

Reconhecimento de Placas Veiculares em Cenários Complexos utilizando o Método do Subespaço Mútuo e Redes Neurais Convolucionais

Anderson S. Jesus^{1,4}, Luiz Carlos S. G. Júnior^{1,3}, José Elislande B. S. Linhares^{1,2}, Georgio S. Colares^{1,5}, Bernardo B. Gatto¹, Newton P. Bitar¹, Eulanda S. Miranda¹, Frederico S. Pinagé¹, Waldir S. S. Júnior¹

¹Federal University of Amazonas (UFAM), AM-Brazil

²Federal Institute of Amazonas (IFAM), AM-Brazil

³Conecthus Amazon Institute of Technology and Biotechnology, AM-Brazil

⁴Institute Center of Development Research in Software Technology (ICTS), AM-Brazil

⁵Sidia Institute of Science and Technology (Sidia/Samsung), AM-Brazil

Emails: anderson.souza@grupoicts.com.br, luiz.junior@conecthus.org.br, breno.linhares@ifam.edu.br, georgio.colares@sidia.com, newtonpbitar@gmail.com, {bernardo, emsantos}@icomp.ufam.edu.br, {fredericopinage, waldirjr}@ufam.edu.br

Resumo—Neste artigo, propõem-se dois sistemas para reconhecimento de placas veiculares. O primeiro sistema usa o método do subespaço mútuo e o segundo utiliza CNN. Os sistemas propostos são utilizados em ambientes complexos, caracterizado por veículos em movimento e com variação na iluminação e resolução. Para auxiliar no processo de identificação e localização da placa veicular na imagem são empregadas técnicas de pré-processamento e de detecção de veículos (YOLO). Os resultados obtidos foram promissores, apresentando como acurácia 56% para o método de subespaço mútuo e 99% para o método baseado em redes neurais.

Palavras-Chave—Detecção de placas veiculares, método do subespaço, CNN.

Abstract—In this paper, two complete systems for license plate recognition are proposed. The first system uses the mutual subspace method and the second uses CNN. The proposed systems are used in complex environments, characterized by moving vehicles and with variation in lighting and resolution. The systems use pre-processing and vehicle detection techniques (YOLO) to identify and locate the license plate in the image. The results obtained were promising, presenting an accuracy of 56% for the mutual subspace method and 99% for the method based on neural networks.

Keywords—Vehicle licence plate detection, subspace method, CNN.

I. INTRODUÇÃO

Segundo o Instituto Brasileiro de Geografia e Estatística, o número da frota de automóveis entre 2010 e 2021 aumentou em 37% [1]. O crescimento no número de automóveis causa uma série de adversidades, tais como aumento da criminalidade e acidentes de trânsito, e como consequência, a necessidade de melhorar os sistemas existentes de identificação de veículos. Este problema, tem encorajado o desenvolvimento de sistemas automáticos para auxiliar no controle de identificação de veículos [2].

A detecção automática de placas veiculares pode contribuir no controle de vários cenários como, por exemplo, violações de excesso de velocidade e cobrança de pedágio, além disso, pode auxiliar no monitoramento de estacionamentos e condomínios residenciais [3].

Recentemente, métodos de detecção e classificação de placas baseados em inteligência artificial foram concebidos, tendo resultados destacáveis em acurácia [4], [5], [6], porém, estes

métodos demandam mais tempo e poder de hardware do que técnicas tradicionais de visão computacional. Podemos concluir então que embora o reconhecimento automático de placas veiculares seja um desafio bastante explorado nos últimos anos, ainda necessita encontrar um equilíbrio entre baixo custo computacional e alta taxa de precisão.

Considerando-se as problemáticas apresentadas, dois sistemas para a detecção e reconhecimento de placas veiculares em ambientes complexos são propostos. O primeiro sistema utiliza CNN (*convolutional neural network*) que são basicamente compostos por camadas de convoluções e de *pooling* para extração de características, seguidos por camadas densas onde realiza-se a inferência do classificador. O segundo sistema utiliza o MSM (*mutual subspace method*) [7] que avalia a similaridade entre objetos através do cálculo da distância entre subespaços de múltiplos vetores.

O artigo está estruturado da seguinte forma: na Seção II, é apresentado os trabalhos relacionados que serviram como base para este artigo. Na Seção III, apresenta-se os sistemas propostos, descrevendo as etapas dos classificadores. Na seção IV descreve-se os procedimentos experimentais, detalhando as bases de dados empregadas, métricas utilizadas e os resultados obtidos. Por último, na Seção V, as conclusões.

A. Contribuições do artigo

A principal contribuição do artigo corresponde a utilização de dois sistemas para o reconhecimento de placas veiculares considerando-se ambientes complexos. Cabe destacar que os métodos de detecção de placas veiculares atuais utilizam conjuntos de imagens que possuem como características terem alta qualidade de resolução e ambiente controlado, o que resulta em uma ótima visibilidade da placa do veículo. Os sistemas propostos no entanto, contemplam ambientes complexos, ou seja, utilizam imagens com grande complexidade de detecção do objeto de interesse. São imagens adquiridas com a câmera e o veículo em movimento; com variações na iluminação e na resolução das imagens; e com *background* complexo.

II. TRABALHOS RELACIONADOS

No estado da arte, pode-se encontrar diversos trabalhos relacionados que abordam o LPR (*license plate recognition*),

como os mencionados na Tabela I. Dentre estes, a pesquisa apresentada por Neto E. C. *et al.* [8] propõe um sistema para reconhecimento de placas veiculares brasileiras cuja aplicação consiste em gerenciar o acesso de usuários em parqueamentos privados. O método proposto é composto por duas etapas: (i) extração de caracteres, em que se utilizou técnicas de DSP (*digital signal processing*) como detector de bordas de Canny e morfologia matemática para segmentação dos objetos de interesse; e (ii) identificação de algoritmos alfanuméricos, em que foram empregadas técnicas de detecção por aprendizagem a partir de arquiteturas como MLP (*multilayer perceptron*) e ELM (*extreme learning machine*), por exemplo. A acurácia do sistema final foi de 98,50%, a 50 FPS, testando-o em uma base de dados de 700 vídeos, com resolução de 640×480 pixels, no formato AVI.

No trabalho proposto por Laroça R. e Menotti D. [4], os autores apresentaram um sistema completo para reconhecimento automático de placas veiculares a partir de vídeos em tempo real, empregando-se o método de redundância temporal para viabilizar uma predição mais robusta. O sistema proposto é formado por duas fases: (i) treinamento, em que se treinou duas redes convolutivas (CNN), sendo uma para a detecção do objeto veículo e outra para a de placa; e (ii) testes, em que se utilizou uma rede convolutiva para segmentação e identificação dos caracteres da placa detectada. A acurácia do método proposto foi de 78,33%, a 35 FPS, utilizando-se a base de dados UFPR-ALPR composta de 150 vídeos capturados em movimento e contendo diferentes tipos de veículos na cena.

Um método de ALPR (*automatic license plate recognition*) baseado em DL é demonstrado no trabalho de Silva, S. M. e Jung C. R. [9]. O sistema contém duas redes baseada em YOLO, onde uma detecta *license plate*, e a outra em cascata reconhece caracteres nesta região. O conjunto de treinamento limitado é compensado utilizando de placas reais misturadas com caracteres sintéticos, aumentando a precisão dos resultados. A acurácia foi de 92,41%, com tempo de predição de 13ms utilizando GPU de alta performance. É aplicado técnica de coerência temporal para melhor reconhecer caracteres em vídeos, usando conjuntos de dados públicos (SSIG, UFPR-ALPR e OpenALPR) de *license plate*. Foi alcançado taxas de precisão de 89,15% para SSIG, 85,42% para UFPR-ALPR e 85,19% para OpenALPR.

No trabalho proposto por Xie F. *et al.* [10] é apresentado um algoritmo de detecção de *license plate* e reconhecimento de caracteres. Este trabalho utiliza a combinação de BPNN (*back-propagation neural network*) e extração de características. Na imagem original é aplicado alongamento de nível de cinza para realce de contraste. O método de projeção integral localizada com precisão as regiões candidatas e reais de *license plate*. O novo modelo de extração de características é proposto usando três conjuntos de características, dois conjuntos de densidade transversal e um de distância de borda. O reconhecimento dos caracteres da *license plate* é treinado através do BPNN. Nos resultados, a taxa de reconhecimento de caracteres é de 97,7% com tempo médio de predição de 46,1ms.

III. SISTEMA PROPOSTO PARA RECONHECIMENTO DE PLACAS VEICULARES

A. Introdução

Nesta seção apresenta-se alguns aspectos significativos a respeito dos sistemas de reconhecimento de placas veiculares

TABELA I
COMPARAÇÃO ENTRE TRABALHOS RELACIONADOS.

Autor	Base de dados	Método LPR	Acurácia	Tempo de predição
Neto E. C. <i>et al.</i> [8]	Autoria Própria	ELM	98,50%	50ms
Laroça R. e Menotti D. [4]	UFPR-ALPR	CNN	78,33%	28,5ms
Silva S. M. e Jung C. R. [9]	SSIG-SegPlate, UFPR-ALPR e OpenALPR	Segmentação CNN Coerência temporal	92,41%	13ms
Xie F. <i>et al.</i> [10]	Autoria Própria	Extração de características BPNN	97,70%	46,1ms

propostos. Na Seção III-B apresenta-se o processo para realização do treinamento para classificação de placas veiculares utilizando como método modelos baseados em CNN. Na Seção III-C é descrito o procedimento para realização do treinamento do reconhecimento de placas veiculares utilizando o MSM.

B. Método baseado em redes neurais convolucionais (CNN)

A Figura 1 ilustra a arquitetura utilizada de redes neurais. Antes da entrada das imagens na arquitetura CNN é realizado o processo de conversão das imagens de RGB para *grayscale*. A equação (1) define a operação necessária para obter a imagem em *grayscale* (1):

$$I = 0,114I_R + 0,587I_G + 0,299I_B \quad (1)$$

onde I é a imagem em *grayscale*, e I_R , I_G e I_B são respectivamente as componentes: vermelha, verde e azul da imagem RGB [11].

A seguir é aplicado a camada de convolução que é composta por filtros que realizarão o processo de extração de características nas imagens, sendo que esse processo pode ser descrito conforme a equação (2),

$$I'(x, y) = \sum_{i=-\frac{N}{2}}^{\frac{N}{2}} \sum_{j=-\frac{M}{2}}^{\frac{M}{2}} F(i, j) I(x + i, y + j) \quad (2)$$

onde N , M são as dimensões do filtro e x , y a posição dos pixels na imagem [12].

Então as matrizes são redimensionadas para se tornar um vetor linear e para finalizar são aplicadas as camadas densas (definida no Algoritmo 1 [12]) que conecta todos os neurônios da camada anterior a cada um dos T neurônios da camada posterior, sendo que T é igual a K (onde K é a quantidade de classes do conjunto de classificação) na camada de saída.

Algoritmo 1: Camada densa

```

1 para todo neurônio  $j$  da camada totalmente conectada
  faça
2   para toda entrada  $i$  da camada anterior faça
3      $u_j = \sum_{i=1}^n x_i w_{i,j} + b_j$ 
4      $y_j = f(u_j)$ 
5   fim
6 fim
```

A cada vez que o conjunto de imagens passa por todas essas etapas do modelo calcula-se o erro e o propaga de volta, com isso ajusta-se os parâmetros do modelo para melhorar a acurácia do processo. Os principais parâmetros ajustados são: os pesos da rede neural (sinapses) e os filtros (extratores de características) [13].

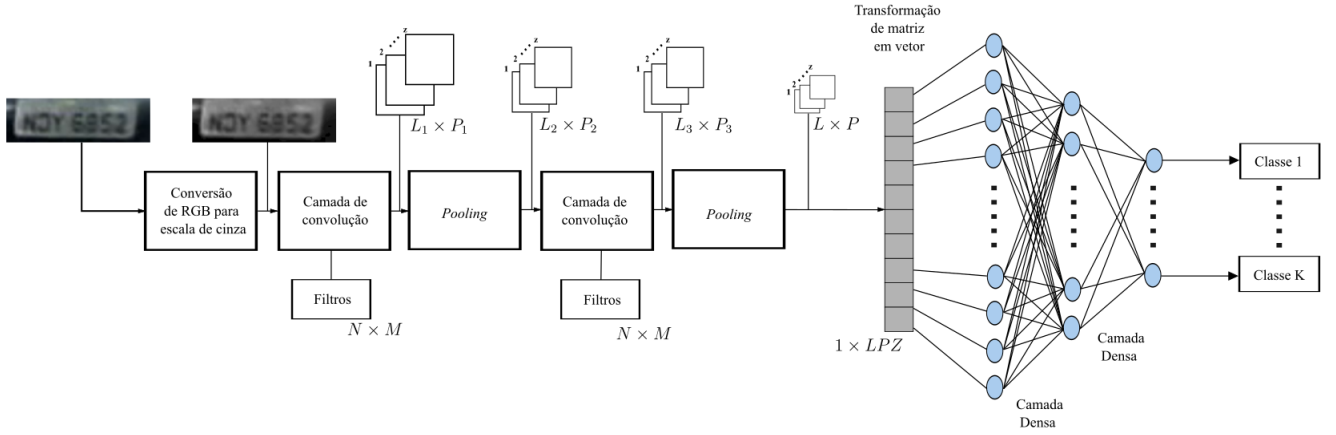


Fig. 1. Diagrama em blocos da estrutura do modelo CNN para treinamento do algoritmo de reconhecimento.

C. Método utilizando subespaço mútuo (MSM)

A Figura 2 descreve as etapas do método de classificação através do MSM. A primeira etapa consiste em realizar a conversão das imagens de RGB para *grayscale*. Em seguida, aplica-se um redimensionamento da imagem [14].

A próxima etapa é concatenar as imagens de uma classe em uma só matriz de dados. Seja v_j um vetor imagem de determinada classe (D_i) do conjunto de imagens e K a quantidade de imagens de cada classe do conjunto de imagens, a concatenação dos vetores imagens de uma classe é definida conforme a equação (3).

$$D_i = [v_1, v_2, \dots, v_{K-1}, v_K] \quad (3)$$

Na etapa seguinte emprega-se a decomposição em valores singulares [15], conhecida como SVD (*singular value decomposition*). Utilizando a equação (4) é determinado os valores singulares σ_i da matriz Q . Logo, com a equação (5), é realizado a decomposição em valores singulares da matriz Q ;

$$\sigma_i = \sqrt{\lambda_i} \quad (4)$$

$$Q = U \Sigma V^T \quad (5)$$

onde U é uma matriz ortogonal $M \times M$, cujos elementos são encontrados pela equação (6). O termo Σ é uma matriz $M \times N$ que possuem como entradas diagonais σ_i que são os valores singulares de Q e as demais entradas da matriz são nulas e V uma matriz ortogonal $N \times N$, composta por um conjunto ortonormal de autovetores da matriz Q , $\{v_1, v_2, \dots, v_N\}$.

$$u_i = \frac{1}{\sigma_i} Q v_i \quad (6)$$

Em seguida aplica-se a decomposição em autovalores de cada conjunto de vetores de classe. Na predição é realizado as mesmas etapas citadas anteriormente e por fim executa-se o processo de MSM.

O MSM executa o reconhecimento nas imagens inseridas através do cálculo da similaridade dos valores singulares que foram registrados como o dicionário de cada classe e os valores singulares que foi obtido a partir dos dados de entrada [16]. Sendo d_p a dimensionalidade do subespaço de entrada

P e d_q a dimensionalidade do subespaço de referencia Q_i (por conveniência, $d_p \leq d_q$), a equação (7) descreve como determina-se os ângulos canônicos entre P e Q ,

$$\cos^2(\theta_i) = \max \frac{(u_i \cdot v_i)^2}{\|u_i\|^2 \|v_i\|^2} \quad (7)$$

onde $u_i \in P$, $v_i \in Q$.

Um método funcional de encontrar os ângulos canônicos tendo Φ_i e Ψ_i (bases ortonormais dos vetores dos subespaços P e Q respectivamente) é por meio de $X = Q^T P$ [12], onde $Q = [\Phi_1, \dots, \Phi_{d_p}]$ e $P = [\Psi_1, \dots, \Psi_{d_q}]$. Com $[k_1, \dots, k_{d_q}]$ sendo os valores singulares da matriz X . Os cossenos dos ângulos canônicos $[\theta_1, \dots, \theta_{d_q}]$ podem ser obtidos como $[k_1, \dots, k_{d_q}]$ [17].

O MSM original usa apenas um ângulo canônico mínimo θ_1 para definir a similaridade. No entanto, é possível utilizar a média dos t menores ângulos canônicos para aumentar a precisão da classificação, isso é praticável já que os outros ângulos canônicos também possuem informações relevantes [18]. O valor S da equação (8) então reflete a similaridade estrutural entre dois subespaços.

$$S = \frac{1}{t} \sum_{i=1}^t \cos^2(\theta_i) \quad (8)$$

IV. PROCEDIMENTOS EXPERIMENTAIS

A. Dataset

Neste artigo, os experimentos foram realizados utilizando a conjunto de imagens criada pelo autor Garcia [19] em sua pesquisa sobre detecção de placas veiculares. Na Figura 3 é apresentado algumas amostras do conjunto de imagens original.

A primeira etapa consistiu em aplicar o detector de objetos YOLO (*you only look me*) [20] para realizar a detecção do veículo com o intuito de diminuir a complexidade para a detecção das placas veiculares.

Na etapa seguinte foi realizada a identificação dos caracteres da placa de forma visual de cada imagem do *dataset*. Por fim foi aplicado aumento de dados para aumentar a quantidade de imagens no *dataset* e para testar a robustez dos sistemas

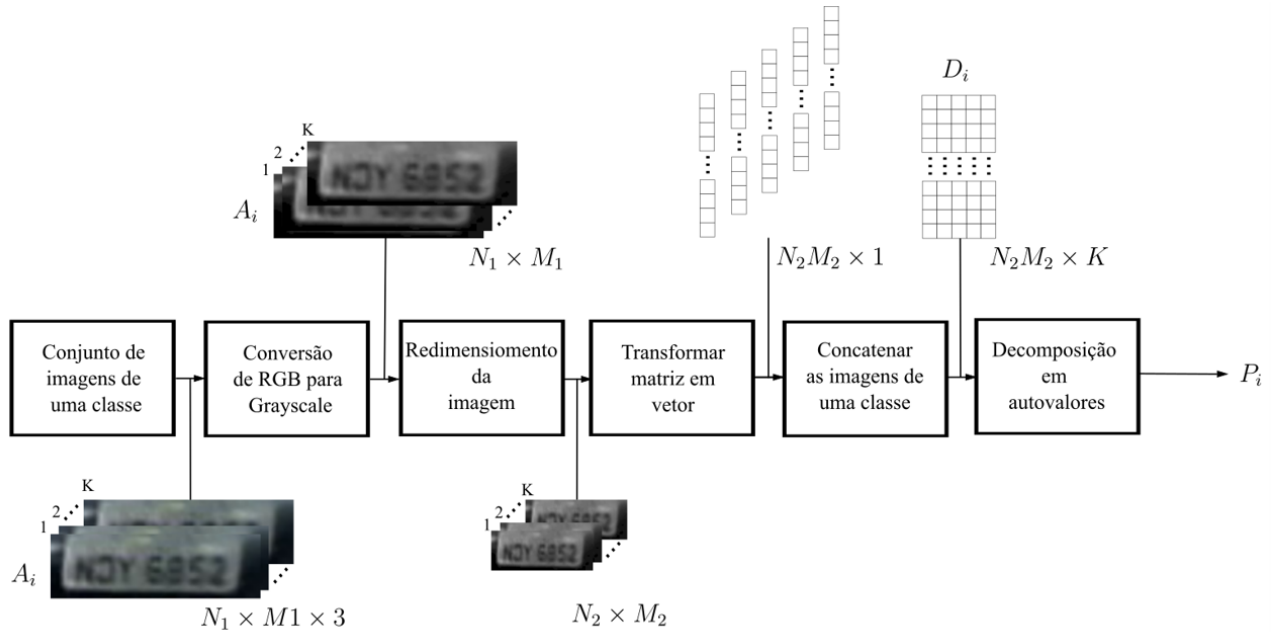


Fig. 2. Diagrama em blocos do processo para treinamento do método de classificação por subespaço.



Fig. 3. Exemplos do conjunto de imagens original.

propostos. As base de dados possuem 32 classes (cada classe diz respeito a uma placa de identificação veicular), cada classe possui 300 imagens, totalizando 9600 imagens [12]. Os *dataset* foram repartidos de forma igual em treinamento, validação e testes; com 3200 em cada partição. Os tipos de degradações aplicados foram: ruído gaussiano, ruído de Poisson, ruído laplaciano, mudança de escala, rotação e mudança de contraste [12]. No total, 7 bases de dados de imagens foram criadas, sendo 6 destas com um tipo de degradação diferente e a última base de dados com a combinação de todas as 6 degradações.

B. Resultados obtidos

O desempenho do modelo proposto foi avaliado através da acurácia. Esta métrica foi escolhida devido ao conjunto de dados possuir a mesma quantidade de exemplos em cada

classe. A Tabela II apresenta os resultados obtidos dos sistemas propostos.

Em todos os *dataset* foram adotados a realização de 50 épocas de treinamentos dos modelos CNN, sendo salvo os pesos da época que apresentou os melhores resultados. O melhor resultado de acurácia (igual a 99%) foi obtido com o *dataset* de degradação do tipo rotação. No geral os resultados são expressivos levando em consideração a alta variação de degradações aplicadas nas bases de dados.

TABELA II
RESULTADOS OBTIDOS DOS SISTEMAS PROPOSTOS.

Degradação	Acurácia da CNN	Acurácia do MSM
Poisson	88%	52%
Contraste	81%	55%
Gaussiano	77%	51%
Laplaciano	86%	51%
Rotação	99%	49%
Escala	93%	56%
Combinadas	98%	50%

No sistema proposto utilizando MSM o parâmetro de dimensionalidade do subespaço foi variado entre os valores 5 e 20 e salvo os melhores resultados. O melhor resultado de acurácia (igual a 56%) foi obtido aplicando degradação de mudança de escala nas imagens. Destaca-se os resultados obtidos no *dataset* que possui todos os tipos de degradações, podendo concluir com isso a capacidade de extrair as informações pertinentes sobre cada classe e excluir as degradações.

Para um melhor entendimento dos resultados obtidos utilizando o método do subespaço foi realizada uma análise das acurácias por cada classe da base de dados com todas as degradações aplicadas. A classe 28 obteve o melhor resultado na base de dados mostrada na Tabela III, enquanto que a classe 12 obteve o pior resultado.

TABELA III
ANÁLISE DOS RESULTADOS POR CLASSE.

Classe	Quantidade de frames	Acurácia
1	12	52%
2	11	95%
3	21	42%
4	13	75%
5	19	21%
6	30	89%
7	15	9%
8	21	39%
9	23	55%
10	17	52%
11	19	85%
12	10	1%
13	77	74%
14	19	68%
15	27	64%
16	19	6%
17	26	64%
18	230	96%
19	11	2%
20	11	92%
21	13	2%
22	12	34%
23	19	62%
24	34	75%
25	20	36%
26	75	45%
27	12	2%
28	22	100%
29	38	74%
30	25	23%
31	24	25%
32	12	36%

No método do subespaço os resultados são mais impactados pela qualidade das amostras do que pela quantidade. Isso pode ser comprovado ao observar na Tabela III que a classe 28 que obteve os melhores resultados de acurácia possui 22 amostras, superando a classe 18 que possui 230 amostras e é a classe com a maior quantidade de amostras.

A predição média utilizando MSM foi de 0,33ms. Já a predição média usando CNN foi de 200ms. Como comparação o método mais rápido entre os trabalhos relacionados obteve 13ms como tempo de predição. É considerável destacar que os trabalhos relacionados utilizaram GPU (*grafic processing unit*) na etapa de predição enquanto que nos métodos propostos nesta pesquisa foi utilizado apenas CPU (*central processing unit*), além disso, alguns dos trabalhos relacionados foram realizados em plataformas e/ou linguagens diferentes.

V. CONCLUSÕES

Neste artigo, foi concebido dois modelos de reconhecimento de placas veiculares em cenários complexos. A primeira abordagem utilizada foi a mutual subspace method e a segunda abordagem foi a convolutional neural network.

Em termos de resultados, o sistema por MSM obteve acurácia global igual a 56% e tempo de detecção médio de 0,33 ms. Para o sistema por CNN a acurácia global foi igual a 99% e tempo médio de detecção foi de 200ms, sendo consideravelmente mais lento do que a detecção utilizando o MSM. Analisando por classe concluímos que o sistema por MSM é altamente dependente da qualidade das imagens, porém não necessita de muitas imagens pra obter uma alta acurácia na classificação. Desta forma, observa-se que o MSM é uma boa opção em aplicações: na qual há limitação de hardware, que são realizadas em tempo real e onde não é possível criar grandes bases de dados. Por outro lado, o método baseado em CNN se encaixa em aplicações: que possui o poder

computacional elevado, onde contém banco de dados robusto para treinamento e que exige grande acurácia na predição.

Para este artigo, podemos observar algumas possibilidades de trabalhos futuros como: utilizar a técnica de super resolução proposta por [19]; explorar outros métodos baseados em subespaço como, por exemplo, o CMSM (*Constrained Mutual Subspace Method*) ou o OMSM (*Orthogonal Mutual Subspace Method*); ampliar os tipos e níveis de degradações aplicadas nas bases de imagens e empregar redundância temporal.

VI. AGRADECIMENTOS

Parte dos resultados desta pesquisa foram subsidiados por ENVISION Indústria de Produtos Eletrônicos LTDA nos termos da Lei Brasileira Federal No. 8.387/91 (SUFRAMA).

REFERÊNCIAS

- [1] IBGE. <https://cidades.ibge.gov.br/brasil/pesquisa/22/28120?ano=2021>, Acesso em maio de 2022.
- [2] K. Yogheedha, A.S.A. Nasir, H. Jaafar, and S.M. Mamduh. Automatic vehicle license plate recognition system based on image processing and template matching approach. In *Proc. Int. Conf. on Computational Approach in Smart Systems Design and Applications (ICASSDA)*, pages 1–8, 2018.
- [3] L. Xie, T. Ahmad, L. Jin, Y. Liu, and S. Zhang. A new CNN-based method for multi-directional car license plate detection. *IEEE Transactions on Intelligent Transportation Systems*, 19(2):507–517, 2018.
- [4] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonçalves, W. R. Schwartz, and D. Menotti. A robust real-time automatic license plate recognition based on the YOLO detector. *Proc. Int. Joint Conf. on neural networks (IJCNN)*, pages 1–10, 2018.
- [5] R. Laroca and D. Menotti. Automatic license plate recognition: An efficient and layout-independent system based on the YOLO detector. In *Anais XXXIII Conf. on Graphics, Patterns and Images*, pages 15–21, Porto Alegre, RS, Brasil, 2020. SBC.
- [6] H. Li, P. Wang, M. You, and C. Shen. Reading car license plates using deep neural networks. *Image and Vision Computing*, 72, 03 2018.
- [7] Hengliang Tan, Ying Gao, and Zhengming Ma. Regularized constraint subspace based method for image set classification. *Pattern Recognition*, 76, 11 2017.
- [8] E. C. Neto, S. L. Gomes, P. P. R. Filho, and V. H. C. De Albuquerque. Brazilian vehicle identification using a new embedded plate recognition system. *Measurement*, 70:36–46, 2015.
- [9] S. S. Montazzolli and C. R. Jung. Real-time license plate detection and recognition using deep convolutional neural networks. *Journal of Visual Communication and Image Representation*, 71:102773, 2020.
- [10] F. Xie, M. Zhang, J. Zhao, J. Yang, Y. Liu, and X. Yuan. A robust license plate detection and character recognition algorithm based on a combined feature extraction model and BPNN. *Journal of Advanced Transportation*, 2018:1–14, 09 2018.
- [11] R. C. Gonzalez and R. C. Woods. *Processamento digital de imagens*. Pearson Educación, 2009.
- [12] A. Sousa. Reconhecimento de placas veiculares em cenários complexos utilizando o método do subespaço. Master's thesis, Universidade Federal do Amazonas, 2021.
- [13] E. Ebermam and R. Krohling. Uma introdução compreensiva às redes neurais convolucionais: Um estudo de caso para reconhecimento de caracteres alfabéticos. *Revista de Sistemas de Informação da FSMA*, 22:49–59, 2018.
- [14] Qi Wang and Yuan Yuan. Learning to resize image. *Neurocomputing*, 131:357–367, 2014.
- [15] G. C. de Andrade and S. A. Santos. Decomposição em valores singulares e técnicas de compressão de dados. *Proc. Series of the Brazilian Society of Computational and Applied Mathematics*, 7(1), 2020.
- [16] H. Sakano, N. Mukawa, and T. Nakamura. Kernel mutual subspace method and its application for object recognition. *Electronics and Communications in Japan (Part II: Electronics)*, 88(6):45–53, 2005.
- [17] K. Fukui and O. Yamaguchi. The kernel orthogonal mutual subspace method and its application to 3D object recognition. In *Asian Conf. on Computer Vision*, pages 467–476. Springer, 2007.
- [18] Kazuhiro Fukui. Subspace methods. *Computer Vision: A Reference Guide*, pages 1–5, 2020.
- [19] L. C. G. da Silva Junior. Detecção de placas de licenciamento veicular utilizando imagens com diferentes resoluções e em ambiente não controlado empregando técnicas de super resolução. 2019.
- [20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proc. IEEE Int. Conf. on computer vision and pattern recognition (CVPR)*, pages 779–788, 2016.