Deep Metric Learning Framework for Vehicle Detection and Tracking

Sreehari Patibandla [1], Dr. Maruthavanan Archana [2], Dr. Rama Chaithanya Tanguturi [3]
[1] Research Scholar, Department of Computer Science & Engineering, Annamalai University, Chidambaram
Tamil Nadu,608001, India.
[2] Assistant Professor, Department of Information Technology, Annamalai University, Chidambaram
Tamil Nadu, 608001, India.
[3] Professor, Department of Computer Science & Engineering, PACE Institute of Technology and Sciences, Ongole, Andhra Pradesh, 523272, India.
[1] sreehari.44@gmail.com, [2] archana.aucse@gmail.com, [3] trchaitanya@gmail.com

Article Info
Page Number: 3463-3471
Publication Issue:
Vol. 71 No. 4 (2022)

Article History
Article Received: 25 March 2022
Revised: 30 April 2022
Accepted: 15 June 2022
Publication: 19 August 2022

Abstract
The identification of vehicles is a challenging task; the gathered information may produce the deviation from mathematical computations. The deep metric learning framework is proposed to detect and classify the vehicles; the proposed model has the process of bounding box construction, vehicle tracking and detection. The embedding vector is used to process the input images and fed into a convolutional neural network for obtaining the vehicle tracking process with the computed L_2 distance within the vehicles and the measurement of the similarity degree between the vehicles. BIT Vehicle Dataset is involved for testing the proposed framework; the dataset has the different kind of vehicle images for performing the evaluation process.

Keywords: Deep Metric Learning, Vehicle detection, Vehicle Tracking, Embedding vector, BIT Vehicle Dataset

1 Introduction

The intelligent traffic maintenance system is an emerging technique which provides an innovative framework for improving the secured transportation. The vehicle classification in traffic system has several applications and the vehicle tracking has a challenging process as the images are gathered from several sources [1]. The vehicle types produce various similarities that reduce the performance of the vehicle classification process. The vehicle identification and tracking process has several issues and it could be reduced using the advanced machine learning technique for performing better classification process [2]. The dataset has the capability of gathered vehicle images using the improved histogram process for detecting the vehicles from the denoised images [3]. The vehicle detection techniques have been developed for providing the highest accuracy, easy way of implementation, and image adaptation. The vehicle classification with traffic flow data has produced the analysis of the planning environment [4]. Additionally, the traffic flow analysis system has produced the monitoring system for understanding the traffic violations and poor traffic management process [5]. The real-time traffic flow data could not be generated through the inefficient vehicle classification system, so the enhanced vehicle tracking process has been framed to overcome this issue [6]. Moreover, artificial intelligent based techniques, machine learning enabled methods are utilized for tracking and classification of vehicles from input images. Vehicle detection is the procedure to recognize the object type with location in an input object [7]. The detection-based algorithms have been segregated into artificial intelligence-based techniques. The techniques have the requirement of knowledge to identify the features of the objects such as the determination of the perfectly matched features of the input frame [8]. Video tracking is the concept of identifying objects and marking the matching frames on successive frames. The bounding box data is normally utilized for tracking the objects and extracting the objects according to the shape and pixel entities. The filter tracking procedures are involved with the bounding box for providing the result in a speed manner. Additionally, vehicle tracking is the challenging process of conventional image processing technique due to the problems of occlusion and low-quality images and the proposed technique has to be developed for providing solutions to the problems.

2 Related Work

Vehicle detection techniques have been used as the object features; the vector is extorted from the input image pixels. The detection process has recognized the vehicles consistent with the shape and characteristics as the background modelling has removed the background. The remaining positions are identified as the important features are determined through the classification procedure for vehicle type identification. The classification methods like occupancy grid [9], dynamic background modelling [10], and tracking pixels [11] have been identified for enhancing the classification results in terms of shape and appearance. The development of feature descriptors has produced the hard for designing to identify the similar vehicles. The machine learning enabled algorithms like Deep Neural Networks (DNN) [12], ResNet [13], and YOLO [14] have been designed to overcome the classification issues and detect vehicles through hidden features with enhanced accuracy. The traffic flow monitoring process has several applications to maintain the detection of vehicles within the specified time. The Vehicle tracking has been done through the extraction of the vehicle trajectory within the specified interval time in the category of direction, speed and entry points of the vehicles. The bounding box features like filter and optical flow which is used to provide the solution to the tracking issues. The optical flow enabled techniques have been provided under the slow process as the Kalman filter technique resists several wrong predictions. Additionally, the tracking results of the trackers has been reduced whenever the total objects enhance within the specific frame and it produces $O(n^3)$ computations in every bounding box that will process the computation by parallel processing methodology. The maintenance of robustness in vehicle detection has been maintained using Faster R-CNN technique [15] as the total pixels are observed with statistical correlation.

3 Proposed Work

The proposed framework for vehicle identification is demonstrated in Fig. 1, which has the process of extracting the vehicle using the embedding vector and the similarity measurement within the vehicle that has been analyzed through the reidentification procedure for tracking the vehicles and storing the vehicle information. In spite of capturing the vehicles, the field of view concept has the issue of overlapping in the longitudinal direction, the dataset images are used to detect the vehicle and provide the information required for reconstructing the bounding box. The identification and tracking of vehicles utilizing the Deep Metric Learning based embedded vectors with the computed L_2 distance within the vehicles and the measurement of the similarity degree between the vehicles.

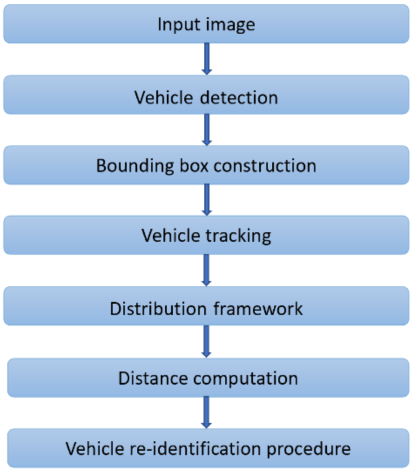

Fig. 1: Proposed image retrieval framework for an RGB image.

The identification of a vehicle through a single capture is based on several angles and locations, the pixel coordinates of the specific points have been extracted, the intrinsic matrix is framed for obtaining the projection, the higher calibration is needed for completing the process. The bounding box is used for obtaining the robust framework of vehicle detection; the dataset enabled training framework is developed. The vehicle size and spatial area has been identified through the bounding box framework concept, the vehicle tracking process has ensured the total amount of similar vehicles in a unique way of successive frames while the vehicles are required to be tracked. The Hungarian procedure is utilized to establishing matching relationship within the tracking and detection box in consecutive frames. The identification of vehicles in a single frame could be analyzed through the short span components that are used for obtaining the vehicles and capturing the data in several ways.
The deep metric learning procedure is implemented to several applications in image classification environment, the restricted capability for representing the nonlinear elements of information has been enabled through the projection model. The mapping process of vehicle images has been involved in the feature spaces and parameters as the feature extraction is needed. The dissimilar vehicle images are combined into the instances of the Convolutional network while the embedding vectors to the specific vehicle have been extracted. The distance within 2 vehicles could be obtained using the L_2 distance framework as the sample values is demonstrated as $Ve^-, Ve,\ Ve^+$ and the Convolutional network is demonstrated as $CN(Ve)$, the framework is computed in Eq. (1).

$$TN(\,Ve, Ve^-, Ve^+) = \begin{bmatrix} ||CN(Ve) - CN(Ve^-)||_2 \\ ||CN(Ve) - CN(Ve^+)||_2 \end{bmatrix} \qquad (1)$$

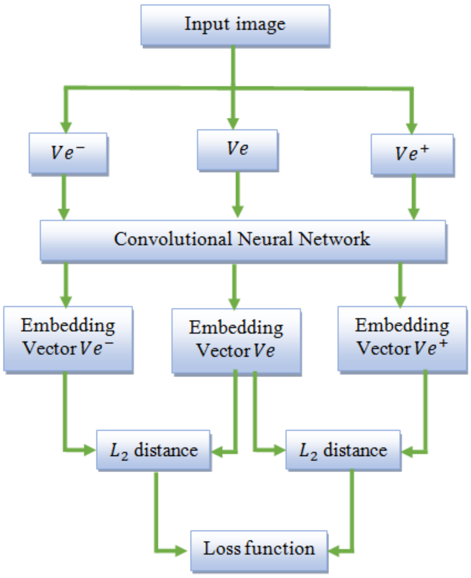

Fig. 2: Deep Metric Learning framework

In the $TN(Ve, Ve^-, Ve^+)$, Ve and Ve^+ demonstrate the similar vehicle, where Ve^- is the remaining vehicle. The distance parameter is utilized for measuring the similarity within the vehicles; the minimum distance will make more similarity of the vehicles. The loss function is used to measure the entire learning procedure in Eq. (2).

$$LF(M,b) = \sum_i \left[||CN(Ve_i) - CN(Ve_i^+)||_2 - ||CN(Ve_i) - CN(Ve_i^-)||_2 \right] \qquad (2)$$

Where M represents the matrix, b is the bias value, γ denotes the minimum value within the vehicles and i is the total number of samples, while the smaller amount of loss function will enhance the performance of the proposed framework. The input image has been classified using the embedding vector whenever the distance within the images of the similar vehicle is having the minimum value, the distance within the images of dissimilar vehicles is higher, the entire framework for Deep metric learning is demonstrated in Fig. 2.

The embedding vector is used for detecting the vehicles from the input images, the vehicle type is classified by detecting the object that demonstrate the spatial data through the probability function, it is combined through several components for identifying vehicles and it is computed in Eq. (3).

$$\Pr(x') = \sum_{i=1}^{n} \theta_i \delta_i (x' | \beta_i, C_i) \qquad (3)$$

Where $\delta_i(x'|\beta_i, C_i)$ demonstrates the bivariate distribution on the vector β_i, θ_i demonstrates the probability function, C_i denotes the covariance matrix. The training procedure is increased through the visualization technique which classifies the images into various forms according to the derivate operators in several directions. The images are decomposed into various categories based on the tuning process and the filter orientation with linear combination. The feature descriptor is the combination of extracting images from the detected image, the pixel intensity is computed in Eq. (4).

$$\tau(x_c) = \arctan\left[\sum_{i=0}^{n}\left(\frac{x_i - x_c}{x_c}\right)\right] \qquad (4)$$

Where $\tau(x_c)$ is the excitation value of the current pixel x_c, n is the total adjacent pixels, x_i demonstrates the adjacent pixel of x_c, the gradient component is computed in Eq. (5).

$$\pi(x_c) = \arctan\left(\frac{v_s^n}{v_s^{n-1}}\right) \qquad (5)$$

The extracted feature vectors are combined with the current feature vectors for classifying the vehicles. The bounding box enabled tracking procedure has traced the vehicles with bounding box data (X, Y, Wt, Ht) of the adjacent frames, the bounding box data is defined in Eq. (6) and Eq. (7).

$$Bb_i = [[X_0^{ctr}, Y_0^{ctr}, Wt_0, Ht_0], \ldots\ldots, [X_n^{ctr}, Y_n^{ctr}, Wt_n, Ht_n]] \qquad (6)$$

$$Bb_j = [[X_0^{ctr}, Y_0^{ctr}, Wt_0, Ht_0], \ldots\ldots, [X_m^{ctr}, Y_m^{ctr}, Wt_m, Ht_m]] \qquad (7)$$

The similarity distance is computed in Eq. (8).

$$Dis_{ij} = \sqrt{\sum_i \sum_j (Bb_i - Bb_j)^2} \qquad (8)$$

Where Bb_i, Bb_j holds the bounding box data for the traced vehicles from the previous to latest frames and the measured distance is stored in an array in Eq. (9).

$$Dis_{ij} = \begin{pmatrix} dis_{11} & \ldots & dis_{1n} \\ \ldots & \ldots & \ldots \\ dis_{m1} & \ldots & dis_{mn} \end{pmatrix} \qquad (9)$$

The computed distances within the objects are updated with the vehicle state using the bounding box coordinates. The system computes the total frame of the object has been predicted through the bounding ox data, while the threshold value maintains the robustness of the tracking procedure by identifying the remaining states of the objects whenever the object are facing the issues of occlusion and illusion, the vehicle tracking updates the prediction process by estimating the remaining states of the vehicles with non-linear function.

4 Performance Evaluation

The proposed framework is tested through the BIT Vehicle Dataset [16] which has 9900 images with dissimilar pixel sizes and it is captured by dissimilar cameras. The proposed DML (Deep Metric Learning) technique is compared with the related techniques of DNN (Deep neural Network) [12], ResNet (Residual Neural Network) [13], and YOLO (You Only Look Once) [14]. It contains several categories of vehicles like trucks, buses, vans, and cars. The captured images are in various views, location and illumination parameters. The further evaluation and capability of the proposed DML framework is measured through the similarity within the vehicles as several angles has been identified from the dataset images. Every vehicle type has the similarity of the vehicles with different viewpoints and the classification accuracy has been measured, the result is demonstrated in Fig. 3.

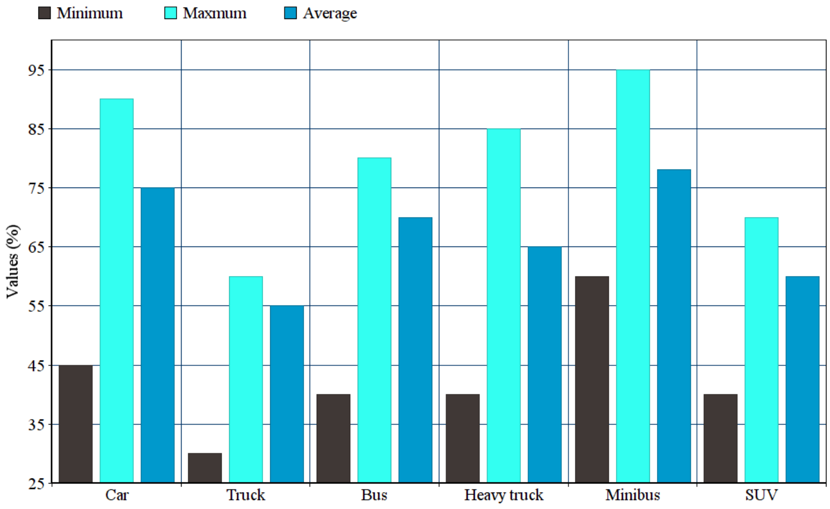

Fig. 3: Vehicle Identification Accuracy

The weighted average is used for measuring the accuracy while the vehicle distribution types of vehicles haven't similar to others. The proposed technique has increased the accuracy by minimizing the generalization of the input data and identifies every case of the training set. The shape and pixel features of every type of vehicle are similar to each other and the nonlinear distribution of vehicle information has enhanced accuracy and it is demonstrated in Fig. 4.

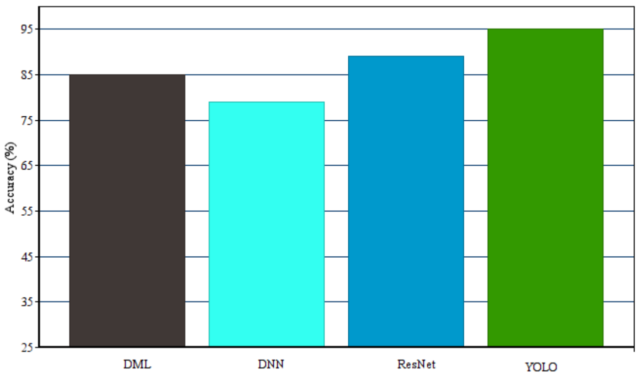

Fig. 4: Accuracy (%)

The overfitting has affected the performance of the vehicle detection and tracking process as the proposed technique is constructed to reduce it. The training time for the proposed technique is measured and compared with the related techniques, it is observed that the proposed technique is used under the reduced training time demonstrated in Fig. 5.

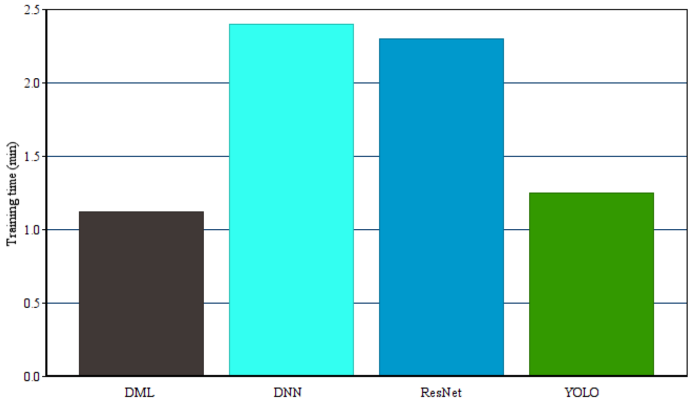

Fig. 5: Training time

The precision value is computed from the correctly predicted values among the total values, recall is the measurement of the positive values from the entire precision value. F1-score is calculated as the mean precision, weighted values of recall and the precision that the false positive values are utilized for distributing the values. The accuracy is computed as the rate of the correctly predicted value from the total value and it is illustrated in Fig. 6. The performance comparison for vehicle tracking is demonstrated in Fig. 7.

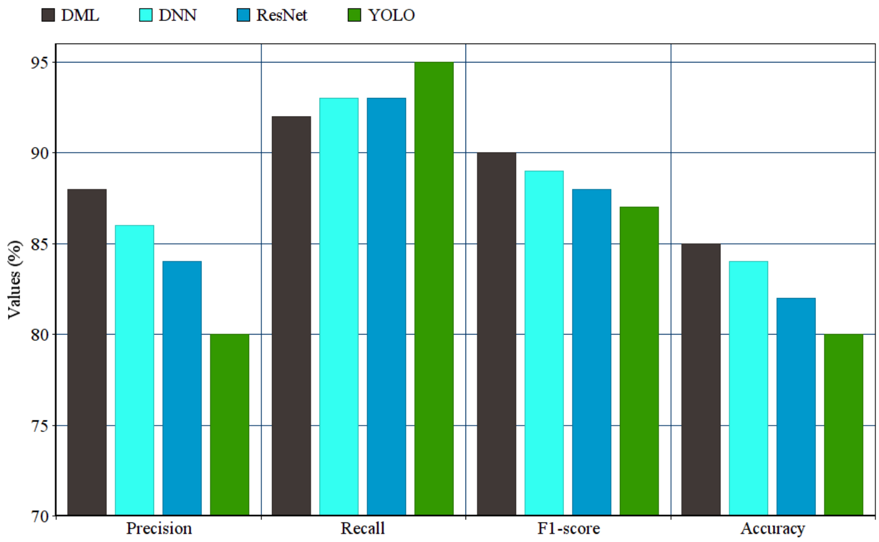

Fig. 6: Performance comparison

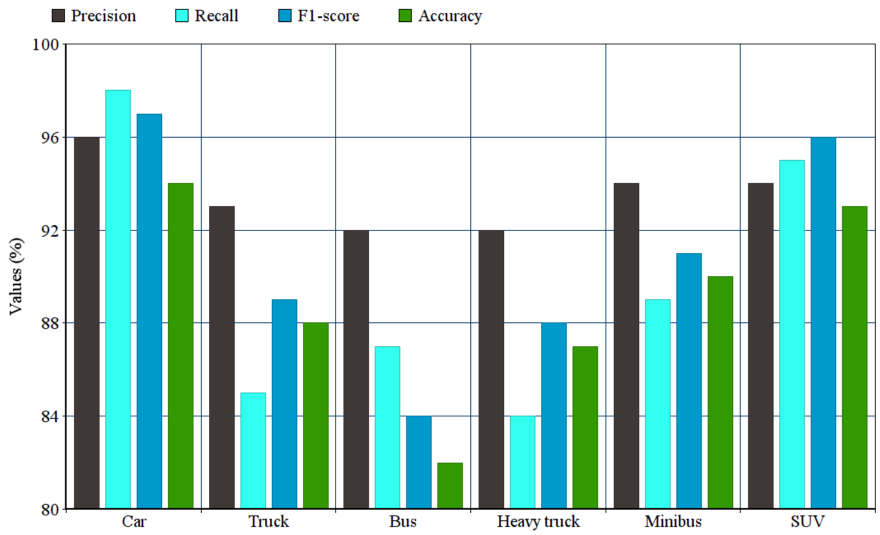

Fig. 7: Vehicle tracking

5 Conclusion

The paper has proposed the Deep Metric Learning Framework (DML) which transfers the information of the vehicle image into different levels of dimensions and generates the embedding vectors. The bounding box construction is used for vehicle tracking and detection through the L_2 distance computation across the vehicles and the similarity degree measurement from the BIT Vehicle Dataset. The proposed technique is compared with other methodologies in terms of accuracy, training time and the performance results proved that the proposed DML technique is performed well.

References

1. F. Li, C.-H. Lee, C.-H. Chen, L.P. Khoo, Hybrid data-driven vigilance model in traffic control center using eye-tracking data and context data, Adv. Eng. nform. 42 (2019) 100940.
2. M. Fernandez-Sanjurjo, B. Bosquet, M. Mucientes, V.M. Brea, Real-time visual detection and tracking system for traffic monitoring, Eng. Appl. Artif. Intell. 85 (2019) 410–420.
3. Z. Yang, L.S. Pun-Cheng, Vehicle detection in intelligent transportation systems and its applications under varying environments: A review, Image Vis. Comput. 69 (2018) 143–154.
4. X. Feng, Y. Jiang, X. Yang, M. Du, X. Li, Computer vision algorithms and hardware implementations: A survey, Integration 69 (2019) 309–320.
5. S. Khan, H. Ali, Z. Ullah, M.F. Bulbul, An intelligent monitoring system of vehicles on highway traffic, in: 2018 12th International Conference on Open Source Systems and Technologies (ICOSST), 2018, pp. 71–75.
6. S. Du, S. Razavi, Fault-tolerant control of variable speed limits for freeway work zone using likelihood estimation, Adv. Eng. Inform. 45 (2020) 101133.
7. N.K. Chauhan, K. Singh, A review on conventional machine learning vs deep learning, in: 2018 International Conference on Computing, Power and Communication Technologies (GUCON), 2018, pp. 347–352.
8. M.B. Khalkhali, A. Vahedian, H.S. Yazdi, Vehicle tracking with kalman filter using online situation assessment, Robot. Autonomous Syst. (2020) 103596.
9. H. Badino, U. Franke, R. Mester, Free space computation using stochastic occupancy grids

and dynamic, in: Programming, Proc. Intl Conf. Computer Vision, Workshop Dynamical Vision, 2007.
10. Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
11. R. Girshick, Fast r-cnn, in: 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440–1448.
12. M. Shah and R. Kapdi, "Object detection using deep neural networks," 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), 2017, pp. 787-790.
13. M. F. Haque, H. Lim and D. Kang, "Object Detection Based on VGG with ResNet Network," 2019 International Conference on Electronics, Information, and Communication (ICEIC), 2019, pp. 1-3.
14. J. Redmon, A. Farhadi, Yolov3: An incremental improvement. arxiv 2018, arXiv preprint arXiv:1804.02767 (2018) 1–6.
15. B. Zhang, L. Zhou, J. Zhang, A methodology for obtaining spatiotemporal information of the vehicles on bridges based on computer vision, Computer- Aided Civil and Infrastructure Engineering 34 (6) (2019) 471–487.
16. https://www.programmersought.com/article/7654351045/