

# **Otimização do Fluxo de Veículos: Contagem Volumétrica Automatizada**

E. Krakauer

H. Pedrini

Relatório Técnico - IC-PFG-18-23  
Projeto Final de Graduação  
Dezembro - 2018

UNIVERSIDADE ESTADUAL DE CAMPINAS  
INSTITUTO DE COMPUTAÇÃO

The contents of this report are the sole responsibility of the authors.  
O conteúdo deste relatório é de única responsabilidade dos autores.

# Otimização do Fluxo de Veículos: Contagem Volumétrica Automatizada

Eric Krakauer<sup>1</sup>, Hélio Pedrini<sup>2</sup>

<sup>1</sup>  
Aluno, Instituto de Computação, Universidade Estadual de Campinas (UNICAMP), Campinas, SP

<sup>2</sup>  
Orientador, Instituto de Computação, Universidade Estadual de Campinas (UNICAMP), Campinas, SP

**Resumo.** O trânsito é um dos maiores problemas enfrentados em grandes centros urbanos e diversos problemas estão associados à ineficiência dos semáforos. Assim, este estudo contempla um projeto para otimizar o fluxo de veículos em cruzamentos semaforizados ao apresentar e avaliar as melhores formas para realizar inovações nos semáforos a fim de reduzir congestionamentos, além de propor um modelo de contagem classificada automatizada utilizando visão computacional. Para isso, pesquisas serão conduzidas com Prefeituras, como as de Paulínia e Americana, e experimentos de campo, além de simulações de tráfego, a fim de chegar a um modelo de sistema possível para otimizar o trânsito em semáforos.

**Palavras-Chave:** Semáforo; Otimização Computacional; Visão Computacional; Contagem Classificada e Engenharia de Tráfego.

**Abstract.** Traffic is a recurrent problem in large cities and many of those issues are associated with the bottleneck in the traffic lights system that is really inefficient. Therefore, this study presents a model to optimize the vehicle flow in traffic lights that analyze and evaluate the best ways to change paradigm and reduce the huge lines in the roads and streets. In addition, this work proposes a computer vision method to do automatic vehicle count. For that, researches will be conducted with some City Halls, such as Paulínia and Americana, and field experiments, due to the reason to build a system model capable to optimize vehicles flow in traffic lights.

**Key Words:** Traffic Light; Computer Optimization; Computer Vision; Qualified Count e Traffic Engineering.

# **Agradecimentos**

Este trabalho representa a conclusão de uma etapa e o início de uma nova fase da minha vida. Assim dedico o trabalho, especialmente, à minha família que me apoiou e ajudou em todos os momentos a fim de conquistar meus objetivos.

Também agradeço aos meus amigos, que tanto me ajudaram ao longo do curso em estudos conjuntos, iniciativas extra-curriculares e conquistas, e o corpo docente da Unicamp e Colégio Rio Branco que me proporcionaram a oportunidade de aprender e ensinar.

# Lista de Figuras

Figura 1: Controlador Semafórico em um semáforo de Paulínia.....	16
Figura 2: (a) Dispositivo para realizar a programação semafórico/(b) Plano do cruzamento .....	17
Figura 3: (a) Controlador Semafórico / (b) Dispositivo para a programação de semáforo idêntico ao da Figura 2(b).....	18
Figura 4: Cruzamento entre a Av. Doutor Luiz da Tela e Av. Oscar Pedroso .....	21
Figura 5: Cruzamento entre Av. Santa Izabel e a R. Antônio Piezori. ....	24
Figura 6: Exemplo de cruzamento com 3 fluxos, mas apenas 2 tempos, devido aos fluxos B e C não se cruzarem. ....	29
Figura 7: Aplicação no Sumo Traffic .....	31
Figura 8: Simulação no Distrito de Barão Geraldo (Campinas - SP) .....	32
Figura 9: Exemplo da interface de definições de situações do trânsito em Boston.....	33
Figura 10: Simulação nos arredores do MIT em Boston (Estados Unidos).....	33
Figura 11: Simulação na cidade de Beijing (China) .....	34
Figura 12: Modelo em alto nível do funcionamento do sistema.....	35
Figura 13: Overview da Solução.....	36
Figura 14: Esquema do Módulo de Semáforos Inteligentes .....	37
Figura 15: Diagrama do algoritmo do controlador .....	39
Figura 16: Imagem utilizada como Entrada do Algoritmo .....	51
Figura 17: Figura 16 convertida para a Escala de Cinza.....	52
Figura 18: Imagem com o histograma equalizado e com o Filtro Gaussiano aplicado .....	52
Figura 19: Comparação entre a imagem original na escala de cinza(a) e a imagem de saída com o filtro Gaussiano(b) .....	52
Figura 20: Testes realizados por Gupta[18] para a identificação da imagem de fundo de referência. ....	55
Figura 21: Imagem com os contornos dos objetos traçados .....	59
Figura 22: Imagem de referência de um carro .....	59
Figura 23: Imagem de referência de um carro com o operador de Canny.....	59
Figura 24: Imagem de referência de um ônibus.....	60
Figura 25: Imagem de referência de ônibus com o operador de Canny .....	60
Figura 26: Imagem de referência de um caminhão.....	60
Figura 27: Imagem de referência de um caminhão com o operador de Canny .....	60
Figura 28: Exemplo de funcionamento do Yolo .....	62
Figura 29: Objetos detectados com Yolo.....	63
Figura 30: Comparativo do Yolo com outros sistemas de classificação.....	64
Figura 31: Figura utilizada para o teste de uma rede neural treinada. ....	66
Figura 32: Resultado da execução do Yolo para a Figura 38 no Terminal.....	67
Figura 33: Imagem gerada pelo processamento da Figura 38 .....	67
Figura 34: Imagem de teste para representar uma situação real de trânsito. ....	67
Figura 35: Resultado da execução do sistema para a Figura 33 .....	68
Figura 36: Imagem de saída do algoritmo para a Figura 33 .....	69

## Lista de Tabelas

Tabela 1: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 1 está aberto.....	22
Tabela 2: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 2 está aberto.....	22
Tabela 3: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 3 está aberto.....	23
Tabela 4: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 1 está aberto.....	24
Tabela 5: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 2 está aberto.....	25
Tabela 6: Tempo médio de otimização para ambos os testes .....	26
Tabela 7: Medidas padrões de automóveis .....	56
Tabela 8: Medidas de Caminhões .....	57
Tabela 9: Resultado da Segmentação.....	61
Tabela 10: Speedup obtido pela utilização do Yolo perante outros sistemas .....	64

# Lista de Algoritmos

Algoritmo 1: Processamento Inicial da Imagem.....51

Algoritmo 2: Definir Imagem de fundo .....54

Algoritmo 3: Segmentação do Objeto.....58

# Sumário

1.Introdução.....	8
1.1 Justificativa.....	9
2. Objetivos.....	10
3. Fundamentos Teóricos de Engenharia de Tráfego.....	11
3.1. Programação do Semáforo.....	11
3.2 Alternativas Existentes ao modelo atual.....	12
4. Pesquisa de Campo.....	15
4.1 Visita à Operação de Trânsito de Paulínia.....	15
4.2 Visita à Operação de Trânsito de Americana.....	17
4.3 Entrevista com o Secretário de Trânsito de Piracicaba.....	19
4.4 Entrevista com o Secretário de Trânsito de Campinas.....	20
5. Estudo de Campo em Barão Geraldo .....	21
5.1 Estudo no Cruzamento Av. Oscar Pedroso Horta e Av Dr. Luís Tela.....	21
5.2 Estudo no Cruzamento da Av.Santa Isabel e R. Antônio Pierozi.....	24
5.3 Análise Comparativa.....	26
6. Solução Proposta .....	27
6.1 Solução.....	27
6.2 Segmento de Clientes.....	27
6.3 Canais.....	28
7. Sistema Proposto .....	29
7.1 Planejamento de Desenvolvimento.....	30
7.2 Simulações de Tráfego.....	31
7.3 Modelagem do Sistema.....	34
8. Um estudo sobre Processamento de Imagens .....	43
8.1 Imagem Digital.....	43
8.2 Escala de Cinza.....	43
8.3 Contraste.....	44
8.4 Filtros.....	45

8.5 Operadores de Bordas.....	46
8.6 Operadores Morfológicos.....	46
9. Aplicações de Visão Computacional.....	48
9.1 Subtração de Fundo.....	48
9.2 Aprendizado de Máquina.....	49
9.3 Modelo de Borda Correspondente.....	49
9.4 Rastreamento e Agrupamento de Pontos Correspondentes.....	49
10. Algoritmo Proposto .....	50
10.1 Método de Identificação de Veículos.....	50
10.2 Sistema de detecção de objetos em tempo real.....	62
11. Conclusão .....	70
Referências .....	72



# 1. Introdução

Nas médias e grandes cidades brasileiras, observa-se um grande problema de mobilidade: o trânsito. Para se ter uma dimensão do problema, segundo estudos da TomTom, atualmente o Rio de Janeiro é a oitava cidade mais congestionada do mundo, enquanto São Paulo perde mais de R\$ 156,2 bilhões com a morosidade do trânsito [1], uma vez que os paulistanos desperdiçam aproximadamente 100 minutos diários em deslocamentos, algo que afeta diretamente a produtividade do trabalhador.

Parte desse problema é causado pelo grande volume de veículos, pois a frota de automotores no Brasil é de mais 68 milhões [2], porém outra parcela é resultado de uma deficiente gestão do tráfego urbano.

Por exemplo, os semáforos são incapazes de otimizar o fluxo de veículos nas ruas e avenidas, isto é, o tempo em que o semáforo permite o fluxo em determinado sentido é incompatível com o volume de automóveis que esperam para passar na outra seção da via.

Tal situação é decorrente do fato da programação semafórica ser realizada de maneira manual, ou seja, um funcionário faz a contagem dos veículos e estima fixamente o tempo de abertura. Em cruzamentos mais movimentados, realiza-se o mesmo procedimento manual em diferentes horários do dia, algo que é extremamente custoso para a prefeitura, mas que permite suprir as oscilações diárias do tráfego [32].

Entretanto, analisando as oscilações que ocorrem em diferentes dias da semana ou resultantes da variação do fluxo de pessoas, como vésperas de feriados ou a abertura de um novo shopping, o método manual torna-se impraticável, já que a atualização frequente desses tempos demandaria um grande número de funcionários e, conseqüentemente, um gasto inviável.

Outro problema relevante é o chamado 'Verde Ocioso', isto é, quando o semáforo permite a passagem para uma via sem veículos transitando, enquanto a via concorrente fica com os seus veículos congestionados. Tal situação ocorre até com os semáforos bem programados e atualizados, pois eles são incapazes de reagir instantaneamente ao trânsito [32].

Uma solução efetiva para o problema seria o uso de semáforos autônomos: um estudo feito pelo Dr. Carlos Gershenson [3] foi capaz de simular um tráfego similar ao da Cidade do México e otimizá-lo em 200% com esse tipo de semáforo.

Dessa forma, a questão de pesquisa que norteia o presente trabalho é: Como otimizar o fluxo de veículos em semáforos a partir de um sistema de controle autônomo e dinâmico? Assim, o estudo se pautará em entender as diversas alternativas existentes para a gestão do tráfego e tentará propor um novo método para atender melhor as necessidades do trânsito no Brasil.

## **1.1 Justificativa**

O estudo será realizado sobre a justificativa de tentar otimizar o tráfego de veículos em grandes centros urbanos devido à deficiente gestão de trânsito. Além disso, no plano pessoal, o autor do estudo vem analisando e estudando sobre engenharia de tráfego e otimizações na sua gestão através da análise e processamento de imagens há alguns anos.

## 2. Objetivos

O estudo tem como objetivo geral: analisar como é feita a gestão de tráfego e programação dos semáforos, além de propor um modelo para a otimização dos sinais de trânsito.

Para se atingir o objetivo geral, tem-se os seguintes objetivos secundários:

- Estudar como é realizada a gestão de tráfego em semáforos em todo Brasil.
- Propor um modelo de empresa viável para melhorar as condições do trânsito.
- Propor um modelo de sistema a ser utilizado para a otimização da programação semafórica.
- Estudar as melhores práticas e conceitos em Processamento de Imagens e Visão Computacional para a implementação de um controlador semafórico.
- Propor um método para a contagem e classificação de veículos em semáforos

### 3. Fundamentos Teóricos de Engenharia de Tráfego

O semáforo é um instrumento utilizado para controlar a passagem de veículos, pessoas, trens e outros meios de transporte. A aplicação é comumente utilizada para realizar a gestão do tráfego de veículos e pessoas nos grandes centros urbanos.

O primeiro semáforo foi instalado em 1868 no Reino Unido. Foi concebido por J. P. Knight, engenheiro especialista em assuntos ferroviários. Esse semáforo tinha dois braços móveis que se accionavam movendo cabos a partir de uma torre. O sistema era parecido com os sinais que regulam o trânsito ferroviário, uma vez que ele continha duas lâmpadas de gás com uma luz vermelha e outra verde. Entretanto, o instrumento não teve uma existência longa, já que em 1869, por causa de um acidente, foi retirado do mercado e até agosto de 1914 não se voltou a instalar um semáforo [11].

Com a expansão da frota dos veículos nas grandes cidades, se tornou necessário implantar as sinaleiras para criar um sistema de gestão de transportes eficaz e seguro. No entanto, o crescimento desenfreado dos automotores em circulação não acompanhou o planejamento urbano em grande parte do mundo, o que acarretou nos intermináveis congestionamentos verificados ao redor do mundo.

Além disso, as tecnologias e inovações para otimizar o fluxo urbano em semáforos também não acompanharam essas mudanças.

Assim um dos maiores problemas e dilemas do trânsito se concentra em como realizar um gestão eficiente dos semáforos.

#### 3.1. Programação do Semáforo

Para entender melhor a questão, é necessário entender o funcionamento clássico da maioria dos semáforos.

Eles possuem três fases: verde (abertura), amarelo (atenção) e vermelho (fechamento), que podem possuir diferentes programações ao longo do dia, ou seja, no período da manhã um determinado farol pode ter um tempo  $x$  de verde e  $y$  de vermelho, mas no fim do dia ele pode ter um tempo  $z$  de verde e  $w$  de vermelho [33].

E é nesse ponto em que se encontra o real problema, como são determinados os tempos de abertura e fechamento de cada fase de um cruzamento?

Atualmente, a forma usual de se determinar tais informações se baseia em um método extremamente manual em que um agente de trânsito realiza um estudo de contagem dos veículos e

determina os tempos de abertura e fechamento desse instrumento com base nas informações coletadas.

É importante ressaltar que algumas cidades, como São Paulo e Campinas possuem um sistema para determinar os tempos ótimos de abertura e fechamento a partir dos dados coletados [31]. No entanto, o sistema foi usado pouquíssimas vezes, pois funcionários da Companhia de Trânsito de Campinas entrevistados, no Capítulo 4, informaram que o sistema era complexo e que não tinham o treinamento necessário para executá-lo.

Assim, por se tratar de uma metodologia extremamente manual e custosa, o processo é realizado poucas vezes ao ano, de acordo com o Secretário de Trânsito de Paulínia no Capítulo 4, o que torna a programação semafórica para cada semáforo desatualizada e inadequada, pois ela não reflete as reais necessidades do trânsito naquele momento.

No item 3.2, abaixo, seguem algumas alternativas já existentes no mercado, mas pouco empregadas por diversos motivos.

### **3.2 Alternativas Existentes ao modelo atual**

Atualmente existem sistemas de semáforos com atuação baseada em diferentes tipos de sensores. Os mais relevantes são:

#### **3.2.1 Laço Indutivo de Cobre**

O sensor mais amplamente utilizado no Brasil é o laço indutivo de cobre [21]. Trata-se de uma bobina de cobre embutida no asfalto, que identifica quando um veículo passa por cima dela. Para a sua instalação, corta-se uma valeta no asfalto e coloca-se uma bobina dentro da valeta. Feito isso, ela é fechada para proteção do sistema. Quando um veículo passa por cima da bobina, ocorre uma perturbação no campo magnético gerado pelo dispositivo e assim ocorre a identificação.

Entretanto, por meio de pesquisas realizadas com fabricantes e de uma consulta ao Secretário de Trânsito e Transporte de Piracicaba, o sistema utilizado não é eficiente e apresenta os seguintes problemas:

- Dificuldade de instalação, devido à necessidade de fechar a via para cortar uma valeta no asfalto, embutir o sistema e depois reconstituir o asfalto;
- Difícil manutenção, pois existem casos em que o asfalto cede devido ao peso dos veículos e danifica as bobinas, sendo necessário realizar um processo similar ao de instalação inicial.

- Para detecção de diferentes tipos de veículos é preciso que o laço possua configurações diferentes (tamanho, formato e número de voltas).
- Dificuldade de instalação em pistas de concreto, pois as ferragens desse tipo de pavimento criam interferências no sensor magnético.
- É necessário um laço por pista, o que eleva o custo.
- A necessidade de contratar analistas de trânsito e programadores de semáforo para criar o algoritmo de abertura e fechamento do sinal, para cada um dos pontos com o sistema.

### 3.2.2 Sensor Ear-It

O sensor “Ear-it” capta informações do trânsito através dos sons emitidos pelos carros. Trata-se de um sistema inovador que está sendo empregado na cidade de Santander na Espanha [5] [6]. Para isso, foram espalhados pela cidade centenas de sensores de som capazes de medir os volumes de tráfego, velocidades dos veículos e eventos sônicos, como disparos provenientes de armas ou sirenes de ambulâncias.

As desvantagens que provam a ineficiência do sistema são [22]:

- Quando os veículos elétricos se tornarem mais populares, perde-se grande parte da funcionalidade do sistema (fator crítico);
- Sistema focado na inteligência das Grandes Centrais de Monitoramento do Trânsito, que são muito custosas.

### 3.2.3 Microondas

O instrumento consiste em um radar que é operado pelos sinais recebidos por microondas. Eles foram projetados para coletar insumos de tráfego nos Estados Unidos e o funcionamento se baseia no fato de que a passagem de um veículo por esse radar, causa uma perturbação no campo eletromagnético a qual é refletida para a antena receptora que identifica o veículo [23].

Embora o sistema apresente baixa sensibilidade a variações climáticas e de luminosidade, esse instrumento apresenta as seguintes desvantagens:

- Apresenta imprecisões para detectar veículos estacionados ou parados.
- O ajuste da altura da antena receptora para proporcionar a largura correta do feixe de energia é extremamente complexo.
- Os custos de implantação são elevados.

### 3.2.4 Câmeras de Monitoramento

Essa tecnologia é muito próxima da solução proposta no presente estudo, e, portanto, neste tópico será exposto um benchmark que evidencia o poder desta tecnologia.

Uma empresa que adotou esse modelo é a startup Surtrac. Ela realizou um projeto piloto em 50 cruzamentos na cidade de Pittsburgh nos EUA com o intuito de otimizar o tráfego da região [7].

Para isso ela instala sensores com câmeras que capturam os dados do trânsito e para cada instante de tempo é criado um novo plano de tráfego para o cruzamento [25].

É importante ressaltar que os sensores instalados têm autonomia para agir de imediato nos semáforos, sem a necessidade da atuação de uma central para tomar a ação.

Esse sistema de tomada de decisão local torna o produto realmente escalável, pois as centrais de informação são normalmente o gargalo das soluções de trânsito por possuírem capacidade finita.

Os resultados obtidos nos trechos urbanos em teste foram uma redução de 25% no tempo de viagem e uma redução de 40% do tempo ocioso dos carros [25].

## **4. Pesquisa de Campo**

Além dos estudos teóricos sobre o funcionamento do trânsito e as alternativas existentes para o controle do tráfego, o trabalho se propôs a conhecer melhor o funcionamento dos semáforos e a sua programação.

Assim, conversou-se com pessoas ligadas a programação semafórica das cidades de Paulínia, Americana e Piracicaba.

### **4.1 Visita à Operação de Trânsito de Paulínia**

No dia 22 de Novembro de 2017 foi realizada uma visita ao município de Paulínia junto ao representante de trânsito da Secretaria de Paulínia, o senhor Alcides Silva. Ele é o responsável por realizar toda a manutenção e programação dos semáforos, além de que também faz o estudo e define os planos de tráfego da cidade.

#### **4.1.1 Overview do Município**

O município de Paulínia fica localizado no estado de São Paulo e possui uma população de 102 mil habitantes, de acordo com dados do IBGE 2010. A cidade se destaca pela intenso crescimento populacional e por apresentar a sétima maior renda per capita do Brasil, em função do grande polo petroquímico ali instalado [27].

Atualmente, Paulínia possui graves problemas de trânsito e 23 cruzamentos semaforizados, compondo um total de 17 controladores, de acordo com o representante da Secretaria de Trânsito entrevistado. É importante ressaltar que um controlador pode controlar mais do que um cruzamento.

#### **4.1.2 Mobilidade Urbana**

Paulínia não possui uma central de trânsito e nem uma operadora responsável por tal operação. Assim, toda a tarefa de gestão do tráfego e semáforos é realizado por uma única pessoa, algo que parece ser uma tendência em municípios menores de acordo com outras visitas técnicas realizadas pelo estudo.



### 4.1.3 Controladores Semafóricos

Todos os controladores do município são fornecidos pela empresa NEWTESC (antiga TESC). No entanto, existem controladores de diferentes versões da empresa.

Os mais antigos precisam que aparelhos de GPS sejam acoplados ao mesmo para garantir a sincronização dos diferentes controladores da cidade a fim de garantir o funcionamento da chamada onda verde.

Já os controladores mais novos são equipados com câmeras que realizam a contagem dos veículos e adaptam os tempos semafóricos de acordo com as diferentes programações inseridas no mesmo.



Figura 1: Controlador Semafórico em um semáforo de Paulínia.

É importante ressaltar que todos os controladores são da mesma empresa em função das dificuldades em ter controladores de diferentes marcas. Isso se deve ao fato de que os controladores possuem um protocolo fechado que impede a comunicação entre aparelhos de outras empresas, além de que existe a necessidade de realizar a capacitação dos funcionários da Secretaria de Transportes para operar os aparelhos.

### 4.1.4 Programação dos Semáforos

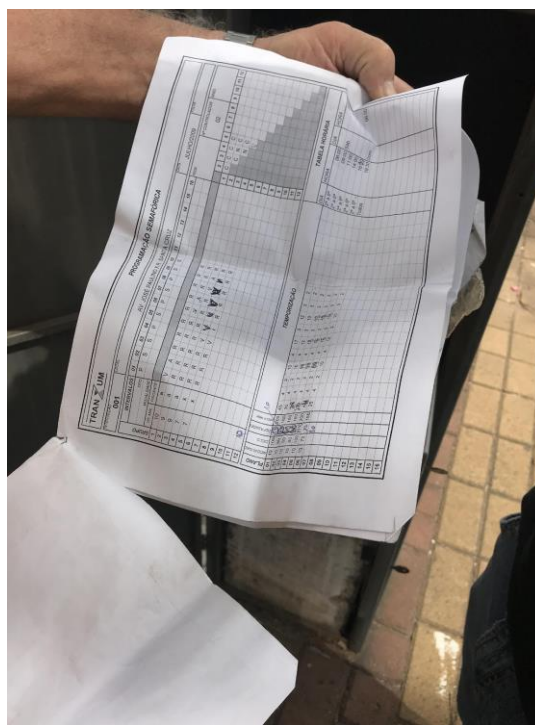
O controle dos semáforos é feito de maneira extremamente arcaica, uma vez que uma pessoa é responsável por todo planejamento e gestão dos mesmos. Assim tal tarefa de maneira geral é realizada manualmente e em cada caixa semafórica do cruzamento existe: o controlador e um papel em que são anotados os tempos atuais de abertura e fechamento de cada semáforo do cruzamento, que possui um pequeno projeto do cruzamento.

Além disso, como a programação é feita manualmente, os tempos semaforicos são determinados quando alguém acha que essa programação não é efetiva, ou seja, a alteração é determinada sem um estudo prévio.

Também é importante ressaltar que em cruzamentos mais complexos, o município contrata alguns profissionais com experiência em programação semaforica para realizar a programação inicial dos semáforos do cruzamento.



(a)



(b)

Figura 2: (a) Dispositivo para realizar a programação semaforico/(b) Plano do cruzamento

#### 4.1.5 Planos para o Futuro

A cidade está planejando abrir uma licitação de aproximadamente R\$2 milhões para modernizar os semáforos da cidade a fim de tentar obter a chamada onda verde. Para tal licitação constam os serviços de instalação de novos controladores e a manutenção dos mesmo ao longo do período previsto na licitação.

#### 4.2 Visita à Operação de Trânsito de Americana

No dia 13 de Dezembro de 2017 foi realizada uma visita ao município de Americana junto ao representante de trânsito da Secretaria de Americana. Ele é o responsável por realizar toda a

manutenção e programação dos semáforos, além de que também faz o estudo e define os planos de tráfego da cidade, assim como o responsável por Paulínia.

#### 4.2.1 Overview do Município

O município de Americana fica localizado no estado de São Paulo e possui uma população de 230 mil habitantes, segundo o IBGE 2010. O município possui 70 cruzamentos semaforizados, compondo um total de 63 controladores, de acordo com o representante da prefeitura entrevistado. Além disso existe um total de 6 radares espalhados pela cidade. Seu PIB per capita é de R\$ 44 mil, tendo a posição 79 no estado e posição 348 no Brasil [28].

#### 4.2.2 Setor de Mobilidade Urbana

O setor de transportes de Americana vive principalmente de contra-partidas da iniciativa privada, ou seja, quando pólos geradores (ex supermercado, shopping) se instalam no município, são exigidas contra-partidas para melhorias no trânsito nas proximidades do estabelecimento instalado.

O município não possui central de trânsito, e assim como foi visto em Paulínia, toda a gestão dos semáforos é feita por uma única pessoa.

#### 4.2.3 Controladores Semaforicos

Pelas dificuldades de se manter controladores de diferentes marcas na rede do município tal qual Paulínia, todos os seus controladores são da NEWTESC.



(a)



(b)

Figura 3: (a) Controlador Semaforico / (b) Dispositivo para a programação de semáforo idêntico ao da Figura 2(b)

#### **4.2.4 Programação Semafórica**

Para se realizar a programação adequada dos semáforos de um cruzamento, é preciso conhecer o volume de veículos que passam ali em cada hora do dia.

No caso de Americana, essa programação é feita por inspeção visual, ou seja, por tentativa e erro são testadas algumas programações até que se chegue em um estado aceitável baseado no tamanho das filas que se formam ali.

Além disso, o responsável pela gestão dos semáforos utiliza muito o Google Maps para ver quais pontos da cidade possuem trechos congestionados (trecho vermelho no aplicativo), o que pode ser um indicativo de programação ultrapassada.

Apesar de conseguirem fazer a contagem volumétrica nos 6 radares instalados na cidade, por se tratarem de poucos pontos de contagem, o município aproveita pouco dos dados coletados nos mesmos, uma vez que essa contagem é não classificada (não se sabe o tipo do veículo passando) e não se realiza o monitoramento do veículo (pesquisa de origem e destino). Quando precisam de dados de pesquisa do tipo origem-destino para fazer alterações no trânsito, os motoristas parados nos semáforos são entrevistados por agentes de trânsito.

#### **4.2.5 Planos para o Futuro**

A cidade está planejando abrir uma licitação para modernizar a forma com que obtém dados do tráfego a partir de câmeras com tecnologia de OCR (*Optical Character Recognition*), permitindo estudos de Origem Destino. Além disso, a cidade pretende abrir em 2018 uma licitação de aproximadamente R\$ 1 milhão para trocar todos os controladores do município.

### **4.3 Entrevista com o Secretário de Trânsito de Piracicaba**

No dia 08/04/2017 foi realizada uma conversa de aproximadamente 40 minutos, por telefone, com o Secretário de Trânsito e Transporte de Piracicaba. Nela houve demonstração de interesse em conhecer mais a fundo o projeto, pois o sistema iria reduzir expressivamente o custo da central e da infraestrutura necessária, uma vez que os custos das centrais de controle e de manutenção são altíssimos.

Por exemplo, segundo dados da CET 2013 (Companhia de Engenharia de Tráfego) da cidade de São Paulo, estão registrados 4263 funcionários. Desses, 2704 pessoas estão trabalhando no

monitoramento, gestão e em operações, como trabalhos de manutenção ou ajuda em acidentes, de trânsito. Segundo o secretário, o uso de semáforos inteligentes poderia diminuir em quase 40% o tamanho dessas centrais.

Ainda segundo ele, seria interessante que o custo do produto fosse feito através do pagamento de mensalidades por instalação do sistema em semáforos, pois isso diluiria o custo total em forma de mensalidades e facilitaria o processo de licitação com a prefeitura.

Além disso, foi sugerida a integração do funcionamento dos semáforos com outras funcionalidade benéficas às prefeituras, como leitura de placas dos veículos para facilitar a busca por carros roubados.

#### **4.4 Entrevista com o Secretário de Trânsito de Campinas**

No dia 13/12/2018 foi realizada uma conversa por telefone com um dos secretários da EMDEC ( Empresa Municipal de Desenvolvimento de Campinas) que é a responsável por toda a gestão de tráfego na região metropolitana de Campinas.

O entrevistado contou ao presente estudo que Campinas possui 585 cruzamentos semaforizados e que os semáforos podem funcionar de três modos diferentes:

- Isolado, ou seja, não depende de outros semáforos.
- Em rede, isto é, os semáforos dessa rede operam pela comunicação entre os aparelhos
- Mestre-Escravo em que um ou mais semáforos dependem de outros semáforos.

Além disso, foi comentado que o município não coleta dados do tráfego e a única informação que a EMDEC possui do trânsito é proveniente dos radares. Assim para coletar qualquer informação de trânsito ou alteração da programação de semáforos é necessário realizar um estudo de campo para contabilizar os carros que passam por determinada região.

Todos os cruzamentos de Campinas são monitorados por uma centra de trânsito conhecida como “CimCamp”, em que os operadores conseguem observar por câmeras as filas nos semáforos e podem alterar para aquele momento o tempo de abertura e fechamento. Também foi dito que a EMDEC possui um sistema para inserir a quantidade de carros passando por cada via do cruzamento e o programa retorna os tempos ótimos de abertura e fechamento de cada lado da via. Entretanto, segundo o entrevistado, o sistema é complexo para os funcionários e a falta de verbas para treinamentos impede a utilização do sistema.

Assim, toda a temporização dos semáforos continua sendo definida por testes de acertos e erros.

## 5. Estudo de Campo em Barão Geraldo

Para entender ainda melhor a situação dos semáforos foi feito um estudo de campo em que será exposto uma validação do problema em semáforos de Campinas.

Buscando validar tal problema e simular o processo manual realizado pelas Companhias de Tráfego para determinar os tempos da programação semaforica, um teste foi realizado em dois cruzamentos de Campinas, no período do almoço, no distrito de Barão Geraldo.

### 5.1 Estudo no Cruzamento Av. Oscar Pedroso Horta e Av Dr. Luís Tela

O primeiro teste, o qual será nomeado de Teste A, foi realizado no cruzamento entre a Av. Doutor Luiz da Tela com a Av. Oscar Pedroso Horta, onde existem três semáforos, dois deles na Av. Oscar Pedroso Horta.



Figura 4: Cruzamento entre a Av. Doutor Luiz da Tela e Av. Oscar Pedroso

Os semáforos da Figura 4 foram enumerados, de 1 a 3, sendo o semáforo 1 correspondente ao da Av. Doutor Luiz da Tela. Analisou-se o tempo em que cada semáforo permanecia aberto. Os tempos cronometrados foram, respectivamente: 24 segundos, 36 segundos e 13 segundos.

O método utilizado para a validação do problema foi baseada na contagem de veículos na via para cada semáforo do cruzamento. Sendo assim, para cada via com o sinal "verde" ou "amarelo" em

que não houvesse tráfego de veículos, o número de carros parados no outro semáforo do cruzamento, isto é, fechado (vermelho) foi contado e armazenado. Caso contrário, a contagem não foi realizada.

Nas Tabelas 1, 2 e 3, foi apresentado a quantidade de carros parados nos semáforos fechados. Além disso, foi definido o 'tempo de possível otimização' como o tempo em que o semáforo manteve-se aberto sem a passagem de nenhum veículo.

É importante ressaltar que a diferença nas Tabelas 1, 2 e 3 se encontra em qual foi a via de referência utilizada, ou seja, qual via se encontrava fechada para cada iteração.

Tabela 1: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 1 está aberto.

(Teste A)

Via 1	Via 2	Via 3	Tempo de Possível Otimização
-	2	0	0:00:04
-	0	0	0:00:00
-	0	0	0:00:00
-	0	8	0:00:01
-	0	0	0:00:00
-	0	0	0:00:00
-	0	1	0:00:06
-	0	0	0:00:00
-	0	0	0:00:00
-	0	0	0:00:00

Tabela 2: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 2 está aberto

(Teste A)

Via 1	Via 2	Via 3	Tempo de Possível Otimização

4	-	4	0:00:03
3	-	6	0:00:03
7	-	5	0:00:10
0	-	0	0:00:00
0	-	6	0:00:03
6	-	0	0:00:12
5	-	8	0:00:05
1	-	1	0:00:08
8	-	0	0:00:09
5	-	8	0:00:04

Tabela 3: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 3 está aberto  
(Teste A)

Via 1	Via 2	Via 3	Tempo de Possível Otimização
10	1	-	0:00:07
0	0	-	0:00:00
0	0	-	0:00:00
0	0	-	0:00:00
0	0	-	0:00:00
0	0	-	0:00:00
0	0	-	0:00:00
0	0	-	0:00:00
0	0	-	0:00:00
3	0	-	0:00:01



## 5.2 Estudo no Cruzamento da Av.Santa Isabel e R. Antônio Pierozzi

As mesmas definições foram aplicadas no segundo semáforo analisado. O segundo teste, o qual foi nomeado como Teste B, ocorreu no cruzamento da Av. Santa Izabel com a R. Antônio Piezori.



Figura 5: Cruzamento entre Av. Santa Izabel e a R. Antônio Piezori.

Nesse cruzamento existem dois semáforos, Figura 5, sendo que o semáforo intitulado como semáforo 1 fica na Av. Santa Izabel e o semáforo 2 controla o fluxo da R. Antônio Piezori. O tempo dos semáforos abertos é de 52 segundos e 27 segundos, respectivamente.

Tabela 4: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 1 está aberto (Teste B)

Via 1	Via 2	Tempo da possível otimização
-	0	00:00:00
-	0	00:00:00

-	0	00:00:00
-	0	00:00:00
-	0	00:00:00
-	0	00:00:00
-	0	00:00:00
-	0	00:00:00
-	0	00:00:00
-	1	00:00:01

Tabela 5: Quantidade de carros parados em cada um dos semáforos para cada iteração onde o semáforo 2 está aberto  
(Teste B)

Via 1	Via 2	Tempo da possível otimização
10	-	00:00:16
13	-	00:00:21
12	-	00:00:20
0	-	00:00:00
0	-	00:00:00
8	-	00:00:01
7	-	00:00:03
8	-	00:00:17
7	-	00:00:09

### 5.3 Análise Comparativa

Finalmente, após a realização dos dois teste: Teste A e Teste B, é possível realizar uma análise comparativa sobre os resultados de ambos experimentos e tirar conclusões para o sistema a ser proposto.

Para isso, foi construída a Tabela 6 que indica o tempo médio de otimização para cada semáforo do cruzamento.

Tabela 6: Tempo médio de otimização para ambos os testes

Semáforo	Tempo Médio
Teste A	
Semáforo 1	00:00:01
Semáforo 2	00:00:06
Semáforo 3	00:00:01
Teste B	
Semáforo 4	00:00:00
Semáforo 5	00:00:10

O teste A indicou que 51,7% das iterações desses semáforos foram ineficientes: houveram situações em que haviam mais de 13 carros parados enquanto não existia fluxo na via que continha o sinal aberto, além de iterações do semáforo onde carros ficaram esperando por 12 segundos sem necessidade.

Já o teste B , mostrou que 45% das iterações foram ineficientes, mas nesse caso houve um número bem maior de carros que ficaram parados: até 18 carros enquanto a outra via não continha fluxo. O tempo que os carros permaneceram parados sem necessidades chegou a 21 segundos durante o período de coleta.

Assim, conclui-se que é possível otimizar a gestão da programação semafórica nos cruzamentos analisados.

## 6. Solução Proposta

No capítulo anterior foi analisado e explicado como é feita a gestão do tráfego através de um estudo de campo, análise das alternativas existentes e pesquisas com Secretarias de Trânsito.

Os dados e informações coletados forneceram *insights* interessantes e que levam à conclusão de que a programação semaforica é extremamente custosa e ineficiente, vide as longas filas nas vias urbanas.

Assim, a convergência e diversidade dos dados, leva o estudo a propor uma nova forma de gerir o trânsito. A seguir será explicado a solução proposta e como o estudo se diferencia das demais tecnologias nesse mercado.

### 6.1 Solução

A solução proposta consiste em otimizar o fluxo de veículos em semáforos a partir de um sistema de controle que utiliza câmeras acopladas em semáforos para identificar os veículos em cada sentido do cruzamento.

A partir disso, um *hardware* conectado as câmeras conta a quantidade de veículos em cada via e determina dinamicamente os tempos em que cada semáforo do cruzamento fica aberto ou fechado.

### 6.2 Segmento de Clientes

A partir dos dados coletados, dos *insights* gerados no estudo feito em semáforos de Campinas e das entrevistas realizadas com algumas Prefeituras, foram determinados os segmentos de clientes para os quais é desejado oferecer a solução proposta para o fluxo ineficiente de veículos.

O early adopter inicial escolhido foram as Secretarias de Trânsito de Transporte Municipais. Tais órgãos possuem interesse no estudo, uma vez que o sistema de controle dinâmico e autônomo pode permitir a redução dos custos das centrais de controle e manutenção e, para efeitos de mobilidade urbana, é interessante a implementação de soluções inovadoras.

Além disso, a otimização do fluxo de veículos pode representar uma melhoria da imagem desse órgão perante a população em geral, haja vista que a população perde diariamente muito tempo no trânsito para realizar deslocamentos pequenos nestas cidades.

Também foi identificado que as Companhias de Trânsito podem ser potenciais clientes. Em grandes cidades, como São Paulo e Campinas, a prefeitura terceiriza o serviço de operação do trânsito numa tentativa de 'dividir' e acelerar o trabalho da Secretaria mencionada acima. Essas empresas,

ligadas às próprias prefeituras, possuem autorização para implementar soluções de tráfego de veículos, pedestres, sinalização de vias, entre outros.

E, por fim, há uma possibilidade de se trabalhar com centros de pesquisa e inovação. Tais centros, como o CPqD (Centro de Pesquisa e Desenvolvimento em Telecomunicações), buscam soluções inovadoras para diversos problemas incluindo mobilidade urbana. Como explicitado acima, o CPqD possui um espaço de pesquisa, o Hiper-Espaço, focado em pesquisa de mobilidade urbana e cidades inteligentes.

No dia 26/05/2017, visitou-se o “Hiper-Espaço” apresentar o projeto para os especialistas em Cidades Inteligentes e Mobilidade Urbana do CPqD. Nessa reunião, obteve-se uma boa visão de negócios e de produto. Eles se mostraram muito interessados em ver um protótipo funcionando, inclusive oferecendo a oportunidade de testar a tecnologia em uma cidade que está recebendo um projeto de Cidades Inteligentes do CPqD.

### **6.3 Canais**

A proposição de valor é a otimização do fluxo de veículos a partir de um sistema de controle de semáforos de forma autônoma e dinâmica, e será entregue aos clientes (companhias de trânsito e prefeituras) através de um canal direto de venda, isto é, a venda será realizada diretamente aos órgãos via contratos de licitação.

De acordo com o Secretário de Transporte de Piracicaba, a melhor maneira de entregar esse valor é por meio de reuniões para apresentações presenciais e demonstrações *in loco* de forma a demonstrar o funcionamento do sistema proposto.

Além disso, como o usuário final da tecnologia será a população das cidades, o negócio será divulgado na mídia (televisão, jornais, e portais da internet) via relações públicas e terá demonstrações em feiras e eventos de mobilidade e tecnologia para cidades, a fim de aumentar a visibilidade do produto e potencializar a concretização do negócio com o cliente pagador (prefeituras e companhias de tráfego municipais).

## 7. Sistema Proposto

A solução consiste em um sistema de controle de semáforos de forma autônoma e dinâmica, acoplado ao próprio semáforo e contido em um invólucro protegido contra efeitos ambientais. Nesse invólucro haverá um *hardware* para processamento de imagens, juntamente com uma câmera apontando para cada direção do cruzamento.

A partir disso, as câmeras ficam responsáveis por realizar a contagem dos veículos em cada via e, finalmente, um *hardware* (controlador semafórico) define dinamicamente o tempo de passagem para cada via.

Como mostra a Figura 6, o sinal para passagem na via A está fechado, mas a quantidade de carros é maior que o estimado pela programação, fazendo com que o tempo previamente estabelecido não seja eficiente. A solução realiza um balanceamento ponderado desse tempo, permitindo que a espera nas diferentes vias seja coerente com a quantidade de veículos, ou seja, as vias com o maior fluxo de veículos ficarão menos tempo fechadas do que as demais vias.



Figura 6: Exemplo de cruzamento com 3 fluxos, mas apenas 2 tempos, devido aos fluxos B e C não se cruzarem.

Além disso o balanceamento é chamado de ponderado, dado que diferentes modelos de veículos possuem uma ponderação diferente, no caso veículos de passeio possuem um peso  $P$ , veículos de transporte público possuem um peso  $10P$  e veículos oficiais/emergenciais, tais quais

ambulâncias e carros de polícia, possuem um peso  $100P$ , em que  $P$  representa uma constante positiva e maior que zero. Dessa forma a solução proposta realiza um balanceamento mais justo e dá preferência para os automóveis mais necessitados.

Portanto, a solução permite desafogar o trânsito através de um gerenciamento autônomo de cada cruzamento.

O Professor Carlos Gershenson, da Universidad Nacional Autónoma de México, realizou um estudo chamado *Self-organizing Systems* [3]. Nele, foram realizados experimentos mostrando que, no tráfego urbano, decisões locais são mais eficientes que as globais.

Seguindo essa linha, a solução apresentada permite melhorar o tráfego em determinados pontos e, como efeito em cadeia, em toda a extensão de uma avenida ou cidade, por exemplo.

## 7.1 Planejamento de Desenvolvimento

A seguir, é apresentado o planejamento para o desenvolvimento do produto:

- Estudo sobre engenharia de tráfego e funcionamento/programação de semáforos em diversas cidades brasileiras e até internacionais.
- Realização de estudo de campo feito em semáforos de Campinas para validar e testar algumas hipóteses sobre o sistema.
- Formular proposta de solução e realizar a modelagem do sistema.
- Simulações de tráfego para a solução proposta.
- Desenvolvimento do algoritmo para a identificação de veículos.
- Desenvolvimento do algoritmo para realizar a contagem, por imagem, de veículos para o controle do tempo de espera em semáforos e realizar a tomada de decisões. Assim, o sistema será capaz de determinar sua própria temporização, de forma autônoma e dinâmica.
- Criação de um invólucro protegido contra efeitos ambientais contendo câmeras e um placa computacional para o processamento das imagens de cada semáforo.
- Criação do *hardware* (controlador semafórico) para a tomada de decisões e realizar as alterações de programação nos semáforos.
- Sistema de fail safe: desenvolvimento de um sistema externo responsável por gerenciar os LEDs indicativos de todos os semáforos, garantindo que não haja abertura/fechamento simultâneos que possam ocasionar acidentes ou colisões. Em caso de falha do *software* ou de algum defeito

no *hardware*, os semáforos do cruzamento passam a atuar em sua programação padrão e a equipe de manutenção é avisada para identificar e solucionar o problema.

- Adição de dispositivo 3G/internet móvel, para comunicação com nuvem de dados e gerenciamento global do sistema. Cada semáforo envia informações para a central de controle a fim de que os operadores pudessem monitorar e também definir as ações quando necessário, além de utilizar as informações obtidas como insumos para a implementação de melhorias.
- Realizar a leitura de placas, a partir das câmeras. Tal funcionalidade permite também que o sistema funcione também como um radar, agregando valor para as prefeituras.

## 7.2 Simulações de Tráfego

Para verificar que a solução proposta é viável e aplicável nos semáforos de grandes cidades, foram feitas algumas simulações de tráfego utilizando o software Sumo Traffic[11], um dos principais simuladores de mobilidade urbana disponíveis.

Desenvolvido pelo DLR, Instituto de Sistemas de Transporte do Centro Aeroespacial da Alemanha, a plataforma *open-source* foi desenvolvida em linguagem de programação C++ para simular os principais aspectos do trânsito em cruzamentos de simulação e até mesmo espaços reais e funcionais, uma vez que o sistema permite importar mapas a partir de uma biblioteca disponível.

Além disso, é possível inserir planos de tráfego reais no sistema e a documentação do software é extremamente detalhada [12].



Figura 7: Aplicação no Sumo Traffic



Também é importante ressaltar que o simulador está sendo utilizado em diversas pesquisas e aplicações ao redor do mundo, como por exemplo:

- Avaliar a performance de semáforos, incluindo a avaliação de novos algoritmos de gestão de tráfego.
- Algoritmos de decisão de rotas de veículos.

No estudo realizado foram realizadas dois modos de simulação para diversos momentos do dia:

- Simulação utilizando o sistema atual de gerenciamento de tráfego (tempos pré-definidos)
- Sistema proposto pelo estudo (tempos dinâmicos).

Para efeitos de teste e para fins comerciais foram realizadas simulações com as cidades de: Campinas (Brasil), Boston (Estados Unidos) e Beijing (China), que podem ser verificadas, respectivamente, nas Figuras 8, 9 e 10.

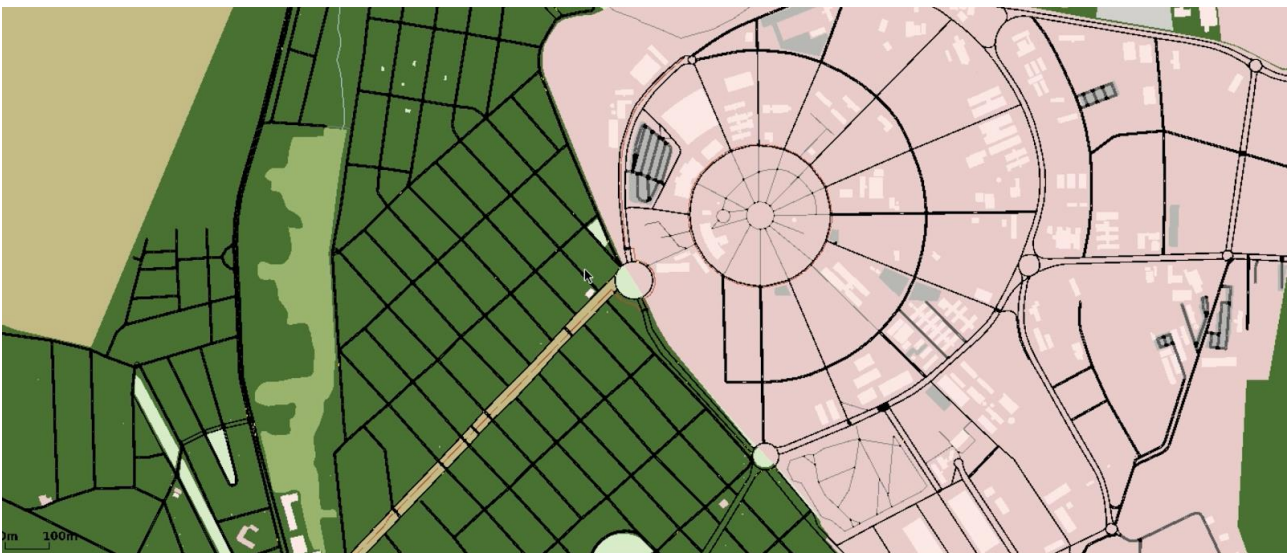
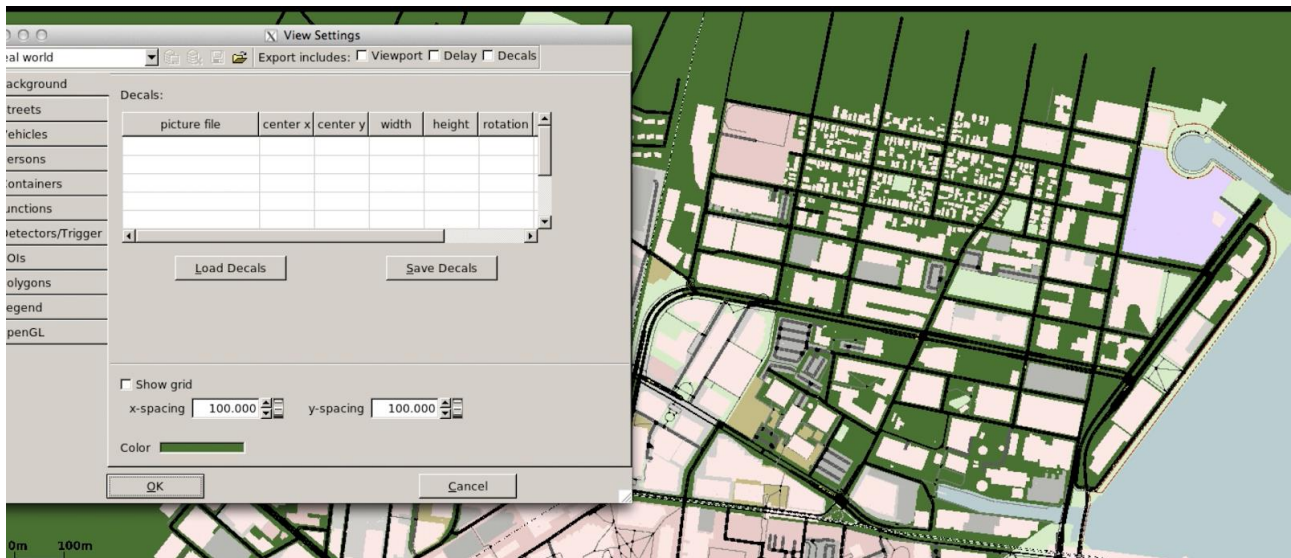


Figura 8: Simulação no Distrito de Barão Geraldo (Campinas - SP)



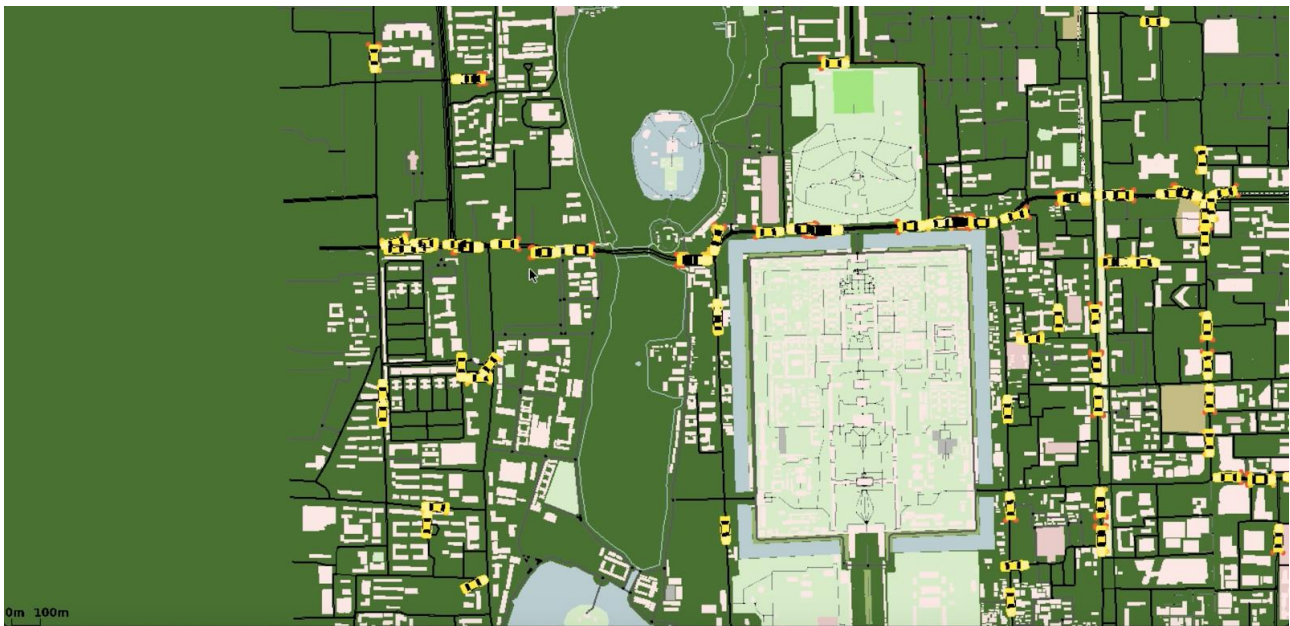


Figura 11: Simulação na cidade de Beijing (China)

Feita a modelagem de cada plano de tráfego e da definição da primeira versão do algoritmo de simulação proposto, chegou-se a valores próximos de otimização dos que aqueles obtidos no estudo de campo realizado na cidade de Campinas.

Os primeiros testes apontaram uma redução de, aproximadamente, 5% no tempo de espera médio de veículos em semáforos. Já os posteriores, com regulagens e algumas modificações propostas, como a forma como é feita a contagem de veículos e o respeito de um ciclo semafórico, fez com que a redução passasse de 10-13,5%.

Acredita-se que em versões futuras é possível obter uma otimização média de quase 25%.

### 7.3 Modelagem do Sistema

Nos capítulos anteriores, o estudo se preocupou em estudar como é feita a gestão do tráfego e dos semáforos com estudo de campo nas ruas de Campinas e visitas técnicas a algumas cidades, como Paulínia e Americana.

Além disso, também foi discutido uma possível solução para a questão da ineficiência de sinais luminosos e a consequente para o fluxo de veículos.

Até o momento o estudo passou pelas fases: de ideação do produto com base nos insights gerados pela pesquisa e, posteriormente, chegou-se a fase de simulação e testes em um simulador de tráfego.

O próximo passo consiste em começar a modelar o sistema necessário para a implementação dos semáforos inteligentes, além de realizar a extração de requisitos para cada módulo a ser



implementado. Na Figura 12 é possível verificar um modelo em alto nível do sistema a ser desenvolvido.

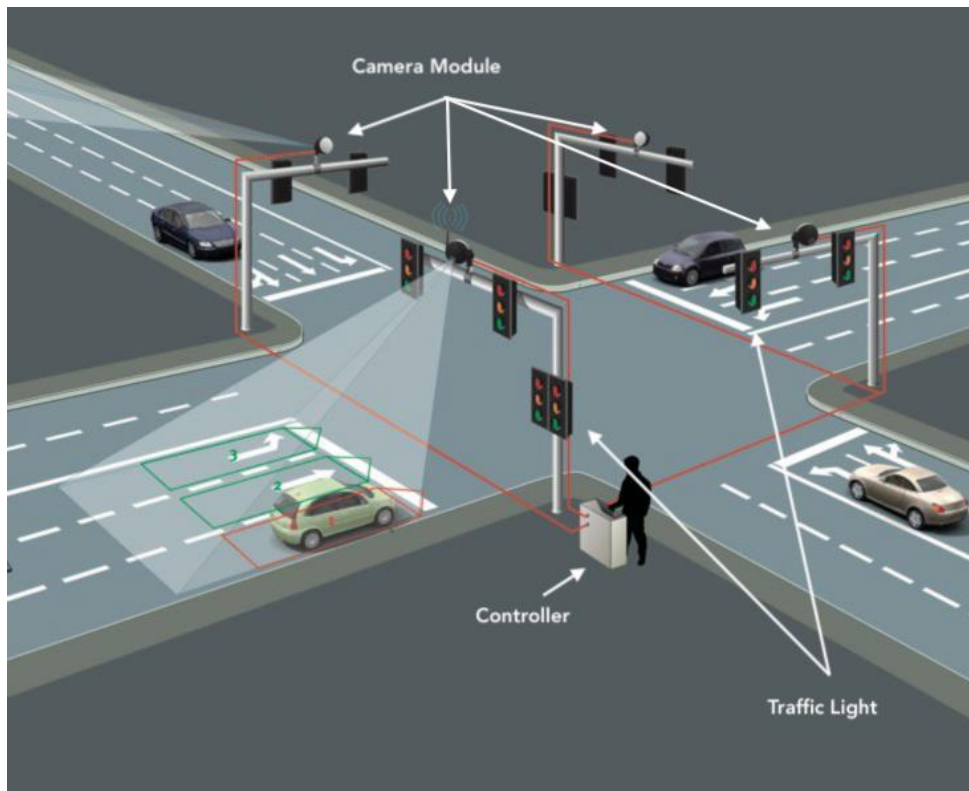


Figura 12: Modelo em alto nível do funcionamento do sistema

O protótipo consiste na criação de um *hardware/software* para a criação de um sistema de controle de semáforos que reage instantaneamente as necessidades do trânsito. O modelo está segmentado em 3 módulos: semáforos inteligentes, central(controlador) e cliente.

Na Figura 13 é possível ver um *overview* da sistema em um nível um pouco mais técnico. É importante ressaltar que cada processo e segmentação será discutida e estudada na sequência.

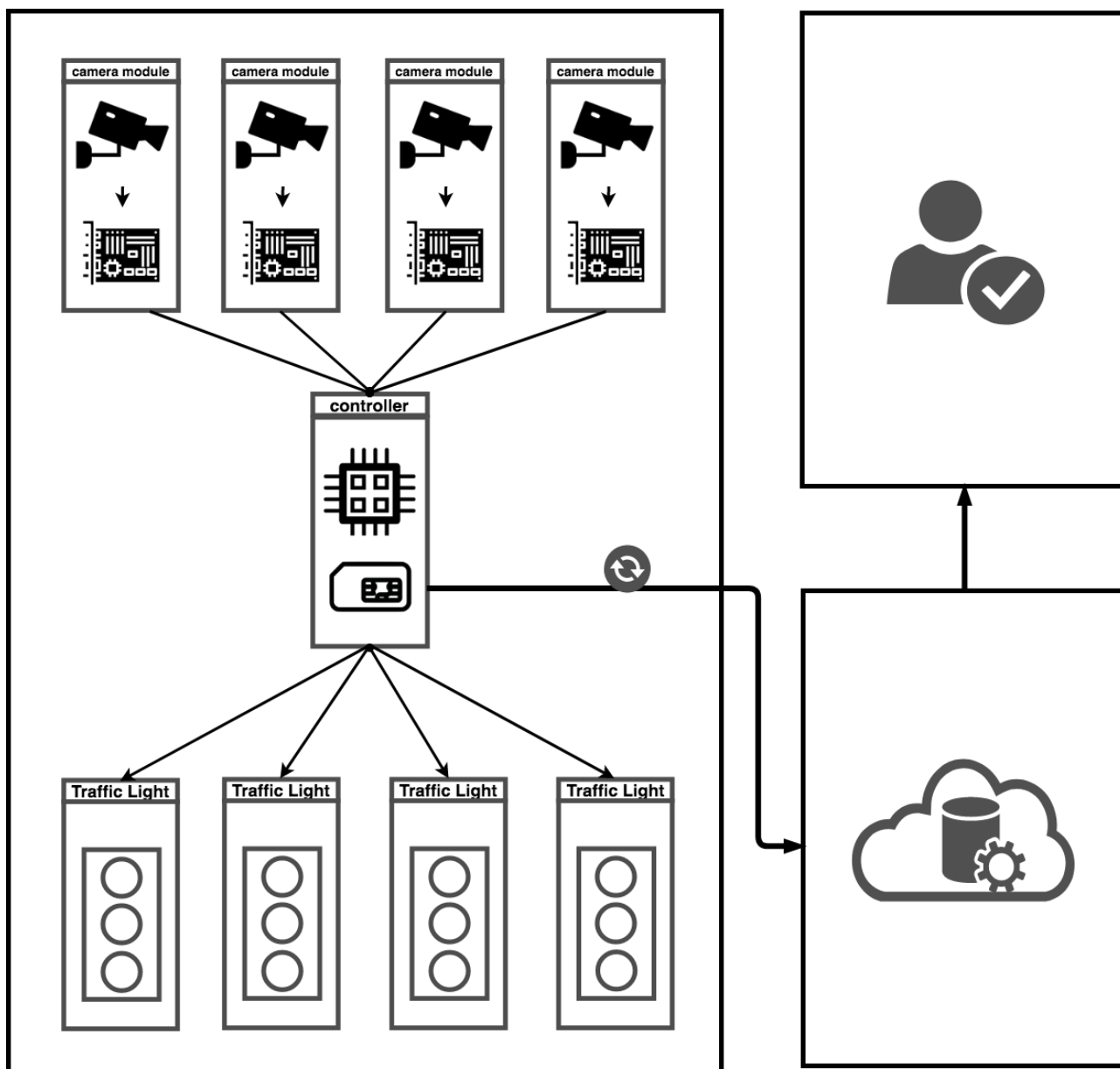


Figura 13: Overview da Solução

### 7.3.1 Semáforos Inteligentes

Semáforos inteligentes consistem em sinais de trânsito que são capazes de identificar certos padrões e alterar seu funcionamento com base na situação encontrada. Assim, na Figura 14, encontra-se um esquema para um sistema de semáforos inteligentes.

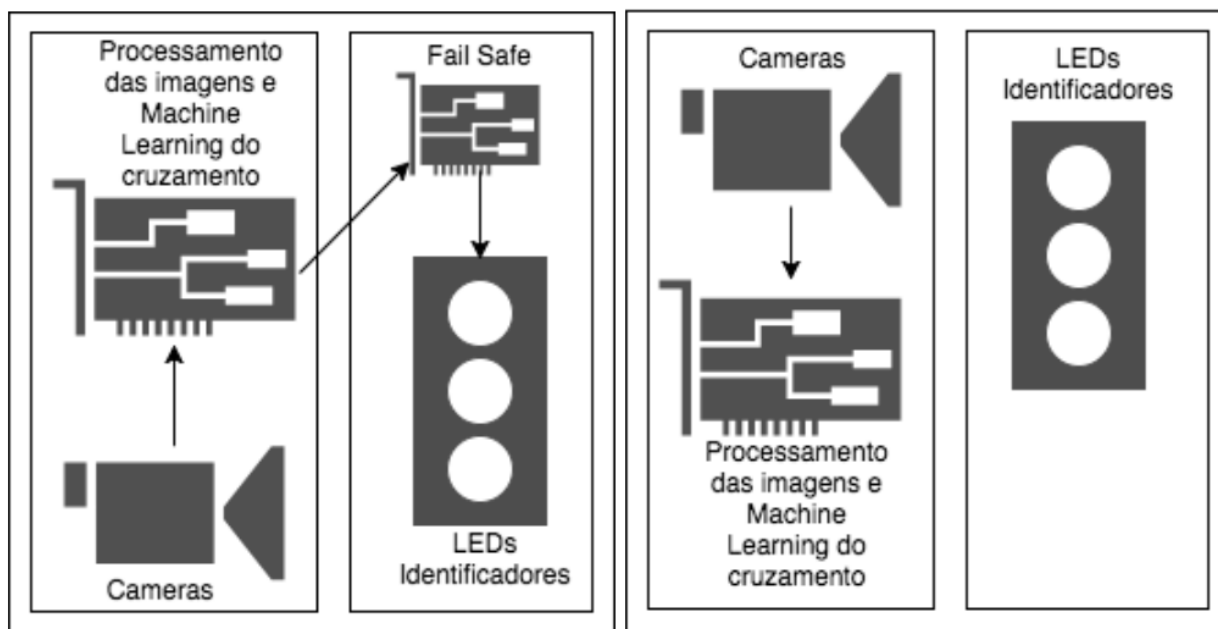


Figura 14: Esquema do Módulo de Semáforos Inteligentes

O módulo de semáforos inteligentes é um método de controle a partir do uso de câmera(s) acopladas aos semáforos com processamento por inteligência artificial.

O sistema é composto por um par de câmeras capazes de identificar objetos e estimar distâncias. As imagens geradas são analisadas pelo algoritmo que, por comparação de outras imagens e padrões pré-definidos, é capaz de reconhecer carros, ônibus, caminhões, entre outros.

Finalmente, a partir dos dados analisados anteriormente, o sistema consegue contar a quantidade de carros que passam pela via monitorada através do hardware de processamento.

#### 7.3.1.1 Método de Reconhecimento de Veículos

Módulo que reconhece os carros existentes a cada instante e consegue acompanhar a movimentação dos carros a fim de contar a quantidade de carros que passaram pela câmera e consequentemente pela via monitorada.

Isso é feito através de um software de reconhecimento de padrões de processamento de imagem, as imagens geradas são comparadas temporalmente (*frame a frame*) e a partir de padrões estabelecidos previamente, o algoritmo é capaz de identificar os veículos motorizados.

Para a realização da identificação de veículos de transporte público ou oficiais seria necessário a instalação de adesivos (*QR CODE*) nos mesmos para facilitar a diferenciação entre esses veículos e os demais.

### *7.3.1.2 Hardware da Câmera*

A câmera propriamente dita, deve possuir um grau de abertura maior que o padrão do mercado, aproximadamente 110°, para garantir uma visão abrangente sobre todas as faixas de uma via. Além de uma visão noturna para garantir que o algoritmo de reconhecimento também funcione em situações de baixa luminosidade, na contagem dos veículos.

Assim um modelo de câmera que poderia ser utilizado inicialmente apresentaria a seguinte configuração:

- Abertura de aproximadamente 110°.
- Alcance de aproximadamente 50 metros.
- Funcionalidade de Infravermelho, para garantir a visão noturna.

### *7.3.1.3 Hardware de Comunicação Interna*

Comunicação entre os dispositivos do mesmo cruzamento, de preferência que não utilize fios, utilizando um baixo bandwidth, ordenados, sem perda de informação e com o mínimo de atraso possível.

A comunicação irá ocorrer por um protocolo de rede LAN, em que cada semáforo enviará a cada instante para o controlador semafórico o número de carros parados e que já passaram pelo cruzamento.

## **7.3.2 Central**

Já a central consiste no controlador semafórico e é a peça chave de todo o cruzamento, uma vez que ela é a responsável por determinar os tempos de verde, vermelho e amarelo de cada semáforo.

Além disso, ela deve possuir conexão com a internet permitindo a troca de informações entre o servidor e a nuvem.

O módulo possui um hardware dedicado a verificação das informações que serão passadas ao semáforos garantindo que não ocorra falhas no cruzamento.

Na Figura 15, é possível observar o diagrama para o fluxo central utilizado no controlador para definir de forma assertiva as ações a serem tomadas.

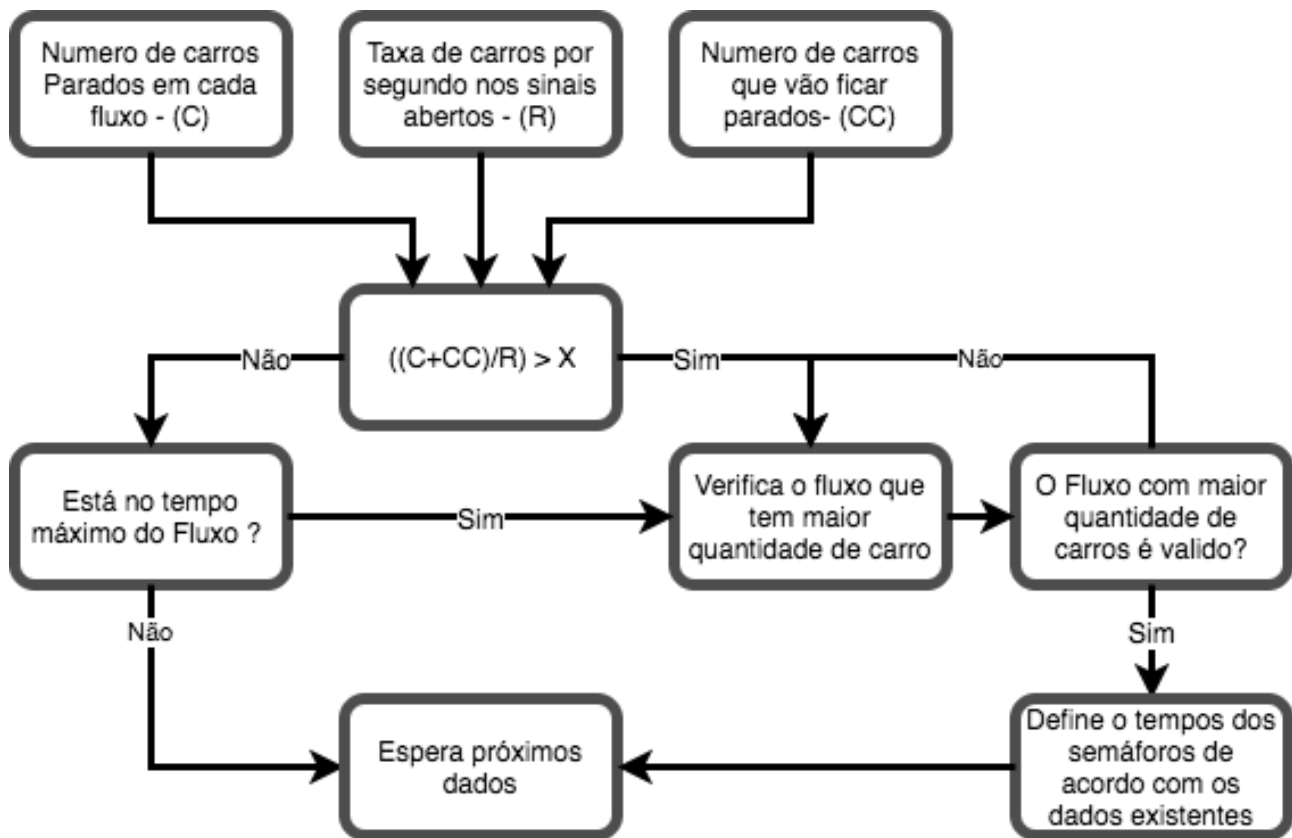


Figura 15: Diagrama do algoritmo do controlador

A variável C no fluxo representa um número fictício de carros gerado através de um pré-processamento dos dados recebidos do módulo de semáforos inteligentes. Já a variável R, representa o taxa de carros passando pelo cruzamento, esse valor é gerado a partir dos dados recebidos pelo módulo 1. A variável CC são os dados recebidos da nuvem depois de um pré-processamento considerando a quantidade de veículos que ficaram paradas em cada sentido no último ciclo.

#### 7.3.2.1 Sistema Fail-Safe

Sistema externo, responsável por gerenciar os LEDs indicativos de todos os semáforos, garantindo que, em hipótese nenhuma, semáforos de duas vias que se cruzam fiquem abertos ao mesmo tempo.

Também, garante-se que em qualquer avaria do sistema (como resultados inconsistentes vindos do aprendizado de máquina), ou falhas gerais/locais, o cruzamento volte a trabalhar na configuração padrão garantindo o seu funcionamento em casos de falha.

Além disso, é importante ressaltar que em casos como esse, o sistema automaticamente informa a central sobre o possível problema, a fim de providenciar de forma rápida os reparos no sistema.



Falhas que devem ser tratadas:

- Caso o sistema de decisão semafórica apresente alguma avaria de informações como, dados desconhecidos, ações não permitidas pela topologia do cruzamento.
- Ausência de informação vinda do sistema de decisão semafórica.

#### *7.3.2.2 Software do Processador*

O software necessita de um hardware modesto, responsável por executar o sistema de decisão semafórica a partir de todas as informações oriundas dos cruzamentos (Módulo de Semáforos Inteligentes).

A cada instante, o sistema vai analisar a quantidade de carros parados em cada via com passagem fechada do cruzamento e a taxa de carros por minuto que estão passando na via(s) aberta(s). Assim, realiza-se uma análise para determinar qual a melhor ação a ser tomada através de algoritmos.

Ele será responsável por retornar um tempo prévio de abertura e fechamento para cada um dos semáforos existentes no cruzamento.

Além disso, define-se o ciclo de um cruzamento como o tempo que o sistema leva para abrir e fechar todos os semáforos do cruzamento. Assim, em todo o ciclo, o algoritmo tem que garantir a passagem dos pedestres pelo menos uma vez em um tempo adequado para a travessia completa da via.

É importante ressaltar que todo ciclo de um cruzamento possuirá um tempo máximo default para a abertura de cada semáforo do cruzamento e que durante o setup do semáforo é necessário definir algumas informações, como quais semáforos não poder ser abertos ao mesmo tempo.

Uma sugestão de preferência seria em ordem crescente de preferência:

- Veículos de Emergência.
- Veículos públicos, como ônibus e vans.
- Demais Veículos.

#### *7.3.2.3 Hardware de Comunicação Externa*

Hardware responsável por enviar as informações úteis do sistema e do trânsito, para o servidor na nuvem. A comunicação será feita através de um protocolo GPRS ou qualquer outro caminho que envie os dados para o servidor na nuvem.

### 7.3.3 Cliente

Já o cliente corresponde a central de controle em que funcionários da Companhia de Tráfego/Prefeitura podem acompanhar o fluxo de veículos nos cruzamentos operantes com o sistema.

Além disso, o sistema trará alguns insumos de trânsito naquele cruzamento, como o tempo atual de cada semáforo e horas de maior fluxo.

É importante ressaltar que outras funcionalidades podem ser implementadas e entregues ao cliente, tais quais a busca por veículos roubados e calcular a velocidade dos carros nas vias.

#### 7.3.3.1 *Visualização dos Insumos de Tráfego*

Plataforma Web em que todas as informações estão disponíveis para os funcionários autorizados.

Assim, os responsáveis pelas centrais terão acesso a todos os insumos de tráfego, como as horas de maior fluxo ou acidentes que ocorreram em tempo real (possibilitando a chamada rápida para os oficiais responsáveis pela resolução do problema).

#### 7.3.3.2 *Controle*

A partir do hardware de comunicação externa é possível que a central de controle altere o tempo de cada semáforo caso necessário, com toda a segurança necessária para que não ocorram fraudes.

Dessa forma, permite-se que em casos específicos o funcionário possa intervir, abrindo o sinal a partir da central.

#### 7.3.3.3 *Alerta de Avarias*

A partir da plataforma web será possível que o sistema reporte à central problemas no hardware do sistema ou incidentes na via:

- A partir do sistema de fail-safe, garante-se que em qualquer tipo de falha ou intervenção na pista, o sistema passe a operar em sua programação padrão. Caso o sistema entre muitas vezes em sua operação padrão ao longo de um dia, o sistema automaticamente informa a central sobre o possível problema, a fim de providenciar de forma rápida os reparos no software ou hardware.

- Em casos de incidentes na pista, como acidentes de trânsito, o sistema automaticamente informa a central sobre o incidente. Assim, os funcionários responsáveis pela central podem chamar rapidamente os órgãos responsáveis pela resolução do problema, como chamar o resgate em caso de acidentes com vítimas.

#### *7.3.3.4 Histórico de Informações*

Existirá uma aplicação na plataforma web que fornecerá um registro de todas as atividades do sistema em um determinado período, ou seja, toda a informação recebida no semáforo será armazenada em um banco de dados relacional.

Assim será possível a recuperação de informações nos últimos anos, a fim de facilitar processos de auditoria e análises sobre o sistema, tais quais quantas vezes o semáforo apresentou falhas ou em que momentos o operador da central precisou realizar alguma intervenção manual.

#### *7.3.3.5 Sistema de Atualização*

Através da plataforma web, o sistema indicará que existem atualizações de software disponíveis para aquele cruzamento.

A atualização ocorrerá em background de modo que a operação continue e é feita através do módulo de comunicação externa em que a central envia a informação de update para o semáforo, o semáforo recebe a informação e busca a atualização na cloud através do seu hardware de comunicação com o servidor.

Durante a atualização, o operador responsável deve garantir que os procedimentos de segurança a seguir sejam seguidos:

1. A atualização deve ser aprovada por pelo menos o operador e seu supervisor.
2. Deve ser realizada no período de menor fluxo no cruzamento (esse dado é fornecido pelo log de informações do semáforo).
3. Durante a atualização, o semáforo deve operar no seu modo de fail-safe.
4. No momento que for necessário reiniciar o sistema, um agente de trânsito deve estar presente no cruzamento para garantir a ordem e a segurança dos veículos.

## 8. Um estudo sobre Processamento de Imagens

Nos últimos capítulos, estudou-se sobre como é feita a gestão de tráfego em grandes cidades e foi proposto um sistema para alterar o paradigma do funcionamento de semáforos em cruzamentos urbanos.

A partir de agora o estudo se propõe a trabalhar em modelar os algoritmos que podem compor o sistema proposto pelo projeto.

Para isso, será feita uma breve revisão de processamento de imagens e de propostas para a identificação de objetos e veículos utilizando Visão Computacional, no Capítulo 6.

Feito isso, o Capítulo 7 será focado em desenvolver as possíveis soluções propostas na modelagem do sistema do capítulo anterior.

### 8.1 Imagem Digital

Uma imagem digital é uma função bidimensional,  $f(x, y)$ , em que  $x$  e  $y$  são coordenadas de um plano espacial. Além disso o valor da função retorna a intensidade do pixel ou nível de cinza da imagem desse ponto, ou seja, uma imagem é formada por milhões de pixels (a menor unidade constituinte em processamento de imagens).

Quando os valores das coordenadas e de  $f$  são discretas, obtém-se a chamada imagem digital.

O valor de cada ponto da imagem possui um valor numérico que é limitado conforme a escala de cor escolhida. Por exemplo, na escala RGB, os níveis de cinza vão de 0 a 255 e são usados 8 bits por pixel.

O formato de cores RGB é o modelo mais usado no mundo em que ele se baseia nas cores primárias Vermelho, Verde e Azul e pode gerar mais de 16 milhões de cores, ou seja,  $256^3$  utilizando a combinação das três cores primárias utilizadas. Também destacam-se os modelos CMYK, usado para a fabricação de tonners para a impressão e o HSI.

### 8.2 Escala de Cinza

A escala de cinza representa um modelo de imagem formado por pixels em que a intensidade pode variar entre tons de preto e de branco. Assim combinados, eles podem formar tonalidades de cinza. Neste caso, quanto maior o valor de cinza de um pixel, maior a quantidade de luz no mesmo dado que o valor máximo de 1 é o tom branco.

Em pesquisas e trabalhos que envolvem processamento de imagens é bastante comum a utilização da escala de cinza para evitar distorções causadas pela luz e por ter um menor custo computacional, de acordo com Gonzalez e Woods [13].

## 8.3 Contraste

Define-se contraste como a diferença entre os pontos de intensidade máxima e mínima de uma imagem. Imagens com alta variação de intensidade possuem alto contraste. Já imagens de baixo contraste tem a impressão de pouco brilho.

Problemas no contraste são causadas por baixa ou intensa iluminação, além problemas na construção da mesma, o que faz com o histograma da imagem não apresente um caráter uniforme.

Para garantir uma definição melhor e uma assertividade maior durante o tratamento de uma imagem, utilizam-se algumas técnicas de realce de contraste. No estudo, será discutido o método de equalização do histograma.

### 8.3.1 Equalização do Histograma

O processo de equalização de histograma é, segundo Patel e Goswami[14], um método para normalizar a distribuição da probabilidade de ocorrência de valores de intensidade na imagem, ou seja, as intensidades dos pontos da imagem são modificados de forma com que o histograma tenha a forma desejada.

Para a execução deste filtro, tal mapeamento é realizado através da função de distribuição acumulativa. Para o histograma  $H(i)$ , a sua distribuição acumulada é:

$$H'(i) = \sum_{0 < j < i} H(j)$$

Com o histograma  $H'(i)$  obtido, cada pixel da imagem equalizada é calculado a partir do valor de intensidade de cada pixel da imagem origem através da equação:

$$\textit{Imagem Equalizada}(x, y) = H'(\textit{Imagem}(x, y))$$

Com a aplicação do processo de equalização de histograma é possível perceber uma melhora significativa, especialmente no contraste, em imagens. Contudo, se aplicarmos este processo em uma

imagem cujos pixels já estão espalhados por toda a escala de intensidade, o efeito praticamente não será notado.

## **8.4 Filtros**

A filtragem no domínio espacial é uma técnica de processamento de imagens que normalmente manipula os valores de uma vizinhança para modificar a imagem digital (atuando diretamente nos pixels da imagem). Em geral, esta técnica é computacionalmente mais eficiente e requer menos recursos de processamento para sua realização. A imagem digital é representada por uma matriz de pixels, sendo que a filtragem no domínio espacial trabalha com uma máscara, uma janela, que percorre toda a imagem realizando as operações de filtragem desejadas.

A aplicação de filtros é uma técnica de processamento de imagens que manipula os valores dos pixels mais próximos ao pixel analisado por meio de uma máscara (janela deslizante) de forma a realizar alguma alteração pertinente ao processo sendo realizado.

Se a operação realizada sobre os pixels da imagem for linear, o filtro é chamada de filtragem espacial linear; caso contrário, é chamado de filtragem espacial não linear.

Além disso, como a imagem é representada por uma matriz, para realizar a operação basta definir uma máscara e desliza-la por todos os pixels da imagem.

### **8.4.1 Filtro da Média**

É um filtro linear de suavização, ou seja, ele trabalha diretamente nos valores de média dos pixels.

A aplicação do filtro consiste em substituir o pixel analisado pela média dos seus vizinhos e o resultado esperado é a perda de nitidez da imagem.

Também é interessante ressaltar que filtros com essas características são conhecidos como passa-baixa.

### **8.4.2 Filtro da Mediana**

Assim como o Filtro da Média, o Filtro da Mediana também é linear e consiste em substituir o pixel analisado pela mediana dos pixels na sua vizinhança.

O resultado esperado de sua aplicação é o aumento do foco e nitidez da imagem.

## 8.5 Operadores de Bordas

Agora, quando se tem a necessidade de identificar bordas ou outra característica como curvas de uma imagem, os filtros mais utilizados são Roberts, Sobel e Prewit.

Por exemplo, o Filtro de Sobel é utilizado para identificar bordas e contornos dos objetos da imagem. Para isso, ele calcula o gradiente da intensidade da imagem em cada ponto, dando a direcção da maior variação de claro para escuro e a quantidade de variação nessa direcção. Assim, obtém-se uma noção de como varia a luminosidade em cada ponto e as variações intensas de luminosidade correspondem às bordas dos objetos.

## 8.6 Operadores Morfológicos

O objetivo da morfologia matemática é extrair informações relativas à geometria e topologia de uma imagem, através de outro conjunto totalmente definido, denominado elemento estruturante. Conhecendo o tamanho e o elemento estruturante é possível realizar algumas operações, como a dilatação e a erosão, que realizam algumas alterações nos pixels.

Na erosão os pixels que se enquadram no elemento estruturante são deletados, enquanto que na dilatação, de modo análogo, o conjunto dos pixels tem a sua forma alterada.

Os operadores morfológicos são amplamente utilizados em processamento de imagens, uma vez que eles são utilizados nos processos de abertura e fechamento.

A abertura consiste na aplicação de uma erosão seguida de uma dilatação. O resultado esperado da aplicação é suavizar o contorno dos objetos e eliminar pontos isolados em imagens binárias.

Já o fechamento se baseia na operação inversa a abertura em que se aplica uma dilatação seguida de uma erosão. Tal processo tem por objetivo de fundir descontinuidades e alonga formas finas.

### 8.6.1 Erosão

A erosão diminui ou afina um objeto em uma imagem binária, sendo que detalhes menores do que o elemento estruturante são removidos .

A erosão de A por B é indicada por  $E(A, B)$ , é definida como:

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

### 8.6.2 Dilatação

Já a dilatação engrossa os contornos de um objeto. A forma específica e a extensão são determinados pelo formato do elemento estruturante.

A dilatação de A por B, indicada por  $D(A, B)$ , é definida como:

$$A \oplus B = \{z \mid [(B^{\wedge})z \cap A] \subseteq A\}$$



## 9. Aplicações de Visão Computacional

Após estudar e discutir alguns conceitos de processamento de imagens, o trabalho analisará algumas aplicações de Visão Computacional no campo de mobilidade urbana.

Assim é possível criar tecnologias para otimizar a locomoção de pessoas e aumentar a segurança no trânsito com veículos e semáforos autônomos, além de assistentes pessoais para alguns indivíduos, como deficientes visuais. Desta forma, seria possível criar uma cidade realmente conectada e inteligente, a qual toma decisões conforme algumas situações e padrões identificados pelos algoritmos de identificação e análise de imagens, ou seja, obtêm-se as chamadas Smart Cities.

Para isso, utiliza-se o campo de Visão Computacional, o qual possui esse nome por representar algoritmos que tentam simular a visão humana, ou seja, a partir das imagens processadas e capturas, um sistema de computador analisa a mesma e determina qual a melhor ação para essa situação conforme padrões pré-definidos.

No campo de simulações de tráfego e contagem de veículos passando por um cruzamento será analisado a área de segmentação de objetos para separar os objetos do fundo. A partir disso, será possível identificar os objetos e classificá-los, como veículos, pessoas e árvores.

No entanto, analisando o trabalho de Gonzalez e Woods [13], a aplicação de algoritmos de processamento de imagens no tráfego urbano apresenta diversas dificuldades, uma vez que as informações são coletadas em ambientes abertos, ou seja, o processamento das imagens sofre influências das variações de luz e clima, além de que os objetos desejados possuem padrões muito diferentes, tremor da câmera devido ao vento, visibilidade na chuva e ocorrência de casos de oclusão em que um objeto se sobrepõe ao outro e atrapalha na identificação.

Ainda assim os processos de contagem de veículos existentes podem ser divididos em alguns grupos, conforme Yang[15]:

### 9.1 Subtração de Fundo

Os algoritmos existentes para esse grupo se baseiam na utilização de duas imagens, a imagem a ser analisada e uma imagem de referência de fundo.

Assim, a ideia é subtrair as duas imagens para a obtenção e identificação dos objetos.

Além disso é importante a definição de um limiar em que o pixel que for inferior ao limiar pertence ao fundo.

Entretanto, esse método sofre grande interferência dos processos de variação da iluminação e da oclusão.

Uma alternativa para melhorar o problema da volatilidade da intensidade de luz, seria utilizar uma imagem de referência para cada período do dia.

## **9.2 Aprendizado de Máquina**

Já o grupo de Aprendizado de Máquina consiste no ato de treinar e ensinar um computador uma série de formas e padrões de forma que a máquina consiga interpretar os fatos e tomar decisões de forma autônoma.

Assim, a detecção de objetos consiste em uma classificação binária que separa os objetos do fundo. Feito isso, basta selecionar o objeto e identificá-lo conforme sua base de dado.

No entanto, esse procedimento sofre grande interferência do processo de oclusão e a variedade de modelos de veículos, uma vez que existe a necessidade de catalogar a maioria dos tipos de veículos e ainda assim tem o problema que sempre os modelos sofrem alterações das montadoras.

## **9.3 Modelo de Borda Correspondente**

Esse método consiste em detectar os carros através da segmentação das bordas da imagem. Assim, os objetos identificados são comparadas às bordas de um modelo de veículo pré-definido.

Embora o processamento computacional para a contagem volumétrica seja baixo, essa abordagem apresenta o problema de que ela acaba não sendo muito precisa e exige a existência de um banco de dados gigantesco para o processo de contagem classificada.

## **9.4 Rastreamento e Agrupamento de Pontos Correspondentes**

Para esse grupo, a segmentação consiste em detectar e monitorar um conjunto de pontos relacionados às características do objeto ao longo de alguns frames.

Então, agrupa-se os possíveis objetos e é feita a classificação baseada em uma base de dados existentes de veículos.

## 10. Algoritmo Proposto

Finalmente, chega-se a etapa final desse estudo e ao resultado prático das análises dos capítulos anteriores.

Foi comentado que a Visão Computacional é uma área de computação que tem o objetivo de extrair informações a partir de imagens digitais pré-processadas.

Assim, a partir das definições e referencial teórico estudados anteriormente, o estudo irá propor um método para a identificação de veículos para o sistema modelado no Capítulo 7. Para isso, o capítulo será dividido em duas etapas distintas.

Em um primeiro momento, será definido um método ingênuo para a identificação de veículos, ou seja, a proposta apenas identifica algum veículo, não fazendo a distinção e classificação dos mesmos.

Feito isso, a partir do método inicial, será feita uma análise do software de reconhecimento de objetos em tempo real, chamada Yolo.

### 10.1 Método de Identificação de Veículos

O primeiro método consiste em realizar um processo de identificação simples de veículos. Para isso, o estudo será dividido em algumas etapas: entrada dos dados, processamento inicial da imagem, reconhecer imagem de fundo, segmentar objetos.

#### 10.1.1 Entrada dos Dados

A entrada consiste no vídeo fornecido pelas câmeras instaladas nos semáforos. O vídeo será segmentado em diversos frames para ter uma imagem de cada instante de tempo.

#### 10.1.2 Processamento Inicial da Imagem

Para facilitar o processo de identificação dos veículos, serão aplicados alguns filtros nos frames para otimizar o processamento e reduzir diferenças de ruído e iluminação das imagens a serem comparadas.

Inicialmente, a imagem será levada para a escala de cinza. A transformação da imagem de entrada para a escala de cinza reduz de forma significativa a complexidade do processamento dos dados. No entanto, perde-se a cor do objeto, a qual é um elemento importante para a classificação.

Feito isso, a imagem passa pelo processo de equalização do histograma para as diferenças na intensidade de luz entre os pixels de frames diferentes serem mínimas. Também é necessário aplicar algum filtro de suavização passa-baixa para reduzir os ruídos da imagem.

Neste caso, será escolhido o filtro Gaussiano, haja vista que o mesmo apresenta bons resultados na remoção de ruído impulsivo e é um bom conservador de bordas naturais dado que preserva descontinuidades sem borrar.

Algoritmo 1: Processamento Inicial da Imagem

**Entrada:** *Frame de Entrada*

**Saída:** *Imagem pré-processada*

$img \leftarrow \text{Equalização do Histograma}(input)$

$img \leftarrow \text{Aplicação do filtro Gaussiano}(img)$

A equalização do histograma é extremamente simples, uma vez que basta utilizar a função de "equalizeHist()" disponível na biblioteca OpenCV [30] que recebe a imagem a ser equalizada como entrada.

Em seguida, para aplicar o filtro Gaussiano, basta definir uma janela deslizante e calcular a média entre os pixels vizinhos ao pixel central.

#### 10.1.2.1 Testes e Resultados

A fim de testar o funcionamento do sistema, o algoritmo 1 foi executado para uma imagem arbitrária retirada da internet:



Figura 16: Imagem utilizada como Entrada do Algoritmo

Inicialmente a Figura 16 é convertida para a escala de cinza:



Figura 17: Figura 16 convertida para a Escala de Cinza

Feito isso, o algoritmo 1 realiza a equalização do histograma e finalmente aplica um Filtro Gaussiano para suavizar a imagem e reduzir ruídos.



Figura 18: Imagem com o histograma equalizado e com o Filtro Gaussiano aplicado



Figura 19: Comparação entre a imagem original na escala de cinza(a) e a imagem de saída com o filtro Gaussiano(b)

### 10.1.3 Reconhecer a Imagem de Fundo

O reconhecimento da imagem de fundo é uma etapa extremamente importante do procedimento para identificar os objetos, uma vez que precisamos diferenciar o que é fundo e o quais são os potenciais objetos de interesse.

Uma ideia simples e inicial consiste em escolher uma imagem de fundo em que não possua objetos em movimento ou estacionados na via para alguns períodos do dia, manhã, meio-dia, tarde, noite e madrugada.

Para análises estacionárias, as quais denominamos estudos realizados em ambientes controlados e que não exigem um monitoramento em tempo real, o procedimento é efetivo.

No entanto, para o estudo, essa estratégia pode não obter bons resultados, em função do vento, variação de luminosidade dentre outros.

Assim, o estudo tentará propor um outro método baseado no método desenvolvido por Gupta [18].

O método consiste em calcular a média de cada um dos pixels da imagem, uma vez que o valor médio é aquele que mais se repete e o fundo deve corresponder a uma quantidade de pixels extremamente superior do que os objetos.

Para isso, cria-se uma matriz com as mesmas dimensões que os frames de entrada e vai acumulando o valor do pixel para cada frame recebido.

Quando o último frame é recebido, basta dividir o valor de cada pixel pela quantidade de quadros e é obtida a imagem de fundo.

Tal procedimento apresenta bons resultados, uma vez que o processamento é rápido e pode ser atualizado de forma simples, algo necessário para o sistema proposto, dado que as condições de tráfego se alteram rapidamente.

Algoritmo 2: Definir Imagem de fundo

**Entrada:** Número de frames (*frame\_number*) e sequência de imagens pré-processadas (*img*)

**Saída:** Imagem de fundo

*frame\_count*  $\leftarrow 0$

enquanto *frame\_count*  $\neq$  *frame\_number*, faça:

*img*  $\leftarrow$  ler nova imagem

*x*  $\leftarrow$  altura da imagem

*y*  $\leftarrow$  largura da imagem

    Se *frame\_number* = 0, então:

*acc\_matrix*  $\leftarrow$  *img*

    Se não:

        Para *i* de 1 até *x*, faça:

            Para *j* de 1 até *y*, faça:

$acc\_matrix[i, j] = acc\_matrix[i, j] +$   
  *img[i, j]*

*frame\_count*  $\leftarrow$  *frame\_count* + 1

*background* = *acc\_matrix* / *frame\_number*

### 10.1.3.1 Testes e Resultados

Para o algoritmo 2 não foi possível a realização de testes dado que não foi encontrada uma sequência de imagens de um mesmo referencial que servisse de teste para o problema.

Além disso, o estudo não conseguiu autorização da prefeitura de São Paulo para coletar dados do trânsito a partir de algum semáforo. De forma a exemplificar o funcionamento do algoritmo, segue na Figura 20, abaixo, os testes realizados por Gupta[18] para comprovar o funcionamento de sua teoria e cuja qual o algoritmo 3 foi baseado.

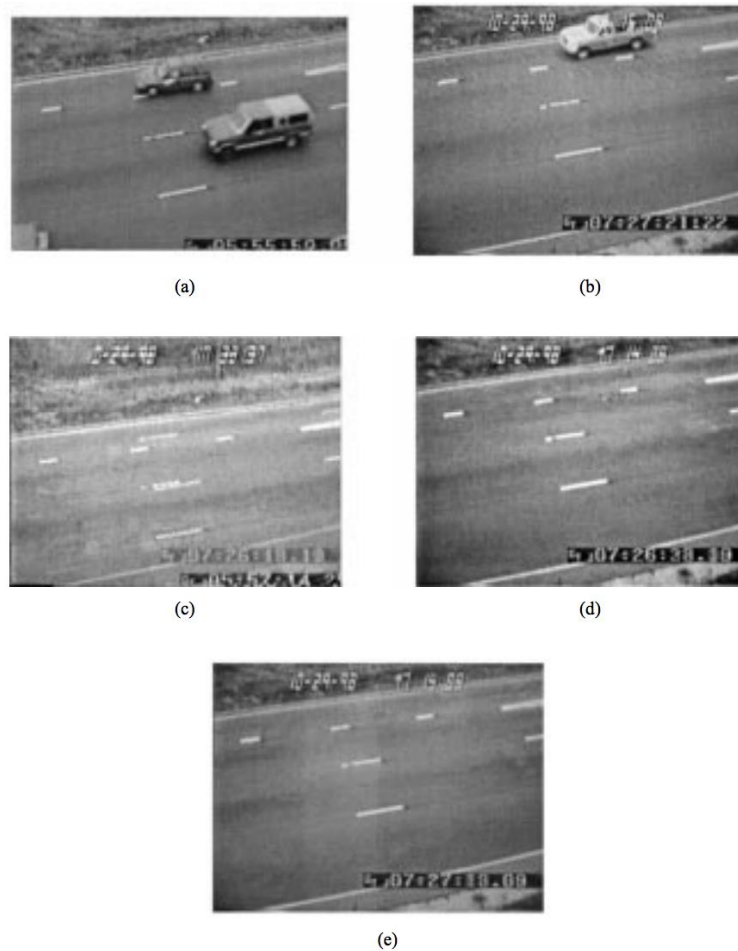


Figura 20: Testes realizados por Gupta[18] para a identificação da imagem de fundo de referência.

As imagens (a) e (b) representam frames consecutivos da via analisada e a imagem ( c ) representa o fundo identificado pelo algoritmo pelas duas imagens anteriores, isto é, foi feita a média



de todos os pixels da imagem (a) e (b), o que resultou na imagem ( c ). De modo análogo, as figuras (d) e (e) representam as imagens de fundo de referência para segundos posteriores.

#### 10.1.4 Segmentar Objetos

Com a imagem de fundo reconhecida, chega-se a etapa mais importante do primeiro método estudado, uma vez que uma segmentação mal feita pode comprometer todo o trabalho já feito. Assim, essa etapa consiste em selecionar os objetos de interesse e descartar os demais.

A partir da imagem de referência de fundo identificada no item acima, sabe-se o que é fundo ou não. Assim, toma-se os pixels que não fazem parte do fundo.

Para esses pixels basta aplicar algum operador de borda, como Sobel, e determinar as bordas e contornos dos objetos para frames subsequentes. Assim, compara-se frames para identificar se um objeto está em movimento e caso esteja em movimento, basta verificar se a área do objeto está contida nos tamanhos máximos e mínimos de veículos.

A definição dos tamanhos médios dos veículos foi feita com base em pesquisas na literatura e em catálogos de veículos [16][17].

Tabela 7: Medidas padrões de automóveis

Tipo de veículo	Comprimento (m)	Largura (m)
Moto	2,00	1,00
Pequeno	4,20	2,00
Médio	4,70	2,10
Grande	5,50	2,50

Assim, define-se as áreas padrões de veículos acrescidos de um erro de 2,5% para os seguintes valores:

- Moto:  $(2 * 1) * 1,025 = 2,05\text{m}^2$
- Veículo Mínimo:  $(4,2 * 2) * 1,025 = 8,61\text{m}^2$

- Veículo Máximo:  $(5,5 * 2,5) * 1,025 = 14,1m^2$

Tabela 8: Medidas de Caminhões

Tipo de Caminhão	Comprimento (m)	Largura (m)
Veículo Urbano de Carga	6,80	2,20
Leve	14,00m	2,6
Semi-Pesado	14,00m	2,6m
Pesado	20,00m	6,3m

Assim, define-se as áreas padrões de modo análogo ao anterior:

- Caminhão Mínimo:  $(6,8 * 2,2) * 1,025 = 15,34m^2$
- Caminhão Máximo:  $(20 * 6,3) * 1,025 = 129,15m^2$

Já para os ônibus urbanos, estima-se que seu comprimento varia de 11 a 15 m e sua largura de 2,2 a 2,8 m. Assim, chega-se:

- Ônibus Mínimo:  $(11 * 2,2) * 1,025 = 24,805m^2$
- Ônibus Máximo:  $(15 * 2,8) * 1,025 = 43,05m^2$

Para definir se um objeto pode ser classificado como um veículo, o algoritmo proposto toma a área do objeto e classifica com base nas tabelas.

Embora o método seja extremamente simples, ele é um método rápido e eficiente para determinar de forma aproximada a contagem volumétrica de veículos. Além disso, é um método muito mais preciso para a definição dos tempos de abertura e fechamento dos semáforos do que o método de contagem manual praticado pelas Companhias de Tráfego.

No entanto, ele apresenta a desvantagem de que ele não é 100% preciso e não permite uma contagem classificada precisa que o sistema da SmartFlow exige.

O algoritmo 3 consiste em delimitar e definir os contornos dos objetos utilizando o operador de Sobel e a função de contours do OpenCV. Com os contornos definidos, tomar a área de cada objeto e comparar seu tamanho com os tamanhos mínimos e máximos de cada segmento.

Algoritmo 3: Segmentação do Objeto

**Entrada:** Imagem pré-processada (*img\_input*)

**Saída:** Dicionário com a quantidade de cada tipo de veículo  
(*vehcles*)

1. *img\_canny*  $\leftarrow$  *Canny*(*img\_input*)

2. *img\_contour*, *contours*  $\leftarrow$  *FindContours*(*img\_canny*)

3. *DrawContours*(*img\_contour*)

4. *img\_area*  $\leftarrow$  *contourArea*(*contours*)

5. Se *img\_area*  $\geq$  *Veículo Min* && *img\_area*  $<$  *Veículo Max*

6.     *vehcles*[*car*] += 1

7. Se *img\_area*  $\geq$  *Onibus Min* && *img\_area*  $<$  *Onibus Max*

8.     *vehcles*[*bus*] += 1

9. Se *img\_area*  $\geq$  *Caminhao Min* && *img\_area*  $<$  *Caminhao*

*Max*

10.    *vehcles*[*truck*] += 1

#### 10.1.4.1 Testes e Resultados

Para averiguar o funcionamento do Algoritmo 3, foram feitos alguns testes para comprovar seu funcionamento.

Inicialmente, tomou-se a Figura 16, a qual já foi pré-processada no algoritmo 1. A partir dela, foi aplicado o operador de Canny para definir as bordas dos objetos e facilitar a segmentação.

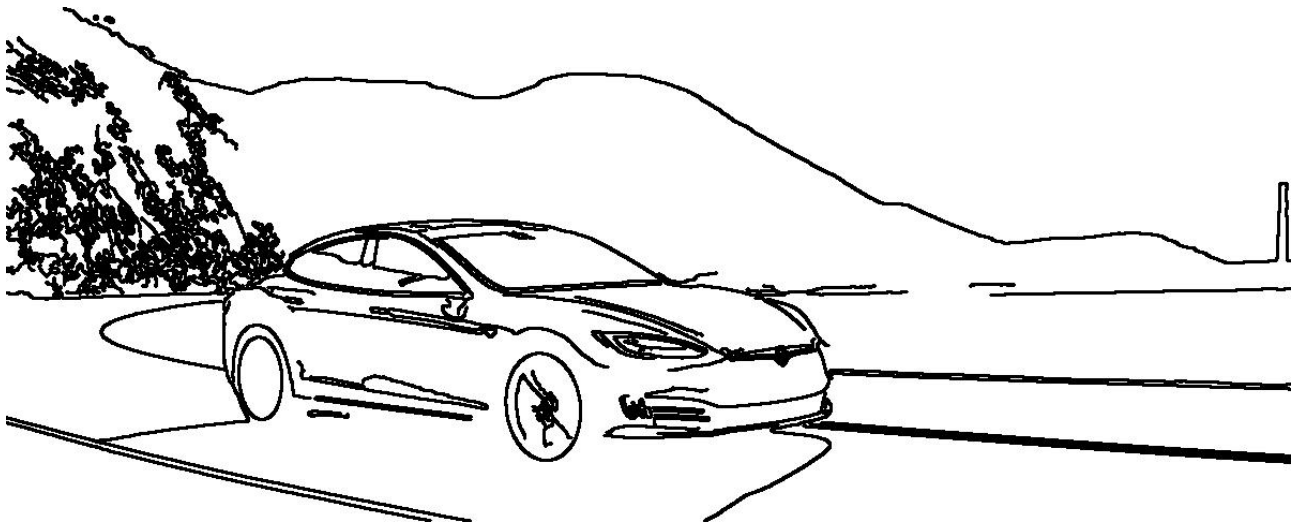


Figura 21: Imagem com os contornos dos objetos traçados

No entanto, como não existiam outras imagens com esse mesmo fundo, não foi possível determinar o fundo da imagem. Algo que impossibilitou a segmentação do objeto.

Assim, foi necessário a utilização de imagens mais simples e sem fundo para analisar a segmentação dos objetos:



Figura 22: Imagem de referência de um carro

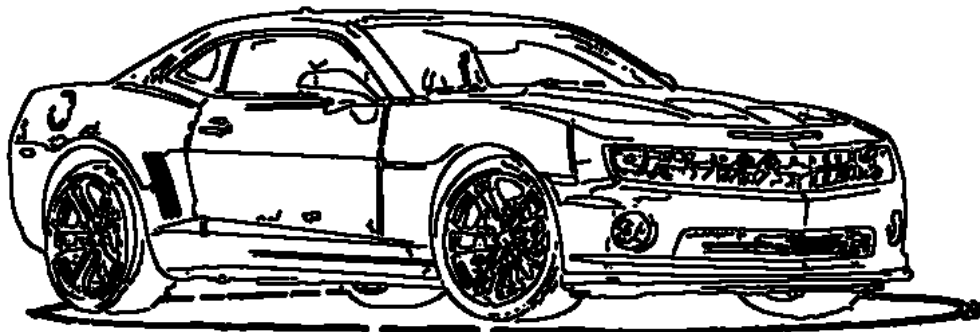


Figura 23: Imagem de referência de um carro com o operador de Canny



Figura 24: Imagem de referência de um ônibus

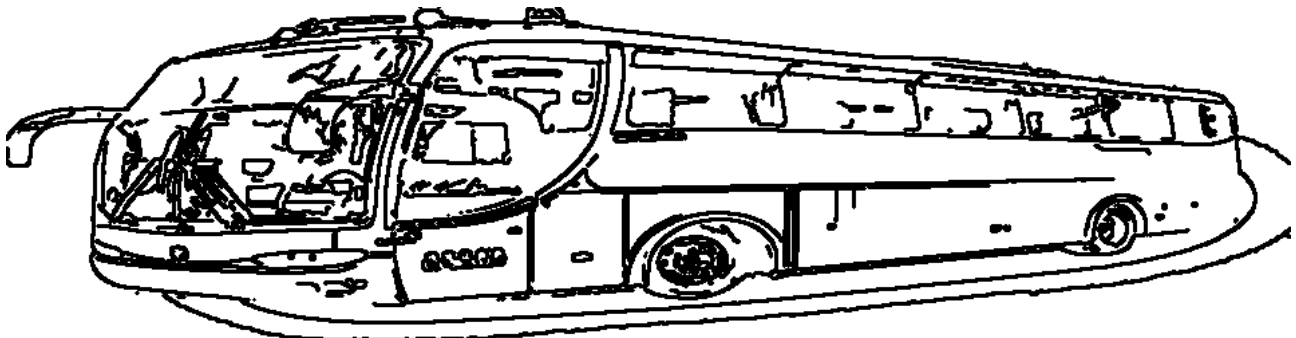


Figura 25: Imagem de referência de ônibus com o operador de Canny



Figura 26: Imagem de referência de um caminhão

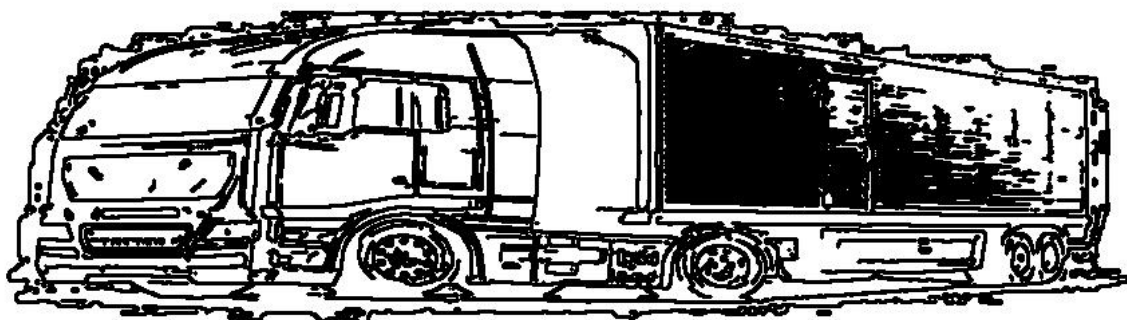


Figura 27: Imagem de referência de um caminhão com o operador de Canny

A partir da Figura 23, Figura 25 e Figura 27 seria possível tentar classificar os objetos de interesse.

No entanto, o estudo se deparou com um seguinte problema, as imagens utilizadas não refletem as dimensões reais dos veículos. Assim, foi necessário realizar uma conversão de escala para cada imagem para se chegar as dimensões obtidas.

Tabela 9: Resultado da Segmentação

Imagem	Área (m2)	Resultado
Figura 31	10,794	Carro
Figura 33	34,823	Ônibus
Figura 35	57,020	Caminhão

O método acima consegue de maneira simples e eficiente contar e até classificar os veículos conforme os padrões pré-estabelecidos, de acordo com a Tabela 9.

Entretanto, tal método possui algumas deficiências devido a algumas aproximações. Por utilizar, um algoritmo baseado em médias das imagens para calcular a imagem de fundo de referência, a média calculada pode representar imperfeições eventuais.

Além disso, o fato de classificar os objetos a partir de comparações entre as áreas dos seus contornos com uma tabela pré-definida, apresenta alguns problemas de confiabilidade, haja vista que objetos aleatórios, como uma caçamba, podem ser identificada como um veículo, os tamanhos dos veículos são aproximações genéricas e podem existir conflitos e oclusão dos veículos, isto é, um veículo pode ser parcialmente ocultado por outro.

Isto poderia ser resolvido com um processamento de imagens áreas para pista de um cruzamento, mas isso elevaria de forma significativa os custos financeiros do sistema por cruzamento, algo que representaria uma perda do diferencial estratégico do projeto perante as alternativas existentes. Também é importante ressaltar que o mercado de automóveis possui grande volatilidade. Assim, todo ano novos modelos são lançados e suas dimensões alteradas.

Uma solução possível para esse problema seria manter um banco de dados relacional com as informações de todos os veículos registrados no Brasil e mundo. Assim para cada tamanho de objeto seria feita uma busca para encontrar o tipo de veículo a ser classificado.

Outro ponto interessante que pode ser implementado seria definir a área de movimento de veículos, ou seja, corresponde a área em que se encontra o maior número de veículos se

movimentando. Tal algoritmo pode melhorar os resultados obtidos pelo método em função de que a realização da segmentação dos objetos reduz as chances de identificar automóveis estacionados ou objetos estranhos.

## 10.2 Sistema de detecção de objetos em tempo real

No item 10.1 foi estudado um método simples e de certo modo eficiente para a contagem de veículos. Entretanto, tal abordagem não apresenta a precisão necessária para o funcionamento por completo do sistema proposto, uma vez que a classificação dos veículos é feita com base em aproximações de áreas e ocorrências de oclusão entre os veículos podem alterar a classificação dos objetos. Além de que seria impossível classificar os veículos com mais prioridade, como os de polícia e emergenciais.

Assim, o estudo fará uso do software de código aberto Yolo, um sistema para a detecção de objetos em tempo real.

### 10.2.1 Yolo

A ferramenta pode identificar mais de 20 tipos de objetos, tais quais carros, motos, ônibus e veículos oficiais. Além disso, existe a possibilidade de você mesmo treinar o algoritmo para identificar novos tipos de objetos [20].

O funcionamento desse classificador se baseia em aplicar uma rede neural em todos os pixels da imagem selecionada. Feito isso, o algoritmo divide a imagem em regiões e determina as probabilidades de se encontrar determinado objeto em tal região, como pode ser visto na imagem retirada do artigo oficial do Yolo [19]:

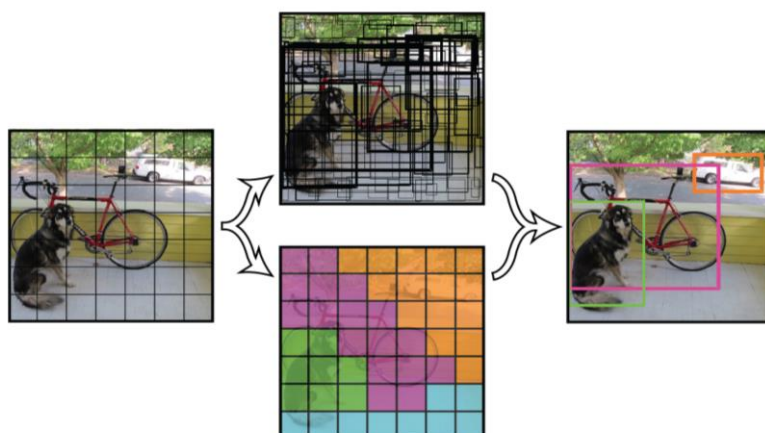


Figura 28: Exemplo de funcionamento do Yolo



Após o procedimento da Figura 28, basta aplicar algum limiar para selecionar e delimitar os objetos.

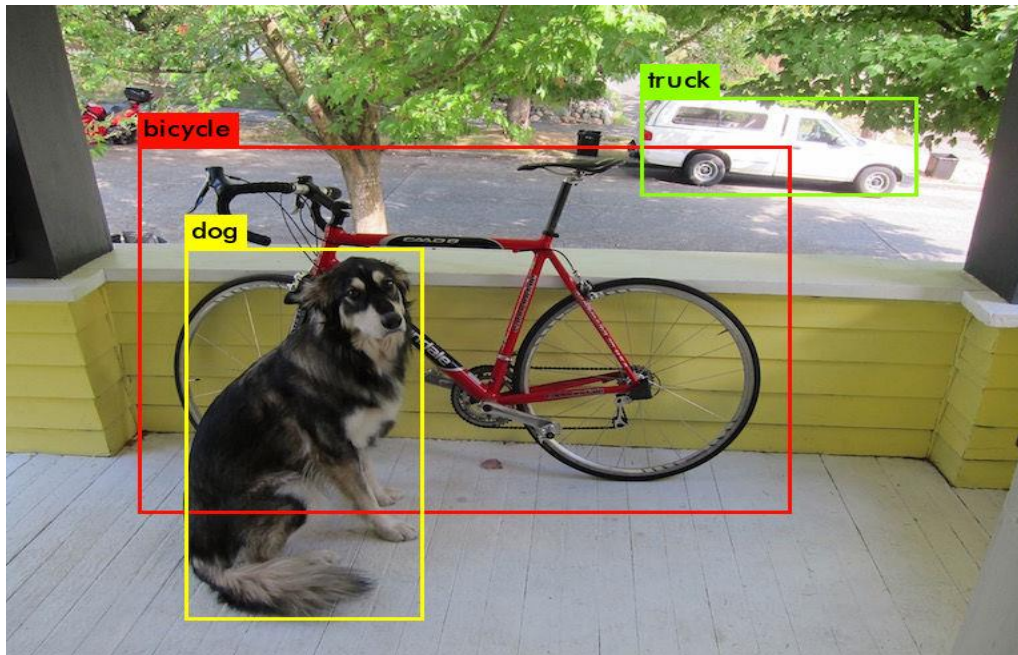


Figura 29: Objetos detectados com Yolo

### 10.2.2 Comparação entre Sistemas de Detecção de Objetos

Comparando com outros sistemas, o Yolo possui a vantagem de que a predição olha para a imagem como um todo e utiliza uma única rede neural o que representa ganhos significativos no processamento das predições.



Tal resultado pode ser visto da imagem abaixo extraída de [19]:

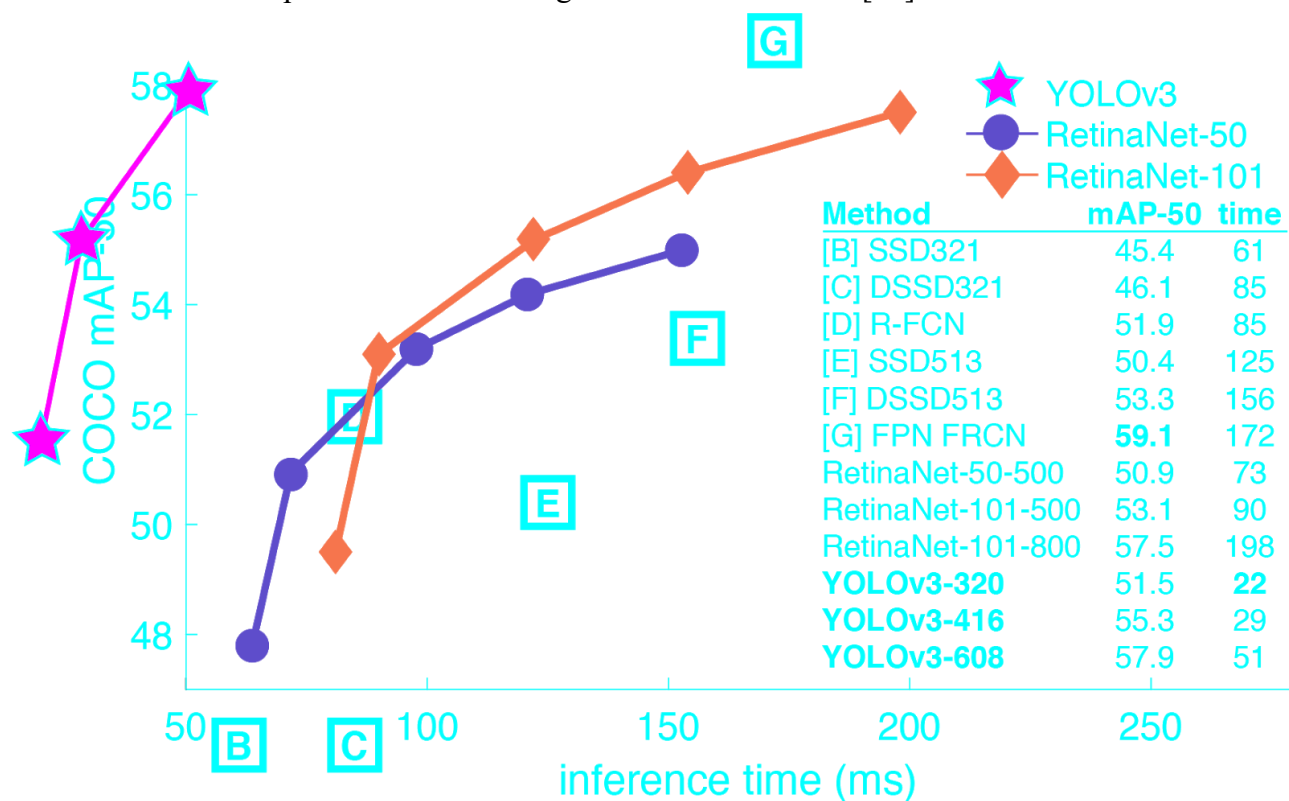


Figura 30: Comparativo do Yolo com outros sistemas de classificação

Tabela 10: Speedup obtido pela utilização do Yolo perante outros sistemas

Método	YOLOv3-320	YOLOv3-416	YOLOv3-608
SSD321	2,77	2,10	1,20
DSSD321	3,86	2,93	1,67
R-FCN	3,86	2,93	1,67
SSD513	5,68	4,31	2,45
DSSD513	7,10	5,38	3,06
FPN FRCN	7,82	5,93	3,37
RetinaNet-50-500	3,31	2,52	1,43

RetinaNet-101-500	4,10	3,10	1,76
RetinaNet-101-800	9	6,83	3,88

Dado que o Speedup é um número que mede a performance relativa entre dos sistemas e definida por:  $S = T_{\text{lento}} / T_{\text{rápido}}$ , em que T representa o tempo de cada sistema.

Analisando a Tabela 11, é possível verificar que todas as versões do YOLO apresentam um ganho de performance significativo em relação aos outros sistemas de detecção de imagens em tempo real, uma vez que o processamento utiliza apenas uma rede neural, algo que melhora bastante o desempenho.

Também é importante ressaltar que a versão 320 possui melhor desempenho do que a 416 e a 618. Isso se deve ao fato de que a primeira versão processa menos detalhes e utiliza menos memória de processamento do que as demais.

### 10.2.3 Execução com CUDA

Uma forma de otimizar o processamento e segmentação dos objetos consiste em compilar o sistema com o CUDA, uma API desenvolvida pela Nvidia para a computação de algoritmos em paralelo em que o processamento é realizado diretamente na GPU [28].

Essa opção do YOLO é extremamente interessante para o sistema, uma vez que o trânsito é extremamente volátil, o que torna necessário o processamento de diversos frames para pequenas frações de segundo.

Além disso, pelo processamento ser realizado localmente a disponibilidade de memória não é grande. Assim se faz necessário um processamento rápido e que gaste pouca memória no processo.

É importante ressaltar que o modelo com CUDA pode ser processado em tempo real com uma webcam conectada a um computador, algo que comprova a eficácia e a viabilidade do YOLO ao sistema proposto, haja vista que uma Webcam possui baixa resolução e a detecção em tempo real exige um processamento rápido.

### 10.2.4 Testes

Para a execução dos testes iniciais com o Yolo é necessário compilar o Darknet, a qual é a rede neural utilizada pelo Yolo. Para baixar a rede neural, basta executar os seguintes comandos no terminal [27]:

- `git clone https://github.com/pjreddie/darknet`
- `cd Darknet`
- `make`
- `wget https://pjreddie.com/media/files/yolov3.weights`
- `./darknet detect cfg/yolov3.cfg <path>/yolov3.weights imagem`
  - *Em que o path representa o caminho até o arquivo e a imagem representa a imagem a ser selecionada para o processamento.*

Feito isso, é possível começar a executar alguns testes com o sistema. Inicialmente, serão feitos alguns testes com uma rede neural já treinada para mostrar o funcionamento da rede neural:

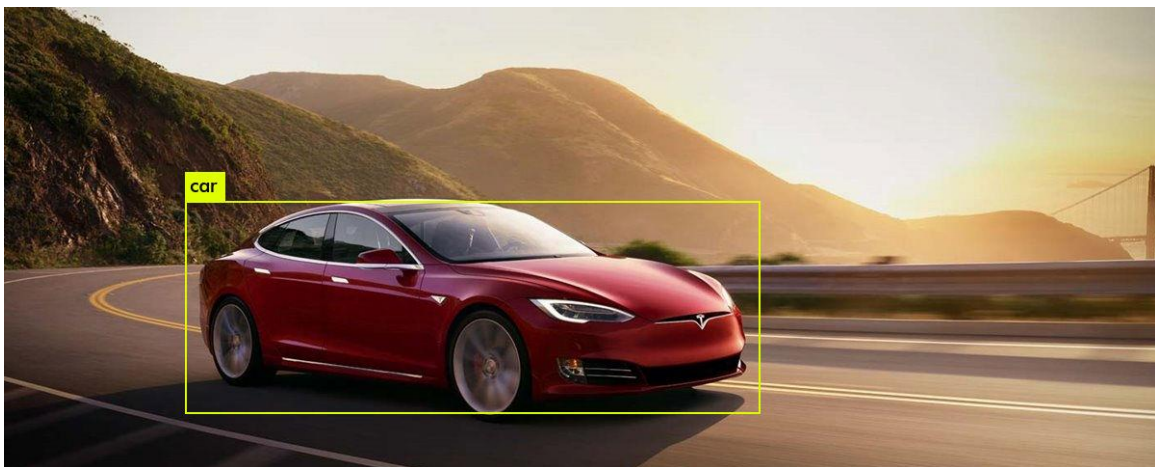


Figura 31: Figura utilizada para o teste de uma rede neural treinada.



Figura 32: Resultado da execução do Yolo para a Figura 38 no Terminal

```
100 yolo  
Loading weights from yolov3.weights...Done!  
data/image1.jpg: Predicted in 22.428059 seconds.  
car: 100%
```

Figura 33: Imagem gerada pelo processamento da Figura 38

Analisando a Figura 32 e Figura 33, percebe-se que o algoritmo identificou de forma correta o objeto da Figura 31 em um curto espaço de tempo (22s).

Em seguida, será feito um teste com a Figura 34, para simular o funcionamento do Yolo em uma situação real de trânsito:



Figura 34: Imagem de teste para representar uma situação real de trânsito.

Como resposta, obtém-se os seguintes resultados:

```
data/image2.jpg: Predicted in 22.737037 seconds.  
truck: 77%  
truck: 71%  
car: 71%  
car: 99%  
car: 99%  
car: 99%  
car: 99%  
car: 98%  
car: 97%  
car: 97%  
car: 97%  
car: 96%  
car: 96%  
car: 96%  
car: 96%  
car: 95%  
car: 95%  
car: 94%  
car: 94%  
car: 93%  
car: 92%  
car: 92%  
car: 91%  
car: 91%  
car: 90%  
car: 88%  
car: 86%  
car: 85%  
car: 84%  
car: 83%  
car: 83%  
car: 83%  
car: 82%  
car: 82%  
car: 82%  
car: 79%  
car: 78%  
car: 78%  
car: 78%  
car: 76%  
car: 73%  
car: 73%  
car: 69%  
car: 69%  
car: 68%  
car: 63%  
car: 61%  
car: 60%  
car: 59%  
car: 59%  
car: 52%  
car: 51%  
person: 93%  
person: 74%
```

Figura 35: Resultado da execução do sistema para a Figura 33



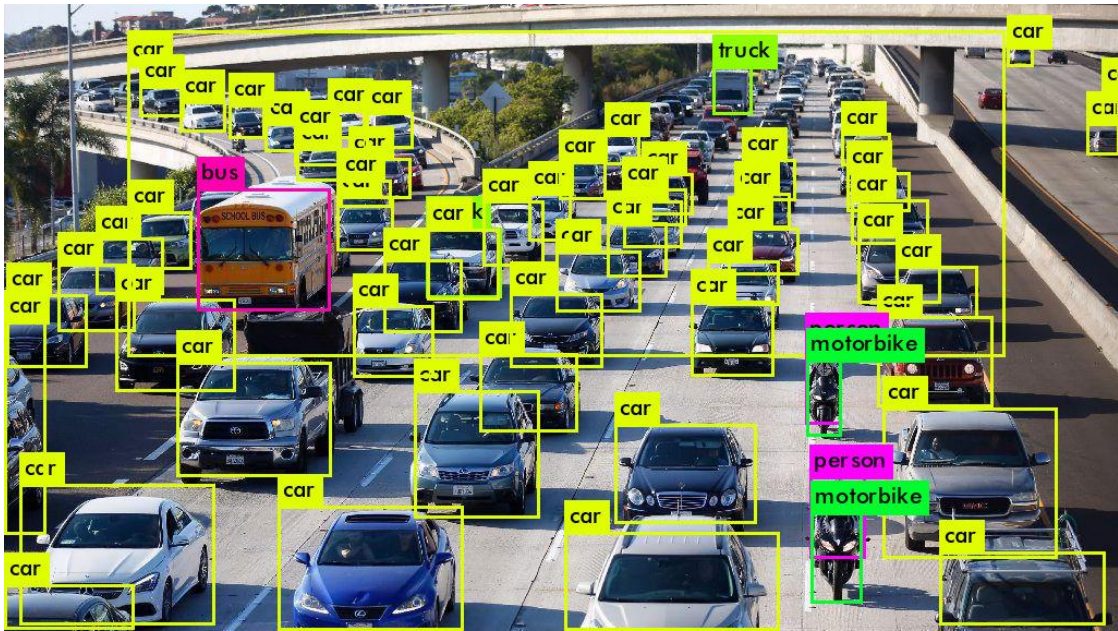


Figura 36: Imagem de saída do algoritmo para a Figura 33

A partir da Figura 35 e da Figura 36, percebe-se que os objetos foram detectados com uma precisão menor do que o primeiro teste, em função da intersecção entre as áreas dos objetos. No entanto, como pode ser verificado, os objetos são identificados de forma correta.

Além disso, é importante ressaltar que o tempo de execução foi de 22s, ou seja, o tempo de execução foi o mesmo para a Figura 31, evidenciando que a predição é feita de maneira extremamente eficiente para 1 objeto ou múltiplos objetos.

Para otimizar ainda mais a predição seria necessário compilar o Yolo juntamente com a API CUDA. No entanto, tal processo exige um processador(GPU) da NVIDIA. De acordo com a literatura [20], seria possível obter resultados até 500 vezes mais rápidos e eficientes com o CUDA, algo necessário para o processamento em tempo real.

É inegável, pois, que utilizando o software do Yolo é possível obter bons resultados de classificação e otimização do fluxo de veículos em semáforos. Isso se deve a boa precisão da tecnologia e que seu processamento ser rápido e eficiente, haja vista que ele foi projetado para identificar objetos em tempo-real e até com câmeras de baixa resolução.

Para adicionar o sistema do Yolo ao sistema estudado será necessário desenvolver uma API de Comunicação para enviar as imagens capturadas pela câmera até a camada do controlador em que será feito o processamento e classificação das imagens, tal qual especificado na modelagem do sistema.

## 11. Conclusão

O presente estudo se pautou em discutir e analisar maneiras de otimizar o fluxo de veículos em semáforos na primeira etapa do trabalho. Posteriormente, desenvolveu-se uma proposta para melhorar o trânsito em sinais de trânsito utilizando conceitos de visão computacional.

Para isso, foi necessário validar o problema com entrevistas e estudo de campo, além realizar uma modelagem do sistema necessário para extrair os requisitos e facilitar a construção de um método para automatizar o fluxo de veículos.

A primeira abordagem consistia em identificar o fundo de referência da imagem para diferenciar o fundo dos objetos e segmentar os objetos. Feito isso, bastava comparar a área do objeto com as medidas padrões de veículos definida na Tabela 7 e Tabela 8.

Embora tal procedimento, tenha apresentado bons resultados para imagens com poucos objetos, o algoritmo apresentou problemas para identificar a imagem de fundo em vias congestionadas e em alguns casos os veículos não foram identificados de modo preciso, haja vista que existiam casos de oclusão e, sem a verificação se um veículo se encontrava em movimento ou não, alguns objetos estacionários passaram a ser considerados objetos válidos de análise.

Já a segunda análise de método se baseou em implantar o software Yolo, de código aberto, ao sistema, uma vez que ele consiste em um programa para a detecção de objetos em tempo real utilizando conceitos de aprendizado de máquina.

Por utilizar apenas uma rede neural e por analisar a imagem como um todo, o projeto apresenta resultados superiores ao de outros de detectores, além de que seria possível utilizá-lo juntamente com CUDA, o que aumentaria ainda mais a eficiência do processo.

É inegável, pois, que o estudo apresentou resultados satisfatórios para uma análise geral do problema do congestionamento em conformidade com o planejamento, uma vez que foram feitas pesquisas para entender e validar o problema, além de simulações, definição de um sistema e proposta de algoritmo para realizar uma contagem volumétrica automatizada.

Também é importante ressaltar que o projeto apresentou algumas limitações, como a dificuldade de realizar testes em tempo real dos algoritmos em função da impossibilidade de verificar o funcionamento da ideia em semáforos.

Assim o sistema proposto pode vir a ser mais uma solução para melhorar o congestionamento em grandes metrópoles, dado que ele apresenta baixos custos de implantação e os algoritmos apresentam grande eficiência computacional. Futuramente, pode-se apontar que o aperfeiçoamento dos algoritmos, realização de testes em tempo real em ambientes dinâmicos e a construção de um

controlador semafórico conforme especificado no Capítulo 5 devem ser os próximos passos do trabalho.



## Referências

[1] Página Web: Brasil perde R\$ 156,2 bilhões do PIB com a morosidade do trânsito em São Paulo. Disponível em:

<[http://agencia.fapesp.br/brasil\\_perde\\_r\\_1562\\_bilhoes\\_do\\_pib\\_com\\_a\\_morosidade\\_do\\_transito\\_em\\_sao\\_paulo/21984/](http://agencia.fapesp.br/brasil_perde_r_1562_bilhoes_do_pib_com_a_morosidade_do_transito_em_sao_paulo/21984/)> Acesso em 10/06/2017

[2] Página Web: Frota de Veículos no Brasil. Disponível em:

< <https://ibpt.com.br/noticia/2640/REAL-FROTA-CIRCULANTE-NO-BRASIL-E-DE-65-8-MILHOES-DE-VEICULOS-INDICA-ESTUDO>> Acesso em 10/06/2017

[3] GERSHENSON, Carlos. **Design and Control of Self Organising Systems**. Cidade do México. CopIt ArXives, 2007.

[4] Página Web: Aumento da Frota de Veículos. Disponível em: < <https://g1.globo.com/carros/noticia/frota-brasileira-de-veiculos-cresce-12-em-2017-diz-sindipecas.ghml> > Acesso em 10/08/2018

[5] Página Web: Sistema Ear-It 1. Disponível em: < <http://www.euronews.com/2014/10/13/in-the-smart-city-of-santander-the-walls-have-ears> > Acesso em 15/08/2018

[6] Página Web: Sistema Ear-It 2. Disponível em: < <https://www.citylab.com/transportation/2014/10/a-new-approach-to-traffic-control-put-ears-to-the-ground/382177/> > Acesso em 15/08/2018

[7] Página Web: Sistema Surtrac. Disponível em: < <http://spectrum.ieee.org/cars-that-think/robotics/artificial-intelligence/pittsburgh-smart-traffic-signals-will-make-driving-less-boring> > Acesso em 19/10/2018

[8] Página Web: Sistema Surtrac. Disponível em: <<http://liberal.com.br/cidades/americana/americana-tera-17-cameras-em-sistema-integrado-com-cidades-da-regiao-595160/>> Acesso em 19/10/2018

- [9] Página Web: Case da Miovision. Disponível em: <  
<http://ca.reuters.com/article/technologyNews/idCAKBN0LE2V720150210> > Acesso em 19/10/2018
- [10] Página Web: Valor de Mercado da Miovision. Disponível em:  
 <<https://pitchbook.com/profiles/miovision-technologies-profile-investors-funding-valuation-and-analysis>> Acesso em 10/10/2018
- [11] Página Web: História do Semáforo. Disponível em: <  
<https://pt.wikipedia.org/wiki/Sem%C3%A1foro> > Acesso em 02/10/2018
- [12] Página Web: Sumo Traffic. Disponível em: <  
[http://sumo.dlr.de/wiki/Simulation\\_of\\_Urban\\_MObility\\_-\\_Wiki](http://sumo.dlr.de/wiki/Simulation_of_Urban_MObility_-_Wiki) > Acesso em 05/09/2018
- [13] E RICHARD E. WOODS, R. C. G. **Processamento de Imagens Digitais**, 1a Edição. Edgard Blucher, 1999.
- [14] Patel, Surabhi; Goswami, Mausumi. In: Surabhi. **Comparative Analysis of Histogram Equalization Techniques**. Contemporary Computing and Informatics (IC3I), 2014 International Conference on: IEEE, 2014
- [15] YANG, JUN; WANG, Y. S. A. L. Z. Z. B. X. J. **Feature fusion for vehicle detection and tracking with low angle cameras**. Applications of Computer Vision (WACV), 2011 IEEE, p. 382–388, 2011.
- [16] Página Web: Dimensões de Veículos. Disponível em: <  
<http://www.ebanataw.com.br/recebimento/garagem02.htm> > Acesso em 31/10/2018
- [17] Página Web: Dimensões de Caminhões. Disponível em: <  
<http://www.guiadotrc.com.br/noticias/not.asp?ID=21457> > Acesso em 31/10/2018
- [18] GUPTE, S; MASOUD, O; MARTIN, R.F.K; PAPANIKOLOPOULOS, N.P. **Detection and Classification of Vehicles**. IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 3, NO. 1, MARCH 2002.

[19] REDMON, J; DIVVALA, S; GIRSHICK, R; FARHADI, A. **You Only Look Once: Unified, Real-Time Object Detection**. University of Washington , Allen Institute for AI , Facebook AI Research, MAY 2016.

[20] Página Web: Exemplos de aplicações do YOLO. Disponível em: < <https://github.com/pjreddie/darknet/wiki/YOLO:-Real-Time-Object-Detection> >. Acesso em 15/09/2018.

[21] Página Web: Laço Indutivo de Cobre. Disponível em: < <https://www.almitec.com.br/laco-indutivo> > Acesso em 05/11/2018

[22] Página Web: Sensor Ear-It 3. Disponível em: < <https://ec.europa.eu/digital-single-market/en/news/ear-it-using-sound-picture-world-new-way> > Acesso em 05/11/2018

[23] Página Web: Sensor Microondas. Disponível em: < <http://blogmasterwalkershop.com.br/arduino/arduino-utilizando-o-sensor-microondas-rcwl-0516/> > Acesso em 06/11/2018

[24] Página Web: Polo Petroquímico de Paulínia. Disponível em: < <http://blogmasterwalkershop.com.br/arduino/arduino-utilizando-o-sensor-microondas-rcwl-0516/> > Acesso em 04/11/2018

[25] Página Web: Surtrac. Disponível em: < <https://www.rapidflowtech.com/> > Acesso em 04/11/2018

[26] Página Web: Dados da Cidade de Americana. Disponível em: < <https://www.rapidflowtech.com/> > Acesso em 04/11/2018

[27] Página Web: Instalação da Darknet. Disponível em: < <https://pjreddie.com/darknet/install/> > Acesso em 15/09/2018

[28] Página Web: CUDA. Disponível em: < <https://developer.nvidia.com/cuda-toolkit> > Acesso em 25/09/2018

[29] Página Web: YOLO v3. Disponível em: < <https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b> > Acesso em 15/09/2018

[30] Página Web: OpenCV. Disponível em: < <https://opencv.org/> > Acesso em 04/05/2018

[31] Página Web: Sistema para metrificar tempos dos semáforos. Disponível em: < <https://super.abril.com.br/mundo-estranho/como-os-tecnicos-calculam-o-tempo-que-o-semaforo-deve-ficar-verde/> > Acesso em 06/11/2018

[32] AZEREDO, Luiz Ernesto. **Seis Fatores Essenciais para o Sincronismo entre Semáforos**. EMDEC, Campinas - SP, Maio de 2014.

[33] Página Web: Funcionamento dos Semáforos. Disponível em: < <https://extremamentecurioso.blogspot.com/2013/04/como-os-semaforos-funcionam.html> > Acesso em 12/11/2018