



UNIVERSIDADE FEDERAL DO AMAZONAS  
FACULDADE DE TECNOLOGIA  
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

DETECÇÃO DE PLACAS VEICULARES EM  
AMBIENTES AÉREOS BASEADO EM MÉTODOS DE  
APRENDIZAGEM PROFUNDA

José Elislande Breno de Souza Linhares

Manaus - Amazonas

Julho de 2022

José Elislande Breno de Souza Linhares

DETECÇÃO DE PLACAS VEICULARES EM  
AMBIENTES AÉREOS BASEADO EM MÉTODOS DE  
APRENDIZAGEM PROFUNDA

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, PPGEE, da Universidade Federal do Amazonas, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Prof. D.Sc. Waldir Sabino da Silva Júnior

## Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

L755d Linhares, José Elislande Breno de Souza  
Detecção de placas veiculares em ambientes aéreos baseado em métodos de aprendizagem profunda / José Elislande Breno de Souza Linhares . 2022  
58 f.: il. color; 31 cm.

Orientador: Waldir Sabino da Silva Júnior  
Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal do Amazonas.

1. Imagens aéreas. 2. Detecção de placas veiculares. 3. Super-resolução. 4. GAN - Generative Adversarial Network. 5. Equalização de histograma. I. Silva Júnior, Waldir Sabino da. II. Universidade Federal do Amazonas III. Título

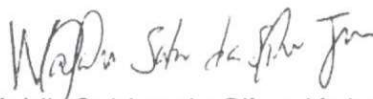
JOSÉ ELISLANDE BRENO DE SOUZA LINHARES

**UMA DETECÇÃO DE PLACAS VEICULARES EM AMBIENTES  
AÉREOS BASEADO EM MÉTODOS DE APRENDIZAGEM  
PROFUNDA**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Amazonas, como requisito parcial para obtenção do título de Mestre em Engenharia Elétrica na área de concentração Controle e Automação de Sistemas.

Aprovado em 01 de julho de 2022.

**BANCA EXAMINADORA**



Prof. Dr. Waldir Sabino da Silva Júnior, Presidente  
Universidade Federal do Amazonas



Prof. Dr. Carlos Augusto de Moraes Cruz, Membro  
Universidade Federal do Amazonas



Prof. Dr. Gabriel Matos Araujo, Membro  
Centro Federal de Educação Tecnológica/RJ

*À minha família.*

# Agradecimentos

- Agradeço a Deus pelo dom da vida e por todas as conquistas alcançadas.
- Agradeço à minha esposa, Hoffeman Jussara Hodrigues Colares, por todo amor, suporte, cuidado e carinho.
- Agradeço à minha mãe, Maria Alice Bizerril de Souza Linhares, e ao meu pai, Mariano Coêlho Linhares, que me educaram e me deram as condições necessárias para continuar estudando.
- Agradeço aos amigos e colegas da UFAM, Erickson Alves, Arllem Farias, Thiago Cavalcante, Weider Serruia, Robson Cruz, Helton Nogueira e Rosmael Miranda, pela amizade durante os anos de graduação, mestrado e da vida.
- Agradeço ao meu orientador, Prof. Dr. Waldir Sabino, pelo direcionamento, suporte, paciência e conselhos durante o desenvolvimento deste trabalho.
- Agradeço à Universidade Federal do Amazonas (UFAM) por permitir a realização desta dissertação.

Resumo da Dissertação apresentada à UFAM como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica

## DETECÇÃO DE PLACAS VEICULARES EM AMBIENTES AÉREOS BASEADO EM MÉTODOS DE APRENDIZAGEM PROFUNDA

José Elislande Breno de Souza Linhares

Orientador: Waldir Sabino da Silva Júnior

Programa: Pós-Graduação em Engenharia Elétrica

Nesta dissertação, considerando-se as dificuldades de detecção de pequenos objetos devido à baixa qualidade visual e resolução espacial das imagens, bem como a pouca informação de contexto, propõe-se uma metodologia para detecção de placas veiculares em ambientes aéreos, onde o objeto de interesse apresenta baixa resolução em pixel em relação à imagem de entrada. A metodologia proposta é composta por três sistemas distintos. Duas utilizam técnicas de melhoria de qualidade da imagem e uma não as utiliza. A principal contribuição desta dissertação é a organização de uma base de dados composta por imagens aéreas para avaliar o desempenho da metodologia proposta. Como segunda contribuição, tem-se a utilização de métodos de melhoria de qualidade visual baseado em aprendizagem e processamento digital de imagens para detecção de placas veiculares em ambientes aéreos. Como resultados obtidos, verifica-se que os sistemas propostos apresentam resultados similares, em termos de acurácia global. Ao analisar os resultados por grupo de imagens, observa-se que o sistema proposto com equalização de histograma apresenta os melhores resultados com acurácia de até 85,67% em condições adversas ensolaradas e 99,33% em condições adversas sombreadas.

Palavras-chave: Imagens aéreas; Detecção de placas veiculares; Super-resolução; GAN; Equalização de histograma.

Abstract of Dissertation presented to UFAM as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering

DETECTION OF LICENSE PLATES IN AERIAL ENVIRONMENTS BASED  
ON DEEP LEARNING METHODS

José Elislande Breno de Souza Linhares

Advisor: Waldir Sabino da Silva Júnior

Department: Postgraduate in Electrical Engineering

In this dissertation, considering the difficulties of detecting small objects due to the low visual quality and spatial resolution of the images, as well as the little context information, we propose a methodology for detecting license plates in aerial environments, where the object of interest has low pixel resolution compared to the input image. The proposed methodology is composed of three distinct systems. Two use image quality improvement techniques and one does not. The main contribution of this dissertation is the organization of a database composed of aerial images to evaluate the performance of the proposed methodology. As a second contribution, there is the use of visual quality improvement methods based on learning and digital image processing to detect license plates in aerial environments. As obtained results, it appears that the proposed systems present similar results, in terms of global accuracy. When analyzing the results by group of images, it is observed that the proposed system with histogram equalization presents the best results with an accuracy of up to 85.67% in adverse sunny conditions and 99.33% in adverse shaded conditions.

Keywords: Aerial imagery; License plate detection; Super resolution; GAN; Histogram equalization.



# Sumário

<b>Abreviações</b>	<b>xii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Objetivos . . . . .	3
1.2 Organização da Dissertação . . . . .	4
<b>2 Fundamentos Teóricos</b>	<b>5</b>
2.1 Processamento Digital de Imagens . . . . .	5
2.2 Redes Neurais Convolucionais e Adversárias Generativas . . . . .	7
2.3 Detecção de Objetos . . . . .	10
2.4 Super-Resolução por Interpolação e Aprendizagem . . . . .	11
2.4.1 Super-Resolução baseada em Interpolação . . . . .	11
2.4.2 Super-Resolução baseada em Aprendizagem . . . . .	12
2.5 Trabalhos Relacionados . . . . .	13
<b>3 Metodologia Proposta</b>	<b>18</b>
3.1 Introdução . . . . .	18
3.2 Sistema de detecção de placa de licenciamento veicular sem super-resolução . . . . .	19
3.3 Sistema de detecção de placa de licenciamento veicular com super-resolução . . . . .	23
3.4 Sistema de detecção de placa de licenciamento veicular com equalização de histograma . . . . .	26
3.5 Parâmetros dos sistemas propostos . . . . .	31
<b>4 Base de Dados</b>	<b>32</b>

4.1	Base de dados dos modelos pré-treinados utilizados . . . . .	32
4.2	Base de dados de testes . . . . .	33
<b>5</b>	<b>Experimentos e Resultados</b>	<b>38</b>
5.1	Introdução . . . . .	38
5.2	Setup . . . . .	38
5.3	Métricas de desempenho . . . . .	39
5.4	Sistema de detecção de placa de licenciamento veicular sem super- resolução . . . . .	40
5.5	Sistema de detecção de placa de licenciamento veicular com super- resolução . . . . .	42
5.6	Sistema de detecção de placa de licenciamento veicular com equaliza- ção de histograma . . . . .	45
5.7	Comparação dos resultados dos sistemas propostos . . . . .	49
<b>6</b>	<b>Conclusão</b>	<b>50</b>
6.1	Proposta para Trabalhos Futuros . . . . .	51
	<b>Referências Bibliográficas</b>	<b>53</b>

# Lista de Figuras

3.1	Diagrama em blocos para o sistema proposto sem super-resolução. . .	20
3.2	Diagrama em blocos para o sistema proposto com super-resolução. . .	24
3.3	Diagrama em blocos para o sistema proposto com CLAHE. . . . .	27
3.4	Exemplos de imagens originais e equalizadas, com seus histogramas. .	30
4.1	Exemplos de rotulação de VLP com regiões não pertencentes ao objeto.	35
4.2	Exemplos de amostras de frames da base de dados de testes. . . . .	37

# Lista de Tabelas

1.1	Divisão da base de dados em grupo de imagens. . . . .	3
2.1	Tabela comparativa entre a método proposto e trabalhos relacionados.	17
3.1	Parâmetros dos três sistemas propostos para detecção de VLP. . . . .	31
4.1	Termos utilizados na pesquisa de vídeos no YouTube. . . . .	34
4.2	Descrição da formatação dos vídeos e do cenário capturado. . . . .	36
5.1	Resultados do sistema para detecção de VLP sem super-resolução. . .	41
5.2	Resultados do sistema sem super-resolução por grupo de imagens. . .	42
5.3	Resultados do sistema para detecção de VLP com super-resolução. . .	44
5.4	Resultados do sistema com super-resolução por grupo de imagens. . .	45
5.5	Resultados do sistema para detecção de VLP com CLAHE. . . . .	48
5.6	Resultados do sistema proposto com CLAHE por grupo de imagens .	48
5.7	Resultados dos três sistemas propostos utilizando o modelo YOLOv2.	49
5.8	Resultados dos três sistemas propostos utilizando o modelo YOLOv3.	49

# Abreviações

**YOLO** - *You Only Look Once*

**SRGAN** - *Generative Adversarial Network for Super-Resolution*

**HR** - *High-Resolution*

**LR** - *Low-Resolution*

**SR** - *Super-Resolution*

**G** - *Generator*

**D** - *Discriminator*

**GAN** - *Generative Adversarial Network*

**CNN** - *Convolutional Neural Network*

**CCTV** - *Closed Circuit TV*

**VANT** - *Veículo Aéreo Não Tripulado*

**WPOD-NET** - *Warped Planar Object Detection Network*

**CLAHE** - *Contrast Limited Adaptive Histogram Equalization*

**RGB** - *Red Green Blue*

**ReLU** - *Rectified Linear Unit*

**GT** - *Ground-Truth*

**MSE** - *Mean Squared Error*

**ALPR** - *Automatic License Plate Recognition*

**OCR** - *Optical Character Recognition*

**YOLOv2** - *You Only Look Once Version 2*

**PASCAL-VOC** - *PASCAL-Visual Object Classes*

**SSD** - *Single Shot Multibox Detector*

**STN** - *Spacial Transform Network*

**YOLOv3** - *You Only Look Once Version 3*

**Faster R-CNN** - *Faster Region-Convolutional Neural Network*

**COWC** - *Cars Overhead With Context*  
**OGST** - *Oil and Gas Storage Tank*  
**IoU** - *Intersection of Union*  
**AOLP** - *Application-Oriented License Plate*  
**SRCNN** - *Super-Resolution Convolutional Neural Network*  
**HSV** - *Hue Saturation Value*  
**DIV2K** - *DIVerse 2K*  
**CLIP** - *Contrastive Language-Image Pre-Training*  
**VLP** - *Vehicle License Plate*  
**TP** - *True Positive*  
**TN** - *True Negative*  
**FP** - *False Positive*  
**FN** - *False Negative*  
**mAP** - *mean Average Precision*  
**COCO** - *Common Objects in Context*

# Capítulo 1

## Introdução

Imagens aéreas são tipos de imagens digitais adquiridas por aeronaves, como VANT, por exemplo. A aquisição destas imagens ocorre, na maioria das vezes, em diferentes níveis de altitude, com um amplo campo de visão, com diferentes *point-of-views* e com objetos apresentando escala uniforme [1]. Em relação à última característica, destaca-se que objetos apresentam escala uniforme, quando detectados por uma câmera em um nível de altitude fixa. A detecção de objetos em imagens aéreas adquiridas por VANT tem diversas aplicações como, por exemplo, em segurança e vigilância [2] e em busca e resgate [3]. Nas tarefas de detecção de objetos, que consistem na correta localização e determinação de um objeto de interesse numa cena, surgem desafios quando a aquisição dessas imagens ocorre em ambientes aéreos. Dentre os desafios existentes, tem-se a relacionada ao pequeno tamanho (i.e., área menor que  $32 \times 32$  pixels, que é igual à  $1024$  pixels) [4], e a consequente baixa resolução espacial dos objetos, devido à aquisição de imagens em elevadas altitudes. Uma possível abordagem para tal desafio é utilizar redes neurais para identificar e classificar o objeto de interesse por meio de arquiteturas, como a *you only look once* (YOLO) [5]. A detecção de objetos de tamanho médio (i.e., área entre  $32 \times 32$  e  $96 \times 96$  pixels) e grande (i.e., área maior que  $96 \times 96$  pixels, que é igual à  $9216$  pixels) [4] tem sido utilizada com frequência em algumas pesquisas, no entanto, apenas alguns estudos abordam a detecção de pequenos objetos, devido à dificuldade de detectá-los. A razão é decorrente de sua baixa resolução espacial em pixel e pouca informação de contexto [6, 7]. A super-resolução por meio de redes adversárias generativas (SRGAN) [8] é um método que consiste em estimar

uma imagem *high resolution* (HR) a partir de uma imagem *low resolution* (LR), denominando-a de imagem *super resolution* (SR). O método por SRGAN tem o intuito de preservar as características originais da cena, devido à execução da operação de *pixel shuffle* que super-resolve a imagem de entrada a nível de subpixel.

A metodologia proposta se diferencia dos trabalhos presentes no estado da arte como, por exemplo, o apresentado por Kim *et al.* [9], por Lee *et al.* [10], por Silva e Jung [11], por Rabbi *et al.* [12] e por Khazaei *et al.* [13]. No trabalho proposto por Kim *et al.*, o objeto de interesse são os modelos dos veículos e as imagens de baixa resolução processadas são provenientes de câmeras de circuito fechado de televisão (CCTV). No trabalho proposto por Silva e Jung, o objeto de interesse são placas veiculares detectadas em ambientes não controlados, assim como no trabalho apresentado por Khazaei *et al.* Nesta dissertação, em contrapartida, foca-se em placas veiculares como objetos de interesse e as imagens de baixa resolução são extraídas de vídeos, que contenham imagens aéreas, selecionados da plataforma YouTube [12]. O trabalho de Lee *et al.*, de modo diferente da proposta desta dissertação, utiliza imagens de baixa resolução de vídeos de vigilância de tráfego. No trabalho proposto por Rabbi *et al.*, o objeto de interesse são veículos e as imagens são adquiridas de satélites. Considerando os desafios existentes em processar imagens adquiridas em ambientes aéreos, a metodologia proposta nesta dissertação provê uma detecção mais robusta que a empregada pelos pesquisadores.

Nesta dissertação, considerando-se as dificuldades de detecção de pequenos objetos devido à baixa qualidade visual e resolução espacial das imagens, bem como a pouca informação de contexto, propõe-se uma metodologia para detecção de placas veiculares em ambientes aéreos composta por três sistemas distintos. Duas utilizam técnicas de melhoria de qualidade da imagem e uma não as utiliza. Para avaliar o desempenho da metodologia proposta, divide-se a base de dados em sete grupos com características comuns, sendo cada grupo representado por uma condição climática adversa predominante. Como resultados obtidos, verifica-se que existem grupos de imagens que funcionam bem (por exemplo, os grupos  $G1$  até  $G3$ ) quando processados pelos sistemas propostos, enquanto existem outros que falham (por exemplo, os grupos  $G6$  até  $G7$ ). Na Tabela 1.1, apresentam-se os grupos de imagens utilizados.

Considerando a dificuldade de acesso a bases de dados específicas para de-



Tabela 1.1: Divisão da base de dados em grupo de imagens.

Grupo	Condição
G1	Ensolarado
G2	Ensolarado
G3	Sombreado
G4	Nublado
G5	Nublado
G6	Noturno
G7	Noturno

tecção de placas veiculares em ambientes aéreos, a principal contribuição desta dissertação é a organização de uma base de dados composta por imagens aéreas para avaliar o desempenho da metodologia proposta. Esta base de dados contém 1600 imagens coloridas extraídas de sete diferentes vídeos disponibilizados na plataforma YouTube, sendo todas com *background* complexo, isto é, com veículos parados ou em movimento, apresentando diferentes condições climáticas e variações na iluminação.

Como contribuição secundária, tem-se a avaliação de um sistema de detecção de placas veiculares em imagens melhoradas. A razão é decorrente da dificuldade de detectar placas veiculares com baixa qualidade visual e resolução espacial em pixel. Ou seja, a qualidade destes objetos está comprometida, o que dificulta a sua representação, tornando-se difícil o processo de detecção. Para isso, duas abordagens são utilizadas: SR com GAN e CLAHE.

## 1.1 Objetivos

O objetivo principal deste trabalho é propor uma metodologia para detecção de placas veiculares em imagens aéreas, onde o objeto de interesse apresenta baixa resolução em pixel, a partir da utilização de métodos baseados em aprendizagem profunda.

### Objetivos Específicos

Os objetivos específicos deste trabalho são os seguintes:

- Conceber uma metodologia para detecção de placas veiculares, que é composta por três sistemas, sendo um sistema de referência, um sistema que utiliza super-resolução e um sistema que utiliza super-resolução com pré-processamento.

- Organizar (seleção, edição, agrupamento) uma base de dados para avaliar o desempenho da metodologia proposta, com base em vídeos que contenham imagens aéreas, em diferentes resoluções em pixel.
- Definir um modelo de pesquisa que sirva como alicerce para trabalhos futuros.

## 1.2 Organização da Dissertação

Esta dissertação está organizada, conforme a seguir. No Capítulo 2, apresentam-se os fundamentos teóricos da dissertação. No Capítulo 3, é descrita a metodologia proposta. No Capítulo 4, apresenta-se a base de dados utilizada na dissertação. No Capítulo 5, são disponibilizados os experimentos e os resultados obtidos para cada sistema proposto. No Capítulo 6, apresentam-se a conclusão da pesquisa e as propostas para trabalhos futuros.

# Capítulo 2

## Fundamentos Teóricos

### 2.1 Processamento Digital de Imagens

Uma imagem digital 2-D é representada por uma matriz bidimensional  $\mathbf{I}_{M \times N}$ , onde as dimensões  $M$  e  $N$  determinam, respectivamente, a quantidade de linhas e colunas. A localização do par  $(m, n)$  fixa cada elemento ou pixel da imagem [14]. O valor de  $\mathbf{I}(m, n)$  representa a intensidade luminosa no pixel localizado em  $(m, n)$ .

Cada imagem possui uma medida que determina o nível de representação ou tamanho: a resolução [14]. Esta medida caracteriza-se, por exemplo, pela resolução espacial, que é definida pelo número de pixels utilizados para compor a cena adquirida por uma fonte de imagem. Seja  $N$  o número de colunas e  $M$  o de linhas de uma imagem, a resolução de uma imagem é dada pela quantidade  $N \times M$  (por exemplo,  $1280 \times 720$  e  $1920 \times 1080$ ).

Uma imagem digital normalmente é dividida em dois tipos: imagens em escala de cinza e imagens coloridas. A depender da necessidade da aplicação, utilizam-se em etapas de pré-processamento para conversão em um espaço de cores mais adequado. As imagens de intensidade ou escala de cinza constituem matrizes 2-D com um único valor inteiro não-negativo, que depende da resolução de bits, associado a cada pixel no plano matricial. Diferentemente, as imagens RGB ou coloridas são matrizes 2-D que relacionam três valores inteiros não-negativos para cada pixel, sendo cada valor correspondente ao componente vermelho (**R**), verde (**G**) e azul (**B**). Ou seja, configuram-se como três planos matriciais 2-D com dimensões  $N \times M \times 3$  [14].

As imagens digitais são classificadas, por exemplo, quanto ao ambiente de

aquisição, em imagens terrestres e aéreas. As imagens terrestres, também denominadas de imagens ao nível do solo, caracterizam-se, geralmente, por exibirem *views* de alta resolução, com proximidade em relação à fonte de aquisição (i.e., câmera) e com perspectivas normalmente horizontais [15, 16]. As imagens aéreas adquiridas por VANTs retratam cenas com ampla área de cobertura, com objetos orientados arbitrariamente (i.e., *views* frontais, laterais ou de pássaro) e com ambiente complexo [1, 15, 16]. Adicionalmente, a depender do nível de altitude na qual a imagem aérea for adquirida, pode-se ter imagens de baixa (10 ~ 30 m), média (30 ~ 70 m) e alta altitude (> 70 m) [1].

Outro conceito importante em processamento digital de imagens é o *histograma*. O histograma de uma imagem consiste em um gráfico que apresenta a frequência de ocorrência de cada intensidade de nível de pixel presente na imagem [14]. Um histograma é definido como uma função discreta conforme Equação (2.1) [17].

$$h(r_k) = n_k, \quad (2.1)$$

onde  $r_k$  é o  $k$ -ésimo valor de intensidade e  $n_k$  é o número de pixels da imagem com intensidade  $r_k$ .

Ocasionalmente, torna-se necessário tratar a imagem a ser processada, por estar em baixa qualidade visual. Para isso, utilizam-se técnicas de realce de contraste. A equalização de histograma é um método de realce de imagem que produz uma distribuição uniforme das intensidades de nível de pixel para melhoria do contraste da imagem [17], [14], [18]. Por considerar toda a imagem de entrada, o contraste tende a ser máximo. A equalização de histograma é definida conforme Equação (2.2).

$$y_k = \sum_{j=0}^k p_x(j) = \frac{1}{T} \sum_{j=0}^k n_j, \quad (2.2)$$

onde  $k = \{0, 1, 2, \dots, L - 1\}$ , a frequência acumulada do  $k$ -ésimo nível é dado por  $n_k$  e o número total de pixels na imagem é  $T$ .

Considerando que a equalização de histograma clássica maximiza o efeito de contraste por toda a imagem, existem métodos como, o CLAHE (do inglês, *contrast*

*limited adaptive histogram equalization*) [18], por exemplo, que ajusta o contraste em uma parte específica da imagem. O método CLAHE consiste em um processo de equalização de histograma por regiões para limitar o contraste a um nível desejado, chamado de limite de recorte. Basicamente, o processo de equalização de histograma CLAHE é composto por três etapas: i) histograma de imagem, em que se calcula o histograma de cada região  $P \times P$  obtida na imagem, contabilizando a frequência de ocorrência de cada intensidade de nível de pixel presente na região; ii) equalização de histograma, em que se utiliza a função de normalização (i.e., dividindo-se cada frequência de intensidade de nível de pixel pelo número total de pixels na imagem) e a função de distribuição acumulada (i.e., somando-se cada fração de pixel com intensidade menor ou igual a um limiar  $L$ ) e; iii) limite de recorte, em que se aplica a função de limite de recorte pra ajustar o contraste de cada região a um nível desejado.

## 2.2 Redes Neurais Convolucionais e Adversárias Generativas

As redes neurais convolucionais (CNNs) são arquiteturas de aprendizagem profunda projetadas para tratar dados estruturados em formato de *grid*, tendo-se dedicado muita atenção nos últimos anos em dados de imagens digitais [19]. Para caracterizar uma CNN, torna-se necessário definir alguns conceitos de redes neurais. Apresentam-se estes conceitos, conforme a seguir:

- **Camadas:** Basicamente, uma CNN é constituída por camadas, dimensionada em altura, largura e profundidade. O conceito de profundidade de uma camada  $d_q$  está relacionada ao número de *feature maps* da camada, enquanto que o conceito de rede neural propriamente dita está relacionada ao número de camadas hierárquicas desta rede. As quatro camadas básicas de uma rede convolucional são: convolucional, ReLU, de *pooling* e *fully connected* [19]. A nomenclatura de cada camada está normalmente associada à operação efetuada ao longo da rede, sendo duas operações as mais usuais: convolução e *pooling*.
- **Feature maps:** As *feature maps* são as camadas de uma CNN. Quando se

processa uma imagem RGB, por exemplo, a *feature map* da camada de entrada consiste em uma matriz tridimensional, devido aos três canais de cores. A *feature map* das camadas ocultas (i.e., das camadas intermediárias) é uma imagem multidimensional, devido às transformações realizadas ao longo da rede [20].

- **Filtros:** Um filtro, também denominado de *kernel* ou máscara, é uma matriz quadrada que contém dimensões  $K \times K$ . É um dos componentes utilizados na operação de convolução [21] para extração de *features*.
- **Tensores:** Os tensores são um conjunto de dados no formato de uma matriz n-dimensional. Considera-se a unidade básica de representação e manipulação de dados por redes neurais [21].
- **Extração de *features*:** Consiste em uma etapa de execução de uma CNN, em que um descritor fornece informações visuais necessárias para representar com robustez a grande quantidade de objetos existentes [20].
- **Operação de convolução:** A operação de convolução, no contexto de uma rede neural, é utilizada para a extração de *features* e consiste na soma de produtos dos tensores  $w$  e  $h$ , sendo o tensor  $w$  o filtro da respectiva camada, e o tensor  $h$  a imagem original, quando operada na camada de entrada, ou uma *feature map* quando operada em camadas ocultas, resultando em um tensor, também conhecido como *feature map* da camada subsequente. A operação de convolução é definida conforme Equação (2.3) [19].

$$h_{ijp}^{(q+1)} = \sum_{r=1}^{F_q} \sum_{s=1}^{F_q} \sum_{k=1}^{d_q} w_{rsk}^{(p,q)} h_{i+r-1, j+s-1, k}^{(q)} \cdot \begin{cases} \forall i \in \{1, \dots, L_q - F_q + 1\} \\ \forall j \in \{1, \dots, B_q - F_q + 1\} \\ \forall p \in \{1, \dots, d_{q+1}\} \end{cases} \quad (2.3)$$

Na Equação (2.3), tem-se  $d_q$  filtros por dimensão  $F_q \times F_q$  na camada  $q$ . As variáveis  $w^{(p,q)}$  e  $h^{(q)}$  correspondem, respectivamente, ao filtro  $p$  aplicado na camada  $q$  e a *feature map* da camada  $q$  ou a imagem original, quando se tratar

da camada de entrada. As dimensões da *feature map* resultante dependem da camada imediatamente anterior e do número de filtros aplicados, sendo a altura igual a  $L_{q+1} = L_q - F_q + 1$ , a largura igual a  $B_{q+1} = B_q - F_q + 1$  e a profundidade  $d_{q+1}$  igual ao número de filtros aplicados.

- Operação de *pooling*: A operação de *pooling* consiste em aplicar uma função  $f$  sobre parte de uma *feature map*, com o objetivo de reduzir a dimensionalidade (i.e., diminuir a complexidade computacional do classificador) sem alterar sua profundidade. Quando o valor máximo é retornado nesta operação, a abordagem utilizada é chamada de *pooling* máximo [22].
- *Stride*: Consiste na quantidade de movimentos que um filtro realiza sobre uma *feature map*, tanto para direita, quanto para baixo. Denota-se *stride* por  $t = u$ , em que  $u$  corresponde ao número de passadas. Aplica-se nas operações de convolução e *pooling* [21].
- Camada ReLu: Uma camada ReLu (do inglês, *rectified linear unit*) realiza a função de ativação ReLU nos valores de cada parâmetro em uma *feature map*. Esta função possui saída igual à zero para entradas negativas e saída igual ao próprio valor de entrada para entradas positivas [22].
- Camada *fully connected*: A camada *fully connected* corresponde à última de uma rede neural, sendo utilizada para conectar todos os parâmetros das *feature maps* da rede neural à um estado (i.e., classe) de saída [19].

As redes adversárias generativas são arquiteturas de aprendizagem profunda que operam em conjunto com dois modelos de redes neurais: um modelo generativo e um modelo discriminativo. Um modelo generativo é responsável por gerar, sinteticamente, dados parecidos com um conjunto de dados *ground-truth* (GT). Adicionalmente, um modelo discriminativo é responsável por classificar dados como reais ou falsos, partindo de entradas provenientes de uma base de dados GT e sintéticas [19]. Por se tratar de uma arquitetura com numerosas camadas ocultas, torna-se necessário utilizar estruturas capazes de resolver o problema de saturação da precisão do modelo. Estas estruturas são os blocos residuais que são capazes de realizar conexões de atalho entre camadas a fim de evitar a retropropagação de pesos nulos,

quando não ocorre a ativação para as próximas camadas da rede. Um bloco residual pode ser definido conforme Equação (2.4) [23].

$$y = \mathcal{F}(x, W_i) + x, \quad (2.4)$$

onde  $x$  e  $y$  são os vetores de entrada e saída da camada atual e a função  $\mathcal{F}(x, W_i)$  se refere ao mapeamento residual a ser aprendido. A operação  $\mathcal{F} + x$  representa a conexão de atalho para próxima camada.

Uma abordagem clássica que explica o funcionamento de uma GAN é o exemplo dos falsificadores de dinheiro e a polícia. Analogamente, enxerga-se uma rede generativa  $G$  como um falsificador de notas e a rede discriminativa  $D$  como a polícia fiscalizando o falsificador. Neste relacionamento, as duas redes funcionam de maneira adversária, em que  $G$  busca produzir dados falsos capazes de enganar  $D$ , ao passo que  $D$  busca ser capaz de diferenciar dados reais de sintéticos [19]. Encontra-se o ponto de equilíbrio formulando o problema mini-max. O problema mini-max é definido conforme Equação (2.5) [24].

$$\min_G \max_D (D, G) = \mathbb{E}_{\mathbf{x} \sim p_{data}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))] \quad (2.5)$$

Na Equação (2.5), o objetivo é minimizar a parcela  $\log(1 - D(G(\mathbf{z})))$ , em que  $G$  produz dados sintéticos semelhantes aos dados reais, e maximizar a parcela  $\log D(\mathbf{x})$ , em que  $D$  acerta corretamente a classificação.

## 2.3 Detecção de Objetos

A detecção de objetos é uma tarefa da área de visão computacional cujo objetivo é classificar e localizar o objeto de interesse numa determinada imagem [20]. Entre as abordagens que implementam a tarefa de detecção de objetos, a *you only look once* (YOLO) [5] é um método de aprendizagem profunda que vem trazendo resultados promissores, com alta velocidade de detecção. YOLO é um detector de único estágio, em que os objetos são detectados diretamente na imagem de entrada por meio de caixas delimitadoras, contendo, inclusive, a probabilidade da classe re-



lacionada ao objeto. Por esse motivo, a detecção é muito rápida, além de conciliar uma boa aprendizagem nas representações gerais dos objetos em cena. Este detector utiliza CNNs [25] para prever tanto as múltiplas caixas delimitadoras, quanto as probabilidades de classes, a partir de uma imagem de entrada dividida em uma *grid*  $S \times S$ . No YOLOv2 clássico [26], utilizam-se 24 camadas convolucionais, responsáveis por extrair as *features* da imagem de entrada, e 2 camadas *fully connected*, responsáveis por retornar as probabilidades e as coordenadas de localização do objeto de interesse.

## 2.4 Super-Resolução por Interpolação e Aprendizagem

Super-Resolução (SR) é um processo de aprimoramento de qualidade visual que consiste em reconstruir uma imagem em alta resolução (HR) a partir de uma imagem de baixa resolução (LR) [8]. Dividem-se os algoritmos de SR existentes, basicamente, em duas categorias: métodos clássicos (i.e., por interpolação) e métodos baseados em aprendizagem, sendo esta a categoria que vem recebendo maior interesse nos últimos anos, com abordagens envolvendo, por exemplo, redes adversárias generativas para super-resolução (SRGANs) [8].

### 2.4.1 Super-Resolução baseada em Interpolação

Inperpolação [17] é uma técnica de processamento digital de imagens utilizada para redimensionamento de imagens, seja para aumentar, seja para reduzir suas dimensões espaciais. Empregam-se diferentes métodos para interpolação, quando o objetivo é super-resolver uma imagem de entrada: vizinho mais próximo, bilinear e bicúbica. No entanto, a interpolação búbica é o método que proporciona melhores resultados, quando comparados com os demais. Este método possui a capacidade de preservar os contornos dos objetos, evitando, assim, efeitos de serrilhamento. A operação da interpolação bicúbica é definida conforme Equação (2.6) [17].

$$v(e, f) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} e^i f^j, \quad (2.6)$$

onde  $a_{ij}$  são os coeficientes dos 16 vizinhos mais próximos e  $e$  e  $f$  correspondem às linhas e colunas referentes ao ponto  $(e, f)$ .

## 2.4.2 Super-Resolução baseada em Aprendizagem

SRGAN é um *framework* de aprendizagem profunda que utiliza o conceito das GANs para gerar imagens super-resolvidas. Para caracterizar uma SRGAN, torna-se necessário definir os conceitos de *downsampling* e *upsampling*. O *downsampling* é uma técnica de processamento digital de imagens utilizada para reduzir a resolução espacial de uma imagem, enquanto que o *downsampling* é uma técnica inversa, isto é, utiliza-se para obter uma imagem de resolução espacial superior àquela originalmente fornecida como entrada [27]. Durante a etapa de treinamento, imagens de alta resolução (HR) são pré-processadas, utilizando uma técnica de *downsampling*, para produzir imagens de baixa resolução (LR). A rede geradora aprende a super-resolver imagens LR, com base na operação de pixel *shuffle*. Esta operação, também conhecida como convolução de subpixel, executa o *upsampling* a nível de subpixel, especificamente na última *feature map* da rede neural. Diferentemente de como ocorre na arquitetura SRCNN [28], que é no início da rede, o que demanda um custo computacional elevado para executar. Este processo de convolução utiliza *stride* de  $\frac{1}{r}$  no espaço LR, a partir de um filtro  $W_s$  de tamanho  $k_s$ . Dessa maneira, os pesos a nível de pixel não são ativados e, portanto, não são calculados. A operação de *upsampling* por pixel *shuffle* é definida conforme Equação (2.7) [29].

$$\mathbf{I}^{SR} = \mathcal{PS}(W_L * f^{L-1})(\mathbf{I}^L R) + b_L, \quad (2.7)$$

onde  $\mathcal{PS}$  é o operador periódico de *shuffling* que reorganiza os elementos do tensor com dimensões  $H \times W \times Cr^2$  para o tensor com dimensões  $rH \times rW \times C$  e  $W_L$  é o operador de convolução com dimensões  $n_{L-1} \times r^2C \times k_L \times k_L$ . O operador  $\mathcal{PS}$  é definido conforme Equação (2.8) [29].

$$\mathcal{PS}(T)_{x,y,c} = T_{\lfloor \frac{x}{r} \rfloor, \lfloor \frac{y}{r} \rfloor, c \cdot r \cdot \text{mod}(y,r) + c \cdot \text{mod}(x,r)}, \quad (2.8)$$

onde  $x$  e  $y$  são as coordenadas de saída do pixel no espaço HR,  $r$  é o fator de escala de super-resolução,  $\lfloor \cdot \rfloor$  é o operador piso e  $\text{mod}$  é o operador módulo. Posteriormente,

a imagem de super-resolvida (SR) da rede geradora e a respectiva imagem HR são submetidas a uma rede discriminadora  $D$ , responsável por classificá-las em imagens *ground-truth* ou gerada. Na saída da rede discriminadora, realiza-se a calibração das redes  $G$  e  $D$ , por meio da propagação das perdas adversária e de conteúdo, que compõe a perda perceptiva da SRGAN. A função de perda perceptiva é definida conforme Equação (2.9) [8].

$$l^{SR} = l_X^{SR} + 10^{-3}l_{Gen}^{SR}. \quad (2.9)$$

O componente  $l_X^{SR}$  está associado à perda de conteúdo, enquanto o componente  $10^{-3}l_{Gen}^{SR}$  está associado à perda adversária, em que juntos são responsáveis pela extração das *features* das imagens HR e SR para minimização do erro quadrático médio (MSE). Na etapa de testes, após o encontro do ponto de equilíbrio da fórmula do problema mini-max (ver equação (2.5)), aplica-se um modelo pré-treinado  $G$  para gerar imagens SR a partir de imagens LR.

## 2.5 Trabalhos Relacionados

No trabalho proposto por Silva e Jung [11], apresenta-se um sistema de reconhecimento automático de placas veiculares (ALPR). Neste, o objeto de interesse, ou seja, a placa veicular, encontra-se ligeiramente distorcida. Adicionalmente, utiliza-se uma variedade de cenários como, por exemplo, com o objeto de interesse situado em vista frontal ou oblíqua ou com veículos situados em níveis de distância (próximo, intermediário, distante) em relação à câmera. Os autores utilizam métodos usuais para detecção de veículos, placas veiculares e reconhecimento óptico de caracteres (OCR) [30]. O sistema proposto é composto por três etapas principais: (i) detecção de veículos, em que se utiliza um modelo pré-treinado baseado em YOLO [5] sob uma estrutura Darknet [31]. A versão da arquitetura foi a segunda [26] (YOLOv2) e a base de dados de treinamento empregada foi a PASCAL-VOC [32]; (ii) detecção de placas, em que as regiões contendo veículos são processadas para detectar placas veiculares, utilizando-se um *bounding box*. Para este objetivo, foi proposta uma rede WPOD-NET de detecção e correção de perspectiva de placas veiculares distorcidas inspirada em três arquiteturas: YOLO, detector de disparo único (SSD) [33] e

rede de transformações espaciais (STN) [34]. Esta rede foi implementada sob uma estrutura Tensorflow [35] e (iii) OCR, em que os caracteres das placas veiculares são segmentados a partir de uma rede YOLO sob uma estrutura Darknet chamada OCR-NET. Os resultados foram obtidos a partir de experimentos sob quatro bases de dados independentes: OpenALPR [36], SSIG Test [37], AOLP RP [38] e CD-HARD (base própria dos autores). Em termos de resultados, a acurácia foi de 89,33%.

Na pesquisa desenvolvida por Khazaei *et al.* [13], apresenta-se um método de detecção de placas veiculares iranianas com execução em tempo real. O sistema é utilizado em ambientes não controlados (*e.g.*, diferentes resoluções e iluminação). O sistema proposto é dividido em: (i) treinamento, em que uma base de dados criada com classes de diferentes vistas (*e.g.*, frontais e traseiras) é empregada numa rede YOLOv3 [39] para gerar um modelo de detecção; e (ii) teste, em que o modelo treinado é usado e a imagem de entrada é transformada numa rede de propostas de regiões, cujo objeto de interesse é detectado a partir dessas propostas. Os resultados experimentais foram obtidos a partir de um conjunto de dados criado e ampliado a partir de técnicas de escala e corte, por exemplo. Para remover ruídos existentes e lidar com a elevada iluminação, melhorando o desempenho do modelo gerado, as imagens de treino foram pré-processadas com filtro passa-baixa e com equalização de histograma, respectivamente. A avaliação do sistema considerou as métricas da matriz de confusão [40], precisão e recall, obtendo uma acurácia de 97,90%.

No trabalho publicado por Kim *et al.* [9], apresenta-se um sistema de reconhecimento de modelos de veículos em imagens de baixa resolução. O objeto de interesse são veículos que depois são utilizados para classificar o seu modelo. O método proposto processa imagens de câmeras de circuito fechado de televisão (CCTV) que possuem baixa resolução, dificultando o reconhecimento de veículos. O sistema é composto por três etapas: (i) detecção de veículos, em que um modelo treinado em YOLOv3 é utilizado para detectar veículos e recortar a região de interesse do objeto identificado; (ii) super-resolução de imagens, em que o *frame* de baixa resolução recortado é fornecido como entrada de uma rede de super-resolução baseada em GAN (SRGAN) [8] para melhorar sua qualidade; e (iii) classificação dos modelos de veículos, em que a imagem super-resolvida é fornecida como entrada de uma rede

neural convolucional (CNN) [25] que, ao considerar a direção do veículo na imagem (*e.g.*, frontal, lateral e traseira), aplica ao modelo de classificação correspondente àquela direção. Como resultados obtidos, obteve-se com o uso de super-resolução a acurácia de 78% no sistema proposto em ambientes não-controlados.

Na pesquisa proposta por Rabbi *et al.* [12], apresenta-se um método de detecção de objetos pequenos em imagens de baixa resolução. O objeto de interesse são veículos contidos em imagens de satélites. As principais contribuições da pesquisa foram a proposta de uma arquitetura aprimorada de SRGAN, que melhora a qualidade da imagem e das bordas do objeto de interesse, e uma base de dados de imagens de satélites com tanques de armazenamento de gás e óleo (OGST). A arquitetura proposta consiste de: (i) rede de super-resolução projetada em GAN, formada por três componentes (geradora, discriminadora e aprimoramento de borda) que gera dados super-resolvidos a partir de um treinamento com um par de imagens de baixa (LR) e alta resolução (HR) e melhora a qualidade das bordas do objeto de interesse devido a ruídos existentes empregando um operador Laplaciano; e (ii) uma rede de detectores composta pelas redes Faster R-CNN [41] e SSD capazes de localizar os veículos por meio de um *bounding box*. Os resultados experimentais foram obtidos sob 2 bases de dados: COWC [42] e OGST (base própria dos autores). A acurácia do método proposto foi de 93,60% no COWC e 81,40% no OGST.

No trabalho desenvolvido por Lee *et al.* [10], apresenta-se um método de reconhecimento de placas veiculares em vídeos de vigilância de tráfego. O objeto de interesse são placas veiculares com tamanho pequeno (*i.e.*, menor que 60x60 pixels) contidos em imagens de baixa resolução. A arquitetura proposta inclui: (i) treinamento, em que um modelo é gerado numa rede YOLOv2 para detectar veículos, placas e caracteres coreanos, considerando informações de contexto hierárquicas entre o veículo e a placa. Ou seja, para detectar a placa, é necessário detectá-la no *bounding box* do veículo. Além disso, um modelo é gerado numa rede SRGAN para super-resolver o objeto de interesse em LR; (ii) teste, em que uma imagem de câmera de vigilância é fornecida como entrada na rede detecção de veículos e placas que, depois que a placa é obtida em LR, é alimentada na rede de super-resolução para reconstruí-la em HR e segmentar os caracteres coreanos. O resultados experimentais foram obtidos a partir de um conjunto de dados heterogêneo (*e.g.*, placas pequenas,

placas pequenas e inclinadas, placas medianas, etc) coletado e anotado pelos próprios autores, obtendo uma acurácia de 86,45% em ambientes não-controlados.

Na Tabela 2.1, apresenta-se uma comparação resumida das principais informações dos trabalhos relacionados, levando-se em consideração a metodologia proposta. Nesta, destaca-se que a acurácia apresentada não considera a base de dados de testes organizada nesta dissertação, sendo apenas os resultados obtidos pelos autores em seus respectivas bases de dados. As pesquisas realizadas pelos autores Silva e Jung [11] e pelos autores Khazaei *et al.* [13] não abordam o método de super-resolução baseado em GAN para detecção do objeto de interesse. Adicionalmente, nenhum dos trabalhos relacionados investigam a detecção de pequenos objetos utilizando imagens adquiridas por VANT, que corrobora o entendimento de que esta área de estudo ainda é pouco explorada. Portanto, a principal contribuição desta dissertação é propor uma metodologia para detecção de placas veiculares em ambientes aéreos baseado no método de super-resolução por GAN, com o objeto de interesse apresentando baixa resolução em pixel em relação à imagem de entrada. A base de dados utilizada nesta pesquisa é organizada a partir de imagens aéreas, predominantemente por VANT, extraídas de vídeos em diferentes resoluções, sendo a placa veicular o objeto de interesse. Nesta, apresenta-se um cenário diversificado, composto de imagens que possuem diferentes condições de clima, iluminação e variação de angulação da câmera. Diferentemente, dos trabalhos propostos pelos autores Rabbi *et al.* [12], que é concebida de imagens de satélite de baixa resolução cujo objeto de interesse são veículos.

Tabela 2.1: Tabela comparativa entre a método proposto e trabalhos relacionados.

Autoria	Base de Dados	Características da Imagem	Objeto de Interesse	Método para Detecção do Objeto de Interesse	Método para Super-Resolução	Acurácia
Silva e Jung [11]	OpenALPR [36], SSIG Test [37], AOLP RP [38] e CD-HARD (Autoria de Silva e Jung)	Câmera fixa	Placa Veicular	WPOD-NET [11]	-	89,33%
Khazae <i>et al.</i> [13]	Autoria de Khazae	Câmera fixa	Placa Veicular	YOLOv3 [39]	-	97,90%
Kim <i>et al.</i> [9]	Autoria de Kim	CCTV	Modelo de Veículo	CNN [25]	SRGAN [8]	78,00%
Rabbi <i>et al.</i> [12]	COWC [42] e OGST [43] (Autoria de Rabbi)	Satélite	Veículo	Faster R-CNN [41] e SSD [33]	EESRGAN [12]	93,60% e 84,40%
Lee <i>et al.</i> [10]	Autoria de Lee	CCTV	Placa Veicular	YOLOv2 [26]	SRGAN [8]	86,45%
<b>Método proposto</b>	Autoria Própria	VANT e CCTV	Placa Veicular	WPOD-NET [11]	Fast-SRGAN [44]	54,81%

# Capítulo 3

## Metodologia Proposta

### 3.1 Introdução

Nesta dissertação, propõe-se uma metodologia para detecção de placa de licenciamento veicular em ambientes aéreos. Para isso, utiliza-se a técnica de super-resolução baseado em GAN para super-resolver os veículos detectados devido à dificuldade de identificar e localizar o objeto de interesse, por apresentar baixa qualidade visual e baixa resolução espacial em pixel. A metodologia proposta é composta por três sistemas distintos, sendo um sistema de referência, um sistema que utiliza super-resolução e um sistema que utiliza super-resolução com equalização de histograma. O sistema de referência é utilizado para comparação da metodologia proposta. Neste, emprega-se o detector YOLO em duas versões distintas para detecção de veículos e o detector WPOD-NET para detecção de placas veiculares. Considerando a presença de vários veículos por frame em um *background* complexo, seleciona-se um único veículo para detecção de uma única placa veicular, devido a base de dados utilizada para avaliar o desempenho da metodologia proposta conter um único rótulo de veículo e placa veicular por frame. O sistema proposto com super-resolução é utilizado para super-resolver o veículo detectado que contém a placa veicular em baixa resolução em pixel. Emprega-se uma técnica baseado em GAN para super-resolução do veículo detectado e uma técnica de redimensionamento de imagem da placa veicular para mantê-la nas mesmas proporções do veículo. O sistema proposto com equalização de histograma é utilizado para realce da imagem do veículo detectado, por apresentar baixo contraste. Para isso, emprega-se o método CLAHE para



ajustar o contraste da imagem super-resolvida do veículo a um nível desejado. Em seguida, detecta-se a placa veicular e a redimensiona nas mesmas proporções do veículo. Destaca-se que foi investigado um quarto sistema que utiliza a super-resolução na imagem de entrada. Neste, identificaram-se dois problemas que inviabilizou sua utilização nesta dissertação. O primeiro problema é referente ao elevado tempo para execução do sistema, principalmente, por conta da super-resolução da imagem de entrada com diferentes resoluções em pixel (i.e.,  $1920 \times 1080$  pixels e  $1280 \times 720$  pixels). O segundo problema é referente à insuficiência de memória durante o processo de super-resolução da imagem de entrada, que faz a execução falhar.

## 3.2 Sistema de detecção de placa de licenciamento veicular sem super-resolução

Na Figura 3.1, apresenta-se o primeiro sistema proposto, denominado por sistema de detecção de placa de licenciamento veicular (VLP) sem super-resolução, que será utilizado como referência (*baseline*). Conforme comentado na seção anterior, este sistema será comparado, em termos de desempenho, com os demais sistemas propostos nesta dissertação. Para conceber o sistema proposto de referência, adaptou-se dois blocos de um sistema similar [11], proposto por Silva e Jung. Especificamente, os blocos: (i) para detecção de veículos e; (ii) para detecção de placas veiculares. Além disso, não foi empregado o algoritmo de correção de perspectiva da placa veicular, por haver uma discrepância no valor da métrica IoU, quando comparado com os rótulos manuais gerados pelo software LabelImg [45], ou seja, os rótulos manuais são delimitados por retângulos, enquanto que os rótulos detectados por correção de perspectiva são delimitados sobre os contornos irregulares do objeto de interesse. O sistema proposto de referência (*baseline*) é composto por cinco blocos. Cada um será descrito, conforme a seguir.

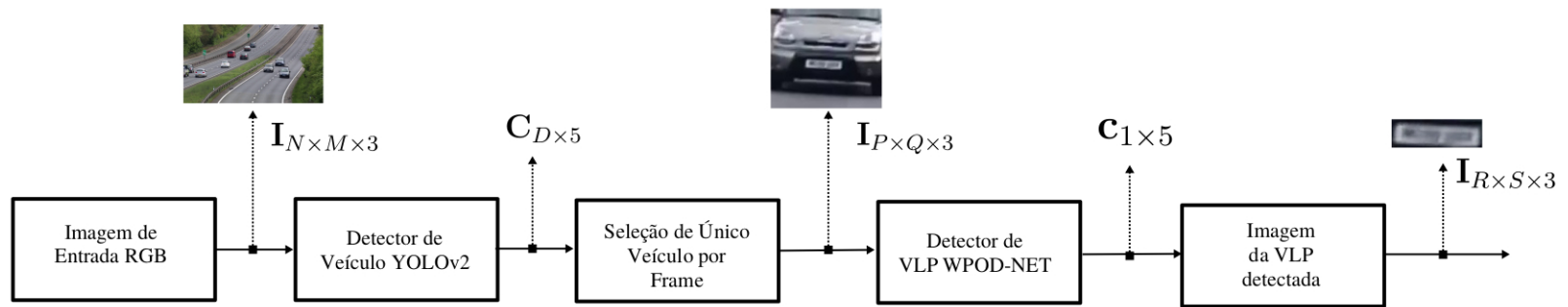


Figura 3.1: Diagrama em blocos para o sistema de detecção de placa de licenciamento veicular sem super-resolução (*baseline*).

## Imagem de Entrada RGB

A imagem de entrada utilizada neste sistema é colorida<sup>1</sup>. A resolução utilizada foi de  $1920 \times 1080$  pixels e  $1280 \times 720$  pixels para avaliar o comportamento do sistema em diferentes resoluções. No Capítulo 4, descrevem-se, com maiores detalhes, as imagens de entrada para os sistemas propostos.

## Detector de Veículo YOLOv2

Neste bloco, é realizada a detecção dos veículos contidos na imagem de entrada. Embora o nome do referido bloco utilize a versão 2 do detector YOLO por empregar um modelo pré-treinado na arquitetura YOLOv2 [26], utilizando-se o PASCAL-VOC *dataset* [32], emprega-se, além deste, a versão 3. Considerando que o referido modelo pré-treinado é capaz de detectar 20 classes diferentes, foram selecionadas apenas as classes *carro* e *ônibus*, por se tratar de tipos de veículos. A arquitetura empregada nesta dissertação foi a YOLOv2 clássica, composta por 23 camadas convolucionais para extração de características e 5 camadas de *maxpooling* para redução da dimensionalidade das *feature maps*.

A saída deste bloco, que corresponde a uma matriz  $\mathbf{C}_{D \times 5}$ , onde a dimensão  $D$  representa cada instância do objeto detectado na imagem e a outra dimensão representa o número de variáveis do formato YOLO, é descrita a seguir. Primeiramente, considera-se que um objeto é detectado por meio de um *bounding box*, sendo necessário apenas dois vértices para formação do retângulo, um vértice para o canto superior esquerdo  $t_l$  e outro vértice para o canto superior direito  $b_r$ . A seguir, o formato YOLO de saída é definido pelos valores  $x$ ,  $y$ ,  $w$  e  $h$ , sendo o par ordenado  $(x,y)$  o centro do *bounding box* e  $w$  e  $h$  a largura e a altura do *bounding box*, respectivamente. Os valores de saída deste bloco são obtidos conforme Equações (3.1) até

---

<sup>1</sup>Uma imagem colorida ou no espaço de cores RGB é composta por três canais: Vermelho (**R**), Verde (**G**) e Azul (**B**).

(3.4) [26].

$$x = \sigma(t_x) + c_x, \quad (3.1)$$

$$y = \sigma(t_y) + c_y, \quad (3.2)$$

$$w = p_w e^{t_w}, \quad (3.3)$$

$$h = p_h e^{t_h}, \quad (3.4)$$

onde  $\sigma$  é a função de ativação sigmóide, o par ordenado  $(c_x, c_y)$  são as coordenadas da centróide do *bounding box*,  $p_w$  e  $p_h$  são a largura e altura do *bounding box anterior*<sup>2</sup> e  $t_x, t_y, t_w$  e  $t_h$  são os *bounding boxes* que a rede detecta para cada célula<sup>3</sup> de um *feature map* de saída. Para utilizar o formato de entrada dos parâmetros da métrica IoU no bloco seguinte, bem como para a avaliação do desempenho do sistema, foi realizada a conversão para as variáveis  $t_l$  e  $b_r$  conforme Equações (3.5) e (3.6).

$$t_l = \left( x - \frac{w}{2}, y - \frac{h}{2} \right), \quad (3.5)$$

$$b_r = \left( x + \frac{w}{2}, y + \frac{h}{2} \right). \quad (3.6)$$

## Seleção de Único Veículo por Frame

Neste bloco, é realizada a seleção de único veículo por frame utilizando a métrica IoU. Esta métrica compara dois rótulos, um é o rótulo manual e o outro é rótulo detectado. Considera-se rótulo manual (ou *ground-truth*) o *bounding box*  $b_{GT}$  que é gerado por um software de rotulação manual, enquanto que o rótulo detectado é o *bounding box*  $b_P$  que é gerado por um detector de objetos. Seleciona-se o veículo detectado que apresentasse IoU acima de um limiar, ou seja,  $IoU \geq L$ . No entanto, se existir mais de um valor IoU que atendesse o limiar para um mesmo veículo detectado, então somente o maior IoU era utilizado no bloco seguinte e os demais veículos eram ignorados. A métrica IoU é definida conforme Equação (5.1) [46].

<sup>2</sup>Um *bounding box anterior* corresponde a um caixa de ancoragem (do inglês, *anchor box*) que é responsável por aprender a localização, forma e tamanho ideal para um objeto de interesse.

<sup>3</sup>O detector YOLO divide a imagem de entrada em uma *grid*  $S \times S$ . Cada unidade da *grid* corresponde a uma célula que poderá ser responsável por detectar o objeto de interesse.

$$IoU_P^{GT} = \frac{\cap(b_{GT}, b_P)}{\cup(b_{GT}, b_P)}, \quad (3.7)$$

onde  $\cap(b_{GT}, b_P)$  é a interseção entre os rótulos manuais e detectados e  $\cup(b_{GT}, b_P)$  é a união entre os rótulos manuais e detectados.

## Detector de VLP WPOD-NET

Neste bloco, é realizada a detecção da placa veicular a partir do único veículo selecionado anteriormente. Neste, emprega-se um modelo pré-treinado na arquitetura WPOD-NET [11], utilizando-se amostras das bases de dados *Cars* [47], SSIG [37] e AOLP *dataset* [38]. A arquitetura empregada nesta dissertação foi a WPOD-NET clássica, composta por 21 camadas convolucionais e 4 camadas de *maxpooling* com *stride* igual a 2. Esta arquitetura é baseada em YOLO, portanto, apresenta o mesmo formato de saída. A saída do detector WPOD-NET é representada por um vetor linha  $\mathbf{c}_{1 \times 5}$ , composta por cinco valores:  $c$ ,  $x$ ,  $y$ ,  $w$  e  $h$ . Assim como na saída do detector YOLOv2, foi realizada a conversão de seus valores em função de  $t_l$  e  $b_r$ .

### 3.3 Sistema de detecção de placa de licenciamento veicular com super-resolução

Na Figura 3.2, apresenta-se o segundo sistema proposto, denominado por sistema de detecção de placa de licenciamento veicular (VLP) com super-resolução. Para conceber este sistema, adicionou-se o bloco de super-resolução de imagem ao sistema de referência (*baseline*) descrito na Seção 3.2. Para este sistema, conforme

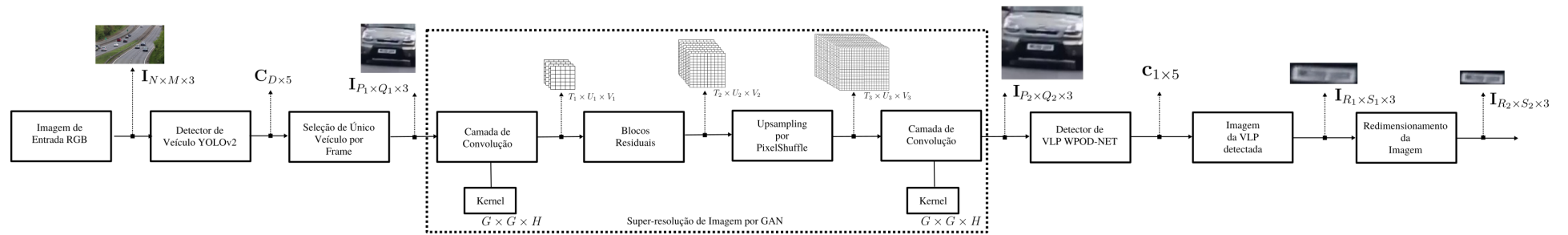


Figura 3.2: Diagrama em blocos para o sistema de detecção de placa de licenciamento veicular com super-resolução.

descrito brevemente na Seção 3.1, utilizou-se o método de super-resolução baseado em GAN [8]. O segundo sistema proposto é composto por dez blocos. Cada um dos blocos adicionados será descrito, conforme a seguir.

## Camada Convolutacional e Bloco Residual

No bloco *Camada Convolutacional*, realiza-se a extração das características da imagem de entrada para a super-resolução baseada em GAN. Para isso, emprega-se a operação de convolução, utilizando o filtro (ou kernel)  $\mathbf{W}_{G \times G \times H}$ , e a imagem de entrada do veículo detectado  $\mathbf{I}_{P_1 \times Q_1 \times 3}$ , quando ocorrer na primeira camada  $L$  da rede neural, ou a *feature map*, quando ocorrer em camadas ocultas  $L + n$ . A saída da camada convolutacional é a *feature map* da próxima camada  $L + 1$ , que é executada aplicando-se a função de ativação ReLU<sup>4</sup> [19]. A rede neural geradora  $G$  utilizada no bloco de super-resolução de imagem por GAN possui 10 camadas convolucionais. No próximo bloco, utiliza-se 5 blocos residuais, que são estruturas compostas por camadas convolucionais e de ativação ReLU empregadas para realizar conexões de atalho entre camadas para evitar a saturação da precisão do modelo. Assim, os pesos do modelo são atualizados nas camadas seguintes, por haver conexões de atalho, caso a próxima camada não seja ativada.

## *Upsampling* por Pixel *Shuffle*

Neste bloco, utiliza-se uma adaptação no modelo pré-treinado da rede geradora  $G$  para que a operação de *upsampling* por pixel *shuffle* contemple imagens de entrada com múltiplas resoluções. Este procedimento está presente na SRGAN clássica, proposta por Haza [44], porém apresenta um erro ao informar que o modelo pré-treinado considera apenas imagens de entrada de resolução  $96 \times 96 \times 3$ . Portanto, realiza-se a substituição das dimensões  $P_1 \times Q_1 \times 3$  de entrada na primeira camada da rede neural para receber quaisquer valores de resolução e, em seguida, reconstrói-se a arquitetura do modelo copiando os pesos do modelo antigo para um novo. Assim, o novo modelo pré-treinado da rede geradora  $G$  está habilitado para

---

<sup>4</sup>A função de ativação ReLU (do inglês, *rectified linear unit*), é uma função de saída igual à zero para entradas negativas e de saída igual ao próprio valor de entrada para entradas positivas.

super-resolver imagens de entrada de diferentes dimensões.

## Redimensionamento de Imagem

Nesta dissertação, realiza-se a redução das dimensões da imagem da placa veicular detectada, por apresentar uma proporção quatro vezes maior em relação à imagem do veículo detectado. Para reduzir as dimensões da imagem da placa veicular detectada, são selecionados os valores  $t_l$  e  $b_r$  do *bounding box*. Então, são divididos os componentes  $x$  e  $y$  de cada variável pelo fator de escala de super-resolução, isto é,  $4\times$ . Assim, as coordenadas de localização da placa veicular estão nas mesmas proporções do veículo detectado. A operação de redimensionamento de imagem é definida conforme as Equações (3.8) e (3.9).

$$t_{l<resize>} = \frac{1}{4} \left( x - \frac{w}{2}, y - \frac{h}{2} \right), \quad (3.8)$$

$$b_{r<resize>} = \frac{1}{4} \left( x + \frac{w}{2}, y + \frac{h}{2} \right), \quad (3.9)$$

onde  $t_{l<resize>}$  e  $b_{r<resize>}$  são os cantos superior esquerdo e inferior direito do *bounding box* da placa veicular reduzida em  $4\times$ . O redimensionamento da imagem da placa veicular detectada tornou-se necessário, em razão da métrica IoU utilizar, na avaliação de desempenho do sistema proposto com super-resolução, os rótulos manuais e detectados para comparação. Assim, o rótulo manual do objeto de interesse está na escala original, isto é, em  $1\times$ . Portanto, o rótulo detectado permanece na escala  $1\times$ . Isso evita discrepâncias nos valores obtidos.

### 3.4 Sistema de detecção de placa de licenciamento veicular com equalização de histograma

Na Figura 3.3, apresenta-se o terceiro sistema proposto, denominado por sistema de detecção de placa de licenciamento veicular (VLP) com equalização de histograma. Este sistema é concebido, adicionando-se os blocos de histograma de imagem e equalização de histograma ao segundo sistema proposto descrito na Seção



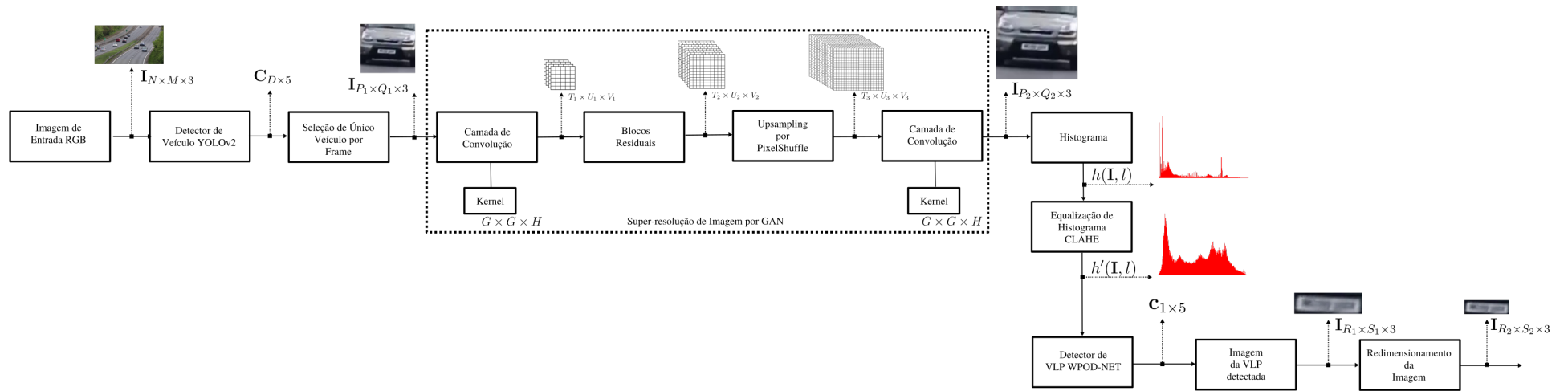


Figura 3.3: Diagrama em blocos para o sistema de detecção de placa de licenciamento veicular com equalização de histograma.

3.3. Para este sistema, conforme descrito brevemente na Seção 3.1, utilizou-se o método de equalização de histograma CLAHE [18]. O terceiro sistema proposto é composto por doze blocos. Cada um dos blocos adicionados será descrito, conforme a seguir.

## Histograma de Imagem

Nesta dissertação, para preservar o canal de cores da imagem original no processo de equalização de histograma, converte-se a imagem RGB para o espaço de cores HSV e, em seguida, aplica-se a função de histograma  $h$  no canal de brilho  $V$  da imagem. A conversão da imagem no espaço de cores RGB para HSV é definida conforme Equações (3.10) até (3.15) [48].

$$M = \max\{r, g, b\}, \quad (3.10)$$

$$m = \min\{r, g, b\}, \quad (3.11)$$

$$c = M - m, \quad (3.12)$$

$$v = M, \quad (3.13)$$

$$h = \begin{cases} \left(\frac{g-b}{c} \bmod 6\right) \times 60, & r = M, c \neq 0 \\ \left(\frac{b-r}{c} + 2\right) \times 60, & g = M, c \neq 0 \\ \left(\frac{r-g}{c} + 4\right) \times 60 & b = M, c \neq 0 \\ 0 & c = 0, \end{cases} \quad (3.14)$$

$$s = \frac{c}{v}, \quad (3.15)$$

onde  $M$  é o maior valor entre os canais R, G e B,  $m$  é o menor valor entre os canais R, G e B e  $c$  é a diferença entre o maior e o menor valor definido anteriormente.

## Equalização de Histograma CLAHE

A equalização de histograma CLAHE [18] é usada nesta dissertação por existir imagens super-resolvidas com baixo contraste. Considerando que um modelo de rede neural é uma estrutura sensível a variações de pixel, utiliza-se um processo

de equalização de histograma baseado em regiões por proporcionar a melhoria da qualidade da imagem do objeto de interesse. Na Figura 3.4, apresentam-se três exemplos de imagem, uma original do veículo detectado e outra modificada pelo método CLAHE. Além disso, exibem-se seus respectivos histogramas. Emprega-se este método para aprimorar a saída do bloco de super-resolução de imagem por GAN devido à existência de falsos-negativos durante a detecção de placas veiculares. O realce no contraste das imagens super-resolvidas é uma alternativa para melhoria da taxa de detecção, por possibilitar a diferenciação do objeto de interesse em relação ao fundo, a partir da distribuição dos valores de intensidade de pixels da imagem do veículo detectado.

O processo de equalização de histograma CLAHE é descrito a seguir. Inicialmente, divide-se a imagem de entrada em uma *grid*  $P \times P$  e calcula-se o histograma de cada região obtida na imagem, contabilizando a frequência de ocorrência de cada intensidade de nível de pixel presente na região. Em seguida, equaliza-se o histograma usando a normalização (i.e., dividindo-se cada frequência de intensidade de nível de pixel pelo número total de pixels na imagem) e a função de distribuição acumulada (i.e., somando-se cada fração de pixel com intensidade menor ou igual a um limiar  $L$ ). Por fim, aplica-se o limite de recorte  $\beta$  pra ajustar o contraste. O limite de recorte é definido conforme Equação (3.16).

$$\beta = \frac{M}{N} \left( 1 + \frac{\alpha}{100} (s_{max} - 1) \right), \quad (3.16)$$

onde  $\alpha$  é o fator de recorte;  $s_{max}$  é a inclinação máxima desejada; e  $M$  e  $N$  são a quantidade de linhas e colunas da região da imagem.

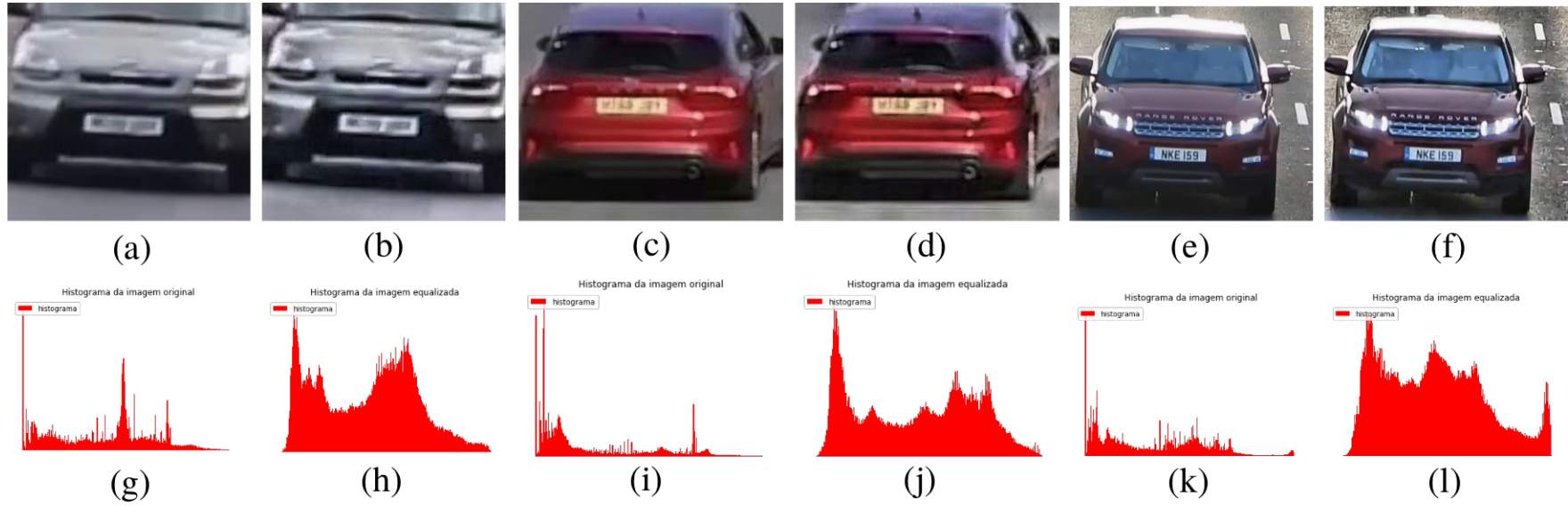


Figura 3.4: Nas Figuras 3.4(a) até 3.4(f), apresentam-se a imagem original e a equalizada por CLAHE de três veículos detectados. Nas Figuras 3.4(g) até 3.4(l), apresentam-se os histogramas da imagem original e equalizada, respectivamente.

### 3.5 Parâmetros dos sistemas propostos

Na Tabela 3.1, são apresentados os parâmetros utilizados nos três sistemas propostos desta dissertação, bem como a descrição de cada um.

Tabela 3.1: Parâmetros dos três sistemas propostos para detecção de placa de licenciamento veicular.

Parâmetro	Descrição	Sistema proposto		
		Detecção de placa de licenciamento veicular sem super-resolução	Detecção de placa de licenciamento veicular com super-resolução	Detecção de placa de licenciamento veicular com equalização de histograma
$\mathbf{I}_{N \times M \times 3}$	Imagem de entrada	✓	✓	✓
$\mathbf{I}_{P_1 \times Q_1 \times 3}$	Imagem do veículo detectado	✓	✓	✓
$\mathbf{I}_{P_2 \times Q_2 \times 3}$	Imagem super-resolvida do veículo detectado		✓	✓
$\mathbf{I}_{R_1 \times S_1 \times 3}$	Imagem da placa veicular detectada	✓	✓	✓
$\mathbf{I}_{R_2 \times S_2 \times 3}$	Imagem redimensionada da placa veicular detectada		✓	✓
$T_1 \times U_1 \times V_1$	Dimensões da <i>feature map</i> resultante da 1 <sup>o</sup> camada convolucional		✓	✓
$T_2 \times U_2 \times V_2$	Dimensões da <i>feature map</i> resultante dos blocos residuais		✓	✓
$\mathbf{C}_{D \times 5}$	Matriz composta pela classe e coordenadas de localização do veículo detectado	✓	✓	✓
$\mathbf{c}_{1 \times 5}$	Vetor composto pela classe e coordenadas de localização da placa veicular detectada	✓	✓	✓
$h(\mathbf{I}, l)$	Histograma da imagem super-resolvida			✓
$h'(\mathbf{I}, l)$	Histograma equalizado da imagem super-resolvida			✓

# Capítulo 4

## Base de Dados

A base de dados utilizada nos experimentos contempla outro objetivo desta dissertação, sendo considerada uma pequena contribuição. Esta base de dados é organizada para avaliar o desempenho da metodologia proposta para detecção de placas veiculares em ambientes aéreos. Considerando a utilização, nesta pesquisa, de modelos pré-treinados para detecções de veículos e de placas veiculares, bem como para a super-resolução de imagem, na Seção 4.1, descrevem-se de forma breve as bases de dados utilizadas por esses modelos. Na Seção 4.2, descreve-se em detalhes o processo de aquisição da base de dados de testes, bem como a sua organização para uso nos experimentos desta dissertação.

### 4.1 Base de dados dos modelos pré-treinados utilizados

Os modelos pré-treinados empregados nesta dissertação são utilizados pelos autores Silva e Jung [11] para detecções de veículos e de placas veiculares e pelo autor Reza [44] para super-resolução de imagem. Apresenta-se uma breve descrição de cada base de dados utilizada pelos modelos pré-treinados, conforme a seguir:

- A base de dados PASCAL *visual object classes* (VOC) [32] é utilizada para detecção de objetos, com suporte a 20 classes diferentes (por exemplo, *car* e *bus*). No desafio de 2010, esta base de dados possuía 5011 imagens para treino e validação e 4952 para testes. No trabalho proposto pelos autores

Silva e Jung [11], utiliza-se um modelo pré-treinado por esta base dados, na arquitetura YOLOv2, para detecção de veículos, por apresentar uma precisão de 76,8% mAP.

- As bases de dados *Cars* [47], SSIG [37] e AOLP [38] são empregadas pelos autores Silva e Jung para treinamento de uma rede WPOD-NET para detecção de placas veiculares. Utilizam-se 196 imagens no total, sendo 105 provenientes da base de dados *Cars*, 40 do subconjunto de treinamento da base de dados SSIG e 51 do subconjunto de aplicação da lei do trânsito da base de dados AOLP. Na base de dados *Cars*, selecionam-se imagens contendo placas veiculares com *layouts*, como dos padrões europeu e americano. Na base de dados SSIG, selecionam-se imagens com placas veiculares brasileiras, enquanto que na AOLP são selecionadas placas tailandesas.
- A base de dados *diverse 2K* (DIV2K) [49] é usada para treinamento de uma rede neural apresentada pelo autor Raza [44], com o objetivo de super-resolver vídeos em tempo real, por meio de redes adversárias generativas, fornecendo como entrada vídeos de baixa resolução. Nesta, considera-se um *benchmark* para super-resolução de única imagem, contendo 1000 imagens RGB, com resolução espacial de  $2K$  e diferentes níveis de degradação. Esta base de dados está dividida em 800 imagens para treino e 200 para validação e testes.

## 4.2 Base de dados de testes

A base de dados de testes empregada nesta dissertação é de autoria própria e foi concebida entre os meses de outubro e dezembro de 2021, a partir da seleção de vídeos compartilhados na plataforma YouTube [50]. Utilizam-se critérios específicos para a busca dos vídeos, considerando os sistemas propostos e o problema de pesquisa. Dentre os critérios de pesquisa, elegem-se vídeos adquiridos por VANT em movimento ou estático, contendo veículos se movendo, seja se aproximando, seja se distanciando da fonte de aquisição (i.e., câmera). A grande quantidade de vídeos disponíveis para consulta, bem como a especificidade dos critérios para atendimento desta dissertação, exigiu um tempo razoável para seleção dos vídeos. Destaca-se que, durante o processo de seleção, existiam muitos vídeos com viés (i.e., adquiridos em

ambientes controlados cujas condições são conhecidas e manipuladas pelo usuário), tornado-se inadequados para a pesquisa desta dissertação. Como exemplo, podem-se citar vídeos adquiridos por VANT com a função de *tracking* habilitada, ou seja, com o VANT rastreando e se locomovendo de acordo com a posição de um único objeto de interesse. Este cenário é diferente do abordado em ambientes complexos. Na Tabela 4.1 a seguir, apresentam-se os termos utilizados durante a pesquisa por vídeos no YouTube, com o intuito de facilitar o processo de seleção de vídeos, com base nos critérios utilizados nesta dissertação. Destaca-se que o ambiente em que os vídeos são adquiridos é diversificado, com predominância de condições de clima ensolarado, sombreado ou nublado e em períodos diurnos ou noturnos, com alta e baixa iluminação, respectivamente. Um ambiente ensolarado é caracterizado pela presença de luz do dia que normalmente introduz interferências de sombras, caracterizando também como um ambiente sombreado. Quando uma imagem é adquirida em ambiente noturno, existe uma carência de luz, oferecendo quase nenhuma informação a nível de textura dos objetos na cena. Adicionalmente, imagens adquiridas em ambientes nublados são caracterizadas por cenas com falta de nitidez e detalhes, dificultando a percepção dos contornos dos objetos no *background* [1]. Todo este cenário contribui para a análise dos sistemas propostos em um *background* complexo.

Tabela 4.1: Termos utilizados na pesquisa de vídeos no YouTube.

Plataforma	Termos
Youtube	<i>static drone footage road,</i> <i>static drone footage highway,</i> <i>static drone footage traffic,</i> <i>drone footage cars road,</i> <i>roads drone shots</i>

Ao final do processo de buscas, selecionam-se 7 vídeos. Estes vídeos são adquiridos em duas resoluções distintas: i)  $1920 \times 1080$  pixels e ii)  $1280 \times 720$  pixels. Para criação da base de dados de testes, cada vídeo é pré-processado por um algoritmo de extração de frames, gerando uma base de dados de 1600 imagens coloridas. Na Tabela 4.2 a seguir, apresentam-se as especificações de formatação dos vídeos originais, as características do cenário capturado, bem como o *link* para acesso no YouTube. Por conter imagens aéreas e ser adquirido de um câmera de CCTV fixa de vigilância, o último vídeo é selecionado com a finalidade de avaliar a



robustez dos sistemas propostos em um cenário de baixa luminosidade e qualidade de imagem. Na Figura 4.2, apresentam-se amostras de imagens de cada um dos sete vídeos que compõe a base de dados de testes utilizadas nesta dissertação.

O processo de rotulação da base de dados de testes é descrita a seguir. Inicialmente, investiga-se o uso de um *framework* de rotulação automática de imagens. Dentre as alternativas existentes, a ferramenta Amazon SageMaker Ground Truth [51] para rotulação de dados em grande escala foi utilizada. No entanto, por limitação de dados rotulados na versão gratuita, substitui-se por um software de rotulação manual. Para conceber um conjunto de coordenadas de referência (i.e., padrão ouro) dos objetos de interesse, utiliza-se um software de rotulação manual chamado LabelImg [45], que permite rotular regiões contendo veículos e placas veiculares. Define-se como a rotulação manual: 1 (um) objeto veículo e 1 (um) objeto placa veicular por imagem de entrada para avaliação do sistema proposto. Utilizam-se todas as instâncias de um veículo fixado aleatoriamente até seu desaparecimento na cena. Os veículos e as placas veiculares são rotulados por caixas delimitadoras, totalizando 3200 rotulações, sendo 1600 para veículos e 1600 para placas veiculares. Porém, dependendo da angulação de captura das imagens aéreas, surgem objetos de interesse com perspectiva alterada (distorcida), contendo regiões não pertencentes ao objeto em questão. Na Figura 4.1 a seguir, apresenta-se um exemplo de rotulação da placa veicular com este cenário.

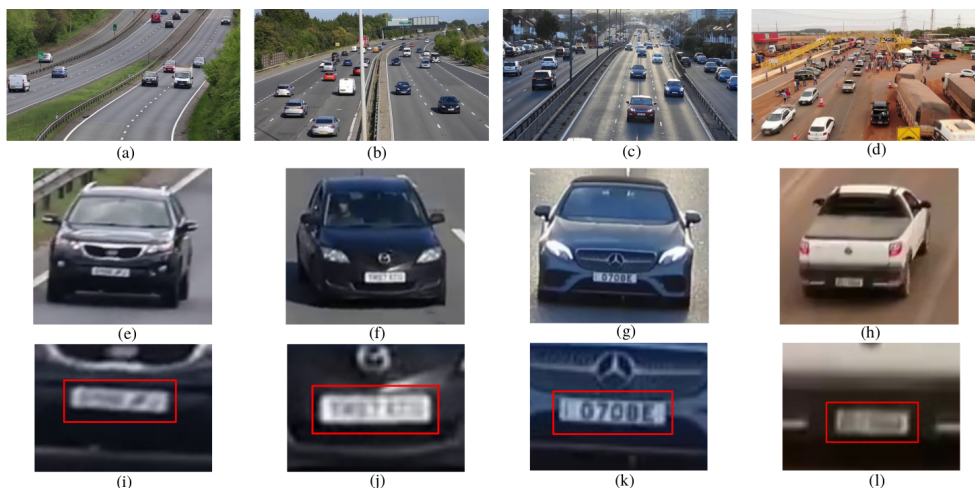


Figura 4.1: Exemplos de rotulação de placas veiculares com regiões não pertencentes ao objeto. Nas Figuras 4.1(a) até 4.1(d), apresentam-se as imagens originais. Nas Figuras 4.1(e) até 4.1(h), apresentam-se as imagens dos veículos. Nas Figuras 4.1(i) até 4.1(l), apresentam-se as imagens das placas veiculares rotuladas.

Tabela 4.2: Descrição da formatação dos vídeos e características do cenário capturado.

Vídeo	Resolução	fps	Duração	Quantidade de imagens	Características	Link para acesso
Vídeo 1	1920 × 1080	25	12s	300	Fonte de aquisição (VANT) parado, com objeto de interesse se aproximando. Período do dia: manhã e tarde. Condição climática: sol e sombreamento.	<a href="https://www.youtube.com/watch?v=UM0hX7nomi8">https://www.youtube.com/watch?v=UM0hX7nomi8</a> *1
Vídeo 2	1280 × 720	25	12s	300	Fonte de aquisição (VANT) parado, com objeto de interesse se aproximando. Período do dia: manhã e tarde. Condição climática: sol e sombreamento.	<a href="https://www.youtube.com/watch?v=YdW1U2hz6gs">https://www.youtube.com/watch?v=YdW1U2hz6gs</a>
Vídeo 3	1920 × 1080	25	12s	300	Fonte de aquisição (VANT) parado, com objeto de interesse se aproximando. Período do dia: manhã e tarde. Condição climática: sol e sombreamento.	<a href="https://www.youtube.com/watch?v=GCj8x7ou-U8">https://www.youtube.com/watch?v=GCj8x7ou-U8</a> *1
Vídeo 4	1920 × 1080	25	12s	300	Fonte de aquisição (VANT) em movimento, aproximando-se do objeto de interesse. Período do dia: tarde. Condição climática: nublado.	<a href="https://www.youtube.com/watch?v=DcljGB0Fluc">https://www.youtube.com/watch?v=DcljGB0Fluc</a>
Vídeo 5	1920 × 1080	25	12s	300	Fonte de aquisição (VANT) em movimento, aproximando-se do objeto de interesse. Período do dia: tarde. Condição climática: nublado.	<a href="https://www.youtube.com/watch?v=HDKdNVFDDDM">https://www.youtube.com/watch?v=HDKdNVFDDDM</a> *2
Vídeo 6	1920 × 1080	25	2s	50	Fonte de aquisição (VANT) parado, com objeto de interesse se aproximando. Período do dia: noite. Baixa iluminação, objeto de interesse com reflexão e ofuscamento devido aos faróis acesos.	<a href="https://www.youtube.com/watch?v=xEtM111Afhc">https://www.youtube.com/watch?v=xEtM111Afhc</a>
Vídeo 7	1280 × 720	25	2s	50	Fonte de aquisição (CCTV) fixo, com objeto de interesse se aproximando. Período do dia: noite. Baixa qualidade.	<a href="https://www.youtube.com/watch?v=i93yGtlq0yU">https://www.youtube.com/watch?v=i93yGtlq0yU</a>

\*1 Os vídeos 1 e 3 não estão mais disponíveis para consulta, devido às contas associadas estarem encerradas.

\*2 O vídeo 5 não está disponível para consulta, devido ao *link* armazenado durante o processo de seleção não corresponder ao vídeo original utilizado para a composição da base de dados de testes e por não

tê-lo encontrado novamente para visualização.

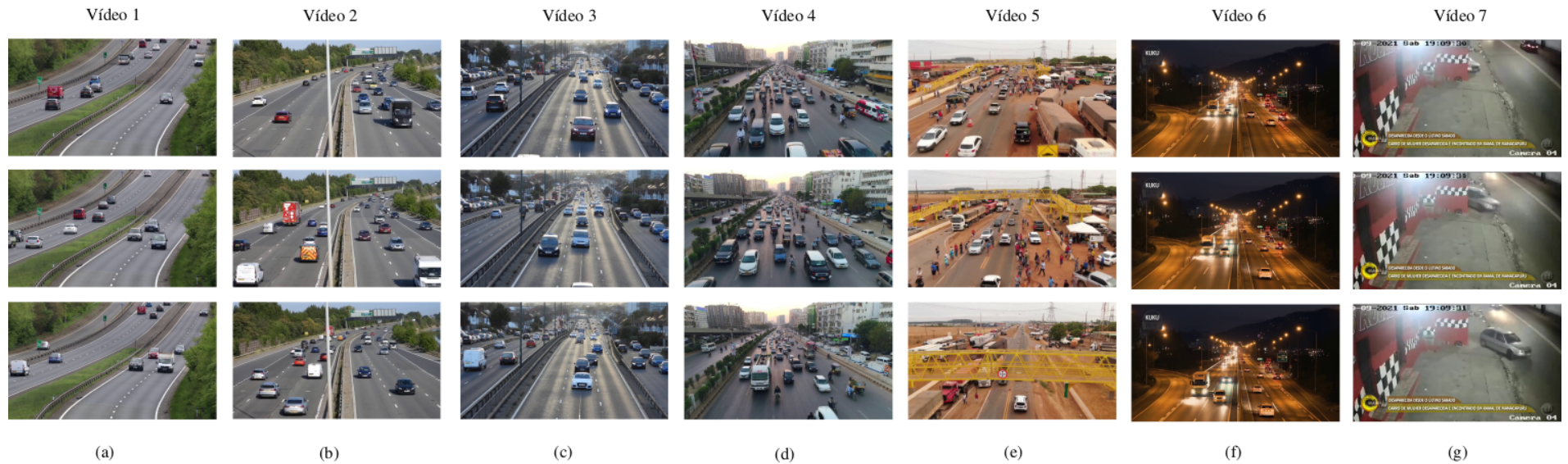


Figura 4.2: Da esquerda para direita: amostras de 3 frames dos vídeos de 1 a 7 que compõem a base de dados de 1600 imagens. Nas Figuras 4.2(a) até 4.2(e), apresentam-se imagens em condições ensolaradas e nubladas. Nas Figuras 4.2(f) e 4.2(g), apresentam-se imagens em condições noturnas, sendo a Figura 4.2(f) adquirida por VANT e a Figura 4.2(g) por CCTV.

# Capítulo 5

## Experimentos e Resultados

### 5.1 Introdução

Nesta seção, serão apresentadas as etapas do procedimento experimental, bem como a análise dos resultados de forma a validar a metodologia proposta para detecção de placas veiculares em ambientes aéreos.

Durante a execução dos experimentos, pretende-se responder os seguintes questionamentos da pesquisa:

- Supondo as imagens de entrada contendo condições climáticas adversas predominantes (por exemplo, ensolarada, sombreada ou nublada). Existe divergência de resultados considerando-se uma estratificação da base de dados?
- Supondo a dificuldade de detecção de placas veiculares em ambientes aéreos devido à sua baixa qualidade e resolução espacial em pixel. Se aplicarmos uma técnica de pré-processamento por equalização de histograma, existe mudança nos resultados?
- Se estratificarmos um grupo com imagens com baixa qualidade (por exemplo, noturna), o que ocorre com os sistemas? Existe uma discrepância nos resultados?

### 5.2 Setup

O *setup* dos experimentos desta dissertação é feito, conforme a seguir:

1. Ambiente de desenvolvimento: Os algoritmos são implementados utilizando o sistema operacional Linux, distribuição Ubuntu, na versão 18.04.5 LTS de 64 bits, em um processador Intel(R) Xeon(R) CPU @ 2,30GHz. Utiliza-se o serviço em nuvem Google Colaboratory [52] como ambiente de desenvolvimento, com habilitação da GPU gratuita, com a seguinte configuração: NVIDIA Tesla K80, 12GB, CUDA versão 11.1, cuDNN versão 8.0.5.
2. Para concepção dos algoritmos, emprega-se a linguagem de programação Python [53], sendo a versão 2.7.17 para execução do algoritmo de detecção de veículos e a versão 3.7.13 para execução dos demais algoritmos dos experimentos.
3. Bibliotecas: Em conjunto, utilizam-se as seguintes bibliotecas: OpenCV [54] na versão 4.1.2.30, Keras [55] na versão 2.2.4, Tensorflow [35] na versão 1.5.0 para execução do algoritmo de detecção de placas veiculares e na versão 2.4.1 para execução dos demais algoritmos dos experimentos e Numpy [56] versão 1.9.5.

### 5.3 Métricas de desempenho

Consideram-se, para avaliação do desempenho dos sistemas propostos, as métricas IoU e acurácia. A métrica IoU é definida conforme Equação (5.1) [46].

$$IoU_P^{GT} = \frac{\cap(b_{GT}, b_P)}{\cup(b_{GT}, b_P)}, \quad (5.1)$$

onde  $\cap(b_{GT}, b_P)$  é a interseção entre os rótulos manuais e detectados e  $\cup(b_{GT}, b_P)$  é a união entre os rótulos manuais e detectados.

A métrica acurácia é definida conforme Equação (5.2) [21].

$$Acurácia = \frac{TP + TN}{N}, \quad (5.2)$$

onde  $TP$  é o verdadeiro-positivo,  $TN$  é o verdadeiro-negativo e  $N$  é o número total de imagens.

## 5.4 Sistema de detecção de placa de licenciamento veicular sem super-resolução

O procedimento experimental do sistema proposto para detecção de placa de licenciamento veicular sem super-resolução (*baseline*), apresentado na Seção 3.2, é descrito, conforme a seguir:

1. Nesta etapa, o sistema processa cada imagem de entrada RGB. A base de dados de testes é composta por 1600 imagens, sendo 350 imagens em resolução  $1280 \times 720$  pixels e 1250 imagens em  $1920 \times 1080$  pixels. No Capítulo 4, apresenta-se a descrição detalhada da base de dados processada por este sistema.
2. Em seguida, executa-se o bloco Detector de Veículo YOLOv2 para detecção de veículos a partir da adaptação do sistema proposto pelos autores Silva e Jung. Emprega-se este bloco por sua velocidade de processamento e pelo suporte às classes *car* e *bus* no modelo pré-treinado baseado no PASCAL-VOC *dataset*. Adicionalmente, executa-se um modelo pré-treinado baseado no COCO *dataset*. Utilizam-se os seguintes parâmetros nos testes, com seus valores padrão: *vehicle\_threshold* = 0,5, *batch* = 1 e *subdivisions* = 1. Ao final desta execução, retornam-se a imagem do veículo detectado, a partir da imagem de entrada, e suas coordenadas de localização em pixels armazenadas em arquivo de texto.
3. Na terceira etapa, executa-se o bloco Seleção de Único Veículo por Frame, considerando a métrica IoU. Nesta, compara-se o rótulo manual com cada rótulo detectado do veículo presente na imagem e seleciona-se aquele que apresenta maior IoU. Para isso, emprega-se o limiar  $IoU \geq 0,5$ .
4. Na quarta etapa, executa-se o bloco Detector VLP WPOD-NET para detecção de placas veiculares. Utiliza-se o seguinte parâmetro, com seu valor padrão: *lp\_threshold* = 0,5. Emprega-se este bloco por reconhecer placas veiculares de diferentes *layouts*, um cenário comum em ambientes complexos. Ao final desta execução, retornam-se as coordenadas de localização em pixels da placa detectada, a partir da imagem do veículo detectado anteriormente.

5. No final deste experimento, a saída do sistema são o veículo e a placa veicular detectada, bem como as métricas de avaliação de desempenho (i.e., matriz de confusão e acurácia), considerando as rotulações manuais e detectadas dos dois objetos de interesse.

## Análise dos resultados

Na Tabela 5.1, apresentam-se os resultados obtidos. O desempenho deste sistema, em termos de acurácia, é de 94,50% para detecção de veículos <sup>1</sup> e 52,56% para detecção de placas veiculares. Considerando que uma quantidade maior de veículos detectados possibilita que o sistema processe mais placas veiculares, esta discrepância entre os valores de acurácia das detecções de veículos e de placas veiculares existe em razão da dificuldade do bloco Detector de VLP WPOD-NET identificar e localizar o objeto de interesse, que está em baixa resolução.

Tabela 5.1: Resultados do sistema para detecção de placa de licenciamento veicular sem super-resolução (*baseline*). O sistema completo está detalhado na Seção 3.2.

Métricas do Detector de Veículo				Métricas do Detector de Placa					
Matriz de Confusão				Acurácia	Matriz de Confusão				Acurácia
TN	FP	FN	TP		TN	FP	FN	TP	
0	0	88	1512	94,50%	0	0	759	841	52,56%

Adicionalmente, divide-se a base de dados de testes em sete grupos. Cada grupo de imagens contém uma condição climática adversa predominante, sendo: ensolarada, sombreada, nublada ou noturna. Na Tabela 5.2, apresentam-se os resultados obtidos considerando esta especificidade. Os grupos de imagens com melhor desempenho, quando processados pelo sistema, são:  $G3$ ,  $G2$  e  $G1$ , em ordem. Para os grupos de imagens  $G1$  e  $G2$ , que correspondem à condição climática adversa ensolarada, a acurácia para detecção de placa veicular é de 75,67% (com  $TP = 227$  e  $N = 300$ ) e 86,33% (com  $TP = 259$  e  $N = 300$ ), respectivamente. Para o grupo de imagens  $G3$ , que corresponde à condição climática adversa sombreada, a acurácia para detecção de placa veicular é de 98,33%, sendo  $TP = 295$  e  $N = 300$ .

No sistema proposto, os grupos de imagens, que correspondem às condições

<sup>1</sup>Os resultados obtidos para a detecção de veículos não são verdadeiros (apenas os de placas veiculares), uma vez que tem viés e por ter apenas um veículo anotado por frame na base de dados utilizada.

nubladas ( $G4$  e  $G5$ ) e noturnas ( $G6$  e  $G7$ ), não possuem bons resultados para detecção de placas veiculares, apresentando a acurácia de 18,00% (com  $TP = 54$  e  $N = 300$ ) como melhor resultado. Em relação aos grupos com condições noturnas, observa-se que este sistema não possui detecções positivas, apresentando uma acurácia de 0,00% (com  $TP = 0$  e  $N = 50$ ).

Portanto, neste sistema, o desempenho global para detecção de placas veiculares é, em termos de acurácia, de 52,56%. Para os grupos de imagens  $G1$ ,  $G2$  e  $G3$ , o desempenho, em termos de acurácia, é de 75,67%, 86,33% e 98,33%, respectivamente. Estes grupos de imagens apresentam resultados superiores em relação aos demais grupos de imagens. Para os grupos de imagens  $G4$  até  $G7$ , não se verificam detecções positivas para os grupos de imagens noturnas e o grupo de imagem que apresenta o melhor resultado é o  $G5$ , com 18,00% de acurácia. Destaca-se que as imagens do grupo  $G7$  não são adquiridas por VANT, tratando-se de imagens adquiridas por câmera CCTV fixa.

Tabela 5.2: Resultados do sistema para detecção de placa de licenciamento veicular sem super-resolução (*baseline*) por grupo de imagens.

Grupo	Condição	Métricas do Detetor de Veículo					Métricas do Detetor de Placa				
		TN	FP	FN	TP	Acurácia	TN	FP	FN	TP	Acurácia
Grupo 1	Ensolarado	0	0	2	298	99,33%	0	0	73	227	75,67%
Grupo 2	Ensolarado	0	0	1	299	99,67%	0	0	41	259	86,33%
Grupo 3	Sombreado	0	0	1	299	99,67%	0	0	5	295	98,33%
Grupo 4	Nublado	0	0	3	297	99,00%	0	0	294	6	2,00%
Grupo 5	Nublado	0	0	46	254	84,67%	0	0	246	54	18,00%
Grupo 6	Noturno	0	0	27	23	46,00%	0	0	50	0	0,00%
Grupo 7	Noturno	0	0	8	42	84,00%	0	0	50	0	0,00%

## 5.5 Sistema de detecção de placa de licenciamento veicular com super-resolução

O procedimento experimental do sistema proposto para detecção de placa de licenciamento veicular com super-resolução, apresentado na Seção 3.3, é descrito, conforme a seguir:

1. Nesta etapa, o sistema processa cada imagem de entrada RGB. A base de dados de testes é composta por 1600 imagens, sendo 350 imagens em resolução  $1280 \times$



720 pixels e 1250 imagens em  $1920 \times 1080$  pixels. No Capítulo 4, apresenta-se a descrição detalhada da base de dados processada por este sistema.

2. Em seguida, executa-se o bloco Detector de Veículo YOLOv2 para detecção de veículos a partir da adaptação do sistema proposto pelos autores Silva e Jung. Emprega-se este bloco por sua velocidade de processamento e pelo suporte às classes *car* e *bus* no modelo pré-treinado baseado no PASCAL-VOC *dataset*. Adicionalmente, executa-se um modelo pré-treinado baseado no COCO *dataset*. Utilizam-se os seguintes parâmetros nos testes, com seus valores padrão: *vehicle\_threshold* = 0,5, *batch* = 1 e *subdivisions* = 1. Ao final desta execução, retornam-se a imagem do veículo detectado, a partir da imagem de entrada, e suas coordenadas de localização em pixels armazenadas em arquivo de texto.
3. Na terceira etapa, executa-se o bloco Seleção de Único Veículo por Frame, considerando a métrica IoU. Nesta, compara-se o rótulo manual com cada rótulo detectado do veículo presente na imagem e seleciona-se aquele que apresenta maior IoU. Para isso, emprega-se o limiar  $IoU \geq 0,5$ .
4. Na quarta etapa, executa-se o bloco Super-resolução de Imagem por GAN, a partir da adaptação do sistema proposto pelo autor Raza, para super-resolver a imagem do veículo detectado, que está em baixa resolução. Emprega-se este bloco por sua velocidade de processamento e pela capacidade de super-resolver imagens com a preservação das características originais da cena. Utiliza-se, nos testes, o seguinte parâmetro com valor  $new\_input\_shape = [None, None, 3]$  para permitir que o modelo pré-treinado receba imagens de diferentes resoluções. Ao final desta execução, retorna-se a imagem super-resolvida do veículo detectado, com fator de escala  $4\times$ .
5. Na quinta etapa, executa-se o bloco Detector VLP WPOD-NET para detecção de placas veiculares. Utiliza-se o seguinte parâmetro, com seu valor padrão:  $lp\_threshold = 0,5$ . Emprega-se este bloco por reconhecer placas veiculares de diferentes *layouts*, um cenário comum em ambientes complexos. Ao final desta execução, retornam-se as coordenadas de localização em pixels da placa detectada, a partir da imagem do veículo detectado anteriormente.

6. Na sexta etapa, executa-se o bloco Redimensionamento da Imagem para reduzir as dimensões das coordenadas de localização da placa detectada. Emprega-se este bloco para manter as coordenadas de localização do veículo e da placa veicular detectada nas mesmas proporções durante a avaliação do desempenho do sistema proposto.
7. No final deste experimento, a saída do sistema são o veículo e a placa veicular detectada, bem como as métricas de avaliação de desempenho (i.e., matriz de confusão e acurácia), considerando as rotulações manuais e detectadas dos dois objetos de interesse.

## Análise dos resultados

Na Tabela 5.3, apresentam-se os resultados obtidos. O desempenho deste sistema para detecção de veículos é igual ao do sistema de referência. Para a detecção de placas veiculares, o desempenho deste sistema é, em termos de acurácia, de 51,25%. Para a base de dados de testes utilizada no experimento deste sistema, os resultados são inferiores à *baseline*, em razão do baixo contraste das imagens dos veículos detectados.

Tabela 5.3: Resultados do sistema para detecção de placa de licenciamento veicular com super-resolução. O sistema completo está detalhado na Seção 3.3.

Métricas do Detector de Veículo				Métricas do Detector de Placa					
Matriz de Confusão				Acurácia	Matriz de Confusão				Acurácia
TN	FP	FN	TP		TN	FP	FN	TP	
0	0	88	1512	94,50%	0	0	780	820	51,25%

Na Tabela 5.4, apresentam-se os resultados obtidos por grupo de imagens. Neste sistema, os grupos de imagens apresentam um decréscimo no desempenho para detecção de placas veiculares, quando processados pelo bloco Super-resolução de imagem por GAN. Os grupos de imagens com melhor desempenho, quando processados pelo sistema, são:  $G3$ ,  $G2$  e  $G1$ , em ordem. Para os grupos de imagens  $G1$  e  $G2$ , que correspondem à condição climática adversa ensolarada, a acurácia para detecção de placa veicular é de 75,00% (com  $TP = 225$  e  $N = 300$ ) e 84,00% (com  $TP = 252$  e  $N = 300$ ), respectivamente. Para o grupo de imagens  $G3$ , que

corresponde à condição climática adversa sombreada, a acurácia para detecção de placa veicular é de 97,33%, sendo  $TP = 292$  e  $N = 300$ .

No sistema proposto, os grupos de imagens, que correspondem às condições nubladas ( $G4$  e  $G5$ ) e noturnas ( $G6$  e  $G7$ ), não possuem bons resultados para detecção de placas veiculares, apresentando a acurácia de 15,00% (com  $TP = 45$  e  $N = 300$ ) como melhor resultado. Em relação aos grupos com condições noturnas, observa-se que este sistema não possui detecções positivas, apresentando uma acurácia de 0,00% (com  $TP = 0$  e  $N = 50$ ).

Portanto, neste sistema, o desempenho global para detecção de placas veiculares é, em termos de acurácia, de 51,25%. Para os grupos de imagens  $G1$ ,  $G2$  e  $G3$ , o desempenho, em termos de acurácia, é de 74,00%, 84,00% e 97,33%, respectivamente. Estes grupos de imagens apresentam resultados superiores em relação aos demais grupos de imagens. Para os grupos de imagens  $G4$  até  $G7$ , não se verificam detecções positivas para os grupos de imagens noturnas e o grupo de imagem que apresenta o melhor resultado é o  $G5$ , com 15,00% de acurácia.

Tabela 5.4: Resultados do sistema para detecção de placa de licenciamento veicular com super-resolução por grupo de imagens.

Grupo	Condição	Métricas do Detector de Veículo					Métricas do Detector de Placa				
		TN	FP	FN	TP	Acurácia	TN	FP	FN	TP	Acurácia
Grupo 1	Ensolarado	0	0	2	298	99,33%	0	0	75	225	75,00%
Grupo 2	Ensolarado	0	0	1	299	99,67%	0	0	48	252	84,00%
Grupo 3	Sombreado	0	0	1	299	99,67%	0	0	8	292	97,33%
Grupo 4	Nublado	0	0	3	297	99,00%	0	0	294	6	2,00%
Grupo 5	Nublado	0	0	46	254	84,67%	0	0	255	45	15,00%
Grupo 6	Noturno	0	0	27	23	46,00%	0	0	50	0	0,00%
Grupo 7	Noturno	0	0	8	42	84,00%	0	0	50	0	0,00%

## 5.6 Sistema de detecção de placa de licenciamento veicular com equalização de histograma

O procedimento experimental do sistema proposto para detecção de placa de licenciamento veicular com equalização de histograma, apresentado na Seção 3.4, é descrito, conforme a seguir:

1. Nesta etapa, o sistema processa cada imagem de entrada RGB. A base de dados de testes é composta por 1600 imagens, sendo 350 imagens em resolução  $1280 \times$

720 pixels e 1250 imagens em  $1920 \times 1080$  pixels. No Capítulo 4, apresenta-se a descrição detalhada da base de dados processada por este sistema.

2. Em seguida, executa-se o bloco Detector de Veículo YOLOv2 para detecção de veículos a partir da adaptação do sistema proposto pelos autores Silva e Jung. Emprega-se este bloco por sua velocidade de processamento e pelo suporte às classes *car* e *bus* no modelo pré-treinado baseado no PASCAL-VOC *dataset*. Adicionalmente, executa-se um modelo pré-treinado baseado no COCO *dataset*. Utilizam-se os seguintes parâmetros nos testes, com seus valores padrão: *vehicle\_threshold* = 0,5, *batch* = 1 e *subdivisions* = 1. Ao final desta execução, retornam-se a imagem do veículo detectado, a partir da imagem de entrada, e suas coordenadas de localização em pixels armazenadas em arquivo de texto.
3. Na terceira etapa, executa-se o bloco Seleção de Único Veículo por Frame, considerando a métrica IoU. Nesta, compara-se o rótulo manual com cada rótulo detectado do veículo presente na imagem e seleciona-se aquele que apresenta maior IoU. Para isso, emprega-se o limiar  $IoU \geq 0,5$ .
4. Na quarta etapa, executa-se o bloco Super-resolução de Imagem por GAN, a partir da adaptação do sistema proposto pelo autor Raza, para super-resolver a imagem do veículo detectado, que está em baixa resolução. Emprega-se este bloco por sua velocidade de processamento e pela capacidade de super-resolver imagens com a preservação das características originais da cena. Utiliza-se, nos testes, o seguinte parâmetro com valor  $new\_input\_shape = [None, None, 3]$  para permitir que o modelo pré-treinado receba imagens de diferentes resoluções. Ao final desta execução, retorna-se a imagem super-resolvida do veículo detectado, com fator de escala  $4\times$ .
5. Na quinta etapa, executa-se o bloco Equalização de Histograma CLAHE, apresentado pelo autor Reza, na imagem super-resolvida do veículo detectado. Emprega-se este bloco para aprimorar a saída do bloco Super-resolução de imagem por GAN do sistema proposto na Seção 3.3, devido à existência de falsos-negativos durante a execução do bloco Detector VLP WPOD-NET. Utilizam-se, nos testes, os seguintes valores de parâmetros: *clipLimit* = 2,0

e  $tileGridSize = (8, 8)$ . Ao final desta execução, retorna-se a imagem equalizada do veículo detectado.

6. Na sexta etapa, executa-se o bloco Detector VLP WPOD-NET para detecção de placas veiculares. Utiliza-se o seguinte parâmetro, com seu valor padrão:  $lp\_threshold = 0,5$ . Emprega-se este bloco por reconhecer placas veiculares de diferentes *layouts*, um cenário comum em ambientes complexos. Ao final desta execução, retornam-se as coordenadas de localização em pixels da placa detectada, a partir da imagem do veículo detectado anteriormente.
7. Na sétima etapa, executa-se o bloco Redimensionamento da Imagem para reduzir as dimensões das coordenadas de localização da placa detectada. Emprega-se este bloco para manter as coordenadas de localização do veículo e da placa veicular detectada nas mesmas proporções durante a avaliação do desempenho do sistema proposto.
8. No final deste experimento, a saída do sistema são o veículo e a placa veicular detectada, bem como as métricas de avaliação de desempenho (i.e., matriz de confusão e acurácia), considerando as rotulações manuais e detectadas dos dois objetos de interesse.

## Análise dos resultados

Na Tabela 5.5, apresentam-se os resultados obtidos. O desempenho deste sistema para detecção de veículos é igual ao dos sistemas anteriores. Para a detecção de placas veiculares, o desempenho deste sistema é, em termos de acurácia, de 54,81%. Conforme Tabelas 5.1, 5.3 e 5.5, os resultados deste sistema são superiores aos demais sistemas propostos, em razão da equalização do histograma das imagens super-resolvidas dos veículos detectados. A aplicação deste método de pré-processamento possibilita a diferenciação do objeto de interesse em relação ao fundo a partir da distribuição dos valores de intensidade presentes nos pixels da imagem super-resolvida.

Na Tabela 5.6, apresentam-se os resultados obtidos por grupo de imagens. Os grupos de imagens com melhor desempenho, quando processados pelo sistema, são:  $G3$ ,  $G1$  e  $G2$ , em ordem. Para os grupos de imagens  $G1$  e  $G2$ , que correspondem

Tabela 5.5: Resultados do sistema para detecção de placa de licenciamento veicular com equalização de histograma. O sistema completo está detalhado na Seção 3.4.

Métricas do Detector de Veículo				Métricas do Detector de Placa					
Matriz de Confusão				Acurácia	Matriz de Confusão				Acurácia
TN	FP	FN	TP		TN	FP	FN	TP	
0	0	88	1512	94,50%	0	0	723	877	54,81%

à condição climática adversa ensolarada, a acurácia para detecção de placa veicular é de 85,67% (com  $TP = 257$  e  $N = 300$ ) e 81,33% (com  $TP = 244$  e  $N = 300$ ), respectivamente. Para o grupo de imagens  $G3$ , que corresponde à condição climática adversa sombreada, a acurácia para detecção de placa veicular é de 99,33%, sendo  $TP = 298$  e  $N = 300$ .

No sistema proposto, os grupos de imagens, que correspondem às condições nubladas ( $G4$  e  $G5$ ) e noturnas ( $G6$  e  $G7$ ), não possuem bons resultados para detecção de placas veiculares, apresentando a acurácia de 21,00% (com  $TP = 63$  e  $N = 300$ ) como melhor resultado. Em relação aos grupos com condições noturnas, observa-se que este sistema não possui detecções positivas, apresentando uma acurácia de 0,00% (com  $TP = 0$  e  $N = 50$ ).

Portanto, neste sistema, o desempenho global para detecção de placas veiculares é, em termos de acurácia, de 54,81%. Para os grupos de imagens  $G1$ ,  $G2$  e  $G3$ , o desempenho, em termos de acurácia, é de 85,67%, 81,33% e 99,33%, respectivamente. Estes grupos de imagens apresentam resultados superiores em relação aos demais grupos de imagens. Para os grupos de imagens  $G4$  até  $G7$ , não se verificam detecções positivas para os grupos de imagens noturnas e o grupo de imagem que apresenta o melhor resultado é o  $G5$ , com 21,00% de acurácia.

Tabela 5.6: Resultados do sistema para detecção de placa de licenciamento veicular com equalização de histograma por grupo de imagens.

Grupo	Condição	Métricas do Detector de Veículo					Métricas do Detector de Placa				
		TN	FP	FN	TP	Acurácia	TN	FP	FN	TP	Acurácia
Grupo 1	Ensolarado	0	0	2	298	99,33%	0	0	43	257	85,67%
Grupo 2	Ensolarado	0	0	1	299	99,67%	0	0	56	244	81,33%
Grupo 3	Sombreado	0	0	1	299	99,67%	0	0	2	298	99,33%
Grupo 4	Nublado	0	0	3	297	99,00%	0	0	285	15	5,00%
Grupo 5	Nublado	0	0	46	254	84,67%	0	0	237	63	21,00%
Grupo 6	Noturno	0	0	27	23	46,00%	0	0	50	0	0,00%
Grupo 7	Noturno	0	0	8	42	84,00%	0	0	50	0	0,00%

## 5.7 Comparação dos resultados dos sistemas propostos

Na Tabelas 5.7 e 5.8, apresenta-se uma comparação resumida entre os três sistemas propostos neste trabalho de pesquisa, considerando a utilização dos modelos pré-treinados YOLOv2 e YOLOv3. Para o emprego do modelo pré-treinado YOLOv3, adapta-se o bloco Detector de Veículo YOLOv2. A coluna *variação* mostra a diferença dos resultados, em termos de acurácia, em relação à *baseline*. O modelo pré-treinado YOLOv2 apresenta um desempenho superior ao YOLOv3<sup>2</sup>, quando analisam-se os resultados por grupo de imagens.

Tabela 5.7: Comparação dos resultados obtidos entre os três sistemas propostos, considerando o modelo pré-treinado YOLOv2 para execução do bloco Detector de Veículo YOLOv2.

Grupo	Condição	Métricas do Detector de Veículo YOLOv2	Métricas do Detector de Placa WPOD-NET ( <i>Baseline</i> )	Métricas do Detector de Placa WPOD-NET (Adição de SR)		Métricas do Detector de Placa WPOD-NET (Adição de SR+CLAHE)	
		Acurácia	Acurácia	Varição	Acurácia	Varição	Acurácia
Grupo 1	Ensolarado	99,33%	75,67%	-0,67%	75,00%	+10,00%	85,67%
Grupo 2	Ensolarado	99,67%	86,33%	-2,33%	84,00%	-5,00%	81,33%
Grupo 3	Sombreado	99,67%	98,33%	-1,00%	97,33%	+1,00%	99,33%
Grupo 4	Nublado	99,00%	2,00%	0,00%	2,00%	+3,00%	5,00%
Grupo 5	Nublado	84,67%	18,00%	-3,00%	15,00%	+3,00%	21,00%
Grupo 6	Noturno	46,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Grupo 7	Noturno	84,00%	0,00%	0,00%	0,00%	0,00%	0,00%

Tabela 5.8: Comparação dos resultados obtidos entre os três sistemas propostos, considerando o modelo pré-treinado YOLOv3 para execução da adaptação do bloco Detector de Veículo YOLOv2.

Grupo	Condição	Métricas do Detector de Veículo YOLOv3	Métricas do Detector de Placa WPOD-NET ( <i>Baseline</i> )	Métricas do Detector de Placa WPOD-NET (Adição de SR)		Métricas do Detector de Placa WPOD-NET (Adição de SR+CLAHE)	
		Acurácia	Acurácia	Varição	Acurácia	Varição	Acurácia
Grupo 1	Ensolarado	95,67%	69,67%	+1,00%	70,67%	+13,66%	83,33%
Grupo 2	Ensolarado	89,00%	59,00%	0,00%	59,00%	-1,00%	58,00%
Grupo 3	Sombreado	100,00%	88,00%	-0,33%	87,67%	0,00%	88,00%
Grupo 4	Nublado	99,67%	1,00%	-0,33%	0,67%	2,33%	3,33%
Grupo 5	Nublado	82,33%	12,67%	-3,00%	9,67%	-2,00%	10,67%
Grupo 6	Noturno	100,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Grupo 7	Noturno	78,00%	0,00%	0,00%	0,00%	0,00%	0,00%

<sup>2</sup>O modelo pré-treinado YOLOv3 possui 74 camadas convolucionais. Utiliza-se para detecção de objetos pequenos, apresentando uma precisão de 57,9% mAP no COCO *dataset* [39].

# Capítulo 6

## Conclusão

Nesta dissertação, propõe-se uma metodologia para detecção de placas veiculares de baixa resolução em pixel, a partir de vídeos que contenham imagens aéreas. Como sistema de referência (*baseline*), utilizam-se as arquiteturas YOLOv2 e YOLOv3, para detecção de veículos, e WPOD-NET, para detecção de placas veiculares. No sistema proposto com super-resolução, utiliza-se a arquitetura GAN para super-resolver a imagem do veículo detectado. No sistema proposto com equalização de histograma, utiliza-se o método CLAHE para o realce do contraste da imagem super-resolvida. Estes dois últimos sistemas aplicam os métodos de super-resolução e equalização de histograma para a detecção do objeto de interesse.

Uma base de dados de testes é organizada para a avaliação do desempenho da metodologia proposta nesta dissertação. Apresenta-se o processo de organização, conforme a seguir. Inicialmente, realiza-se uma pesquisa por vídeos na plataforma YouTube, com base em critérios pré-estabelecidos para seleção. Dentre os critérios, destacam-se vídeos adquiridos em períodos diurnos e noturnos, com elevada e baixa iluminação, em diferentes condições climáticas (i.e., ambiente ensolarado e nublado), fonte de aquisição parada e em movimento, com objeto de interesse se aproximando ou se distanciando. No final deste processo, selecionam-se 7 vídeos, totalizando 1600 imagens RGB, após o processo de conversão em frames.

Como resultados obtidos, o sistema de referência (*baseline*) apresenta uma acurácia de 52,56%, enquanto que os sistemas com super-resolução e com equalização de histograma apresentam, respectivamente, uma precisão de 51,25% e 54,81%. Pode-se concluir que o sistema proposto para detecção de placa de licenciamento



veicular com equalização de histograma apresenta o melhor desempenho, em termos de acurácia. Ao analisar os resultados obtidos por grupo de imagens, destaca-se que o sistema proposto para detecção de placa de licenciamento veicular com equalização de histograma apresenta o melhor resultado para os grupos  $G1$  até  $G3$  (cujas condições climáticas são ensolaradas e sombreadas), com acurácia de 85,67% , 81,33% e 99,33%, respectivamente. Outro ponto observado é a ausência de detecções positivas para o grupo de imagens noturnas (i.e., os grupos  $G6$  e  $G7$ ), demonstrando que a baixa iluminação afeta a detecção da placa veicular, mesmo com o realce do contraste da imagem.

Portanto, a partir da análise dos resultados, conclui-se que o sistema proposto para detecção de placa de licenciamento veicular com equalização de histograma CLAHE é uma boa opção para melhoria de desempenho de modelos pré-treinados em ambientes aéreos, pois, na ausência de modelos de redes neurais treinados para estes cenários específicos, pode-se explorá-lo em diferentes aplicações.

## 6.1 Proposta para Trabalhos Futuros

Nesta seção, apresentam-se algumas propostas para trabalhos futuros relacionados ao estudo de caso descrito nesta dissertação. Estas propostas estão elencadas a seguir:

- Investigar outros métodos para detecção de placa veicular, como, por exemplo, o *contrastive language-image pre-training* (CLIP), que pode ser empregado tanto para a técnica de super-resolução, quanto para a tarefa de detecção de objetos. Além de permitir o treinamento de redes neurais sem a necessidade de grande volume de dados, este método apresenta melhores resultados, quando comparado a outras arquiteturas;
- Empregar o método de consistência temporal e geométrica, processando imagens no domínio do tempo para combinar os resultados anteriores e posteriores à imagem observada, visando uma detecção mais robusta para cada placa veicular;
- Desenvolver dois novos sistemas para detecção de placa de licenciamento veicu-

lar, sendo um que não utiliza super-resolução com equalização de histograma e outro que utiliza equalização de histograma anterior à super-resolução da imagem do veículo detectado, com o intuito de analisar o comportamento em relação à base de dados de testes organizada nesta dissertação.

# Referências Bibliográficas

- [1] DU, D., OTHERS, “The unmanned aerial vehicle benchmark: Object detection and tracking”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 370–386, 2018.
- [2] SARKAR, S., TOTARO, M. W., ELGAZZAR, K., “Intelligent Drone-based Surveillance: Application to Parking Lot Monitoring and Detection”. In: *Proceedings of the Unmanned Systems Technology*, p. 1102104, 2019.
- [3] QUAN, A., HERRMANN, C., SOLIMAN, H., “Project vulture: A prototype for using drones in search and rescue operations”. In: *Proceedings of the International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pp. 619–624, IEEE, 2019.
- [4] TONG, K., WU, Y., ZHOU, F., “Recent advances in small object detection based on deep learning: A review”. In: *Proceedings of the Image and Vision Computing*, p. 103910, 2020.
- [5] REDOMN, J., OTHERS, “You only look once: Unified, real-time object detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.
- [6] BAI, Y., OTHERS, “SOD-MTGAN: Small Object Detection via Multi-Task Generative Adversarial Network”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 206–221, 2018.
- [7] LIU, Y., OTHERS, “Performance comparison of deep learning techniques for recognizing birds in aerial images”. In: *Proceedings of the IEEE International Conference on Data Science in Cyberspace (DSC)*, pp. 317–324, 2020.

- [8] LEDIG, C., OTHERS, “Photo-realistic single image super-resolution using a generative adversarial network”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4681–4690, 2017.
- [9] KIM, J., OTHERS, “Vehicle model recognition using SRGAN for low-resolution vehicle images”. In: *Proceedings of the International Conference on Artificial Intelligence and Pattern Recognition*, pp. 42–45, 2019.
- [10] LEE, Y., OTHERS, “Accurate license plate recognition and super-resolution using a generative adversarial networks on traffic surveillance video”. In: *Proceedings of the IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, pp. 1–4, 2018.
- [11] SILVA, S. M., JUNG, C. R., “License plate detection and recognition in unconstrained scenarios”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 580–596, 2018.
- [12] RABBI, J., OTHERS, “Small-object detection in remote sensing images with end-to-end edge-enhanced GAN and object detector network”. In: *Proceedings of the Remote Sensing*, p. 1432, 2020.
- [13] KHAZAEI, S., OTHERS, “A Real-Time License Plate Detection Method Using a Deep Learning Approach”. In: *Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence*, pp. 425–438, Springer, 2020.
- [14] SOLOMON, C., BRECKON, T., *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*. Wiley-Blackwell, 2010.
- [15] JAYNES, C., “View alignment of aerial and terrestrial imagery in urban environments”. In: *Proceedings of the International Workshop on Integrated Spatial Databases*, pp. 3–19, 1999.
- [16] DING, J., OTHERS, “Learning roi transformer for oriented object detection in aerial images”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2849–2858, 2019.

- [17] GONZALEZ, R. C., WOODS, R. E., *Processamento Digital de Imagens*. 3rd ed. Prentice Hall, 2008.
- [18] REZA, A. M., “Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement”. In: *Proceedings of journal signal processing systems for signal, image and video technology*, pp. 35–44, VLSI, 2004.
- [19] AGGARWAL, C., *Neural Networks and Deep Learning: A Textbook*. Springer, 2018.
- [20] ZHAO, Z. Q., OTHERS, “Object detection with deep learning: A review”. In: *Proceedings of the Transactions on Neural Networks and Learning Systems*, pp. 3212–3232, IEEE, 2019.
- [21] MICHELUCCI, U., *Applied Deep Learning: A Case-Based Approach to Understanding Deep Neural Networks*. Apress Media, 2018.
- [22] KETKAR, N., *Deep Learning with Python: A Hands-on Introduction*. Apress Media, 2017.
- [23] HE, K., OTHERS, “Deep residual learning for image recognition”. In: *Proceedings of Conference on Computer Vision and Pattern Recognition*, pp. 770–778, IEEE, 2016.
- [24] GOODFELLOW, I., OTHERS, “Generative adversarial nets”. In: *Proceedings of the Advances in Neural Information Processing Systems*, 2014.
- [25] ALBAWI, S., MOHAMMED, T. A., AL-ZAWI, S., “Understanding of a convolutional neural network”. In: *Proceedings of the International Conference on Engineering and Technology (ICET)*, pp. 1–6, 2017.
- [26] REDMON, J., FARHADI, A., “YOLO9000: better, faster, stronger”. In: *Proceedings of Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, IEEE, 2017.
- [27] DUMITRESCU, D., BOIANGIU, C., “A study of image upsampling and downsampling filters”. In: *Proceedings of the Computers*, p. 30, 2019.

- [28] DONG, C., OTHERS, “Image super-resolution using deep convolutional networks”. In: *Proceedings of transactions on pattern analysis and machine intelligence*, pp. 295–307, IEEE, 2015.
- [29] SHI, W., OTHERS, “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network”. In: *Proceedings of Conference on Computer Vision and Pattern Recognition*, pp. 1874–1883, IEEE, 2016.
- [30] MEMON, J., OTHERS, “Handwritten optical character recognition (OCR): A comprehensive systematic literature review (SLR)”. In: *IEEE Access*, pp. 142642–142668, 2020.
- [31] REDMON, J., “Darknet: Open Source Neural Networks in C”, <http://pjreddie.com/darknet/>, 2013–2016.
- [32] EVERINGHAM, M., OTHERS, “The pascal visual object classes (voc) challenge”. In: *Proceedings of International journal of computer vision*, pp. 303–338, 2010.
- [33] LIU, W., OTHERS, “Ssd: Single shot multibox detector”. In: *Proceedings of the European Conference on Computer Vision*, pp. 21–37, 2016.
- [34] JADERBERG, M., OTHERS, “Spatial transformer networks”. In: *Proceedings of the Advances in Neural Information Processing Systems*, pp. 2017–2025, 2015.
- [35] ABADI, M., OTHERS, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems”, <http://tensorflow.org/>, 2015.
- [36] OPENALPR, “Benchmarks”, <https://github.com/openalpr/benchmarks>, 2014.
- [37] GONÇALVES, G. R., OTHERS, “Benchmark for license plate character segmentation”. In: *Proceedings of the Journal of Electronic Imaging*, p. 053034, IEEE, 2016.

- [38] HSU, G. S., CHEN, J. C., CHUNG, Y. Z., “Application-oriented license plate recognition”. In: *Proceedings of the transactions on vehicular technology*, pp. 552–561, IEEE, 2012.
- [39] REDMON, J., FARHADI, A., “Yolov3: An incremental improvement”. In: *arXiv preprint arXiv:1804.02767*, 2018.
- [40] LEVER, J., KRZYWINSKI, M., ALTMAN, N., “Classification evaluation”, <https://doi.org/10.1038/nmeth.3945>, 2016.
- [41] GIRSHICK, R., “Fast r-cnn”. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, 2015.
- [42] MUNDHENK, T. N., OTHERS, “A large contextual dataset for classification, detection and counting of cars with deep learning”. In: *Proceedings of the European Conference on Computer Vision*, pp. 785–800, 2016.
- [43] RABBI, J., CHOWDHURY, S., CHAO, D., “Oil and Gas Tank Dataset”, doi:10.17632/bkxj8z84m9.3, 2020.
- [44] HAZA, R., “Fast-SRGAN”, <https://github.com/HasnainRaz/Fast-SRGAN>, 2019.
- [45] LIN, T., “LabelImg”, <https://github.com/tzutalin/labelImg>, 2015.
- [46] WU, Q., ZHOU, Y., “Real-time object detection based on unmanned aerial vehicle”. In: *Proceedings of the Data Driven Control and Learning Systems Conference (DDCLS)*, pp. 574–579, IEEE, 2019.
- [47] KRAUSE, J., OTHERS, “3d object representations for fine-grained categorization”. In: *Proceedings of the international conference on computer vision workshops*, pp. 554–561, IEEE, 2013.
- [48] CORDERO, A. U., “A high performance terabyte-order RGB to HSV parallel conversion implementation”, 2015.
- [49] AGUSTSSON, E., TIMOFTE, R., “Ntire 2017 challenge on single image super-resolution: Dataset and study”. In: *Proceedings of the conference on*

*computer vision and pattern recognition workshops*, pp. 126–135, IEEE, 2017.

- [50] HURLEY, C., CHEN, S., KARIM, J., “YouTube”, <https://www.youtube.com/>, 2005.
- [51] AMAZON, “Amazon SageMaker Ground-Truth”, <https://aws.amazon.com/pt/sagemaker/>, 2018.
- [52] TEAM, C., “Google Colaboratory”, <https://colab.research.google.com/>, 2018.
- [53] TEAM, P. C., “Python: A dynamic, open source programming language”, <https://www.python.org/>, 2019.
- [54] BRADSKI, G., “The OpenCV Library”, *Dr. Dobb’s Journal of Software Tools*, 2000.
- [55] CHOLLET, F., OTHERS, “Keras”, <https://keras.io>, 2015.
- [56] HARRIS, C. R., OTHERS, “Array programming with NumPy”, *Nature*, pp. 357–362, 2020.
- [57] DREYFUS, G., *Neural Networks: Methodology and Applications*. Springer, 2005.