UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA

ISRAEL LAURENSI ROSA
RAFAEL JOSÉ GUIMARÃES
VÍTOR DE OLIVEIRA TOZZI

DETECÇÃO E RASTREAMENTO DE CICLISTAS EM VIAS PÚBLICAS POR MEIO DE TÉCNICAS DE VISÃO COMPUTACIONAL

MONOGRAFIA

CURITIBA

2016

ISRAEL LAURENSI ROSA
RAFAEL JOSÉ GUIMARÃES
VÍTOR DE OLIVEIRA TOZZI

DETECÇÃO E RASTREAMENTO DE CICLISTAS EM VIAS PÚBLICAS POR MEIO DE TÉCNICAS DE VISÃO COMPUTACIONAL

Monografia apresentada à disciplina de Trabalho de Conclusão de Curso do Departamento Acadêmico de Informática da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Bogdan Tomoyuki Nassu

CURITIBA

2016

Ministério da Educação
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
Câmpus Curitiba
Diretoria de Graduação e Educação Profissional
Departamento Acadêmico de Informática
Coordenação do Curso de Bacharelado em Sistemas de Informação

TERMO DE APROVAÇÃO

"DETECÇÃO E RASTREAMENTO DE CICLISTAS EM VIAS PÚBLICAS POR MEIO DE TÉCNICAS DE VISÃO COMPUTACIONAL"

por

"ISRAEL ANDRÉ LAURENSI ROSA, RAFAEL JOSÉ GUIMARÃES, VÍTOR DE OLIVEIRA TOZZI"

Este Trabalho de Conclusão de Curso foi apresentado às _____ do dia **06** de **dezembro** de **2016** como requisito parcial à obtenção do grau de Bacharel em Sistemas de Informação na Universidade Tecnológica Federal do Paraná - UTFPR - Câmpus Curitiba. O(a)(s) aluno(a)(s) foi(ram) arguido(a)(s) pelos membros da Banca de Avaliação abaixo assinados. Após deliberação a Banca de Avaliação considerou o trabalho _____.

_____ **Prof. Bogdan Tomoyuki Nassu** (Presidente - UTFPR/Curitiba)	_____ **Profa. Leyza Elmeri Baldo Dorini** (Avaliador 1 - UTFPR)
_____ **Prof. Gustavo Benvenutti Borba** (Avaliador 2 - UTFPR)	_____ **Profa. Leyza Elmeri Baldo Dorini** (Professor Responsável pelo TCC – UTFPR/Curitiba)
_____ **Prof. Leonelo Dell Anhol Almeida** (Coordenador(a) do curso de Bacharelado em Sistemas de Informação – UTFPR/Curitiba)	

"A Folha de Aprovação assinada encontra-se na Coordenação do Curso."

AGRADECIMENTOS

Primeiramente gostaríamos de agradecer por toda a ajuda do nosso professor Bogdan Tomoyuki Nassu, que nos ofereceu o tema e dedicou seu tempo para correções e reuniões, nos aconselhando durante todo o desenvolvimento do projeto. Agradecemos pela paciência e pelo auxílio, que foram cruciais para o resultado final.

Também gostaríamos de agradecer à professora Tatiana Gadda, que se fez presente para contribuir com a interdisciplinaridade desse projeto, tornando o trabalho muito mais interessante do ponto de vista social.

Agradecemos à Universidade Tecnológica Federal do Paraná pela oportunidade de formação e por toda a estrutura disponibilizada.

Agradecemos aos nossos colegas Rodolpho de Castro Alves e Lucas Cyulik, que forneceram o local e os aparatos necessários para a obtenção dos dados utilizados para o desenvolvimento do projeto e aplicação do experimento.

Também somos gratos a todos nossos outros colegas, amigos e professores, por todos os momentos vividos durante a graduação e todo o conhecimento adquirido nesses anos. Em especial, ao Thiago Matos, que sempre esteve ao nosso lado.

Israel Laurensi agradece a toda sua família pelo carinho e suporte, especialmente a sua mãe, Leda Laurensi, que tanto se fez presente na graduação de seu filho, dando-lhe conselhos e fazendo-o superar todos os momentos difíceis, mesmo quando não podia estar presencialmente ao seu lado.

Rafael José agradece à sua família, que sempre esteve presente em todos os momentos, principalmente sua mãe, Rosana José, que sempre o incentivou a dar o seu melhor em qualquer situação, e Julia Ulson Tretel, por toda a paciência em momentos difíceis e todo o apoio dado, tanto na escrita da monografia, quanto aconselhando e ouvindo sempre que necessário.

Vítor Tozzi agradece a sua família por todo o suporte e ensinamentos que lhe tornaram capaz de buscar essa conquista em sua vida, aos professores por todo o aprendizado no decorrer da graduação e aos amigos pelo apoio em vários momentos desta jornada.

RESUMO

ROSE, Israel Laurensi; Guimarães, Rafael José; TOZZI, Vítor de Oliveira. DETECÇÃO E RASTREAMENTO DE CICLISTAS EM VIAS PÚBLICAS POR MEIO DE TÉCNICAS DE VISÃO COMPUTACIONAL. 59 f. Monografia – Departamento Acadêmico de Informática, Universidade Tecnológica Federal do Paraná. Curitiba, 2016.

A utilização de bicicletas como meio de transporte vem chamando a atenção de cidades que buscam disponibilizar maneiras para que os ciclistas possam trafegar em segurança, em troca da utilização de um meio de transporte saudável e uma redução na emissão de poluentes produzidos por veículos motorizados. Diante disso, uma solução computacional foi desenvolvida com o objetivo de realizar uma análise quantitativa dos ciclistas que utilizam as vias públicas registradas em um vídeo, assim como uma discussão sobre os obstáculos encontrados no rastreamento e detecção destes ciclistas. A detecção e rastreamento dos ciclistas se faz pela diferença de *frames*, rastreamento de cantos e vetores de movimento para buscar a correspondência dos componentes conexos encontrados entre os *frames*. A classificação é realizada por aprendizado de máquina (SVM), previamente treinada com imagens extraídas dos vídeos, utilizando uma versão modificada do algoritmo SIFT como descritor destas imagens. Os resultados obtidos foram medianos em relação a contagem e descrição dos ciclistas e as dificuldades encontradas foram expostas na avaliação do protótipo.

Palavras-chave: Visão Computacional, Ciclistas, Detecção e Rastreamento

ABSTRACT

ROSE, Israel Laurensi; Guimarães, Rafael José; TOZZI, Vítor de Oliveira. DETECTING AND TRACKING BICYCLES IN PUBLIC ROADS WITH COMPUTATIONAL VISION TECHNIQUES. 59 f. Monografia – Departamento Acadêmico de Informática, Universidade Tecnológica Federal do Paraná. Curitiba, 2016.

The use of bicycles as a means of transport has attracted the attention of cities searching to provide ways for cyclists to travel safely in exchange for the use of a healthy means of transport and a reduction in the emission of pollutants produced by motor vehicles. For this, we developed a solution with the objective of performing a quantitative analysis of the cyclists using the public roads recorded in a video, as well as a discussion about the obstacles encountered in the tracking and detection of these cyclists. Detecting and tracking cyclists is done by the difference of frames, tracking of corners and motion vectors to search matching of related components found between the frames. The classification is performed by machine learning (SVM), which was previously trained with images extracted from the videos, using a modified version of the SIFT algorithm as descriptor of these images. The results obtained were medium in relation to the counting and description of the cyclists and the difficulties encountered were exposed in the evaluation of the prototype.

Keywords: Computer Vision, Cyclists, Detection and tracking

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

SIFT	Scale-Invariant Feature Transform
SVM	Support Vector Machine
SURF	Speeded Up Robust Features
HOG	Histograms of Oriented Gradients
LBP	Local Binary Pattern
MLP	MultiLayer Perceptron Network
RBFN	Radial Basis Function Network
MHI	Motion History Image
MEI	Motion Energy Image
DCNN	Deep Convolutional Neural Network

SUMÁRIO

1 INTRODUÇÃO

Nos últimos anos, houve um crescente interesse na análise automática de atividades relacionadas ao tráfego urbano. Fatores como o fácil acesso a dados, a melhoria de infraestruturas de câmeras e sensores, assim como o aumento no número de técnicas de análise e processamento de dados, possibilitam esse cenário (BUCH et al., 2011). Diversos estudos já realizados na área sustentam essa afirmação, como por exemplo, a análise da interação de ciclistas e carros a partir de técnicas de visão computacional (SAYED et al., 2013) ou a análise do tráfego a partir do rastreamento de veículos (COIFMAN et al., 1998).

Tecnologias para monitoramento do tráfego de veículos podem ser classificadas de duas formas, segundo (MATHEW, 2014): intrusivos e não intrusivos. Sistemas intrusivos são baseados, por exemplo, em laços indutivos, tubos pneumáticos e sensores magnéticos. Estes sistemas normalmente são mais baratos, mas têm instalação mais complicada, e exigem manutenção frequente devido ao uso constante da via ou por causa de infiltrações de água da chuva. Sistemas não intrusivos, por sua vez, são baseados em câmeras e sensores que interveem pouco ou nada no funcionamento da via. Câmeras possuem também a vantagem de produzir mais dados sobre o cenário e objetos de interesse - um mesmo vídeo pode conter dados sobre diversos modais que trafegam na via, o que possibilita a análise de diferentes objetos simultaneamente. É importante notar, no entanto, que câmeras e sensores são sensíveis às mudanças climáticas e outros fatores externos - como luminosidade, sombras, objetos obstruindo o campo de visão, dentre outros - os quais influenciam diretamente na qualidade das imagens dos vídeos, podendo, em alguns casos, dificultar a extração de informações a respeito de determinados objetos.

1.1 DEFINIÇÃO DO PROBLEMA

O problema tratado neste trabalho é a detecção e rastreamento de ciclistas em vídeos. Dessa forma, o contexto geral do trabalho pode ser compreendido como o desen-

volvimento de um algoritmo capaz de reconhecer e rastrear ciclistas - diferenciando-os de carros, pedestres, motocicletas etc - a partir de um vídeo de entrada. A entrada esperada para o algoritmo é um vídeo, que pode ser definido como uma sequência de imagens (ver Figura 1), denotados como os *frames* do vídeo. Os vídeos devem ser gravados por uma câmera fixa e devem ser, essencialmente, de uma vista superior de um local no qual existe um fluxo de ciclistas.

Figura 1: Imagem capturada de um vídeo utilizado como entrada.

Uma consideração importante feita para a implementação da solução proposta foi a de que os ciclistas trafegam, aproximadamente, em sentido paralelo às vias nos vídeos, ou seja, da esquerda para a direita e vice-versa.

1.2 VISÃO GERAL

A partir de uma entrada de vídeo, espera-se que o algoritmo de detecção possa distinguir entre os diferentes objetos presentes nas gravações. Os objetos de interesse para este trabalho são os ciclistas, sendo que o objetivo final do algoritmo de detecção é encontrar ciclistas no vídeo. Um obstáculo já esperado neste quesito é como o algoritmo pode distinguir motociclistas de ciclistas, pois ambos apresentam várias características em comum a partir de uma visão de processamento de imagem. Outro possível obstáculo é o tratamento de objetos em movimento muito próximos um do outro, evidenciando um problema comum em visão computacional, que é a distinção entre objetos de tamanhos diferenciados. Diante disso, um problema esperado é o de como computar tal problema

para que o algoritmo compreenda a linha de separação entre objetos próximos um do outro.

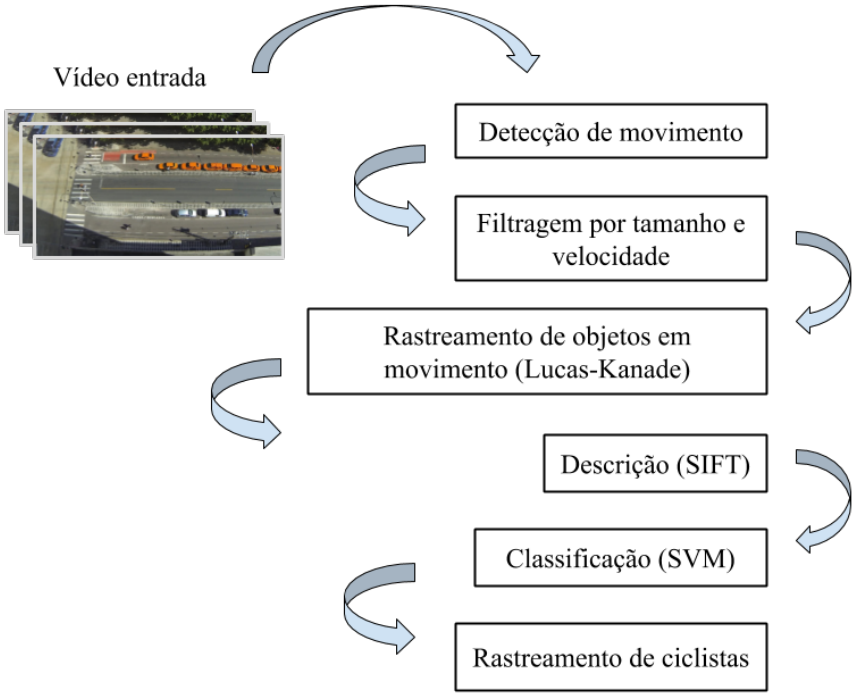
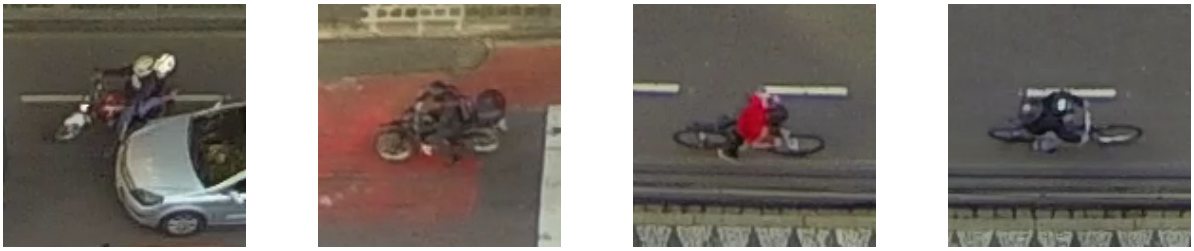


Figura 2: Resumo do processo geral a ser executado pelo algoritmo.

A Figura 2 ilustra o processo geral que espera-se ser executado pela solução computacional. Esse processo é compreendido em 6 etapas. O primeiro passo é detectar regiões que possuam movimento por meio de um algoritmo de comparação de *frames* do vídeo. O objetivo é focar em áreas que possivelmente contenham um ciclista em movimento. O segundo passo é um complemento do primeiro e consiste em filtrar determinados objetos encontrados de acordo com o seu tamanho no vídeo, uma vez que conhecemos de antemão as dimensões que não representam um ciclista (muito grandes ou muito pequenas), assim como filtrar pela velocidade do objeto nos diferentes *frames*. Vetores de movimento são extraídos das regiões dos objetos detectados, a fim de rastreá-los no próximo *frame*. Ainda para o rastreamento, o algoritmo de Lucas-Kanade foi utilizado para encontrar pontos das regiões dos objetos detectados em *frames* seguintes.



(a) (b) (c) (d)

Figura 3: Exemplos negativos (a e b) e exemplos positivos (c e d) para o treinamento do SVM.

Os objetos com tamanho adequado são então descritos por uma versão modificada do algoritmo SIFT (*Scale-Invariant Feature Transform*), que computa as diversas características locais de uma imagem. Os valores gerados pelo SIFT são testados na etapa de classificação, a partir da utilização do SVM (*Support Vector Machine*). O SVM é treinado previamente com diversas imagens de ciclistas, consideradas positivas, e diversas imagens que não contêm ciclistas, consideradas negativas (ver Figura 3). A partir da classificação do SVM, o objeto é então rastreado caso seja ciclista, ou ignorado, caso contrário.

Figura 4: Demonstração do rastreamento e contagem dos ciclistas.

A Figura 4 ilustra um exemplo do resultado esperado. A linha em verde identifica o trajeto realizado pelo objeto (trajeto este que é computado pelo algoritmo); a marcação do retângulo em verde identifica o objeto detectado; e a marcação em amarelo representa

a linha para contagem de ciclistas, ou seja, o ciclista é contabilizado assim que passar pela marcação. Dentre as informações relevantes para o trabalho, podemos citar aqui:

(a) análise quantitativa do fluxo de ciclistas;

(b) identificação e contagem de ciclistas em vias que são exclusivas para veículos motorizados.

A partir disso, um protótipo da solução computacional foi desenvolvido e testado, dados os vídeos de entrada e as hipóteses levantadas como premissa.

Os resultados obtidos se mostraram não muito satisfatórios devido a diversas características dos vídeos gravados, tais como: luz do sol gerando sombras nos objetos de interesse, o que dificultou na filtragem por tamanho do componente conexo; ciclistas que trafegam na ciclofaixa superior (mais distantes da câmera) possuem um padrão diferente dos que trafegam na ciclofaixa de baixo, o que dificultou na classificação e também na filtragem pelas dimensões esperadas de um ciclista; instabilidade da câmera, o que dificultou na distinção de objetos estáticos de objetos dinâmicos (que se moviam durante o vídeo).

Além destes, o número de imagens para treino do classificador utilizado (SVM) foi baixo, o que ocasionou em um resultado mediano em relação a distinção entre ciclistas e não-ciclistas. Para alguns vídeos, a taxa de acerto máxima foi de 67%, enquanto para outros foi muito baixa (36%).

1.3 OBJETIVO GERAL

O objetivo geral deste trabalho é propor e testar uma abordagem que possibilite a análise da movimentação de ciclistas em amostras de vídeos na cidade de Curitiba por meio do uso do algoritmo SIFT (*Scale-Invariant Feature Transform*) modificado, utilizado para descrição das imagens de interesse, em conjunto com o classificador SVM (*Support Vector Machine*), utilizado para classificar os objetos entre ciclistas e "não-ciclistas".

1.4 OBJETIVOS ESPECÍFICOS

- Investigar técnicas de processamento de imagens para detecção, reconhecimento e rastreamento de objetos em vídeo;

- Delimitar uma região na cidade de Curitiba para obter imagens e vídeos do tráfego de ciclistas, a fim de realizar as análises necessárias;
- Desenvolver uma solução computacional para a análise quantitativa do fluxo de ciclistas na região delimitada.

1.5 ORGANIZAÇÃO DO DOCUMENTO

O restante desse trabalho está organizado da seguinte forma: O Capítulo 2 apresenta alguns trabalhos que possuem relação direta ou indireta com este trabalho. O Capítulo 3 expõe os principais temas relacionados à pesquisa, detalhando os métodos e teorias a serem utilizados no projeto, de forma a dar um embasamento teórico para o trabalho. O Capítulo 4 apresenta o desenvolvimento do trabalho como um todo, explicando como foi feita a coleta dos dados e como foi feita a implementação da solução com detalhes. O Capítulo 5 apresenta os principais resultados obtidos, bem como alguns problemas encontrados no desenvolvimento. O Capítulo 6 traz as conclusões da pesquisa realizada e do trabalho desenvolvido, envolvendo uma análise da importância do projeto no contexto em que está inserido, bem como quais podem ser os futuros trabalhos relacionados, frente a tudo que foi estudado neste trabalho.

2 TRABALHOS RELACIONADOS

Este capítulo apresenta alguns trabalhos relacionados aos temas abordados neste trabalho. A Seção 2.1 cita alguns trabalhos referentes a área de planejamento urbano e mobilidade e a relação desta com as ciclofaixas e o uso da bicicleta. A Seção 2.2 cita alguns trabalhos que expõem métodos propostos para o monitoramento de vias públicas.

2.1 PLANEJAMENTO URBANO E CICLOVIAS

O trabalho desenvolvido está inserido no contexto do estudo e análise de aspectos urbanísticos de uma cidade, com o propósito de mesclar aspectos de Tecnologias da Informação e Comunicação no dia-a-dia das pessoas.

O uso da bicicleta, de acordo com Medeiros e Duarte (2013), como um modo de transporte tem sido vinculado a diversos benefícios para a cidade, de forma a promover a equidade social e como um meio de transporte saudável e amigável ao meio ambiente. Diversos estudos nessa área têm sido realizados buscando compreender os impactos de conciliar diferentes meios de transporte nos espaços urbanos, como os trabalhos de Silva (2011), Shaheen et al. (2010) e Hamilton-Baillie (2008).

Silva (2011) destaca em seu trabalho o crescente aumento na taxa de motorização, e como esse crescimento pode afetar o setor de transportes em geral. A hipótese levantada e confirmada pela autora é a de que o aumento desta taxa no cenário urbano provoca impactos negativos à circulação e à mobilidade das pessoas. A motorização excessiva se torna preocupante uma vez que provoca a degradação da circulação e da qualidade de vida nos grandes centros urbanos.

Ainda segundo a autora, por meio de uma análise dos dados fornecidos por órgãos governamentais, as principais consequências do aumento da taxa de motorização impactam diretamente em questões ambientais, assim como na segurança das pessoas. Além disso, a autora traz à luz da discussão as seguintes conclusões, como um dos resultados de seu trabalho: políticas referentes ao incentivo do uso do transporte público são escassas

e mal administradas; o aumento na taxa de motorização ocasiona a redução do índice de mobilidade e influencia a segurança no trânsito, ocasionando um aumento em acidentes e mortes.

Hamilton-Baillie (2008), a partir de um estudo histórico sobre o surgimento das discussões em torno do termo "*shared space*" - focando, principalmente, em regiões da Europa -, reflete em seu trabalho sobre alguns aspectos relacionados à convivência entre pessoas em espaços destinados para a locomoção de um lugar para outro. O termo se refere ao problema de conciliar o convívio entre pessoas e os diferentes meio de locomoção - bicicletas, carros, ônibus etc - em espaços compartilhados. Segundo o autor, o conceito de "*shared space*", de que todos os pedestres se movem e interagem no seu uso do espaço baseado em protocolos sociais informais e negociações, não é nada novo.

Um dos aspectos relatado pelo autor é em relação à percepção da segurança no transporte, uma vez que há um declínio no ciclismo e caminhada como modos de transporte, e o aumento da dependência do carro para locomoção. Isso, como destaca o autor, ocasiona uma série de preocupações em relação ao espaço que está sendo usado pelas pessoas. Uma medida de intervenção a isso foi a separação do espaço destinado para locomoção de pessoas e o espaço para veículos, enfatizado pelo autor como o "princípio da segregação". Assim, o planejamento e regulamentação desses espaços se tornaram preocupações a partir do momento que o tráfego de veículos aumentou.

Shaheen et al. (2010) destacam os principais aspectos relacionados aos sistemas de uso compartilhado de bicicletas públicas oferecidos em alguns países da Europa, Ásia e nas Américas. Os autores ressaltam 3 pontos principais que puderam concluir a partir de seus estudos: (a) redução no uso de bicicletas pessoais; (b) aumento do uso da bicicleta para atividades diárias; e (c) aumento da percepção de como a bicicleta pode ser usada como um meio conveniente de transporte. Além destes, o estudo demonstra como o uso frequente da bicicleta pode reduzir a emissão de gases poluentes, uma vez que grande parte dos usuários deixam de utilizar veículos motorizados para locomoção em troca da bicicleta. No entanto, os autores ainda enfatizam os problemas e obstáculos que tais sistemas podem vir a enfrentar, como: roubos, alto custo tecnológico (caso existam sistemas de informação por trás), instalação e manutenção da infraestrutura física necessária, além de problemas de segurança, tanto em relação as bicicletas quanto em relação a segurança dos ciclistas ao trafegarem pela cidade.

Cabe aqui citar alguns fatores elencados no Planejamento Cicloviário - Diagnóstico Nacional - GEIPOT (2001) como possíveis causas para o baixo uso da bicicleta como meio

de transporte (GEIPOT, 2001, p. 16). Esses fatores, dentre outros, são:

- aumento do volume do tráfego motorizado;
- aumento do número de acidentes graves com ciclistas na via pública;
- baixo valor dos automóveis usados com muitos anos em circulação;
- maior distância entre os locais de moradia e trabalho;
- falta de respeito ao ciclista e impunidade no trânsito.

Em meio a outros, é importante notar que alguns dos fatores elencados possuem relação direta com aspectos já observados na literatura em outros países, em especial as questões de segurança e de como isto afeta a percepção do uso da bicicleta como um meio conveniente de transporte no dia a dia.

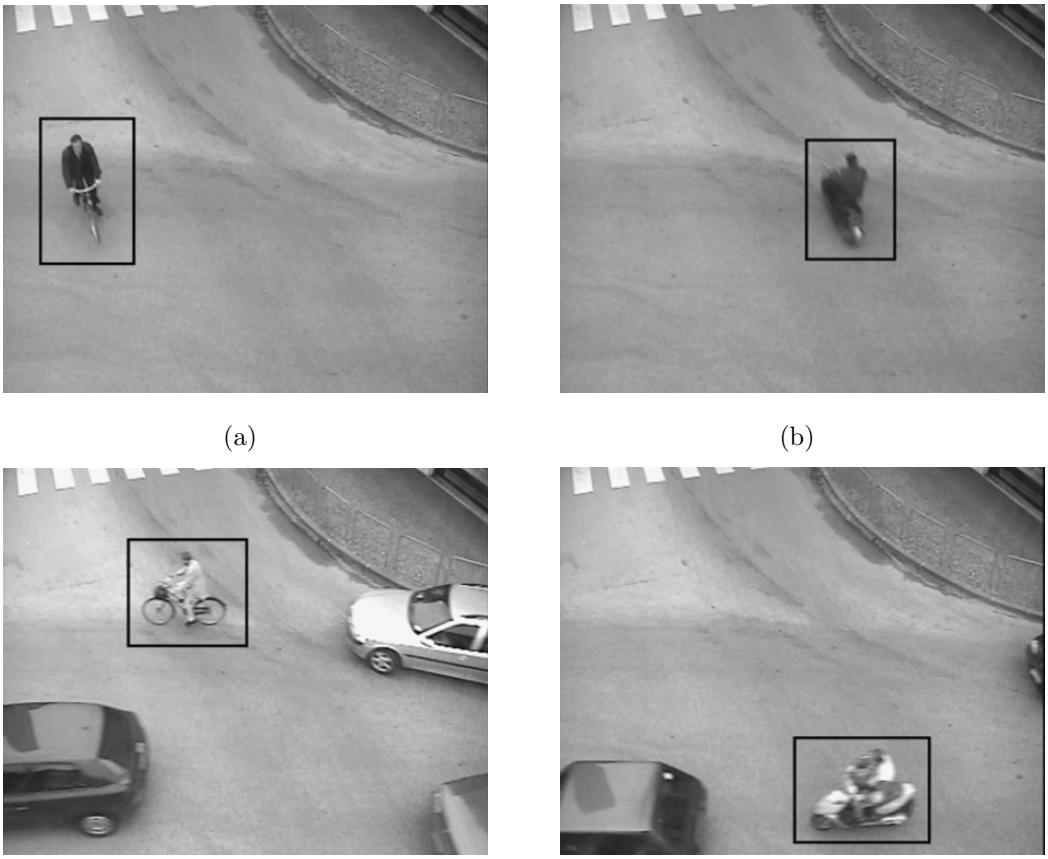
As discussões observadas na literatura a respeito do espaço urbano e de ciclovias serviram como base e motivação para o desenvolvimento do trabalho como um todo, afinal, são estes que definem o contexto geral em que este trabalho está inserido. Ademais, os resultados apresentados por nós neste trabalho podem servir como base para estudos futuros, tomando como base os diversos termos discutidos nesta seção, além de outros presentes na mesma área de estudo.

2.2 ANÁLISE DE FLUXO POR DETECÇÃO E RASTREAMENTO

Diversos métodos de detecção e rastreamento podem ser encontrados na literatura. Alguns destes são aplicados diretamente ao problema de monitoramento - em tempo real ou não - do fluxo de tráfego em vias públicas, como o trabalho de Messelodi et al. (2007), Luvizon et al. (2014), Silva et al. (2013) e LabProdam (2016), discutidos nesta seção.

Messelodi et al. (2007) observaram que certas características em imagens de ciclistas e motociclistas poderiam ser utilizadas para distinguir ambos em sistemas de monitoramento por vídeos. Os autores descrevem dois casos principais, ilustrados na Figura 5, a partir do método proposto: (a) quando o objeto está sendo observado de lado (Figura 5(c) e 5(d)), a região central das rodas da bicicleta tende a ser mais semelhante ao fundo da imagem comparada a das motocicletas; (b) quando o objeto está sendo observado de frente ou de trás (Figura 5(a) e 5(b)), é possível distinguir bicicletas de motocicletas a

partir da análise do tamanho das rodas, uma vez que as de motocicletas tendem a ser mais grossas. É importante ressaltar que a análise feita pelos autores depende do posicionamento da câmera, a qual não pode estar muito distante dos objetos - pois faz uso dos detalhes para distinguir -, e depende também do ângulo de visão do vídeo, a fim de capturar imagens que contenham ciclistas vistos tanto de lado (aproximadamente), quanto de frente ou de trás. Para a nossa abordagem, estes métodos não poderiam ser utilizados, dada a angulação dos vídeos e a distância da câmera para os objetos de interesse.



(a) (b)

(c) (d)

Figura 5: Exemplos de ciclistas e motociclistas vistos de frente (a), de trás (b) e de lado (c e d). Fonte: Messelodi et al. (2007).

Luvizon et al. (2014) propuseram um método para estimar a velocidade de veículos motorizados a partir da detecção e rastreamento da placa do veículo. O método proposto detecta, primeiramente, as regiões de interesse a partir do movimento, separando objetos estáticos de objetos dinâmicos (em movimento), utilizando um algoritmo de subtração de fundo e o MHI (*Motion History Image*). A partir disto, as regiões filtradas servem como entrada para o algoritmo detector de placas, o qual utiliza algumas características da imagem para encontrar a região envolvente da placa. Ainda, o algoritmo *Scale-Invariant*

Feature Transform (SIFT) é utilizado para fazer uma estimativa inicial do deslocamento, extraindo as principais características da região, dadas a angulação e escala das imagens em diferentes momentos do vídeo. O rastreamento das placas é feito por meio do algoritmo de Kanade-Lucas-Tomasi (KLT). A velocidade do veículo é estimada comparando os principais pontos detectados em diferentes *frames* do vídeo, levando em consideração as medidas do mundo real.

Silva et al. (2013) apresentam um método para a detecção e rastreamento de motocicletas em uma via pública. O objetivo do trabalho é diferenciar objetos em duas classes, sendo: motocicletas ou "não-motocicletas". A Figura 6(a) ilustra o cenário para detecção e rastreamento das motocicletas. A abordagem proposta pelos autores utiliza subtração de fundo como um passo inicial para encontrar objetos em movimento, como ilustrada na Figura 6(b). A partir disso, as características de cada componente conexo encontrado são extraídas pelo uso de quatro abordagens diferentes: *Speeded Up Robust Features* (SURF), proposto por Bay et al. (2006); Transformada de Haar, proposta por Rafiq e Siddiqi (2009); *Histograms of Oriented Gradients* (HOG), proposto por Dalal e Triggs (2005); e *Local Binary Pattern* (LBP), proposto por Ojala et al. (1996).

(a) (b)

Figura 6: Monitoramento para detecção automática de motocicletas uma via pública. Fonte: Silva et al. (2013).

As características extraídas das imagens são utilizadas para distinguir os objetos. Assim, para a classificação, tomando as características extraídas como entrada, os autores utilizam três algoritmos diferentes, sendo eles: *MultiLayer Perceptron Network* (MLP), *Radial Basis Function Network* (RBFN) e *Support Vector Machine* (SVM). Um ponto importante observado, a partir dos testes realizados pelos autores, é o de que o uso do classificador SVM com o algoritmo LBP forneceu os melhores resultados para distinguir

motocicletas de bicicletas.

Em 2016, o LabProdam (2016) - Laboratório de Inovação da Prefeitura de São Paulo - iniciou um projeto voltado para o monitoramento de uma via pública da cidade de São Paulo. O *software* desenvolvido utiliza as imagens captadas por uma câmera fixa apontada para um local com uma vista superior de uma ciclofaixa, cujo objetivo é detectar e rastrear ciclistas no vídeo. A Figura 7 ilustra o sistema de monitoramento em tempo real, o qual utiliza o algoritmo proposto pelo LabProdam. Nas imagens - 7(a) durante o dia e com o tempo chuvoso e 7(b) durante a noite sem chuva -, a ciclofaixa é a via central em vermelho, e os números próximos a ela indicam a contagem de ciclistas que cruzaram pela ciclofaixa naquele dia.

(a) (b)

Figura 7: Monitoramento em tempo real feito pelo LabProdam de uma via pública de São Paulo. Fonte: LabProdam (2016).

O método utilizado pelos autores do projeto para detecção e rastreamento de ciclistas pode ser descrito da seguinte maneira:

- delimitação do espaço para processamento;
- separação do que é fundo da imagem (estático) do que é dinâmico (pessoas, carros, ônibus e afins);
- avaliação do objeto: se o objeto possui as dimensões de uma bicicleta e tem uma trajetória e velocidade semelhantes às de um ciclista, então é categorizado como bicicleta, caso contrário, o objeto é ignorado.

É importante notar que a contagem dos ciclistas é feita somente na área delimitada pelo usuário. Isso ressalta um ponto importante: o uso das dimensões e velocidade

média para classificar um ciclista pode não funcionar caso a área delimitada englobe regiões em que outros veículos transitam, como motocicletas, que podem não ser classificadas de maneira correta. Em nossa abordagem, a filtragem utilizando as dimensões de um ciclista é utilizada como um passo preliminar, a fim de eliminar objetos que, na maior parte das vezes, não são ciclistas. No entanto, apenas essa filtragem não é suficiente para distinguir motocicletas e bicicletas, uma vez que em nossos vídeos as vias são compartilhadas. A partir disso que surge a necessidade do uso de classificadores.

3 ABORDAGEM PARA DETECÇÃO E RASTREAMENTO DE CICLISTAS

A abordagem proposta para resolução do problema apresentado neste trabalho tem como entrada uma sequência de vídeos. A partir destes, é feita a detecção de objetos que estão em movimento. Objetos que possuem tamanho muito pequeno ou muito grande - menores ou maiores do que os esperados para um ciclista - são descartados, bem como aqueles que possuam uma velocidade muito menor ou maior do que as esperadas para um ciclista. Na sequência, a descrição destes é feita com o uso do algoritmo SIFT modificado e estas características são então classificadas como ciclista ou não ciclista pela SVM, a qual foi treinada previamente com imagens positivas (ciclistas) e negativas (não ciclistas). Caso seja um ciclista, o rastreamento deste é feito nos *frames* subsequentes, caso contrário, o rastreamento não é feito.

As seções seguintes detalham os métodos para a realização deste projeto e estão divididas da seguinte maneira. Na Seção 3.1 são definidos os conceitos mais básicos usados para o desenvolvimento desse trabalho. Na Seção 3.2 são apresentados os dados usados como entrada para o processo de detecção e rastreamento, bem como as características necessárias destes dados. Na Seção 3.3 são explicados os métodos usados na identificação dos objetos em movimento e da segmentação dos possíveis ciclistas. Na Seção 3.4 são apresentadas as técnicas usadas para diferenciação dos ciclistas dos demais objetos encontrados em movimento.

3.1 DEFINIÇÕES BÁSICAS

3.1.1 REPRESENTAÇÃO DE IMAGENS

Uma imagem pode ser definida como uma função bidimensional de luminosidade $f(x, y)$, onde x e y são as coordenadas espaciais e o valor de f em (x, y) é proporcional à luminosidade do cenário naquele ponto (GONZALEZ; WOODS, 2000).

Uma imagem digital é uma imagem que foi discretizada em ambas coordenadas

espaciais e de luminosidade. Assim, pode ser representada por um vetor ou uma série de vetores bidimensionais, um para cada canal de cor. Dado que uma imagem possui um valor N de largura e um valor M de altura, dizemos que uma imagem é representada por M linhas e N colunas:

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \cdots & f(1,N-1) \\ f(2,0) & f(2,1) & f(2,2) & \cdots & f(2,N-1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \cdots & f(M-1,N-1) \end{bmatrix} \quad (11)$$

Cada elemento do vetor bidimensional, ou matriz, é chamado de *pixel*. Os termos **imagem** e **pixel** serão utilizados neste trabalho para denotar uma imagem digital e seus elementos, respectivamente.

3.1.2 VIDEO

Um vídeo é um padrão de intensidade espaço-temporal 3D/4D, isto é, um padrão espacial de intensidade que varia com o tempo. Outro termo comumente usado para vídeo é sequencia de imagens, sendo que uma imagem é representada por uma sequência temporal de imagens fixas (TEKALP, 1995). Dada a notação de uma imagem por $f(x,y)$, temos que o *frame* $f(x,y,t)$ é um imagem — ou quadro — de um vídeo em um instante t.

3.2 DADOS DE ENTRADA

A entrada do sistema é um vídeo $f(x,y,t_0), f(x,y,t_1), ..., f(x,y,t_n)$ onde t_n é a duração total do vídeo. No contexto deste trabalho, é importante que o vídeo seja capturado de uma posição superior à via monitorada, distanciando de forma que os objetos de interesse — ciclistas — caibam completamente no quadro.

Duas considerações importantes devem ser feitas em relação aos vídeos: (a) a superfície da via é aproximadamente plana e sem grandes mudanças na inclinação, o que evita a distorção na aparência dos objetos; (b) as vias capturadas nos vídeos são aproximadamente horizontais, o que implica que a trajetória dos ciclistas corta o plano

da imagem aproximadamente na horizontal.

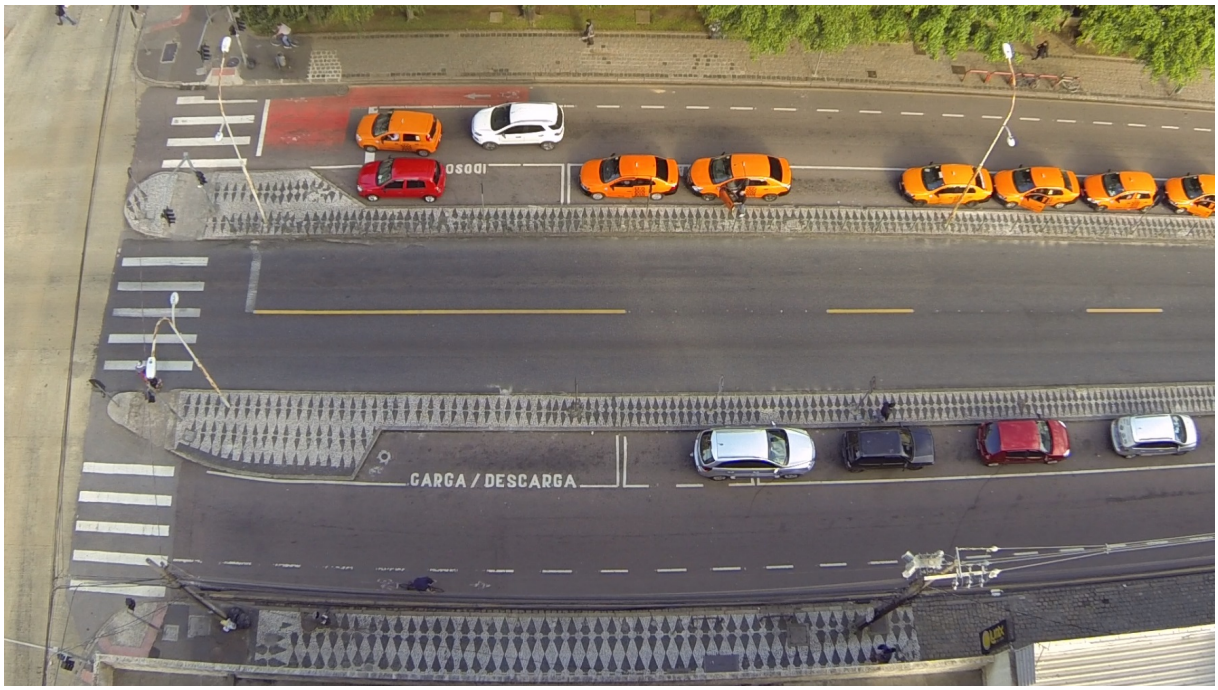


Figura 8: Vista superior do local onde as gravações foram realizadas.

A Figura 8 ilustra um possível ângulo para as gravações dos vídeos. Por uma questão prática, é interessante que o vídeo mostre pelo menos uma faixa da via completamente, ou mais se possível, com isso permitindo o monitoramento de um espaço maior com menos câmeras.

Algumas regiões do vídeo não possuem tráfego de ciclistas - como as calçadas. Isto implica que podemos diminuir o processamento evitando que tais regiões sejam processadas pelo algoritmo. Além disso, é importante que sejam definidas regiões de interesse para que estas possam ser analisadas de acordo com o que representam. Assim, a via superior e inferior podem ser consideradas como vias compartilhadas, e a via do meio como exclusiva para veículos. A partir disso, é possível evitar determinadas regiões que não possuem tráfego de ciclistas e ainda intitular determinadas regiões para que contenham um significado para a contagem dos ciclistas detectados.

As demarcações das regiões de interesse, para o local da Figura 8, estão ilustradas na Figura 9. As regiões de interesse são determinadas previamente, a fim de delimitar para o algoritmo quais os locais em que a detecção e rastreamento precisam ser feitos.

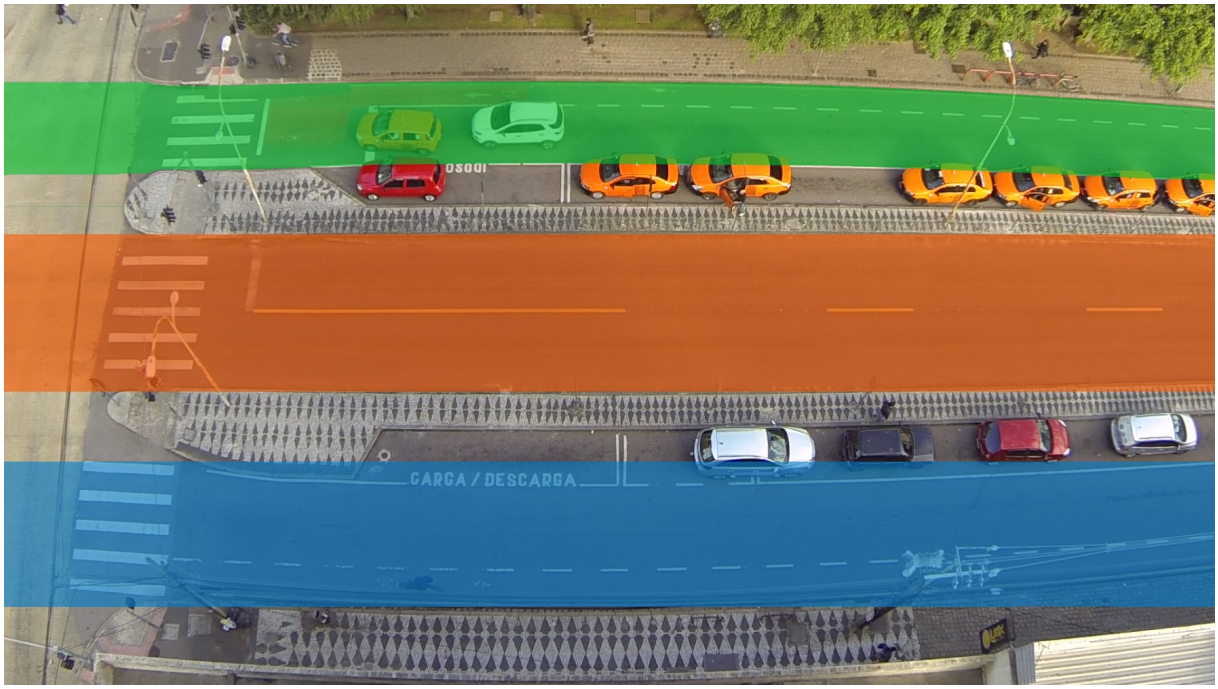


Figura 9: Delimitações das áreas de interesse em um *frame* do vídeo. Em verde a via compartilhada superior, em azul a via compartilhada inferior e em laranja a via exclusiva para ônibus e veículos de emergência.

Assim, de acordo com a Figura 9, temos que:

- Faixa superior (em verde) representa a via compartilhada com o tráfego indo para a esquerda;
- Faixa inferior (em azul) representa a via compartilhada com o tráfego indo para a direita;
- Faixa intermediária (em laranja) representa a via exclusiva para ônibus e veículos de emergência.

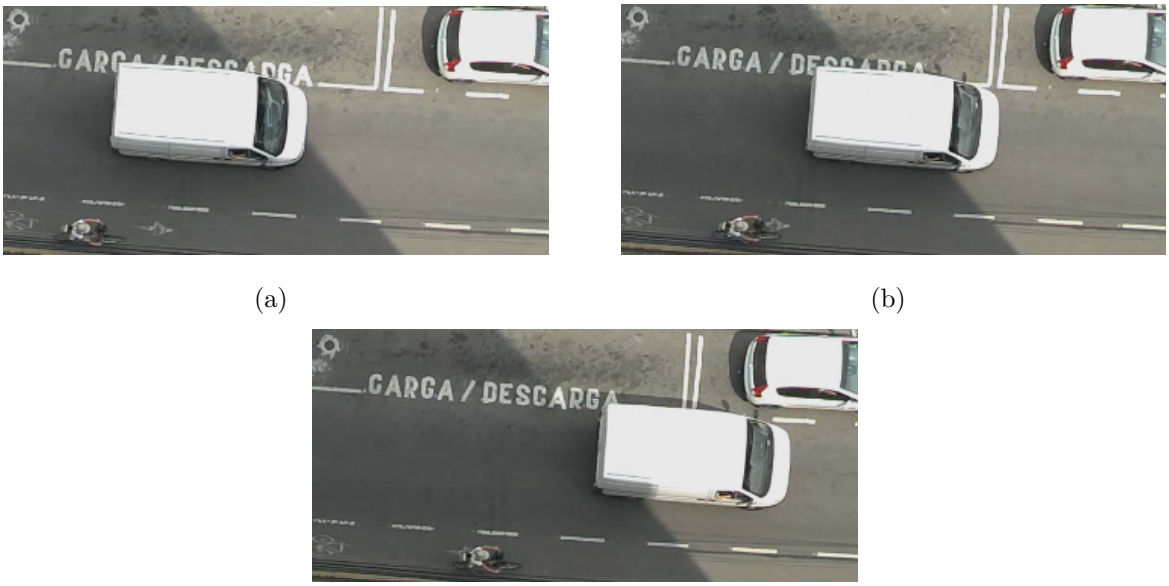
Um fator interessante a ser citado sobre essas marcações é que elas são baseadas na premissa de que a via é utilizada pelo meio de transporte designado a ela. A partir destas premissas, e das saídas geradas pelo *software*, é possível, em trabalhos futuros, verificar o uso correto das vias.

3.3 DETECÇÃO DE MOVIMENTO

Métodos para detecção e métodos para rastreamento diferem-se em vários fatores, ainda que estejam muito interligados em relação às aplicações práticas dos algoritmos.

Segundo Beaugendre et al. (2012), o processo de detectar objetos utiliza informações de "baixo nível", ou seja, o valor dos pixels da imagem, enquanto o processo de rastrear utiliza informações de "alto nível" dos objetos em movimento, como a velocidade, tamanho, aparência, dentre outros.

Na abordagem que propomos para detecção e rastreamento de ciclistas, a detecção de movimento é usada em um passo inicial, que detecta objetos em movimento para posteriormente classificá-los como ciclistas ou não-ciclistas. Este método recebe como entrada uma sequência sucessiva de *frames* que após processados produzem como saída uma imagem binária, representando os objetos em movimento na cor branca, e o fundo ou cenário na cor preta. A Figura 10 mostra uma sequência (não subsequente) de *frames* de um veículo e um ciclista se movimentando em uma rua.



(a) (b)

(c)

Figura 10: Sequência de *frames*.

O resultado produzido pela diferença de *frames* são várias regiões de interesse que podem representar pedestres, ciclistas, automóveis ou qualquer objeto que apresentar movimento. Os métodos descritos a seguir que envolvem a comparação entre dois *frames* levam em conta o *frame* $f(x, y, t)$ em um instante t e o seu *frame* subsequente $f(x, y, t+1)$.

3.3.1 MHI E MEI

O *Motion History Image* (MHI) é uma imagem com valor escalar na qual a intensidade é uma função do quão recente é o movimento (BOBICK; DAVIS, 2001),

de modo que os objetos são representados em escala de cinza, sendo que os *pixels* de maior intensidade simbolizam as mudanças mais recentes. Esta técnica é robusta para representar movimento, sendo usada em aplicações relacionadas ao reconhecimento de movimentos em cenários urbanos (BABU; RAMAKRISHNAN, 2004).

O MHI é computado de acordo com a Equação 12, sendo $f_{mhi}(x, y, t)$ uma imagem em escala de cinza, levando em conta os *frames* $f(x, y, t-1)$ e $f(x, y, t)$.

$$f_{mhi}(x,y,t) = \begin{cases} \tau & \text{se } \left|f(x,y,t-1) - f(x,y,t)\right| > \xi \\ max((f_{mhi}(x,y,t)-1), 0) & \text{caso contrário} \end{cases} \tag{12}$$

sendo τ um valor que representa a duração esperada em unidade de *frames* e ξ a sensibilidade que determina se para um dado pixel houve alteração significativa a ponto de considerar como movimento. No presente trabalho, estes parâmetros foram fixados em $\tau = 5$ e $\xi = 30$

Partindo do MHI computado, podemos obter uma imagem binária $f_{mhi}(x, y, t)$ conhecida como MEI (*Motion Enery Image*) (BOBICK; DAVIS, 2001). Sua representação é dada pela equação a seguir:

$$f_{mei}(x,y,t) = \begin{cases} 1 & \text{se } f_{mhi}(x,y,t) > 0 \\ 0 & \text{caso contrário} \end{cases} \tag{13}$$

Seguindo esta abordagem, o MEI foi utilizado para produzir a máscara de movimento (ver Figura 11), a qual representa, a cada *frame* do vídeo, quais são os objetos que estão em movimento.

(a)

(b)

Figura 11: Imagem em escala de cinza (a) e a respectiva representação do MHI binarizado (MEI) desta imagem (b).

3.3.2 SEGMENTAÇÃO DE COMPONENTES CONEXOS

Assim que obtida a máscara de movimento, o próximo passo é analisar a vizinha de cada *pixel* da imagem $f(x, y, t)$ sinalizado como "em movimento" na máscara de movimento. Caso um *pixel* da vizinha seja encontrado como "em movimento", este é adicionado a uma estrutura de dados e o processo é repetido para *pixel* encontrado. O agrupamento destes *pixels* torna-se um componente conexo, também denominado como *blob*.

Para a rotulagem dos componentes utilizamos a abordagem conhecida como *flood fill*, que tem como objetivo rotular um componente conexo por inundação. Na utilização desta técnica, podemos inundar um componente conexo com um determinado valor que representa unicamente aquele componente conexo. Para Bradski e Kaehler (2008) o *flood fill* pode também ser utilizado para obter, a partir de uma imagem de entrada, máscaras que podem ser usadas para rotinas subsequentes para acelerar ou restringir o processamento para apenas os pixels indicados pela máscara.

Consideramos em nossa solução o algoritmo do *flood fill* utilizando pilha, exposto de forma simplificada no Algoritmo 1:

Algoritmo 1 Floodfill

```
1:  procedure FLOOD(label, f, x_seed, y_seed, targetColor)
2:      push(stack, (x_seed, y_seed))
3:      while stack is not empty do
4:          (x, y) ← pop(stack)
5:          f(x, y) ← label
6:          for (each neighbor f(x', y') of f(x, y)) do
7:              if (f(x', y') == targetColor and inBounds((x', y'))) then
8:                  push(stack, (x', y'))
9:              end if
10:         end for
11:     end while
12: end procedure
```

O algoritmo inicia uma busca a partir das posições iniciais x_seed e y_seed em sua vizinhança por valores de $f(x, y)$ iguais ao da posição inicial. A variável *targetColor* foi utilizada para facilitar o entendimento do algoritmo, porém supõe-se que o valor de *targetColor* seja o mesmo da posição inicial. A função *inBounds* utilizada no algoritmo faz apenas uma verificação se os valores de x e y estão dentro das margens de altura e largura da imagem f. Assim que satisfeitas as condições de uma posição da vizinhança, esta posição é adicionada a uma pilha. Quando a pilha estiver vazia, significa que todas posições que foram empilhadas já foram rotuladas, ou seja, o componente está completamente rotulado.

A busca de vizinhança utilizada no algoritmo é de quatro direções, sendo elas: cima, baixo, esquerda e direita (BRADSKI; KAEHLER, 2008) confome a Figura 12. A Figura 13 exibe um exemplo simples do processo de rotulagem.

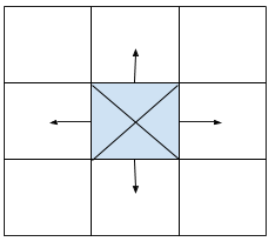


Figura 12: Análise de vizinhança em quatro direções (cima, baixo, direita e esquerda).

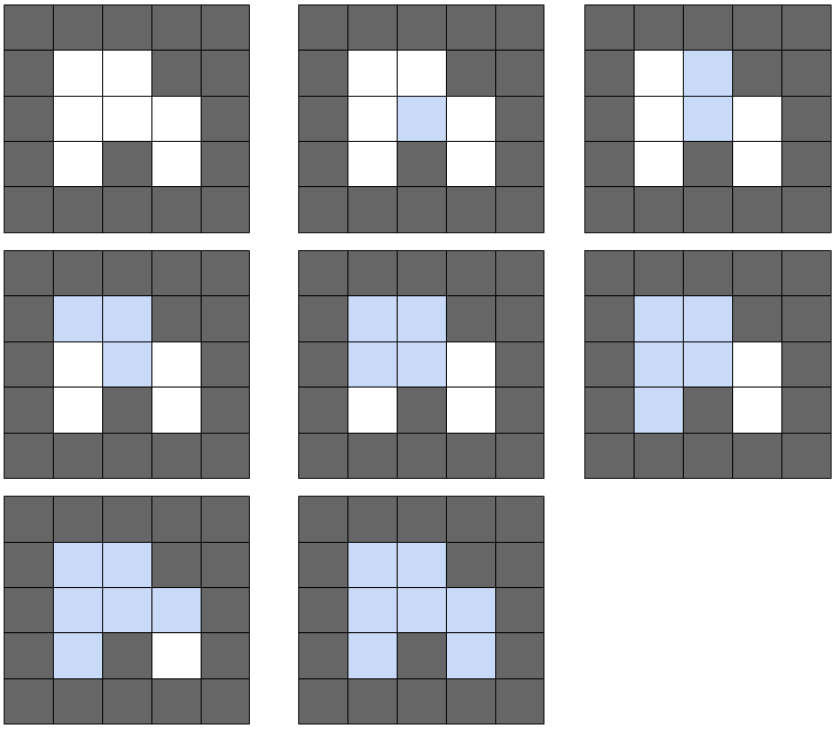


Figura 13: Exemplo ilustrativo da rotulagem utilizando pilha.

A partir do momento que um componente conexo é rotulado, este passa por um filtro parametrizado por duas dimensões — tamanhos máximos e mínimos dos componentes conexos — o qual descarta os que não se encaixarem no intervalo do tamanho definido como aceitável. As dimensões utilizadas neste processo são baseadas nos tamanhos máximos e mínimos que os ciclistas apresentam nas imagens. Esta técnica simples, além de remover ruídos, acaba por remover também componentes conexos de dimensões acima das dimensões esperadas, que podem representar carros, ônibus e outros veículos que trafegam na mesma área que os ciclista e que não precisam ser detectados e rastreados.

Portanto, seja $\mathbb{L}$ um conjunto de componentes conexos, e dada uma função $H(B_h, B_w)$ que retorna 1 caso os parâmetros B_h e B_w que representam altura e largura respectivamente estiverem dentro das dimensões aceitáveis, temos a seguinte expressão para cada componente conexo B:

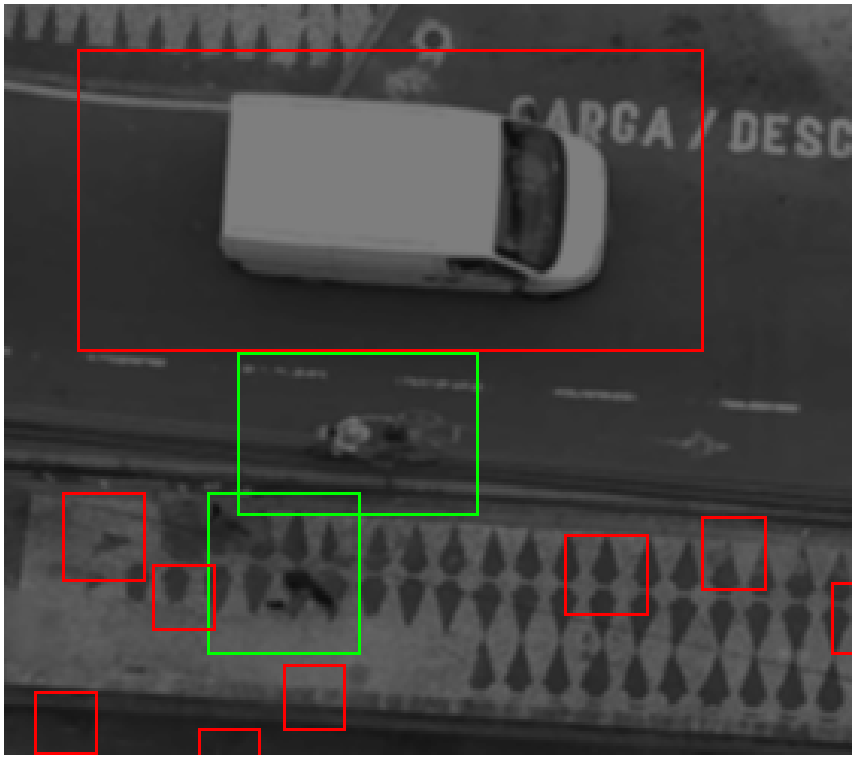
$$\forall B, Bi \in \mathbb{L} = \begin{cases} H(B_h i, B_w i) = 1, & \text{adiciona o componente conexo à lista} \\ H(B_h i, B_w i) = 0, & \text{descarta o componente conexo} \end{cases} \tag{14}$$

Na Figura 14 é possível ver o descarte dos componentes conexos marcados em forma de retângulo pelas suas dimensões de altura e largura. Nos componentes conexos

aceitos (em verde), podem-se identificar na imagem um ciclista e um pedestre. Dos componentes conexos recusados, pode-se identificar um veículo no retângulo maior e ruídos nos retângulos menores. É importante notar que o uso do termo "ruído" aqui se refere a pequenos componentes conexos que são desconsiderados, sejam estes criados por pequenas movimentações da câmera ou pela movimentação de objetos muitos pequenos no vídeo.

Além disso, uma filtragem com a velocidade dos objetos é realizada a fim de eliminar objetos que estejam muito rápido ou muito devagar para serem considerados como um ciclista. Esse parâmetro visa somente eliminar objetos que possuam a velocidade menor ou maior do que as esperadas de um ciclista. Isso se deve ao fato de que um ciclista pode não estar a uma velocidade muito alta - quanto um carro -, mas pode estar com uma velocidade aproximada a de uma pessoa caminhando.

(a)



(b)

Figura 14: Imagem original (a) e em (b) os componentes conexos aceitos em verde e descartados em vermelho.

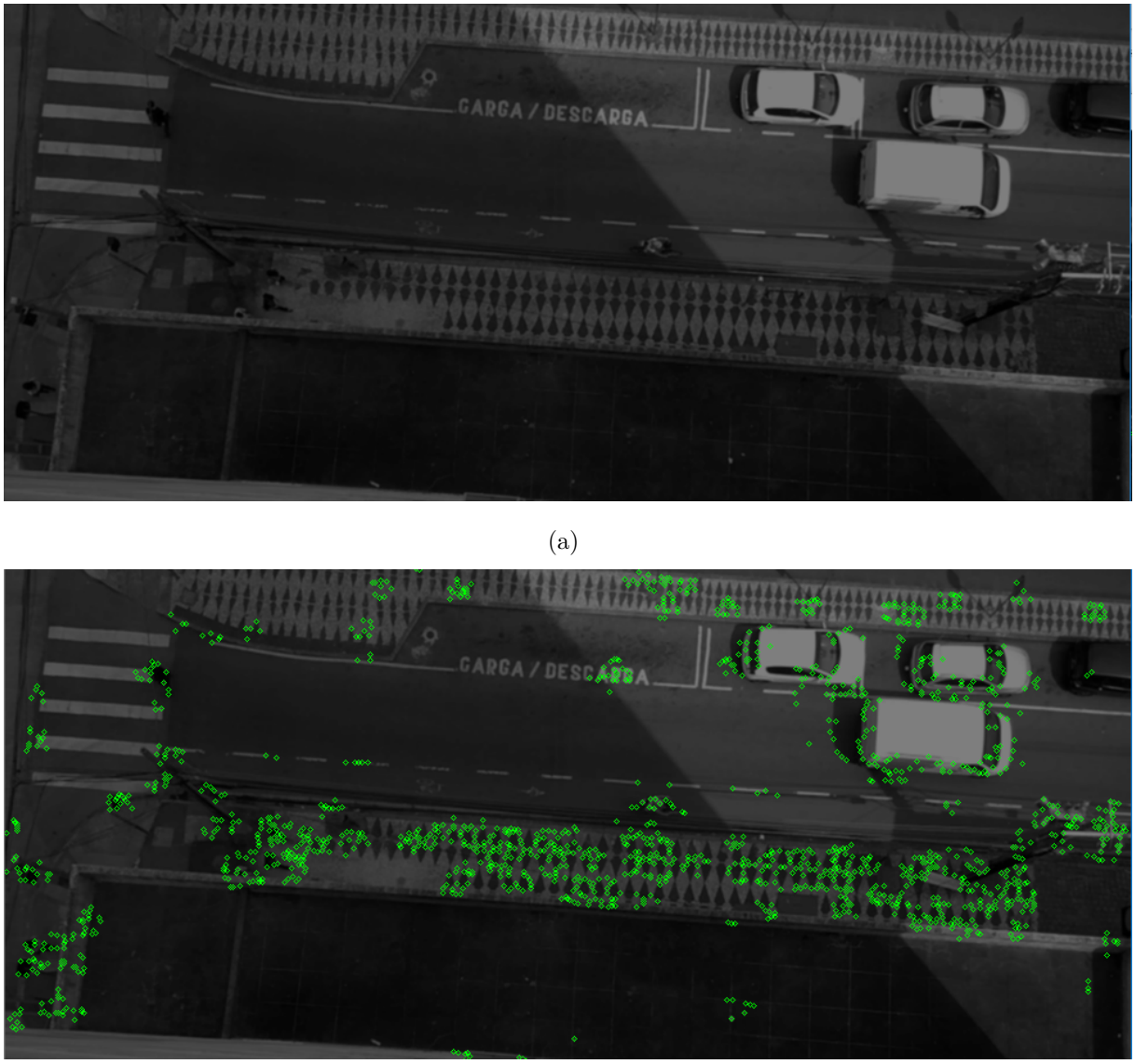
3.3.3 RASTREAMENTO DE PONTOS

Com a máscara de movimento e a lista de componentes conexos produzidas pelas etapas anteriores, o próximo passo é rastrear os objetos entre os *frames*. Para isto, é preciso encontrar pontos destes objetos de forma que seja possível encontrá-los no próximo *frame*. Deste modo, podemos definir este processo em duas etapas: encontrar os pontos do objeto em um instante t e depois localizar a correspondência destes pontos em $t + 1$. Existem muitos tipos de características locais que se podem rastrear em uma imagem. Assim, uma vez que encontrar pontos se trata de uma abordagem local, é importante utilizar uma técnica robusta para a procura de sua correspondência em outra imagem - neste caso, em um *frame* seguinte no vídeo.

Segundo Bradski e Kaehler (2008), ao rastrear objetos, é recomendado utilizar características relevantes — pontos únicos ou quase únicos e parametrizáveis — de forma que possam ser comparados com características ou outros pontos em uma outra imagem. Os autores ainda sugerem que a busca deve ser feita por pontos que possuam alguma diferença significativa entre eles.

Segundo os autores um ponto cuja magnitude do gradiente tem um valor alto pode estar associado à alguma borda. Neste contexto, temos uma característica chamada de canto sendo um ponto que possui derivadas com valor absoluto alto em mais de uma direção, e que contém informações suficientes para ser encontrada em um *frame* seguinte.

A definição mais comum de canto foi proposta por Harris e Stephens (1988) como uma matriz contando as derivadas de segunda ordem das intensidades da imagem. Shi e Tomasi (1994) mais tarde descobriram que era possível obter bons resultados de forma que o menor valor entre dois autovalores fosse maior que um limiar mínimo. O método de Shi e Tomasi para detecção de cantos, conhecido também como *Good Features To Track* não somente foi suficiente como em alguns casos produziu resultados mais satisfatórios que o método de Harris (BRADSKI; KAEHLER, 2008). Uma ilustração dos cantos encontrados utilizando o método de Shi e Tomasi pode ser visualizada na Figura 15. O método otimiza a seleção de features não restringindo apenas à cantos, mas também *features* com boas texturas para o fim de otimizar o rastreamento.



(a)

(b)

Figura 15: Imagem de origem (a) e os cantos encontrados pelo método de Shi e Tomasi (b).

3.3.4 VETORES DE MOVIMENTO

Para que uma característica detectada possa ser rastreada, é necessário encontrar as correspondências nos *frames* seguintes. Uma vez que estão sendo utilizadas informações locais (pontos), pode-se utilizar o algoritmo de Lucas et al. (1981), conhecido também como Lucas-Kanade, que utiliza pontos para determinar o fluxo óptico.

Bradski e Kaehler (2008) descrevem o trabalho de Kanade-Lucas como um algoritmo que pode ser aplicado em um contexto esparso porque se baseia apenas em informações locais que são extraídas de uma pequena janela em torno de cada um dos

pontos de interesse. Em relação à janela mencionada, isso nos leva ao problema que o tamanho da janela pode impossibilitar que o ponto correspondente seja encontrado em um próximo *frame* devido à distância ser maior que a janela. Neste caso, temos o algoritmo de Kanade-Lucas piramidal, que começa a partir do mais alto nível de uma pirâmide de imagens (menor detalhe) e descendo para níveis inferiores (maior detalhe). Esta técnica nos permite obter vetores de movimento. Um vetor de movimento é descrito como um par de pontos conectados que tem em sua origem a posição de um determinado ponto em um *frame* anterior e como fim a sua posição correspondente no *frame* atual.

Deste modo, em nossa solução utilizamos o algoritmo piramidal de Lucas-Kanade, o qual recebe como entrada as imagens $f(x, y, t-1)$ e $f(x, y, t)$, e produz como saída as correspondências dos pontos de $f(x, y, t-1)$ em $f(x, y, t)$. Algumas restrições devem ser impostas para essa saída. Uma delas é em relação à distância percorrida pelo ponto entre $f(x, y, t-1)$ e $f(x, y, t)$. Para isso, a distância euclidiana é calculada entre os pontos nos diferentes *frames* e, a partir de uma distância mínima, o ponto é descartado ou considerado como válido para aquele determinado componente conexo. Além disso, é possível obter a direção que o ponto se movimentou.

A Figura 16 ilustra os vetores de movimento para cada canto rastreado na imagem. O agrupamento de todos os vetores de movimento de um componente conexo determinam o fluxo ótico do objeto.

Figura 16: Vetores de movimento de um ciclista e um veículo.

3.3.5 AJUSTE DE COMPONENTES CONEXOS

Tendo os vetores de movimento associados aos seus respectivos componentes conexos que também já foram filtrados baseados em suas dimensões máximas e mínimas, são realizados dois refinamentos: analisar a quantidade de vetores de movimento por componente conexo e uma análise estatística do agrupamento dos vetores de movimento parecidos entre si.

O primeiro método restringe-se apenas a validar se o total de vetores de movimento associados a um componente conexo é maior que um determinado valor fixo. Isto ajuda na eliminação de componentes conexos que possuam vetores de movimento muito esparsos. Em nossa abordagem, utilizamos um número de vetores n de modo que $n \geq 15$.

A segunda etapa consiste em calcular o desvio padrão dos deslocamentos em x e y dos vetores de movimento, a fim de obter um valor aceitável indicando que os vetores de movimento sejam parecidos entre si. Para isto, primeiramente calculamos as médias dos deslocamentos em x e y para cada um dos n vetores. Seja um vetor de movimento

contendo um par de pontos, temos os deslocamentos d_x e d_y. As Equações 15 e 16 são respectivamente as médias de deslocamentos em x e y.

$$\bar{d_x} = \frac{\sum_{i=1}^{n}(x'_i - x_i)}{n} \tag{15}$$

$$\bar{d_y} = \frac{\sum_{i=1}^{n}(y'_i - y_i)}{n} \tag{16}$$

A partir das médias de deslocamento calculadas, calculamos o desvio padrão para x (equação 17) e y (equação 18).

$$\sigma_x = \sqrt{\frac{\sum_{i=1}^{n}((x'_i - x_i) - \bar{d_x})^2}{n-1}} \tag{17}$$

$$\sigma_y = \sqrt{\frac{\sum_{i=1}^{n}((y'_i - y_i) - \bar{d_y})^2}{n-1}} \tag{18}$$

Em seguida, cada um dos vetores de movimento é validado pela seguinte expressão:

$$((x'_i - x_i - \bar{d_x}) \leq T\sigma_x) \wedge ((y'_i - y_i - \bar{d_y}) \leq T\sigma_y) \tag{19}$$

Quando a condição é satisfeita, os vetores de movimento são adicionados a uma lista. Consideramos σ_{max} um desvio padrão máximo aceitável para um conjunto de vetores de movimento onde se σ_x ou σ_y forem maiores que σ_{max} e o número de vetores de movimento adicionados à lista for igual ao número total de vetores de movimento do componente conexo, o valor do limiar T assume $\frac{T}{2}$ e uma nova iteração é realizada desde o processo em que é calculada a média de deslocamento de cada um dos vetores de movimento. Em nossa abordagem, adotamos o $\sigma_{max} = 1$ e $T = 3$ inicialmente. O decrescimento de T é feito de modo a diminuir a tolerância a cada iteração que os vetores de movimento apresentarem pouca diferença entre si. Essas operações podem ser visualizadas no Algoritmo 2 que tem como entrada uma lista de vetores de movimento, uma outra lista de vetores de movimentos a ser populada com os vetores semelhantes entre si e o valor inicial do limiar.

Neste pontos temos um conjunto de componentes conexos e um conjunto de vetores de movimento. Sendo assim, podemos fazer a associação de cada conjunto de vetores de movimento com o componente conexo correspondente.

Algoritmo 2 AjustBlobs

```
1: procedure AdjustBlobs(vectors_in, vectors_out, threshold)
2:     inliers
3:     done ← false
4:     while done == false do
5:         mean_dx ← calcMeanX(vectors_in)
6:         mean_dy ← calcMeanY(vectors_in)
7:         stddev_dx ← calcStdDevX(vectors_in, mean_dx)
8:         stddev_dy ← calcStdDevY(vectors_in, mean_dy)
9:         ok ← false
10:        while ok == false do
11:            ok ← true
12:            for (each vector in vectors_in) do
13:                dev_x ←|vector_end_x − vector_start_x − mean_dx|
14:                dev_y ←|vector_end_y − vector_start_y − mean_dy|
15:                if dev_x ≤ threshold ∗ stddev_dx and dev_y ≤ threshold ∗ stddev_dy
   then
16:                    add(inliers, vector)
17:                end if
18:            end for
19:            if count(inliers) == count(vectors_in) and (stddev_dx > MAX_stddev or
   stddev_dy > MAX_stddev) then
20:                removeAll(inliers)
21:                threshold ← threshold/2
22:                ok ← false
23:            end if
24:        end while
25:        if (stddev_dx > MAX_stddev or stddev_dy > MAX_stddev) then
26:            vectors_in ← inliers
27:            removeAll(inliers)
28:        else
29:            done ← true
30:        end if
31:    end while
32:    if count(inliers) == 0 then
33:        removeAll(vectors_in)
34:    else if count(inliers) > MIN_MOTION_VECTORS then
35:        vectors_out ← inliers
36:    end if
37: end procedure
```

3.4 RECONHECIMENTO DE CICLISTAS

Na etapa de segmentação dos componentes conexos, são encontrados os componentes que estão dentro das dimensões aceitáveis para que estes sejam considerados possíveis ciclistas. Porém, essa filtragem pelo tamanho não é suficiente para determinar se o objeto sendo rastreado é de fato um ciclista.

Como Pinto et al. (2008) afirmam, o maior problema do reconhecimento visual de objetos no mundo real é que, devido a mudanças de iluminação, posição, fundo e outras transformações espaciais, um objeto pode gerar um número tendendo ao infinito de imagens em duas dimensões. Apesar desse problema, o cérebro humano consegue realizar o reconhecimento sem esforço algum. Por isso, muitas soluções de reconhecimento de imagens são feitas para tentar emular as habilidades visuais dos humanos.

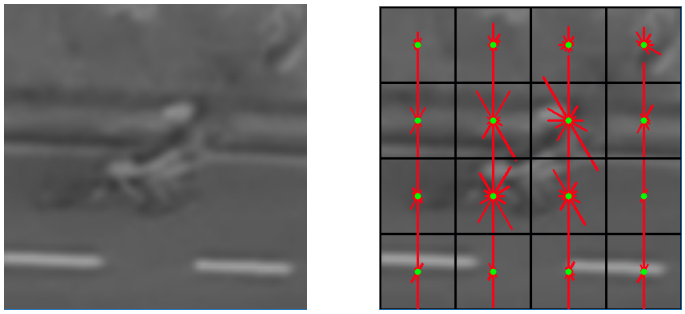
Descritores são, de maneira geral, métodos para caracterizar aspectos únicos de uma imagem, gerando um resumo que pode ser usado para representá-la. A saída dos algoritmos descritores normalmente é um vetor de números que pode ser comparado com um outro descritor para obter um grau de comparação dependendo da métrica usada (WINDER; BROWN, 2007). Esses aspectos podem ser originados de diversos atributos da imagem, como histogramas, estatísticas de cor, cantos e bordas. Porém, nem sempre essas características resumem a imagem de uma maneira robusta, pois estão sujeitas a diversas variações do ambiente. Um dos maiores desafios no reconhecimento de objetos usando visão computacional é encontrar quais são as características que diferem um objeto de outro. Para o reconhecimento destes em um cenário do mundo real, é necessário que os aspectos usados em um descritor tenham pelo menos robustez ou invariância a iluminação, perspectiva e outros tipos de transformações que podem ocorrer normalmente em objetos (LOWE, 2004).

O SIFT (LOWE, 2004) é um algoritmo de extração de características com uma abordagem baseada no comportamento de células complexas do córtex cerebral da visão dos mamíferos.

A abordagem do SIFT usada nesse trabalho é uma versão simplificada do algoritmo proposto por Lowe (2004). A região de interesse é previamente definida usando os métodos de detecção de movimento (ver Seção 3.3) e a filtragem de componentes conexos (ver Seção 3.3.2). A partir disso, o algoritmo faz uma descrição resumida das características da região encontrada. A região é dividida em subregiões e para cada uma delas é feito um histograma baseado na orientação dos gradientes. A quantidade de subregiões

e de orientações diferentes pode ser definida antes da execução do algoritmo. Devido a algumas simplificações no algoritmo para se adequar melhor à solução proposta, essa abordagem não possui invariância a rotação e escala, como no algoritmo clássico. Porém, estes fatores não são cruciais para a solução do problema proposto. Isso se deve ao fato de que é conhecido que os ciclistas presentes nos vídeos não possuem grandes variações de tamanho e que trafegam, na maior parte do tempo, em paralelo às ruas, ou seja, também não possuem grandes variações de rotação.

A saída do algoritmo SIFT é um vetor com os valores dos histogramas gerados, com $N \times M$ valores, onde N é o número de subregiões que a região de interesse foi dividida, e M é a quantidade de ângulos usados no histograma. Para obter robustez a variações de iluminação, os valores são normalizados. A Figura 17 exemplifica uma imagem de um ciclista dividida em 16 subregiões, sendo cada subregião representada com o seu respectivo histograma de orientações, baseados em 8 ângulos diferentes.



(a) (b)

Figura 17: Subregiões e histogramas (b) de um ciclista (a) geradas pelo SIFT.

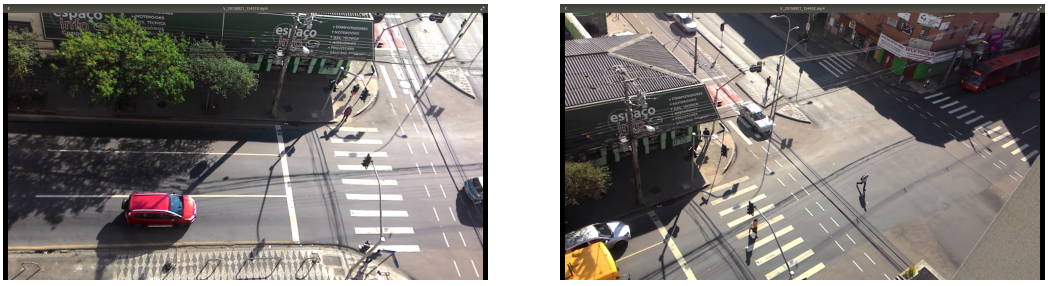
Os valores de saída do SIFT servirão de entrada para uma SVM (*Support Vector Machines*) (HEARST et al., 1998), que será usada como classificador para determinar quais valores representam um ciclista.

4 IMPLEMENTAÇÃO

Neste capítulo, será apresentado como a abordagem descrita foi implementada em um protótipo para a prova de conceito. A Seção 4.1 descreve o local e como os vídeos foram obtidos. A Seção 4.2 descreve como a biblioteca OpenCV foi utilizada para a implementação do protótipo. Na Seção 4.3 estão expostos os principais parâmetros utilizados na implementação da solução computacional proposta para este trabalho.

4.1 COLETA DE DADOS

Os dados utilizados para este trabalho são gravações de vídeo de uma avenida no centro da cidade de Curitiba/PR. As gravações foram realizadas a partir de uma vista superior do local, com o objetivo de englobar no vídeo a via compartilhada de carros e ciclofaixa, assim como a via dedicada para ônibus.



(a) (b)

Figura 18: Dois possíveis locais para realizar as gravações.

A Figura 18 mostra dois cenários cogitados para as gravações. Alguns problemas dificultaram o desenvolvimento da solução computacional e podem ser percebidos nestas imagens. São eles: ângulo da câmera para capturar a ciclovia e as outras vias e poluição visual obstruindo grande parte das áreas as quais são de suma importância (ciclovia e via do ônibus). Questões como intensa ou baixa luminosidade e sombras são inevitáveis em cenários abertos, como os das Figuras 18 e 19.

Figura 19: Outro possível local para as gravações.

A Figura 19 ilustra o segundo cenário cogitado para as gravações. Diversos aspectos nesta imagem levaram este local a ser o escolhido para as demais gravações, como: posicionamento da câmera; o campo de visão poderia ser facilmente extendido para englobar a via do ônibus juntamente com a ciclovia; vista superior, para este trabalho, facilita no processamento para detectar e rastrear objetos.

Figura 20: Vista superior do local para as gravações.

A Figura 20 ilustra um exemplo das gravações dos vídeos definitivos, utilizados na implementação e teste da prova de conceito. A partir disso, foram realizadas diversas gravações, em dias diferentes, conforme as necessidades de novos vídeos e imagens surgiram. Um problema enfrentado foi em relação ao equipamento utilizado para gravação, o

que acabou por produzir vídeos limitados em certos aspectos. Uma dessas limitações foi a grande distância da câmera para a rua, a qual fez com que os objetos a serem detectados e rastreados ficassem muito pequenos e com poucos detalhes.

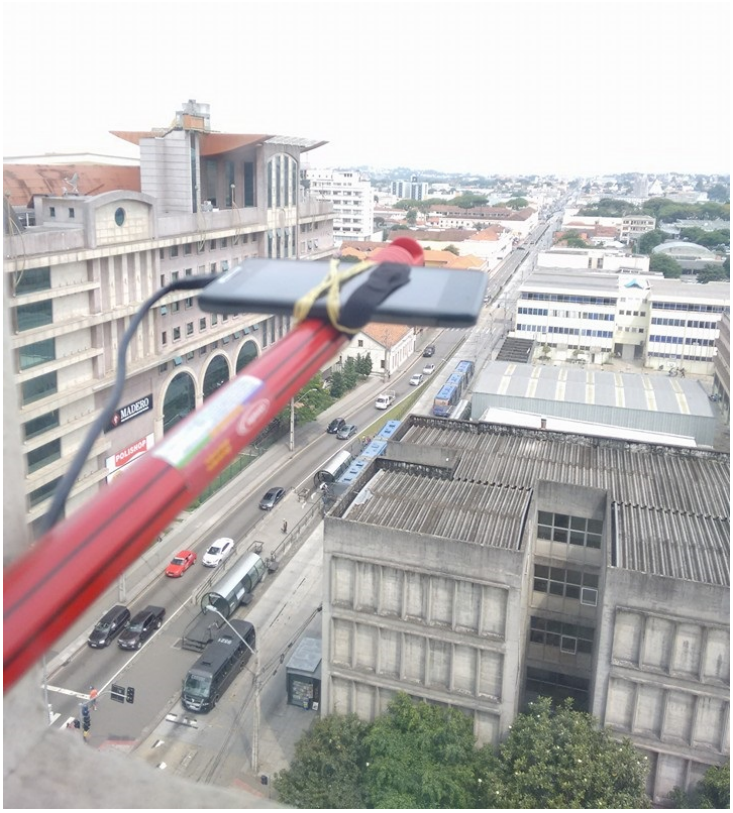


Figura 21: Primeiras gravações.

Outro problema encontrado foi em relação à estabilidade dos vídeos gravados. A Figura 21 ilustra uma das primeiras gravações realizadas com um equipamento improvisado. A partir de um certo momento, um novo equipamento passou a ser utilizado, que proporcionava maior estabilidade na câmera, que ficava presa em uma janela usando uma ventosa.

Os vídeos utilizados no experimento tinham como resolução 1920x1080 *pixels*, e foram divididos em duas categorias: vídeos utilizados para treino da SVM e vídeos utilizados para teste. Para treino, conforme descrito na Tabela 1, foram gravados 2 vídeos, totalizando aproximadamente 45 minutos de gravação.

Tabela 1: Características dos vídeos gravados para treino.

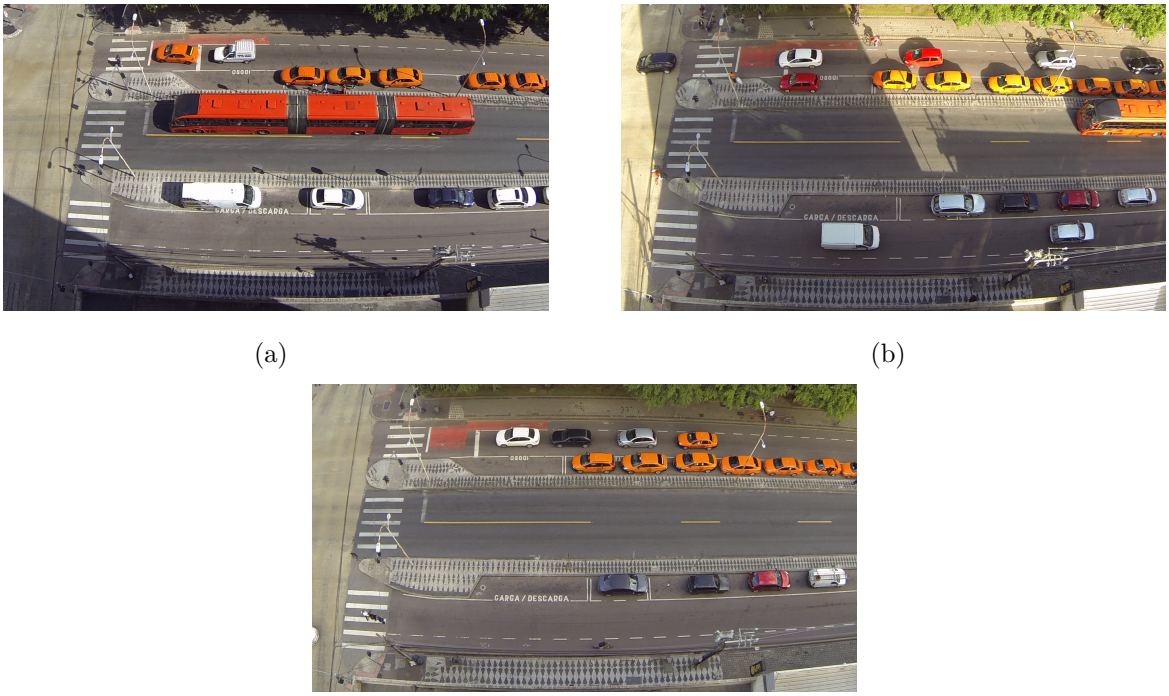
Vídeo	**Duração (minutos)**	**Resolução (*pixels* x *pixels*)**	***Frames* por segundo**	**Condição do tempo**
01	33	1920x1080	60	Pouco sol
02	12	1920x1080	60	Nublado
Total	**45**	**-**	**-**	**-**

A Tabela 2 apresenta as características dos vídeos utilizados para o teste. Deste modo, foram gravados 6 vídeos, totalizando aproximadamente 1 hora e 30 minutos de gravação (90 minutos). Estes 6 vídeos foram subdivididos em categorias, de acordo com a condição climática do dia das gravações.

Tabela 2: Características dos vídeos gravados para teste.

Vídeo	**Duração (minutos)**	**Resolução (*pixels* x *pixels*)**	***Frames* por segundo**	**Condição do tempo**
01	14	1920x1080	60	Muito sol
02	17	1920x1080	60	Muito sol
03	17	1920x1080	60	Pouco sol
04	17	1920x1080	60	Pouco sol
05	17	1920x1080	60	Pouco sol
06	08	1920x1080	60	Pouco sol
Total	**90**	**-**	**-**	**-**

As gravações foram realizadas no período da tarde, em diferentes dias, utilizando a mesma posição da câmera. A Figura 22 ilustra a categorização da condição climática observada nos vídeos gravados. Em alguns vídeos há uma presença alta de luz natural - ou seja, muito sol (Figura 22(a)) -, enquanto em outros vídeos haviam nuvens cobrindo a luz do sol (Figura 22(c)) ou sombras eram projetadas pelas edificações presentes nos arredores da região das gravações (Figura 22(b)). Cabe ressaltar que estas condições do tempo, como classificadas nas Tabelas 1 e 2 dizem respeito ao que foi observado como predominante nos vídeos, não excluindo o fato de que houveram pequenas mudanças climáticas durante as gravações. Não foram gravados vídeos em dias chuvosos.



(a) (b)

(c)

Figura 22: Categorização da condição climática presente nas gravações dos vídeos. Três condições foram percebidas nos vídeos, sendo: muito sol (a), pouco sol (b) e nublado (c).

Para obter o número de ciclistas presentes em cada vídeo gravado, foi feita uma contagem manual para todos os vídeos de teste. Dessa forma, o ponto de partida para que o protótipo possa ser avaliado se baseia nos dados desta contagem manual. A Tabela 3 detalha quantos ciclistas foram encontrados em cada um dos 6 vídeos gravados.

Tabela 3: Contagem manual dos ciclistas nos vídeos.

Vídeo	**Duração (minutos)**	**Ciclistas**
01	14	17
02	17	33
03	17	24
04	17	33
05	17	29
06	08	13
Total	**90**	**149**

Os vídeos 01 e 02 representam o conjunto 01 da Tabela 2, nos quais havia a

presença de muito sol durante as gravações. Os vídeos 03 ao 06 representam o conjunto 02 da Tabela 2, nos quais havia pouco sol.

4.2 OPENCV

OpenCV (Open Source Computer Vision Library) é uma biblioteca multiplataforma de código aberto desenvolvida pela Intel, sob a licença BSD. Sua utilização oferece suporte no desenvolvimento de aplicações na área de visão computacional para as linguagens de programação C, C++, Java e Phyton.

Atualmente em sua versão 3.0, a biblioteca conta com algoritmos tipicamente conhecidos e utilizados na literatura de visão computacional compreendendo algoritmos clássicos, algoritmos do estado da arte de visão computacional e algoritmos de aprendizado de máquinas. Estes algoritmos podem ser utilizados em situações que envolvem, por exemplo: reconhecimento e detecção de faces, identificação de objetos, rastreamento de objetos em movimento, extração de modelos em três dimensões de objetos, rastreamento de pessoas em vídeos e outros.

A biblioteca foi utilizada para a implementação dos métodos, na linguagem de programação C++. A versão utilizada do OpenCV foi a 2.4.13.

4.3 PARÂMETROS GERAIS DA SOLUÇÃO COMPUTACIONAL

4.3.1 DETECÇÃO DE MOVIMENTO

4.3.1.1 FILTRAGEM DOS COMPONENTES CONEXOS

A fim de eliminar ruídos e restringir os objetos a serem processados mais adiante no algoritmo, uma filtragem pelo tamanho dos componentes conexos foi necessária (ver subseção 3.3.2). As dimensões adotadas, após uma análise das imagens utilizadas na solução, foram fixadas em:

- Altura mínima: 20 *pixels*
- Altura máxima: 120 *pixels*
- Largura mínima: 30 *pixels*
- Largura máxima: 240 *pixels*

Dados estes parâmetros fixos, temos a definição do conjunto de componentes conexos:

$$\mathbb{L} = \{B \in L \mid 20 \leq B_h \leq 120 \wedge 30 \leq B_w \leq 240\} \tag{21}$$

Dessa forma, o conjunto $\mathbb{L}$, nesta etapa do algoritmo, é composto por apenas componentes conexos que estejam de acordo com as dimensões especificadas.

4.3.1.2 RASTREAMENTO DE PONTOS

Para o rastreamento dos objetos de interesse, primeiramente foi utilizado um método oferecido pelo OpenCV (ver subseção 4.2) para encontrar cantos que implementa a solução proposta por Shi e Tomasi (ver subseção 3.3.3). O método chamado de *GoodFeaturesToTrack* recebe como entrada uma imagem e como saída produz uma lista de pontos com as *features* encontradas. Além disto para este método foram utilizados outros parâmetros como: número máximo de cantos a serem encontrados $Corner_{Max} = 0$ de forma que quando houver mais cantos do que os encontrados os mais fortes são retornados, uma qualidade mínima aceita $Corner_{Threshold} = 0.001$, uma distância mínima entre os cantos $Corner_{MinDist} = 0$, a máscara de movimento (ver 3.3.1) utilizada para encontrar cantos apenas em regiões de interesse e o tamanho da janela $Corner_{WinSize} = 5$.

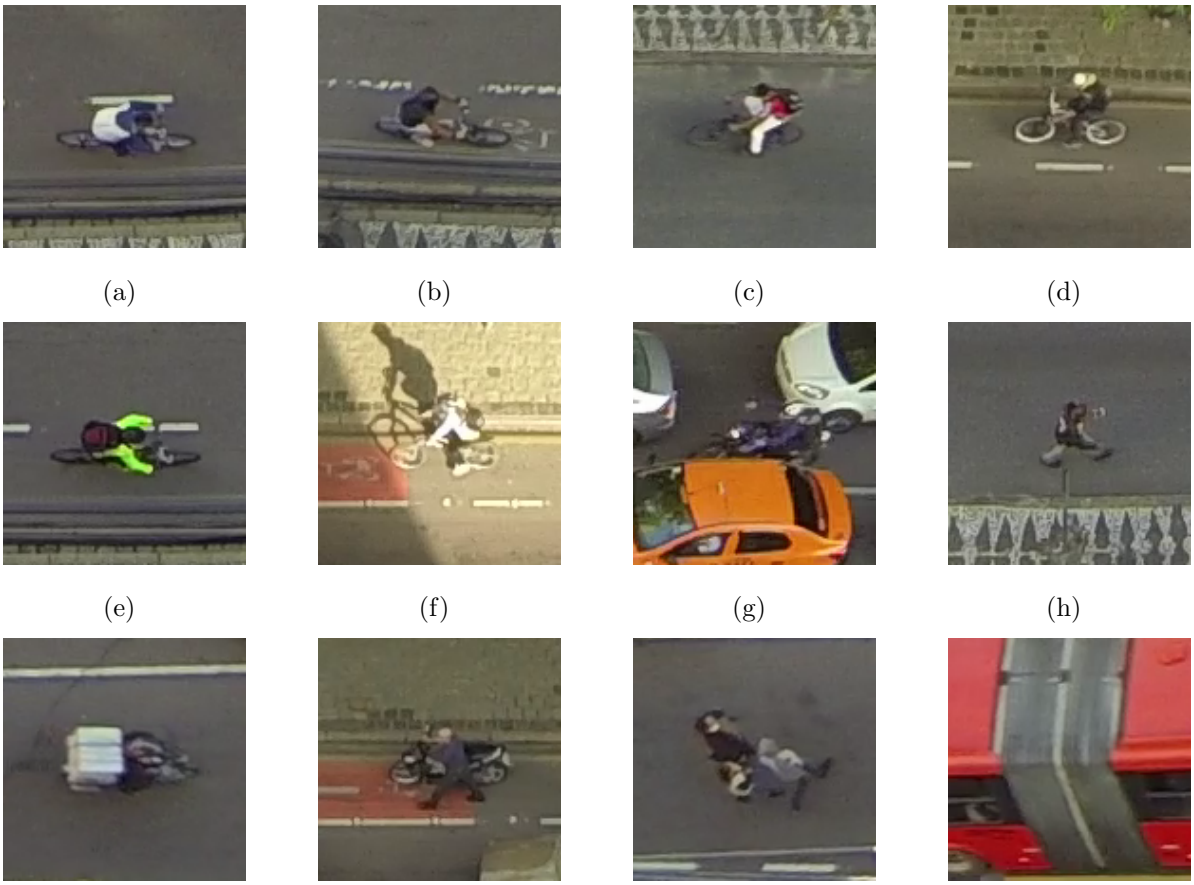
4.3.2 VETORES DE MOVIMENTO

A obtenção dos vetores de movimento se dá pela utilização do método *calcOpticalFlowPyrLK* que implementa a versão piramidal do algoritmo de Lucas-Kanade (ver subseção 3.3.4). Os parâmetros utilizados como entrada por este método são as duas imagens $f(x, y, t-1)$ e $f(x, y, t)$, os pontos de interesse encontrados em $f(x, y, t-1)$ pelo *GoodFeaturesToTrack*, e como saída uma lista contendo os pontos correspondentes encontrados em $f(x, y, t)$. Além disto, o método utiliza outros parâmetros os quais optou-se por utilizar seus valores padrão. Dentre eles o tamanho da janela W em que é feita a busca de um ponto, definidos com largura $W_{width} = 21$ e altura $W_{height} = 21$. Neste método é possível definir quantos níveis serão utilizados para a pirâmide de imagens geradas pelo algoritmo piramidal, definidos em $K = 3$.

4.3.3 DESCRIÇÃO E CLASSIFICAÇÃO DE CICLISTAS

A partir das regiões de interesse encontradas nos passos anteriores, é necessário que estas sejam descritas e determinadas, para diferenciar os ciclistas de outros elementos. A abordagem do algoritmo SIFT usada (ver Seção 3.4) recebe uma região de interesse a ser descrita. Essa região possui 84 *pixels* de largura por 84 de altura, dividida em 144 regiões (uma *grid* com 12x12 sub-regiões), e calculando um histograma para 12 orientações diferentes, totalizando 1728 valores de saída para o descritor. Esses valores foram definidos desta maneira para que o descritor gere um resultado contendo mais detalhes.

Na etapa de classificação dos ciclistas, o treinamento do SVM foi realizado usando 87 imagens de exemplos positivos, e 87 imagens para exemplos negativos, que foram extraídos de alguns dos vídeos obtidos (ver Seção 4.1). A Figura 23 mostra seis exemplos positivos para a SVM (a até f) e seis exemplos negativos (g até l). Essas imagens foram descritas pelo SIFT, e seus vetores de 1728 valores serviram de entrada para a classificação.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

Figura 23: Exemplos positivos (a até f) e exemplos negativos (g até l) para o treinamento da SVM.

5 EXPERIMENTO

Neste capítulo estão expostos os principais resultados obtidos a partir da aplicação da abordagem proposta. A Seção 5.1 apresenta os resultados obtidos da contagem feito pelo protótipo desenvolvido. As Seções 5.2 e 5.3 apresentam, respectivamente, as métricas para avaliar os resultados e a avaliação do protótipo considerando os resultados obtidos.

5.1 RESULTADOS

A partir da contagem manual, os mesmos vídeos listados na Tabela 3 foram utilizados como entrada para o protótipo desenvolvido. A saída principal do protótipo é o número de ciclistas encontrados em cada vídeo.

Figura 24: Região de contagem do protótipo desenvolvido. A linha em amarelo identifica o ponto em que um objeto é contado caso esteja classificado como ciclista.

A Figura 24 ilustra como a contagem foi feita no protótipo. A linha em amarelo

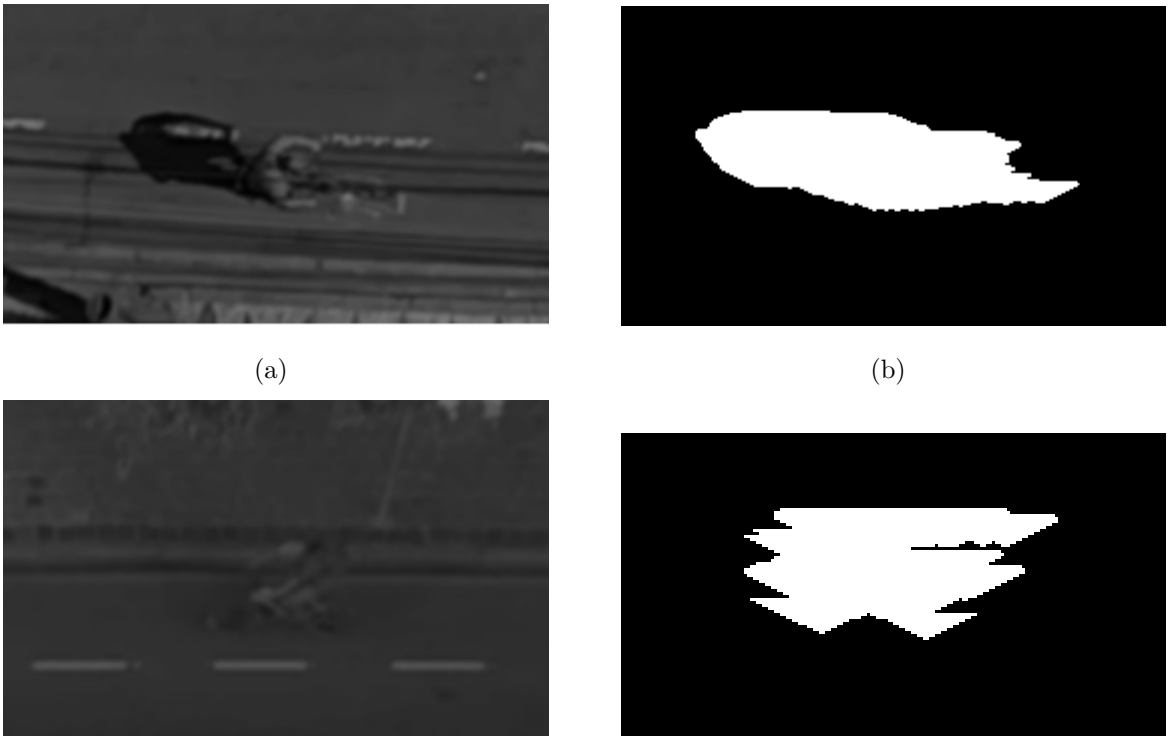
representa o ponto de contagem, ou seja, os objetos só são contados caso cruzem pela região e estejam classificados como ciclistas pelo protótipo. Apesar de ser possível evitar alguns falsos-positivos utilizando esta abordagem para contagem, alguns outros problemas se derivam deste método, como: ciclistas sendo detectados fora da região de contagem, e na região não ser identificado; o ponto de contagem não muda para os diferentes vídeos e, por consequência, prejudicou alguns vídeos que tem mais interferência da luz do sol, pois o classificador demora mais para identificar um ciclista.

A Tabela 4 detalha a contagem feita pelo protótipo desenvolvido. Alguns pontos importantes foram percebidos nos vídeos gravados e valem a pena ser mencionados junto aos resultados, pois dificultaram a classificação dos objetos como ciclistas. O primeiro ponto é em relação à luz do sol. Apesar de o SIFT ser robusto a variações de iluminação, a luz do sol se mostrou um obstáculo no sentido que destaca mais as sombras dos objetos, dependendo da posição da objeto em relação a câmera.

Tabela 4: Contagem feita pelo protótipo nos vídeos em relação a contagem manual.

Vídeo	**Contagem protótipo (núm. de ciclistas)**	**Contagem manual (núm. de ciclistas)**
01	7	17
02	21	33
03	17	24
04	8	33
05	25	29
06	12	13
Total	**90**	**149**

A Figura 25 ilustra esse caso. Na Figura 25(a) há um ciclista na ciclofaixa de baixo e seu componente conexo gerado pelo protótipo em 25(b). Na Figura 25(c) e 25(d) há um ciclista na ciclofaixa de cima, e seu componente conexo, respectivamente. Percebe-se a diferença no padrão dos "*blobs*" gerados, no qual o primeiro é mais largo e achatado e no segundo a largura e altura são mais proporcionais.



(a) (b)

(c) (d)

Figura 25: Variação do tamanho do componente conexo de acordo com a quantidade de luz do sol presente no vídeo.

O problema gerado pelas sombras é o de que elas formam um componente conexo muito maior do que o esperado e observado em vídeos com pouca influência da luz do sol, pois os componentes do ciclista e da sombra se juntam e formam um só. Assim, as dimensões do componente conexo, em grande parte das vezes, ultrapassam as dimensões esperadas de um ciclista. Uma possível solução para isto seria aumentar as dimensões para filtrar os ciclistas, no entanto, este parâmetro é muito sensível em todos os vídeos, pois há outros objetos que podem passar a ser considerados na classificação se as dimensões forem alteradas (tanto para menor quanto para maior).

Uma outra consequência que vale a pena ser mencionada em relação a influência da luz do sol, é a de que ela interfere diretamente nas características que se destacam em cada *frame* do vídeo. Assim, os descritores computados para as regiões de interesse são afetados, pois a luz do sol gera gradientes mais "fortes" em determinadas regiões, as quais possivelmente não teriam pontos de interesse descritas pelo algoritmo do SIFT caso houvesse uma influência menor da luz do sol.

(a) (b)

Figura 26: Sombras dos objetos projetadas e definição da região de interesse errada.

O segundo ponto a ser mencionado é em relação a grande variação de detalhes nos vídeos. Dado que o número de exemplos para treino da SVM com o SIFT modificado não foi muito grande, o classificador obteve dificuldades para distinguir os objetos em regiões que apresentavam muitas características. A Figura 27 ilustra a região de interesse - a qual contém um ciclista -, enviada para o classificador. Pode-se perceber que a região acima do ciclista influenciou nos resultados do classificador pois o SIFT a considerou como uma possível região de interesse.

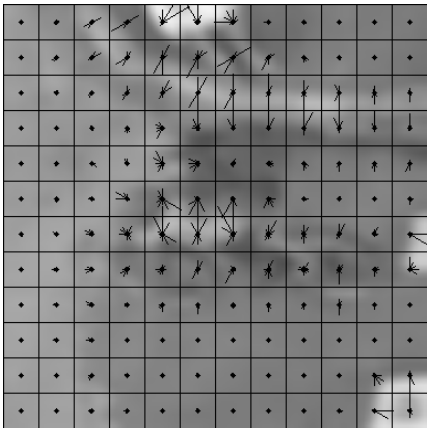


Figura 27: Subregiões e valores dos histogramas gerados pelo SIFT modificado de uma região caótica.

Dessa forma, diversas regiões ao longo do vídeo influenciaram negativamente o processo de classificação, tais como: faixa de pedestre; faixas da via dos carros e faixas da ciclofaixa; calçada.

5.2 MÉTRICAS

As métricas utilizadas para avaliação dos resultados partem do pressuposto de que existem duas classes distintas presentes no resultados do algoritmo: ciclistas (positiva) e não ciclistas (negativa).

A Tabela 5 detalha a categorização das classes de acordo com os possíveis resultados do algoritmo. As categorias utilizadas são: verdadeiro positivo (VP), verdadeiro negativo (VN), falso positivo (FP) e falso negativo (FN).

	Categorias	
Resultado algoritmo (classes)	**Positivo**	**Negativo**
Ciclista (positivo)	VP	FP
Não ciclista (negativo)	FN	VN

Tabela 5: Critério para avaliação das classes baseado em categorias.

A fim de avaliar os resultados obtidos pelo algoritmo em relação aos dados observados manualmente, foram utilizadas as seguintes métricas, especificadas nas Equações 23 e 24: sensibilidade e precisão (proporção de verdadeiros positivos).

Sensibilidade (ou *recall*), dada por S é a proporção de verdadeiros positivos dentre o total de verdadeiros:

$$S = \frac{VP}{VP + FN} \tag{23}$$

Sensibilidade é a medida em relação ao quanto o classificador foi capaz de acertar se uma imagem é um ciclista, dado que seja um ciclista.

Precisão, dada por P é a proporção de verdadeiros positivos em relação a todas as classificações positivas:

$$P = \frac{VP}{VP + FP} \tag{24}$$

A precisão é importante no sentido que quantifica o quanto o classificador acertou em classificar a imagem de um ciclista como um ciclista nos vídeos.

5.3 AVALIAÇÃO

A partir da utilização das métricas definidas na Seção 5.2, foi possível averiguar a precisão e sensibilidade do protótipo desenvolvido. A Tabela 6 expõem os principais resultados e a aplicação das métricas para cada um dos 6 vídeos.

Tabela 6: Aplicação das métricas de avaliação nos resultados do protótipo.

Vídeo	**Contagem protótipo (núm. de ciclistas)**	**Contagem manual (núm. de ciclistas)**	**VP**	**FP**	**FN**	**Sensibilidade (%)**	**Precisão (%)**
01	7	17	4	3	13	24	57
02	21	33	13	8	20	39	62
03	17	24	9	8	15	38	53
04	8	33	5	4	28	15	56
05	25	29	9	16	20	31	36
06	12	13	8	4	5	62	67
MÉDIA	**15**	**24,8**	**8**	**7,1**	**16,8**	**34,7**	**55**

É importante perceber a diferença na contagem manual para a contagem feita pelo protótipo. Uma simples análise com base nesta diferença não é o suficiente para quantificar a qualidade do protótipo, afinal diversos fatores devem ser considerados, tais quais já foram mencionados em seções anteriores como possíveis obstáculos para o classificador. No entanto, tal diferença evidência uma falha no protótipo em classificar ciclistas presentes nos vídeos. A partir desta constatação, três possíveis motivos se fazem presente na avaliação desse resultado: (a) mudança drástica nas dimensões dos ciclistas, pois estas são sensíveis a iluminação (por conta da sombra) e também mudam conforme a posição do ciclista em relação a câmera, o que ocasiona na não classificação destes como sendo ciclistas; (b) ponto de contagem fixo e com uma região muito pequena pode ocasionar de não classificar alguns ciclistas exatamente naquele espaço definido, mesmo que a detecção e rastreamento do ciclista tenha sido feita antes ou depois da linha de contagem; (c) ciclofaixa próxima da calçada ocasiona que alguns componentes conexos dos ciclistas se juntem com os de pedestres que estão ali muito próximos, o que ocasiona no problema das dimensões do componente ultrapassarem as medidas esperadas de um ciclista.

Os principais problemas percebidos no vídeo 01 - o qual falhou em detectar di-

versos ciclistas - foram os já mencionados anteriormente. No vídeo 04, o qual também falhou em classificar diversos ciclistas -contagem do protótipo foi de 8 enquanto contagem manual foi 33, como evidenciado na Tabela 6-, não havia influência da luz do sol. No entanto, neste vídeo houve uma predominância de ciclistas que trafegaram pela ciclofaixa de baixo, a qual se mostrou muito problemática para detectar e principalmente, rastrear os ciclistas. O principal obstáculo observado foi os fios dos postes de luz, os quais obstruíam parcialmente a ciclofaixa, gerando "ruídos" nas imagens.

Em alguns vídeos também foi percebido a falha na distinção de ciclistas para motociclistas. Isso é consequência do baixo número de imagens de treino para a SVM. No vídeo 03, por exemplo, algumas motos foram classificadas como sendo ciclistas. O vídeo 06 foi o que apresentou os melhores resultados. Isso se deve a alguns fatos: vídeo com pouca influência da luz do sol; ciclistas não andavam muito perto uns dos outros, o que facilitou na classificação e rastreamento; e movimento de tráfego baixo, o que diminuiu a taxa de erro do classificador.

6 CONCLUSÃO

Neste trabalho foi apresentada uma solução para detectar e rastrear ciclistas usando métodos de visão computacional. De um ponto de vista de aplicação, a solução encontrada pode servir para estudos relacionados ao planejamento urbano com foco no espaço em que o ciclista ocupa na cidade, auxiliando na prevenção de acidentes e facilitando o trânsito em grandes áreas urbanas de maneira geral, a partir das contribuições geradas pelo uso de um sistema de monitoramento eficaz.

A utilização do algoritmo SIFT modificado em conjunto com a SVM se mostrou dinâmico o suficiente para que os resultados possam ser melhorados a partir de uma melhoria da base de entrada, ou seja, da qualidade dos vídeos apresentados ao algoritmo.

Além, disso, a solução proposta também poderia se aplicar a outros tipos de modais, considerando os métodos usados. Algumas mudanças nos parâmetros poderiam render resultados interessantes para outros tipos de meio de transporte, ou até mesmo para pedestres.

Os resultados encontrados nos experimentos não foram tão satisfatórios, no entanto, o estudo e averiguação dos motivos relacionados as falhas contribuem para que trabalhos futuros possam se aproveitar das contribuições apresentadas neste trabalho. Assim, como um possível estudo futuro, este trabalho poderia servir como base para o desenvolvimento de uma solução computacional mais robusta, levando em conta os apontamentos feitos aqui: realizar experimentos com vídeos que contenham ciclistas mais próximos da câmera, para que as características detectadas pelo SIFT sejam mais relevantes, assim como o aperfeiçoando de algumas fases da abordagem proposta, como a classificação usando a SVM, que necessitaria de mais exemplos de entrada para ter resultados mais precisos.

O uso de redes neurais convolucionais aplicadas no mesmo contexto deste trabalho pode se mostrar mais vantajoso e obter resultados mais satisfatórios. Isso se deve ao fato de que o algoritmo do SIFT é facilmente influenciado por “ruídos” na imagem,

como sombras, faixas e fiação elétrica, enquanto uma DCNN (*Deep Convolutional Neural Network*) poderia reconhecer os padrões de um ciclista sem ser muito afetada por outros elementos na imagem.

Um outro possível estudo futuro que possui relação com este trabalho, e que se torna relevante diante do contexto aqui estudado, é a forma como as ciclovias são utilizadas pelas pessoas, partindo de um ponto de vista da área de planejamento urbano.

REFERÊNCIAS

BABU, R. V.; RAMAKRISHNAN, K. Recognition of human actions using motion history information extracted from the compressed video. **Image and Vision computing**, Elsevier, v. 22, n. 8, p. 597–607, 2004.

BAY, H.; TUYTELAARS, T.; GOOL, L. V. Surf: Speeded up robust features. **ECCV**, p. 404–417, 2006.

BEAUGENDRE, A. et al. Enhanced moving object detection using tracking system for video surveillance purposes. In: IEEE. **Visual Communications and Image Processing (VCIP), 2012 IEEE**. [S.l.], 2012. p. 1–6.

BOBICK, A. F.; DAVIS, J. W. The recognition of human movement using temporal templates. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, v. 23, n. 3, p. 257–267, 2001.

BRADSKI, G.; KAEHLER, A. **Learning OpenCV: Computer vision with the OpenCV library**. [S.l.]: "O'Reilly Media, Inc.", 2008.

BUCH, N.; VELASTIN, S. A.; ORWELL, J. A review of computer vision techniques for the analysis of urban traffic. **Intelligent Transportation Systems, IEEE Transactions on**, IEEE, v. 12, n. 3, p. 920–939, 2011.

COIFMAN, B. et al. A real-time computer vision system for vehicle tracking and traffic surveillance. **Transportation Research Part C: Emerging Technologies**, Elsevier, v. 6, n. 4, p. 271–288, 1998.

DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. **CVPR**, p. 886–893, 2005.

GONZALEZ, R. C.; WOODS, R. E. **Processamento de imagens digitais**. [S.l.]: Edgard Blucher, 2000.

HAMILTON-BAILLIE, B. Shared space: reconciling people, places and traffic. **Built environment**, Alexandrine Press, v. 34, n. 2, p. 161–181, 2008.

HARRIS, C.; STEPHENS, M. A combined corner and edge detector. In: CITESEER. **Alvey vision conference**. [S.l.], 1988. v. 15, p. 50.

HEARST, M. A. et al. Support vector machines. **IEEE Intelligent Systems and their Applications**, IEEE, v. 13, n. 4, p. 18–28, 1998.

LABPRODAM. **Laboratório de Inovação da Prefeitura de São Paulo - Contador de Ciclistas**. 2016. Disponível em: <http://saopauloaberta.prefeitura.sp.gov.br/index.php/iniciativa/contador-de-ciclistas>. Acesso em: 30 de outubro de 2016.

LOWE, D. G. Distinctive image features from scale-invariant keypoints. **International journal of computer vision**, Springer, v. 60, n. 2, p. 91–110, 2004.

LUCAS, B. D.; KANADE, T. et al. An iterative image registration technique with an application to stereo vision. In: **IJCAI**. [S.l.: s.n.], 1981. v. 81, n. 1, p. 674–679.

LUVIZON, D. C.; NASSU, B. T.; MINETTO, R. Vehicle speed estimation by license plate detection and tracking. **IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**, p. 6563–6567, 2014.

MATHEW, T. V. **Intrusive and Non-Intrusive Technologies. Technical report**. 2014.

MEDEIROS, R. M.; DUARTE, F. Policy to promote bicycle use or bicycle to promote politicians? bicycles in the imagery of urban mobility in brazil. **Urban, Planning and Transport Research**, Taylor & Francis, v. 1, n. 1, p. 28–39, 2013.

MESSELODI, S.; MODENA, C. M.; CATTONI, G. Vision-based bicycle/motorcycle classification. **Pattern Recognition Letters**, v. 28, p. 1719–1726, 2007.

OJALA, T.; PIETIKAINEN, M.; HARWOOD, D. A comparative study of texture measures with classification based on featured distributions. **Pattern Recognition**, v. 29, no. 1, p. 51–59, 1996.

PINTO, N.; COX, D. D.; DICARLO, J. J. Why is real-world visual object recognition hard? **PLoS Comput Biol**, Public Library of Science, v. 4, n. 1, p. e27, 2008.

RAFIQ, H.; SIDDIQI, M. Haar transformation of linear boolean function. **2009 International Conference on Signal Processing Systems**, p. 802–805, 2009.

SAYED, T.; ZAKI, M. H.; AUTEY, J. Automated safety diagnosis of vehicle–bicycle interactions using computer vision analysis. **Safety science**, Elsevier, v. 59, p. 163–172, 2013.

SHAHEEN, S. A.; GUZMAN, S.; ZHANG, H. Bikesharing in europe, the americas, and asia. **Transportation Research Record: Journal of the Transportation Research Board**, v. 2143, p. 159–167, 2010.

SHI, J.; TOMASI, C. Good features to track. In: IEEE. **Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on**. [S.l.], 1994. p. 593–600.

SILVA, E. R. da. **Análise do crescimento da motorização no Brasil e seus impactos na mobilidade urbana**. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, 2011.

SILVA, R. et al. Automatic motorcycle detection on public roads. **Clei Electronic Journal**, v. 16, number 03, paper 04, 2013.

TEKALP, A. M. **Digital video processing**. [S.l.]: Prentice-Hall, Inc., 1995.

WINDER, S. A.; BROWN, M. Learning local image descriptors. In: IEEE. **2007 IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.], 2007. p. 1–8.