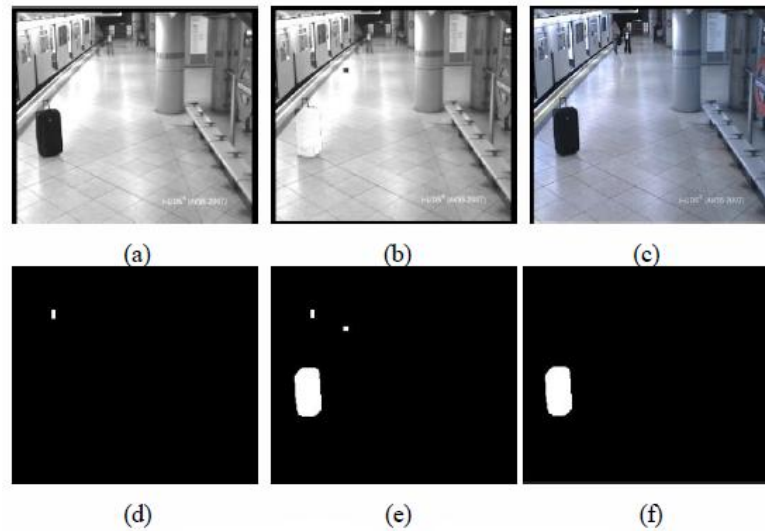


Figura 6 - O resultado da detecção de um objeto estático. (Adaptada de [8]).

O sistema apresentado em [9] utiliza um modelo de mistura de Gaussianas [5] para efetuar a subtração do plano de fundo. É também aplicada a remoção de sombras. Através da análise das arestas verifica se um objeto foi ou não abandonado (Figura 7).

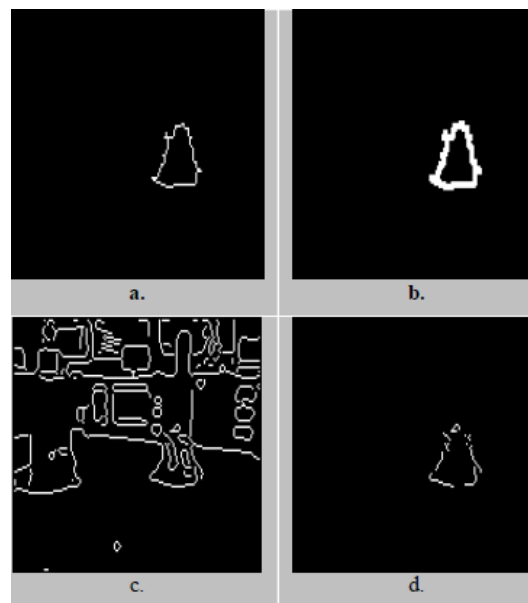


Figura 7 - Distinção entre objeto abandonado e removido. (a) arestas do *blob*. (b) arestas do *blob* dilatadas. (c) arestas detetadas na imagem. (d) contém o resultado da operação de AND binário entre (b) e (c). (Adaptada de [9]).

Em [10] é feita a avaliação das variações das arestas (pequenos movimentos) de objetos que permanecem estáticos durante um período de tempo estendido. Na (Figura 8) podemos observar que uma parte do corpo de uma pessoa sentada no banco permaneceu estática mais tempo do que o valor de *threshold* definido e foi assim assinalada como potencial candidato a objeto abandonado (*pixels* azuis). Contudo como a pessoa produz alguns movimentos com a parte superior do corpo, a pontuação de objeto abandonado decresce gradualmente na parte superior do *blob* azul. Pelo contrário, a mala que é assinalada a vermelho forma contornos mais estáveis ao longo do tempo e produz uma pontuação de objeto abandonado superior.



Figura 8 - Distinção entre pessoas estáticas e objetos abandonados. (Adaptada de [10]).

Os autores do método [11] propõem a utilização de dois modelos de *background* a diferentes velocidades de atualização para a deteção de objetos abandonados ser mais robusta e é também feita a identificação do proprietário do objeto abandonado através da análise da divisão de *blobs* (Figura 9).

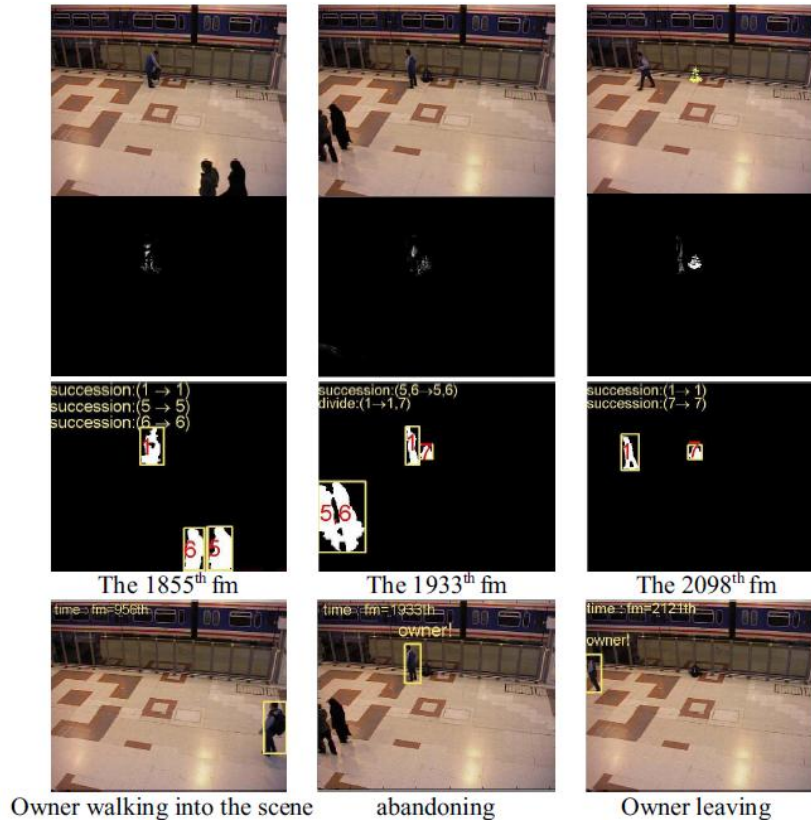


Figura 9 - Exemplo da deteção de objeto abandonado e deteção do proprietário. (Adaptada de [11]).

Em [12] é utilizado um modelo de mistura de Gaussianas (GMM) ao qual é incorporada uma componente de remoção de sombras. A classificação dos objetos abandonados é feita com recurso a um campo aleatório de Markov.

No método [13] dois modelos de misturas de Gaussianas (GMM [5], um de longa duração e outro de curta duração) são construídos no espaço de cor RGB para obter duas máscaras do primeiro plano. O impacto de alterações da iluminação é reduzido ao utilizar o método RRF (*Radial Reach Filter*). Para classificar os objetos estáticos como bagagem abandonada é utilizado o rácio altura/largura do objeto e um classificador SVM baseado no HOG (*Histogram of Oriented Gradient*) [14].

A metodologia apresentada em [15] estuda a relação entre os diversos objetos para tentar identificar de uma forma semântica várias atividades suspeitas em espaços públicos. As características que extrai de cada um dos objetos são a sua

posição, velocidade e direção. É analisada a distância, o alinhamento e a diferença de velocidade entre os vários objetos em cena. Na máscara do primeiro plano também é analisada a união ou separação de *blobs* (Figura 10).

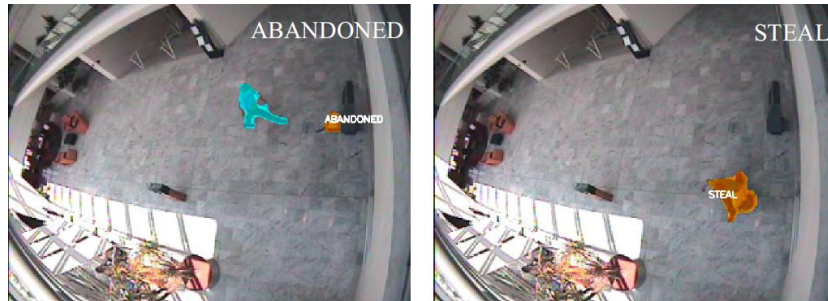


Figura 10 - Exemplo da análise da união ou separação de *blobs* para tentar inferir se o objeto foi abandonado ou roubado. (Adaptada de [15]).

O trabalho [16] apresenta uma abordagem que gere diferentes categorias de alarme em função do *ranking* interno obtido pelo alerta, com vista a reduzir o número de falsos-positivos (Figura 11). Os três principais atributos visuais analisados para cada objeto são o seu nível estático, a diferença ao nível de aparência em relação ao *background*, e se foi ou não abandonado (isto é, se o *blob* surgiu da divisão de um outro *blob* em movimento). O *ranking* gerado através destes atributos permite descartar alarmes criados por artefactos de luz e colocar num nível intermédio as pessoas parcialmente imóveis por um longo período de tempo.

O método [17] associa cada objeto abandonado ao seu proprietário. Parte da premissa que todos os objetos são movidos pelo seu proprietário, e que todos os objetos que se movem em cena são humanos. O algoritmo de subtração do plano de fundo é aplicado a pequenos blocos da imagem em vez de ser *pixel a pixel*, utilizando para isso algumas características *haar-cascade*.

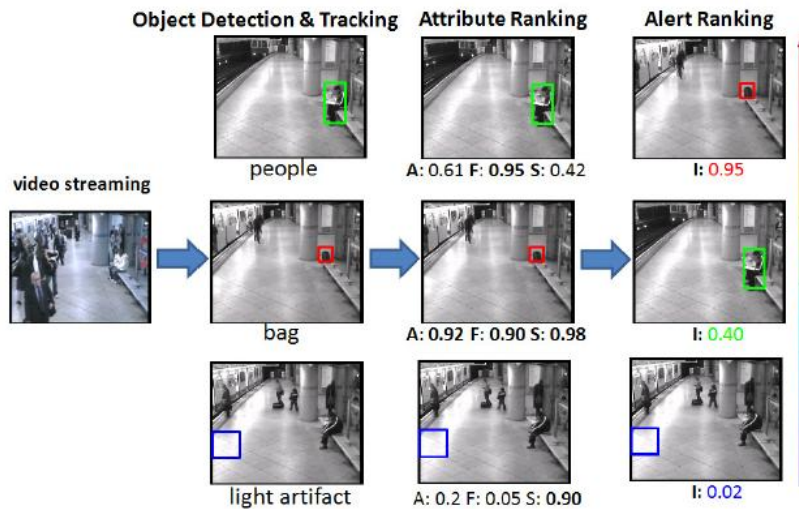


Figura 11 - Gráfico do fluxo geral do sistema. Um objeto abandonado tende a ter pontuações elevadas do seu nível estático (S), de distinção com o *background* (F), e de abandono (A). Estas pontuações são depois passadas a um classificador que prioriza os alertas pela sua importância (I). (Adaptada de [16]).

O trabalho [18] propõe a deteção de objetos estáticos através da análise de máscaras do primeiro plano. Este processo é tipicamente articulado em duas etapas: 1) a deteção de objetos não pertencentes ao plano de fundo (máscara de primeiro plano); 2) a deteção de objetos estáticos na máscara de primeiro plano. Os principais desafios são as mudanças bruscas da iluminação, sombras, e em situações complexas do mundo real em que um objeto pode estar parado e retomar o movimento ou então sofrer de oclusão.

Os autores de [19] estudam o processamento e deteção em processadores com recursos limitados (ARM e Atom). Apresentam duas abordagens para a deteção de objetos estáticos. A primeira, passa pela deteção de componentes ligados na máscara resultante da diferença entre um modelo de *background* em que a taxa de atualização é mais rápida do que a do modelo de *background* de referência. Os objetos resultantes nesta máscara são assinalados como estáticos.

A segunda abordagem cria uma fila de tamanho fixo com imagens consecutivas do primeiro plano (*foreground*). Quando a fila está totalmente ocupada a imagem mais antiga é descartada e todas as outras são movidas em uma posição.

Um AND binário de todas estas imagens mostra-nos a área que é comum a todas, que consiste na prática a todos os objetos estáticos das imagens de *foreground* presentes na fila.

O tamanho da fila define quantas imagens estamos a analisar, e com este método podemos procurar por objetos estáticos nas últimas  $N$  *frames*. Sabendo a *frame rate* (fps) alcançada pela aquisição de imagem e restante processamento do método, podemos converter este número em tempo, definindo assim o período pelo qual pretendemos analisar o nível estático de um objeto para o classificar como abandonado.

Como podemos observar na Figura 12, o método [20] propõe a criação de um modelo de *background* de referência, e uma série de outros  $M$  modelos de *background* referentes às *frames* consecutivas  $K$  de um determinado período. Os autores sugerem que o valor escolhido para  $M$  seja uma terça parte da velocidade de aquisição de vídeo. A subtração de todos estes modelos pelo modelo de *background* de referência resulta nas regiões onde será feita a análise dos objetos estáticos.

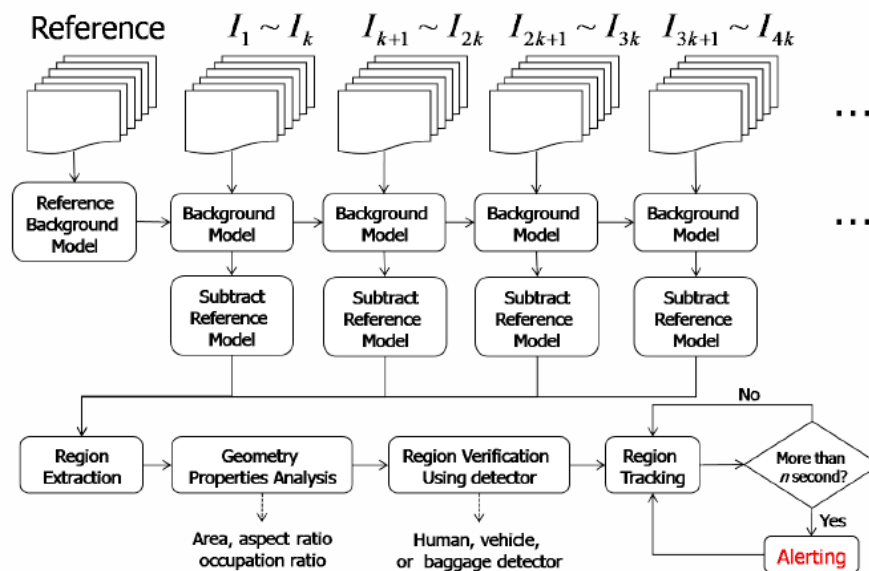


Figura 12 - Diagrama do método proposto em [20]. (Adaptada de [20]).

O trabalho [21] sugere bons resultados na detecção de objetos abandonados utilizando o método de *tracking* de múltiplos objetos Recursive-RANSAC (Figura 13).



Figura 13 - Detecção de objetos estáticos utilizando R-RANSAC. (Adaptada de [21]).

Em [22] é apresentada uma *framework* para detetar de forma robusta e eficaz objetos abandonados e removidos da cena, baseada na subtração do plano de fundo (BGS - *background subtraction*) e na análise do primeiro plano (*foreground*) complementados pelo *tracking* para reduzir os falsos positivos.

Neste sistema o background é modelado com recurso a três misturas Gaussianas (GMM [5]). De forma a lidar com situações complexas, diversas melhorias foram implementadas para remoção de sombras, adaptação a variações repentinas de luminosidade, redução de fragmentos, e para estabilizar a taxa de atualização da aquisição de vídeo com *frame rate* variável.

Os mesmos modelos de misturas gaussianas utilizados para o BGS são usados para detetar regiões de primeiro plano estáticas sem qualquer custo computacional adicional.

Além disso, os tipos de regiões estáticas (abandonado ou removido) são determinados usando um método que explora a informação de contexto das máscaras de primeiro plano, o que supera significativamente as abordagens anteriores baseadas em técnicas de detecção de arestas. Com base no tipo de

região estática e em parâmetros definidos pelo utilizador, é proposto um método de correspondência para detetar objetos removidos ou abandonados.

Foi também integrado um método de deteção de pessoas para distinguir objetos estáticos de pessoas que permanecem imóveis num determinado local.

A eficácia do método proposto foi testada nas soluções de videovigilância inteligente da IBM em aplicações de segurança pública em grandes cidades e avaliadas em diversas bases de dados de imagens públicas, como a "*Image library for intelligent detection systems*" (i-LIDS) [23], e a "*IEEE Performance Evaluation of Tracking and Surveillance Workshop 2006*" (PETS 2006) [24].

Os autores defendem que o método é eficiente para ser executado em tempo real, sendo robusto a mudanças rápidas de iluminação e a oclusões em ambientes complexos.

Na nossa pesquisa este artigo além de ser o que possui um maior número de citações (segundo o *IEEE Xplore Digital Library* em 57 publicações e 2 patentes), é aquele que especifica mais detalhadamente os diferentes módulos de uma *framework* completa para deteção de objetos abandonados. Decidimos assim utilizá-lo como ponto de partida para o nosso trabalho.

## 3. Análise do Método de Referência

Neste capítulo é apresentada a metodologia descrita no artigo de referência “*Robust Detection of Abandoned and Removed Objects in Complex Surveillance Videos*” [22], que servirá de base inicial para este trabalho.

### 3.1. Método de referência

Os autores do artigo de referência [22] propõem uma nova solução para detetar objetos abandonados e removidos, que surge para melhorar uma versão anterior que pode ser encontrada em [25]. Comparado com o trabalho anterior, existem algumas melhorias entre as quais se destacam:

- o método atual fornece resultados mais precisos para classificar regiões estáticas como objetos abandonados ou objetos removidos;
- o trabalho anterior não era capaz de distinguir humanos numa posição estática de outro tipo de objetos. Neste é integrado um módulo de deteção de humanos a curtas, médias e longas distâncias da câmara;
- o trabalho anterior não utilizava nenhuma informação de seguimento do objeto (*tracking*), neste são utilizadas as trajetórias que os objetos percorrem como informação complementar.

A Figura 14 mostra o diagrama do sistema, que inclui quatro componentes principais:

- (a) deteção do plano de fundo (BGS) e das regiões estáticas;
- (b) deteção do tipo de objeto (abandonado ou removido);
- (c) alerta de deteção de objetos abandonados/removidos;
- (d) integração da deteção de humanos e da informação de *tracking* para a deteção de objetos abandonados e removidos.

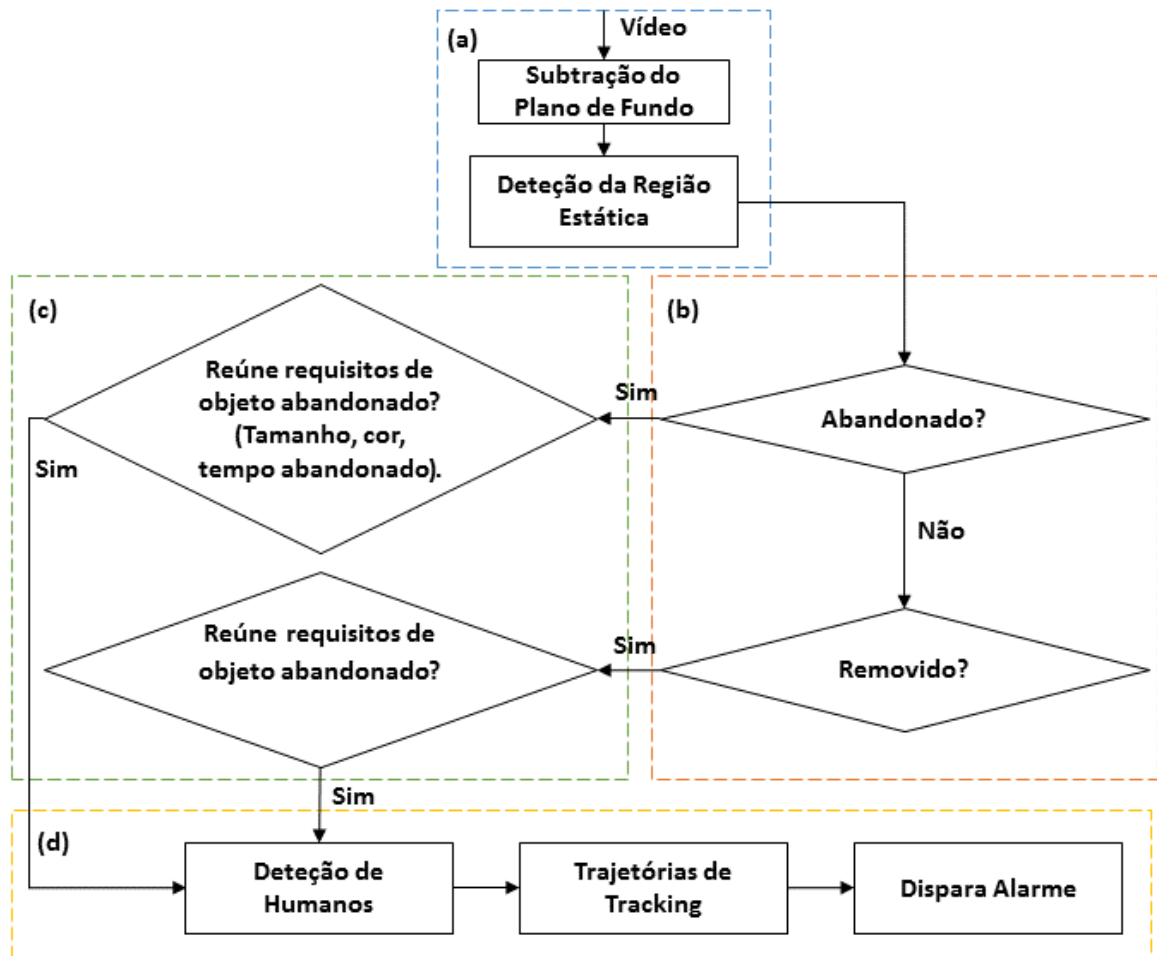


Figura 14 - *Framework* para detecção de objetos abandonados. (Adaptada de [22]).

No geral o trabalho apresentado em [22] oferece as seguintes contribuições para a detecção de objetos abandonados ou removidos em cenários complexos de videovigilância:

- É aplicada uma mistura de Gaussianas para classificar partes do primeiro plano (*foreground*) como objetos em movimento, objetos abandonados, ou objetos removidos (fantasmas), enquanto se faz a atualização do plano de fundo.

- Diferentes limiares (*thresholds*) são utilizados para obter a máscara de primeiro plano e a máscara de regiões estáticas.

- Para a máscara de primeiro plano, a informação de intensidade e de textura são integradas para remover sombras e fazer o algoritmo funcionar em variações rápidas de iluminação.

- Para a máscara de regiões estáticas, foi desenvolvido um método de segmentação para detetar o tipo de região estática (abandonado ou removido), que melhora significativamente as técnicas anteriores baseadas na detecção de arestas.

- Um algoritmo de correspondência é utilizado para detetar se o objeto está abandonado ou removido há tempo suficiente, ainda que sofra de oclusão parcial. Aqui, oclusão parcial significa que apenas parte do objeto está a ser tapado da perspetiva da câmara, ou uma oclusão total que ocorre durante um curto período de tempo (menor que aquele definido pelo utilizador para classificar um objeto como abandonado ou removido).

- Um histórico dos objetos que pertencem ao plano de fundo é mantido para fazer o algoritmo de correspondência robusto a variações de luz.

- A deteção de humanos a diferentes distâncias da câmara (curta, média e longa) é integrada na *framework* para distinguir as pessoas paradas de objetos abandonados.

### 3.2. Subtração do plano de fundo

Stauffer e Grimson [5] introduziram uma mistura de  $K$  Gaussianas (sendo  $K$  um número de 3 a 5) para construir o modelo de plano de fundo adaptativo e detetar os objetos em movimento. Para um *pixel*  $X$  no tempo  $t$ , a probabilidade do pixel pode ser escrita da seguinte forma [5]:

$$P(X_t) = \sum_{i=1}^K \omega_{i,t} * \eta (X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (1)$$

onde  $K$  é o número de misturas Gaussianas;

$\omega_{i,t}$  é a estimativa do peso de atualização da distribuição Gaussiana  $i$ ;

$\mu_{i,t}$  é a média da distribuição Gaussiana  $i$ ;

$\Sigma_{i,t}$  é a matriz de covariância da distribuição Gaussiana  $i$ ;

$\eta$  é a função densidade de probabilidade Gaussiana.

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(X_t - \mu)^T \Sigma^{-1} (X_t - \mu)} \quad (2)$$

em que  $\eta$  é a dimensão de intensidade do pixel  $X$  e

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t-1} + \alpha(M_{k,t}) \quad (3)$$

onde  $\alpha$  é a taxa de aprendizagem e  $M_{k,t}$  é 1 para o modelo que corresponde e 0 para os restantes modelos. Assumindo que as componentes vermelho, verde e azul (RGB) de cada *pixel* são independentes e têm as mesmas variâncias  $\sum k, t = \sigma_k^2 I$ . Após as Gaussianas serem ordenadas pelo valor de  $\omega/\sigma$ , as primeiras  $B$  distribuições são escolhidas como modelo do plano de fundo, onde

$$B = \arg \min_b (\sum_{k=1}^b \omega_k > T) \quad (4)$$

e  $T$  é a porção mínima do plano de fundo. Os parâmetros  $\mu$  e  $\sigma$  para distribuições sem correspondência permanecem os mesmos.

Os parâmetros da distribuição que correspondem a uma nova observação são atualizados da seguinte forma:

$$\begin{aligned} \mu_t &= (1 - \rho)\mu_{t-1} + \rho X_t \\ \sigma_t^2 &= (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T (X_t - \mu_t) \\ \rho &= \alpha\eta(X_t | \mu_k, \sigma_k). \end{aligned} \quad (5)$$

Na implementação [5] dois parâmetros importantes  $\alpha$  e  $T$  precisam de ser definidos. Em [22] os autores utilizam  $K = 3$  (três Gaussianas),  $\alpha = 0.005$ , e  $T = 0.4$ , e o método de subtração do plano de fundo foi implementado quer para imagens em tons de cinza como em RGB.

O método de mistura de Gaussianas é robusto a leves mudanças na iluminação, movimentos periódicos de objetos do background, objetos com movimento lento, mudanças de cenário de longo prazo, e ruídos da câmara.

No entanto ele não se consegue adaptar a mudanças rápidas de iluminação e não lida bem com as sombras. Uma série de técnicas foram desenvolvidas para

melhorar o desempenho do método de mistura de Gaussianas: [26], [27], [28] e [29].

De forma a tornar este método capaz de suportar mudanças rápidas de iluminação, é integrada a informação de textura na máscara do *foreground* para remover áreas com falsos-positivos utilizando as características do gradiente, uma vez que a textura nas áreas com falsos-positivos causados por alterações da iluminação devem ser similares à textura do plano de fundo (*background*), e o valor do gradiente é menos sensível a variações da iluminação e capaz de fornecer um valor preciso da medida da diferença da textura local. Para remover da máscara de *foreground* falsos-positivos causados por sombras, é feita uma correlação entre as intensidades de cada *pixel* do *foreground* na *frame* atual com os da imagem de *background*.

### 3.3. Detecção de objetos estáticos

Por objetos estáticos entendemos aqueles que permanecem imóveis na mesma posição durante um período de tempo relativamente longo.

O plano de fundo é modelado usando três misturas Gaussianas (semelhante a Tian et al. [34]), e as regiões estáticas são detetadas sem qualquer custo computacional extra. Geralmente a primeira distribuição Gaussiana mostra os *pixels* persistentes e representa a imagem do plano de fundo. As variações repetitivas e regiões relativamente estáticas são atualizadas na segunda Gaussiana. A terceira Gaussiana representa os *pixels* com rápidas mudanças. A segunda mistura Gaussiana do modelo de plano de fundo é utilizada para detetar se um *pixel* do *foreground* pertence à região estática da imagem:

$$pixel \in \text{região estática}, \text{ se } \omega_{B+1} > T. \quad (6)$$

Se o peso da segunda Gaussiana para um *pixel* é maior que um determinado limiar, esse *pixel* pertence à região estática. O processo de componentes ligados é efetuado tanto na máscara de *foreground* como na máscara com as regiões estáticas.

Para a implementação do sistema é necessário resolver os seguintes problemas:

- 1) Quando é que devemos transferir pixels da região estática para o modelo do plano de fundo (primeira distribuição Gaussiana)?
- 2) Como ajustar a taxa de atualização do modelo para vídeos com *frame rates* variáveis?
- 3) Como reduzir a fragmentação das regiões estáticas?

**1) Tratamento da região estática:** fragmentos no primeiro plano são bastante habituais em muitas técnicas de subtração do plano de fundo (BGS). No método de misturas de Gaussianas as diferentes partes de uma região estática são frequentemente atualizadas para o modelo do *background* a diferentes velocidades com base na similaridade dos valores dos *pixels* entre a região estática e o modelo do plano de fundo. Ao transferir a região estática para o modelo do plano de fundo quando a região estática é maior (isto é, antes de começar a encolher), podemos evitar a fragmentação do primeiro plano (*foreground*). Para transferir a região estática para o modelo do plano de fundo, redefinimos o peso da região estática com o peso máximo, que foi definido no programa. A média e a variância da segunda distribuição Gaussiana é trocada pela primeira distribuição para cada pixel na máscara de região estática.

**2) Atualização do modelo de BGS a uma taxa fixa para vídeos com diferente frame rate:** A maioria dos métodos de subtração de plano de fundo adaptativos, atualizam os modelos do plano de fundo com base nas imagens de entrada e um parâmetro de frequência pré-definido. Neste caso, os modelos do plano de fundo são atualizados a diferentes velocidades em vídeos com *frame rate* variável ainda que o parâmetro da taxa de atualização seja o mesmo. Em sistemas profissionais de videovigilância, que usam como entrada vídeos capturados em tempo real, a *frame rate* varia drasticamente mesmo para a mesma câmera devido às várias aplicações em execução na máquina e a complexidade do cenário. Para detetar objetos abandonados e removidos no método de mistura de Gaussianas, o tempo definido para classificar um objeto como abandonado/removido está diretamente relacionado com a taxa de

atualização do modelo. Para garantir a estabilidade entre o tempo que o objeto é abandonado ou removido até que o sistema deteta a região estática, atualizamos o modelo de plano de fundo baseado no *timestamp* em vez do número incremental da *frame*.

- 3) **Definir dois *thresholds* para a máscara de primeiro plano e a máscara da região estática:** De modo a evitar fragmentos na região estática, usamos dois diferentes limiares de peso para a máscara de primeiro plano e a da região estática. No método de subtração de plano de fundo com mistura de Gaussianas, as diferentes partes de uma região estática são frequentemente atualizadas para o modelo do plano de fundo a diferentes velocidades com base na similaridade dos valores dos *pixels* entre a região estática e o modelo do plano de fundo. Alguns *pixels* na região estática são frequentemente atualizados para o modelo de *background* antes da região estática estar completamente tratada. Usamos um valor de *threshold* inferior para a máscara da região estática e um maior para a máscara de primeiro plano. Esta abordagem foi também explorada por Boulton et al. [30] para a modelação do plano de fundo. Mais recentemente, Zhang et al. usou a mesma ideia numa ferramenta mais genérica, argumentando que “dois *thresholds* são melhores do que apenas um” [31] para aplicações de visão por computador.

### 3.4. Deteção de objetos abandonados e removidos

Após as regiões estáticas serem detetadas e tratadas (isto é, transferidas para o modelo do plano de fundo), é necessário classificar se esse evento se deve ao facto do objeto ter sido abandonado ou removido.

Poucos métodos têm sido propostos na literatura para classificar regiões estáticas como objetos abandonados ou removidos. Existem técnicas que se baseiam na análise da intensidade das arestas ao longo da região estática na imagem do plano de fundo e a imagem atual [32], [33]. A intuição é que, em muitos casos, cobrindo o plano de fundo com um objeto iremos introduzir mais arestas na imagem devido aos limites do objeto. Baseado neste pressuposto, a região de primeiro plano pode

ser classificada como objeto abandonado se a imagem de fundo contém menos arestas que a imagem atual (ao longo do *blob* estático do *foreground*), e inversamente para os objetos removidos.

Apesar destes métodos funcionarem bem para cenários com um plano de fundo simples, não são adequados para ambientes complexos que envolvem multidões, trânsito e oclusões.

Os autores de [22] descrevem as duas limitações principais que surgem nestas condições:

- 1) O pressuposto de energia das arestas é claramente violado quando o plano de fundo contém muitas e intensas arestas.
- 2) Nos cenários em que o objeto sofre constantemente oclusão, é possível que apenas uma parte do objeto seja tratado. Neste caso, a região estática não irá conter as arestas que estão a sofrer oclusão, potencialmente apresentando arestas menos intensas.

O aspeto chave do método [22] para resolver estes problemas é explorar os arredores do objeto estático para classificá-lo como abandonado ou removido. Segundo os autores, a informação circundante tem características ricas para inferir o que está no interior do objeto, uma vez que tem sido demonstrado pelos bons resultados obtidos por técnicas de *inpainting* [34].

As técnicas de *inpainting* podem ser utilizadas para “preencher” os *blobs* da máscara de primeiro plano, de forma a que a imagem resultante possa ser comparada com a imagem de *background* para determinar se o objeto foi abandonado ou removido. Contudo esta operação é computacionalmente pesada e pode falhar em regiões grandes com texturas complexas.

Em vez de iniciar das arestas até ao interior do *blob* como nas técnicas de *inpainting*, a estratégia dos autores passa pelo oposto. Começam nos limites do *blob* que representa o objeto estático e utilizam um processo de segmentação que cresce para o exterior do *blob*, de forma a verificar como a região estática se encaixa na sua vizinhança. Esta segmentação é feita com recurso a um descritor de similaridade local [35].

A Figura 15 ilustra a ideia básica da técnica.

Quando um objeto é abandonado, o primeiro passo é aplicar uma erosão ao *blob* que representa a região estática do primeiro plano, de forma a garantir que os seus limites caem completamente dentro do objeto. Os limites da região que sofreu erosão são mostrados na linha a tracejado da Figura 15 (a). De seguida, esses pontos da fronteira são utilizados como sementes no processo de segmentação nas máscaras de arestas da imagem, as setas da figura indicam o sentido em que a região cresce.

O resultado desta segmentação é mostrado na Figura 15 (b). O crescimento da região irá parar na fronteira do objeto.

O mesmo processo de segmentação é então aplicado à imagem de *background*, como se mostra na Figura 15 (c). Neste caso, podemos ver que a região segmentada na Figura 15 (d) é muito maior.

O objeto é classificado como abandonado ou removido com base na comparação do tamanho destas duas regiões segmentadas. Se a segmentação do plano de fundo for maior do que a segmentação na *frame* atual o objeto é classificado como abandonado. Se acontecer o contrário, o objeto é classificado como removido.

No caso de as regiões segmentadas apresentarem tamanhos semelhantes não é disparado nenhum alarme, visto que pode ser uma situação em que ocorreram mudanças na iluminação.

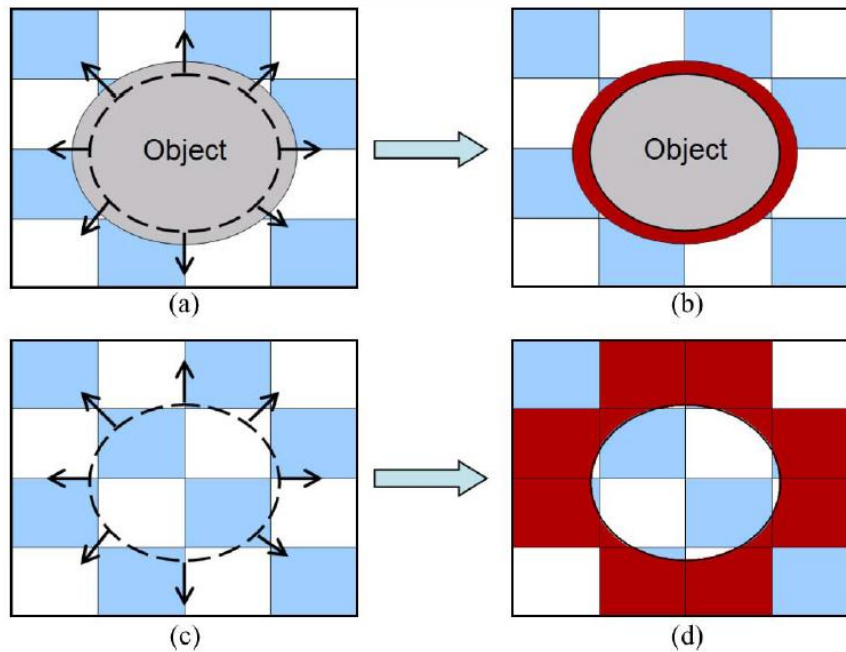


Figura 15 - Método de *region growing* para detecção de objeto abandonado. (Adaptada de [22]).

A Figura 16 mostra um exemplo onde um objeto é abandonado, em que as 4 imagens da linha superior se referem às operações na imagem de primeiro plano (*foreground*) e as 4 imagens da linha inferior às do plano de fundo (*background*).

A Figura 16 (e) e (f) mostra a máscara de detecção estática após sofrer erosão ser sobreposta na *frame* atual e na do plano de fundo, respetivamente. Finalmente, a Figura 16 (g) e (h) mostram as áreas segmentadas após o processo de crescimento da região.

Claramente, a região segmentada no plano de fundo é maior do que a região segmentada na *frame* atual. Como resultado, o objeto é classificado como abandonado.

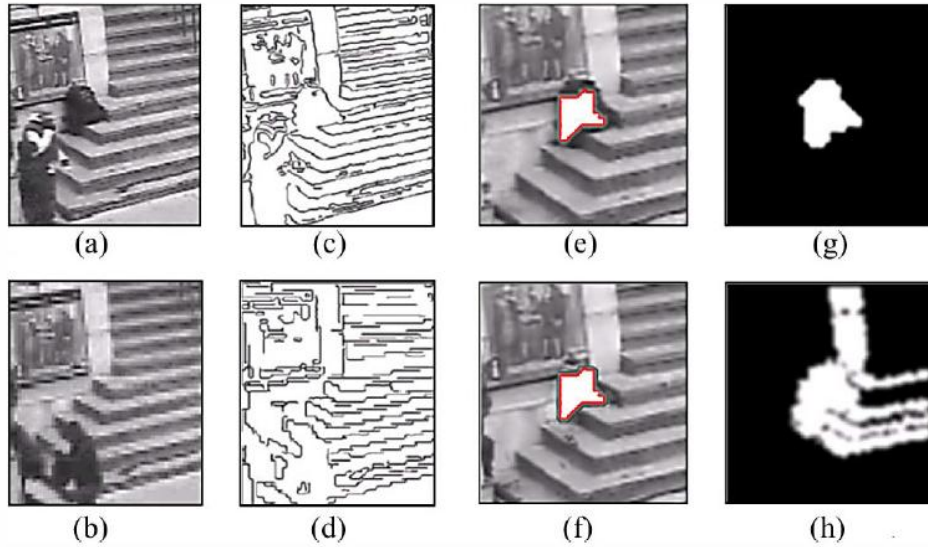


Figura 16 - Exemplo da detecção de um objeto abandonado. (Adaptada de [22]).

Depois de uma região estática ser classificada como abandonada ou removida, algumas condições precisam de ser verificadas antes de ser disparado um alarme.

Os seguintes parâmetros são definidos pelo utilizador:

- Tamanhos: tamanho mínimo e máximo do objeto em *pixels*.
- Regiões de interesse: áreas da imagem seleccionadas manualmente, no interior das quais se pretende proceder à detecção.
- Tempo mínimo: indica quanto tempo uma região do primeiro plano que corresponde a um objeto abandonado ou removido deve permanecer estática na cena, de modo a disparar um alerta.

### 3.5. Implementação do método de referência

De seguida, apresentamos algumas imagens com a implementação do módulo de detecção de objetos abandonados do método de referência.

A Figura 17 contém o plano de fundo estimado, a imagem de entrada e a respetiva máscara de segmentação. Como podemos observar, existe alguma fragmentação dos *blobs*. A Figura 18 e a Figura 19 mostram o resultado da implementação do descritor de similaridade local sobre a imagem de plano de fundo e a *frame* atual,

respetivamente. A localização do centroide do objeto estático detetado na máscara da Figura 20 é depois projetada na Figura 18 e na Figura 19 para confrontarmos o tamanho das zonas similares a essa localização, como demonstram a Figura 21 e a Figura 22.

Tal como referido anteriormente, a comparação entre os tamanhos (número de pixels) das áreas similares do modelo do plano de fundo e da *frame* atual, permitem avaliar se ocorreu o abandono de um objeto.



Figura 17 - Modelo do plano de fundo estimado, imagem de entrada e respetiva máscara de segmentação.



Figura 18 - Resultado da aplicação do descritor de similaridade local sobre o plano de fundo, e máscara com as arestas dessas zonas.

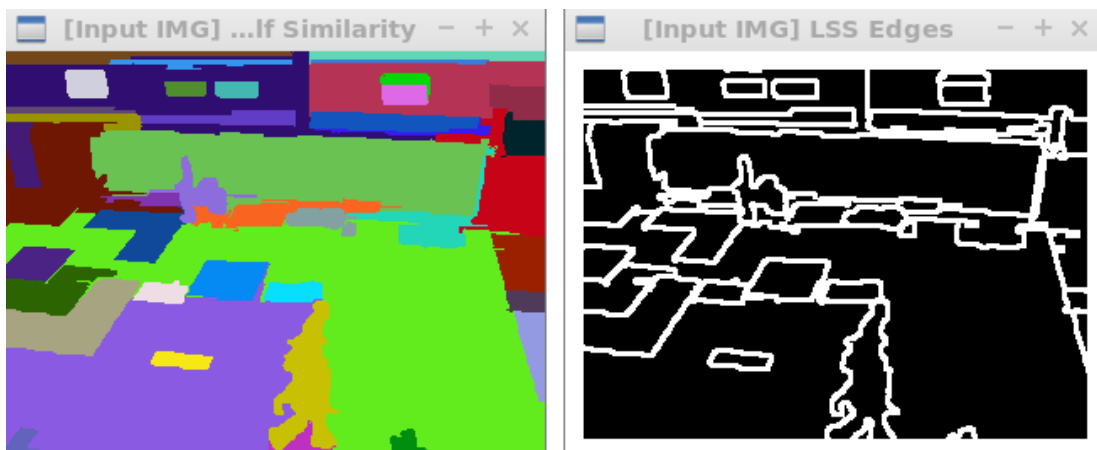


Figura 19 - Resultado da aplicação do descritor de similaridade local sobre a imagem de entrada, e máscara com as arestas dessas zonas.

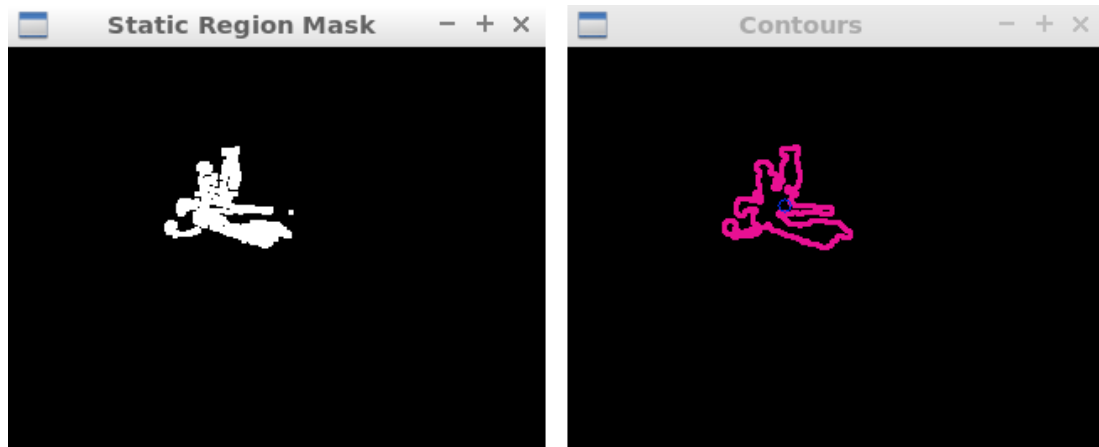


Figura 20 - Máscara da região estática e os seus contornos.

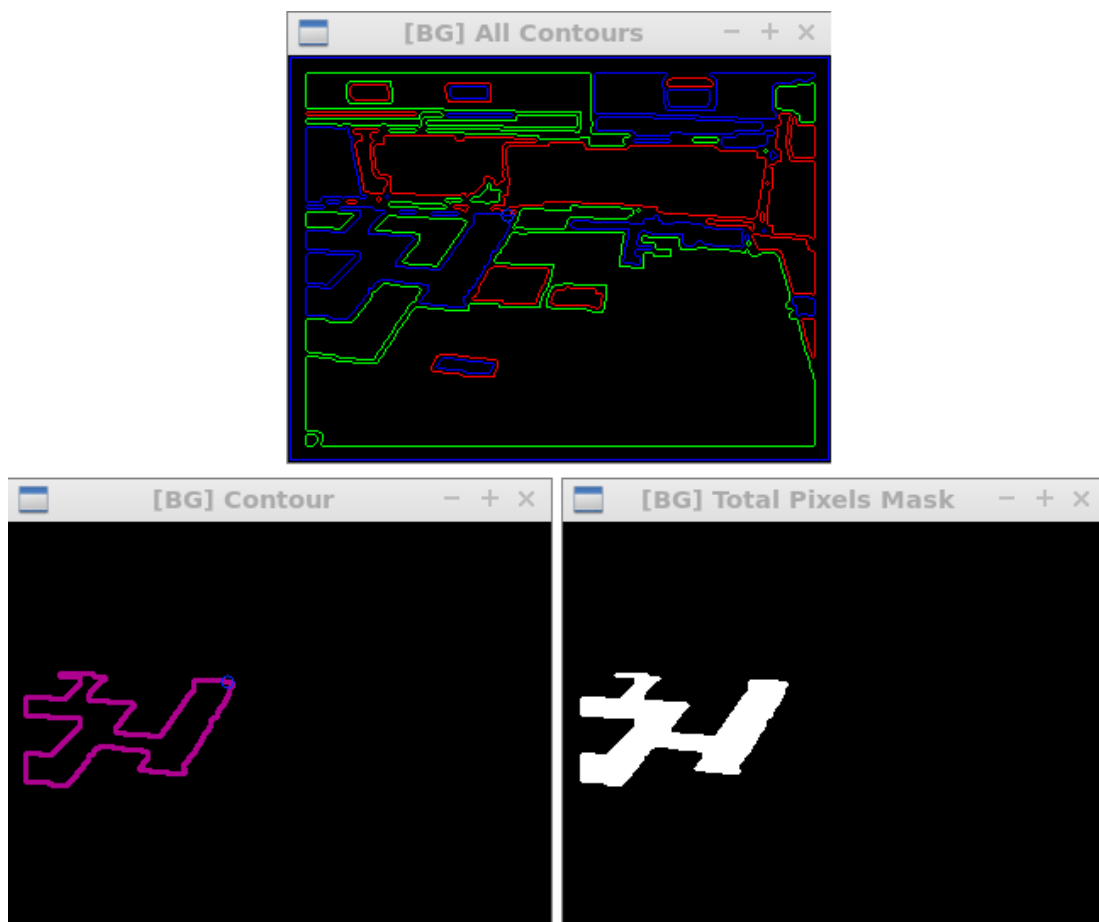


Figura 21 - Região segmentada do plano de fundo em função do centroide da máscara da região estática.

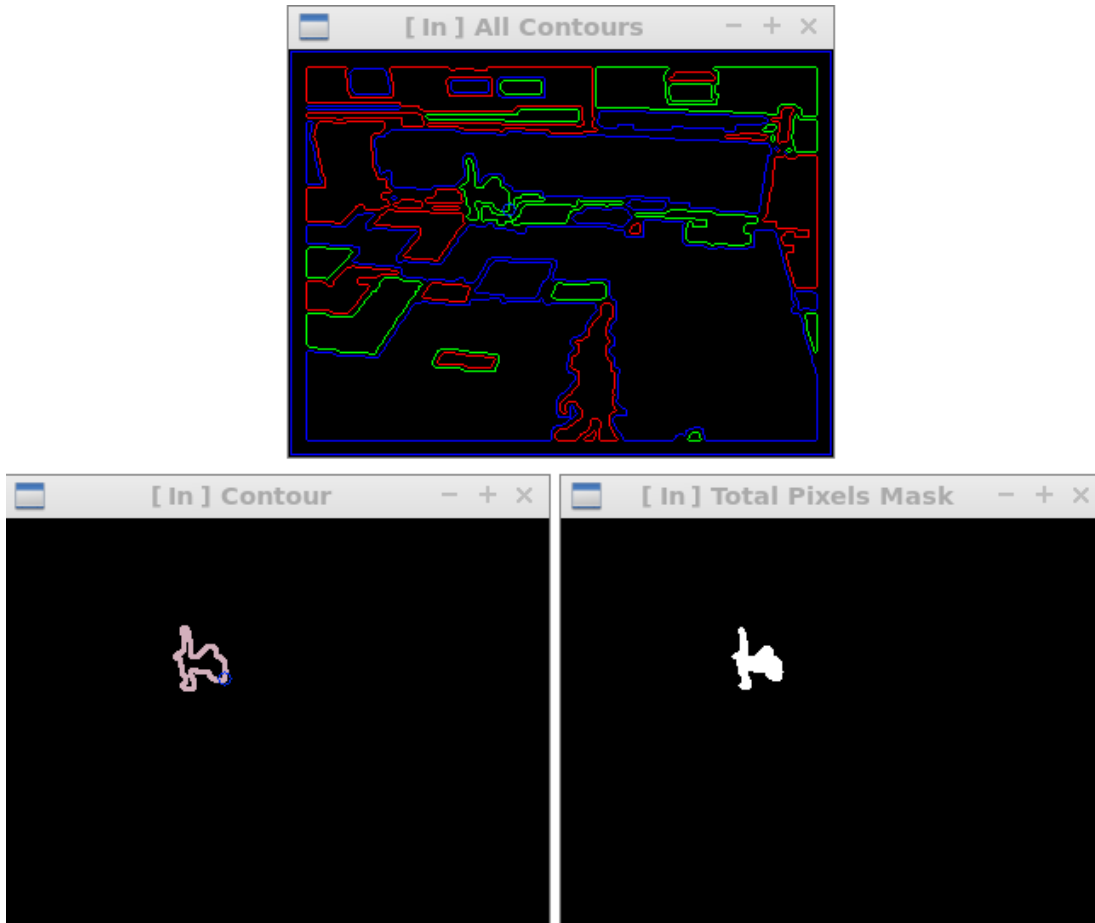


Figura 22 - Região segmentada da imagem de entrada em função do centroide da máscara da região estática.

### 3.6. Limitações

A precisão da detecção de objetos abandonados é influenciada negativamente por diversos fatores:

1. Os diversos limiares (*thresholds*) presentes no método tornam difícil a sua adaptação em cenários distintos, sem que ocorra um esforço considerável na parametrização destas variáveis. Os autores de [22] não indicam no artigo qual o valor ideal para utilizar como limiar de peso na segunda gaussiana (de forma a detetar os pixels que pertencem à máscara de objetos estáticos), nem quantas vezes a área da máscara de similaridade do plano de fundo deve ser maior do que a área da máscara de similaridade da *frame* atual para que se considere o objeto como abandonado.

2. São gerados vários falsos-alarmes na região estática durante os primeiros minutos de execução do método, até a atualização do modelo do plano de fundo estabilizar, especialmente se existirem objetos em movimento no instante da inicialização.
3. Fragmentos do primeiro plano devido a erros na subtração do plano de fundo;
4. Tamanho do objeto abandonado é demasiado pequeno ou está a sofrer de oclusão;
5. Condições climatéricas adversas, como a chuva e neve provocam o aumento do número de falsos-alarmes;
6. Com pouca iluminação é diminuída a capacidade de distinguir um objeto de outro, aumentando a taxa de erro;
7. A deteção de objetos estáticos em cenários extremamente lotados (trânsito, multidões, etc) é muito mais difícil, o que conduz a maiores taxas de erro;
8. Alterações na iluminação podem causar problemas. Se estas ocorrerem no intervalo de tempo em que um objeto é abandonado mas antes do alarme ser acionado, o *background* é atualizado com o item abandonado, sem que este seja detetado. Se a mudança de iluminação é apenas temporária, em seguida o sistema é capaz de recuperar utilizando o anterior modelo de fundo. Falsos-negativos neste cenário podem ocorrer somente quando a mudança é persistente.
9. Situações de baixo contraste, como por exemplo um saco preto sobre um fundo preto, podem levar a falhas na deteção.

## 4. Método Proposto

### 4.1. Pré-Processamento

Por forma a melhor preparar as imagens provenientes da câmara para as etapas de processamento, e assim aumentar a eficácia de todo o processo, implementaram-se as seguintes funcionalidades:

- **Equalização do histograma**

Permite obter a máxima variância do histograma, obtendo uma imagem com maior contraste.

- **Desfoque gaussiano**

Para suavização da imagem (passa-baixo).

- **Estabilização de vídeo**

Reduz o movimento angular do vídeo, quando este é obtido por uma câmara com posição instável durante o tempo de exposição (exemplo: câmara instalada num automóvel em andamento, ou num poste que oscila com o vento).

O processo consiste em:

1) Encontrar a transformação da *frame* anterior para a atual utilizando o fluxo óptico de todas as *frames*. A transformação consiste em três parâmetros:  $dx$ ,  $dy$ ,  $da$  (ângulo).

2) Acumular as transformações para obter a "trajetória" de  $x$ ,  $y$ , ângulo, em cada *frame*.

3) Suavizar a trajetória utilizando uma janela deslizante de  $X$  *frames* (número pré-definido).

4) Nova\_Transformação = Transformação + (Trajetória\_Suavizada - Trajetória)

5) Aplicar a nova transformação ao vídeo

## 4.2. Subtração do plano de fundo

A subtração de plano fundo é um passo fundamental da aplicação que consiste na comparação da imagem atualmente observada com outra imagem que representa uma estimativa do plano de fundo. As regiões que possuem uma diferença acima de um determinado limiar entre a imagem de entrada e a imagem de referência (plano de fundo ou *background*) indicam a localização de objetos de interesse (primeiro plano ou *foreground*).

Para esta etapa foi escolhido o algoritmo **SuBSENSE (Self-Balanced SENSitivity SEgmenter)** [36].

Como o algoritmo GMM [5] utilizado pelo método de referência apresenta bastantes limitações, decidimos analisar alguns *benchmarks* acadêmicos que se focam no teste e classificação de algoritmos de detecção de mudança e movimento em vídeos. Os workshops “*IEEE Change Detection*” de 2012 [37] e de 2014 [38] fornecem *datasets* muito interessantes, assim como os resultados alcançados pelos diversos algoritmos nesses desafios.

A Tabela 1 e a Figura 23 apresentam a comparação dos resultados entre o método que selecionamos e o GMM [5] para o *dataset* CDnet 2012 [37], e de igual forma a Tabela 2 e a Figura 24 mostram os resultados para *dataset* CDnet 2014 [38].

**Tabela 1 - Comparação entre os diferentes métodos de Subtração de Plano de Fundo (BGS) com as pontuações gerais e por categoria para o *dataset* CDnet 2012 [37]. (Adaptada de [36]).**

| <i>Método</i><br>BGS | <i>FScore</i><br>CDnet<br>(2012) | <i>FScore</i><br>Baseline | <i>FScore</i><br>Cam.<br>Jitter | <i>FScore</i><br>Dyn. BG | <i>FScore</i><br>Interm.<br>Object<br>Motion | <i>FScore</i><br>Shadow | <i>FScore</i><br>Thermal |
|----------------------|----------------------------------|---------------------------|---------------------------------|--------------------------|--|-------------------------|--------------------------|
| SuBSENSE [36]        | <b>0.826</b>                     | <b>0.950</b>              | <b>0.815</b>                    | <b>0.818</b>             | 0.657  | <b>0.899</b>            | <b>0.817</b>             |
| CwisarD              | 0.778                            | 0.908                     | 0.781                           | 0.809                    | 0.567  | 0.841                   | 0.762                    |
| Spectral-360         | 0.777                            | 0.933                     | 0.716                           | 0.787                    | 0.566  | 0.884                   | 0.776                    |
| DPGMM                | 0.776                            | 0.929                     | 0.748                           | 0.814                    | 0.542  | 0.813                   | 0.813                    |
| SGMM-SOD             | 0.766                            | 0.921                     | 0.672                           | 0.688                    | <b>0.715</b>                                 | 0.865                   | 0.735                    |
| PBAS                 | 0.753                            | 0.924                     | 0.722                           | 0.683                    | 0.575  | 0.860                   | 0.756                    |
| PSP-MRF              | 0.737                            | 0.929                     | 0.750                           | 0.696                    | 0.565  | 0.791                   | 0.693                    |

|         |       |       |       |       |       |       |       |
|---------|-------|-------|-------|-------|-------|-------|-------|
| SC-SOBS | 0.728 | 0.933 | 0.705 | 0.669 | 0.592 | 0.779 | 0.692 |
| ViBe+   | 0.722 | 0.871 | 0.754 | 0.720 | 0.509 | 0.815 | 0.665 |
| KDE     | 0.672 | 0.909 | 0.572 | 0.596 | 0.409 | 0.803 | 0.742 |
| ViBe    | 0.668 | 0.870 | 0.600 | 0.565 | 0.507 | 0.803 | 0.665 |
| GMM [5] | 0.662 | 0.825 | 0.597 | 0.633 | 0.520 | 0.737 | 0.662 |

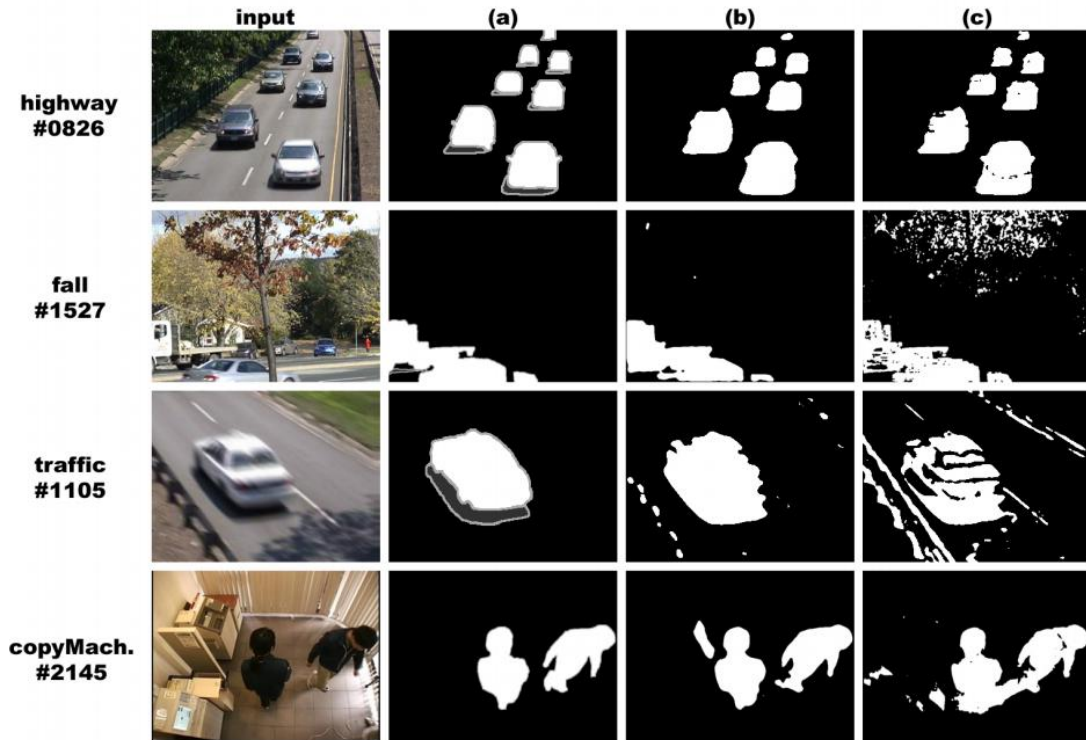


Figura 23 - Exemplo da aplicação dos métodos de BGS no *dataset* CDnet 2012 [37]. (a) *Ground truth* para a *frame* atual; (b) Resultados SuBSENSE [36]; (c) Resultados GMM [5]. (Adaptada de [36]).

Tabela 2 - Comparação entre os diferentes métodos de Subtração de Plano de Fundo (BGS) com as pontuações gerais e por categoria para o *dataset* CDnet 2014 [38]. (Adaptada de [36]).

| Método BGS    | FScore CDnet (2012+ 2014) | FScore CDnet 2014 | FScore Bad Weath. | FScore Low FPS | FScore Night Videos | FScore PTZ | FScore Turbulence |
|---------------|---------------------------|-------------------|-------------------|----------------|---------------------|------------|-------------------|
| SuBSENSE [36] | 0.741                     | 0.639             | 0.862             | 0.645          | 0.560               | 0.348      | 0.779             |
| FTSG          | 0.728                     | 0.600             | 0.823             | 0.626          | 0.513               | 0.324      | 0.713             |
| CwisarDH      | 0.681                     | 0.549             | 0.684             | 0.641          | 0.374               | 0.322      | 0.723             |
| Spectral-360  | 0.673                     | 0.558             | 0.757             | 0.644          | 0.483               | 0.365      | 0.543             |
| Bin Wang's    | 0.658                     | 0.501             | 0.767             | 0.469          | 0.380               | 0.135      | 0.756             |

|         |       |       |       |       |       |       |       |
|---------|-------|-------|-------|-------|-------|-------|-------|
| SC-SOBS | 0.596 | 0.437 | 0.662 | 0.546 | 0.450 | 0.041 | 0.488 |
| KNN     | 0.594 | 0.492 | 0.759 | 0.549 | 0.420 | 0.213 | 0.520 |
| KDE     | 0.571 | 0.445 | 0.757 | 0.548 | 0.436 | 0.037 | 0.448 |
| GMM [5] | 0.569 | 0.461 | 0.738 | 0.537 | 0.410 | 0.152 | 0.466 |

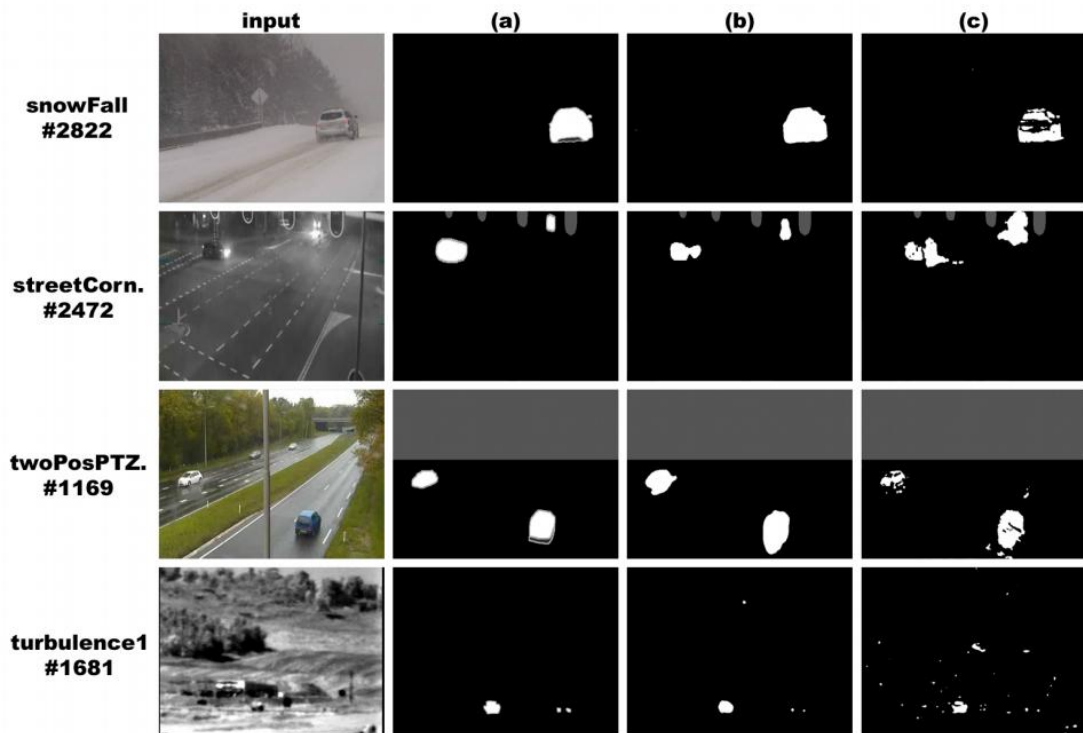


Figura 24 - Exemplo da aplicação dos métodos de BGS no *dataset* CDnet 2014 [38]. (a) *Ground truth* para a *frame* atual; (b) Resultados SuBSENSE [36]; (c) Resultados GMM [5]. (Adaptada de [36]).

O processo de subtração de fundo inclui a inicialização e manutenção do modelo do plano de fundo (Figura 26), e através da diferença entre este modelo e a imagem de entrada (Figura 25) podemos realizar a detecção do primeiro plano (Figura 27). No SuBSENSE [36] isto é feito com base na análise da alteração da cor de cada *pixel*, e em padrões binários de similaridade local (Local Binary Similarity Patterns - LBSP).

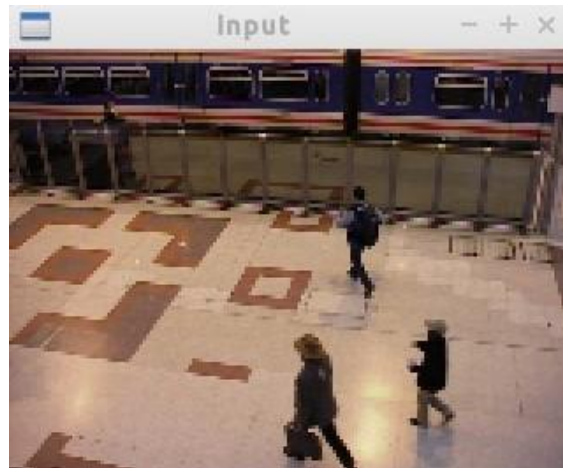


Figura 25 - Imagem de entrada.



Figura 26 - Modelo estimado do plano de fundo.

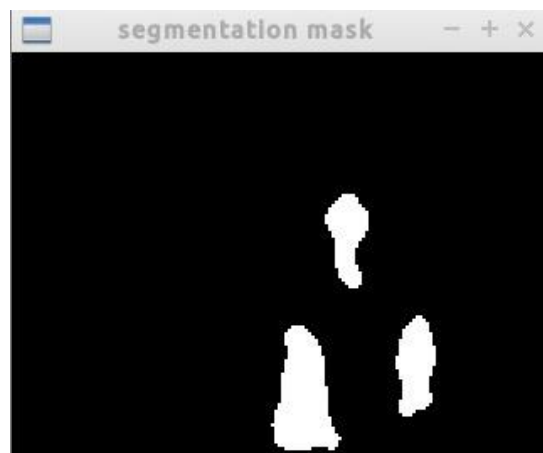


Figura 27 - Segmentação dos pixels detetados como primeiro plano.

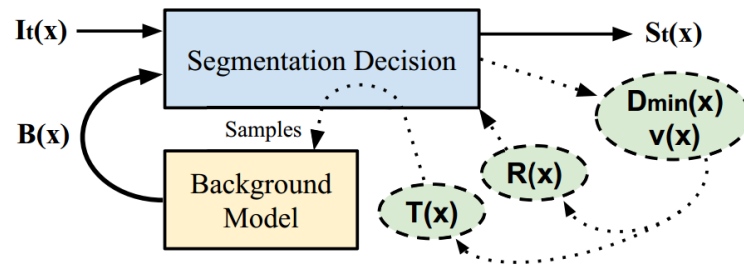


Figura 28 - Esquema do SuBSENSE. (Adaptada de [36]).

A Figura 28 representa a visão geral do funcionamento do algoritmo SuBSENSE [36]. As linhas a tracejado indicam mecanismos de feedback. Neste contexto  $I_t(\mathbf{x})$  consiste na representação RGB/LBSP do pixel  $x$  obtida na imagem atual da sequência a analisar.  $B(\mathbf{x})$  contém  $N$  amostras assinaladas como plano de fundo para o pixel em questão.  $St(\mathbf{x})$  é o valor de saída da segmentação.  $R(\mathbf{x})$  controla os limiares das decisões internas de segmentação.  $T(\mathbf{x})$  controla os limiares de atualização do plano de fundo, e finalmente tanto  $Dmin(\mathbf{x})$  como  $v(\mathbf{x})$  controlam dinamicamente as variáveis anteriores ao monitorizar o resultado do plano de fundo.

O algoritmo SuBSENSE [36] procede à modelação do plano de fundo com sensibilidade de aprendizagem a nível local, que é automaticamente ajustada utilizando um sistema de *feedback*.

Cada *pixel* do plano de fundo é caracterizado por um conjunto de  $N$  amostras. Estas amostras contêm a informação de cor RGB e das características de padrões binários locais de similaridade (LBSP).

Os autores sugerem um valor de 35 a 50 amostras por *pixel* para melhores resultados. Tipicamente este valor  $N$  é utilizado para equilibrar a precisão e sensibilidade dos métodos baseados em amostras. A utilização de menos amostras gera modelos mais sensíveis mas menos precisos, e vice-versa.

As amostras são comparadas com a respetiva observação da imagem de entrada no tempo  $t$ , para classificar o *pixel* na coordenada  $x$  como *foreground* (1) ou *background* (0):

$$S_t(x) = \begin{cases} 1, & \text{se } \# \{ \text{dist}(I_t(x), B_n(x)) < R, \forall n \} < \#_{min} \# \\ 0, & \text{em caso contrário} \end{cases} \quad (7)$$

$S_t$  é o mapa resultante da segmentação, e  $\text{dist}(I_t(x), B_n(x))$  retorna a distância entre a observação atual e uma determinada amostra do plano de fundo,  $R$  é o *threshold* de distância máxima e  $\#_{min}$  o número mínimo de *samples* necessários por *pixel* para modelar o *background*.

Neste contexto um valor de  $R$  pequeno significa que o modelo tem de ser muito preciso de modo a classificar os *pixels* como plano de fundo. Usando um valor de  $R$  maior leva ao aumento da resistência contra mudanças irrelevantes, mas também torna mais difícil detetar objetos do primeiro plano que sejam muito semelhantes ao plano de fundo.

Os autores de [36] defendem que  $\#_{min} = 2$  é um compromisso razoável entre a resistência ao ruído e a complexidade computacional.

O modelo do plano de fundo é ajustado para cada *pixel* da seguinte forma:

- 1) Cada *pixel* do plano de fundo tem uma probabilidade (ajustada automaticamente) de ser substituído.

O facto destas amostras serem aleatoriamente substituídas em vez de se ter em conta a última vez que foram modificadas garante um forte histórico de representações do *background* tanto de curto prazo como de longo prazo.

De igual forma, uma vez que as novas amostras só podem ser inseridas quando um *pixel* local é reconhecido como plano de fundo, esta abordagem evita que objetos estáticos do primeiro plano sejam assimilados muito rapidamente (algo que é frequente acontecer em métodos que utilizam estratégias de atualização clássicas).

O trabalho [36] refere que em teoria, esta abordagem de atualização implica que existindo contraste suficiente, alguns objetos do primeiro plano nunca seriam incorporados no modelo de fundo.

Na prática, o ruído causa sempre uma erosão gradual do primeiro plano, o que significa que todos os objetos estáticos acabarão por ser classificados como plano de fundo.

- 2) A segunda etapa de atualização é um passo de difusão espacial que permite que regiões homogêneas com o plano de fundo possam ser absorvidas de forma muito mais rápida.

Por outras palavras, artefactos que poderiam permanecer no modelo do plano de fundo devido à remoção de um objeto que estava antes em cena, podem ser eliminados rapidamente uma vez que são muito semelhantes às outras partes do *background*. Este passo de difusão melhora também a coerência espacial do plano de fundo, ao ponto de ligeiros movimentos da câmara poderem ser tolerados.

O recurso à informação da textura presente nas características LBSP evita a propagação das amostras além dos limites dos objetos. Simplificando, mesmo quando uma amostra é propagada erradamente de uma região da imagem para outra, a hipótese de obter correspondência à nova região é muito baixa devido à utilização das características LBSP que podem detetar uma alteração ao nível da textura perto da fronteira dos objetos.

Um objeto do primeiro plano que se encontre estático e que apresente uma cor semelhante à do plano de fundo pode ser corretamente classificado como *foreground* por um longo período de tempo, desde que a sua fronteira seja perceptível.

### 4.3. Remoção de Sombras

Em certas condições a remoção de sombras (Figura 29) é um passo crítico para aumentar a fiabilidade na deteção e seguimento de objetos. O problema surge na etapa de subtração de plano de fundo, quando vários objetos diferentes são detetados formando um único *blob* devido às sombras propagadas que os unem. As sombras afetam também o modelo de aparência dos objetos, prejudicando os resultados dos algoritmos de classificação e reconhecimento.

O algoritmo de subtração de plano de fundo escolhido (SuBSENSE [36]) já possui uma grande resistência a sombras, ainda assim optou-se por incluir na ferramenta desenvolvida alguns métodos que podem ser utilizados adicionalmente: [39], [40], [41], [42] e [43].

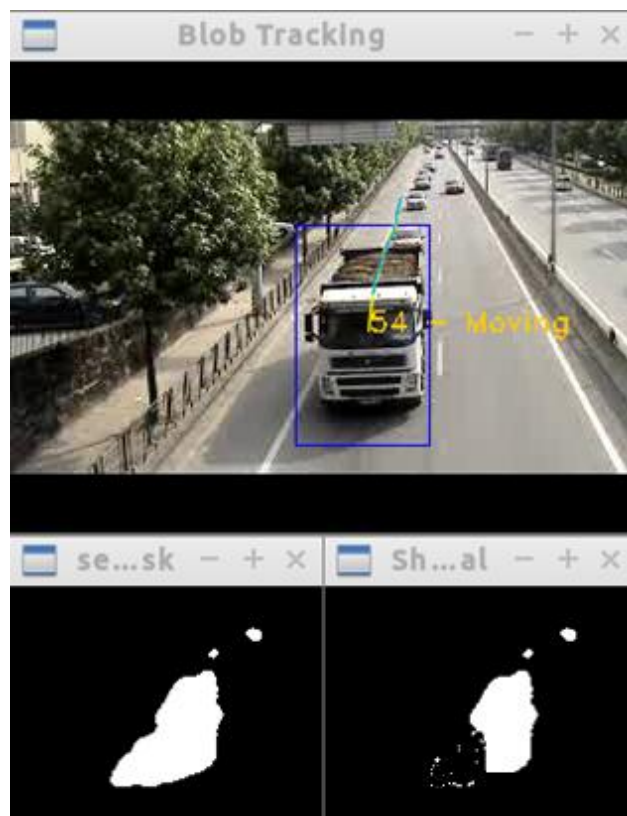


Figura 29 - No canto inferior direito, máscara resultante da aplicação do método de remoção de sombras [39].

#### 4.4. Seguimento (*Tracking*)

O seguimento de um objeto no período durante o qual se encontra no campo de visão do sistema é um passo fulcral de toda a aplicação.

De forma a armazenar a trajetória de cada um dos objetos, é utilizada a informação do ponto que representa o seu centro de massa ao longo da sequência de imagens do vídeo.

Para evitar erros quando o objeto não é detetado numa determinada imagem, ou quando vários objetos se cruzam criando oclusões na perspetiva da câmara, utilizamos para cada um dos objetos um filtro de *Kalman* de forma a estimar a sua posição atual com base na velocidade e direção prévia.

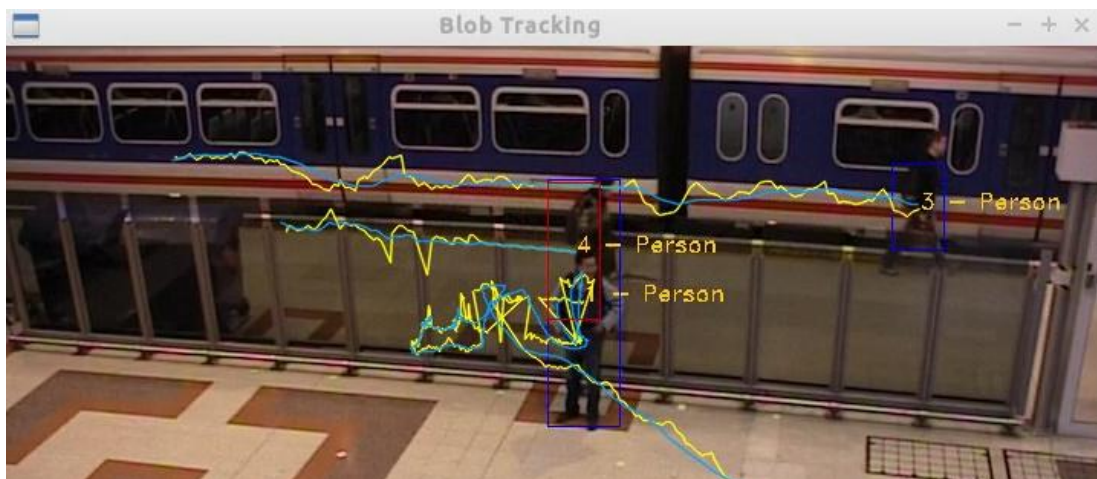


Figura 30 - Linhas amarelas representam a leitura real da posição. Linhas azuis representam a estimativa do filtro de *Kalman*.

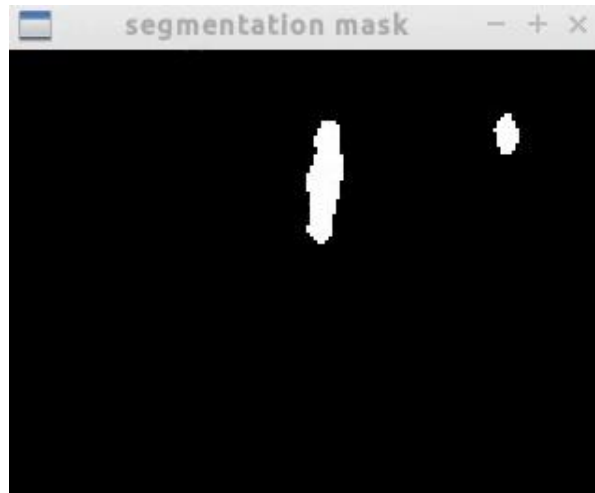


Figura 31 - *Pixels* detetados como primeiro plano para o exemplo da Figura 30.

Na Figura 30 podemos observar um caso em que a pessoa com ID 4 (caixa a vermelho) está a sofrer oclusão por parte da pessoa com ID 1, estando apenas a ser utilizado o filtro de Kalman para estimar a sua posição atual.

Devido a esta oclusão temporária as duas pessoas com ID 1 e 4 surgem unidas na máscara de segmentação (Figura 31), formando um único objeto. De forma a reduzirmos o erro que estas oclusões podem provocar no nosso sistema é particularmente útil termos o historial da trajetória dos objetos e a previsão da sua deslocação nos instantes seguintes.

#### 4.5. Classes dos objetos em movimento

Após a deteção de um novo objeto em movimento este pode ser classificado numa das seguintes classes: desconhecido, pessoa ou automóvel.

Ao longo das diversas *frames* que cada objeto está em cena, os classificadores para deteção pessoas e de automóveis são executados. Caso ocorra uma deteção bem-sucedida é incrementado um voto (automóvel ou pessoa) na respetiva classe desse objeto. Caso seja superado um mínimo previamente estipulado de classificações bem-sucedidas, o objeto é classificado como pertencente à classe que obteve maior votação, caso contrário permanece na classe “desconhecido”.

## 4.6. Detecção de automóveis

A deteção de automóveis (Figura 32) recorre a classificadores Haar Cascade (Figura 33) treinados para imagens que contêm a frente ou a traseira dos veículos.

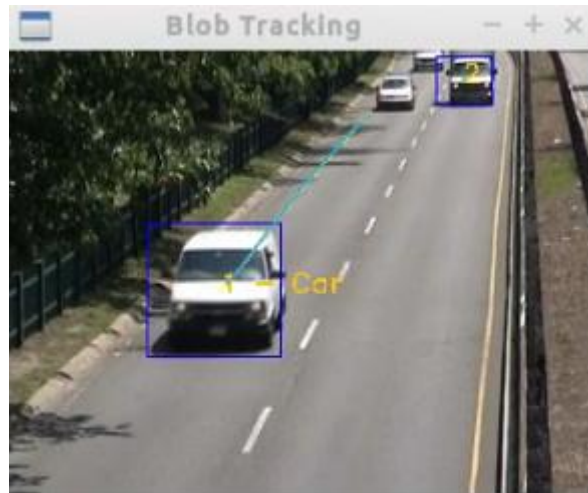


Figura 32 - Exemplo da deteção de automóveis.



Figura 33 - Classificador deteta com sucesso o automóvel com ID 2 da Figura 32.

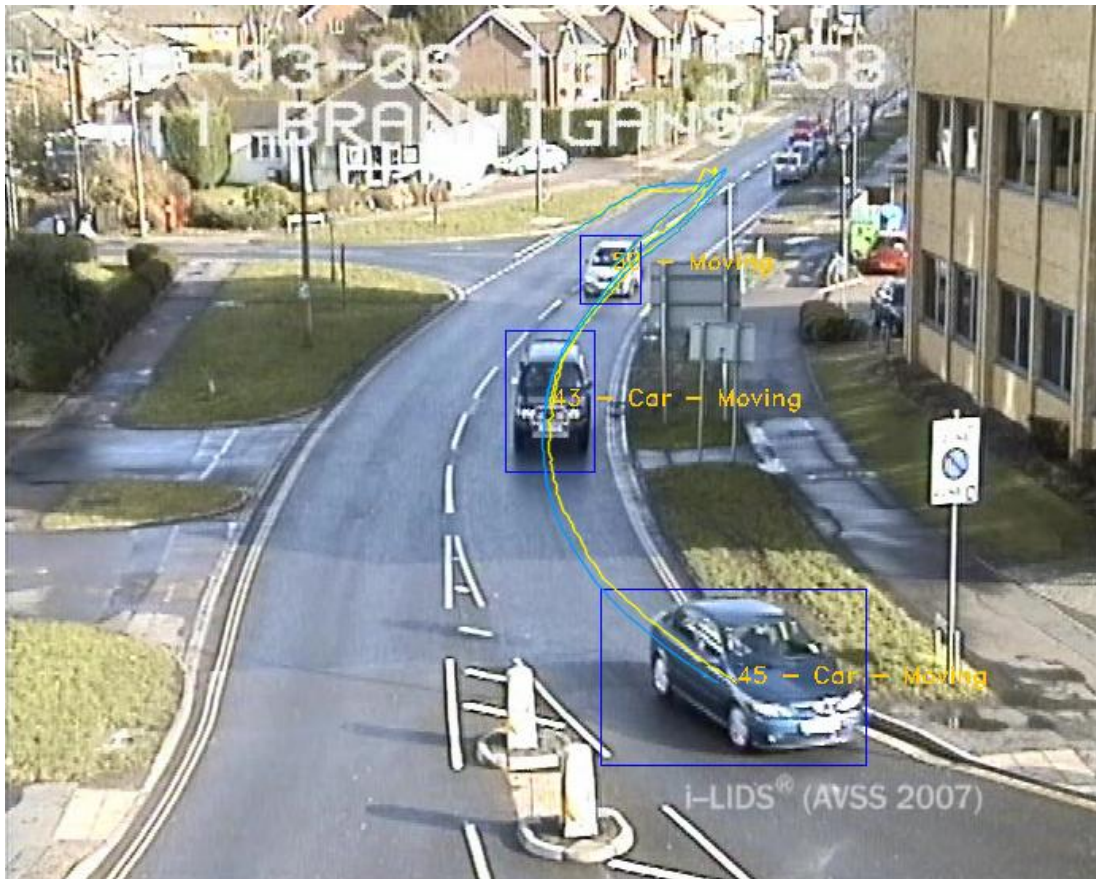


Figura 34 - Resultado da detecção de automóveis no dataset AVSS PV Medium.

## 4.7. Detecção de pessoas

Para a detecção de humanos foram implementados os seguintes métodos:

### 4.7.1. Detecção simples

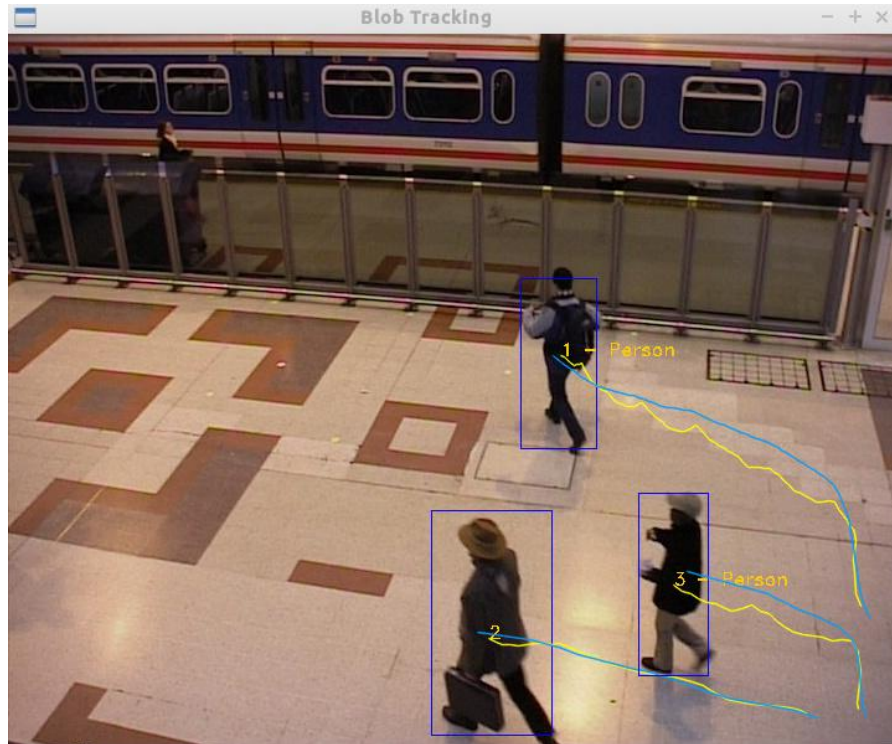


Figura 35 - Exemplo da detecção de pessoas.

Procedemos à detecção da silhueta das pessoas (Figura 35) utilizando a técnica HOG (Histograms Of Oriented Gradients).

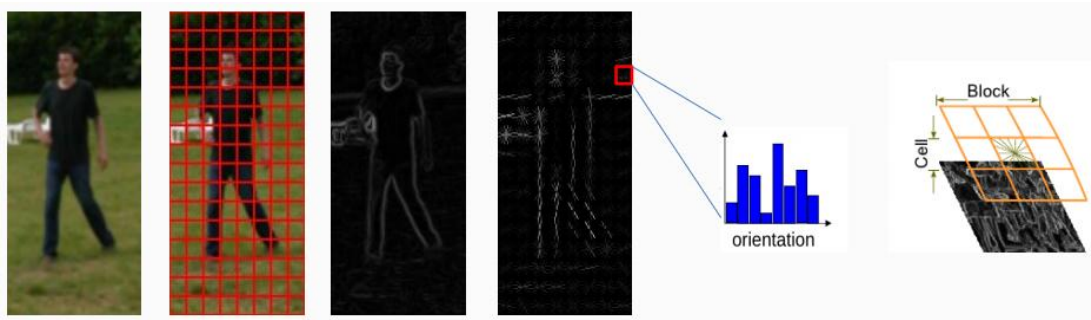


Figura 36 - Algoritmo HOG - Histograms of Oriented Gradients [14].

Conforme podemos observar na Figura 36 o algoritmo HOG [14] divide a imagem em várias células (por exemplo, quadrados de  $8 \times 8$  pixels). Para cada célula, o histograma ponderado do gradiente é calculado sobre 8 diferentes orientações (ângulos entre 0 e 180 graus). O histograma é posteriormente normalizado em blocos maiores, e a classificação feita com recurso a uma SVM linear.

O HOG tem um excelente desempenho pois a característica extraída é capaz de descrever formas complexas e as arestas dos objetos de forma eficaz, sendo robusta contra pequenas deformações.

Além disso, apresenta uma boa invariância a diferentes condições de iluminação e contraste, e é inspirada no SIFT [44] (*Scale-invariant feature transform*).

#### 4.7.2. Head and Shoulders Haar Cascade - Múltiplas Pessoas

É possível ativar um classificador *haar cascade* para deteção de cabeça e ombros (HS - *Head and Shoulders*). Desta forma podemos identificar um grupo de pessoas ainda que as mesmas se desloquem relativamente próximas e formem um único *blob* na máscara de deteção de movimento (Figura 37).

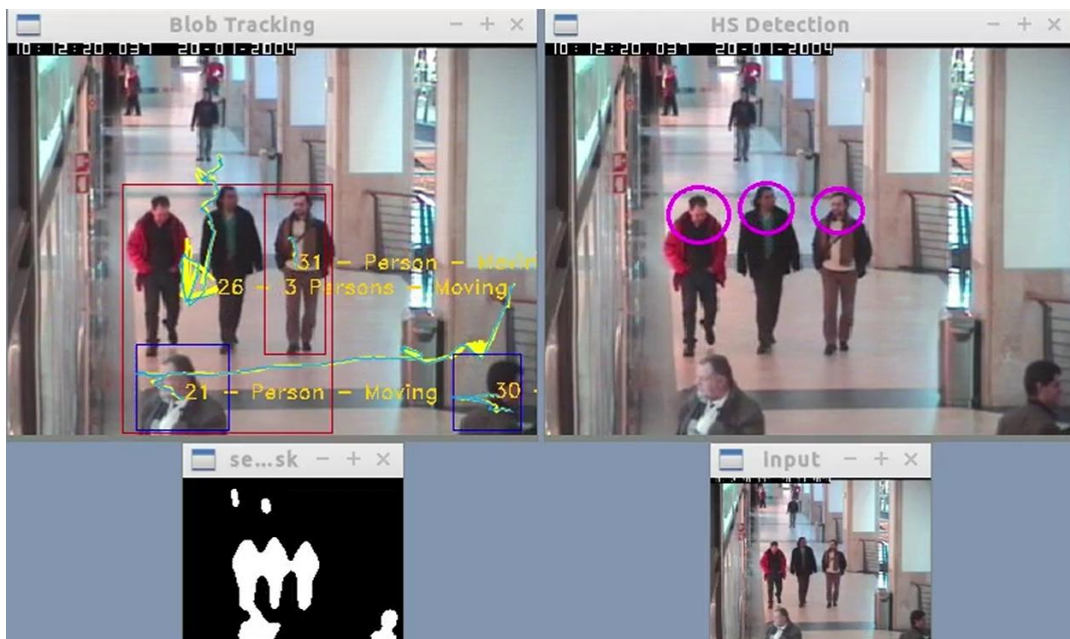


Figura 37 - Exemplo da deteção de um grupo de pessoas com o classificador *Head and Shoulders*.

#### 4.8. Deteção de objetos abandonados

Após alterarmos o método de subtração de fundo, optamos por alterar também o método de deteção de objetos abandonados.

O SuBSENSE [36] possui uma camada de pós-processamento que procede à suavização da imagem com filtro mediano e aplica algumas operações morfológicas na máscara de primeiro plano, de forma a eliminar *blobs* irrelevantes e a preencher os buracos no seu interior.

As imagens de segmentação do primeiro plano resultantes são estáveis e apresentam poucos fragmentos (ver máscara binária no canto inferior esquerdo da Figura 38) comparativamente às do método GMM [5] que é muito sensível a variações de iluminação e ao próprio movimento dos objetos ao longo do plano de fundo.

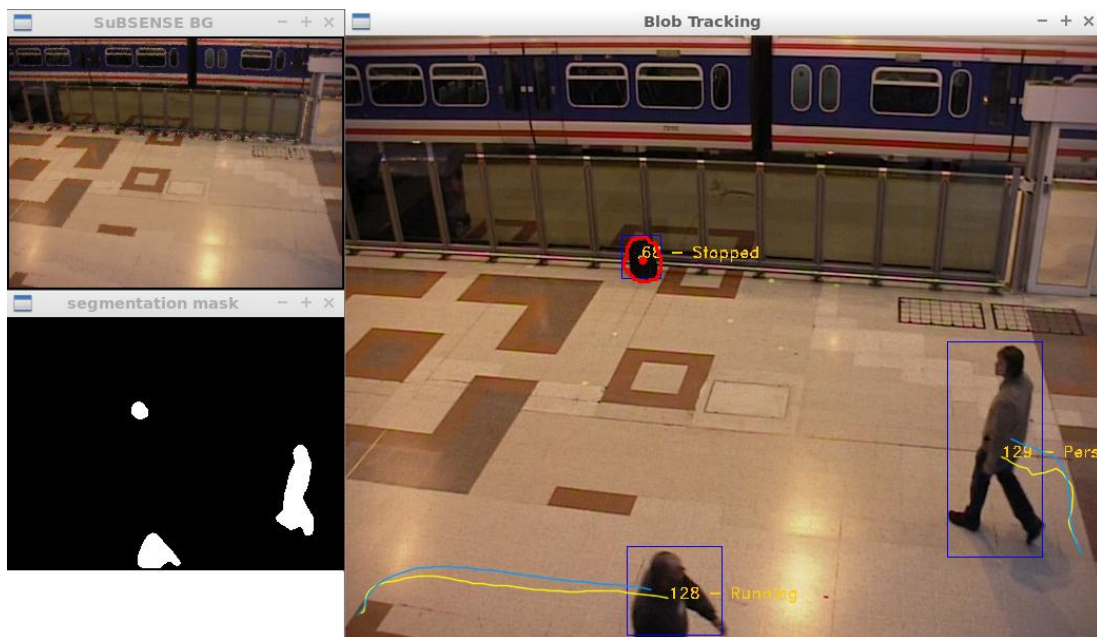


Figura 38 - Alarme de mala abandonada em estação de metro - *Dataset PETS 2006*.

Gravamos um historial das máscaras do primeiro plano, adicionando um novo exemplar a cada 30 segundos, tal como exemplificado na Figura 39. Este valor revelou-se razoável para os vídeos dos *datasets* que utilizamos para testes, mas poderá ter que ser ajustado em outros contextos.

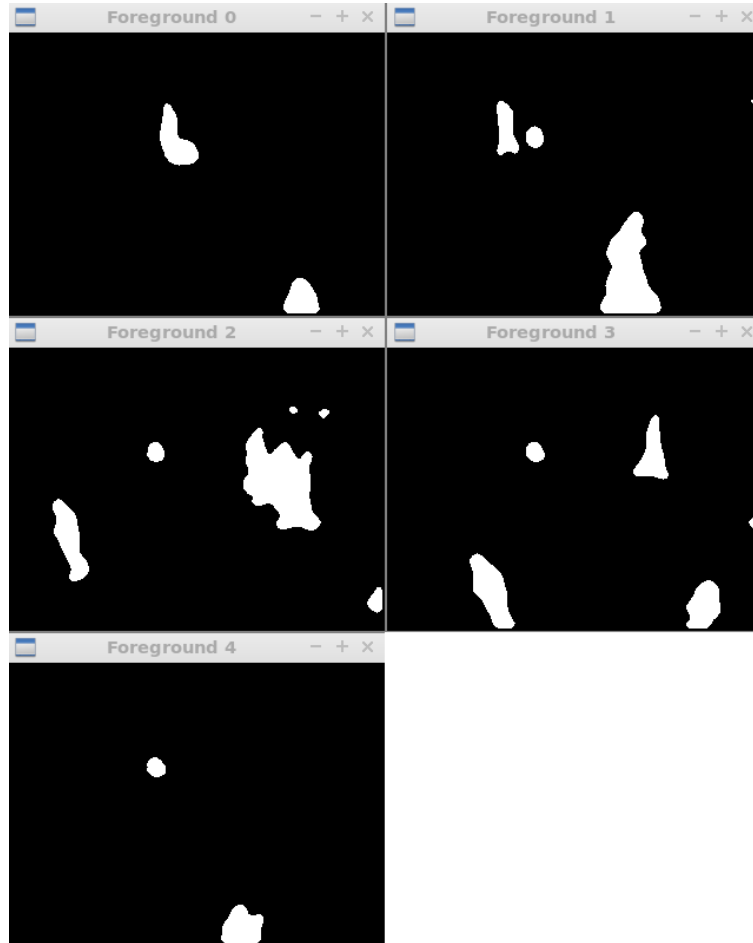


Figura 39 - Máscaras de primeiro plano capturadas a cada 30 segundos, que originaram o alarme da Figura 38.

O nosso historial é uma janela móvel que poderá conter até 10 máscaras de primeiro plano, ou seja, a análise dos últimos 5 minutos.

São feitos AND's binários do historial de forma faseada. Começamos por agrupar apenas as 3 máscaras mais recentes, e vamos acrescentando progressivamente uma mais antiga que as antecede. Se ao longo deste processo os blobs resultantes dos AND's binários (Figura 40) possuírem o mesmo modelo de aparência em pelo menos 3 das imagens de input gravadas simultaneamente com o historial das máscaras do primeiro plano (a cada 30 segundos), então assinalamos o objeto como abandonado.

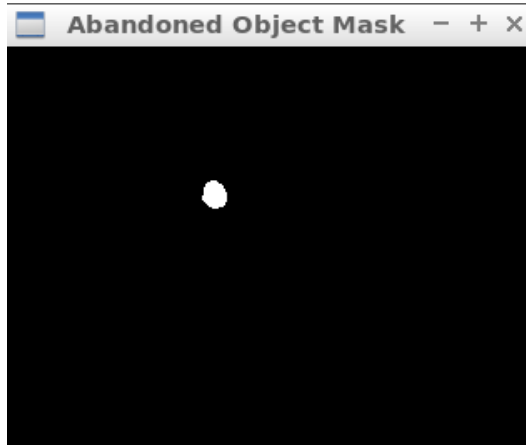


Figura 40 - Máscara de objeto abandonado obtida após análise das máscaras de primeiro plano da Figura 39.

Isto permite uma grande resistência a oclusões tal como mostra a Figura 41. Por exemplo, se um objeto for observado durante um minuto, sofrer uma oclusão total e prolongada por mais de 2 minutos e de seguida voltar a ser observado, poderá ainda assim ser classificado como abandonado.

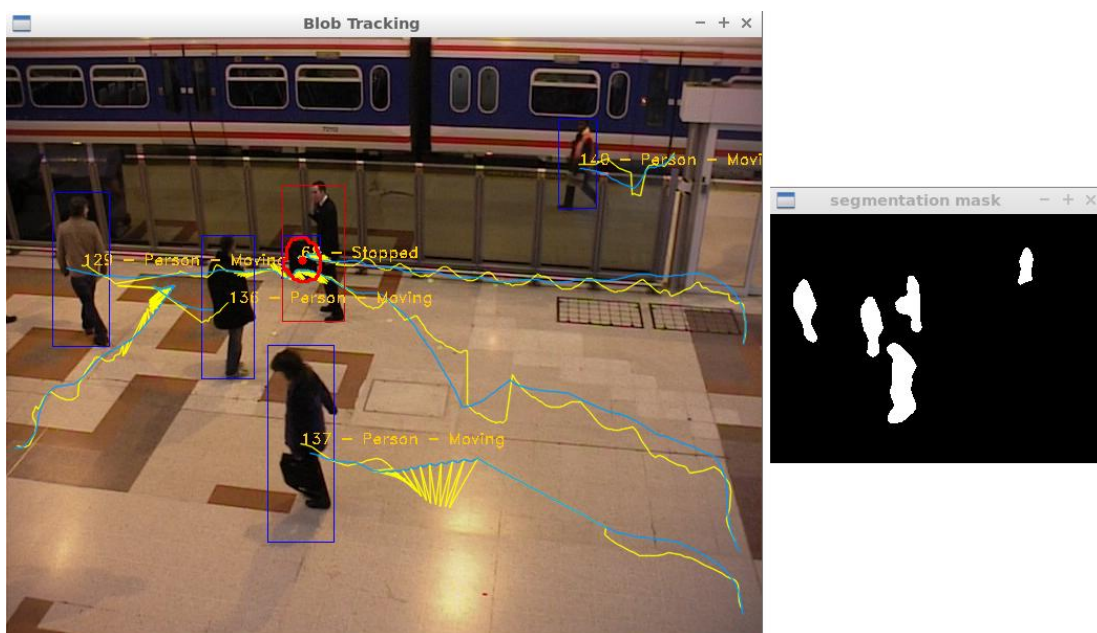


Figura 41 - Resistência a oclusões - Dataset PETS 2006.

## 4.9. Classificação do tipo de objeto abandonado

Optamos por introduzir esta camada na aplicação pois a classificação do tipo de objeto que foi abandonado permite obter uma maior contextualização ao nível do disparo de alarmes, sem que seja forçosamente necessária a intervenção de um humano para avaliar o conteúdo da imagem e classificar posteriormente o objeto. Permite assim pré-configurar filtros de seleção ou exclusão das classes de objetos abandonados que geram alarme, de forma a ajustar a aplicação ao ambiente em que é utilizada. Podemos definir previamente algumas classes de objetos que no ambiente em questão o seu abandono ou não representam um fator crítico, evitando assim disparar alarmes desnecessários. Ou seguir uma abordagem contrária, enunciando as classes de objetos que devem gerar alarmes.

Para o treino e utilização da rede neuronal responsável pela classificação dos objetos abandonados utilizamos o Caffe [45].

O Caffe [45] é uma framework simples e adaptável que fornece suporte ao treino e desenvolvimento geral de redes neuronais convolucionais (Figura 42) e outros modelos de *deep learning*. A sua licença de utilização é BSD, foi desenvolvido em C++ mas possui também *bindings* para Python e MATLAB. Além do processamento convencional por CPU possibilita também a utilização do processamento de placas gráficas (GPU) através de CUDA. É uma ferramenta muito utilizada para investigação, aplicações industriais de grande escala, e prototipagem de projetos na área de visão computacional, reconhecimento de fala, e multimédia.

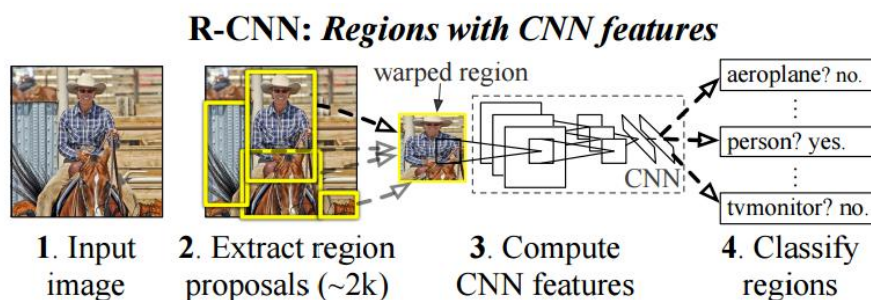


Figura 42 - As etapas do processo R-CNN para deteção de objetos utilizado pelo Caffe [45].

Para a rede neuronal utilizamos algumas das classes da ImageNet [46], que é um conjunto de imagens organizadas de acordo com a hierarquia da WordNet [47] (os substantivos, que são cerca de 80.000), em que cada nodo da hierarquia é representado por centenas ou milhares de imagens. Atualmente disponibiliza em média 500 imagens por nodo (Figura 43), com classificações de qualidade anotadas manualmente por humanos.

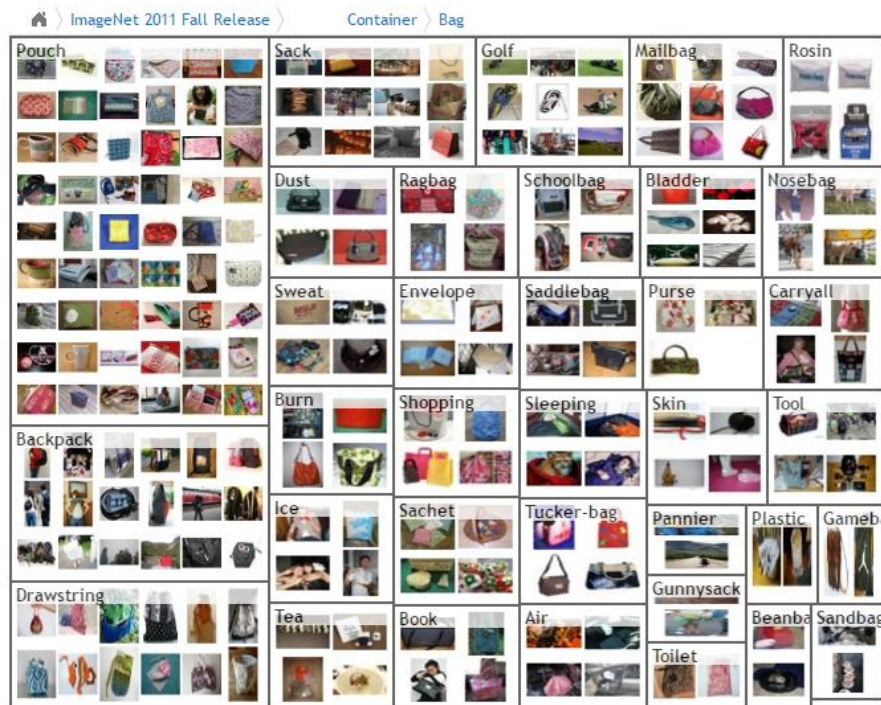


Figura 43 - ImageNet: Sub-classes da classe “bag” (saco).

## 5. Resultados e Discussão

Neste capítulo apresentamos resultados da aplicação do nosso método em alguns vídeos dos *datasets* i-LIDS [23], PETS 2006 [24], PETS 2007 [48] e ABODA [49].

### 5.1. Casos de sucesso

Na Figura 44 mostramos a detecção eficaz de uma mala abandonada numa estação de metro. A Figura 45 contém as últimas máscaras de primeiro plano guardadas e a Figura 46 a máscara correspondente de detecção de objeto abandonado.

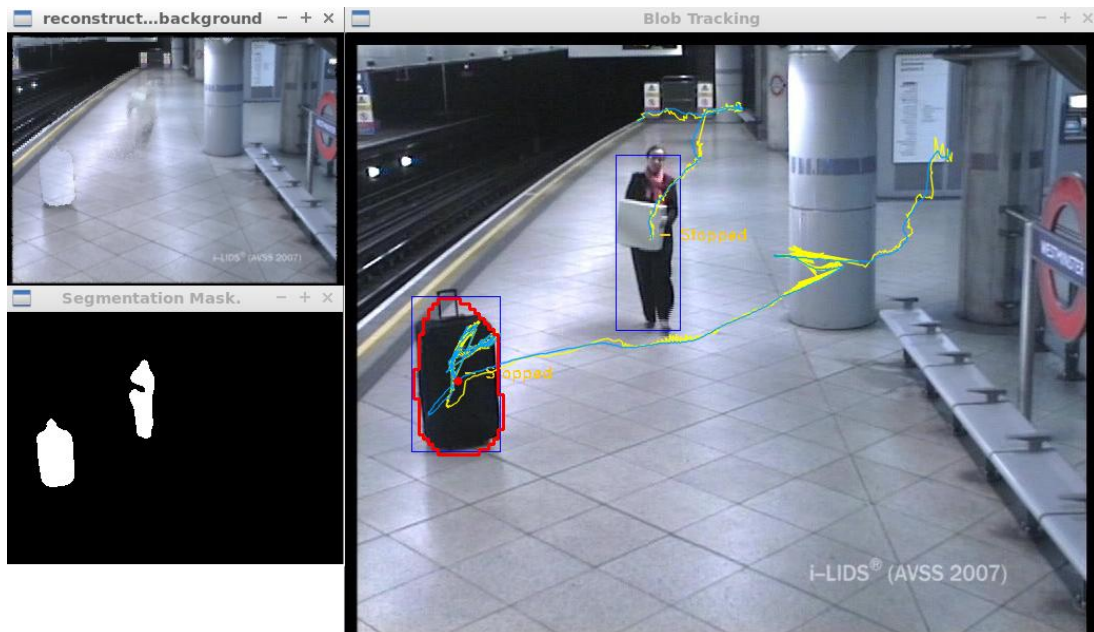


Figura 44 - Detecção de mala abandonada - *Dataset AVSS AB Easy*.

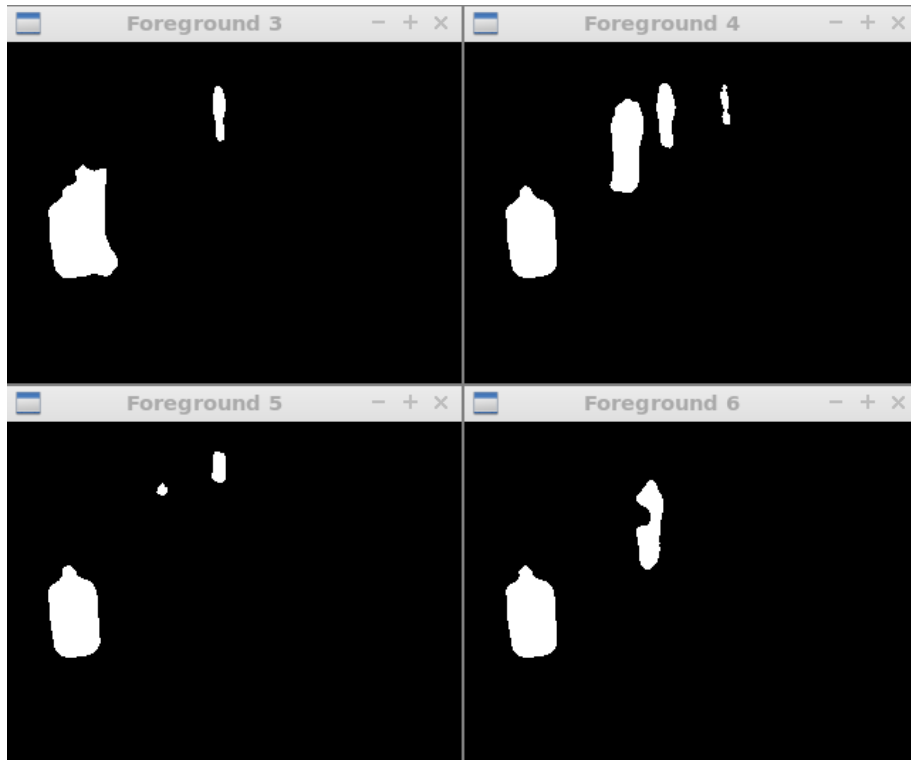


Figura 45 - Máscaras de *foreground* - *Dataset AVSS AB Easy*.

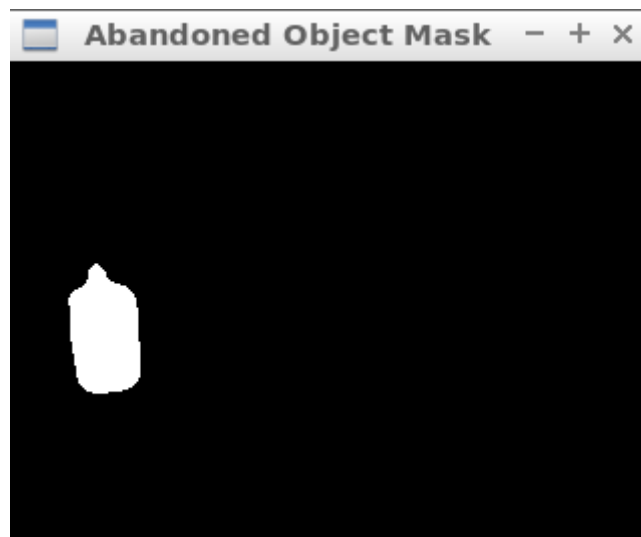


Figura 46 - Máscara de objeto abandonado - *Dataset AVSS AB Easy*.

Na Figura 47 detetamos o estacionamento prolongado de um automóvel em local proibido, com alarme assinalado a vermelho, e cuja máscara de objeto abandonado resultante pode ser vista na Figura 48.

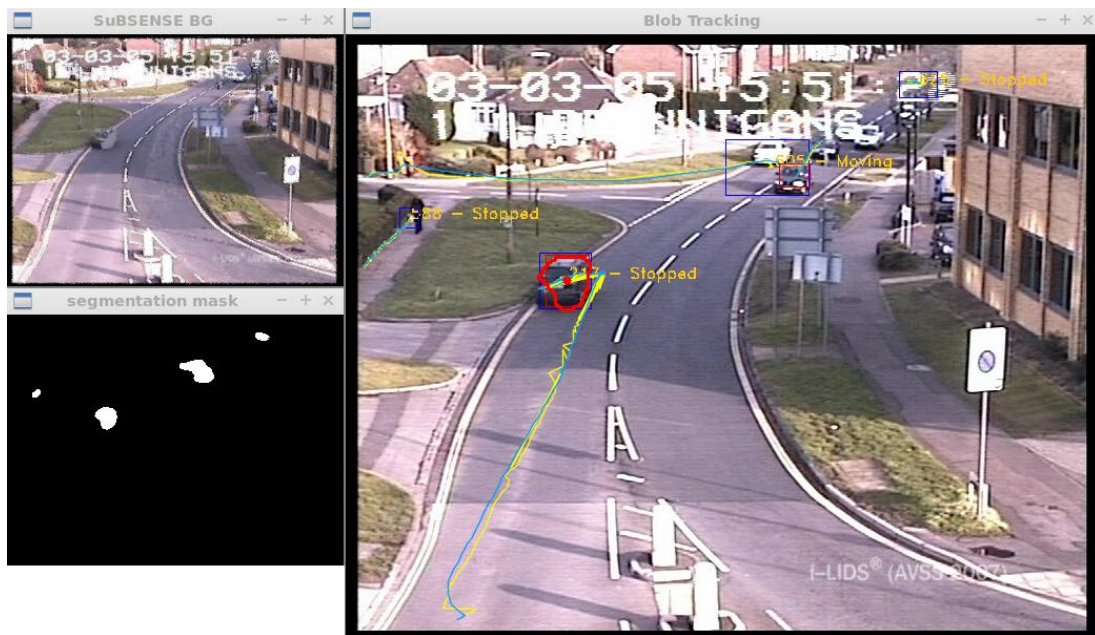


Figura 47 - Detecção de automóvel estacionado (abandonado) em local proibido - *Dataset AVSS PV Medium*.

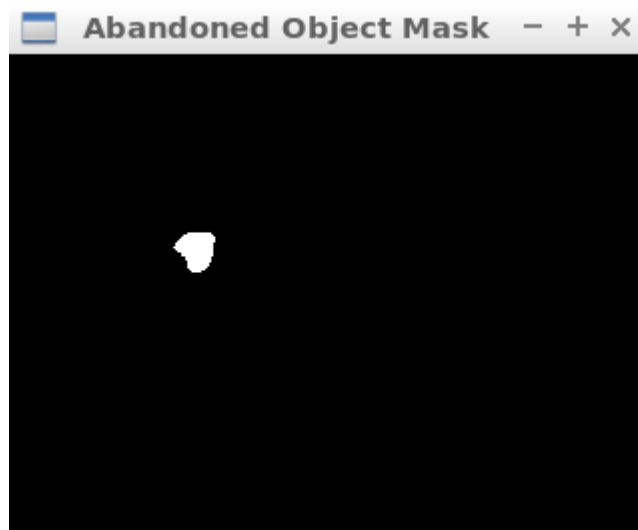


Figura 48 - Máscara de objeto abandonado da Figura 47.

Na Figura 49 uma mala eficazmente detetada como abandonada num cenário de aeroporto.

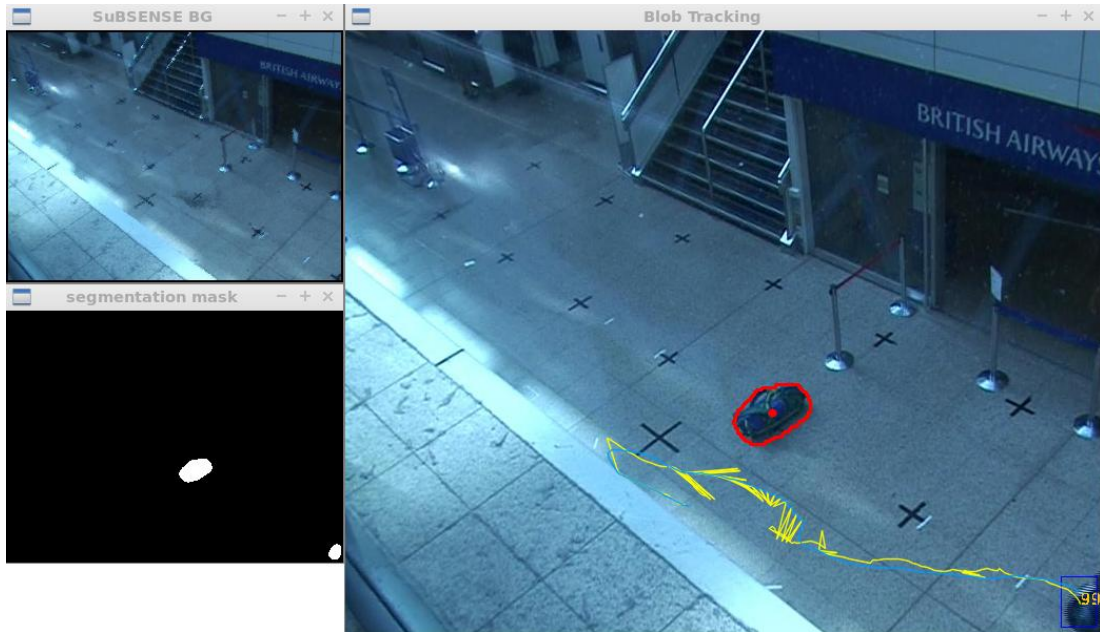


Figura 49 - Detecção de mala abandonada - *Dataset PETS 2007*.

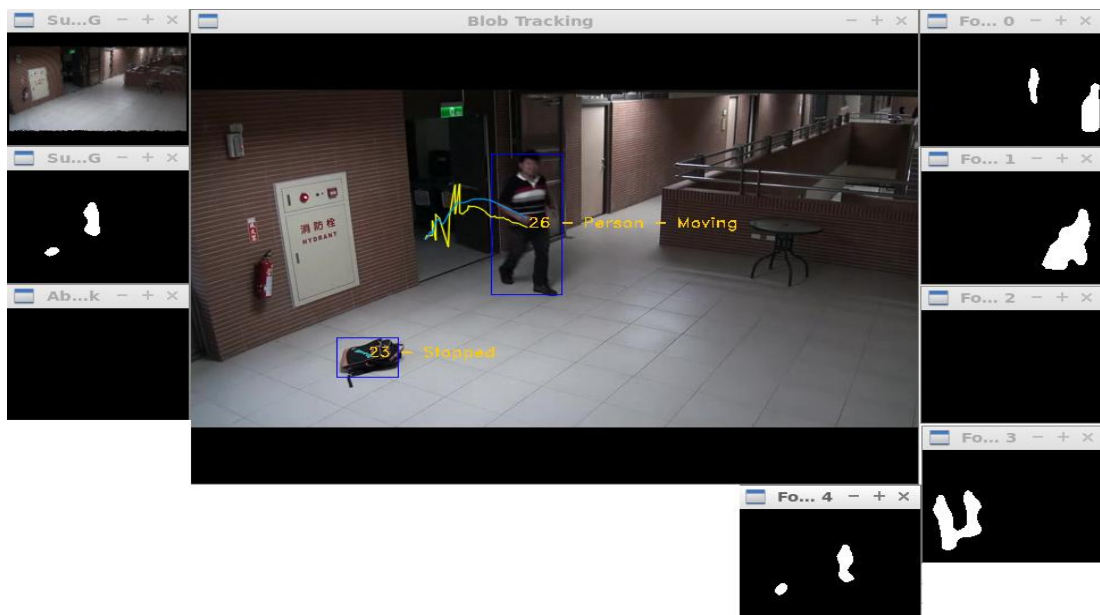


Figura 50 - Frame posterior ao abandono do objeto com ID 23 - *Dataset ABODA Video 01*.

Na parte inferior da Figura 51 podemos ver as máscaras de primeiro plano que deram origem ao alarme de objeto abandonado, e na Figura 50 o estado do sistema imediatamente após o abandono da mochila ter ocorrido.

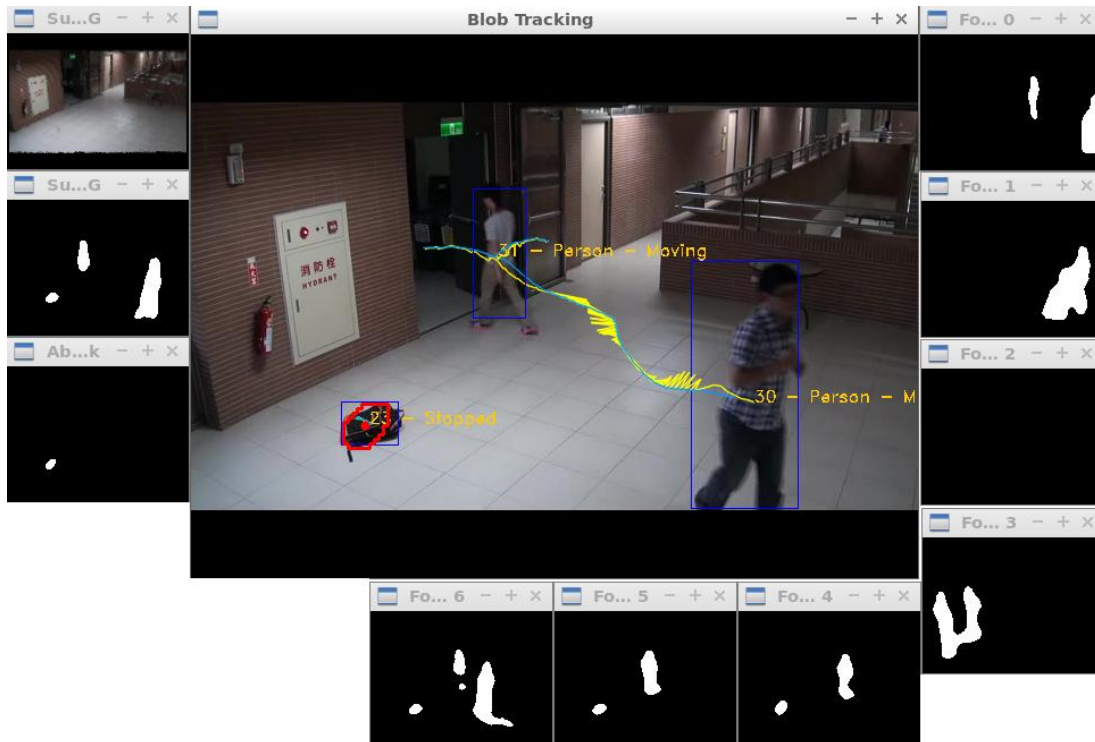


Figura 51 - Detecção de objeto abandonado - *Dataset ABODA Video 01*.

A Figura 52 e a Figura 53 exibem a detecção bem-sucedida de uma mala abandonada num passeio público, representando o estado do sistema desde que o objeto é abandonado até ao alarme ser disparado. A Figura 54 contém as máscaras com o histórico do primeiro plano, utilizadas para disparar o alarme.

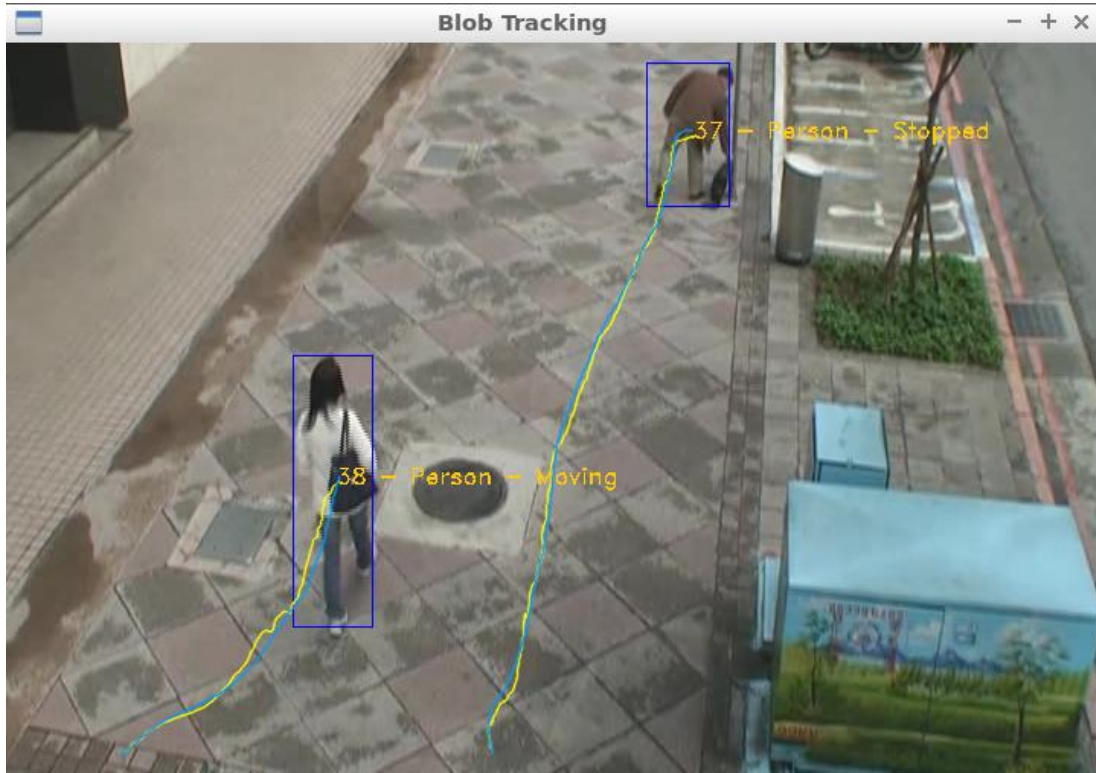


Figura 52 - *Frame em que ocorre o abandono da mala - Dataset ABODA Video 02 dataset.*

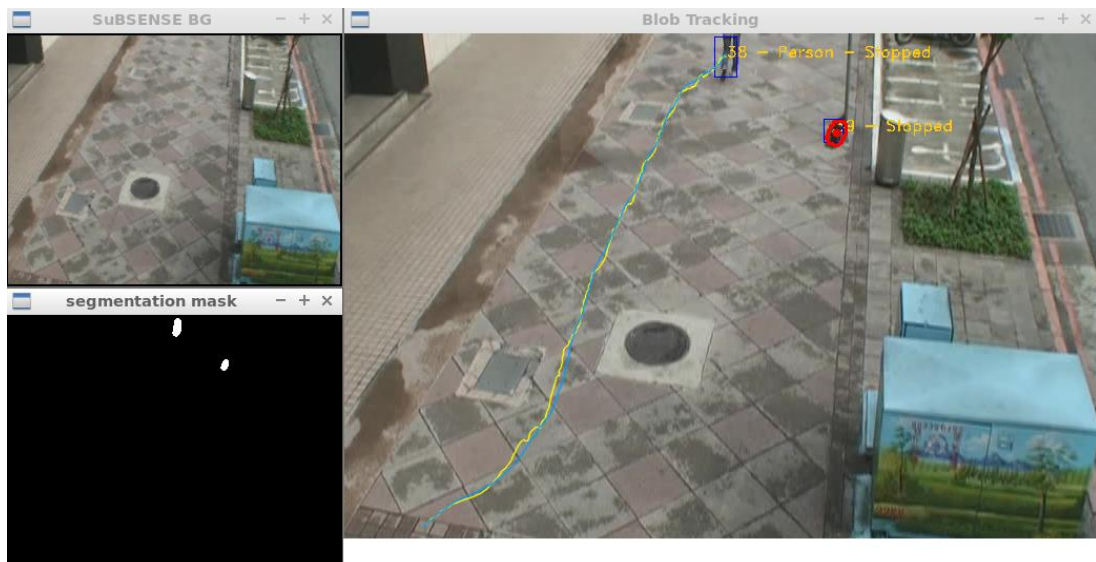


Figura 53- *Alarme de objeto abandonado - Dataset ABODA Video 02.*

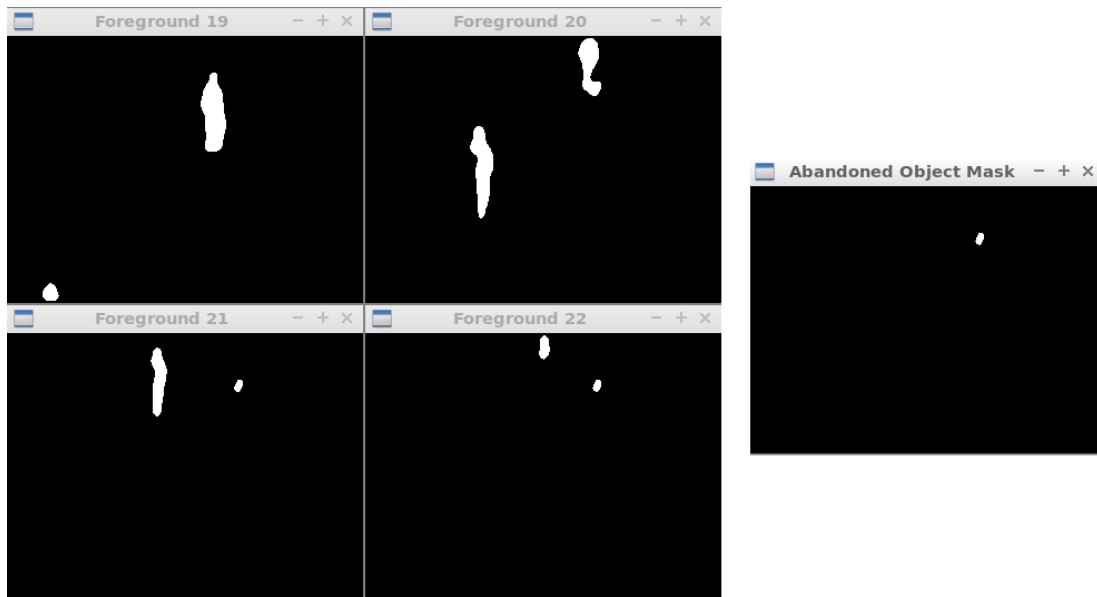


Figura 54 - Máscaras de *foreground* e máscara de objetos abandonados que originaram o alarme da Figura 53.

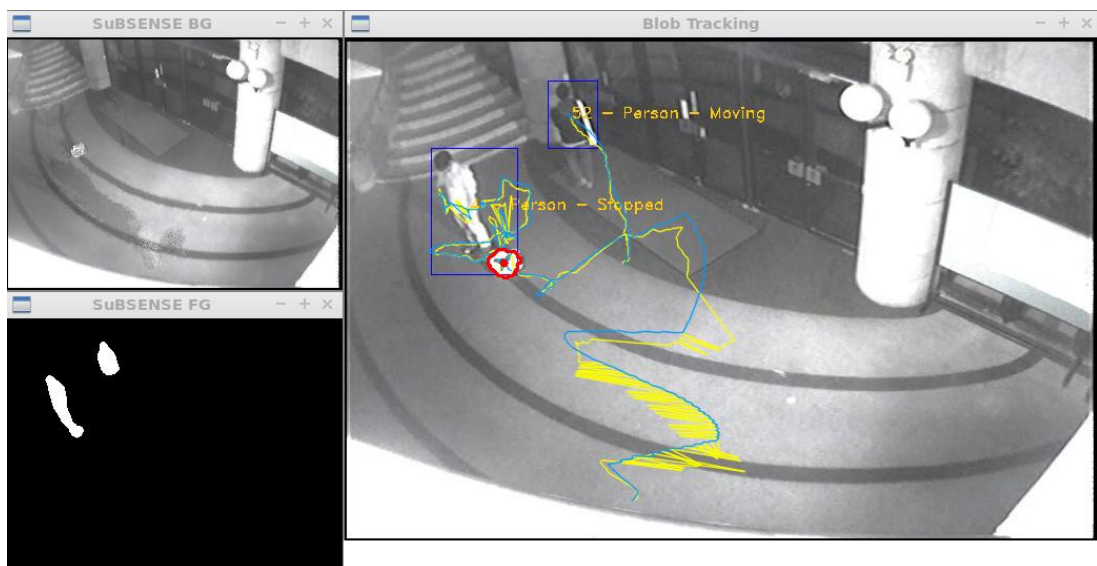


Figura 55 - Detecção de objeto abandonado em imagens em tons de cinzento - *Dataset ABODA Video 05*.

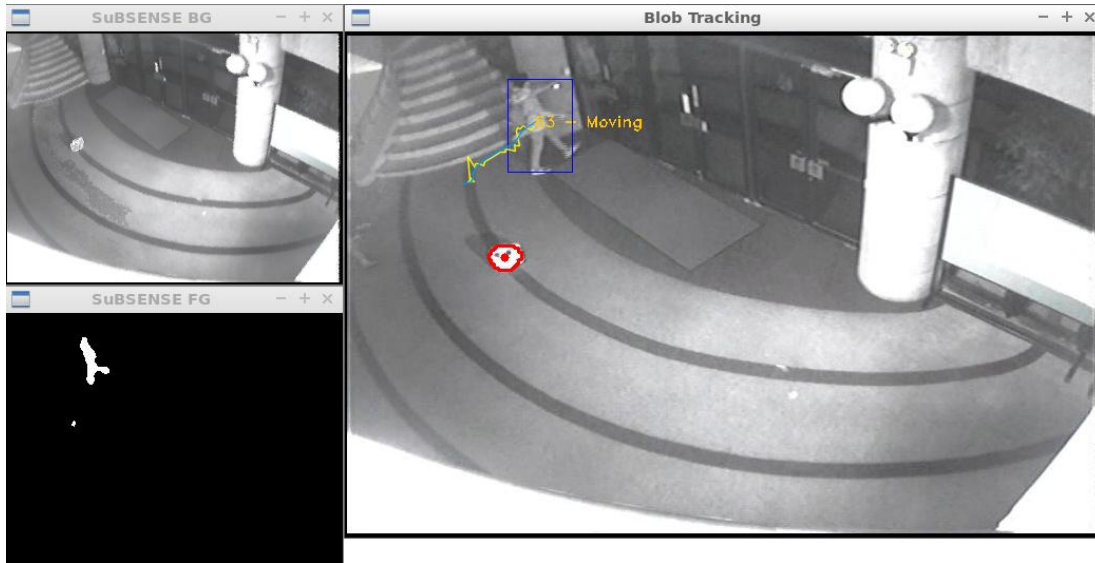


Figura 56 - Detecção de objeto abandonado em imagens em tons de cinzento - *Dataset ABODA Video 05*.

A Figura 55 e a Figura 56 apresentam a detecção eficaz de um objeto abandonado no hall de entrada de um hotel, com vídeo original na escala de cinza.

Na Figura 57 e na Figura 59 podemos observar imagens que foram passadas ao nosso módulo que tenta inferir a classe do objeto com recurso a uma rede neuronal previamente treinada. A Figura 58 e a Figura 60 exibem respetivamente os resultados da classificação, contendo na coluna de texto à esquerda o grau de confiança normalizado, e na coluna à direita o nome (ou conjunto de sinónimos) que designam a classe.



Figura 57 - Exemplo de imagem passada à *framework* Caffe

```
nirvana@sYs ~/workspace/caffe_classification/Release $ ./caffe_classification trolley_sample.jpg
----- Prediction for trolley_sample.jpg -----
0.299910 - "n03709823 mailbag, postbag"
0.250988 - "n03976467 Polaroid camera, Polaroid Land camera"
0.132975 - "n02769748 backpack, back pack, knapsack, packsack, rucksack, haversack"
0.073387 - "n04372370 switch, electric switch, electrical switch"
0.019845 - "n02916936 bulletproof vest"
```

Figura 58 - Predição da rede neuronal para a Figura 57



Figura 59 - Exemplo de imagem passada à *framework* Caffe

```
nirvana@sYs ~/workspace/caffe_classification/Release $ ./caffe_classification plant.jpg
----- Prediction for plant.jpg -----
0.443132 - "n03991062 pot, flowerpot"
0.186778 - "n04522168 vase"
0.078275 - "n07742313 Granny Smith"
0.066076 - "n07930864 cup"
0.048942 - "n02909870 bucket, pail"
```

Figura 60 - Predição da rede neuronal para a Figura 59

## 5.2. Falhas no método

Na Figura 61 notamos que a aprendizagem do modelo de plano de fundo ocorreu de forma demasiado rápida. Este facto levou a que o objeto abandonado deixasse de ser detetado como primeiro plano e fosse incluído no plano de fundo antes de ser disparado um alarme no intervalo de tempo definido.

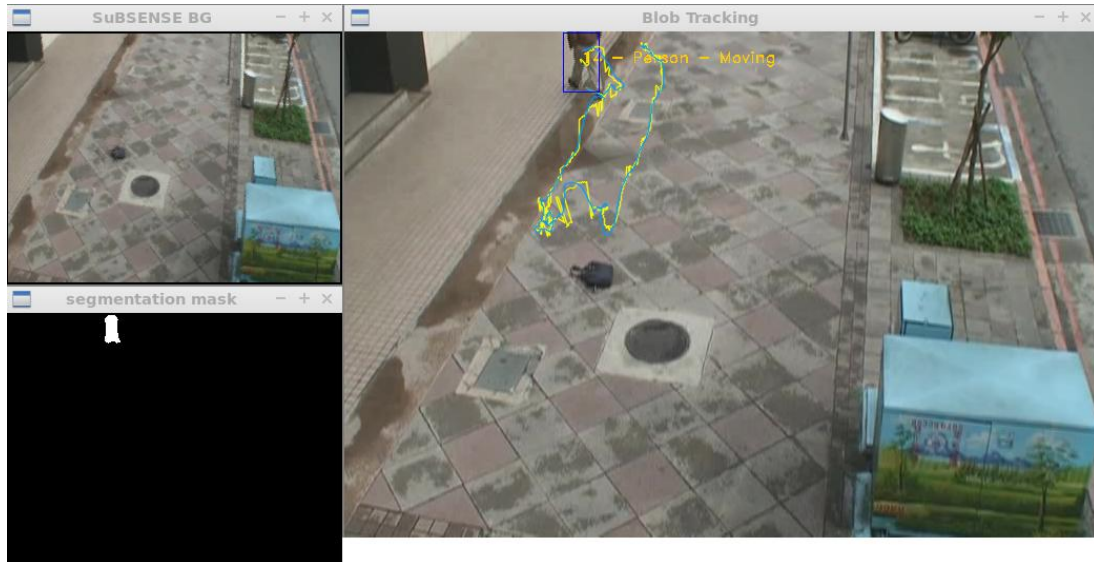


Figura 61 - Background aprende informação demasiado rápido - *Dataset ABODA Video 03*.

Podemos observar na Figura 62 que o modelo de background das frames anteriores (canto superior direito da imagem - SuBSENSE BG) já apresentava pixels escuros que revelam pouca distância RGB aos pixels da mala abandonada, sendo desta forma associados mais rapidamente ao conjunto de samples de background para aqueles pixels. De forma a obter uma *frame rate* elevada estamos a redimensionar a imagem de entrada para apenas  $160 \times 120$  pixels, facto que também contribui negativamente na extração das características LBSP visto que as arestas ficam menos intensas.

Seria possível através de alguma parametrização tornar a aprendizagem de plano de fundo mais resistente a alterações, com o inconveniente de diminuir consideravelmente a resistência do algoritmo a mudanças de iluminação, sombras, e aos movimentos repetitivos do ambiente (árvores a abanar, movimento da água, chuva, etc).

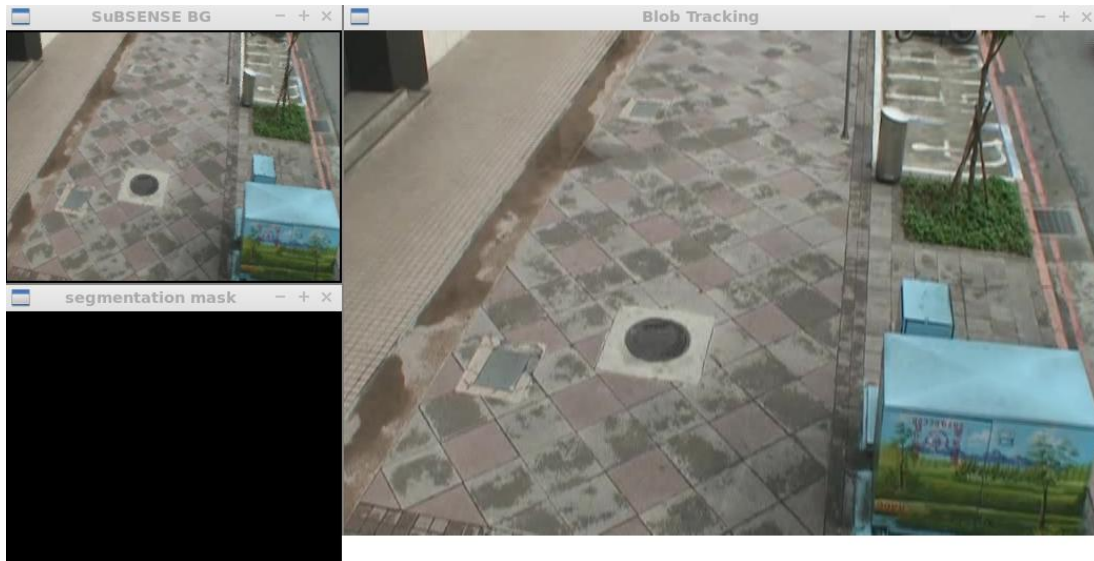


Figura 62 - Exemplo de *frame* anterior à da Figura 61 - *Dataset ABODA Video 03*.

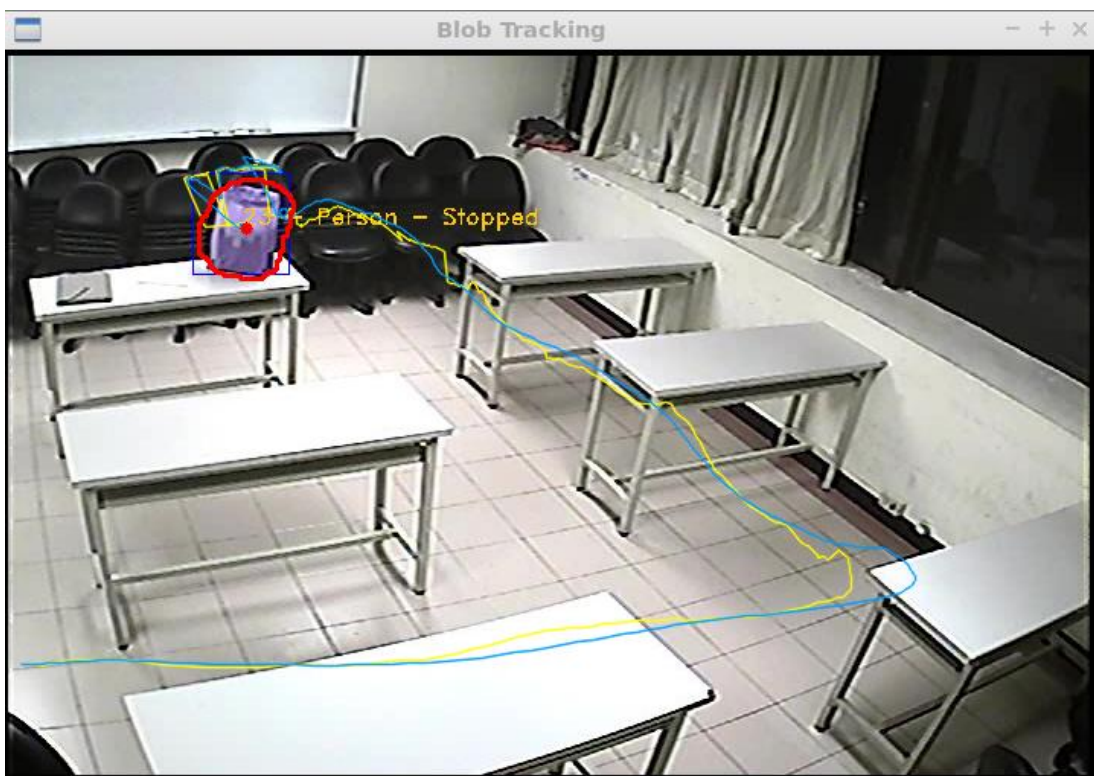


Figura 63 - Detecção de mala abandonada - *Dataset ABODA Video 07*.

No caso da Figura 63 podemos analisar um erro de associação da informação de *tracking*. A mala foi corretamente identificada como abandonada, porém a trajetória que o seu proprietário fez previamente assim como a classificação

“Pessoa” ficaram associadas ao objeto. Como podemos verificar na Figura 64 e na Figura 65 que correspondem respectivamente aos instantes anteriores e posteriores ao abandono, quando ocorreu a separação do *blob* original que agregava a mala e o seu proprietário, foi dado um novo ID ao proprietário, e o ID bem como a informação de tracking anterior permaneceu no objeto.

Para tornar o método mais robusto poderíamos construir um modelo de aparência para cada *blob*, e no caso de ocorrer uma divisão (*split*) de um *blob* dando origem a novos *blobs*, proceder a uma identificação com base nesse modelo de aparência por forma a atribuir corretamente o ID e a informação de *tracking* antiga ou iniciar um novo seguimento.

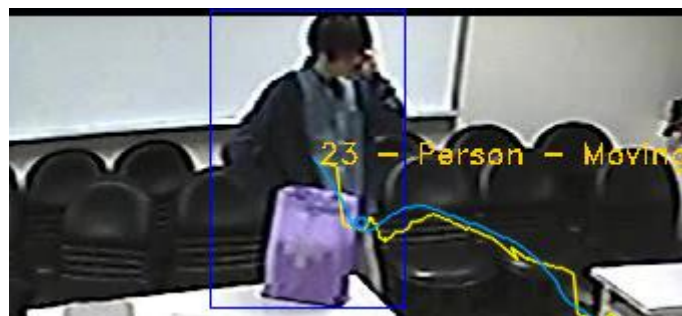


Figura 64 - Informação de *tracking* antes do abandono do objeto - *Dataset ABODA Video 07*.



Figura 65 - Informação de *tracking* após o abandono do objeto - *Dataset ABODA Video 07*.

O método de subtração de fundo utilizado é menos eficaz nas imagens em tons de cinza. Na gravação de vídeo utilizando infravermelhos, partes do objeto podem ser aprendidas para o modelo de plano de fundo a uma maior velocidade, gerando um duplo alarme para o mesmo objeto, como podemos observar na Figura 66.

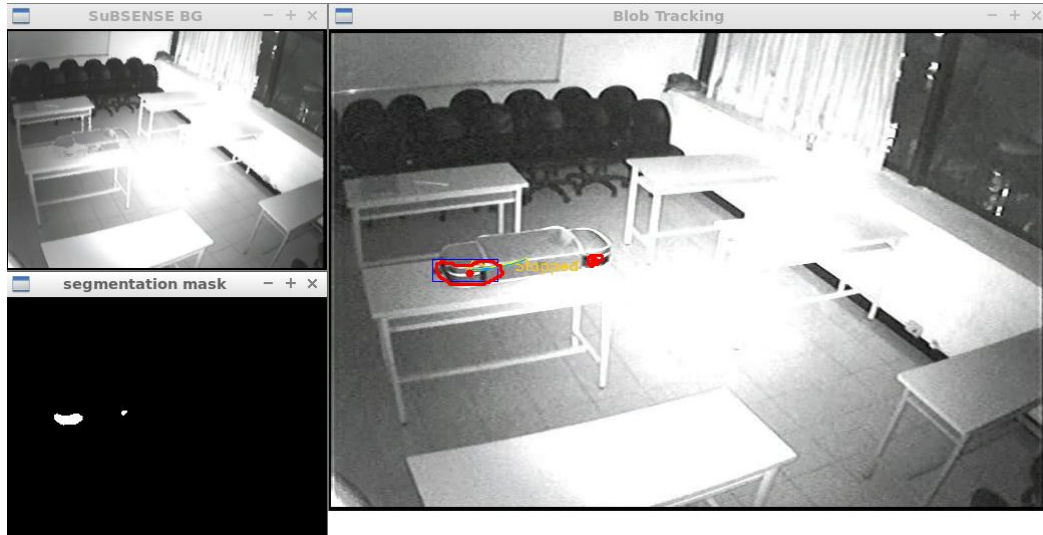


Figura 66 - Duplo alarme para o mesmo objeto - *Dataset ABODA Video 08*.

Comparativamente na Figura 67, podemos constatar que quando temos informação de cor, a passagem dos *pixels* para o *background* decorre de forma mais lenta e homogênea.

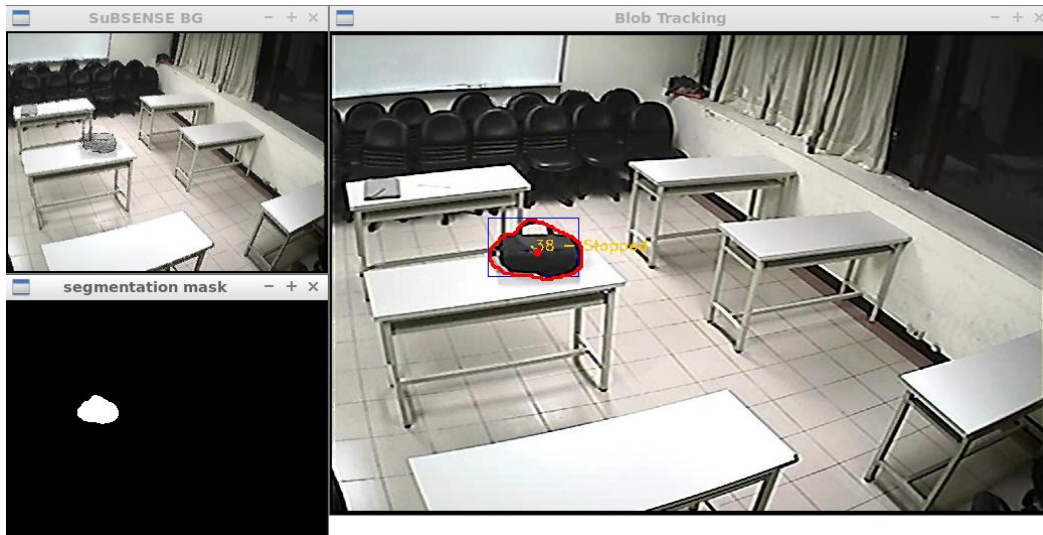


Figura 67 - Detecção eficaz com imagens RGB - *Dataset ABODA Video 07*.

### 5.3. Resultados alcançados

Tabela 3 - Resultados para os datasets i-LIDS [23] e ABODA [49].

| <b>Dataset / Sequência</b> | <b>Número de Objetos Abandonados</b> | <b>Verdadeiros Positivos</b> | <b>Falsos Positivos</b> |
|----------------------------|--------------------------------------|------------------------------|-------------------------|
| i-LIDS / AB Easy           | 1                                    | 1                            | 0                       |
| i-LIDS / AB Medium         | 1                                    | 1                            | 1                       |
| i-LIDS / AB Hard           | 4                                    | 3                            | 1                       |
| ABODA / Video 01           | 1                                    | 1                            | 0                       |
| ABODA / Video 02           | 1                                    | 1                            | 0                       |
| ABODA / Video 03           | 1                                    | 0                            | 0                       |
| ABODA / Video 04           | 1                                    | 1                            | 1                       |
| ABODA / Video 05           | 1                                    | 1                            | 0                       |
| ABODA / Video 06           | 1                                    | 1                            | 0                       |
| ABODA / Video 07           | 1                                    | 1                            | 0                       |
| ABODA / Video 08           | 1                                    | 1                            | 1                       |
| ABODA / Video 09           | 1                                    | 1                            | 0                       |
| ABODA / Video 10           | 1                                    | 1                            | 1                       |
| ABODA / Video 11           | 1                                    | 1                            | 2                       |

## 6. Conclusões e Trabalho Futuro

### 6.1. Conclusões

Após uma análise às técnicas existentes de visão por computador para detecção automática de objetos abandonados, foi desenvolvida com sucesso uma ferramenta eficaz que minimiza ou supera parte das limitações encontradas em métodos anteriores.

Destacamos a inclusão do algoritmo SuBSENSE [36] que reflete atualmente o estado-da-arte em algoritmos de segmentação do plano de fundo. Sendo das primeiras etapas neste tipo de ferramentas, um bom desempenho deste processo de segmentação é fulcral para garantir bons resultados pois evita a propagação de erros para as componentes de seguimento, de classificação e de detecção de objetos abandonados.

O acréscimo do módulo que permite a classificação do tipo de objeto abandonado com recurso a uma rede neuronal, pode dar um contributo importante e flexibilizar a utilização deste tipo de ferramentas em diferentes contextos do mundo real.

### 6.2. Trabalho futuro

No decorrer do trabalho foram assinalados alguns pontos em que o nosso método pode ser melhorado:

- Utilizar um modelo de aparência dos objetos mais robusto (conjunto de características) na detecção de objetos abandonados. Foram feitos alguns testes rápidos com detetores e descritores de características (por exemplo SIFT [44] e SURF [50]), porém como as imagens dos objetos tinham pouca resolução muitas das vezes não era possível os detetores encontrarem características interessantes para os discriminar.

- Criar um modelo de aparência dos objetos para o *tracking*, de forma a que quando ocorre a divisão de um *blob* (com surgimento de pelo menos um novo *blob*), possamos associar a informação de *tracking* antiga ao *blob* correto.
- Na fase de divisão de *blobs* é também possível criarmos uma associação entre objetos, para que possamos associar um objeto mais tarde classificado como abandonado ao seu proprietário.
- A detecção de objetos removidos (ou roubados) pode ser feita utilizando um *historial* do *background* ou analisando a alteração nas máscaras de *feedback* do algoritmo SuBSENSE [36].

---

## Obras Citadas

- [1] G. G. M. M. e. C. R. S. Ferrando, "A New Method for Real Time Abandoned Object Detection and Owner Tracking," *2006 IEEE International Conference on Image Processing*, pp. 3329-3332, 8-11 Oct. 2006.
- [2] S. A. N. C. R. M. O. M. e. N. P. N. Bird, "Real time, online detection of abandoned objects in public areas," *Proceedings 2006 IEEE International Conference on Robotics and Automation*, pp. 3775-3780, 15-19 May 2006.
- [3] J.-Y. C. e. L.-G. C. H.-H. Liao, "A Localized Approach to Abandoned Luggage Detection with Foreground-Mask Sampling," *IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance*, pp. 132-139, 1-3 Sept. 2008.
- [4] C.-Y. Lin e W.-H. Wang, "An Abandoned Objects Management System Based on the Gaussian Mixture Model," *International Conference on Convergence and Hybrid Information Technology*, pp. 169-175, Aug. 2008.
- [5] C. S. a. E. L. Grimson, "Adaptive background mixture models for real-time tracking," *Proc. CVPR99*, pp. II-2246-2252, Jun 1999.
- [6] A. F. Otoom, H. Gunes e M. Piccardi, "Automatic Classification of Abandoned Objects for Surveillance of Public Premises," *CISP 08 Congress on Image and Signal Processing*, vol. 4, pp. 542-549, 27-30 May 2008.
- [7] Q. Li, Y. Mao, Z. Wang e W. Xiang, "Robust Real-Time Detection of Abandoned and Removed Objects," *ICIG '09 Fifth International Conference on Image and Graphics*, pp. 156-161, 20-23 Sept. 2009.
- [8] W. Wang e Z. Liu, "A New Approach for Real-Time Detection of Abandoned and Stolen Objects," *2010 International Conference on Electrical and Control Engineering (ICECE)*, pp. 128-131, 25-27 June 2010.
- [9] J. Raheja, C. Malireddy, A. Singh e L. Solanki, "Detection of abandoned objects in real time," *2011 3rd International Conference on Electronics Computer Technology (ICECT)*, vol. 2, pp. 199-203, 8-10 April 2011.
- [10] J. Pan, Q. Fan e S. Pankanti, "Robust abandoned object detection using region-level analysis," *2011 18th IEEE International Conference on Image Processing (ICIP)*, pp. 3597-3600, 11-14 Sept. 2011.

- [11] L. Xiya, W. Jingling e Z. Qin, "An Abandoned Object Detection System Based on Dual Background and Motion Analysis," *2012 International Conference on Computer Science & Service System (CSSS)*, pp. 2293-2296, 11-13 Aug. 2012.
- [12] K. Muchtar, C.-Y. Lin, L.-W. Kang e C.-H. Yeh, "Abandoned object detection in complicated environments," *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1-6, 2013.
- [13] X. Li, C. Zhang e D. Zhang, "Abandoned Objects Detection Using Double Illumination Invariant Foreground Masks," *2010 20th International Conference in Pattern Recognition (ICPR)*, pp. 436-439, 23-26 Aug. 2010.
- [14] N. D. a. B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 886-893, 2005.
- [15] M. Elhamod e M. Levine, "Automated Real-Time Detection of Potentially Suspicious Behavior in Public Transport Areas," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, pp. 688-699, June 2013.
- [16] Q. Fan, P. Gabbur e S. Pankanti, "Relative Attributes for Large-Scale Abandoned Object Detection," *2013 IEEE International Conference in Computer Vision (ICCV)*, pp. 2736-2743, 1-8 Dec. 2013.
- [17] L. Chang, H. Zhao, S. Zhai, Y. Ma e H. Liu, "Robust abandoned object detection and analysis based on online learning," *2013 IEEE International Conference in Robotics and Biomimetics (ROBIO)*, pp. 940-945, 12-14 Dec. 2013.
- [18] L. Maddalena e A. Petrosino, "Stopped Object Detection by Learning Foreground Model in Videos," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, pp. 723-735, May 2013.
- [19] L. Sole, A. Sonawane, S. Shinde e V. Mane, "Video analytics for abandoned object detection and its evaluation on atom and ARM processor," *2013 IEEE International Conference in Computational Intelligence and Computing Research (ICCIC)*, pp. 1-6, 26-28 Dec. 2013.
- [20] Wahyono, A. Filonenko e K.-H. Jo, "Detecting abandoned objects in crowded scenes of surveillance videos using adaptive dual background model," *2015 8th International Conference in Human System Interactions (HSI)*, pp. 224-227, 25-27 June 2015.

- 
- [21] K. Ingersoll, P. Niedfeldt e R. Beard, "Multiple target tracking and stationary object detection in video with Recursive-RANSAC and tracker-sensor feedback," *2015 International Conference in Unmanned Aircraft Systems (ICUAS)*, pp. 1320-1329, 9-12 June 2015.
- [22] R. S. F. H. L. A. H. a. M. T. S. Y. Tian, "Robust Detection of Abandoned and Removed Objects in Complex Surveillance Videos," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, pp. 565-576, Sept. 2011.
- [23] AVSS, "i-Lids bag and vehicle detection challenge," *IEEE International Conference on Advanced Video and Signal based Surveillance*, September 2007.
- [24] "PETS 2006," *Ninth IEEE International Workshop Performance Evaluation of Tracking and Surveillance*, 2006.
- [25] R. S. F. a. A. H. Y. Tian, "Real-time detection of abandoned and removed objects in complex environments," *Proc. IEEE Int Workshop Visual Surveillance (in conjunction with ECCV'08), Marseille, France*, 2008.
- [26] J. W. A. K. a. W. Y. H. Eng, "Novel Region-based modeling for human detection within high dynamic aquatic environment," *Proc. CVPR*, pp. II-390-397, 2004.
- [27] M. Harville, "A framework for high-level feedback to adaptive, per-pixel, mixture-of-Gaussian background models," *Proc. ECCV*, p. 37-49, 2002.
- [28] P. K. a. R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," *Proc. 2nd Eur. Workshop Adv. Video Based Surveillance Systems*, p. 149-158, 2001.
- [29] M. L. a. A. H. Y. Tian, "Robust and efficient foreground analysis for real-time video surveillance," *IEEE CVPR*, Jun 2005.
- [30] R. M. X. G. P. L. C. P. W. Y. a. A. E. T. Boulton, "Frame-rate omnidirectional surveillance and tracking of camouflaged and occluded targets," *Proc. IEEE Int. Workshop Visual Surveillance*, p. 48-55, 1999.
- [31] T. B. a. R. J. T. Zhang, "Two thresholds are better than one," *Visual Surveillance Workshop, Minneapolis*, 2007.
- [32] R. F. Y. Z. L. B. a. A. H. L. Chen, "An integrated system for moving object classification in surveillance videos," *IEEE Int. Conf. Adv. Video Signal-Based*

*Surveillance*, p. 52-59, 2008.

- [33] Z. Z. W. Y. a. A. J. L. P. L. Venetianer, "Stationary target detection using the objectvideo surveillance system," *IEEE Int. Conf. Adv. Video Signal-Based Surveillance, London, U.K.*, Sep. 2007.
- [34] G. S. V. C. a. C. B. M. Bertalmío, "Image inpainting," *SIGGRAPH 2000, New Orleans, LA*, Jul 2000.
- [35] D. Ramanan, "Using Segmentation to Verify Object Hypotheses," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2007.
- [36] P.-L. St-Charles, G.-A. Bilodeau e R. Bergevin, "Flexible Background Subtraction with Self-Balanced Local Sensitivity," *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 414, 419, 23-28 June 2014.
- [37] P.-M. J. F. P. J. K. P. I. N. Goyette, "changedetection.net: A New Change Detection Benchmark Dataset," *IEEE Change Detection Workshop CVPR*, 2012.
- [38] P.-M. J. F. P. J. K. Y. B. a. P. I. Y. Wang, "CDnet 2014: An Expanded Change Detection Benchmark Dataset," *Proc. IEEE Workshop on Change Detection (CDW-2014) at CVPR-2014*, pp. 387-394, 2014.
- [39] S. e. al., "Cast shadow segmentation using invariant color features," *CVIU*, 2004.
- [40] H. e. al., "Shadow elimination for effective moving object detection by Gaussian shadow modeling," *IVC*, 2003.
- [41] S. e. al., "Improved shadow removal for robust person tracking in surveillance scenarios," *ICPR*, 2010.
- [42] H. & Chen, "Moving cast shadow detection using physics-based features," *CVPR*, 2009.
- [43] L. & Distanto, "Shadow detection for moving objects based on texture analysis," *PR*, 2007.
- [44] D. G. Lowe, "Object recognition from local scale-invariant features," *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, pp. 1150-1157, 1999.
- [45] Y. a. S. E. a. D. J. a. K. S. a. L. J. a. G. R. a. G. S. a. D. T. Jia, "Caffe:

- 
- Convolutional Architecture for Fast Feature Embedding,” 2014.
- [46] O. D. J. S. H. K. J. S. S. M. S. H. Z. K. A. K. A. B. M. B. A. F.-F. L. Russakovsky, “ImageNet Large Scale Visual Recognition Challenge,” *ILSVRC*, 2015.
- [47] C. Fellbaum, “WordNet: An Electronic Lexical Database,” *Bradford Books*, 1998.
- [48] “PETS 2007,” *Tenth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2007.
- [49] S.-C. C. C.-S. C. D.-T. L. a. Y.-P. H. K. Lin, “(ABODA) ABandoned Objects DATaset,” *IEEE Transactions on Information Forensic and Security (TIFS)*, 2015.
- [50] T. T. L. V. G. Herbert Bay, “SURF: Speeded Up Robust Features,” *Katholieke Universiteit Leuven*, 2006.
- [51] M. Elhamod e M. Levine, “Real-Time Semantics-Based Detection of Suspicious Activities in Public Spaces,” *2012 Ninth Conference on Computer and Robot Vision (CRV)*, pp. 268-275, 28-30 May 2012.
- [52] R. Tripathi, A. Jalal e C. Bhatnagar, “A framework for abandoned object detection from video surveillance,” *2013 Fourth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*, pp. 1-4, 18-21 Dec. 2013.
- [53] K. Lin, S.-C. Chen, C.-S. Chen, D.-T. Lin e Y.-P. Hung, “Left-Luggage Detection from Finite-State-Machine Analysis in Static-Camera Videos,” *2014 22nd International Conference in Pattern Recognition (ICPR)*, pp. 4600-4605, 24-28 Aug. 2014.
- [54] B. Hu, Y. Li, Z. Chen, G. Xiong e F. Zhu, “Research on abandoned and removed objects detection based on embedded system,” *2014 IEEE 17th International Conference in Intelligent Transportation Systems (ITSC)*, pp. 2968-2971, 8-11 Oct. 2014.
- [55] K. Lin, S.-C. Chen, C.-S. Chen, D.-T. Lin e Y.-P. Hung, “Abandoned Object Detection via Temporal Consistency Modeling and Back-Tracing Verification for Visual Surveillance,” *IEEE Transactions in Information Forensics and Security*, vol. 10, pp. 1359-1370, July 2015.
- [56] R. S. F. H. L. A. H. a. M. T. S. Y. Tian, “Robust Detection of Abandoned and Removed Objects in Complex Surveillance Videos,” *IEEE Transactions on*

*Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41,  
pp. 565-576, Sept. 2011.