International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-2, December 2019

Bengali License Plate Detection using Viola-Jones Algorithm

Ahmed Mortuza Saleque, Fahim Shahriar Chowdhury, Md. Rakin Ashraf Khan, Rashadul Kabir, AKM Bahalul Haque

***Abstract**: Image classification has been a rapidly developing field over the previous decade, and the utilization of Convolutional Neural Networks (CNNs) and other deep learning techniques is developing rapidly. However, CNNs are compute-intensive. Another algorithm which was broadly utilized and keeps on being utilized is the Viola-Jones algorithm. Viola-Jones adopts an accumulation strategy. This means Viola-Jones utilizes a wide range of classifiers, each looking at a different part of the image. Every individual classifier is more fragile than the last classifier since it is taking in fewer data. At the point when the outcomes from every classifier are joined, be that as it may, they produce a solid classifier. In this paper, we would like to develop a model that will be able to detect the Bengali license plates of, using the Viola-Jones Algorithm with better precision. It can be utilized for various purposes like roadside help, road safety, parking management, etc*

Keywords: **Viola-Jones Algorithm, License Plate Detector, Cascade Architecture**

I. INTRODUCTION

Deep learning has contributed a whole lot in the field of machine learning for many years now, particularly in computer vision. In the present time, Convolutional Neural Networks (CNN) are considered to be a very robust technique to detect object and faces in this field. CNN combines three architectural ideas to ensure some degree of shift and a steady distortion: local receptive fields shared weights and sometimes, spatial or temporal subsampling [1]. Over the years many development and work have been done to increase the efficiency and accuracy of CNN object detection.

Revised Manuscript Received on December 30, 2019.
***** Correspondence Author

Ahmed Mortuza Saleque*, Department of Electrical and Electronic Engineering, American International University-Bangladesh, Dhaka, Bangladesh. Email: ahmed.saikot@gmail.com

Fahim Shahriar Chowdhury, Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh. Email: fahim.shahriar@northsouth.edu

Md. Rakin Ashraf Khan, Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh. Email: rakin.ashraf@northsouth.edu

Rashadul Kabir, Department of Electrical and Computer Engineering, Presidency University, Dhaka, Bangladesh. Email: rashadk@pu.edu.bd

AKM Bahalul Haque, Department of Electrical and Computer Engineering, North South University, Dhaka, Bangladesh. Email: bahalul.haque@northsouth.edu

Region-based CNN and its evolutions Fast R-CNN [2] and Faster R-CNN [3] are still to date considered as the state-of-the-art technology in the field. Most recently You only look once [YOLO] is a very robust unified framework to detect objects in real-time [4].

But in the early days of object detection (especially, facial recognition) The Viola-Jones Object Detection framework was the first to give competitive rates back in the 2000s [5]. It has traditionally solved the problem of detecting faces rather quickly in real-time. Many point & shoot cameras and mobile devices adapted the framework to detect faces in real-time. Many other devices also use some kind of Viola-Jones framework to this day. This framework is quite simple to understand and due to its high detection rate (true-positive rate) and fast processing, the result tends to be more accurate. The framework works on different weak classifiers and combines them to create a stronger classifier to finally detect objects based on Haar-like features [5].

CNN based object detectors are accurate but they are quite complex and resource-heavy in nature. They require a very large set of training data and time. Fast R-CNN, Faster R-CNN, and YOLO all are usually GPU and sometimes CPU heavy computational frameworks. Mostly they use high-end GPU (e.g. GeForce GTX Titan and newer models) and server-grade CPU's [2, 3, 4, 7]. On the other hand, the Viola-Jones framework originally was working on a consumer-grade CPU compared to the latest CNN methods. The framework performed face detection on a conventional Intel Pentium III CPU, which clocks at 700Mhz [6]. This gives the Viola-Jones framework a great advantage; cost-effective and a feasible solution for real-time object detections in handheld devices.

Having less computational power and space compared to a full-fledged machine it is easier and efficient to use a cascaded framework. AdaBoost [8] was used to boost the original framework to construct simple and efficient classifiers from the computationally efficient features for final feature selection. Only a small number of features was needed to be evaluated during the run time, which makes it process a quite large dataset faster. This cascade is also quite simple and the structure is homogenous, which dramatically reduces computation time with improved detection accuracy.

The work of C. P. Papa Georgiou and others [9] had quite a great influence on developing the Viola-Jones framework. In the paper, a framework was presented for object detection in disordered scenes, based on the use of an overcomplete dictionary of basis functions combined with statistical learning techniques.

Ultimately it gives a compact representation of an object class which is used as an input. This system was also reliant on learning from examples and it didn't require handcrafted models. The system was able to detect objects in various regions in the image and on different scales.

Traditionally, the Viola-Jones framework was an effective solution for detecting faces. While CNN's are good at detecting objects but it's impractical to use it for face detection [7]. Viola-Jones framework was also developed to effectively work on other domains like automobile and pedestrian detections [6]. In this paper, we use this algorithm to detect Bengali License Plates.

II. RESEARCH METHODOLOGY

A. Data Acquisition

The first step is the image acquisition stage. We captured nearly two thousand images of the license plate of different types of vehicles all over Dhaka city. The images of the vehicles were captured using the cellphone camera. The locations were: parking lot of universities and shopping malls, outdoor parking, running vehicles from central roads of Dhaka city. We've got a better view while standing on the foot over bridge and then took the pictures of various vehicles. Pictures were taken from the front and the back of the vehicles, at different angles, at different times of the day and night time. This process spans over two months period. Figures 1,2 & 3 show some of the original images that were taken.

Fig. 1. Image of a Bus from a foot-over bridge (Front side)

A major constraint that we had to keep in mind, is the image of the vehicle should be captured in such a way that the selected input image contains a rear or front view of the vehicle with the number plate. Our system captures two images of observed vehicles:

1. a context image of the vehicle and its immediate surroundings, and

2. an image of the license plate. The images we captured are divided into two parts. They are:

- Positive Images
- Negative Images

Positive images are those images where the license plate can be seen clearly and a human being can detect and read the plate with ease. On the other hand, negative images are the ones which came out as blurry, license plates that can't be seen clearly or angled in an odd position and thus it can't be detected. Also, the images that do not contain the Bengali license plate can also serve the purpose of negative images.

Images were usually taken from the rear side of a vehicle, yet an image of the front of a vehicle might be taken when the vehicle is upheld into a parking lot, left in a driving aisle, or parked in any other way which provides better visibility of the vehicle's front. Pictures taken are not of a resolution that allows for distinguishing proof of vehicle inhabitants, assuming any.

Fig. 2. Image of a vehicle from the road (Read side)

Fig. 3. Image of a vehicle from a garage parking (Front Side)

B. Data Preprocessing

Images are usually captured in an RGB (Red, Green, and Blue) color model. The captured image is influenced by numerous components like optical framework, distortion, system noise, lack of exposure or excessive relative motion of camera or vehicle thus resulting in a degradation of a captured vehicle image hence adversely influencing the results of the overall image processing. As a correction mechanism, an image pre-processing stage is introduced to take care of any errors that may have occurred during the image acquisition stage. Firstly, we changed the aspect ratio of an image which describes the proportional relationship between its width and its height. So, we cropped the images to 4:3 format so that it was better to detect vehicle license plates. We made sure the photos which were clean, license plates were clearly visible were slotted into Positive Images Folder. Whereas, the photos which were blurry and the license plate of vehicles were not clearly visible were kept in Negative Images Folder.

After sorting out the desired positive images, they were manually labeled. We labeled the license plates which are clearly visible with a fixed ration bounding box. Some examples of the positive, negative and labeled images are shown in figures 4, 5 & 6.

Fig. 4. Some of the Positive Images

Fig. 5. Some of the Negative Images

Fig. 6. Labeled Images

III. ALGORITHM

Our detector is solely base on the Viola-Jones algorithm. Usually, the viola-jones algorithm is used in the detection of faces, but its cascade structure and feature extraction are also quite useful for detection other things, e.g. license plates, numbers. The algorithm is structured on three bases. Integral Image, AdaBoost and the Cascade Structure.

Integral image is an image representation that allows a very fast feature evaluation. The images are classified base on the value of simple features. This set of features will be used and this feature is similar to Haar Basis function, in some cases, related filters are more complex than the original function itself. Three kinds of features were used: Two rectangle features, Three rectangle features, and Four rectangle features. The integral image can be computed from an image using a few operations per pixels. And then the features can be computed in constant time. AdaBoost is very important for the boosting of the model. It enables the model to classify faster. In the Viola-Jones algorithm, the original AdaBoost procedure is simply modified, so the weak learner can be constrained and that each weak classifier that was returned was only dependent on a single feature. That's why each stage of the boosting process, a new weak classifier. The cascade which is actually a degenerate decision tree. A positive outcome from the primary classifier triggers the assessment of a second classifier which has been also changed to accomplish high detection rates. A positive result from the second classifier then triggers the third classifier. A negative result at any point leads to the immediate rejection of the particular sub-window. Subsequent classifiers are trained using the previous example which passes all the previous stages. As a result, the second classifier faces a more difficult task than the first. These previous steps usually go into the training section of our algorithm. After images are labeled the training process begins and at the very fast step all the haar like features are extracted from all the positive images. Weak classifiers are defined by the algorithm itself and then it combines the weak classifier to produce a stronger classifier.

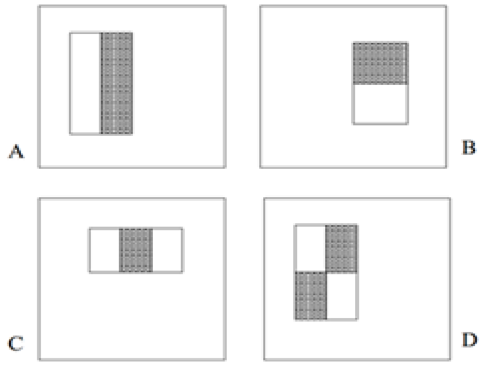

Fig. 7.

Example rectangle features shown relative to the enclosing detection window. The sum of the pixels which lie within the white rectangles is subtracted from the sum of pixels in the gray rectangles. Two-rectangle features are shown in (A) and (B). Figure (C) shows a three-rectangle feature, and (D) a four-rectangle feature

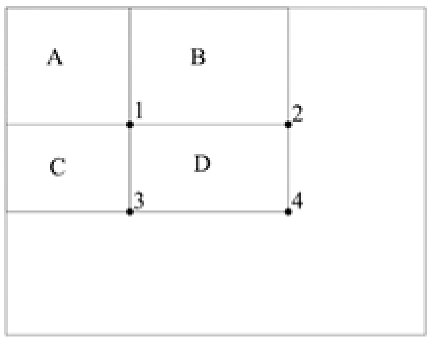

Fig. 8.

The sum of the pixels within the rectangle are being computed with references from four arrays. The value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is A+B, at location 3 is A+C, and at location 4 is A+B+C+D. The sum within D can be computed as 4+1-(2+3) Finally, after the training is done all the combined information and the variables of the training data are outputted on an XML file which will be used as our detector in the detection stage. In the detection stage, the samples should be specified. The samples then will be segmented by a function and frame by frame detection will begin. As mentioned, the rectangular features before the whole picture will be scanned.

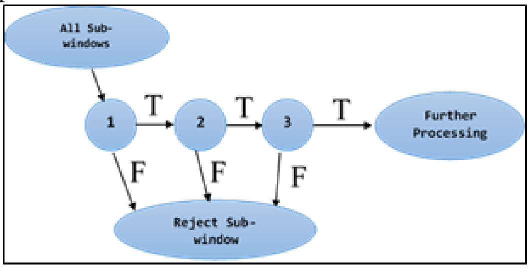

Fig. 9. A visual representation of the detection cascade

When the license plate is detected in a frame or multiple frames it will be annotated with a bounding box labeled as License Plate and will show it on a video player when the sample video or image is running. And finally, the plates that are detected and is residing in the bounding box will be extracted and saved into the desired file format. It is important to note that the license plates are all the same size and maintains a width to height ratio. Our bounding box will be constrained so anything smaller than the actual license plate or bigger than the plate will be excluded from the final detection. Figure 10 and 11 shows our flowchart of the training and detection model.

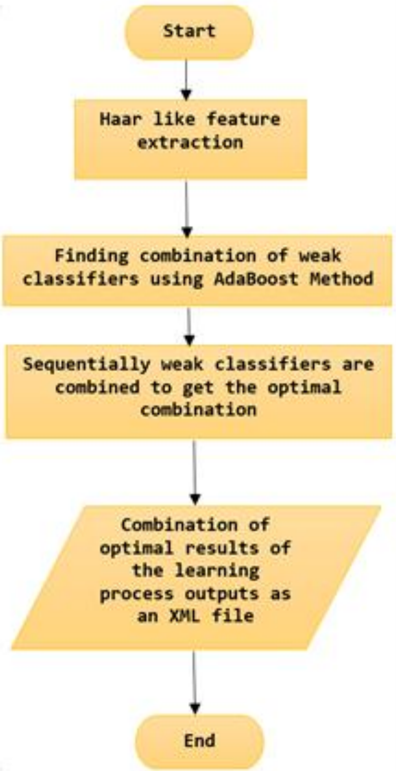

Fig. 10. Training stage

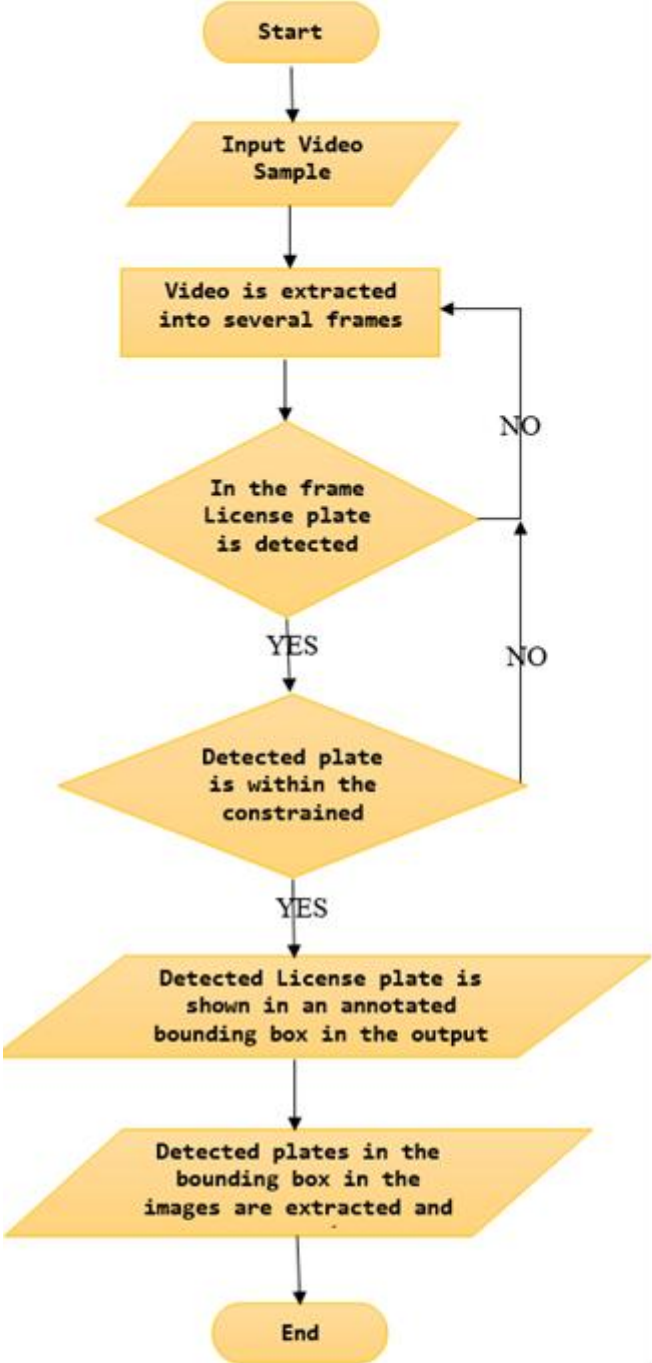

Fig. 11. Detection stage

IV. SIMULATION

We ran our License Plate Detection simulation in MATLAB R2018b on an Intel® Core™ i5-8265U, 1.6GHz quad-core processor with Turbo Boost (up to 3.9GHz) and 6MB cache with 8GB 2400MHz DDR4 RAM. At the beginning of the simulation, we kept all the necessary folders and data in a single drive of the machines solid-state drive (SSD). We made sure that the machine was not running any other programs in the background and the RAM had no extra usage in other programs. We ran the simulation in two parts. First, the training and then the detection itself.

A. Training

We start by loading the .mat session file containing the positive images. All the positive images are labeled manually in this file. Then a two-column table is created from this file. The first column contains the specific image file name and the second column contains M-by-4 matrices of [x y width height] bounding boxes specifying object locations. The table row is equal to the number of positive images used. Then the positive image and negative image folder are set.

After this, we use the trainCascadeObjectDetector (outputXMLFileName, postivieInstances, negativeImages) function to train our model. Depending on the number of positive and negative image the total time of the training session varies. Additional parameters, such as; factor of negative samples, number of cascade stages, false alarm rate, true positive rate, and feature type can be passed through the function parameter. After the training session, the newly trained classifier will be created on an XML file. Using this file we will proceed to the detection stage of the simulation.

B. Detection

At first, we Specify the folder containing the sample videos or images. Using the detectLL() function the license plates will be detected and returned in a GUI video player. In this video player, the detection will be shown in a rectangular bounding box. Successfully detected license plate number will be stored in a text document or any other format according to the use.

V. RESULT AND DISCUSSION

A. Description of the video samples

All the video samples and image samples are collected from Dhaka City area. Video sample 01 and 02 consists of a single car facing towards the camera. Video Sample 03 features a busy road with pedestrians and multiple vehicles. In this video, there are 4 possible license plates visible and others are mostly blurry. Video sample 04 features the same kind of scenario but with more vehicles and different lighting condition. There are also 4 possible detections in the video. Video sample with multiple cars has three parts. The first part consists of 5 front viewing vehicles, 2nd part consists of negative images that are not to be detected by the detection algorithm and in the last part 6 pictures of license plates that are taken in an angle. Lastly, in the image sample, there are 12 cars in different orientations.

B. *Result of the Simulation Sessions*

Table I Simulation for 110 Positive Samples

Positive Sample Used		**Number of Negative Images**		
110		1000	4000	8000
Target Cascade Stages		10	20	15
Completed Cascade Stages		9	11	9
Training time		6 min	85 min	10 min
Video Sample 01		Somewhat Detects	Detects	Somewhat Detects
Video Sample 02		Detects	Detects	Detects
Video Sample 03		No Detection	No Detection	No Detection
Video Sample 04		No Detection	No Detection	No Detection
Video Sample Multiple Cars	Frontal	1 out of 5	2 out of 5	1 out of 5
	Negative Detection	1	0	1
	Angled	0	0	0
Image Sample of Multiple Cars		6 out of 12, 1 negative detection	6 out of 12	6 out of 12
Overall Detection Rate		30%	36%	30%

Table II Simulation for 188 positive Image Samples

Positive Sample Used		**Number of Negative Images**		
188		1000	4000	8000
Target Cascade Stages		20	20	15
Completed Cascade Stages		7	8	12
Training time		11 min	14 min	15 min
Video Sample 01		Detects	Detects	Detects
Video Sample 02		Detects, with some negative detection	Detects	Detects, with some negative detection
Video Sample 03		Multiple negative detections	2 positive and multiple negative detections	Multiple negative detections
Video Sample 04		Multiple video detections	2 positive and multiple negative detections	Multiple negative detections
Video Sample Multiple Cars	Frontal	5 out of 5	5 out of 5	4 out of 5
	Negative Detection	1	0	0
	Angled	2 out of 6	2 out of 6	2 out of 6
Image Sample of Multiple Cars		6 out of 12	10 out of 12	10 out of 12 and 2 negative detections
Overall Detection Rate		54%	75%	64%

Table III Simulation for 325Positive Image Samples

Positive Sample Used	**Number of Negative Images**		
325	1000	4000	8000
Target Cascade Stages	15	20	20

Retrieval Number: B7689129219/2019©BEIESP
DOI: 10.35940/ijitee.B7689.129219
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Completed Cascade Stages		10	8	12
Training time		65 min	61 min	63 min
Video Sample 01		Detects and some negative detections	Detects	Detects
Video Sample 02		Detects and some negative detections	Detects and some negative detections	Detects and some negative detections
Video Sample 03		Detects license plates with some negative detections	Multiple Detections and some negative detections	Multiple Detections and some negative detections
Video Sample 04		Detects license plates with some negative detections	2 detections and some negative detections	3 positive detections with some negative detections
Video Sample Multiple Cars	Frontal	5 out of 5	5 out of 5	5 out of 5
	Negative Detection	0	1	0
	Angled	4 out of 6 and 2 negative detections	4 out of 6	4 out of 6
Image Sample of Multiple Cars		12 out of 12	12 out of 12	12 out of 12 with 5 negative detections
Overall Detection Rate		89%	90%	93%

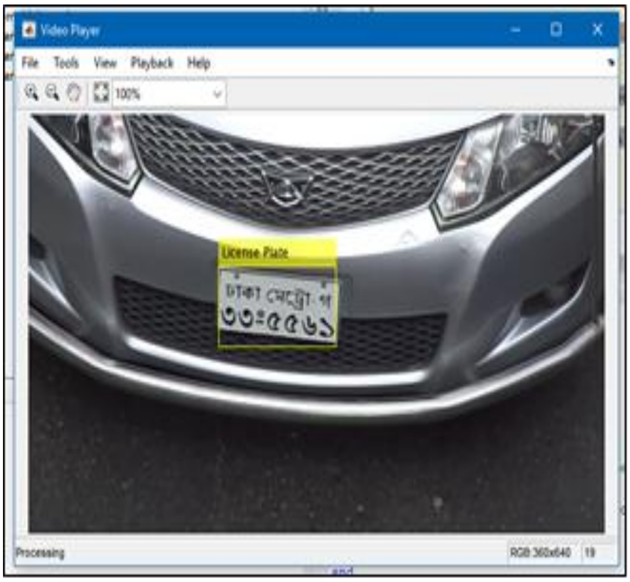

Figure 12. Detection in the video sample

Figure 13. Detection in video sample 02

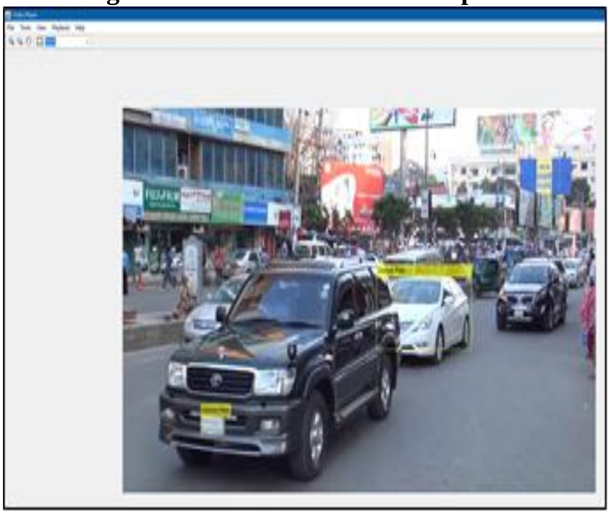

Figure 14. Detection in video sample 03

C. Discussion

Analyzing the results of our simulation it is clear that the number of positive images is a big factor in the positive detection of a license plate. When the number of positive images was small, it performed well in detecting single-vehicle scenarios. But as soon as the detector tries to detect multiple cars the detection rate decreases and most of the time in multiple car scenario it fails to detect anything at all. It also failed to detect any license plates that are positioned in an angle. As we increase the number of positive images the detection rates improve. On the other hand, it is ideal to have at least the double number of negative images than the positive image. We can see with more positive images and negative images the training can achieve to complete more stages and the overall detection rates are also higher. Initially, we started with 1000 negative samples and simulated with up to 8000 negative samples. One interesting thing we noticed is that with fewer positive samples and a larger number of negative images the training takes very small time. It happens because of the smaller number of positive samples used. The detector in the training session creates negative samples based on the provided negative images.

But it is solely dependent on how much positive images are being considered in that particular stage. If the number is initially small there will be fewer negative samples generated and the training will be done before even accessing all the negative images provided. As the high number of negative samples don't contribute to any kind of direct advantage it is efficient and time effective to use less amount of negative sample depend on the amount of the positive image that will be used. Considering a safe ratio of the positive sample and the negative sample is important for improving the accuracy of the model. Another thing to mention is, all the positive and negative images we used were in RGB. But we experimented with the images. We converted all the images to grayscale and then ran the same tastes with the same number of samples. The overall detection rate wasn't impressive than the presented one, so we decided not to convert the images into greyscale in the first place. Though more experiments can be done to see how the model performs on greyscale video footage and images.

VI. CONCLUSION

We have developed a model based on the original Viola-Jones algorithm to solve a problem that is not usually solved by this particular algorithm. We presented the usefulness of the Viola-Jones algorithm. It is effective and accurate in the field of object detection alongside human face recognition. After the training phase, the model detected the license plate of a single-vehicle with 100% accuracy. It also successfully detects multiple license plates on different vehicles with an overall accuracy of 93%. The performance of the model can be improved by providing more positive images to achieve higher accuracy on license plate detection.

REFERENCES

1. Y. LeCun and Y.Bengio. Convolutional networks for images, speech and time series. The handbook of brain theory and neural networks, 1995.
2. R. B. Girshick. Fast R-CNN, 2015.
3. S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks, 2016.
4. J. Redmon, S. Divvala, R. Girshick, A. Farhadi. You Only Look Once: Unified, Real-Time Object Detection.
5. P. Viola, M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features.
6. P. Viola, M. Jones. Robust Real-Time Face Detection, 2003.
7. H. Li, Z. Lin, X. Shen, J. Brandt, G. Hua. A Convolutional Neural Network Cascade for Face Detection.
8. Y. Freund, R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting, 1996.
9. C. P. Papageorgiou, M. Oren, T. Poggio. A General Framework for Object Detection, 1998.

AUTHORS PROFILE

Ahmed Mortuza Saleque received Bachelor Degree in Electrical and Electronic Engineering from American International University, Bangladesh in 2010. In the same year, he joined the Department of Electrical and Electronic Engineering, American International University Bangladesh (AIUB) as Lecturer. He received Master of Engineering in Telecommunications from American International University Bangladesh in 2012. Both in his Bachelor and Master's, he received the prestigious Summa Cum Laude distinction. From 2013 to 2015, he was awarded Erasmus Mundus scholarship for Master's program. He received joint Master degree in Sustainable Transportation and Electrical Power Systems (EMMC-STEPS) from University of Wisconsin Madison, USA, University of Nottingham, UK, University of Oviedo, Spain and University of Rome, Italy. He was Visiting Scholar at the Wisconsin Electric Machines and Power Electronics Consortium (WEMPEC), University of Wisconsin Madison, USA, in 2015. He is currently working as an Assistant Professor, Department of EEE, American International University Bangladesh (AIUB).

Fahim Shahriar Chowdhury is doing his Bachelors in Computer Science and Engineering in the Department of Electrical and Computer Engineering, North South University, Bangladesh. He is set to graduate at the end of Fall 2019. He has research interest in Machine Learning Techniques and Artificail Intelligence and enthusiastic about going abroad to pursue higher education and to build a career in the Data Science field. He has also served as the President of North South University Diversity Club, which worked to achieve true and proper "Diversity" in and outside the campus.

Md. Rakin Ashraf Khan is an undergrad student majoring in Computer Science and Engineering in the Department of Electrical and Computer Engineering, North South University, Bangladesh. He'll be completing his degree in 2019. His field of interest includes Machine Learning, Artificial Intelligence, and Business Analytics. Currently he is working on his final undergrad project "Frame-Level Features for Large Scale Video Classification".

Rashadul Kabir received his Bachelor Degree in Electrical and Electronic Engineering from American International University, Bangladesh in 2011. In 2014 he completed his MS in Electrical Engineering from State University of New York at New Paltz with a research focus in Computer Architecture. After coming to Bangladesh he joined Presidency University in 2015, where he received the best faculty award from the ECE Department in 2016. In 2017, he joined North South University as a part time faculty. His research interests include Computer Architecture and Supervised Learning.

AKM Bahalul Haque is currently working as a faculty member of Department of Electrical and Computer Engineering, North South University. He completed his bachelor in Computer Science from Bangladesh and his Master of Science in Information Technology from Germany. His area of research is Cyber Security and Software Engineering. He has research papers published in International Journal and Prestigious conferences.

Retrieval Number: B7689129219/2019©BEIESP
DOI: 10.35940/ijitee.B7689.129219
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication