Mapana - Journal of Sciences
2023, Vol. 22, Special Issue 2, 245-260
ISSN 0975-3303|https://doi.org/10.12723/mjs.sp2.13

Machine Learning based Vehicle Counting and Detection System

Jayamalar T*, Krishnaveni N†

Abstract

The study of how machines perceive instead of humans is known as vehicle detection or computer vision object identification. The primary purpose of a vehicle detection system is to identify one or multiple vehicles within the input images and live video feed. The dataset is used to train image processing algorithms for tasks like detection and tracking. To pinpoint the defects and strength of each image processing system, assessment criteria are used to develop, train, test, and compare them. To recognize, track, and count the vehicle in images and videos, the image processing algorithms such as CNN YOLOv3 and SVM are implemented. The main goal and intention of this work is to develop a system that can intelligently identify and track automobiles in still images and moving movies. The results demonstrated that CNN-based YOLOv3 does a good job of detecting and tracking vehicles.

* 1,2Avinashilingam Institute for Home Science and Higher Education for women, Coimbatore, India jayamalarit.avi@gmail.com,

* krishnaveni.adu@gmail.com

Keywords: Vehicle Detection, YOLOv3, SVM, Histogram, Sliding Window

1 Introduction

The primary objective of this endeavor is to create a system capable of identifying and monitoring vehicles in images and videos, regardless of whether the vehicles are stationary or in motion. This system will employ vehicle detection techniques on roadways for purposes such as vehicle monitoring, counting, traffic analysis, and vehicle categorization across various scenarios and applications. Vehicle detection and counting is a useful traffic analytics technique that may be used on highways and city streets in a variety of weather and traffic conditions. The system may be permanently established using existing or newly installed video cameras. To identify and categorize automobiles, a variety of approaches and procedures can be employed. CNN-YOLOv3 and SVM are used in this study. Contactless detection and classification of numerous vehicle kinds (truck, bus, automobile, bike, etc). To implement YOLOv3 classifier and SVM, These models have the capability to anticipate the image's category, distinguishing between whether it portrays a vehicle or something other than a vehicle. The algorithms used in these approaches include YOLOv3 and SVM. Among others, the CNN algorithm namely YOLOv3 and SVM are used to detect and classify the vehicles in real time images and video. There should be a need for improvement in the existing methods of vehicle detection and counting, so this research work aims to detect vehicles in an automated manner.

2 Literature Study

The objective of the vehicle detection system, as outlined in reference [5], is to locate one or multiple cars within the input image by employing a sliding window approach. In vehicle detection systems, the sliding window technique is employed in two ways. According to [11], the vehicle detection system applies local characteristics using sliding windows and Heat map. The capacity to identify the vehicle using the sliding window and HOG is also possible [18]. The dataset is the image of an unmanned aerial vehicle. The supplied image may be shown in many windows of varied sizes. Edge detection can be applied to discover objects in UAV

photos [24] and track the vehicles. It shows how to use edge detection to identify fake things. To discover straight lines on a vehicle, edge detection technique is employed. With the help of a larger threshold, the software suppresses background noises first.

As stated in reference [26], the process of distinguishing between artificial and natural objects involves the utilization of color features and edge detection. This distinction is achieved by extracting nine attributes from the multiple color channels within the source image. These qualities are utilized to distinguish between manufactured and natural items by defining the edges and changes. The real geo-referenced images and non geo-referenced aircraft photographs shot in the same place at different times are used [2]. The extraction of shadow and surface highlights are employed to construct a milestone recognition framework. These attributes are used in the recommended approach to offer information about surface introduction, shape, and shading.

The vehicle identifying thickness estimation approach is proposed by [8]. The angle vectors were established in the edge guide of the raised images. At the objective’s limit, the heads of the inclination vectors should fundamentally alter, and the calculation of the standard deviation for the slope vectors is necessary. Consequently, with a predefined threshold, it becomes possible to detect vehicles within this measurement.

The study utilized aerial photographs captured from a Turkish street, achieving 86% accuracy in F-measure. According to [5], the vehicle localization in provincial circumstances is implemented using two stages. In the first stage, a Harris corner indicator is employed to indicate places of interest for the images. A successful sliding window approach is then implemented. According to [5], vehicle localization in provincial circumstances is implemented using two stages. In the first stage, a Harris corner indicator is employed to indicate places of interest for the images. A successful sliding window approach is then implemented.

3 Methodology

The system has the capability to identify and monitor vehicles within video frames, followed by categorizing them into three distinct sizes depending on their dimensions. As depicted in Figure 1, the proposed system consists of three components: background learning, foreground extraction, and vehicle classification. Background subtraction, a conventional method for extracting the foreground image or detecting moving objects, is employed. As can be seen in the diagram, real-time video and photos are used as input data for vehicle detection. The input video is read frame by frame, and each frame is preprocessed. To extract the features that comprise vehicle information, HOG, heat map, and sliding window are used. The two models, CNN and SVM, will be trained using the extracted data. The car is detected after model training. The identified vehicles are processed once more to draw a box around them in order to track them in the input video. Vehicles are tallied based on class-id, vehicle box coordinates, and confidence ratings using non-max suppression.

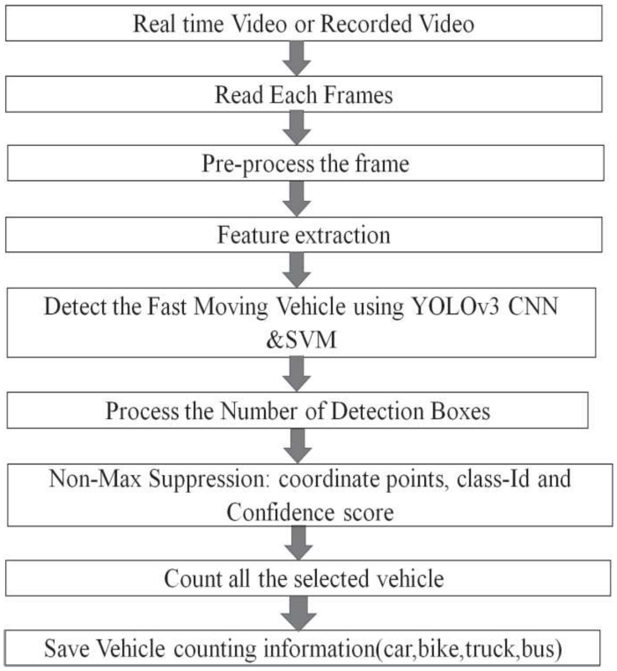

Real time Video or Recorded Video
Read Each Frames
Pre-process the frame
Feature extraction
Detect the Fast Moving Vehicle using YOLOv3 CNN &SVM
Process the Number of Detection Boxes
Non-Max Suppression: coordinate points, class-Id and Confidence score
Count all the selected vehicle
Save Vehicle counting information(car,bike,truck,bus)

Fig. 1. Model development summary

Color Histogram

In the realm of photography, a histogram serves as a portraying visually by illustrating the quantity of pixels within an image falling within a specific range of brightness or color. A traditional luminance histogram, for instance, displays the pixel count for each level of luminance or brightness, ranging from black to white. In this type of histogram, the larger the peak on the graph, the greater the number of pixels at that particular brightness level. Similarly, a color histogram adheres to the identical underlying concept, but instead of depicting shades of black, it reveals the pixel distribution among the three primary colors. Essentially, it's a graphical representation that showcases the color intensity of each RGB (Red, Green, Blue) color channel individually. Figure 2 represents the heatmap of the test image prior to applying a threshold.

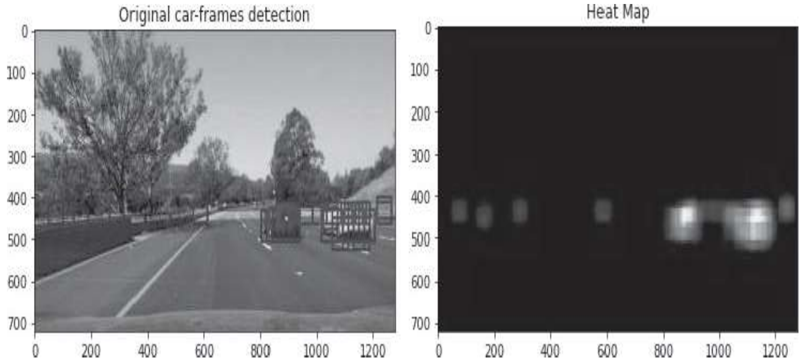


Fig 2. Heat map on the testing image (before threshold)

The boxes drawn around the automobiles are becoming increasingly precise. The heatmap is then applied to each box to increase the pixel countby one. Next, it's essential to apply a threshold value to the image in order to eliminate pixels with low values, identify the pixels related to every vehicle's number, and create the vital bounding boxes. Figure 3 illustrates the heatmap of the image for testing after this thresholding process.

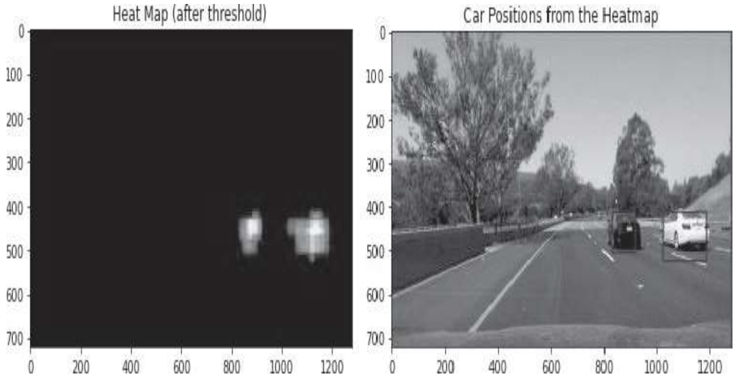
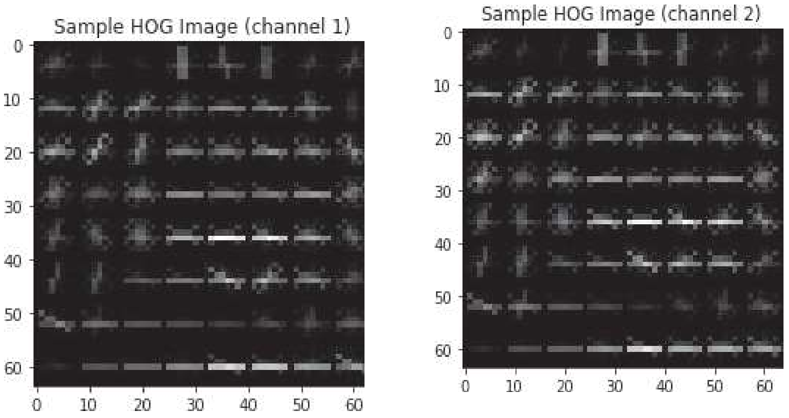


Fig. 3. Heatmap on the testing image(afterthreshold)

Histogram of Oriented Gradients (HOG)

A function descriptor is a prospective of offering a photo or a picture spot that streamlines it by drawing out one of the most important states and eliminating the remainder. The technique matters the variety of times a gradient positioning shows up in a certain area of a picture. The HOG work descriptor utilizes gradient path distributions (histograms) as functions. Since gradients have excellent amplitude near the sides and edges, they are testing to see (areas of sudden deepness modifications). Gradients (x and y by-products) in a photo are significant because of the truth they include some range additional information regarding the geometry of an item compared to level areas. The complying with action is to select the appropriate specifications for educating the classifier to anticipate the vehicles in the input picture. Figure 4 depicts the extraction of HOG features from a mock-up vehicles, utilizing both channel 1 and channel 2.



(a) HOG with channel 1 (b) HOG with channel 2

Fig. 4. HOG features extraction from one sample of the vehicles

Contours represent the outer edges of shapes and are instrumental in shape identification and recognition. To assess the effectiveness of contour detection, the Canny edge detection is applied to a binary image. The process of contour detection is facilitated by the "cv2.FindContours()" method within OpenCV. The algorithms employed in this procedure are elaborated upon below.

Convolutional Neural Network

YOLO V3 is a Convolutional Neural Network (CNN) that spots items in real-time. CNNs are classifier- based frameworks that can translate inbound pictures as ready arrays of truths and apprehend patterns. YOLO has the benefit of being considerably quicker compared to various networks while yet maintaining precision. YOLO and various convolutional neural community formulas "rack up" locations through assessing them to established classifications. Favorable detections of something course they many thoroughly end up being mindful of with are highlighted in high-scoring locations. For instance, YOLO can be utilized to differentiate unique ranges of cars in a websites visitors feed mainly based upon which components of the

video clip.

The algorithmic steps of CNN are given below:

- Prepare dataset for training
- The vehicle and non-vehicle dataset is splitted into train and test data with 80% and 20% proportion.
- The training dataset is shuffled using random_seed in train_test_split()
- Assigning labels and features i.e., the label namely vehicle and non-vehicle as 0 and Homogenizing X and converting labels of training data into categorical data using min_max_scaler()
- Split X and Y for use in CNN.
- Define, collect and train the CNN model using train and test data
- Accuracy and score of model is calculated

Support Vector Machines (SVM)

Support Vector Device (SVM) is a prominent Monitored Discovering method for Category and Regression issues. Nevertheless, it's extensively talking utilized in Device Learning how to fix category issues. The SVM technique is to discover the good line or choice limit for categorizing n-dimensional keen on training to ensure that succeeding info aspects can be other than problems placed in the beautiful classification. The remarkable option limit is deemed a hyperplane.

SVMs employ extreme points or vectors to define the hyperplane. These crucial instances are referred to as support vectors, and the technique itself is known as Support Vector Machine. The SVM algorithm finds applications in various tasks such as face recognition, image classification, and text categorization. SVMs come in two primary types such as Linear SVM and Nonlinear SVM. The Linear SVM is designed for datasets that exhibit linear separability, meaning the data can be divided into two classes using a single straight line. In such cases, it is referred to as linearly separable data. Non-linear SVM deals with datasets that cannot be effectively separated using a straight line. In these scenarios, the classifier used is known as Non-linear SVM. These SVM variations cater to different data characteristics and are applied accordingly to achieve optimal results in various problem domains.

In this work, Linear SVM is used to classify and detect the vehicle and non-vehicle using SVM hyperplane. The working of SVM algorithm using Linear SVM is explained below. The dataset with two labels namely vehicle and non-vehicle with features. SVM classifier can categorise the pair of coordinates (x1, x2) as vehicle or non-vehicle. In this two-dimensional space, the two classes are distinguished using a straight line. However, there may be several different lines capable of separating these classes. Consequently, the SVM technique rally around to identify the best line or verdict edge, which is termed a hyperplane. The SVM method calculates the point of intersection between the lines representing both classes. The margin denotes the distance between these support vectors and the hyperplane. The primary objective of SVM is to maximize this margin, aiming for the hyperplane that offers the widest separation. Ultimately, the ideal hyperplane is the one that achieves the utmost scope.

The algorithm steps are as follows:

- Data Pre-processing using min_max_scaler() and train_test_split() to split the dataset into train and test data.
- Τραινινγ τηε ΣςΜ χλασσιφιερ ον τηε τραινινγ δατα.
- Υσινγ τηε τραινεδ μοδελ το πρεδιχτ ουτχομεσ φορ τηε τεστ δατα.
- Generating the confusion matrix to evaluate performance.
- Visualizing the prediction result using bar chart and table

Vehicle Classification Module

The suggested system's third and final module is classification. Proper contours are obtained after using the foreground extraction module. The properties of these contours, including area, centroid, size, solidity and aspect ratio are extracted and employed for the categorization of the vehicles.

Vehicle Counting

Verify whether the centroid of a vehicle has reached or passed an imaginary line within the ROI in order to tally the vehicle tot up. The imaginary line occurs when two ROI locations are connected diagonally. The system counts the car after the centroid of a vehicle in ROI passes the made-up line. The variable "Total" and the variable of the relevant category increase when the centroid of the vehicle crosses the pretended string in ROI.

4 Results and Discussions

The major goal is to select the optimum model for vehicle detection. Once the dataset is trained, then the sliding window will be created. The system required to locate the area that the classifier would run. The function that returns the refined windows where, the classifier predict vehicles as the result. Then the procedure is executed that draws the main window around the detected automobiles. Fig 5 shows the recognized vehicles in the drawn window surrounding the vehicle.

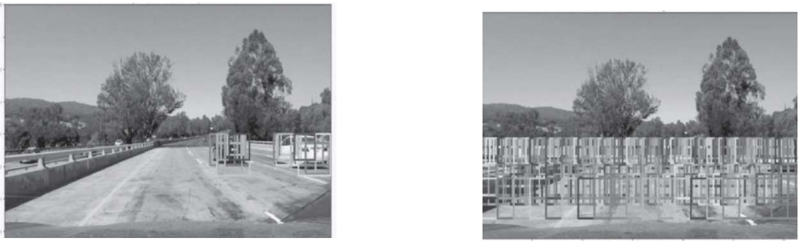
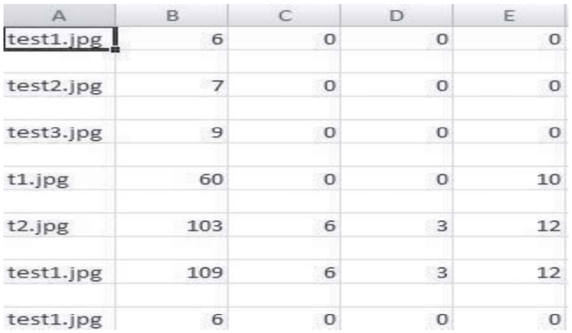


Fig 5: Refined Sliding Windows

The metrics include accuracy score, error, precision, recall, and f1 score are used as evaluation measures. Precision describes how many of the instances that are accurately predicted turned out to be positive. The traintestsplit() method from the Scikit-Learn package divides the data into two parts: training and testing. In general, the percentages for both are estimated to be around 80% for training and 20% for testing. The StandardScaler() method is used to standardize the data in the SVC classifier operation. After completing these steps, the SVM classifier has been trained, and the fallout of the methods are shown in the table 1

Table 1: Algorithm Comparison

Models	**Parameters**	
	Time(Seconds)	**Accuracy**
SVM	1.009	0.985
CNN	0.07	0.989



A	B	C	D	E
test1.jpg	6	0	0	0
test2.jpg	7	0	0	0
test3.jpg	9	0	0	0
t1.jpg	60	0	0	10
t2.jpg	103	6	3	12
test1.jpg	109	6	3	12
test1.jpg	6	0	0	0

Fig. 6. Counting of vehicles in images.

The bar charts in a comparable manner in Figure 7 shows outcome of both the models are displayed as in order to comprehend and compare them

Table 2. Performance Evaluation

Performance Measure/Algorithm	SVM	CNN
Precision(%)	98.02	99.11
Recall(%)	98.24	99.21
F1-Score(%)	99.08	99.18
Accuracy(%)	98.57	98.92
Error(%)	9.1	7.0

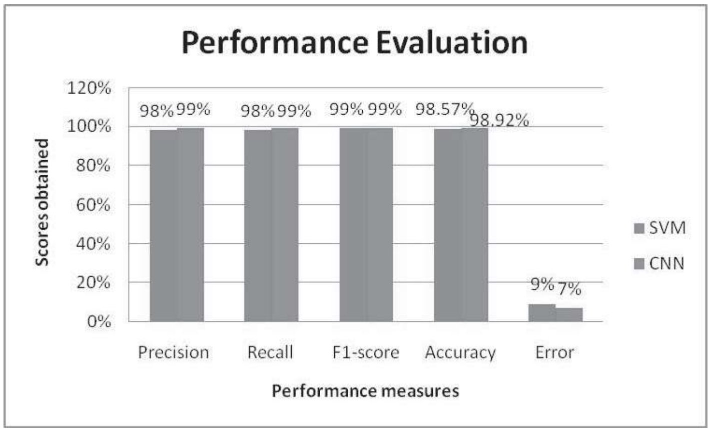


Fig. 7. Performance Evaluation

The figure 6 and Table 2 depicts that the F1-score is same for SVM and CNN and obtained 99%. The precision for SVM is 98% where CNN obtained 99%. The Recall for SVM is 98% and for CNN is 99%. Finally, the accuracy for SVM is 98.57% but CNN obtained 98.92%. from the figure, it shows that CNN performs better than SVM for detecting the vehicle in real time video and images.

5 Conclusion and Future Scope

The identification and monitoring of vehicles are integral components of object detection, a technology applied in various contexts like traffic management, urban environments, and other domains. This matter is growing in significance over time. The goal is to use existing image pre- processing techniques and tools to increase the performance of these algorithms and models. This research implemented two-vehicle detection and tracking classifier algorithms. Support Vector Machine (SVM) and CNN using YOLOv3 are the two models. These two techniques were the most often applied in the existing work. As a result, these models are selected and compared in order to determine which model was the better of the two. Numerous techniques have been employed to enhance precision and achieve optimal outcomes. Both models underwent training using identical datasets, with the findings demonstrating that CNN outperformed SVM.

There are activities and alternatives that can be included in the study or worked on separately. The most significant subject is vehicle detection, which is increasing every day. Although there are many current studies on vehicle recognition and tracking, implementing such a technique into practice in a real-world setting has been and still is complicated. Because machine learning and deep learning algorithms are always upgrading, it's a good idea to construct an updated deep learning model for vehicle recognition and tracking. Other deep learning algorithms for vehicle detection and tracking with improved performance can be developed in the future.

References

1. Andrew, W. M. and Victor, M. (2003), Handbook of International Banking(London: Edward Elgar Publishing Limited), 350-358 2)
2. Bambrick, N. (2018). Support vector machines: A simple explanation. línea]. Disponible en: https://www. kdnuggets. com/2016/07/support-vector-machines-simple-explanation. html.
3. Basak, D., Pal, S., & Patranabis, D. C. (2007). Support Vector Regression Neural Information Processing–Letters and Reviews.

4. Berni, J. A., Zarco-Tejada, P. J., Suárez, L., & Fereres, E. (2009). Thermal and narrowband multispectral remote sensing for vegetation monitoring from an unmanned aerial vehicle. IEEE Transactions on geoscience and Remote Sensing, 47(3), 722-738.
5. Chen, X., & Meng, Q. (2015, November). Robust vehicle tracking and detection from UAVs. In 2015 7th International Conference of Soft Computing and Pattern Recognition (SoCPaR) (pp. 241-246). IEEE.
6. Chen, X. (2016). Automatic vehicle detection and tracking in aerial video (Doctoral dissertation, Loughborough University).
7. Kalghatgi, M. P., Ramannavar, M., & Sidnal, N. S. (2015). A neural network approach to personality prediction based on the big-five model. International Journal of Innovative Research in Advanced Engineering (IJIRAE), 2(8), 56-63.
8. Kanistras, K., Martins, G., Rutherford, M. J., & Valavanis, K. P. (2013, May). A survey of unmanned aerial vehicles (UAVs) for traffic monitoring. In 2013 International Conference on Unmanned Aircraft Systems (ICUAS) (pp. 221-234). IEEE.
9. Khan, K., Baharudin, B. B., & Khan, A. (2009, June). Mining opinion from text documents: A survey. In 2009 3rd IEEE International Conference on Digital Ecosystems and Technologies (pp. 217-222). IEEE.
10. Koza, J. R., Bennett, F. H., Andre, D., & Keane, M. A. (1996). Automated design of both the topology and sizing of analog electrical circuits using genetic programming. In Artificial intelligence in design'96 (pp. 151-170). Springer, Dordrecht.
11. Noh, S., Shim, D., & Jeon, M. (2015). Adaptive sliding-window strategy for vehicle detection in highway environments. IEEE Transactions on Intelligent Transportation Systems, 17(2), 323-335.

12. Patel, P. J., Patel, N. J., & Patel, A. R. (2014). Factors affecting currency exchange rate, economical formulas and prediction models. International Journal of Application or Innovation in Engineering & Management, 3(3), 53-56.
13. Faqih, A., Lianto, A. P., & Kusumoputro, B. (2019, January). Mackey-Glass chaotic time series prediction using modified RBF neural networks. In Proceedings of the 2nd International Conference on Software Engineering and Information Management (pp. 7- 11).
14. Razakarivony, S., & Jurie, F. (2016). Vehicle detection in aerial imagery: A small target detection benchmark. Journal of Visual Communication and Image Representation, 34, 187-203.
15. Russell, S. J., &Norvig, P. (2016). Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited.
16. Sahli, S., Ouyang, Y., Sheng, Y., & Lavigne, D. A. (2010, April). Robust vehicle detection in low-resolution aerial imagery. In Airborne Intelligence, Surveillance, Reconnaissance (ISR) Systems and Applications VII (Vol. 7668, pp. 164-171). SPIE.
17. Schumaker, R. P., & Chen, H. (2009). Textual analysis of stock market prediction using breaking financial news: The AZFin text system. ACM Transactions on Information Systems (TOIS), 27(2), 1-19.
18. Susaki, J. (2015). Region-based automatic mapping of tsunami-damaged buildings using multi-temporal aerial images. Natural Hazards, 76(1), 397-420.
19. Tenti, P. (1996). Forecasting foreign exchange rates using recurrent neural networks. Applied Artificial Intelligence, 10(6), 567-582.
20. The Federal Reserve Board. (2004) “FRB:Speech, Bernanke -- International Monetary Reform and Capital Freedom-- October14, 2004”.

21. Tsai, Y. C., Chen, J. H., & Wang, J. J. (2018). Predict Forex Trend via Convolutional Neural Networks. arXiv preprintar Xiv:1801.03018.
22. Tseng, F. M., Tzeng, G. H., Yu, H. C., & Yuan, B. J. (2001). Fuzzy ARIMA model for forecasting the foreign exchange market. Fuzzy sets and systems, 118(1), 9-19.
23. Van Gerven, M., & Bohte, S. (2017). Artificial neural networks as models of neural information processing. Frontiers in Computational Neuroscience, 11, 114.
24. Viola, P., Jones, M. J., & Snow, D. (2005). Detecting pedestrians using patterns of motion and appearance. International Journal of Computer Vision, 63(2), 153-161.
25. Vyklyuk, Y., Vukovic, D., & Jovanovic, A. (2013). Forex prediction with neural network: USD/EUR currency pair. Актуальні проблеми економіки, (10), 261-273
26. Wang, X., Zhu, H., Zhang, D., Zhou, D., & Wang, X. (2014). Vision-based detection and tracking of a mobile ground target using a fixed-wing UAV. International Journal of -Advanced Robotic Systems, 11(9), 156.