

Research Article

Recognizing the Damaged Surface Parts of Cars in the Real Scene Using a Deep Learning Framework

Mahboub Parhizkar  and **Majid Amirfakhrian** 

Department of Mathematics, Central Tehran Branch, Islamic Azad University, Tehran, Iran

Correspondence should be addressed to Majid Amirfakhrian; amirfakhrian@gmail.com

Received 30 May 2022; Revised 10 July 2022; Accepted 18 July 2022; Published 10 August 2022

Academic Editor: Araz Darba

Copyright © 2022 Mahboub Parhizkar and Majid Amirfakhrian. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Automatically recognizing the damaged surface parts of cars can noticeably diminish the cost of processing premium assertion that leads to providing contentment for vehicle users. This recognition task can be conducted using some machine learning (ML) strategies. Deep learning (DL) models as subsets of ML have indicated remarkable potential in object detection and recognition tasks. In this study, an automated recognition of the damaged surface parts of cars in the real scene is suggested that is based on a two-path convolutional neural network (CNN). Our strategy utilizes a ResNet-50 at the beginning of each route to explore low-level features efficiently. Moreover, we proposed new mReLU and inception blocks in each route that are responsible for extracting high-level visual features. The experimental results proved the suggested model obtained high performance in comparison to some state-of-the-art frameworks.

1. Introduction

The evaluation of car parts is a challenging task that mainly originates from the insurance industry. The issue of automated assessment of damaged parts of a car represents the foremost challenge in the damage assessment and auto repair industry [1, 2]. This field of study has a number of application areas ranging from accidental damage evaluation for car insurance companies to car evaluation companies such as body shops and car rentals [3]. In evaluating a vehicle, the damaged parts can take any form containing missing parts, minor and major dents, and scratches. Generally, the evaluation region has an important level of noise such as oil, grease, or dirt that makes an inaccurate recognition challenging [1, 4]. Also, the recognition of some specific parts is the first stage in the repair industry for having an accurate labor and parts evaluation where the presence of dissimilar car sizes, shapes, and models makes the task even more difficult for a model based on a machine learning strategy to perform well [5].

Nowadays, many machine learning algorithms have been broadly used in many industries to bring down the

charge of manual endeavors including object recognition [6, 7], exterior car body damage-detection [8], image encoding [9], and healthcare (organ, skeletal, body pose, tumor/cancer segmentation) [10, 11].

Detecting the damaged parts of the outer surface in various kinds of cars has received great attention in the field of machine vision in recent years, and many frameworks have been proposed for reducing the claim leak issue [2, 12, 13].

However, applying frameworks using ML strategies in this field is a very challenging task. This challenging task is because of some issues including light reflection, the presence of unidentified objects surrounding vehicles, scene illumination, and background detection [14, 15].

There are two main ML strategies for recognizing damaged outer parts of a car including hand-crafted feature extracting methods and deep learning models [8, 16].

Amirfakhrian et al. [3] proposed a clustering approach based on the fuzzy similarity criteria and changing the color space for recognizing the damaged parts of a car. They used a similarity score among two images that is computed using the color spectrum. Parhizkar et al. [8] suggested a cascade

convolutional neural network to recognize the damaged parts even in the presence of high illumination variation. Moreover, the Kirsch compass kernels are used to produce some edge maps for creating an encoded image. They used two textural descriptor approaches namely local binary pattern (LBP) and local directional number pattern (LDN) to obtain more informative features from the original image. Shirode et al. [17] suggested a deep learning model to recognize the damaged parts of a car using two separate CNN model. The first model (VGG16) is utilized to identify the damaged parts, locations, and their severities. The second model is able to mask out the precise damaged regions.

Our strategy to recognize the car damaged parts is based on learning a convolutional neural network (CNN) architecture utilizing the collected dataset, which consists of 3,000 images with different dimensions captured from different cameras.

The proposed CNN classifies all pixels inside the image into the background, damaged parts, and normal parts. All pixels including normal and damaged parts of the car have their own classes. For instance, we have two classes for normal Windshield and damaged Windshield. So, overall classes include 20 classes for car parts and one class for the background. All car parts have 10 categories including windshield, hood, front bumper, rear bumper, fenders, trunk lid, front doors, back doors, roof, and quarters.

To solve the problems of two-stage models (car detection and recognizing the damaged parts), we suggest a two-route CNN model for exploring both global and local features. The main contributions of this study are listed as follows:

- (1) A new two-route CNN model that automatically finds and localizes damaged parts of the car inside an input image
- (2) Employ a transfer learning approach to find more informative details from a real-scene image
- (3) Applying local and global patches to the CNN model for increasing the final performance of the model

2. Materials and Methods

In this part, our datasets and a detailed description of the model architecture are described. The suggested model is shown in detail in Figure 1.

2.1. Proposed Convolutional Neural Network. Convolutional neural network architectures are broadly utilized in the field of computer vision, such as medical image analysis, object detection, and action detection. In these networks, features and patterns inside the image can be explored by convolution operation [18–20]. The lower convolutional layers (Conv layers) can extract some features such as curves, lines, and edges. The deeper convolutional layers are able to learn more complex hidden patterns inside the image [21, 22].

The convolution operation in a Conv layer is implemented using a convolution filter (kernel), and its parameters are learned during the learning process. During the

convolving procedure, each filter is convolved with the input image to compute an informative feature map [23]. It should be mentioned that the dimension of the convolution filter is always smaller than the dimension of the input image. In another word, a convolution filter slides over an input image and calculates the dot product between the convolution filter and the input at each spatial position [1, 13, 24].

In this study, we employed two ResNet-50 models to explore low-level features, and three 3×3 new.mReLU blocks in each route are responsible for extracting high-level visual features. A residual neural network (ResNet) stacks residual blocks sequentially is an artificial neural network (ANN). ResNet-50 is 50 layers deep including 48 Conv layers, one maximum pooling, and one average pooling layer [25]. In order to recognize each part of the car more precisely, we need a network that is able to detect more informative features. So, we employed a pretrained ResNet-50 architecture that trained on ImageNet at the beginning of the framework [26, 27].

The extracted features of the ResNet-50 model are used as the input of the 3×3 mReLU block. In order to achieve a better segmentation result, three 3×3 mReLU blocks are utilized sequentially.

As the performance of the ResNet-50 has been proved in the field of image classification (extracting high-quality features of images on ImageNet), we employed this model to explore some informative details about the damaged car and background. Moreover, by applying deeper layers (more feature extraction layers), a better result in detecting damaged parts of a car will be obtained.

The employed 3×3 mReLU architecture is shown in detail in Figure 2. This model was inspired by the concept of learning multiple patterns using intermediate layers in a CNN model [25, 28, 29]. In other words, utilizing stacking up the mReLU pipeline is more efficient than a simple linear chain of convolution layers to classify a varying-scale object. In the suggested mReLU block, four 3×3 filters (small receptive fields) are used to solve the problem of overfitting and allow the framework to use a deeper architecture [28]. Also, the scale/shift layer is applied after the second concatenation layer to apply some trainable biases and weights [30, 31].

A negation layer is employed to multiply -1 into the output of the previous layer to enhance the exploration process of informative features. We applied three and two Conv layers before the negation layers for improving the performance of the feature extraction process. Moreover, some bottleneck layers (1×1 Conv layers) are used to reduce the computation procedure [31, 32].

Moreover, we proposed a new inception block that obtains the multiscale nature of car segmentation tasks and enhances the performance of the model when encountering a complex background [20, 30]. This proposed inception block in high detailed descriptions is indicated in Figure 3. This idea is inspired by some works conducted by [31, 32] that decreases the number of feature kernels in each layer. This decreasing the number of parameters leads to maintaining the sparsity of the architecture and improving the computational performance.

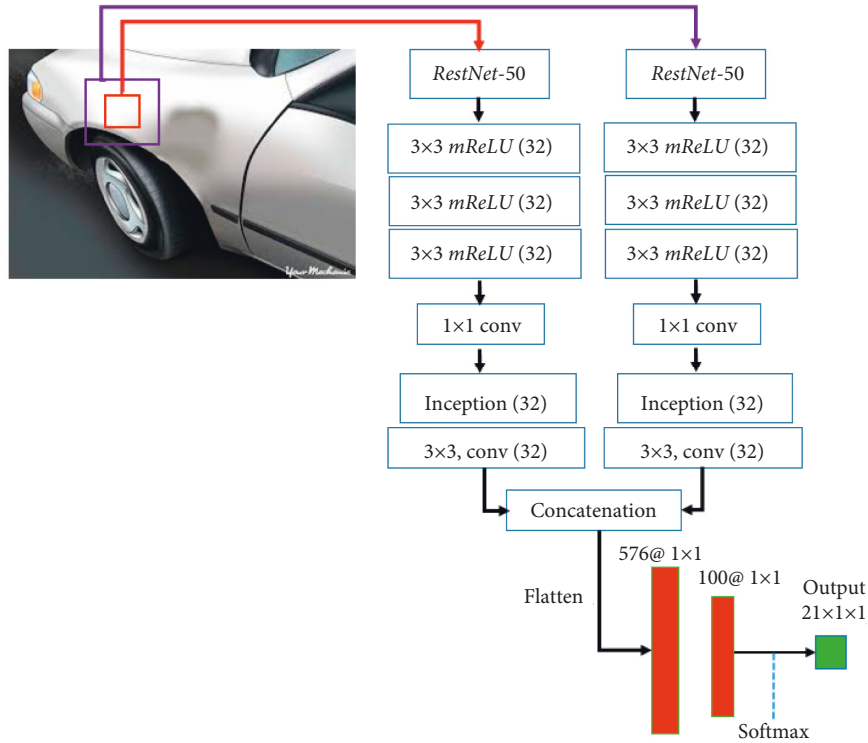


FIGURE 1: Our two-route CNN model using two ResNet-50.

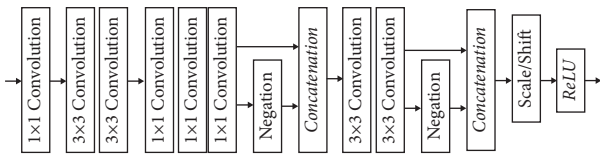


FIGURE 2: Our suggested $3 \times 3mReLU$.

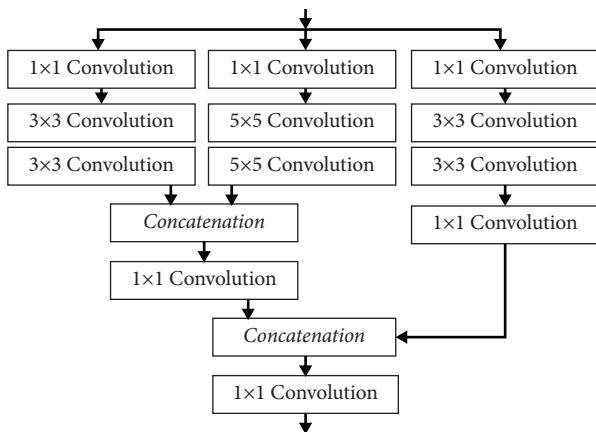


FIGURE 3: Our suggested inception block.

The inception block has three routes for extracting features. At the beginning of each route, a 1×1 convolutional layer is used and its output fed into two convolution layers. Then, the output of the first two routes is concatenated and is fed into another bottleneck

layer. Next, the output of routes one and two is concatenated with the third route. Due to the use of inception and $3 \times 3mReLU$ blocks, the overfitting problems have been satisfactorily addressed.

2.2. Data Augmentation. For effective learning and implementation, a CNN model needs to be trained with a large amount of the training data [18, 33]. The deep learning-based approaches need to be trained on large training datasets for avoiding overfitting and to maximize learning. Besides, the performance and learning accuracy of DL models are improved with ample and high-quality training data. Data augmentation (DA) techniques are used for changing or enhancing a dataset [34, 35].

In this paper, we use some DA techniques including geometric transformations (crop, rotate, and randomly flip), brightness transformation, mirror transformation, Gaussian noise, random elastic deformations, and random intensity variation [36–38].

3. Results and Discussion

The employing dataset includes 3,000 images of different sizes obtained using various cameras. All images were resized to the size of 620×620 before applying to the CNN model. The experiments were carried out employing Python on the NVIDIA Tesla K80 GPU, 8 GB RAM, and Windows 10. Our technique is compared with some studies in terms of car damage detection. We assess the accuracy of the suggested strategy with different criteria that are defined as follows [39–41].

$$\begin{aligned} \text{precision} &= \frac{\text{TP}}{\text{FP} + \text{TP}} \times 100\%, \\ \text{recall} &= \frac{\text{TP}}{\text{FN} + \text{TP}} \times 100\%, \\ \text{IoU} &= \frac{\text{area of overlap}}{\text{area of union}}, \end{aligned} \quad (1)$$

where false positive (FP) is the number of pixels that are incorrectly categorized as the body of a car while true positive (TP) implies the number of pixels that are correctly predicted by the suggested CNN framework. Also, a false negative (FN) indicates classified pixels that do not belong to the car; however, they are wrongly predicted as the car parts.

In Table 1, the results of detecting car parts are listed for 10 parts of the car. As it is clearly shown, the best outcomes for the precision are obtained for fenders, front doors, and windshield, and the worst results are related to quarters, roof, rear bumper, and back doors. For the recall assessment, the fenders, hood, and windshield gained the best results whereas the roof, front bumper, back doors, and quarters gained the worst results. Similar to the recall and precision outcomes, the best outcomes in terms of the IoU were achieved for front doors.

In Table 2, the comparison results for detecting car parts between our model and eight other approaches are listed. As it is clearly indicated, the proposed architecture gained the best results among all the eight other models. Moreover, the FCOMB [12] and VGG [1] strategies have the worst outcome among all the other techniques in terms of all evaluation criteria while the texture descriptor [8] pipeline is the second best model.

By looking deeply, we come to understand that the PANet [42] model indicates a very similar performance to the texture descriptor [8] model, and the VGG model [1] demonstrates an almost alike consequence to the combined feature (YOLOv3) [43] model. The differences between the obtained values of recall and precision criteria from the FCOMB [12] model and the suggested model are great numbers equal to 34% and 33%, respectively.

Table 3 exhibits the results of detecting car parts using the proposed model. The best outcomes in terms of the precision achieved for the trunk lid and front doors and the worst outcomes are related to the rear bumper. For the recall assessment criteria, the roof, trunk lid, windshield, and hood obtained the greatest results whereas the back doors and fenders achieved the worst results. Similar to the recall and precision outcomes, the best outcomes in terms of the IoU were achieved for the roof.

Table 4 indicates the comparison of results for detecting damaged parts of a car using our approach and some of the recently published papers in terms of Recall, Precision, and IoU. By comparing the gained results belonging to all models in Table 4, it is clear that the combined feature (YOLOv3) [43] and CNN [47] models gained the worst outcomes in terms of all criteria. In contrast, the HTC [46] and texture descriptor [8] pipelines obtained the next greatest values for all criteria. Nevertheless, the suggested two-route model

TABLE 1: Quantitative assessment based on recall, precision, and IoU for detecting different parts of a car.

Parts	Precision (%)	Recall (%)	IoU
Windshield	91	94	0.89
Trunk lid	88	92	0.87
Roof	87	89	0.86
Front bumper	88	90	0.83
Rear bumper	87	91	0.81
Front doors	92	92	0.90
Back doors	87	90	0.86
Quarters	87	90	0.85
Fenders	93	95	0.86
Hood	90	93	0.82

TABLE 2: Quantitative comparison between the proposed model and eight other models based on recall, precision, and IoU for detecting different parts of a car.

Architecture	Precision (%)	Recall (%)	IoU
PANet [42]	86	85	0.78
Combined feature (YOLOv3) [43]	66	62	0.61
VGG models [1]	64	62	0.60
FCOMB [12]	56	58	0.55
Improved mask RCNN [44]	69	72	0.64
Mask RCNN [45]	65	70	0.69
HTC [46]	83	84	0.76
Texture descriptor [8]	87	86	0.80
Proposed method	89	92	0.85

TABLE 3: Quantitative assessment based on recall, precision, and IoU for detecting different damaged parts of a car.

Parts	Precision (%)	Recall (%)	IoU
Windshield	93	94	0.91
Trunk lid	96	94	0.93
Roof	92	95	0.92
Front bumper	92	90	0.89
Rear bumper	91	92	0.88
Front doors	95	92	0.91
Back doors	94	90	0.91
Quarters	90	91	0.87
Fenders	93	90	0.88
Hood	92	94	0.89

TABLE 4: Quantitative comparison between the proposed model and eight other models based on recall, precision, and IoU for detecting different damaged parts of a car.

Architecture	Precision (%)	Recall (%)	IoU
PANet [42]	87	86	0.85
Combined feature (YOLOv3) [43]	67	68	0.64
CNN [47]	58	61	0.57
FFNN [48]	88	89	0.86
Mask RCNN [45]	84	81	0.78
VGG models [1]	85	84	0.81
HTC [46]	89	90	0.87
Texture descriptor [8]	90	91	0.88
Proposed method	93	92	0.90

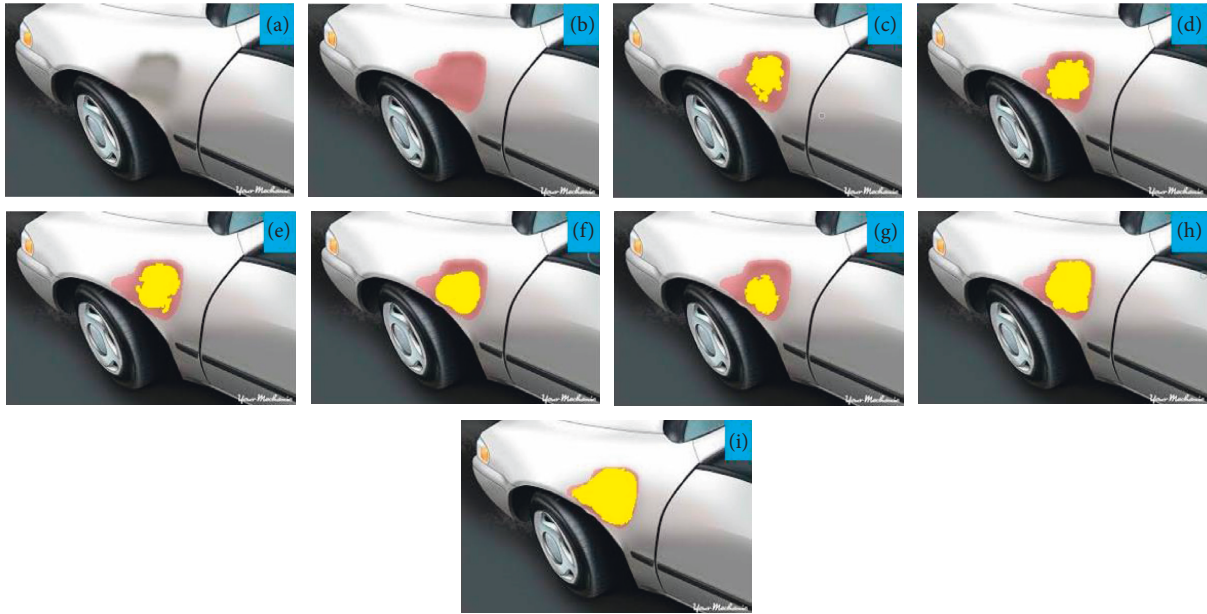


FIGURE 4: A visual comparison between our model and some other techniques. (a) Original image. (b) Labeled image. (c) YOLOv3. (d) Mask RCNN. (e) VGG. (f) FFNN. (g) CNN. (h) Texture descriptor. (i) Proposed model.

gained the best values among all compared models, which denotes its high effectiveness in the achievement of the desired objectives.

Figure 4 denotes the results of detecting a damaged part of a car using different models. As it is clearly shown, the damaged region of the car could not be recognized correctly by YOLOv3, Mask RCNN, FFNN, and CNN. For further explanation, the center region of the fender could not be appropriately recognized by the Mask RCNN, Yolov3, and CNN techniques. This difficulty was addressed during the implementation of the texture descriptor and VGG architectures. The segmentation outcomes obtained using FFNN and texture descriptor demonstrated an improvement in recognizing damaged regions in areas with high illumination variation and could reduce false positives.

4. Conclusion

In this paper, we suggested a deep learning-based model for recognizing damaged parts of the car. We employed a two-route CNN model that is able to extract both global and local features from the input image. In order to minimize the efficiency of the model for segmenting target parts, a pre-trained ResNet-50 model was used at the beginning of the pipeline. Moreover, an inception block and a 3×3 mReLU block were suggested to solve the overfitting. After analyzing the suggested framework, we realized that the outcomes of utilizing a pretrained CNN model that is applied to each route lead to a high segmentation performance with respect to some other models such as FFNN, YOLOv3, HTC, and VGG. To validate the suggested model, it was implemented on a private car collection database, and its outcomes from three measurement criteria were compared with some state-of-the-art techniques, including PANet [42], combined feature (YOLOv3) [43], CNN [47], FFNN [48], mask RCNN

[45], VGG models [1], HTC [46], and texture descriptor [8]. The results indicated that our framework demonstrated better achievements toward the comparative approaches in different terms of other comparative methodologies.

5. Disclosure

The funding sources had no involvement in the study design, collection, analysis or interpretation of data, writing of the manuscript, or in the decision to submit the manuscript for publication.

Data Availability

Data will be available upon request to the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] P. M. Woraratpanya and K. Woraratpanya, "Car damage detection and classification," in *Proceedings of the 11th International Conference on Advances in Information Technology*, Bangkok, Thailand, July 2020.
- [2] J. De Deijn, *Automatic Car Damage Recognition using Convolutional Neural Networks*, M.S. thesis, 2018.
- [3] M. Parhizkar and M. Parhizkar, "Integration of image segmentation and fuzzy theory to improve the accuracy of damage detection areas in traffic accidents," *Journal of Big Data*, vol. 8, no. 1, 147 pages, 2021.
- [4] K. Pasupa, P. Kittiworapanya, N. Hongngern, and K. Woraratpanya, "Evaluation of deep learning algorithms for semantic segmentation of car parts," *Complex & Intelligent Systems*, pp. 1–13, 2021.

- [5] P. Rakshata, "Car damage detection and analysis using deep learning," *Algorithm For Automotive*, vol. 5, no. 6, pp. 1896–1898, 2019.
- [6] A. Uhlmann, Z. Uhlmann, N. F. Glenn et al., "Tree canopy and snow depth relationships at fine scales with terrestrial laser scanning," *The Cryosphere*, vol. 15, no. 5, pp. 2187–2209, 2021.
- [7] M. Wilmanski, C. Kreucher, and J. Lauer, "Modern approaches in deep learning for SAR ATR," *Algorithms for Synthetic Aperture Radar Imagery XXIII*, vol. 9843, Article ID 98430N, May 2016.
- [8] M. Amirfakhrian and M. Amirfakhrian, "Car detection and damage segmentation in the real scene using a deep learning approach," *International Journal of Intelligent Robotics and Applications*, vol. 6, no. 2, pp. 231–245, 2022.
- [9] Y. Pourasad, R. Ranjbarzadeh, and A. Mardani, "A new algorithm for digital image encryption based on chaos theory," *Entropy*, vol. 23, no. 3, p. 341, 2021.
- [10] S. Jafarzadeh Ghouschi, R. Ranjbarzadeh, S. A. Najafabadi, E. Osgoee, and E. B. Tirkolaei, "An extended approach to the diagnosis of tumour location in breast cancer using deep learning," *Journal of Ambient Intelligence and Humanized Computing*, 2021.
- [11] S. B. Ranjbarzadeh, R. Ranjbarzadeh, A. Amirabadi et al., "Osteolysis: a literature review of basic science and potential computer-based image processing detection methods," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–21, Article ID 4196241, 2021.
- [12] B. Balci, Y. Artan, B. Alkan, and A. Elihos, "Front-view vehicle damage detection using roadway surveillance camera images," *Proceedings of the 5th International Conference on Vehicle Technology and Intelligent Transport Systems*, pp. 193–198, 2019.
- [13] C. T. Kaya and T. Kaya, "Car damage analysis for insurance market using convolutional neural networks," *Intelligent and Fuzzy Techniques in Big Data Analytics and Decision Making*, vol. 1029, pp. 313–321, 2020.
- [14] S. Jayawardena, *Image Based Automatic Vehicle Damage Detection*, 2013.
- [15] T. Kumar, J. Park, M. S. Ali, A. F. M. Uddin, J. H. Ko, and S.-H. Bae, "Binary-Classifiers-Enabled Filters for Semi-supervised Learning," *IEEE Access*, vol. 9, pp. 167663–167673, 2021.
- [16] C. Wang, G. Yan, S. Yu et al., "Fast vehicle and pedestrian detection using improved mask R-CNN," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–15, Article ID 5761414, 2020.
- [17] A. Shirode, T. Rathod, P. Wanjari, and A. Halbe, "Car damage detection and assessment using CNN," *2022 IEEE Delhi Section Conference (DELCON)*, vol. 2022, Article ID 9752971, 2022.
- [18] R. Dorosti, S. Jafarzadeh Ghouschi, S. Safavi et al., "Nerve optic segmentation in CT images using a deep learning model and a texture descriptor," *Complex & Intelligent Systems*, pp. 1–15, 2022.
- [19] K. Kumar and A. Kumar, "Cross-spectral iris recognition using CNN and supervised discrete hashing," *Pattern Recognition*, vol. 86, pp. 85–98, Feb. 2019.
- [20] C. Szegedy, L. Wei, J. Yangqing et al., "Going Deeper with Convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Boston, MA, USA, June 2015.
- [21] S. Vitale, G. Ferraioli, and V. Pascasio, "Multi-objective CNN-based algorithm for SAR despeckling," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 11, pp. 9336–9349, 2021.
- [22] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the - IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2921–2929, December 2016.
- [23] S. Baseri Saadi, N. Tataei Sarshar, S. Sadeghi, R. Ranjbarzadeh, M. Kooshki Forooshani, and M. Bendeche, "Investigation of effectiveness of shuffled frog-leaping optimizer in training a convolution neural network," *Journal of Healthcare Engineering*, vol. 2022, pp. 1–11, Article ID 4703682, 2022.
- [24] K. Yasaka, H. Akai, O. Abe, and S. Kiryu, "Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced CT: a preliminary study," *Radiology*, vol. 286, no. 3, pp. 887–896, 2018.
- [25] F. Naiemi, V. Ghods, and H. Khalesi, "A novel pipeline framework for multi oriented scene text image detection and recognition," *Expert Systems with Applications*, vol. 170, pp. 114549–122021, 2021.
- [26] Z. Tian, M. Shu, P. Lyu et al., "Learning Shape-Aware Embedding for Scene Text Detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
- [27] T. He, W. Huang, Y. Qiao, and J. Yao, "Text-attentional convolutional neural network for scene text detection," *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2529–2541, 2016.
- [28] S. Hong, B. Roh, K.-H. Kim, Y. Cheon, and M. Park, "PVANet: Lightweight Deep Neural Networks for Real-Time Object Detection," 2016, <http://arxiv.org/abs/1611.08588>.
- [29] A. Valizadeh, S. Jafarzadeh Ghouschi, R. Ranjbarzadeh, and Y. Pourasad, "Presentation of a segmentation method for a diabetic retinopathy patient's fundus region detection using a convolutional neural network," *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–14, Article ID 7714351, 2021.
- [30] C. Szegedy, V. Vanhoucke, S. Ioffe, and J. Shlens, "Rethinking the Inception Architecture for Computer Vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, June 2016.
- [31] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, "Inception recurrent convolutional neural network for object recognition," *Machine Vision and Applications*, vol. 32, no. 1, pp. 28–44, 2021.
- [32] R. F. Aljehane and N. O. Aljehane, "An optimal segmentation with deep learning based inception network model for intracranial hemorrhage diagnosis," *Neural Computing & Applications*, vol. 33, no. 20, pp. 13831–13843, 2021.
- [33] A. Chandio, Y. Shen, M. Bendeche, I. Inayat, and T. Kumar, "AUDD: audio Urdu digits dataset for automatic audio Urdu digit recognition," *Applied Sciences*, vol. 11, no. 19, p. 8842, 2021.
- [34] F. Zhan, H. Zhu, and S. Lu, "Scene Text Synthesis for Efficient and Effective Deep Network Training," 2019, <https://arxiv.org/abs/1901.09193v1>.
- [35] M. R. Matin and A. Matin, "Detection of COVID 19 from CT image by the novel LeNet-5 CNN architecture," in *Proceedings of the 2020 23rd International Conference on Computer and Information Technology (ICCIT)*, December 2020.
- [36] F. Wang, S. H. Zhong, J. Peng, J. Jiang, and Y. Liu, "Data augmentation for eeg-based emotion recognition with deep convolutional neural networks," *MultiMedia Modeling*, vol. 10705, pp. 82–93, 2018.

- [37] D. Abdelhafiz, C. Yang, R. Ammar, and S. Nabavi, "Deep convolutional neural networks for mammography: advances, challenges and applications," *BMC Bioinformatics*, vol. 20, no. S11, pp. 281–320, 2019.
- [38] C. Khoshgoftaar and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 60–48, Dec. 2019.
- [39] R. Jafarzadeh Ghouschi, S. Bendeche, M. Amirabadi et al., "Lung infection segmentation for COVID-19 pneumonia based on a cascade convolutional network from CT images," *BioMed Research International*, vol. 2021, pp. 1–16, Article ID 5544742, 2021.
- [40] N. Karimi, R. Ranjbarzadeh Kondrood, and T. Alizadeh, "An intelligent system for quality measurement of Golden Bleached raisins using two comparative machine learning algorithms," *Measurement*, vol. 107, pp. 68–76, Sep. 2017.
- [41] R. Saadi and S. B. Saadi, "Automated liver and tumor segmentation based on concave and convex points using fuzzy c-means and mean shift clustering," *Measurement*, vol. 150, Article ID 107086, 2020.
- [42] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, *Path Aggregation Network for Instance Segmentation*, pp. 8759–8768, 2018.
- [43] H. S. Malik, M. Dwivedi, S. N. Omakar, S. R. Samal, A. Rathi, and E. B. Monis, *EasyChair Preprint Deep Learning Based Car Damage Classification and Detection*, 2020.
- [44] Q. Zhang, X. Chang, and S. B. Bian, "Vehicle-damage-detection segmentation algorithm based on improved mask RCNN," *IEEE Access*, vol. 8, pp. 6997–7004, 2020.
- [45] A. Patil, "Car damage recognition using the expectation maximization algorithm and mask R-CNN," *Information and Communication Technology for Intelligent Systems*, vol. 196, pp. 607–616, May 2020.
- [46] K. Chen, "Hybrid task cascade for instance segmentation," pp. 4974–4983, 2019, <https://github.com/>.
- [47] K. Patil, M. Kulkarni, A. Sriraman, and S. Karande, "Deep learning based car damage classification," *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, vol. 2017, pp. 50–54, 2017.
- [48] A. F. Agarap, "Deep Learning Using Rectified Linear Units (ReLU) Deep Learning Using Rectified Linear Units (ReLU)," pp. 2–8, 2020, <https://arxiv.org/abs/1803.08375>.