

CENTRO UNIVERSITÁRIO EURÍPIDES DE MARÍLIA
FUNDAÇÃO DE ENSINO “EURÍPIDES SOARES DA ROCHA”
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**RECONHECIMENTO DE VEÍCULOS EM
MOVIMENTO A PARTIR DO
PROCESSAMENTO DE IMAGENS AÉREAS**

DANIELLE GUINDAS PERACCINI

ORIENTADOR: PROF. ME. RODOLFO BARROS CHIARAMONTE

Marília - SP
Dezembro/2017

CENTRO UNIVERSITÁRIO EURÍPIDES DE MARÍLIA
FUNDAÇÃO DE ENSINO “EURÍPIDES SOARES DA ROCHA”
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**RECONHECIMENTO DE VEÍCULOS EM
MOVIMENTO A PARTIR DO
PROCESSAMENTO DE IMAGENS AÉREAS**

DANIELLE GUINDAS PERACCINI

Monografia apresentada ao Centro Universitário Eurípides de Marília como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Me. Rodolfo Barros Chiaramonte

Marília - SP
Dezembro/2017



CENTRO UNIVERSITÁRIO EURÍPIDES DE MARÍLIA - UNIVEM
MANTIDO PELA FUNDAÇÃO DE ENSINO "EURÍPIDES SOARES DA ROCHA"

BACHARELADO EM SISTEMAS DE INFORMAÇÃO

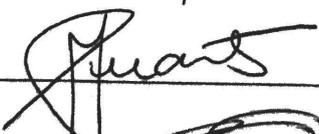
Danielle Guindas Peraccini

Reconhecimento de Veículos em Movimento a Partir do Processamento de Imagens Aéreas

Banca examinadora da monografia apresentada ao Curso de Bacharelado em Sistemas de Informação do UNIVEM/F.E.E.S.R., para obtenção do Título de Bacharel em Sistemas de Informação.

Nota: 10 (Dez)

Orientador: Rodolfo Barros Chiaramonte 

1º. Examinador: Mauricio Duarte 

2º. Examinador: Luan Cardoso dos Santos 

Marília, 28 de novembro de 2017.

*Este trabalho é dedicado aos meus pais,
pela oportunidade de estudo e amor incondicional.*

AGRADECIMENTOS

A Deus por minha vida.

A esta universidade, seu corpo docente, direção, coordenação, administração e demais funcionários, pela oportunidade e trabalho desempenhado em minha graduação.

A todos os docentes do curso de Bacharelado em Ciência da Computação pela dedicação, generosidade e conhecimento proporcionados.

A meu orientador Prof. Rodolfo Barros Chiaramonte, pelo apoio e suporte disponibilizados no desenvolvimento do presente projeto e durante toda a graduação.

A todos os professores que tive a honra de ser aluna e que me proporcionaram conhecimento e valores imensuráveis.

A meus pais pelo amor, paciência, educação, apoio e incentivo incondicional.

Aos amigos, parentes e companheiros de trabalho pelo apoio e carinho.

Em especial aos amigos de curso Frederico Soares, Marina Beretta, Matheus Ferraroni, Thais de Moura e Thiago Costa pela ajuda e companheirismo.

E todos que de forma direta ou indireta participaram e contribuíram para a minha formação, o meu muito obrigado.

*Quem nunca cometeu um erro nunca tentou nada de novo.
(Albert Einstein)*

RESUMO

Com o crescimento da população mundial e, conseqüentemente, dos veículos nas cidades, tornam-se cada vez mais pertinentes soluções que tem objetivo de ajudar na organização e controle do trânsito, pois com mais automóveis transitando, agravam-se os problemas urbano-sociais relacionados ao transporte como congestionamentos, poluição e acidentes, gerando estresse e desconforto aos motoristas. Como meio de obter informações sobre o fluxo de veículos, a fim de contribuir para um melhor planejamento urbano, o uso de técnicas de processamento de imagens e visão computacional tem sido relevante, em razão da facilidade de implementação no monitoramento das ruas. Neste contexto, o trabalho visa realizar o estudo e implementação de técnicas de processamento de imagens e visão computacional direcionadas para a identificação de veículos transitando em uma via pública, através de imagens aéreas captadas com o uso de Veículos Aéreos Não Tripulados(VANTs). O intuito é que, a partir do método proposto, informações relevantes sobre o fluxo de veículos possam ser obtidas e, assim, alimentar sistemas de simulação de trânsito para ajudar na tomada de decisão.

Palavras-chaves: Processamento de imagens, trânsito, identificação de veículos, imagens aéreas, detecção de objetos, segmentação de imagem.

ABSTRACT

With the growth of the world population and, consequently, of the vehicles in the cities, solutions that are aimed at helping in the organization and control of the traffic become more and more pertinent, because with more automobiles passing, urban problems related to the transportation are aggravated, like congestion, pollution and accidents, generating stress and discomfort for drivers. As a way to obtain information about the flow of vehicles, in order to contribute to a better urban planning, the use of image processing techniques and computational vision has been relevant, due to the ease of implementation in the monitoring of the streets. In this context, this paper consists on the study and implementation of image processing techniques in detection of vehicles on a public highway, through aerial images captured with the use of Unmanned Aerial Vehicles (UAVs). The goal is that relevant information about the flow of vehicles can be obtained from the from the proposed method, and, thus, feed traffic simulation systems to aid in decision making and improve urban mobility.

Keywords: Image processing, traffic, object detection, unmanned aerial vehicles, vehicle detection, image segmentation.

LISTA DE FIGURAS

Figura 2.1 – Representação dos principais processos utilizados na técnica de processamento digital de imagens. Imagem adaptada de (GONZALEZ; WOODS, 2010).	22
Figura 2.2 – (a) Imagem original. (b) Imagem topográfica. (c) e (d) Duas fases da inundação. (e) Novas inundações. (f) Início da união entre duas <i>watersheds</i> e a barragem construída entre elas. (g) Barreiras maiores. (h) Linhas finais de <i>watershed</i> , representando a segmentação da imagem. Imagem adaptada de (GONZALEZ; WOODS, 2010)	26
Figura 2.3 – Exemplo de funcionamento da aplicação do método de Subtração de Fundo utilizando uma imagem fixa de referência. Imagem adaptada da documentação do OpenCV.	27
Figura 2.4 – Etapas de um algoritmo de Subtração de Fundo, Pré-Processamento, Modelagem do Fundo, Detecção de Objetos e Validação. Imagem adaptada de (HIGASHINO, 2006)	28
Figura 2.5 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Erosão. Imagem adaptada da Documentação da biblioteca OpenCV.	31
Figura 2.6 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Dilatação. Imagem adaptada da Documentação da biblioteca OpenCV.	32
Figura 2.7 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Abertura. Imagem adaptada da Documentação da biblioteca OpenCV.	33
Figura 2.8 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Fechamento. Imagem adaptada da Documentação do OpenCV.	33
Figura 2.9 – VANT simples. Imagem adaptada de (MOHAMMED et al., 2014).	35
Figura 3.1 – Diagrama representando a arquitetura do sistema proposto por Maria et al. (2016). Imagem adaptada de (MARIA et al., 2016).	38
Figura 3.2 – Exemplos de modelos utilizados no treinamento da aplicação proposta por Maria et al. (2016). Imagem adaptada de (MARIA et al., 2016).	39
Figura 3.3 – Detecção de veículos na ROI definida no sistema de Maria et al. (2016). Imagem adaptada de (MARIA et al., 2016).	39

Figura 3.4 – Diagrama da arquitetura genérica de um sistema de gerenciamento do trânsito utilizado por Ribeiro e Correia (2014). Imagem adaptada de (RIBEIRO; CORREIA, 2014).	40
Figura 3.5 – Representação da VDL utilizada na proposta de Ribeiro e Correia (2014). Imagem adaptada de (RIBEIRO; CORREIA, 2014).	40
Figura 3.6 – Representação do particionamento em multicamadas pelo modelo PCM - <i>partial configuration object model</i> . Imagem adaptada de (QIU; WEN; FAN, 2017).	42
Figura 3.7 – (a) Partes semânticas (b) Configurações parciais. Imagem adaptada de (QIU; WEN; FAN, 2017).	42
Figura 4.1 – Representação dos processos envolvidos no desenvolvimento da aplicação proposta, demonstrando as etapas utilizadas para que o protótipo de sistema alcançasse o objetivo. (Autoria própria)	47
Figura 4.2 – Representação das ROIs utilizadas para a execução e validação do protótipo. a) Região inicial (ROI 1), indicada com um retângulo na cor azul. b) Região 2 (ROI 2), indicada com um retângulo na cor vermelha. c) Região 3 (ROI 3), indicada com um retângulo verde. (Autoria própria)	49
Figura 4.3 – a) Representação de um <i>frame</i> da ROI 1. b) Representação de um <i>frame</i> da ROI 2. c) Representação de um <i>frame</i> da ROI 3. (Autoria própria)	50
Figura 4.4 – a) <i>Frame</i> original do vídeo utilizado no desenvolvimento do trabalho. b) <i>Frame</i> resultante da aplicação do método de <i>Gaussian Blur</i> . (Autoria própria)	51
Figura 4.5 – Representação de um <i>frame</i> do vídeo original utilizado no desenvolvimento do protótipo de sistema. (Autoria própria)	52
Figura 4.6 – Imagem binária de representação de um <i>frame</i> com aplicação do método MOG de subtração de fundo. (Autoria própria)	52
Figura 4.7 – a) <i>Frame</i> resultante da subtração de fundo com MOG, conforme exposto na seção 4.6. b) <i>Frame</i> resultante da aplicação do método de Erosão após a subtração de fundo. (Autoria própria)	53
Figura 4.8 – a) <i>Frame</i> representando a imagem resultante da técnica de Erosão aplicada. b) <i>Frame</i> resultante da aplicação do método de Dilatação na imagem. (Autoria própria)	54
Figura 4.9 – Representação de um <i>frame</i> que demonstra a aplicação das funções cv2.findContours() , cv2.boundingRect() , cv2.rectangle() e cv2.circle() para a detecção dos veículos em movimento. (Autoria própria)	55

Figura 4.10–VDL representada em um <i>frame</i> da ROI 1, na cor vermelha. (Autoria própria)	56
Figura 5.1 – a) Representação de um possível percurso originário da ROI 1. b) Representação de outro caminho possível originário da ROI 1. (Autoria própria)	60
Figura 5.2–Representação de um <i>frame</i> da execução do algoritmo implementado na ROI 1. (Autoria própria)	61
Figura 5.3–Representação de um <i>frame</i> da execução do algoritmo implementado na ROI 2. (Autoria própria)	61
Figura 5.4–Representação de um <i>frame</i> da execução do algoritmo implementado na ROI 2. (Autoria própria)	62

LISTA DE TABELAS

Tabela 3.1 – Comparação de características entre os trabalhos correlatos.	44
Tabela 4.1 – Ferramentas utilizadas para a implementação do protótipo.	47
Tabela 4.2 – Dados sobre a captura das imagens	48
Tabela 5.1 – Validação dos resultados retornados pela aplicação nas três ROIs. .	62

LISTA DE ABREVIATURAS E SIGLAS

GPS	<i>Global Positioning System</i> - Sistema de Posicionamento Global
ICT	Informação e Comunicação Tecnológica
MOG	<i>Mixture of Gaussians</i>
OpenCV	<i>Open Source Computer Vision Library</i>
PCM	Partial Configuration Object Model - Modelo de Objeto de Configuração Parcial
PDI	Processamento Digital de Imagem
ROI	Region of Interest - Região de Interesse
SIG	Sistema de Informação Geográfica
UAV	<i>Unmanned Aerial Vehicle</i> - Veículo Aéreo não Tripulado
VANT	Veículo Aéreo Não Tripulado
VDL	<i>Virtual Detection Line</i> - Linha de Detecção Virtual

LISTA DE SÍMBOLOS

\in	Pertence
\geq	Comparação (Maior ou Igual)
\leq	Comparação (Menor ou Igual)
\neq	Diferente
$>$	Comparação (Maior)
$<$	Comparação (Menor)
$=$	Igual
\subseteq	Subconjunto
\ominus	Erosão
\oplus	Dilatação
\hat{B}	Reflexão de Conjuntos
\cap	Intersecção de Conjuntos
\emptyset	Conjunto Vazio

SUMÁRIO

1	INTRODUÇÃO	15
	Introdução	15
1.1	Problemática, Motivações e Justificativa	15
1.2	Objetivos	17
1.2.1	Objetivo Geral	17
1.2.2	Objetivos Específicos	17
1.3	Metodologia de Desenvolvimento do Trabalho	17
1.4	Organização do Trabalho	18
2	REFERENCIAL TEÓRICO	19
2.1	Processamento de Imagens	19
2.2	Segmentação	24
2.2.1	Limiarização (Thresholding)	24
2.2.2	Segmentação por Crescimento de Região	25
2.2.3	Segmentação usando Watersheds	25
2.3	Subtração de Fundo	26
2.3.1	Subtração de Fundo com Dois ou Mais Quadros Consecutivos	28
2.4	Morfologia Matemática	28
2.5	Elemento Estruturante	29
2.5.1	Erosão	31
2.5.2	Dilatação	32
2.5.3	Abertura e Fechamento	32
2.6	Fluxo Óptico	34
2.7	Imagens Aéreas com o Uso de <i>Drones</i>	35
2.8	Conclusões do Capítulo	36
3	TRABALHOS CORRELATOS	37
3.1	Processamento de imagens aéreas com VANTs e câmeras convencionais	37
3.2	Contagem Automática de Veículos Baseada em Vídeos de Segurança da Internet	39
3.3	Reconhecimento de objetos parcialmente ocultos pelo ambiente	41

3.4	Monitoramento veicular utilizando câmeras de segurança convencionais e visão computacional	43
3.5	Conclusões do Capítulo	43
	4 DESENVOLVIMENTO	46
4.1	Implementação	46
4.2	Ferramentas	47
4.3	Captura das Imagens de Entrada	48
4.4	Definição da Região de Interesse	48
4.5	Pré-Processamento das Imagens	50
4.5.1	Filtro de Gaussian	50
4.6	Subtração de Fundo	51
4.7	Processamento Pós-Subtração de Fundo	53
4.7.1	Erosão	53
4.7.2	Dilatação	54
4.8	Deteccção de Objeto em Movimento	55
4.9	Contagem dos Veículos	56
4.10	Conclusões do Capítulo	57
	5 RESULTADOS	59
5.1	Região de Interesse 1	60
5.2	Região de Interesse 2	61
5.3	Região de Interesse 3	61
5.4	Comparação dos Resultados	62
5.5	Conclusões do Capítulo	63
	6 CONCLUSÃO	64
6.1	Conclusões	64
6.2	Trabalhos Futuros	65
	REFERÊNCIAS	67
	Apêndice A CÓDIGO FONTE DO ALGORITMO	69

Capítulo 1

INTRODUÇÃO

Este capítulo irá apresentar a contextualização sobre o tema deste trabalho, conceituando o assunto abordado na proposta, o problema de pesquisa, as motivações, a justificativa, os objetivos e a metodologia aplicada no desenvolvimento do projeto.

1.1 Problemática, Motivações e Justificativa

O trânsito se tornou motivo de muito desconforto e estresse para a população nos últimos anos, com o não planejamento urbano e crescimento desordenado das cidades, o volume de automóveis em circulação aumenta constantemente e agrava problemas urbano-sociais relacionados ao transporte, como congestionamentos, poluição e acidentes. Segundo Mohammed et al. (2014), a população mundial tende a dobrar até o ano de 2050, esse dado desperta preocupação para as organizações governamentais que precisam investir na elaboração de eficientes planos urbanos, para disponibilizar estruturas e serviços que possam suportar grandes quantidades de habitantes dentro dos municípios.

Neste contexto, tornam-se cada vez mais válidos os recursos que tem como objetivo realizar o monitoramento do fluxo de veículos, a fim de extrair dados relevantes para a elaboração de soluções que visam auxiliar no controle do tráfego. O uso de técnicas de processamento de imagens e visão computacional tem sido bastante explorado para a aquisição dessas informações, principalmente nas áreas relacionadas ao desenvolvimento de *Smart Citys*, em razão da facilidade de implantação e custo benefício dessas propostas em ambientes reais, conforme Mohammed et al. (2014).

As tecnologias que tem como finalidade a administração da movimentação automotiva urbana são, em sua maioria, fundamentadas na utilização de sensores e passes eletrônicos, porém, a facilidade na implantação do monitoramento por imagens, torna a técnica de visão computacional cada vez mais considerada e usada no

gerenciamento do tráfego, além de permitir o aprimoramento de imagens digitais e a análise automática de elementos extraídos de uma cena.

A proposta deste projeto destina-se ao estudo e implementação de técnicas de **processamento de imagens direcionadas para a identificação de veículos** transitando em uma determinada via pública, como um meio de obter informações relevantes sobre o fluxo de automóveis.

Na elaboração de uma ferramenta experimental que ajude na criação de propostas para amenizar os problemas de deslocamento urbano contextualizados, este trabalho pretende utilizar diversos **métodos de computação visual** e a combinação de aspectos de **computação gráfica, processamento de imagem, algoritmos de segmentação e subtração de fundo**. As técnicas consistem em três etapas principais, sendo estas: Aquisição, Processamento e Reconhecimento das imagens.

De acordo com Gonzalez e Woods (2010), o **Processamento Digital de Imagens (PDI)** propõe a modificação de uma imagem por meio do computador, onde a entrada e a saída são imagens, porém, a aplicação do processamento digital possibilita que a qualidade de uma imagem seja aprimorada para uma melhor visão humana ou ainda expor características que não são visíveis na sua forma original.

O fator climático é um elemento de considerável importância na captação de imagens, segundo Qiu, Wen e Fan (2017), pois as cenas podem ser afetadas pela irregularidade na textura e iluminação que nuvens, chuva, sol, vento e outros fatores podem causar. Em consequência, os dados obtidos podem influenciar e prejudicar os resultados do processamento, tornando a análise das informações não tão precisa. No entanto, o uso de imagens como método de obtenção de informações mostra-se cada vez mais eficaz nos diversos domínios de aplicação e, segundo Mohammed et al. (2014), tem sido muito utilizada nas áreas de desenvolvimento de *Smart Citys*.

Pretende-se utilizar um **Veículo Aéreo não Tripulado (VANT)** para a aquisição de cenas aéreas e, assim, aplicar as técnicas de processamento de imagens direcionadas ao reconhecimento de veículos em movimento. Posteriormente, os dados obtidos na interpretação das imagens podem ser integrados a sistemas de simulação com finalidade de estudar propostas de melhorias no trânsito, visando aprimorar a mobilidade urbana, melhorar o desempenho do tráfego de automóveis e cooperando em fatores como a economia de combustível (e emissões de gases poluentes), redução de riscos de acidentes, colisões e confusões de trânsito, melhoria na circulação interna e nos acessos externos, entre outros.

1.2 Objetivos

Esta seção tem finalidade de apresentar o objetivo principal proposto neste trabalho, descrito no tópico 1.2.1 e os objetivos específicos definidos para o desenvolvimento deste projeto, expostos na seção 1.2.2 .

1.2.1 Objetivo Geral

Propõe-se como objetivo geral do projeto realizar o estudo de técnicas de **Processamento Digital de Imagens (PDI)** e de **Visão Computacional**, focadas na detecção de objetos em movimento, e a implementação desses métodos em uma ferramenta experimental para realizar a identificação de veículos em movimento numa via pública através de imagens aéreas, utilizando as técnicas voltadas para a análise de dados multidimensionais. Pretende-se utilizar os métodos de PDI para extrair das cenas informações relevantes sobre o fluxo de veículos em um determinado canal monitorado e efetuar a quantificação dos automóveis detectados.

1.2.2 Objetivos Específicos

Como objetivos específicos para este trabalho, foram definidas as metas descritas nos itens a seguir:

- Identificar e destacar as dificuldades, melhorias e tendências identificadas durante o trabalho a respeito do PDI focado no reconhecimento de veículos em deslocamento em imagens aéreas;
- Testar e validar os resultados obtidos pela ferramenta experimental proposta neste trabalho, de acordo com os valores reais.

1.3 Metodologia de Desenvolvimento do Trabalho

Como metodologia de trabalho para o desenvolvimento desta proposta, foram definidas 4 etapas, sendo essas as seguintes: **Estudo Bibliográfico**, **Estudo das Ferramentas**, **Implementação**, e **Validação**.

1. **Estudo Bibliográfico:** Consiste em realizar o levantamento de informações acerca do tema desta proposta, pesquisas em trabalhos correlatos e um estudo aprofundado das principais ferramentas e dispositivos utilizados na obtenção e manipulação de imagens aéreas para detecção de objetos em movimento.
2. **Estudo das ferramentas:** Se refere ao aprendizado em bibliotecas, componentes e ferramentas mais adequados para o domínio da aplicação.

3. **Implementação:** Representa a implementação do protótipo de sistema, assim como a definição da melhor opção de linguagem de programação para ser utilizada em conjunto com a biblioteca OpenCV, visando obter as melhores técnicas de desenvolvimento na área de visão computacional.
4. **Validação:** Consiste em realizar testes em ambientes reais, utilizando a ferramenta experimental desenvolvida e verificar a integridade dos dados retornados pela aplicação, em comparação com os dados verídicos.

1.4 Organização do Trabalho

Este trabalho está dividido em seis capítulos, incluindo o presente capítulo de introdução, onde é apresentada uma contextualização ao assunto abordado no trabalho, o tema definido, os principais fatores a serem trabalhados para se chegar ao objetivo final e a estrutura do documento.

O Capítulo 2 apresenta a fundamentação teórica do trabalho, a fim de conceituar os recursos e métodos importantes para a identificação de objetos em movimento a partir de imagens.

No Capítulo 3 são apresentados os trabalhos correlatos, utilizados como referência para a elaboração deste projeto, onde foram analisadas as técnicas e ferramentas utilizadas em cada referência.

No Capítulo 4 serão descritos os processos realizados no desenvolvimento da aplicação proposta neste trabalho, para o reconhecimento de veículos em uma via pública.

O Capítulo 5 demonstra a aplicação do protótipo desenvolvido, assim como os resultados obtidos com a execução da ferramenta e a validação dos valores resultantes da aplicação, avaliando a integridade das informações em comparação aos valores reais.

Encerrando, o Capítulo 6 apresenta as conclusões da pesquisa, destacando as dificuldades, tendências e melhorias encontradas durante o desenvolvimento desta proposta. Também expõe ideias a respeito de trabalhos futuros, conforme os resultados obtidos.

O Apêndice A traz o código fonte da implementação do algoritmo de detecção de veículos em movimento com imagens aéreas.

Capítulo 2

REFERENCIAL TEÓRICO

Este capítulo apresenta uma introdução à área de processamento de imagens e vídeos e expõe noções teóricas sobre os recursos, técnicas e ferramentas estudados para o desenvolvimento deste projeto, a fim de gerar uma base de conhecimentos acerca da proposta do trabalho e promover um melhor entendimento sobre do assunto.

Serão abordados filtros e operações necessárias para o tratamento de imagens que serão utilizadas em sistemas que tem objetivo de efetuar a identificação de objetos em movimento.

De acordo com Filho e Neto (1999), o PDI pode ser aplicado em duas principais categorias, sendo a primeira direcionada para o aprimoramento de informações pictóricas para interpretação humana e a segunda para a análise automática por computador de informações extraídas de uma cena. O presente trabalho se adéqua a segunda categoria, pois tem intenção de utilizar técnicas de PDI para a obtenção de informações relevantes sobre uma cena.

2.1 Processamento de Imagens

O PDI pode ser definido como o tratamento de imagens e vídeos utilizando algoritmos computacionais com finalidade de transformar os dados de entrada em cenas resultantes para melhor interpretação, tornando possível o entendimento humana de informações contidas na figura e características não visíveis na imagem de origem.

De acordo com Silva (2014), o processamento de imagens possui ferramentas robustas que possibilitam o desenvolvimento de aplicações que ajudam a solucionar problemas do mundo real.

Segundo Filho e Neto (1999), a exploração de dados através de imagens surgiu com o início do programa espacial norte-americano, com a necessidade de aprimorar a resolução os sinais captados por sondas espaciais e possibilitar a compreensão hu-

mana sobre as imagens obtidas. Necessidades em diversos outros domínios também impulsionaram os avanços aos recursos de PDI que se iniciaram em meados de 1960, conforme Gonzalez e Woods (2010).

Na medicina, por exemplo, o tratamento de imagens se deu necessário devido a dificuldade e necessidade na interpretação de radiografias, tomografias, exames de sangue, ultrassonografias, entre outras vertentes. O PDI nesta área proporcionou grande evolução na detecção prévia de doenças e permitiu melhores análises a casos médicos não conhecidos na ciência.

O rápido crescimento das cidades e de zonas de cultivo agrícola estimularam o uso do método na produção de análises geográficas, meteorológicas, populacionais e ambientais que puderam ser formuladas de forma mais ágil e precisa utilizando as informações obtidas através de dados visuais coletados por câmeras.

Assim como tantos campos de estudo, o processamento de imagens também foi incentivado pelo ramo militar, onde a análise pictórica foi importante para o mapeamento e espionagem de territórios inimigos e rastreamento de alvos durante guerras.

A necessidade de recursos que pudessem garantir maior segurança de informações em softwares e ambientes privados também fomentaram os estudos das técnicas de processamento digital de imagens, viabilizando feitos como a identificação biométrica digital, reconhecimento facial, análise de padrões da íris e retina e reconhecimento de pessoas em vídeos de câmeras de vigilância.

Dado o histórico de contribuições, é possível observar que o PDI é uma área que pode ser explorada em múltiplos domínios e cooperar para a otimização de diversos problemas relacionados a identificação visual de objetos. Os dados oriundos de imagens são frequentemente utilizados em estudos sobre o trânsito, pois demandam baixo custo em comparação a outras propostas para a obtenção de informações sobre o tráfego urbano.

Pode-se observar um grande volume de pesquisas na área de análise e processamento de imagens voltadas para o reconhecimento de veículos e placas de carros, este contexto manifesta-se devido ao comum problema com o trânsito que as grandes cidades enfrentam no mundo todo, onde a quantidade de automóveis unida ao não planejamento urbano afetam o deslocamento dos habitantes, em consequência dos congestionamentos e acidentes que ocorrem diariamente.

As aplicações das técnicas de PDI são realizadas de acordo com um objetivo predefinido, onde cada tipo de prática utiliza determinados métodos de transformação para se chegar a resultados adequados com a finalidade da aplicação. Sendo assim, as técnicas utilizadas no processamento de imagens de radiografias, por exemplo, serão diferentes das manipuladas para a detecção de objetos em figuras de ambientes

amplos.

Conforme Filho e Neto (1999), a interpretação do resultado da aplicação de técnicas de realce, filtragem e suavização, por exemplo, é subjetiva, pois depende do conhecimento prévio do observador acerca das imagens originais e do domínio do sistema.

De acordo com Gonzalez e Woods (2010), o processamento de imagens pode ser definido em alguns passos fundamentais, sendo estes os listados a seguir:

- Aquisição de imagens;
- Filtragem e realce de imagens;
- Restauração de imagens;
- Processamento de imagens coloridas;
- Wavelets e Processamento de Multirresolução;
- Compressão;
- Processamento Morfológico;
- Segmentação;
- Representação e Descrição ;
- Reconhecimento de Objetos.

A figura 2.1 apresenta a relação dessas etapas, indicando qual a saída resultante de cada processo, sendo imagens ou atributos.

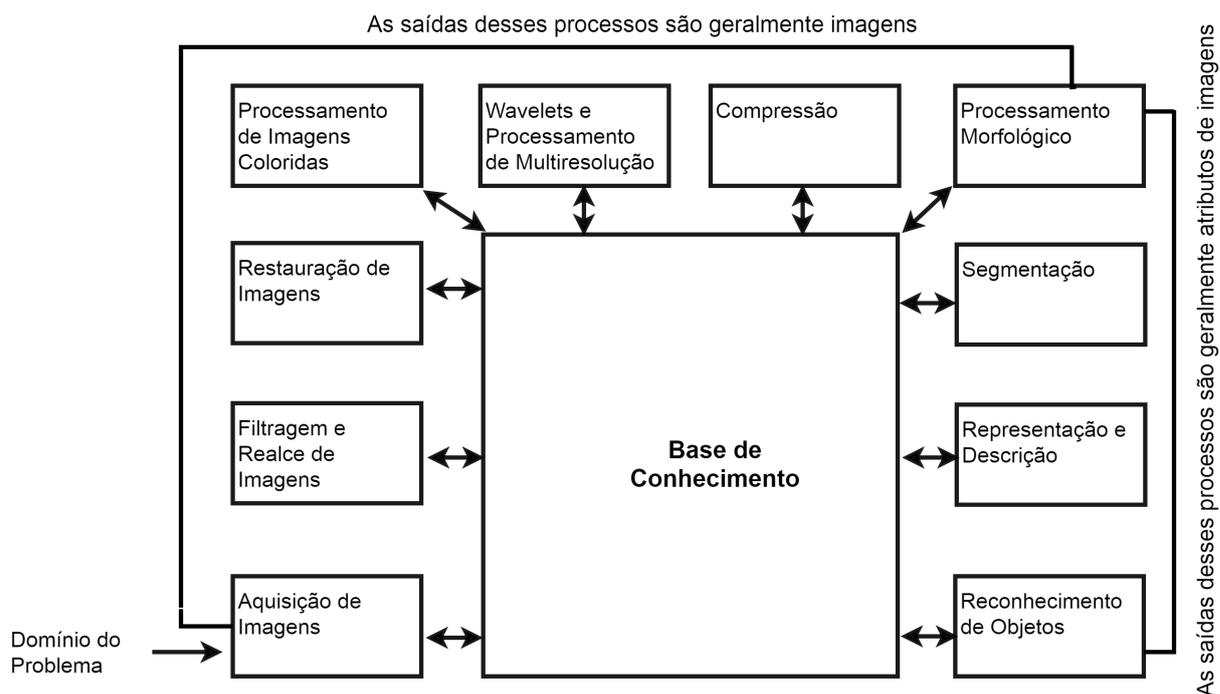


Figura 2.1 – Representação dos principais processos utilizados na técnica de processamento digital de imagens. Imagem adaptada de (GONZALEZ; WOODS, 2010).

A primeira etapa de **Aquisição de Imagens** se refere a forma como os dados de entrada do software de PDI serão adquiridos, podendo ser obtidos de maneira simples com as imagens de entrada já em formato digital ou tendo necessidade de converter esses dados. Neste projeto propõe-se o uso de um **Veículos Aéreo Não Tripulado (VANT)** para a obtenção de cenas aéreas sobre uma via pública.

O estágio de **Filtragem e Realce** diz respeito a uma transformação inicial na imagem de entrada, de modo que a saída seja uma imagem aprimorada, isto é, a ilustração de saída será manipulada com técnicas de realce de acordo com a finalidade da mesma, pois, assim como as técnicas de processamento de imagens, o realce deve ser executado conforme a demanda da aplicação.

A fase de **Restauração de Imagens**, como o nome sugere, consiste na reparação da imagem de entrada e, diferentemente do realce, que é intuitivo, essa etapa segue técnicas mais objetivas, se baseando em modelos matemáticos e probabilísticos.

O estágio de **Processamento de Imagens Coloridas** lida com a transformação das imagens digitais em relação a modelos e cores. A coloração pode ser utilizada como meio de constatar características de interesse na cena analisada.

A etapa **Wavelets e Processamento de Multirresolução** se refere a transformada de *Wavelets*, ferramenta alternativa a transformada de Fourier para o processa-

mento de sinais (sinais de rádio, sísmicos, imagens, etc).

De acordo com Bianchi (2006), enquanto a transformada de Fourier utiliza funções senoidais de diferentes frequências para a representação de sinais, a de *Wavelets* faz o uso de funções de suporte compacto na transformação de um sinal para o espaço de tempo-escala. Na área de processamento de imagens, a transformada de *Wavelets* tem capacidade de representar uma imagem em escala menor sem grandes perdas, preservando informações espaciais e de frequência.

A **Compressão**, como o nome indica, diz respeito a redução do espaço de armazenamento que a imagem ocupa ao ser salva ou transmitida.

O passo de **Processamento Morfológico** lida com a modelagem destinada à análise da forma de um objeto digital, segundo (QUEIROZ; GOMES, 2001). Nesse estágio, os aspectos resultantes, que eram imagens anteriormente, passam a ser representados como atributos de imagens, fornecendo mais que informações pictóricas, conforme é apontado na figura 2.1.

Em visão computacional, a **Segmentação** diz respeito a subdivisão de uma cena em várias regiões, proporcionando a identificação de objetos constituintes na imagem. O parcelamento da imagem deve ser dedicado a cada aplicação específica, pois cada prática exigirá um determinado limite de divisões de acordo com sua finalidade. Segundo Gonzalez e Woods (2010), esta é uma das tarefas mais difíceis e importantes no processamento de imagens, pois o sucesso nos resultados finais de reconhecimento de objetos dependem substancialmente de uma segmentação bem sucedida.

No passo de **Representação e Descrição**, as regiões derivadas da fase de segmentação são representadas como fronteiras ou regiões completas e, em alguns casos, ambas as formas como complementos entre si. A parte de descrição, consiste na definição/classificação das características dos dados e atributos resultantes.

O **Reconhecimento de Objetos** é, basicamente, a identificação dos objetos resultantes do processamento computacional, como por exemplo "Veículos", no caso dessa proposta.

A **Base de Conhecimento** é um banco de dados onde as informações prévias sobre o domínio ficam armazenadas.

Em um sistema de processamento de imagens não é obrigatória a passagem por todos os estágios apresentados no modelo da figura 2.1, cada aplicação irá demandar as etapas necessárias para o seu tipo de interesse. Dessa forma, enquanto uma determinada prática pode exigir a utilização de todos os passos descritos, uma outra pode precisar de apenas três para se chegar ao resultado desejado.

Nos tópicos a seguir serão expostas, com maiores detalhes, as etapas consideradas mais importantes no processamento de imagens, do ponto de vista deste trabalho, expondo suas principais características e técnicas.

2.2 Segmentação

No contexto de visão computacional e processamento de imagens, a **Segmentação** se refere ao processo de fracionamento de uma imagem digital em diversas regiões, proporcionando a subdivisão dos componentes presentes na imagem. O nível de divisão deve ser delimitado de acordo com a necessidade da aplicação, isto é, as divisões são praticadas até que os objetos de detecção requeridos possam ser reconhecidos individualmente.

Devido a capacidade de subdividir imagens em múltiplas parcelas, permitindo a separação dos objetos em uma cena, a fase de segmentação se torna um elemento de grande importância no processamento de imagem, pois é a partir desses fragmentos gerados que os itens poderão ser detectados.

Segundo Gonzalez e Woods (2010), o sucesso da aplicação está diretamente relacionado a qualidade do processo de segmentação no sistema, pois é a partir desse método que os objetos constituintes de uma imagem poderão ser separados e reconhecidos.

Algumas das principais técnicas de segmentação de imagem são a separação por Limiarização, por Crescimento de Região e por Watersheds, o funcionamento de cada processo está apresentado nas seções 2.2.1, 2.2.2 e 2.2.3, respectivamente.

2.2.1 Limiarização (Thresholding)

A Limiarização (ou binarização) é uma técnica de segmentação de imagem que divide as partições em dois modos, separando os objetos do fundo, através de um limiar T (*thresholding*). Dessa forma, a técnica gera uma imagem binária segmentada $g(x,y)$ que pode ser representada pela equação 2.1, onde um “ponto do objeto” é determinado se, na imagem original, $f(x,y) > T$, caso $f(x,y) \leq T$, então o ponto é denominado “ponto de fundo”. Normalmente o valor 1 é atribuído ao objeto e o valor 0 para o fundo.

$$g(x,y) = \begin{cases} 1 & \text{se } f(x,y) > T \\ 0 & \text{se } f(x,y) \leq T \end{cases} \quad (2.1)$$

De acordo com Gonzalez e Woods (2010), os fatores que mais influenciam no processo de limiarização são:

- Separação entre os picos;
- Índice de ruído na imagem;
- Tamanho relativo dos objetos e do fundo;
- Processamento de imagens coloridas;
- Uniformidade da fonte de iluminação;
- Uniformidade das propriedades de reflexão da imagem.

Os valores atribuídos ao limiar *Theresholding* podem ser constantes (limiarização global) ou variáveis (limiarização variável), de acordo com a aplicação. Este método é interessante para as técnicas de análise de imagens, devido a alta velocidade computacional e implementação simples.

2.2.2 Segmentação por Crescimento de Região

O método de segmentação por crescimento de região, assim como o nome sugere, trabalha com o aumento das sub-regiões existentes através do agrupamento de pixels de uma imagem em função da formação de novos subconjuntos maiores. O agrupamento se inicia com um conjunto de pontos denominados semente e, de acordo com a semelhança entre os pixels vizinhos e as sementes, o agrupamento é realizado.

As propriedades de semelhança que definem a união dos pixels são, normalmente, predefinidas de acordo com o domínio da aplicação.

2.2.3 Segmentação usando Watersheds

A palavra *watersheds* significa "Bacias Hidrográficas", o nome é relacionado a forma como a técnica funciona. O conceito baseia-se no fracionamento de dados em três dimensões, onde a análise é efetuada com a imagem topográfica da cena.

Para a segmentação das regiões, estipula-se que as divisões na imagem funcionam como áreas de inundação de bacias hidrográficas, dessa forma, conforme o nível da água sobe de baixo para cima, barreiras são criadas e aumentadas para manter separadamente um território hidrográfico de outro. Os barramentos aumentam conforme o volume da água cresce até que o limite máximo de inundação seja atingido.

Dessa forma, as linhas de *watershed* finais são o resultado da segmentação das regiões da cena. A imagem 2.2 representa as fases do processo de inundação, criação de *watersheds* e barragens.

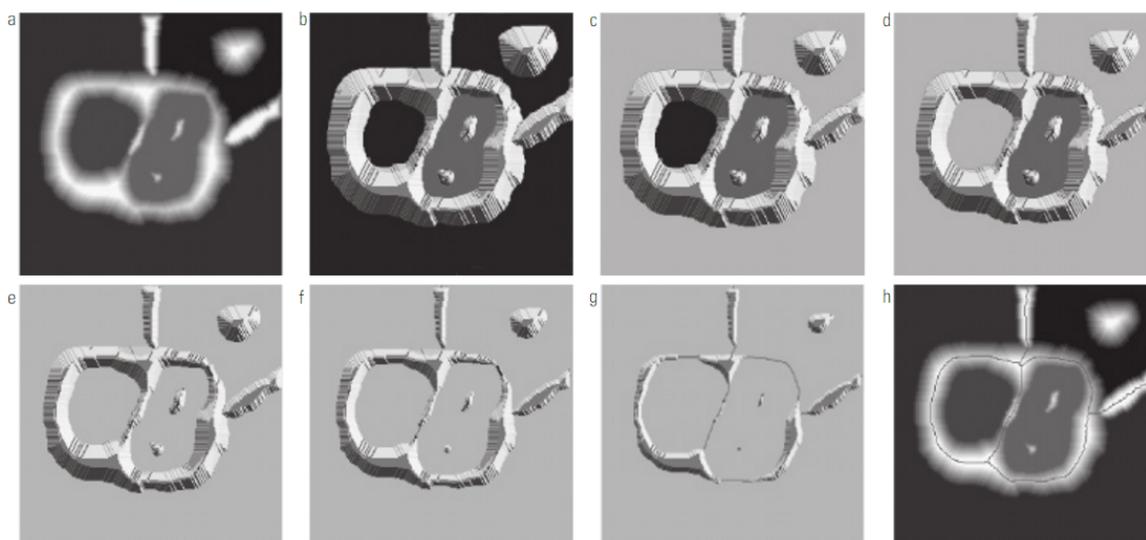


Figura 2.2 – (a) Imagem original. (b) Imagem topográfica. (c) e (d) Duas fases da inundação. (e) Novas inundações. (f) Início da união entre duas *watersheds* e a barragem construída entre elas. (g) Barreiras maiores. (h) Linhas finais de *watershed*, representando a segmentação da imagem. Imagem adaptada de (GONZALEZ; WOODS, 2010)

2.3 Subtração de Fundo

Um dos primeiros passos para se iniciar o processo de reconhecimento de objetos em uma imagem é a separação do *background* e *foreground*, isto é, dividir os componentes da cena (*foreground*) do fundo da imagem (*background*).

A segmentação básica de *background* é realizada utilizando uma imagem de referência, contendo somente o fundo da cena, dessa forma, uma operação de subtração é realizada entre as imagens atual e a de modelo. Esta imagem utilizada como parâmetro não é válida para imagens de ambientes externos não controlados, pois o fundo padrão é uma figura estática.

A figura 2.3 ilustra o funcionamento de uma subtração de fundo simples utilizando uma imagem de referência do fundo. A figura foi obtida na documentação da biblioteca de Visão Computacional **OpenCV** (*Open Source Computer Vision Library*).

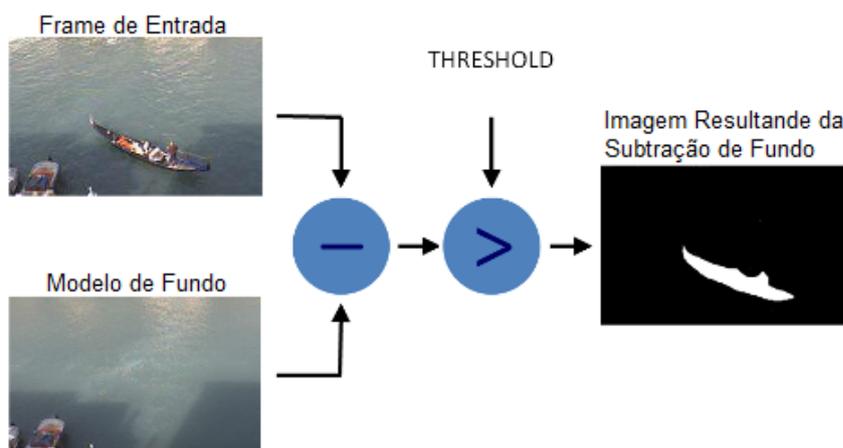


Figura 2.3 – Exemplo de funcionamento da aplicação do método de Subtração de Fundo utilizando uma imagem fixa de referência. Imagem adaptada da documentação do OpenCV.

De acordo com Higashino (2006), algumas situações podem influenciar negativamente na produção do modelo de referência para ser utilizado em ambientes não controlados:

1. Objetos que assemelham-se ao fundo da cena, podem não ser detectados;
2. Imagens de ambientes externos estão suscetíveis a interferência climática (iluminação);
3. Fundo não estático, interferência de outros objetos que não fazem parte do interesse da aplicação (árvores, arbustos, vento, etc);
4. Textura do objeto alvo semelhante à textura do fundo;
5. Mudanças na cor do objeto quando em movimento;
6. Situações onde somente o fundo é observável não são possíveis para a montagem do modelo de referência;
7. Complexidade em diferenciar um objeto que pára de mover-se por um tempo e um objeto que precisa ser incorporado ao modelo;
8. Objetos que alternam entre locomovendo-se e estáticos na cena;
9. Sombras nos alvos de detecção, podem dificultar a precisão de reconhecimento.

A partir desses problemas, Higashino (2006) propôs um conjunto de etapas de subtração de fundo, a figura 2.4 ilustra os 4 processos definidos.

1. **Pré-Processamento:** Trabalha com a conversão do vídeo para um formato que possa ser processado;
2. **Modelagem do Fundo:** Tem função de construir e atualizar o modelo de fundo;
3. **Detecção de Objetos:** Tem objetivo de identificar os pontos não pertencentes ao *background*;
4. **Validação:** Validação dos pontos identificados na etapa 3, removendo as falsas detecções.

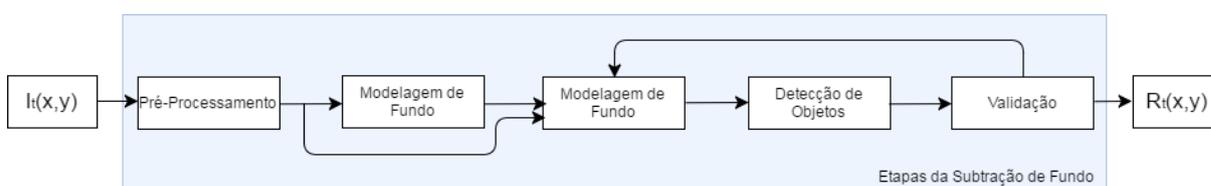


Figura 2.4 – Etapas de um algoritmo de Subtração de Fundo, Pré-Processamento, Modelagem do Fundo, Detecção de Objetos e Validação. Imagem adaptada de (HIGASHINO, 2006)

2.3.1 Subtração de Fundo com Dois ou Mais Quadros Consecutivos

Em um vídeo, a imagem de referência de fundo está sempre mudando, dessa forma, a detecção do objeto em movimento é realizada com a subtração de quadros consecutivos de uma sequência. Segundo Ferreira (2012), este algoritmo pode ser considerado como uma variação do algoritmo de subtração do fundo, visto que ocorre a subtração de imagens, porém, neste caso, a imagem de fundo está em constante mudança com o tempo.

De acordo com Coelho (2014), *Mixture of Gaussians* (MOG) é um método de subtração de fundo que tem a atualização do modelo de fundo através das misturas gaussianas para determinar o valor do pixel de fundo. Este método se adapta de forma eficiente a alterações de movimento e iluminação de objetos em cenas externas, porém, apresenta alguns problemas quando os objetos em movimento ficam parados em alguns momentos.

2.4 Morfologia Matemática

A definição da palavra **Morfologia** se refere ao estudo da forma, da aparência externa da matéria. No processamento de imagens a morfologia tem função de extrair componentes e descrições relevantes dos objetos nas cenas. No modelo de Gonzalez

e Woods (2010), representado na figura 2.1, a morfologia matemática se encontra na transição dos dados de saída, que nas etapas antecedentes eram de imagens e a partir dessa fase passam a ser atributos de imagens.

Segundo Queiroz e Gomes (2001), na análise de imagens, a morfologia digital se fundamenta no conceito de que a imagem é um conjunto de pontos elementares que formam subconjuntos elementares e a inter-relação entre eles formam estruturalmente a morfologia de uma imagem. No processamento de imagens é normalmente utilizada para a correção de imperfeições geradas pela segmentação da cena.

A morfologia digital é baseada na teoria de conjuntos, onde os conjuntos representam os objetos encontrados na imagem em análise. As operações de **translação** e **reflexão** de conjuntos são muito utilizadas no processamento morfológico para tornar possível a produção de operações baseadas em **Elementos Estruturantes (ES)**, que são subconjuntos de imagens.

Em teoria de conjuntos, a Reflexão pode ser representada como:

$$\hat{B} = \{w | w = -b, \text{ para } b \in B\} \quad (2.2)$$

Dessa forma, aplicando a Reflexão em um par de coordenadas (x,y) , obtém-se $(-x,-y)$.

Já a Translação de um conjunto B, no ponto $z = (z_1, z_2)$, representada por pode ser definida como:

$$(B)_z = \{w | w = -b, \text{ para } b \in B\} \quad (2.3)$$

O Processamento Morfológico possui duas operações básicas e essenciais, sendo essas a **Erosão** e a **Dilatação**, e também as formas variantes dessas operações, sendo **Abertura** e **Fechamento**, que serão detalhadas nas seções seguintes 2.5.1, 2.5.2 e 2.5.3.

Segundo Facon (2011) a força da Morfologia Matemática reside no fato de quantificar a intuição do pesquisador, analisando a estrutura geométrica das imagens a partir de um conjunto perfeitamente definido e conhecido pelo usuário chamado de Elemento Estruturante.

2.5 Elemento Estruturante

De acordo com Facon (2011), o **Elemento Estruturante** ou **Kernel** interage com cada entidade contida na imagem, modificando a sua aparência e tamanho, permitindo assim tirar algumas conclusões necessárias. Este elemento deve ser simples

para facilitar a interpretação das imagens e é definido pelos pixels que o formam. Seguindo o exemplo do autor, o Elemento Estruturante pode ser representado com as notações “.” e “•”, sendo “.” um pixel do fundo e “•” um pixel ativo, com função importante na interação com a imagem.

Nas representações das técnicas de Dilatação e Erosão que serão abordadas nas seções 2.5.1 e 2.5.2, B_x representa o elemento estruturante B, centrado no pixel x. De acordo com Facon (2011), os elementos estruturantes mais utilizados em morfologia matemática, são os representados a seguir:

- Elemento Estruturante Horizontal:

$$B_H = \left\{ \begin{array}{ccc} \cdot & \cdot & \cdot \\ \bullet & \bullet & \bullet \\ \cdot & \cdot & \cdot \end{array} \right\}$$

- Elemento Estruturante Vertical:

$$B_V = \left\{ \begin{array}{ccc} \cdot & \bullet & \cdot \\ \cdot & \bullet & \cdot \\ \cdot & \bullet & \cdot \end{array} \right\}$$

- Elemento Estruturante em Cruz:

$$B_C = \left\{ \begin{array}{ccc} \cdot & \bullet & \cdot \\ \bullet & \bullet & \bullet \\ \cdot & \bullet & \cdot \end{array} \right\}$$

- Elemento Estruturante Quadrado:

$$B_Q = \left\{ \begin{array}{ccc} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{array} \right\}$$

- Elemento Estruturante Rhombus:

$$B_R = \left\{ \begin{array}{cccccc} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \bullet & \bullet & \bullet & \cdot \\ \cdot & \bullet & \bullet & \bullet & \bullet & \cdot \\ \cdot & \bullet & \bullet & \bullet & \bullet & \cdot \\ \cdot & \cdot & \bullet & \bullet & \bullet & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{array} \right\}$$

O Elemento Estruturante padrão é o Quadrado, isto é, ao aplicar uma operação de Dilatação ou Erosão, o elemento estruturante utilizado será o padrão (Quadrado), caso outro tipo seja necessitado, será necessário definir qual será utilizado.

2.5.1 Erosão

Em análise de imagens, a erosão é utilizada para a remoção de elementos (pixels) não pertencentes a um padrão definido no sistema (Elemento Estruturante). Normalmente é aplicada para deletar objetos que não são de interesse na aplicação.

A operação de erosão dos conjuntos A por B é representada por $A \ominus B$, sendo A e B conjuntos de números inteiros Z^2 de imagens binárias, a equação 2.4 é uma das maneiras de se definir a erosão entre conjuntos.

$$A \ominus B = \{z \mid (B)_z \subseteq A\} \quad (2.4)$$

Na figura 2.5, podemos observar a demonstração do comportamento da operação de erosão, esta técnica reduz a imagem original para atender um padrão de elemento estruturante previamente definido. Cada prática de processamento computacional de imagens digitais irá demandar um determinado modelo como padrão para operações de erosão, por exemplo.

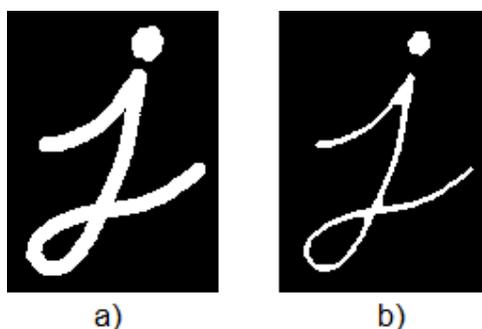


Figura 2.5 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Erosão. Imagem adaptada da Documentação da biblioteca OpenCV.

O Elemento estruturante utilizado na operação representada na imagem 2.5 foi o Quadrado $B_Q = \left\{ \begin{matrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{matrix} \right\}$.

Em suma, no processamento de imagens a manipulação das formas utilizando a técnica de Erosão é importante para aumentar a precisão na detecções dos objetos em cena, visto que o procedimento tem capacidade de remover alguns falso positivos identificados na imagem, conforme Facon (2011).

2.5.2 Dilatação

Diferentemente da Erosão, a operação de Dilatação tende a “aumentar” a imagem, visto que sua principal função é agrupar os pixels próximos para se chegar ao padrão desejado. A equação 2.5 representa a operação de Dilatação.

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\} \quad (2.5)$$

O funcionamento da operação da Dilatação aplicada ao processamento de imagens é evidenciado na figura 2.6. Nesta, observa-se que ocorre o aumento da imagem original.

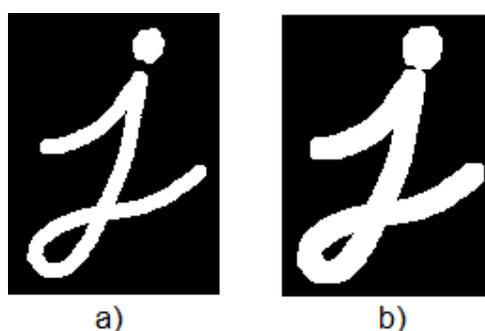


Figura 2.6 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Dilatação. Imagem adaptada da Documentação da biblioteca OpenCV.

O Elemento estruturante utilizado na operação representada na imagem 2.6 foi

o Quadrado $B_Q = \left\{ \begin{array}{ccc} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{array} \right\}$.

Dessa forma, devido a capacidade de transforção de subconjuntos de pixels existentes em uma cena, a aplicação das operações de Dilatação e Erosão são úteis na análise de imagens para o reconhecimento de objetos, visto que proporcionam resultados mais precisos dos objetos presentes na imagem de entrada, de acordo com Facon (2011).

2.5.3 Abertura e Fechamento

Os métodos de Abertura e Fechamento são basicamente a aplicação em sequência dos métodos de Erosão e Dilatação relatados nas seções 2.5.1 e 2.5.2, porém, aplicados diretamente em um único método.

Conforme o Castleman (1996), o método de **Abertura** consiste no processo de Erosão seguido de Dilatação e tem efeito de eliminar pequenos objetos e suavizar

as bordas de grandes objetos sem transformar sua área de forma significativa. Esta função pode ser definida com a equação 2.6, representada a seguir.

$$B \circ S = (B \otimes S) \oplus S \quad (2.6)$$

A figura 2.7 representa a aplicação do método de morfologia matemática de **Abertura**.

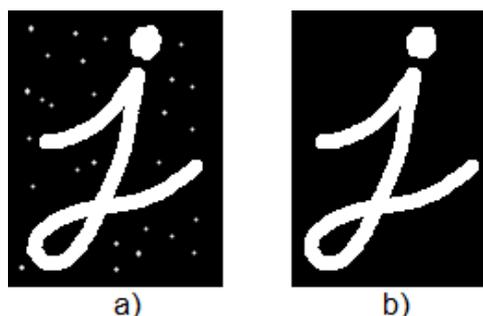


Figura 2.7 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Abertura. Imagem adaptada da Documentação da biblioteca OpenCV.

Em oposição a função de abertura, a operação de **Fechamento** é o processo de Dilatação seguido de Erosão e tem efeito de preencher os "buracos" em objetos, conectando os píxeis vizinhos do objeto. Este método pode ser definido com a equação 2.7 representada a seguir.

$$B \bullet S = (B \oplus S) \otimes S \quad (2.7)$$

A figura 2.8 apresenta a aplicação da técnica de **Fechamento**.

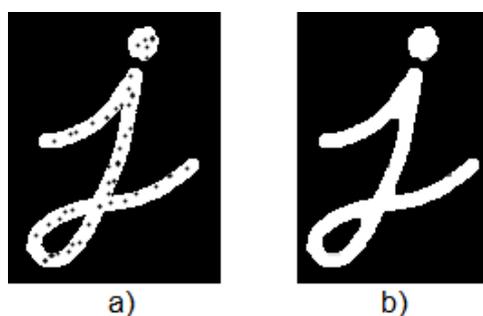


Figura 2.8 – (a) Imagem de entrada. (b) Imagem resultante da aplicação do método de Fechamento. Imagem adaptada da Documentação do OpenCV.

O Elemento estruturante utilizado nas operações de Abertura e Fechamento representadas nas figuras 2.7 e 2.8 foi o Quadrado $B_Q = \begin{Bmatrix} \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet \end{Bmatrix}$.

2.6 Fluxo Óptico

O **Fluxo Óptico** representa o movimento dos pixels em uma sequência de imagens, onde as diferenças na continuidade da cena evidenciam o deslocamento. Segundo Horn e Schunck (1981), o fluxo óptico é a distribuição da velocidade aparente que um padrão de intensidade apresenta quando se move em uma imagem. Em uma cena 2D, o movimento dos objetos 3D são a projeção do movimento 3D na imagem.

O Fluxo Óptico pode fornecer informações importantes para diversas aplicações, como para a análise de imagens, segmentação de regiões, compressão de vídeos, sensores de movimento, detecção de objetos, estimativas de tempo de colisão, visão robótica, entre outras aplicações.

A quantidade e o deslocamento de objetos contidos em uma cena e o movimento do sensor de obtenção das imagens, são fatores que podem influenciar no cálculo do fluxo óptico.

O cálculo do Fluxo Óptico consiste em determinar a velocidade aproximada da movimentação do pixel na imagem e também indicar a direção de locomoção deste. Os métodos para estimar a velocidade de Fluxo Óptico podem ser:

- Métodos Diferenciais;
- Métodos baseados em frequência;
- Métodos baseados em correlação;
- Métodos de múltiplos movimentos;
- Métodos de refinamento temporal.

Dentre os métodos citados, a técnica de Diferenciais é a que mais se destaca entre as estimativas de Fluxo Óptico, segundo Horn e Schunck (1981), a mesma se baseia na mudança da intensidade de brilho da imagem e utilizam a derivada espaço-temporal das intensidades para calcular o movimento.

2.7 Imagens Aéreas com o Uso de *Drones*

De acordo com Marconato (2017), o termo VANT foi adotado pela *Federal Aviation Administration* (FAA) e pela comunidade acadêmica internacional para designar um Veículo Aéreo Não Tripulado, que pode ser controlado através do disparo de comandos por um piloto humano ou efetuar voos autônomos, utilizando uma trajetória pré-programada. Os VANTs são comumente conhecidos como *drones* e foram inicialmente projetados para fins militares durante períodos de guerra, para o transporte de explosivos, análise territorial, combates aéreos, entre outros.

Segundo Marconato (2017), apesar de tornarem-se popularmente conhecidos apenas nos últimos anos, com o uso dos famosos quadricópteros para fins profissionais e de recreação, os VANTs estão presentes em diversas áreas de estudo, como no monitoramento de campos de agricultura, missões militares, supervisão do trânsito, segurança pública, entre outras. A figura 2.9 exibe um VANT simples, tipicamente conhecido.



Figura 2.9 – VANT simples. Imagem adaptada de (MOHAMMED et al., 2014).

Segundo Mohammed et al. (2014), a população mundial está crescendo rapidamente e pode dobrar até 2050, esses dados geram uma preocupação com os serviços e infra-estrutura das cidades, que deverão suportar uma grande quantidade de habitantes e disponibilizar sistemas que otimizem a mobilidade dos cidadãos. Dessa forma, crescem as áreas de interesse focadas no desenvolvimento de *Smart Citys*

para que esses desafios sejam supridos, utilizando soluções de Informação e Comunicação Tecnológica (ICT) de baixo custo.

Neste contexto, os VANTs estão sendo cada vez mais utilizados para contribuir com os objetivos das *Smart Citys*, sendo adotados e integrados em diferentes aplicações e funções. Dessa forma, foi definido neste projeto o uso de imagens aéreas capturadas com um VANT, devido a facilidade, vantagens e tendências do dispositivo no desenvolvimento de soluções que visam melhorar os serviços e mobilidade nos municípios.

2.8 Conclusões do Capítulo

Este capítulo apresentou o embasamento teórico sobre o tema de pesquisa do trabalho, conceituando os principais recursos, ferramentas, modelos e técnicas pesquisados para o desenvolvimento de uma aplicação capaz de efetuar a detecção e quantificação de veículos existentes em uma rua pública, através de imagens aéreas.

A intenção da seção foi proporcionar uma base de conhecimento para o melhor entendimento sobre os conceitos já utilizados no processamento de imagens em função da identificação de objetos, de acordo com pesquisas e análises dos dados acerca do assunto.

Através deste levantamento de dados, foi possível determinar alguns métodos a serem utilizados na implementação da ferramenta experimental e adquirir conhecimento a respeito dos componentes necessários para o desenvolvimento de uma aplicação capaz de detectar veículos em movimento com PDI aplicado a imagens aéreas.

Capítulo 3

TRABALHOS CORRELATOS

Nesta seção serão expostos alguns trabalhos publicados que apresentam ferramentas, técnicas e metodologias interessantes para o desenvolvimento deste projeto. Serão descritas as estratégias utilizadas nestas propostas e a relevância que cada tese pode agregar neste trabalho.

Assim como indicado no Capítulo 2, o PDI é uma área que estuda a transformação da imagem por um processo computacional que trabalha com três principais funções: a captura, a representação e a transformação de imagens. O método possibilita o aumento da qualidade visual de uma imagem, facilitando a interpretação humana e é utilizado para análises em diversos domínios, nos ramos da medicina, astronomia, biologia, geografia, entre outros.

As imagens resultantes do PDI variam de acordo com o contexto de aplicação do método, pois cada tipo de prática é implementado seguindo um propósito específico, isto é, os processos utilizados em cada seguimento exigem diferentes modos de processamento de uma cena para alcançar o resultado desejado.

3.1 Processamento de imagens aéreas com VANTs e câmeras convencionais

Seguindo a mesma intenção do trabalho proposto neste documento, que está direcionado a detecção de veículos através da análise e manipulação de imagens aéreas, o projeto de Maria et al. (2016) apresenta um método para o reconhecimento de veículos através do processamento de imagens obtidas por câmeras fixas e VANTs, onde é apresentada uma comparação na aplicação dos métodos de processamento digital em cada modelo de sensor empregado. A proposta também foca no reconhecimento de automóveis para o monitoramento de estacionamentos.

A definição da Região de Interesse (ROI) das imagens de entrada são esta-

belecionadas de forma diferente em cada tipo de sensor de captura, pois nos vídeos adquiridos com VANTs essa área deve ser estabelecida a cada frame, enquanto as câmeras fixadas tem a ROI marcada apenas uma vez.

Para a detecção dos carros um classificador em cascata foi utilizado, a estratégia foi aplicada seguindo a proposta de um dos métodos mais citados no mundo científico de Viola e Jones (2001), que propôs a combinação de múltiplos classificadores (cascata) em proveito do reconhecimento facial.

Primeiro foi realizado um treinamento do Classificador, onde um conjunto de amostras positivas e negativas foi inserido como referência para o aprendizado.

O algoritmo relatado é composto por duas etapas, sendo: **Pré Processamento** e **Detecção dos Carros**. A figura 3.1 expõe o diagrama da arquitetura utilizada no sistema desenvolvido.

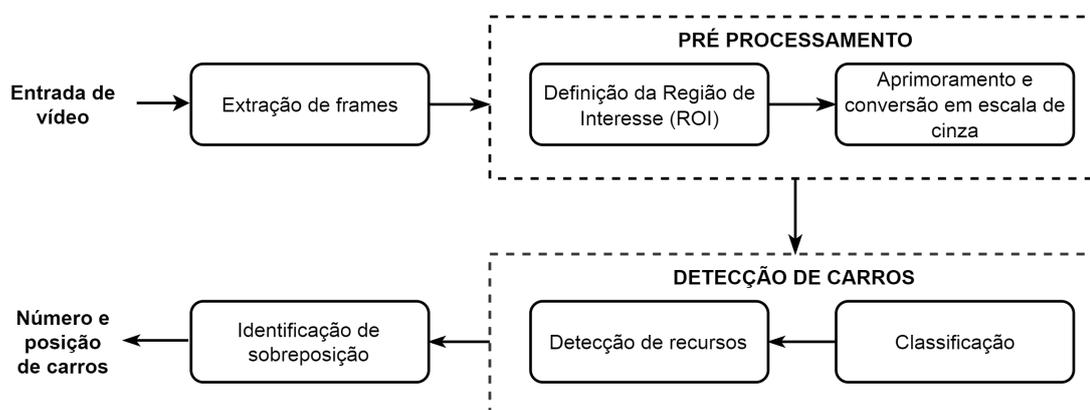


Figura 3.1 – Diagrama representando a arquitetura do sistema proposto por Maria et al. (2016). Imagem adaptada de (MARIA et al., 2016).

Na fase de **Pré Processamento** ocorre a definição da Região de Interesse e a conversão da imagem em escala cinza, a fim de delimitar a quantidade de dados a serem analisados, de acordo com a intenção da aplicação.

Na etapa de **Detecção de Carros**, é utilizado um classificador em cascata para o treinamento da aplicação, conforme descrito anteriormente neste tópico. Este aprendizado é, normalmente, um processo demorado, onde as imagens de referência serão inseridas como modelo para a detecção. Essas imagens são do objeto alvo, mas também de itens negativos, não relevantes.

A figura 3.2 ilustra exemplos de modelos utilizados para o treinamento da aplicação e a figura 3.3 ilustra o funcionamento do sistema proposto por Maria et al. (2016), com as detecções dos veículos na ROI determinada. Os retângulos vermelhos indicam os veículos detectados e as limitações em amarelo representam a ROI determinada.



Figura 3.2 – Exemplos de modelos utilizados no treinamento da aplicação proposta por Maria et al. (2016). Imagem adaptada de (MARIA et al., 2016).

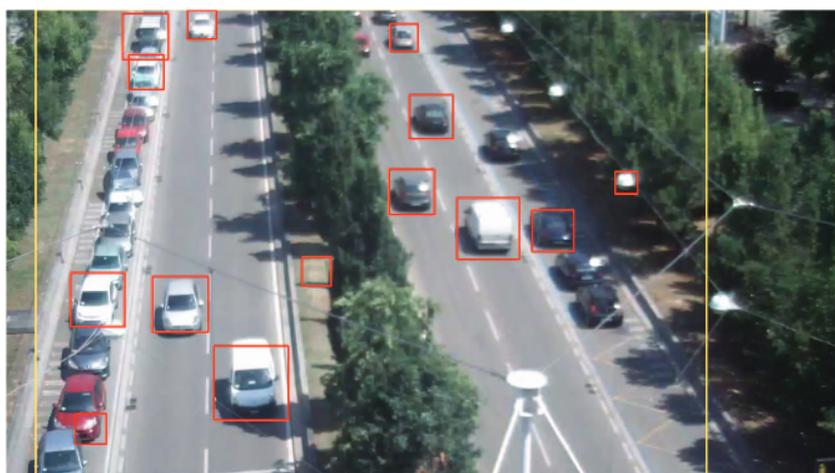


Figura 3.3 – Detecção de veículos na ROI definida no sistema de Maria et al. (2016). Imagem adaptada de (MARIA et al., 2016).

O sistema proposto por Maria et al. (2016) foi desenvolvido utilizando a linguagem de programação C++ com a biblioteca OpenCV, estes recursos demonstram eficiência em aplicações de visão computacional e processamento de imagens.

Segundo Maria et al. (2016) a definição da ROI é um procedimento importante, pois esta técnica favorece a produção de resultados mais precisos e reduz a ocorrência de falsos positivos na análise. Devido a sua relevância, o método é considerado válido do ponto de vista desse trabalho, uma vez que imagens aéreas capturam amplas áreas e trazem diversos elementos não interessantes ao objetivo da aplicação, como casas, prédios e terrenos, por exemplo e aponta a importância da definição de uma ROI para o sucesso da aplicação.

3.2 Contagem Automática de Veículos Baseada em Vídeos de Segurança da Internet

O trabalho de Ribeiro e Correia (2014), propôs utilizar o processamento de imagens em vídeos de vigilância disponíveis na internet para o reconhecimento de veículos em movimento, a fim de fornecer dados relevantes aos motoristas sobre a

situação do trânsito no momento, diferente deste trabalho, que propõe uma forma de obter informações relevantes sobre o fluxo de veículos que possa ser utilizada para alimentar sistemas de simulação de trânsito.

A figura 3.4 ilustra uma arquitetura genérica sobre a proposta de Ribeiro e Correia (2014).

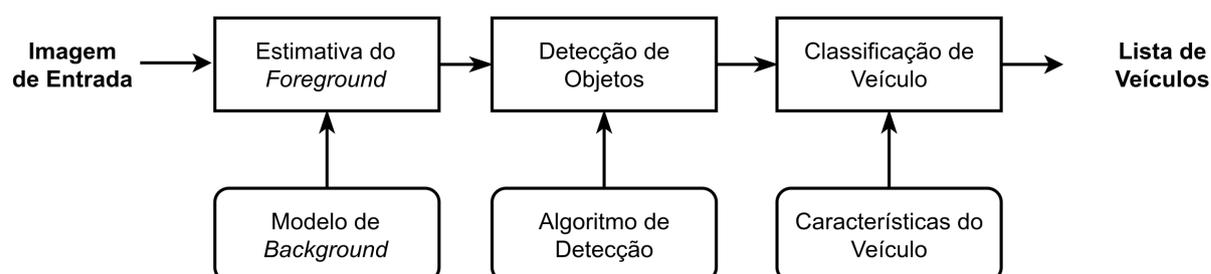


Figura 3.4 – Diagrama da arquitetura genérica de um sistema de gerenciamento do trânsito utilizado por Ribeiro e Correia (2014). Imagem adaptada de (RIBEIRO; CORREIA, 2014).

Neste projeto, a detecção é efetuada utilizando Linhas de Detecção Virtuais (VDLs), a figura 3.5 ilustra a VDL inserida perpendicularmente em uma rodovia para a quantificação dos veículos, de acordo com Ribeiro e Correia (2014), o uso de uma VDL traz vantagens, pois é eficiente independentemente da resolução ou tempo de duração do vídeo.



Figura 3.5 – Representação da VDL utilizada na proposta de Ribeiro e Correia (2014). Imagem adaptada de (RIBEIRO; CORREIA, 2014).

Diferentemente deste trabalho que foi implementado utilizando uma técnica de remoção de *background* disponibilizada na biblioteca de visão computacional OpenCV, o projeto de Ribeiro e Correia (2014) propõe o uso de um novo método de subtração de fundo, que funciona com a identificação do contorno dos objetos, de acordo com a variação dos pixels dos objetos em movimento nas imagens.

Como forma de diminuir a detecção de falso-positivos, o trabalho propõe a realização de algumas validações em relação a área dos objetos alvo, pré-determinando

um valor mediano de área para ser utilizado nas comparações e validar ou não veículos reconhecidos.

O projeto em questão também apresenta comparações de desempenho, validando os resultados obtidos em testes realizados em diferentes períodos (dia e noite).

Para o ponto de vista do trabalho apresentado neste documento, a proposta de Ribeiro e Correia (2014) é interessante, pois apresenta uma aplicação que realiza a contagem de veículos em movimento utilizando uma VDL para o procedimento e propõe alguns modos de diminuir a taxa de detecção de falsos-positivos no sistema.

O trabalho proposto neste documento apresenta diferenciais sobre a teste de Ribeiro e Correia (2014), pois além de utilizar VDLs para a contagem dos veículos, é realizada a definição de ROIs em locais específicos, para que a partir da detecção dos carros em movimento, possam ser gerados dados pertinentes sobre o fluxo de automóveis que prosseguem para determinadas ruas ou avenidas.

3.3 Reconhecimento de objetos parcialmente ocultos pelo ambiente

Grande parte das propostas que focam na identificação de objetos raramente consideram o tratamento de itens parcialmente exibidos nas cenas, isto é, se concentram apenas no objetivo específico do método e não investigam a fundo as possíveis variações que a imagem de entrada pode apresentar. Neste contexto, o trabalho de Qiu, Wen e Fan (2017) propõe um método que dedica-se ao reconhecimento de objetos em situações onde o alvo não está completamente explícito na imagem.

A proposta apresenta um novo método de detecção de objeto ocluso utilizando o modelo de Objeto de Configuração Parcial (PCM - *Partial Configuration Object Model*) de imagens de satélite. Esta técnica é baseada no uso de multicamadas para a representação do elemento de referência no sistema de reconhecimento. As figuras 3.6 e 3.7 demonstram o fracionamento da imagem, salientando as características locais do objeto e as camadas utilizadas no modelo de particionamento PCM.

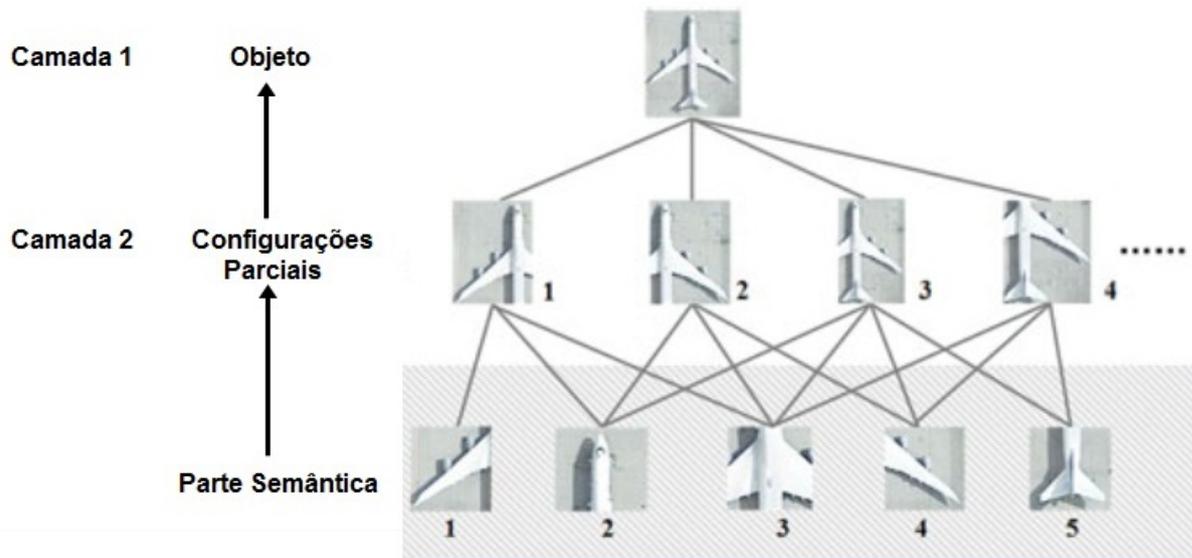


Figura 3.6 – Representação do particionamento em multicamadas pelo modelo PCM - *partial configuration object model*. Imagem adaptada de (QIU; WEN; FAN, 2017).

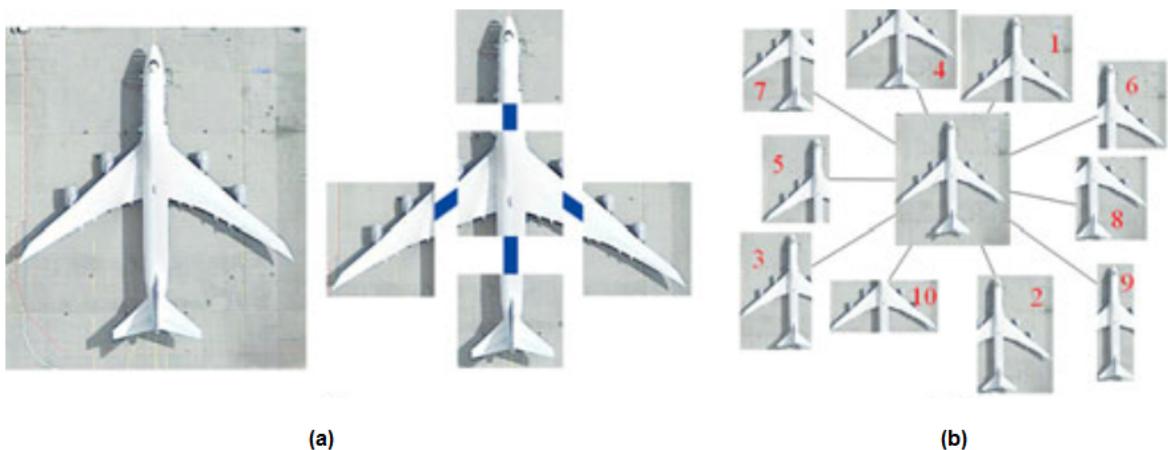


Figura 3.7 – (a) Partes semânticas (b) Configurações parciais. Imagem adaptada de (QIU; WEN; FAN, 2017).

Com os estudos focados no reconhecimento de objetos oclusos em imagens, a proposta de Qiu, Wen e Fan (2017) é considerada interessante, pois é direcionada ao tratamento de regiões ocultas em uma imagem, fator que pode contribuir na identificação de falsos positivos, colaborar para aprimorar a precisão dos resultados finais de detecção de veículos e ajudar nas análises de imagens captadas em ambientes externos não favoráveis.

3.4 Monitoramento veicular utilizando câmeras de segurança convencionais e visão computacional

O projeto de Junior (2016) utiliza os métodos de visão computacional no processamento de imagens e vídeos para a identificação de veículos e também para o cálculo da velocidade dos automóveis que trafegam por uma determinada rua ou avenida monitorada com câmeras convencionais de segurança.

Assim como na tese de Maria et al. (2016), descrita no tópico 3.1 deste capítulo, Junior (2016) propõe o uso da biblioteca de programação OpenCV, ferramenta que é comumente utilizada em propostas direcionadas ao processamento de imagens para reconhecimento de elementos individuais na cena, a mesma se mostra eficiente em visão computacional e aplicações de tempo real. Através do OpenCV é possível implementar ferramentas de interpretação de imagens, suportando operações simples como filtros até operações complexas, como a análise de movimentos, reconhecimento de padrões e reconstrução em 3D.

Para o trabalho proposto, focado na detecção de veículos transitando em vias públicas através do processamento de imagens aéreas, as ferramentas e métodos utilizados para o reconhecimento dos automóveis no projeto de Junior (2016) mostram-se interessantes para este desenvolvimento, visto que colaboram para a identificação rápida de objetos em cenas de ambientes externos.

3.5 Conclusões do Capítulo

Em síntese, este capítulo apresentou alguns trabalhos correlatos, apontando as ferramentas e técnicas utilizadas no processamento de imagens para identificação de objetos. A partir do estudo destas propostas, foi possível obter dados relevantes sobre os recursos eficientes para o desenvolvimento do protótipo de sistema proposto neste documento.

A tabela 3.1 expõe algumas características pertinentes no processamento de imagens e o que é tratado na proposta de cada trabalho correlato. Os itens listados a seguir indicam o trabalho correlato correspondente a cada proposta presente na tabela 3.1:

- Trabalho 1: Referente a proposta descrita no item 3.1;
- Trabalho 2: Referente a proposta descrita no item 3.2;
- Trabalho 3: Referente a proposta descrita no item 3.3;
- Trabalho 4: Referente a proposta descrita no item 3.4.

Tabela 3.1 – Comparação de características entre os trabalhos correlatos.

Características	Trabalho 1	Trabalho 2	Trabalho 3	Trabalho 4
Sensor de Captação das Imagens	VANT	Vídeos da Internet	VANT	Câmeras Convencionais
Background Estático	Não	Não	Sim	Sim
Deteção de Veículos	Sim	Sim	Não	Sim
Região de Interesse	Sim	Não	Sim	Sim
Contagem de Objetos	Não	Sim	Não	Não
Modelos de Objeto Alvo	Sim	Não	Sim	Não

Cada propriedade exposta na tabela 3.1 representa particularidades que os trabalhos correlatos possuem e se isso é comum ou não aos outros projetos. Seguem itens analisados:

1. **Sensor de captação de imagens:** Se refere a qual o tipo de equipamento foi utilizado na aquisição das imagens de entrada de cada sistema.
2. **Background Estático:** É referente ao uso de um modelo fixo de background.
3. **Deteção de veículos:** É a indicação sobre qual é o objeto alvo de identificação da proposta, sendo veículos ou não.
4. **Região de Interesse:** É a indicação se o trabalho utilizou uma Região de Interesse no desenvolvimento.
5. **Contagem de objetos em movimento:** Indica se a aplicação proposta no trabalho correlato realiza a contagem dos veículos em movimento.
6. **Modelos de Objeto Alvo** Indica se a aplicação proposta no trabalho correlato utiliza modelos de objeto alvo para a deteção.

As informações na tabela demonstram que, apesar das diferentes finalidades, as propostas tem vários pontos em comum. Podemos verificar que o referencial aéreo para a obtenção das imagens foi o favorito entre os projetos e que a definição da região de interesse para o processamento de imagens é de grande importância, visto que a maioria dos trabalhos citados utilizaram a técnica e validaram que os resultados utilizando uma ROI são mais precisos.

Apenas um dos trabalhos correlatos (proposta descrita na seção 3.2) apresentou técnicas para a contagem dos veículos, porém, não focou em um propósito específico para gerar dados sobre o tráfego de automóveis a partir dos objetos contados.

Os trabalhos 1 e 3, descritos nas seções 3.1 e 3.3 propõem o uso de modelos de referência dos objetos alvo para que possam ser identificados, diferentemente desta proposta que detecta os veículos em movimento a partir da aplicação dos métodos de **Subtração de Fundo**,

Em suma, cada trabalho correlato apresentou contribuições relevantes em relação aos recursos, técnicas e linguagens ideais para a implementação de um sistema de reconhecimento de veículos em ambientes externos, proporcionando maior entendimento sobre o assunto e auxílio nas decisões sobre a estrutura do projeto.

Capítulo 4

DESENVOLVIMENTO

Este capítulo tem objetivo de apresentar com detalhes os métodos e fluxos utilizados para a produção de um protótipo que demonstra o funcionamento da ferramenta proposta neste documento, apontando os recursos, linguagens e bibliotecas utilizadas no desenvolvimento da aplicação, assim como a importância de cada componente para o projeto.

4.1 Implementação

Para a implementação de um protótipo de sistema que realize o reconhecimento de veículos em movimento em imagens aéreas, foram definidas algumas etapas para o desenvolvimento, representadas na figura 4.1. A figura indica o fluxo utilizado desde a obtenção dos dados de entrada até os valores resultantes da aplicação.

O processo começa com a aquisição das imagens de entrada, seguindo para a definição da ROI, pré-processamento dos dados de entrada, segmentação da imagem, processamento da imagem binária resultante da subtração do *background*, prossegue com a detecção dos veículos em movimento e a quantificação destes, e por fim, tem-se como saída, as imagens resultantes da análise e processamento, os objetos identificados e o número de veículos detectados pelo algoritmo.

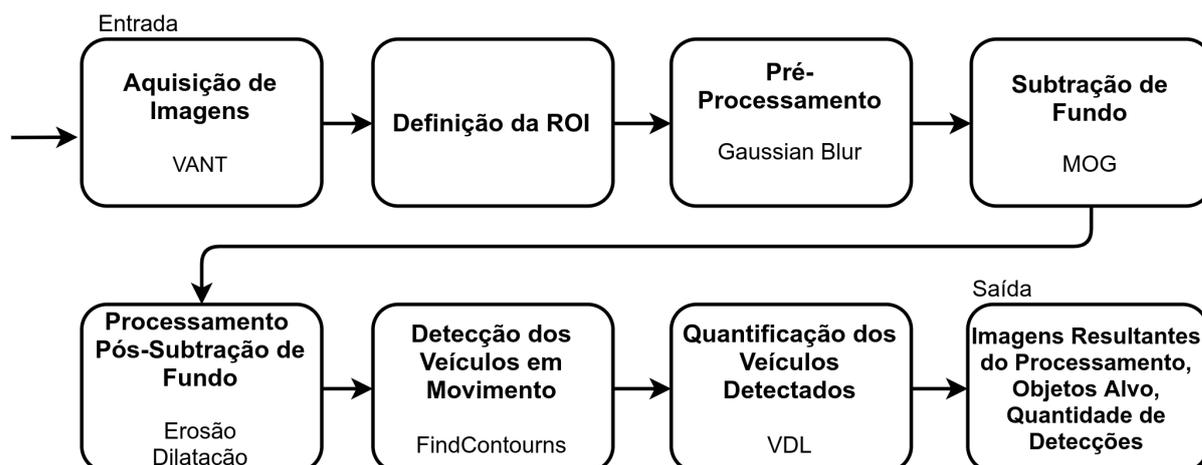


Figura 4.1 – Representação dos processos envolvidos no desenvolvimento da aplicação proposta, demonstrando as etapas utilizadas para que o protótipo de sistema alcançasse o objetivo. (Autoria própria)

Nas próximas seções do presente capítulo, serão apresentadas as técnicas utilizadas em cada etapa indicada na figura 4.1 para a implementação do algoritmo de identificação de veículos em movimento em imagens aéreas com PDI e Visão Computacional.

4.2 Ferramentas

Esta seção expõe as ferramentas utilizadas para o desenvolvimento do protótipo deste trabalho. A tabela 4.1 indica as principais tecnologias utilizadas.

Tabela 4.1 – Ferramentas utilizadas para a implementação do protótipo.

Descrição	Tecnologia
Biblioteca de Visão Computacional	OpenCV 3.0.2
Linguagem de Programação	Python 2.7
Sistema Operacional	Linux (Ubuntu 16.04)
Versão do Kernel	4.4.0

OpenCV (*Open Source Computer Vision Library*) é uma biblioteca na área de Visão Computacional, com código aberto para fins acadêmicos e comerciais que possui interfaces C++, C, Python e Java e suporta Windows, Linux, Mac OS, iOS e Android. Esta tecnologia foi definida como a biblioteca de visão computacional no desenvolvimento do protótipo.

De acordo com Cunha (2013), a biblioteca OpenCV é muito utilizada por órgãos governamentais, empresas e grupos de pesquisas em sistemas desenvolvidos com

finalidade de reconhecer intrusos, efetuar o monitoramento de minas, detecção de acidentes e afogamentos e identificação de buracos em rodovias, por exemplo.

A linguagem de programação definida para o desenvolvimento da aplicação foi Python, que é uma das linguagens que o OpenCV suporta e que foi determinada de acordo com a eficiência e portabilidade apresentada nos trabalhos correlatos pesquisados, conforme o capítulo 3.

4.3 Captura das Imagens de Entrada

Para a aquisição das imagens aéreas utilizadas na produção da aplicação experimental foi utilizado um VANT, com voo efetuado no período da manhã, em um dia com clima favorável para os objetivos da aplicação, isto é, com pouca influência da iluminação solar e ventos fracos, para que as imagens não fossem influenciadas por estes elementos externos. A tabela 4.2 apresenta breves informações sobre o voo realizado.

Tabela 4.2 – Dados sobre a captura das imagens

Descrição	Dados
VANT	Drone Solo 3DR
Câmera	GoPro 4
Clima	Nublado
Horário	8:00 a.m.
Altura	35 metros

O vídeo foi obtido com resolução *Full* HD e altura mediana, de acordo com as necessidades da ferramenta, pois imagens distantes e com baixa resolução, influenciam negativamente a análise e processamento das cenas para a detecção.

4.4 Definição da Região de Interesse

Durante o desenvolvimento da aplicação foi identificado que, devido a instabilidade do vídeo em alguns trechos, elementos do *background* que não são componentes em movimento, estavam sendo detectados como objetos em deslocamento pela aplicação da subtração de fundo, gerando resultados falso-positivos de identificação de veículos. Deste modo, em consequência destas adversidades, tornou-se necessário a definição de uma ROI, para que o reconhecimento de veículos da aplicação resultasse em valores correspondentes a realidade.

A definição de uma ROI também ajuda a diminuir a quantidade de informações a serem processadas, proporcionando melhor desempenho da aplicação, devido ao foco na análise das regiões importantes para o objetivo do sistema.

Neste trabalho, foram definidas no vídeo original três ROIs, sendo as regiões 2 e 3 provenientes da região 1. Essa separação de ROIs foi aplicada para validar a quantidade de veículos identificados nas cenas, pois a soma dos valores resultantes das ROIs 2 e 3, deve ser equivalente ao valor da região 1, visto que são locais subsequentes a região inicial (ROI 1).

Dessa forma, a determinação de diferentes ROIs em pontos específicos do vídeo, proporciona não apenas a identificação dos veículos nestas áreas, mas também uma forma de gerar informações a partir das imagens, com a validação dos resultados de cada ROI, isto é, partindo da afirmação que a soma dos valores resultantes das ROIs 2 e 3 é equivalente ao resultado da ROI 1, são geradas informações sobre o fluxo de carros que transitam para a avenida e os que prosseguem para a rotatória, dados que poderão alimentar sistemas de simulação de trânsito para ajudar na tomada de decisão.

A figura 4.2 demonstra as ROIs definidas na implementação deste trabalho. O retângulo azul é referente a ROI inicial, denominada "ROI 1", o vermelho a "ROI 2" e o verde a "ROI 3".



Figura 4.2 – Representação das ROIs utilizadas para a execução e validação do protótipo. a) Região inicial (ROI 1), indicada com um retângulo na cor azul. b) Região 2 (ROI 2), indicada com um retângulo na cor vermelha. c) Região 3 (ROI 3), indicada com um retângulo verde. (Autoria própria)

A figura 4.3, representa um *frame* de cada ROI aplicada na validação do algoritmo. Os resultados de detecção em cada região estão reportados no Capítulo 5.

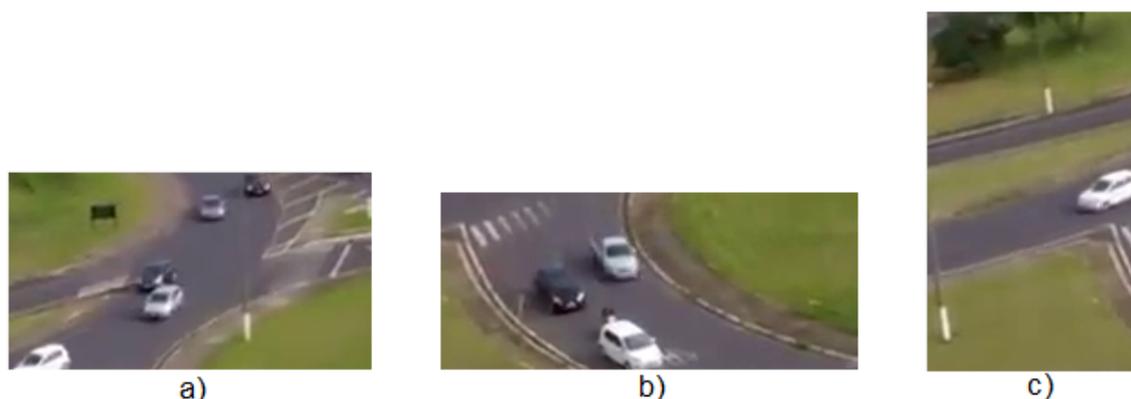


Figura 4.3 – a) Representação de um *frame* da ROI 1. b) Representação de um *frame* da ROI 2. c) Representação de um *frame* da ROI 3. (Autoria própria)

Após a definição da ROI, a análise das imagens de entrada é realizada somente na área interessante para a aplicação, diminuindo assim a detecção de falsos-positivos e a região de processamento.

4.5 Pré-Processamento das Imagens

A Etapa de “Pré-Processamento das Imagens” tem como função principal realizar o tratamento inicial nas imagens, aplicando filtros que possam diminuir os ruídos e suavizar as imagens do vídeo de entrada, para assim, proporcionar melhor desempenho das técnicas de Segmentação e Morfologia Matemática aplicadas nas fases seguintes a esta.

Esta seção descreve o método utilizado na implementação do protótipo para preparar as cenas que serão processadas na detecção dos veículos em movimento.

4.5.1 Filtro de Gaussian

De acordo com Parolin (2011), as informações provenientes dos dispositivos utilizados para a aquisição de imagens são influenciadas por ruídos devido a iluminação, qualidade do hardware e outros fatores. Dessa forma, para diminuir os impactos dos ruídos em um sistema, filtros de suavização são aplicados a fim de melhorar os resultados dos processos principais para o objetivo da aplicação.

Neste trabalho, o método de filtragem **Gaussiano** foi utilizado na fase de **Pré-Processamento** das imagens, para suavizar os ruídos existentes no vídeo original e proporcionar melhores resultados na aplicação das técnicas de segmentação, erosão e dilatação. O método tem como objetivo remover o conteúdo de alta frequência (ruídos e bordas) da imagem.

Uma das principais técnicas disponibilizadas pela biblioteca OpenCV que foi utilizada no desenvolvimento deste projeto é a “*Gaussian Blurring*” e é realizada através da função `cv2.GaussianBlur()` em Python.

A figura 4.4 demonstra a aplicação do método de *Gaussian Blur* em um *frame* do vídeo original, nesta, é possível verificar que a imagem resultante fica um pouco “embaçada”, devido a suavização e remoção das bordas que a técnica realiza.



Figura 4.4 – a) *Frame* original do vídeo utilizado no desenvolvimento do trabalho. b) *Frame* resultante da aplicação do método de *Gaussian Blur*. (Autoria própria)

Após o Pré-Processamento dos dados de entrada, que prepara as imagens para as próximas etapas, as cenas seguem para a fase Subtração de Fundo, descrita na seção 4.6.

4.6 Subtração de Fundo

A etapa de **Subtração de Fundo** é uma das etapas mais importantes na implementação, pois influencia diretamente nos resultados da identificação dos objetos alvo da aplicação.

Conforme descrito em 2.3, a subtração de fundo é um dos principais métodos para se iniciar o processamento de imagens direcionados para a identificação de objetos em imagens e consiste, basicamente, na remoção do *background* da imagem a fim de deixar em evidência apenas os componentes da cena (*foreground*).

Neste trabalho, a subtração de fundo foi realizada para deletar o fundo do vídeo e deixar apenas os objetos em movimento (veículos) em evidência para o reconhecimento destes itens. Para realizar este procedimento no vídeo adquirido, foi definida a utilização do método *Mixture of Gaussians* (MOG) de subtração, disponível na biblioteca OpenCV.

A figura 4.5 representa um *frame* do vídeo original utilizado, e a figura 4.6 mostra a aplicação do método MOG no vídeo utilizado para o desenvolvimento da ferra-

menta. A aplicação do MOG tem como retorno uma imagem binária, com a remoção do fundo estático na cor preta e os objetos em movimento na cor branca.



Figura 4.5 – Representação de um *frame* do vídeo original utilizado no desenvolvimento do protótipo de sistema. (Autoria própria)



Figura 4.6 – Imagem binária de representação de um *frame* com aplicação do método MOG de subtração de fundo. (Autoria própria)

Após a subtração de fundo utilizando MOG, pode-se perceber que além de deixar em evidência os veículos em movimento na cena, alguns itens do ambiente

também ficam expostos no resultado da segmentação. Dessa forma, para remover esses detalhes não interessantes para o sistema proposto, são aplicados métodos de morfologia matemática, a fim de apagar os resquícios da subtração de fundo e realçar os objetos em movimento. As funções utilizadas são as de erosão e dilatação, conforme descritas na seção 2.4 do capítulo 2.

Os tópicos seguintes 4.7.1 e 4.7.2 detalham os procedimentos do uso das técnicas de Erosão e Dilatação neste trabalho.

4.7 Processamento Pós-Subtração de Fundo

A etapa de **Processamento Pós-Subtração de Fundo** se refere aos métodos aplicados nas imagens após o processo de remoção do *background*, com intuito de deletar os fragmentos resultantes da segmentação e realçar os objetos em movimento na imagem binarizada derivada da etapa anterior.

As técnicas utilizadas nesta etapa são as de Erosão e Dilatação, que serão expressas nas seções 4.7.1 e 4.7.2, respectivamente.

4.7.1 Erosão

Conforme descrito em 2.5.1, o processo de Erosão em uma imagem é importante para deletar elementos não interessantes para a aplicação e, assim, aumentar a precisão nas detecções. Após a aplicação da técnica de subtração de fundo, retratada na seção 4.6, é realizado o processo de Erosão na imagem resultante da etapa de subtração do *background*.

A figura 4.7 demonstra a aplicação do método de Erosão na imagem resultante da subtração de fundo, nesta, é possível verificar que grande parte dos ruídos existentes na cena foram deletados, resultando em uma imagem mais "limpa" com foco nos principais elementos em movimento.



Figura 4.7 – a) Frame resultante da subtração de fundo com MOG, conforme exposto na seção 4.6. b) Frame resultante da aplicação do método de Erosão após a subtração de fundo. (Autoria própria)

Conforme 2.5, o Elemento Estruturante utilizado na aplicação da operação de Erosão foi o padrão (Elemento Estruturante Quadrado).

Aplicar a Erosão nas imagens de subtração de fundo é importante para diminuir detecções falso-positivas ao final da aplicação, pois a técnica ajuda a excluir da cena os itens não relevantes para o objetivo do sistema.

Após o procedimento de Erosão, a imagem resultante desta técnica será utilizada para o tratamento de Dilatação, conforme representado na próxima seção 4.7.2.

4.7.2 Dilatação

Conforme detalhado na seção 2.5.2, a dilatação se refere ao processo de “aumentar” os componentes em evidência em uma imagem binária, agrupando os pixels próximos para se chegar a um padrão desejado. Neste trabalho, a técnica foi utilizada após a operação de Erosão, para realçar os objetos em movimento na cena e assim, possibilitar a detecção dos veículos.

A figura 4.8 retrata a aplicação do método de dilatação da biblioteca OpenCV na imagem resultante da Erosão indicada no item anterior 4.7.1, sendo possível verificar o realce dos objetos em movimento.

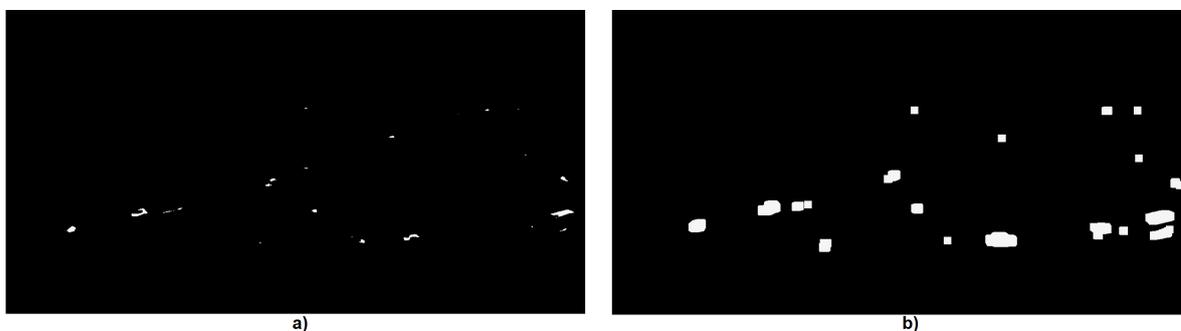


Figura 4.8 – a) Frame representando a imagem resultante da técnica de Erosão aplicada. b) Frame resultante da aplicação do método de Dilatação na imagem. (Autoria própria)

Conforme 2.5, o Elemento Estruturante utilizado na aplicação da Dilatação foi o padrão (Elemento Estruturante Quadrado).

Com a aplicação da operação morfológica de Dilatação, os componentes em movimento são evidenciados na imagem binária, facilitando a detecção dos veículos pelas técnicas utilizadas nas etapas posteriores.

4.8 Detecção de Objeto em Movimento

A Etapa de **Detecção de Objeto em Movimento**, consiste na aplicação de métodos que reconheçam os componentes na cor branca da imagem binária como objetos em movimento. As técnicas para a detecção são realizadas utilizando a imagem binária derivada das etapas anteriores.

Para a detecção dos veículos em movimento, a partir da imagem binária resultante das etapas de Pré Processamento, Subtração de Fundo e Processamento Pós-Subtração de Fundo (aplicação de operações de morfologia matemática), foi utilizada a função `cv2.findContours()` do OpenCV, que consiste em encontrar uma curva que une pontos contínuos que tem a mesma cor ou intensidade. Neste projeto, será utilizada para reconhecer os veículos em movimento formados pela cor branca com fundo preto na imagem binária derivada das etapas anteriores.

Após encontrar grupos de pontos de cor branca na imagem, que são os veículos, foram inseridos retângulos ao redor das detecções utilizando o método “*Straight Bounding Rectangle*”, que está correlacionado a técnica “`cv2.findContours()`” e é aplicado pela função “`cv2.boundingRect()`”. Para desenhar o retângulo de coloração verde nos objetos identificados, foi utilizada a função “`cv2.rectangle()`”.

A partir da inserção de um retângulo como forma de identificar as detecções, foi utilizado o método “**Moments**”, que possibilita o cálculo de algumas características como a área de uma massa de objeto e o centro do componente. Nesta proposta, a função foi utilizada para o cálculo do centro do objeto, e é aplicada através de “`cv2.moments()`”.

Para a exibição do centroide do veículo detectado, foi inserido um círculo de cor preta, aplicando a função “`cv2.circle()`”. A figura 4.9 demonstra o emprego das funções `cv2.findContours()`, `cv2.boundingRect()`, `cv2.rectangle()` e `cv2.circle()` na ROI 1.



Figura 4.9 – Representação de um *frame* que demonstra a aplicação das funções `cv2.findContours()`, `cv2.boundingRect()`, `cv2.rectangle()` e `cv2.circle()` para a detecção dos veículos em movimento. (Autoria própria)

Com a aplicação das técnicas descritas na presente seção, foi possível identificar os veículos em movimento no vídeo, utilizando as imagens provenientes das etapas de Pré-Processamento, Subtração de Fundo e Processamento Pós-Subtração de Fundo, e representá-los no *frame* original do vídeo, conforme apresentado na figura 4.9, que apresenta as técnicas aplicadas na ROI 1.

4.9 Contagem dos Veículos

A Etapa de **Contagem dos Veículos** tem como objetivo efetuar a soma dos veículos que transitam em uma via pública. Para o procedimento foi inserida uma **VDL** (Linha de Detecção Virtual) para que os objetos em movimento sejam quantificados ao passar sobre a linha.

A figura 4.10 representa a reta inserida para a detecção dos veículos que por ela atravessarem.

Com o centroide dos objetos em movimento determinados, conforme exposto na seção 4.8, é possível efetuar a comparação entre o valor “y” do centroide, com o valor “y” da “Linha de Detecção”, isto é, para que um veículo seja identificado, é utilizado um contador simples que é incrementado sempre que o valor da coordenada “y” do ponto (x,y) do centroide do veículo detectado tenha valor equivalente ao valor “y” da VDL determinada.

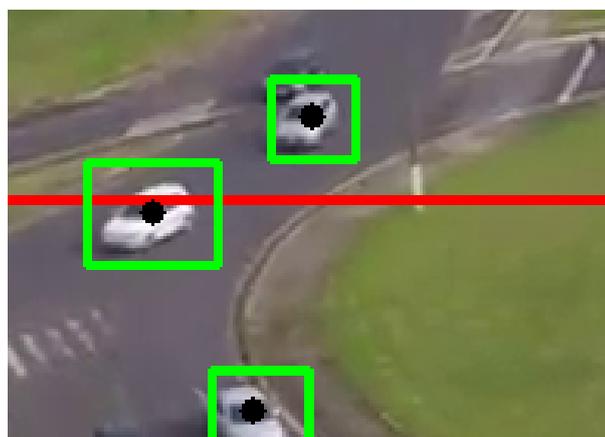


Figura 4.10 – VDL representada em um *frame* da ROI 1, na cor vermelha. (Autoria própria)

A contagem é efetuada somente nos objetos em movimento que cruzam a VDL. As linhas de detecção tem os parâmetros ajustados para cada ROI, devido a diferença no ângulo de cada rua no vídeo original.

4.10 Conclusões do Capítulo

Este capítulo expôs os detalhes do processo de implementação realizado no desenvolvimento do protótipo de sistema proposto neste documento, apresentando as etapas definidas para o projeto, assim como a função e importância de cada técnica utilizada para alcançar o objetivo da aplicação.

Durante a implementação do sistema foram identificadas algumas complexidades acerca dos processos de reconhecimento dos veículos em deslocamento nas cenas aéreas, em sua maioria causados por componentes do ambiente que influenciam diretamente na integridade dos resultados da aplicação.

Alguns itens identificados foram os listados a seguir:

- Falhas na Subtração de Fundo;
- Influência de elementos do ambiente como árvores, sol, vento, monumentos, etc;
- Instabilidade da câmera;
- Ângulo e distância da filmagem;
- Proximidade entre dois ou mais veículos.

No reconhecimento dos veículos, as falhas na segmentação influenciaram na detecção, devido ao realce em elementos do ambiente que não estavam realmente em movimento. A instabilidade da câmera em alguns momentos do vídeo também causou o mesmo problema nos resultados, onde falso-positivos foram detectados como objetos em deslocamento.

Os elementos do ambiente também provocaram algumas adversidades, como por exemplo árvores, vento, sol, monumentos, entre outros. Esses fatores impactam diretamente na integridade dos resultados da aplicação final.

A distância da filmagem pode influenciar na subtração de fundo e nas operações de morfologia matemática, pois em imagens muito distantes, os objetos em movimento ficam pouco evidentes na subtração do *background* e podem acabar sendo “perdidos” na operação de Erosão, por exemplo. De acordo com o ângulo de filmagem da rua ou avenida monitorada, as detecções podem ser favorecidas ou prejudicadas, vídeos mais verticais em relação a rua, por exemplo, proporcionam melhores resultados que cenas na diagonal.

A proximidade dos veículos ocasionou um conflito nos dados resultantes do sistema, ocasionando a detecção de dois veículos como apenas um, nos casos onde a distância entre eles é muito curta.

No capítulo 5 serão expostos os resultados obtidos na execução do algoritmo aplicado nas três ROIs definidas, validando as detecções em cada caso e efetuando uma avaliação geral do funcionamento do protótipo.

Capítulo 5

RESULTADOS

Este Capítulo tem intenção de descrever os resultados obtidos com a execução da ferramenta experimental desenvolvida, e exposta no Capítulo 4, divulgando os resultados adquiridos pela aplicação.

Conforme representado na seção 4.4, foram determinadas três diferentes ROIs para a validação da aplicação, sendo a segunda e terceira ROI derivadas da ROI inicial, de forma proposital, para que a comparação nos resultados provenientes de cada caso pudesse ser realizada.

Deste modo, os testes realizados para a validação do algoritmo foram baseados nos resultados de cada ROI, onde a soma dos valores resultantes da execução com as regiões 2 e 3 deve ser equivalente ao valor retornado pela execução com a ROI 1, visto que os veículos de saída da ROI 1 possuem apenas duas opções de caminho, um que passa pela ROI 2 e outro pela ROI 3. Dessa forma, podemos definir:

$$QtdVeiculos_ROI1 = QtdVeiculos_ROI2 + QtdVeiculos_ROI3 \quad (5.1)$$

Onde **QtdVeiculos_ROI1** é a quantidade de veículos retornada na execução do algoritmo desenvolvido com a ROI 1, **QtdVeiculos_ROI2** é a quantidade de veículos retornada pela aplicação utilizando a ROI 2 e **QtdVeiculos_ROI3** é o valor resultante do sistema para a ROI 3.

Na figura 5.1 é possível verificar os possíveis acessos dos veículos que transitam pela ROI 1.

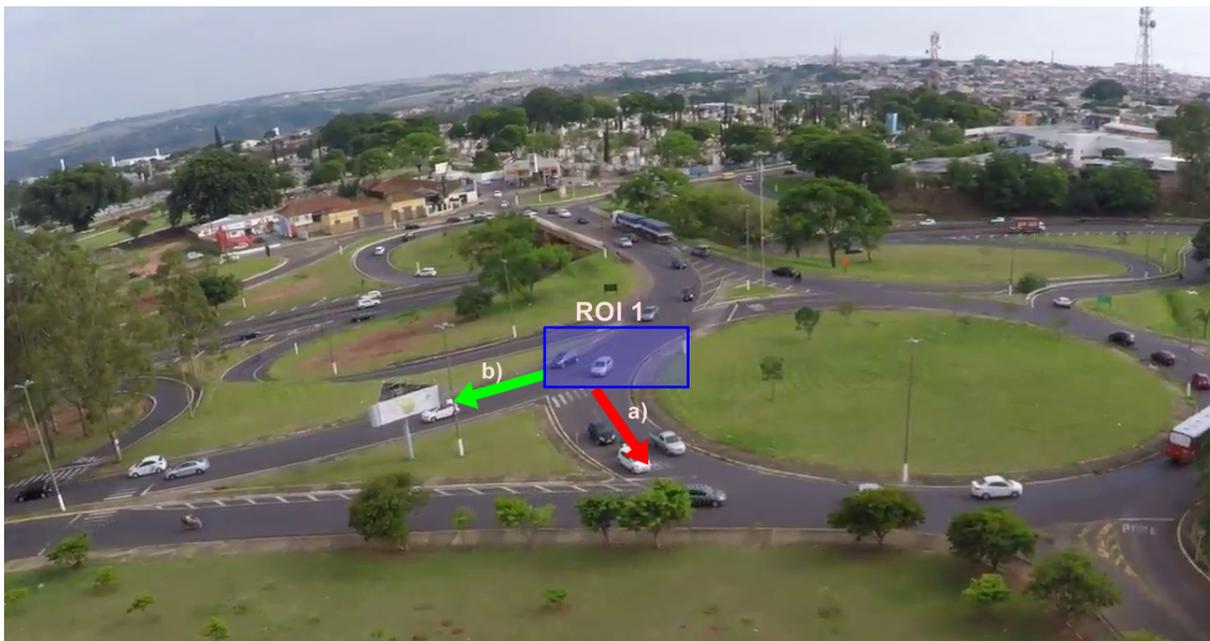


Figura 5.1 – a) Representação de um possível percurso originário da ROI 1. b) Representação de outro caminho possível originário da ROI 1. (Autoria própria)

O caminho “a” indicado na figura 5.1 se refere a ROI 2, onde os veículos prosseguem para a rotatória (flecha de cor vermelha), enquanto o acesso “b” para a avenida é referente a ROI 2 (flecha de cor verde), ambas originadas da região inicial ROI 1.

Para a validação dos resultados, foi utilizado um pequeno trecho do vídeo original que tem de 26 segundos de duração. As seções 5.1, 5.2 e 5.3 apresentam os testes realizados em cada ROI e os resultados obtidos em cada validação.

5.1 Região de Interesse 1

A execução do algoritmo implementado na ROI 1 retornou 17 veículos detectados na contagem, porém, através de análise visual, foi possível verificar que ocorreram influências no reconhecimento, onde dois veículos foram detectados como apenas um, devido a curta proximidade entre os objetos em movimento. O valor real de carros que cruzaram a VDL definida foi de 18 veículos.

Com o movimento da câmera em alguns momentos, foi possível verificar que alguns elementos do ambiente foram detectados como objetos em movimento, porém, como não atravessam a VDL, não foram quantificados.

A figura 5.2 apresenta um *frame* das imagens resultantes do algoritmo de detecção de veículos implementado.



Figura 5.2 – Representação de um *frame* da execução do algoritmo implementado na ROI 1. (Autoria própria)

5.2 Região de Interesse 2

Na aplicação do algoritmo na ROI 2, os parâmetros da VDL foram alterados para que a linha fosse posicionada no início da ROI, a fim de que os veículos que estavam previamente na rotatória, não fossem detectados como originários da ROI 1.

Nesta execução, o sistema retornou resultados compatíveis com a realidade, sendo 4 veículos detectados e quantificados, mas foi possível perceber que o método de subtração de fundo utilizado (MOG), tem problemas para realizar a incorporação de objetos parados ao modelo de fundo, no entanto, o caso não influenciou no resultado da aplicação.



Figura 5.3 – Representação de um *frame* da execução do algoritmo implementado na ROI 2. (Autoria própria)

5.3 Região de Interesse 3

Para a execução na ROI 3, foi necessário alterar alguns parâmetros da VDL, pois a rua presente nas cenas da ROI 3 tem angulação diferente das imagens das ROIs 1 e 2, sendo necessário inserir a linha de detecção verticalmente, diferentemente das retas horizontais adicionadas nas outras regiões.

A figura 5.4 apresenta a aplicação da VDL de forma perpendicular a rua monitorada na execução do algoritmo na ROI 3.

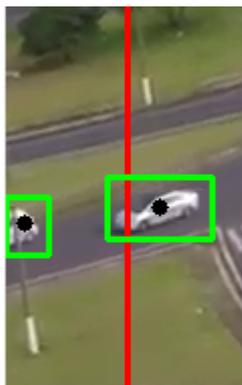


Figura 5.4 – Representação de um *frame* da execução do algoritmo implementado na ROI 2. (Autoria própria)

5.4 Comparação dos Resultados

A tabela 5.1 apresenta os valores reais (quantidade de veículos existentes na cena em cada ROI) e os valores resultantes da aplicação. Os valores reais foram obtidos efetuando a contagem “manual” dos veículos detectados, de acordo com identificação visual.

Tabela 5.1 – Validação dos resultados retornados pela aplicação nas três ROIs.

ROI	VALORES REAIS	RESULTADOS DA APLICAÇÃO	PRECISÃO (%)
ROI 1	18	17	94.44
ROI 2	4	4	100.00
ROI 3	14	11	78.57

Com os resultados obtidos é possível verificar que a aplicação experimental não fornece resultados cem por cento compatíveis com a realidade, porém, gera valores próximos e relevantes, a tabela 5.1 apresenta a taxa de precisão dos resultados de cada ROI. Foi identificado que nem todos os veículos que ultrapassam a **VDL** são reconhecidos, pois, em alguns casos, quando dois ou mais carros estão muito próximos, são detectados como apenas um objeto alvo, provocando retornos de dados divergentes.

A validação dos resultados foi realizada com a contagem visual dos veículos presentes no vídeo utilizado para os testes da ferramenta. A partir dos resultados e dificuldades encontrados, foram identificados ajustes, tendências e melhorias a respeito do uso de PDI para a detecção de veículos em movimento em ambientes externos.

5.5 Conclusões do Capítulo

Este Capítulo apresentou os resultados obtidos com os testes de validação do algoritmo aplicado nas três diferentes ROIs definidas no projeto.

O Capítulo 6 traz as conclusões finais do projeto, descrevendo as vantagens, desvantagens e melhorias identificadas na produção do trabalho proposto neste documento.

Capítulo 6

CONCLUSÃO

O presente capítulo tem finalidade de expor uma análise sobre a produção do trabalho e da ferramenta experimental, com intuito de destacar os objetivos alcançados, as dificuldades enfrentadas e as melhorias idealizadas no projeto.

6.1 Conclusões

Neste trabalho foi realizado o estudo e implementação de técnicas de processamento de imagens e de visão computacional direcionadas para a detecção de veículos em movimento em uma via pública, utilizando cenas aéreas captadas com o uso de VANTs. O intuito da proposta é que os resultados obtidos possam ser integrados a sistemas de simulação de soluções com finalidade de auxiliar no controle e organização do trânsito.

O algoritmo de reconhecimento de veículos em movimento através de imagens aéreas implementado neste projeto, é uma solução apresentada como forma de obter informações relevantes sobre o fluxo de veículos nas ruas e, assim, contribuir para a elaboração de propostas de planos urbanos.

Esta proposta de solução não resolve problemas mais complexos relacionados ao domínio do trabalho, como por exemplo o suporte a mudanças climáticas nas imagens (chuva, sol intenso, ventos fortes, etc), detecção de veículos de duas ou três rodas (motos, bicicletas, triciclos, etc), processamento utilizando dispositivos de aquisição de imagens em movimento, levantamento de informações sobre a localização e processamento em tempo real.

A aplicação foi desenvolvida utilizando um vídeo obtido em situações climáticas consideravelmente favoráveis para o objetivo da aplicação, captando as imagens em um horário e região de médio movimento veicular, com tempo nublado, sem muito vento e com pouco sol para não influenciar na iluminação e no processamento das

cenar.

Dessa forma, foi possível realizar a validação dos resultados da aplicação experimental desenvolvida com os valores reais da cena, conforme proposto nos objetivos específicos do trabalho. A comparação dos resultados foi efetuada com a contagem “manual” dos veículos existentes no vídeo, de acordo com a verificação visual das cenas em três diferentes ROIs.

A contagem dos veículos em diferentes regiões proporcionou não apenas a quantificação dos veículos que transitavam pela área, como também a produção de informações relevantes sobre o fluxo de veículos, onde foi possível determinar, a partir da análise e processamento das imagens, um fluxo aproximado de veículos que trafegavam em direção a avenida e que prosseguiram para a rotatória. Isto é, com a aplicação das técnicas de PDI e Visão Computacional foi possível obter dados relevantes sobre o fluxo de veículos através das imagens.

Referente a proposta do trabalho de forma geral, conclui-se que a obtenção de dados sobre o fluxo de veículos em movimento é relevante, devido a facilidade de aplicação em ambientes externos e baixo custo, sem a necessidade do desenvolvimento de um sensor específico, já que podem ser utilizadas desde câmeras convencionais até as mais robustas que fornecem imagens em alta resolução. Porém, com a implementação do projeto foi possível identificar que vários elementos podem influenciar nos resultados, como a existência de árvores e monumentos que ocultam os objetos de interesse, a instabilidade da câmera que causa a detecção de resultados falso-positivos, a importância da definição de uma ROI, para que seja considerada apenas a região onde se encontram os objetos alvo e a necessidade da alteração de alguns parâmetros, conforme a ROI utilizada.

Apesar dessas questões referentes as influências de itens em ambientes não controlados no PDI, o método proposto apresenta competências pertinentes na obtenção de informações sobre o fluxo de veículos em movimento a partir de imagens aéreas e pode ser futuramente melhorado, de acordo com as complexidades e tendências identificadas neste trabalho.

Na seção 6.2, serão expostas ideias e sugestões para trabalhos futuros utilizando esta proposta como ponto de partida.

6.2 Trabalhos Futuros

De acordo com os resultados obtidos no desenvolvimento do trabalho e implementação do protótipo de sistema, foram identificadas melhorias e dificuldades que podem ser tratadas em projetos futuros. Como sugestão recomenda-se considerar o

investimento em propostas com finalidade de expor soluções a respeito do tratamento de questões relacionadas ao clima, pois o estado das imagens interferem diretamente nos resultados da identificação dos veículos, tornando a fase de pré-processamento das cenas uma etapa de grande importância para o sucesso da aplicação.

Também é interessante trabalhar em soluções para a detecção de motocicletas e outros meios de transporte, além de veículos de quatro rodas apenas, para assim, fornecer maiores informações sobre o trânsito.

O fornecimento de informações adicionais como a localização da região monitorada, utilizando GPS (*Global Positioning System* - Sistema de Posicionamento Global) e o processamento dos dados em tempo real também são ideias consideráveis para posteriores aplicações.

REFERÊNCIAS

BIANCHI, M. F. de. *Extração de características de imagens de faces humanas através de wavelets, PCA e IMPCA*. São Carlos: [s.n.], 2006. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/18/18133/tde-10072006-002119/>>.

CASTLEMAN, K. R. *Digital Image Processing*. [S.l.]: Prentice Hall, 1996. ISBN 0-13-211467-4.

COELHO, A. G. M. S. *Localização de pessoas em cenários de Assisted Living*. Dissertação (Mestrado) — Universidade do Porto, 2014.

CUNHA, A. L. B. N. da. *Sistema Automático para Obtenção de Parâmetros Do Tráfego Veicular a Partir de Imagens de Vídeo Usando OpenCV*. 2013. 128 p. Disponível em: <<https://www.google.com/url?q=http://www.teses.usp.br/teses/disponiveis/18/18144/tde-19112013-165611/publico/ANDRECUNHA.pdf{&}sa=U{&}ved=0ahUKEwiVir6DosbKAhWJdR4KHVGbD5cQFggKMAM{&}client=internal-uds-cse{&}usg=AFQjCNGInzz6MNENwDsqVk>>.

FACON, J. *A Morfologia Matemática e suas Aplicações em Processamento de Imagens*. 2011. 61–128 p.

FERREIRA, C. S. *Implementação do Algoritmo de Subtração de Fundo para Detecção de Objetos em Movimento, Usando Sistemas Reconfiguráveis*. 2012.

FILHO, O. M.; NETO, H. V. *Processamento Digital de Imagens*. edição. Rio de Janeiro: Brasport, 1999. ISBN 8574520098.

GONZALEZ, R. C.; WOODS, R. E. *Processamento Digital de Imagens*. 3ª. ed. São Paulo: Pearson Prentice Hall, 2010. ISBN 978-85-8143-586-2.

HIGASHINO, W. A. *Estudo comparativo de algoritmos de subtração de fundo em sequencias de imagens*. Dissertação (Mestrado) — Universidade Estadual de Campinas, 2006.

HORN, B. K. P.; SCHUNCK, B. G. Determining optical flow. *Artif. Intell.*, v. 17, n. 1-3, p. 185–203, 1981. Disponível em: <[http://dx.doi.org/10.1016/0004-3702\(81\)90024-2](http://dx.doi.org/10.1016/0004-3702(81)90024-2)>.

JUNIOR, J. T. *O Monitoramento veicular utilizando câmeras de segurança convencionais e visão computacional*. 2016.

MARCONATO, E. A. *Modelo de Arquitetura em Camadas para Interconexão de Sistemas em VANT*. Tese (Doutorado) — USP - São Carlos, 2017.

MARIA, G. et al. A drone-based image processing system for car detection in a smart transport infrastructure. In: *Proceedings of the 18th Mediterranean Electrotechnical Conference: Intelligent and Efficient Technologies and Services for the Citizen, MELECON 2016*. [S.l.: s.n.], 2016. ISBN 9781509000579.

MOHAMMED, F. et al. UAVs for Smart Cities : Opportunities and Challenges. In: . [S.l.: s.n.], 2014. ISBN 9781479923762.

PAROLIN, A. *Segmentação de Imagens de Pessoas em Tempo Real para Videoconferências*. 2011.

QIU, S.; WEN, G.; FAN, Y. Occluded Object Detection in High-Resolution Remote Sensing Images Using Partial Configuration Object Model. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, v. 10, n. 5, p. 1909–1925, may 2017. ISSN 1939-1404. Disponível em: <<http://ieeexplore.ieee.org/document/7855669/>>.

QUEIROZ, J. E. R. de; GOMES, H. M. Introdução ao processamento digital de imagens. *Revista de Informática Teórica e Aplicada - RITA*, v. 8, 2001. Disponível em: <<http://files.multimedia-unicv.webnode.com/200000006-16be917b69/Rita-Tutorial-PDI.pdf>>.

RIBEIRO, A. R. M.; CORREIA, P. L. Automatic Vehicle Counting based on Surveillance Video Streams. In: *Congresso do Comité Português da URSI*. Portugal: [s.n.], 2014. p. 1–10.

SILVA, B. R. d. A. *Sistema de contagem automática de objetos utilizando processamento digital de imagens em dispositivos móveis*. 2014.

VIOLA, P.; JONES, M. Rapid Object Detection using a Boosted Cascade of Simple Features. In: . [S.l.: s.n.], 2001. ISBN 0769512720.

Apêndice A

CÓDIGO FONTE DO ALGORITMO

Este Apêndice contém o código fonte do algoritmo produzido para a identificação de veículos em movimento a partir de imagens aéreas. O código é representado na linguagem de programação Python utilizando a biblioteca OpenCV na implementação das técnicas de processamento de imagens direcionadas para o reconhecimento de veículos em movimento.

```
1  #!/bin/python
2  #coding: utf-8
3
4  import numpy as np
5  import cv2
6  import math
7
8  #Carrega o vídeo de entrada da aplicação
9  video = cv2.VideoCapture('video-tc2.mp4')
10
11 #Método de Subtração de Fundo (MOG)
12 subFundo = cv2.bgsegm.createBackgroundSubtractorMOG()
13
14 #Inicia contador com zero
15 contador = 0
16
17 while(1):
18
19     # Lê o primeiro frame do vídeo
20     ret, frame = video.read()
21
```

```
22     #Define Região de Interesse (ROI)
23     roi = frame[250:450,600:800] #Região de Interesse 1
24
25     #roi = frame[400:500,570:800] #Região de Interesse 2
26
27     #roi = frame[300:500,480:600] #Região de Interesse 3
28
29     #Dimensões do vídeo
30     height = np.size(roi,0)
31     width = np.size(roi ,1)
32
33
34     if ret == True:
35
36         #Aplica a escala de cinza na imagem
37         imagemCinza = cv2.cvtColor(roi, cv2.COLOR_BGR2GRAY)
38
39         #Aplica o método de suavização "GaussianBlur" para
40         ↪ diminuir os ruídos
41         gaussian = cv2.GaussianBlur(imagemCinza, (15, 15), 0)
42
43         #Aplica a subtração de fundo na imagem
44         imgBinaria = subFundo.apply(gaussian)
45
46         kernel = np.ones((5, 5), np.uint8)
47
48         #Aplica o método de Morfologia Matemática de Erosão
49         #erosao = cv2.erode(imgBinaria, kernel, iterations=1)
50
51         #Aplica o método de Morfologia Matemática de
52         ↪ Dilatação
53         dilatacao = cv2.dilate(imgBinaria,kernel,iterations =
54         ↪ 3)
55
56         #erosao = cv2.erode(dilatacao, kernel, iterations=1)
57
58         #Aplica a função findContours para a detecção dos
59         ↪ objetos em movimento
```

```
56     _, contours, hierarchy = cv2.findContours(dilatacao,
57     ↪ cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
58
59     #Insere linha de detecção (VDL)
60     linhay = (height / 2) + 20
61
62     #linhay = (height / 2) - 20 #Linha para ROI2
63
64     #linhax = (width / 2) #ROI3
65     #cv2.line(roi, (linhax,0), (linhax,height), (255, 0, 0),
66     ↪ 2) #ROI3
67
68     cv2.line(roi, (0,linhay), (width,linhay), (0, 0,
69     ↪ 255), 2)
70
71     for c in contours:
72
73         #Insere um retângulo nos objetos em movimento
74         ↪ encontrados
75         (x, y, w, h) = cv2.boundingRect(c)
76         cv2.rectangle(roi, (x, y), (x + w, y + h), (0,
77         ↪ 255, 0), 2)
78
79         #determina o ponto central do contorno
80
81         momentos = cv2.moments(c)
82
83         xcontorno =
84         ↪ int(momentos['m10']/momentos['m00'])
85         ycontorno = int(momentos['m01']/momentos['m00'])
86
87         centroide = (xcontorno,ycontorno)
88
89         #Desenha um círculo noo ponto central
90         cv2.circle(roi, centroide, 1, (0, 0, 0), 5)
```

```
88         #Conta os veículos detectados, ao passar pela
89         ↪ linha de detecção e desenha um círculo
90         ↪ vermelho no ponto central do contorno
91
92         if (ycontorno == linhay):
93             contador +=1
94             cv2.circle(roi, centroide, 1, (0, 255,
95                 ↪ 0), 8)
96
97         #if (xcontorno == linhax): #ROI3
98             #contador +=1
99             #cv2.circle(roi, centroide, 1, (0, 0,
100                 ↪ 255), 8)
101
102         #cv2.imshow('dilatacao',dilatacao)
103         #cv2.imshow('erosao',erosao)
104
105         #cv2.imshow('imgBinaria',imgBinaria)
106         #cv2.moveWindow('dilatacao', 500, 230)
107
108         #Exibe o video original com as detecções
109         cv2.imshow('Resultado',roi)
110
111         # Ajusta as posição da janela
112         #cv2.moveWindow('Resultado', 100, 230)
113
114
115         k = cv2.waitKey(30) & 0xff
116         if k == 27:
117             break
118
119         #Print do valor de veículos detectados
120         print "Veículos detectados:"+str(contador)
121
```

```
122 video.release()  
123 cv2.destroyAllWindows()
```
