

Detecção de buracos em pavimento asfáltico com base em Processamento Digital de Imagens e Deep Learning

Bianka Tallita Passos
Universidade do Vale do Itajaí - UNIVALI
Itajaí, Santa Catarina, BR
biankatpas@edu.univali.br

Wemerson Delcio Parreira
Universidade do Vale do Itajaí - UNIVALI
Itajaí, Santa Catarina, BR
parreira@univali.br

Anita Maria da Rocha Fernandes
Universidade do Vale do Itajaí - UNIVALI
Itajaí, Santa Catarina, BR
anita.fernandes@univali.br

Eros Comunello
Universidade do Vale do Itajaí - UNIVALI
Itajaí, Santa Catarina, BR
eros.com@univali.br

ABSTRACT

The road infrastructure conditions are directly related to the safety and operational cost of transportation. Potholes are defects in the paving that affect safety on the road. Therefore, identifying potholes is an important step in defining road maintenance and intervention strategies. Among the approaches used to detect defects in roads are vibration techniques, laser scanning and 3D reconstruction, and finally methods that are vision-based. These vision-based methods utilize image processing, considered low cost and that can be performed by common two-dimensional cameras. This research aims to combine digital image processing and deep learning concepts facilitating the recognition of pothole-like defects in road images with asphalt paving. In order to carry out these experiments, different network architectures were used.

KEYWORDS

Redes Neurais Convolucionais, Processamento Digital de Imagens, Buracos, Pavimento, Defeitos

1 INTRODUÇÃO

O estado de conservação das estradas influencia diretamente no custo do transporte e na qualidade do produto transportado. Trechos de estradas ruins (com buracos, falhas, etc) podem causar danos aos veículos, impossibilidade de tráfego e acidentes. Nestes casos, o uso de rotas alternativas, quando disponíveis, podem ser úteis. Porém, essas podem ser mais longas, gerando um maior consumo de combustível.

Em um país de dimensão continental como o Brasil, é importante o monitoramento da situação das rodovias, principalmente, as condições de tráfego. Dado que a malha rodoviária é a responsável pela movimentação da economia entre as regiões do país. De acordo com a 23ª edição da Pesquisa CNT (Confederação Nacional de Transportes) de Rodovias, 59% da malha rodoviária apresenta problemas. Sendo que, em 2018, o percentual era 57%. Também foi observada piora na situação do pavimento (52,4% com problema), da sinalização (48,1%) e da geometria da via (76,3%). Em 2018, a avaliação foi de 50,9%, 44,7% e 75,7% com problemas, respectivamente [1].

Na pesquisa da CNT, são avaliadas as condições de toda a malha federal pavimentada e dos principais trechos estaduais, também pavimentados. Em 2019, foram percorridas todas as cinco regiões

do Brasil, durante 30 dias (de 20 de maio a 18 de junho), por 24 equipes de pesquisadores. Estima-se que, na média nacional, as inadequações do pavimento resultaram em uma elevação do custo operacional do transporte em torno de 28,5%. O maior índice foi registrado na região Norte, mais de 38,5% [1].

A presença de defeitos, tais como buracos, reduz a qualidade da via, o que pode aumentar o número de acidentes, bem como o custo do transporte. Portanto, a avaliação do estado do pavimento é necessária para desenvolver sistemas de manutenção das estradas. O processo de coleta de dados é feito automaticamente, mas a avaliação dos dados é efetuada de forma manual por técnicos da área [2].

A aquisição de imagens em rodovias é um fator que contribui para a análise e manutenção das estradas. Entretanto, essa operação pode ser demorada, subjetiva e de alto custo, pois depende de avaliadores humanos para identificar e marcar os defeitos nas imagens. Como consequência, os governos enfrentam desafios financeiros e físicos para manter as condições das redes rodoviárias [3].

Roberts et al. [3] afirmam que os métodos para identificação de defeitos em estradas presentes na literatura podem ser agrupados em termos do equipamento utilizado, como lasers, imagens, sensores e radares. Outros métodos, menos pesquisados, são baseados em pressão, acústica e vibração. Os métodos baseados em vibrações se destinam apenas a levantamentos preliminares já que carecem de precisão e confiabilidade. Os métodos baseados em reconstrução 3D sofrem basicamente de custos elevados de equipamentos [4]. Os métodos baseados em visão utilizam câmeras e imagens 2D para detectar os buracos e trincas presentes nas estradas asfaltadas. Essa técnica propicia um baixo custo, já que câmeras comuns podem ser utilizadas na captura das imagens [5].

De acordo com Cao et al. [6], dentre os algoritmos de processamento digital de imagens utilizados para detecção de defeitos em estradas destacam-se limiarização, detecção de bordas e métodos de crescimento de região. Estes algoritmos são utilizados para compor um conjunto de características do objeto que, posteriormente, são utilizados como dados de entrada para efetuar o treinamento de classificadores, tais como rede neural e máquina de vetores de suporte. Os classificadores são responsáveis pela identificação dos defeitos presentes no pavimento.

Cao et al. [6] afirmam que métodos baseados em aprendizado profundo (*deep learning*) mudam fundamentalmente a forma de extração de descritores e melhoram o desempenho da detecção

dos defeitos. Com base no exposto, esta pesquisa objetiva englobar conceitos de Processamento Digital de Imagens (PDI) e *deep learning* destinado a identificação de defeitos do tipo buraco em imagens de estradas com pavimento asfáltico.

2 CONTEXTUALIZAÇÃO

Processamento digital de imagens digitais é o termo usado para identificar técnicas de modificação e manipulação de imagens [7]. Dentre as áreas de aplicação de PDI, é possível destacar duas: melhoria de informação visual para a interpretação humana e o processamento de dados de cenas para percepção automática através de máquinas [8].

A Visão Computacional é o processo de modelagem e replicação da visão humana que, utilizando software e hardware, imita a cognição humana e a habilidade do ser humano em tomar decisões de acordo com as informações contidas na imagem. A tomada de decisões ocorre por meio da extração de informações do mundo real ao efetuar o processamento digital de imagens e o reconhecimento de padrões [9].

Deep Learning é uma abordagem de inteligência artificial baseada em redes neurais na qual a rede possui um maior número de camadas – conforme a Figura 1 ilustra – e de operações. Cada camada da rede é responsável por extrair determinada característica do objeto presente na imagem. A informação é passada através de cada camada, com a saída da camada anterior fornecendo entrada para a próxima camada [10].

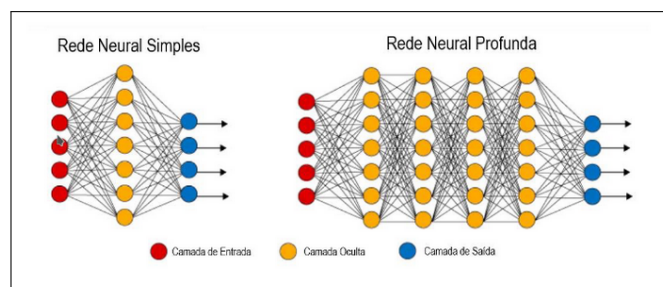


Figura 1: Representação de uma rede neural simples e uma rede neural profunda [10].

O reconhecimento de objetos é uma tarefa de Visão Computacional que consiste em atribuir um rótulo a um objeto presente na imagem. Diversos fatores contribuem para tornar esta tarefa complexa, como a oclusão parcial do objeto, diferenças de iluminação, variações em cenas de fundo, além das diferentes configurações que objetos da mesma classe podem apresentar [11].

Os recentes avanços teóricos e tecnológicos permitiram o uso eficiente de redes neurais de múltiplas camadas nessa tarefa. Uma das principais motivações desses desenvolvimentos é a descoberta automática de uma descrição mais apropriada das características do objeto [10].

Na área de Visão Computacional, técnicas baseadas na extração de descritores e aprendizado de máquina podem ser empregadas para a solução dos problemas relacionados à identificação de defeitos em imagens de estradas asfaltadas.

3 SOLUÇÃO PROPOSTA

O presente trabalho propõe utilizar algoritmos de processamento digital de imagem e *deep learning* para identificar e localizar buracos presentes na cena. A solução proposta visa reduzir o trabalho custoso e subjetivo dos técnicos responsáveis pela tarefa de identificação de defeitos na rodovia. A solução pode ser aplicada em uma etapa anterior à análise manual, com o objetivo de diminuir a quantidade de imagens que precisam ser analisadas pelos técnicos ou como uma forma de substituição da análise manual.

Como existe uma relação direta entre as taxas de acidentes e as condições do pavimento, sistemas automatizados para avaliar o estado das estradas, principalmente, identificando e classificando os defeitos da rodovia, colaboram com a manutenção de condições rodoviárias adequadas aos habitantes [3].

4 PROJETO DO EXPERIMENTO

Para atingir o objetivo deste trabalho, foi desenvolvido um protótipo de software, utilizando a linguagem Python, responsável pela tarefa de identificação de buracos no pavimento. O protótipo é dividido em 4 partes distintas, conforme está descrito na Figura 2.

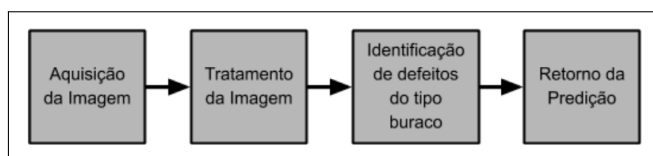


Figura 2: Representação do protótipo.

Primeiramente, a imagem de entrada passa por um pré-processamento com o intuito de remover possíveis ruídos. Após o pré-processamento, inicia-se a etapa de identificação dos objetos (defeitos no pavimento) presentes na cena. Por fim, o protótipo retorna a localização do objeto na imagem, quando o mesmo é detectado. Na sequência são detalhados os procedimentos técnicos adotados nas principais etapas do desenvolvimento do protótipo, a saber: a preparação do *dataset* e o treinamento da arquitetura de rede.

A base de dados utilizada nesta pesquisa é composta por 2235 imagens, das quais 566 apresentam defeito do tipo buraco. No total, a base de imagens possui 1534 amostras de buracos [12]. Abaixo estão listadas as características da base de dados:

- As imagens utilizadas foram disponibilizadas pelo Departamento Nacional de Infraestrutura de Transportes (DNIT) - Protocolo 50650.003556/2017-28;
- As capturas foram realizadas nos anos de 2014 – 2017;
- As imagens são de rodovias dos estados do Espírito Santo, Rio Grande do Sul e do Distrito Federal; e
- As imagens possuem resolução 1024×640 pixels.

A Figura 3 apresenta alguns exemplos de imagens da base de dados.

A base de dados contém a anotação dos objetos (defeitos) presentes na imagem – *ground-truth*. Além da identificação dos buracos, a base também possui anotações dos defeitos do tipo trinca, bem como da rodovia. As marcações estão no formato de máscaras – imagens binárias – separadas para cada tipo de anotação: estrada, trincas e buracos.

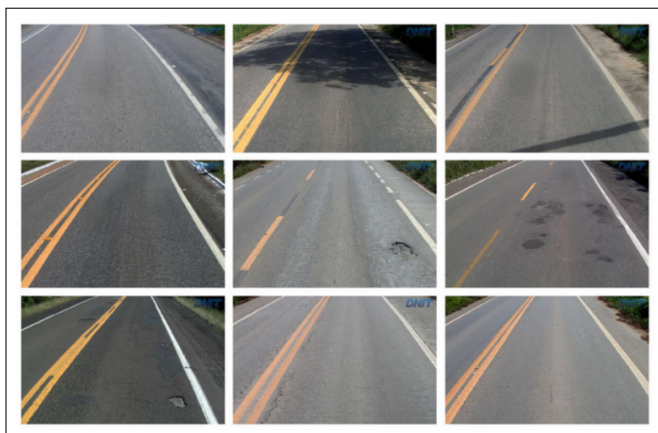


Figura 3: Imagens da base de dados [12].

A anotação da estrada possui a demarcação da região total correspondente à via do veículo, conforme a Figura 4.

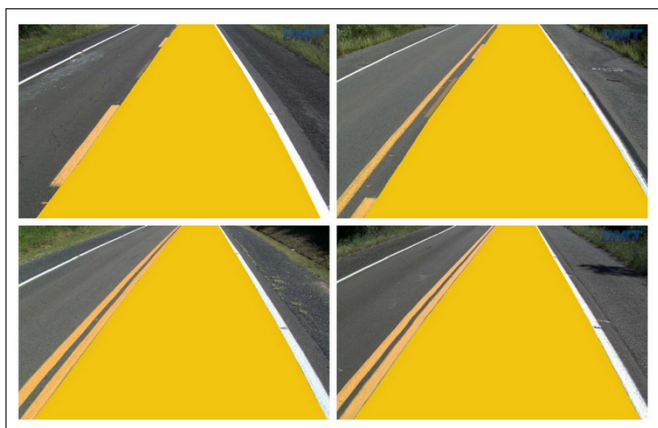


Figura 4: Exemplo de anotação da região da rodovia [12].

A anotação das trincas e dos buracos apresenta a seleção do defeito como um todo, mantendo ao máximo sua forma, conforme exposto na Figura 5.

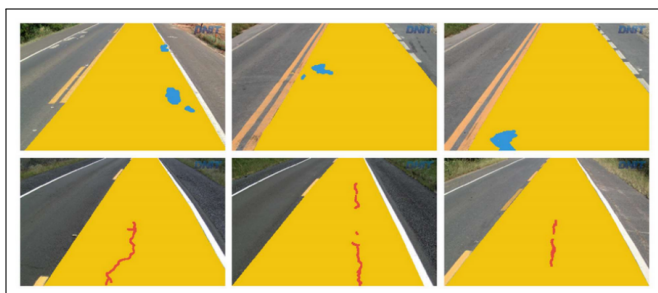


Figura 5: Exemplo de anotação de buracos (azul) e rachaduras (vermelho) [12].

A Figura 6 mostra as máscaras separadas para cada tipo de anotação - estrada, buraco e trinca - que compõem esta base de dados.

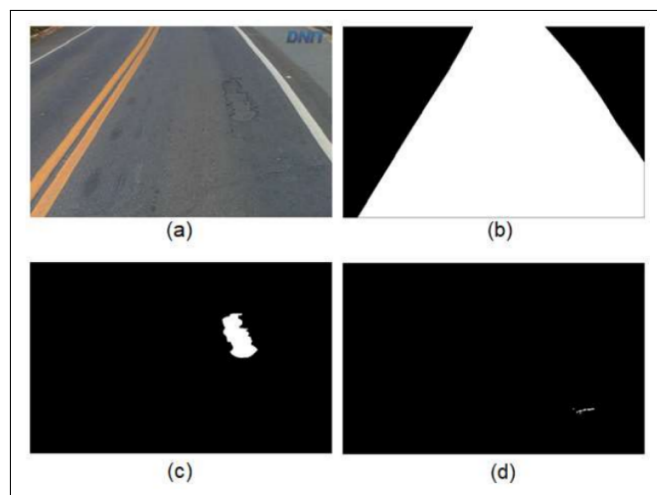


Figura 6: Exemplo da imagem original (a) e das máscaras correspondentes à região da estrada (b), buraco (c) e trinca (d) [12].

O conjunto de imagens foi dividido, sendo que 70% das imagens foram utilizadas para treinamento, 10% para validação e 20% para testes. A base de dados foi dividida de forma randômica por meio da biblioteca *scikit-learn*. Os dados de treinamento são utilizados para que a rede aprenda, enquanto os dados de validação são responsáveis por fornecer uma avaliação imparcial do modelo no conjunto de dados de treinamento enquanto os hiperparâmetros são ajustados. Por fim, a base de testes é utilizada para avaliar o modelo gerado, portanto só é utilizada quando o treinamento finaliza.

A próxima etapa foi efetuar o treinamento da arquitetura de *deep learning*. Nesta etapa, foi efetuado o treinamento, por meio de transferência de aprendizagem, de 3 arquiteturas de detecção de objetos, a saber: MobileNet [13], Inception [14] e Faster RCNN Inception [15]. Para executar essa etapa, foi utilizada a API de Detecção de Objetos do TensorFlow.

O TensorFlow disponibiliza uma API de Detecção de Objetos que permite efetuar o treinamento, bem como a exportação da rede treinada. Para efetuar o treinamento é necessário informar o local onde serão salvos os arquivos de *checkpoint* gerados durante o processo e o arquivo de configuração do treinamento.

No arquivo de configuração foram alterados os parâmetros referentes ao número de classes do novo modelo, o diretório do arquivo de *checkpoint* do modelo pré-treinado utilizado e os diretórios onde se encontram as imagens e demais arquivos necessários para efetuar o treinamento. Também foram alterados os parâmetros referentes a quantidade de imagens lida por etapa (*batch size*) e o número de etapas do treinamento (*steps*).

As redes treinadas foram avaliadas por meio de testes, com um conjunto de imagens desconhecido (20% da base de dados), para verificar sua capacidade de detecção dos defeitos no pavimento.

5 RESULTADOS E DISCUSSÃO

Nesta seção apresenta-se os resultados obtidos a partir dos experimentos realizados com o protótipo responsável por identificar os defeitos no pavimento, bem como discute-se sobre os resultados obtidos.

Para efetuar a análise dos resultados foram aplicadas 3 métricas de avaliação, a saber: precisão, *recall* e *F1-score*. A inferência da rede foi submetida a avaliação humana para determinar o número de falsos positivos (FP), verdadeiros positivos (VP), falsos negativos (FN) e verdadeiros negativos (VN).

A precisão permite identificar problemas de falsos positivos, enquanto o *recall* ou revocação permite detectar problemas de falsos negativos. A métrica *F1-score* é a média harmônica entre a precisão e a revocação. A média harmônica tende a punir valores baixos, dessa forma é possível obter uma visão geral do desempenho do classificador para uma determinada classe [16].

No total, foram efetuados 12 treinamentos, são eles: arquitetura de rede MobileNet com 4.000, 10.000 e 50.000, arquitetura de rede Inception com 4.000, 10.000 e 50.000 épocas, arquitetura de rede Faster RCNN Inception com 200, 500, 1.000, 4.000, 10.000 e 50.000 épocas. No entanto, 6 treinamentos apresentaram resultados relevantes na detecção dos defeitos, são eles: arquitetura de rede MobileNet com 50.000 épocas, arquitetura de rede Inception com 4.000, 10.000 e 50.000 épocas e arquitetura de rede Faster RCNN Inception com 10.000 e 50.000 épocas. A Tabela 1 apresenta a precisão, revocação e *F1-score* obtidos.

Tabela 1: Precisão, recall e F1-score do treinamento.

Arquitetura	Épocas	Precisão	Revocação	F1-score
MobileNet	50.000	0.36	0.68	0.47
Inception	4.000	0.36	0.68	0.45
Inception	10.000	0.36	0.69	0.45
Inception	50.000	0.53	0.45	0.48
Faster RCNN Inception	10.000	0.43	0.27	0.33
Faster RCNN Inception	50.000	0.34	0.76	0.47

Os resultados mostram que as arquiteturas de rede testadas geraram resultados melhores ao serem treinadas por mais épocas. Por meio da análise da precisão é possível concluir que todas as redes detectaram um número considerável de falsos positivos. Sendo que, o treinamento efetuado com a arquitetura de rede Inception por 50.000 épocas gerou valores de precisão e revocação mais equilibrados, conforme exposto pelo valor *F1-score*.

Os experimentos foram efetuados utilizando a imagem inteira como entrada. Logo, é importante ressaltar que durante o treinamento ocorre um redimensionamento da imagem. Algumas arquiteturas reduzem a imagem para uma resolução de 300x300, o que pode prejudicar o aprendizado de objetos pequenos ou com detalhes sutis, como é o caso de defeitos no pavimento. Sendo assim é possível que o treinamento, bem como a extração de características foram comprometidos pela mudança de resolução. Serão efetuados novos experimentos, dividindo a imagem em blocos de diferentes resoluções para comparar com os resultados atuais.

A Figura 7, apresenta amostras da detecção de falsos positivos. É possível observar que a ocorrência de falsos positivos possui relação

com regiões similares a defeitos do tipo buraco nas amostras de trinca, bem como remendos. Sendo assim, infere-se que o treinamento a partir de blocos da imagem permitirá a extração de detalhes do objeto para diferenciar essas regiões dos defeitos do tipo buraco. Outra possibilidade, considerando as marcações disponíveis na base de imagens, é efetuar novos testes utilizando arquiteturas de rede para segmentação semântica (por exemplo, a rede pix2pix).

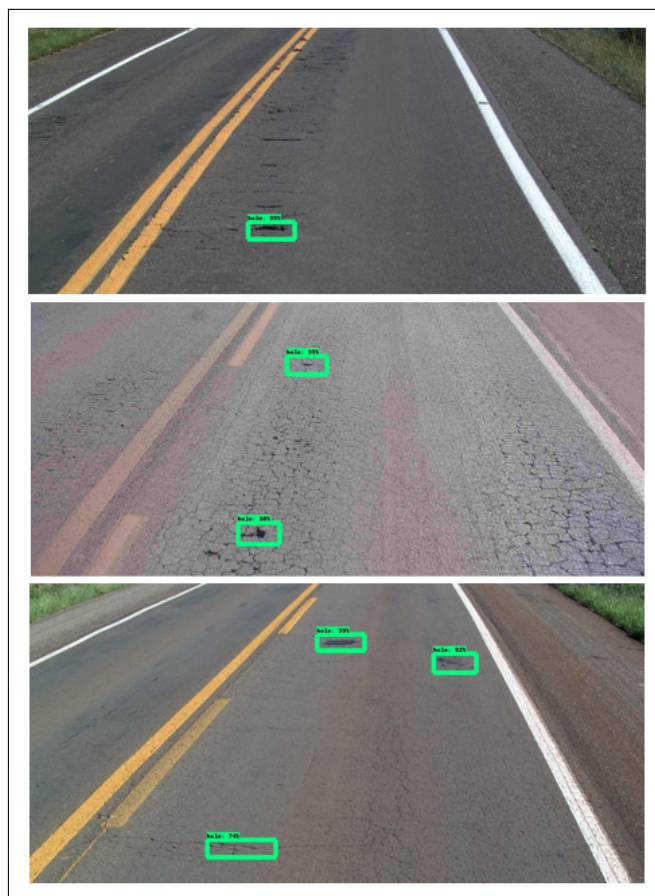


Figura 7: Amostras de falsos positivos.

Na Figura 8, podem ser observados exemplos da identificação dos defeitos (verdadeiros positivos) no pavimento.

A partir dos resultados, foi observado que o *score* - percentual de probabilidade do objeto pertencer a uma determinada classe foi em média 82%, 65%, 77%, 82%, 59% e 75% para as arquiteturas de rede MobileNet com 50.000 épocas, Inception com 4.000, 10.000 e 50.000 épocas e Faster RCNN Inception com 10.000 e 50.000 épocas, conforme exposto na Tabela 2, respectivamente.

O número de épocas é um hiperparâmetro do gradiente descendente que controla o número de passagens completas através do conjunto de dados de treinamento. Logo, uma vez que toda a população é utilizada para encontrar o mínimo local de cada iteração, o treinamento efetuado por mais épocas aumenta as chances de convergência para o mínimo local.

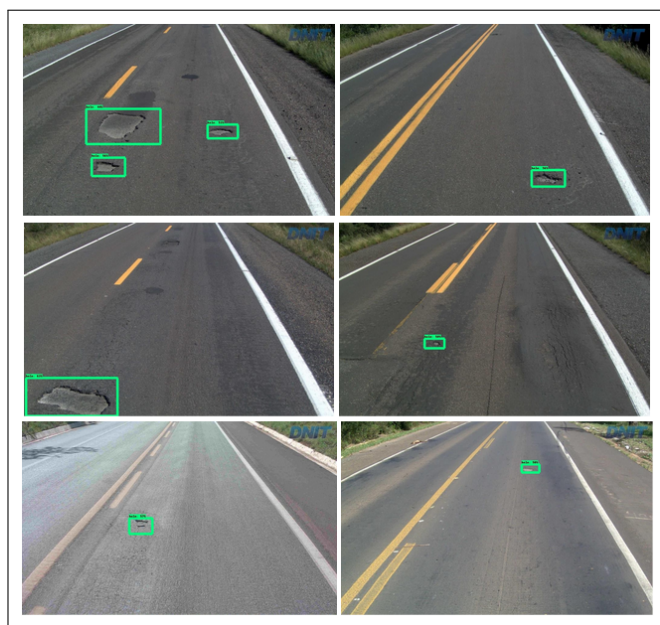


Figura 8: Resultado da identificação dos defeitos no pavimento.

Tabela 2: Score médio do treinamento.

Arquitetura	Épocas	Score
MobileNet	50.000	82%
Inception	4.000	65%
Inception	10.000	77%
Inception	50.000	82%
Faster RCNN Inception	10.000	59%
Faster RCNN Inception	50.000	75%

O tempo médio, utilizando um Processador Intel Core i7 3.5GHz 4MB, para detectar e inferir a qual classe o objeto presente na imagem pertence, foi de 40ms, 90ms, 80ms, 80ms, 1060ms e 980ms para as arquiteturas de rede MobileNet com 50.000 épocas, Inception com 4.000, 10.000 e 50.000 épocas e Faster RCNN Inception com 10.000 e 50.000 épocas, conforme Tabela 3.

Tabela 3: Tempo médio de inferência.

Arquitetura	Épocas	Tempo (ms)
MobileNet	50.000	40
Inception	4.000	90
Inception	10.000	80
Inception	50.000	80
Faster RCNN Inception	10.000	1060
Faster RCNN Inception	50.000	980

Os resultados obtidos forneceram um norte para a continuidade da pesquisa. Serão feitos novos testes, treinando as arquiteturas

por mais épocas, bem como analisando os hiperparâmetros de cada arquitetura. Dentre os resultados obtidos, é possível destacar o resultado a partir da arquitetura de rede Inception e da arquitetura Faster RCNN Inception, ambas treinadas por 50.000 épocas.

A arquitetura de rede Inception destacou-se por detectar um menor número de falsos positivos (39% acima da média) e apresentar um tempo de treinamento e de inferência menor. Já a arquitetura de rede Faster RCNN Inception destacou-se por obter um *recall* 17% acima da média. Por fim, é importante destacar que a arquitetura de rede MobileNet apresentou o menor tempo de inferência (40 ms). Logo, novos testes com essa arquitetura são relevantes para a utilização da mesma em sistemas mobile e embarcados.

6 CONCLUSÕES

As contribuições desta pesquisa estão relacionadas à análise das técnicas de *deep learning* que podem ser empregadas na tarefa de identificação de defeitos no pavimento. Outra contribuição é a averiguação das limitações encontradas na utilização restrita das técnicas de *deep learning* para extrair as características que descrevem os objetos (defeitos). Tais limitações podem ser tanto computacionais – tempo de processamento – quanto de performance – taxa de acertos.

O trabalho atual teve como propósito efetuar a detecção de defeitos do tipo buraco no pavimento asfáltico, sendo possível o incremento do mesmo para identificar remendos, trincas, bem como diferentes categorias dentro de uma mesma classe de defeitos. Experimentos considerando o treinamento com um conjunto de dados maior são relevantes para o aperfeiçoamento deste protótipo de software, posto que técnicas baseadas em *deep learning* tendem a se beneficiar de grandes volumes de dados.

Por fim, outros possíveis desmembramentos desse trabalho são: efetuar o treinamento de outras arquiteturas, bem como fazer um *benchmark* para analisar os resultados de diferentes parametrizações para efetuar o treinamento das arquiteturas de *deep learning* e utilizar métricas que consideram sobreposição de retângulos (IoU - Intersection over Union) e sobreposição exata de pixels para avaliar o resultado.

REFERENCES

- [1] SEST SENAT. Pesquisa cnt de rodovias 2019: relatório gerencial. – 23.ed., Brasília: Cnt: Sest: Senat, 2019, 2019. <https://pesquisarodovias.cnt.org.br/downloads/ultimaversao/gerencial.pdf>. Acesso em: 01 de jun. de 2020.
- [2] Surekha Arjapure and D. R. Kalbande. Review on analysis techniques for road pothole detection. In Millie Pant, Tarun K. Sharma, Om Prakash Verma, Rajesh Singla, and Afzal Sikander, editors, *Soft Computing: Theories and Applications*, pages 1189–1197, Singapore, 2020. Springer Singapore. ISBN 978-981-15-0751-9.
- [3] Ronald Roberts, Gaspare Giancontieri, Laura Inzerillo, and Gaetano Di Mino. Towards low-cost pavement condition health monitoring and analysis using deep learning. *Applied Sciences*, 10(1):319, 2020.
- [4] Christian Koch and Ioannis Brilakis. Pothole detection in asphalt pavement images. *Advanced Engineering Informatics*, 25(3):507–515, 2011.
- [5] Mateus Junior Cassaniga. Detecção de trincas e buracos em estradas asfaltadas, 2017. Anais Computer on The Beach 2017, pages 200–209, 2017.
- [6] W. Cao, Q. Liu, and Z. He. Review of pavement defect detection methods. *IEEE Access*, 8:14531–14544, 2020. doi: 10.1109/ACCESS.2020.2966881.
- [7] José Iguelmar Miranda. Processamento de imagens digitais: Prática usando java tm. *Campinas: Embrapa Informática Agropecuária*, 2006.
- [8] Rafael C Gonzalez and Richard E Woods. *Processamento de imagens digitais*. Editora Blucher, 2000.
- [9] Hélio Pedrini and William Robson Schwartz. *Análise de imagens digitais: princípios, algoritmos e aplicações*. Thomson Learning, 2008.

- [10] Goodfellow I, Bengio Y, and Courville A. *Deep learning*, volume 1. MIT press Cambridge, 2016. ISBN: 978-0262035613.
- [11] Ramesh Jain, Rangachar Kasturi, and Brian G Schunck. *Machine vision*, volume 5. McGraw-hill New York, 1995.
- [12] Bianka Tallita Passos, Mateus Junior Cassaniga, Anita Maria da Rocha Fernandes, Kátia Balvedi Medeiros, and Eros Comunello. Cracks and potholes in road images, 2020. URL <http://dx.doi.org/10.17632/t576ydh9v8.4>. Mendeley Data, V4.
- [13] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, and Adam H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.
- [14] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, and Rabinovich A. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. URL <https://doi.org/10.1109/CVPR.2015.7298594>.
- [15] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7310–7311, 2017.
- [16] Sebastian Raschka and Vahid Mirjalili. *Python machine learning*. Packt Publishing Ltd, 2017.