Object detection and classification by cascade object training

Conference Paper · September 2020

DOI: 10.1109/IEMTRONICS51293.2020.9216377

5 authors, including:

Ahmed Masud Chowdhury
Indiana University-Purdue University Indianapolis

Jafrina Jabin
North South University

Erteza Tawsif Efaz
FAB Lab Dhaka

Md. Ehtesham Adnan
North South University

Object detection and classification by cascade object training

Ahmed Masud Chowdhury
Department of Electrical & Computer Engineering
Indiana University – Purdue University Indianapolis
Indiana 46202, United States
amasudch@purdue.edu

Jafrina Jabin
Department of Electrical & Computer Engineering
North South University
Dhaka 1229, Bangladesh
jafrina.jabin@northsouth.edu

Erteza Tawsif Efaz*
Department of Electrical & Electronic Engineering
Ahsanullah University of Science & Technology
Dhaka 1208, Bangladesh
ertezatawsif@gmail.com

Md Ehtesham Adnan
Department of Electrical & Computer Engineering
North South University
Dhaka 1229, Bangladesh
ehtesham.adnan@northsouth.edu

Ashfia Binte Habib
Department of Electrical & Computer Engineering
North South University
Dhaka 1229, Bangladesh
ashfia.habib@northsouth.edu

***Abstract*— Computer Vision (CV) has become ubiquitous in smart systems for detecting and labeling objects, starting from social media platforms to autonomous vehicles. It requires extensive computation and image processing. In this paper, a model is processed and used to detect various colored cups with saucers from a set of different objects. The system is trained using Cascade Trainer Graphical User Interface (GUI), and the testing is done utilizing MATLAB, discussed in detail. Finally, the model is tested for its efficacy on the S32V234 Evaluation Board (EVB). Our proposed system accomplished its goal by identifying and tagging the objects of interest with maximum possible accuracy.**

Keywords*— *MATLAB, S32V234, object detection, computer vision, HAAR Cascade Classifier

I. INTRODUCTION

Object detection and classification is a core part of Computer Vision (CV). In autonomous systems, image detection is one of the features that can make autonomous vehicles smart. Generally, a system that can identify any object through CV can be considered as a smart system [1-3].

Detection methodologies have progressed on a good note by the exceptional work of Viola & Jones by introducing a speedy object identification theme supporting a boosted cascade of straightforward feature classifiers [4, 5]. Due to the simplicity and effectiveness of HARR-type selections, this system can process pictures extraordinarily quickly with the achievement of high detection rates, which may fulfill the minimum requirements of many feasible applications such as, human face recognition in regards to Human-machine Interaction (HMI). However, abundant work is required before automatic object identification technologies can perform like humans [6].

Viola-Jones' technique of object detection is one amongst the foremost utilized in the resolution of identifying tasks. But this detection pace is achieved including a few incorrect allowances of unspecified objects, although the refusal error for specified objects is considerably low [7, 8]. The rationale is that in essence, Viola-Jones' refusal cascade of classifiers is integrated by AdaBoost to make a robust classifier with every node being a collection of weak classifiers employing HAAR-type selections. Here, the cascaded classifier is a supervised classifier, in which sub-windows of multi-resolutions of an image are tested consecutively against all nodes within the cascade. Whereas, a window passing all nodes is considered as a possibility for any targeted object [9].

In Viola-Jones cascade, every node is meant to possess a high (≈99.9%) detection percentage (fewer false-negatives) at the value of a low frequency of true-positives (≈50%). It suggests a higher degree of false-positives. At any node, a call of refusal stops identifying, while the detection method carries on if any short-lived recognition is found [7, 8]. So, the time for computation is saved hugely due to the extraordinarily lowered specified object sub-windows than unspecified object sub-windows of a picture. Therefore, stress ought to be arranged on sequentially weak classifiers until they maintain a sufficiently low percentage of false-negatives (incorrect refusal for objects of interest), assuring that just about all sub-windows consisting of targeted objects pass all nodes of cascaded classifiers [9].

Our goal of this research is to build a model that can detect and label objects utilizing cascade training and detection. The training is to complete using Cascade Trainer Graphical User Interface (GUI), and the testing by MATLAB. The model is translated then for implementation on the S32K234 Evaluation Board (EVB), which is finally tested for the proof of concept.

The rest of this manuscript is ordered as stated; unit II provides some connected work as well as the premise for HAAR-type options. The associate object detection algorithmic program supported the primitive cascaded classifier is projected very well in unit III. The outcomes are mentioned in unit IV, and at last in unit V, conclusions are extracted.

II. BACKGROUND

There are many kinds of algorithms deployed to identify objects or human faces with expressions as well [1, 6, 9]. All of the methods have their benefits and drawbacks [10-19]. In 2002, T. Ojala & others used Local Binary Pattern (LBP) to classify the textures of images [10]. The LBP is a grayscale insignificant texture driver with effectual distinction, additionally has been employed for the recognition of human faces. Although LBP options have excellent distinctive efficacy, they dropped the native composition underneath particular conditions.

Some strategies concerned the templates, filter, and neural network [11-13], such as the analysis that was administered by Thai Hoang Le [11]. The downside of this applied rule is that the system is too costly. Moreover, this process needs re-analyzation of the pixels of pictures for the measure and efficiency that rely upon the worth of color and strength of light, requiring an extended time, which is usually exhausting to achieve. Since 2012, several investigations occurred victimizing identical practices [14-19], one amongst them is the research by W. Sulistiyo & others that created a technique for reporting participation [16]. And, the hardware is simply a model used for attendance, except as the guard of the entrance. Furthermore, human face recognition with hijab and spectacles has not been accounted for therein analysis.

Indicators supporting the HAAR-like methodology have a considerably low rate for failing detections. The procedure operates properly and identifies objects, revolving at an angle under 30° [4]. Once the angle crosses 30°, the value of strong detections slumps. So, utilizing the quality function of this approach, it is unlikely to identify the object of interest, revolved at any angle. This specificity offers the appliance of this method sophisticated for contemporary industrial technologies [5]. In this experimentation, we presented an object detection and labeling formula in regards to a smart kitchen.

III. MODEL IMPLEMENTATION

The prototype was executed by integrating the selected software-hardware containing preprocessing and training of images of the required objects. Utilizing a microcontroller is incredibly useful because it can be programmed effortlessly. Individuals with the minimum programming experience will be able to code it comfortably because it includes all the necessary options [20-22].

A. *Software & Hardware*

1) *Cascade Trainer*

Cascade Trainer GUI is an algorithm accustomed to train, test, and develop cascade classifier designs. It generally utilizes a graphical interface to fix the values and formulate it to be simple to apply OpenCV mechanisms to train and test classifiers. The cascade classifier is a branch-formatted method, within which Viola-Jones used HAAR-type selections for identification.

The usual HAAR-type selections established by standards, as illustrated in Fig. 1, maybe deployed considering every measure within the boosted classifier for quick computation from the associated integral version of the targeted picture for detection. HAAR-type selections are deep-seated by the HAAR wavelet. A HAAR wavelet is a unique quadrilateral formation (one with large interim and one with small interim) [23].

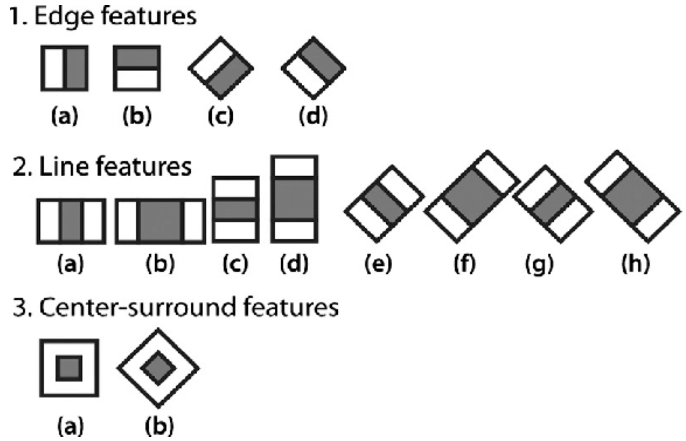

Fig. 1. HAAR-type selections wavelet illustration from OpenCV Cascade Classification (light section indicates 'add that region' and dark section indicates 'subtract that region') [23].

In 2 Dimension (D), the rectangle is represented by white and black. The significance of HAAR-type selections endures if the outcome is more than the outset, by decreasing the pixel's mean rate in black and white squares. These traits might be discovered in any space among a scanning picture window.

2) *MathWorks MATLAB*

The object identifying algorithms typically employ CV techniques to find and classify objects from photographs. By utilizing MATLAB (R2018b), the most suitable object identification procedures with minimal lines of code can be explored. Besides, it will be able to stimulate the method of training by the usage of GPUs [24].

Once the Cascade Trainer generates the .xml file after completing the training, MATLAB reads the file to detect cups with saucers. The vision.CascadeObjectDetector() API is used to create the detector object, which is further fed into the step() method to run the detector object to detect the object of interest. After the detection, the detected region is outlined with a rectangle and captioned as 'cup' to be displayed.

3) *S32V234 EVB*

The S32V234-EVB2 is a development board with maximum execution, enclosed perception, and detector fusion appliances. Supported by 64-bit Arm Cortex-A53 S32V processors, this EVB allows a pair of board choices (including/excluding integral monitor); satisfies all options, together with Automotive LAN, VIU camera, FlexRay, etc. Available are expansion card choices to assist the camera detectors also, as illustrated in Fig. 2, whereas, by deploying the cost-competitive with small form factor EVB, a vision-based case can be solved by its advantages [25].

Fig. 2. S32V234-EVB2 Evaluation System from NXP Semiconductors [25].

B. *Dataset & Preprocess*

A model was trained to detect and label cups, specifically – cups with saucers with different colors from other objects, as illustrated in Fig. 3. For this purpose, a set of 50 positive images were taken to train the model, collected from Google Images. Another set of images was used in the 'cup' category from Caltech101 as well. The google images yielded higher efficiency in detecting and labeling the cups. Thus, they were utilized for detecting and labeling the object of interest – cups with saucer in this case. Besides, for training purposes, 100 negative images were employed, whereas for preprocessing, all the collected low-noise images were resized to 300×300 pixels.

Fig. 3. Cup with saucer.

C. *Setting & Training*

For training purposes, the Cascade Trainer GUI (version 3.3.1) was used, as illustrated in Fig. 4.

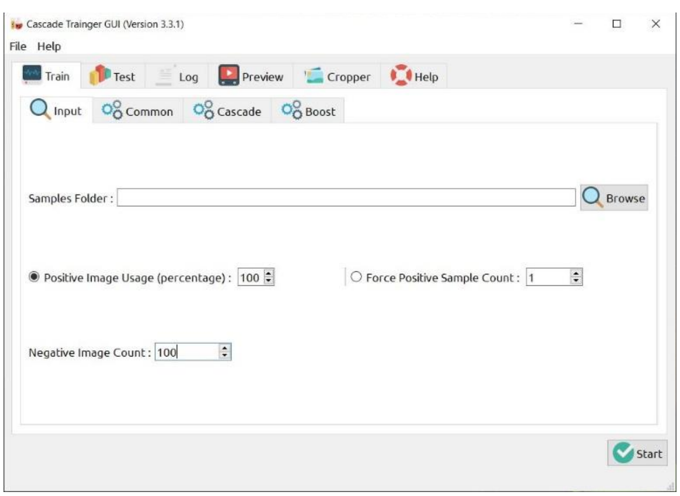

Fig. 4. The Cascade Trainer GUI interface.

The directory to the positive images was placed on the 'Sample Folder:' field of the interface. After this directory was set, the 'Negative Image Count:' field was placed to 100, as 100 negative images were taken. Once all the input specifications were set, the next step was to place the number of training stages, under the 'Common' tab, as illustrated in Fig. 5.

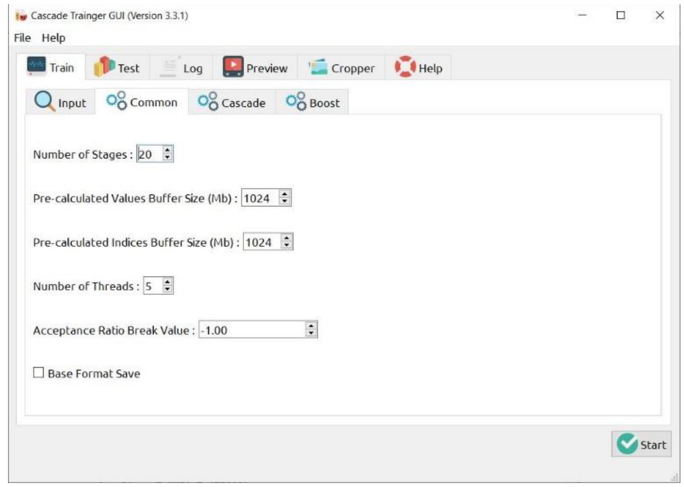

Fig. 5. Common tab specifications.

The 'Number of Stages:' was set to 20 – the higher is the number, the better is the detection accuracy. However, it takes an increasingly long time with every increment. Here, the next step was to place the parameters under the 'Cascade' tab, as illustrated in Fig. 6.

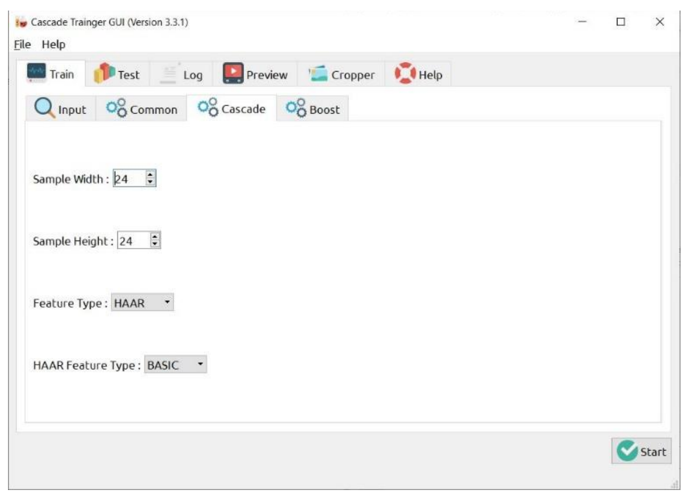

Fig. 6. Cascade training parameters.

The 'Sample Width:' and 'Sample Height:' represents the width and height of the training samples in pixels, respectively. For this model, the 'Sample Width:' and 'Sample Height:' were set to 30. The 'Feature Type:' was placed to 'HAAR,' and the 'HAAR Feature Type:' was set to 'BASIC'.

Once the training parameters were placed, the 'Start' button was deployed to start the training, which usually leads the software to use 'opencv_createsamples' internally to create positive samples. Then it employs an 'opencv_traincascade' application for training the model. When the training completes, it gives out a .xml Cascade Classifier file for detecting cups in images through MATLAB.

IV. Experimental Result

The trained model was tested in both MATLAB R2018b and S32V234 EVB, where the processor, RAM, and OS of the deployed computer were Intel Core i5 8th Generation, 8GB, and Windows 10 Home, respectively. Both the systems successfully detected the cups with varying accuracy. Fig. 7 shows the MATLAB output of the tested images. It can be seen that the model detected all the cups when running on MATLAB with perfect accuracy. However, Fig. 8 shows a reduction in the accuracy of detection in terms of S32V234 yield.

Fig. 7. The MATLAB output.

Fig. 8. The S32V234 output.

To test the developed system, random pictures were used, such as a tree, human brain, and cups with saucers with separate colors. The detection approach was justified for a wide range of possibilities by the usage of a set of mixed objects. Fig. 9 shows an instance of a false-positive output generated when the model was run on the EVB.

Fig. 9. S32V234 false-positive outcome.

When the Caltech101 images were used for training the model, the detection did not yield a satisfactory result. Upon closer inspection, it was identified that the Caltech101 images were noisy, and the edges, as well as the corners, were not distinct enough to be detected by the Cascade Trainer GUI. After training, the model was not able to detect any of the positive images employed for training; thereby, the Google Images results were utilized, which produced better outcomes in the detection of the object of interest.

In the case of testing, the model showed different results for MATLAB and S32V234. The discrepancies in the results could be caused by improper preprocessing of the images. There was an instant when all the cups were detected by the board, but the detection stopped before the screenshot was taken. This might be indicating the lack of capability to operate in real-time for the trained model.

V. Conclusion

The research was done to train a model to detect some objects of interest among diversified objects, which in this case, were cups with saucers with differing colors. Our developed algorithm in terms of CV successfully detected and labeled the test images with varying degrees of accuracy – neutralizing the deficiencies of the primordial Viola-Jones' cascade classifier while maintaining almost zero incorrect refusal; however, there are scopes for improvement. The principles utilized for recognizing the cups could be employed for detecting other objects as well, in a multitude of scenarios.

To enhance the accuracy of the model, more extensive training with a large dataset should be carried out – increasing the number of positive and negative images. Higher computing power could be used, and the discussed number of stages along with samples might be increased to upgrade the learning rate. A comparative study would be carried out by utilizing other training methods for achieving higher performance regarding object detection.

ACKNOWLEDGEMENT

We would like to extend our humble gratitude towards Dr. Mohamed El-Sharkawy – Professor of Electrical & Computer Engineering, Purdue School of Engineering & Technology, for his resolute pedagogical support. Without his support, the compiled work would not have been possible. Furthermore, we also offer our sincerest appreciation towards the Internet of Things Collaboratory lab for all the hardware support which has proven paramount for all the testing and proof of concepts.

REFERENCES

[1] H. Schneiderman, "Feature-centric evaluation for efficient cascaded object detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004, vol. 2, doi: 10.1109/cvpr.2004.1315141.

[2] E. T. Efaz, A. Al Mamun, K. Salman, F. Kabir, S. N. Sakib, and I. Khan, "Design of an indicative featured and speed controlled obstacle avoiding robot," in *2019 International Conference on Sustainable Technologies for Industry 4.0, STI 2019*, Dec. 2019, doi: 10.1109/STI47673.2019.9068018.

[3] E. T. Efaz, M. M. Mowlee, J. Jabin, I. Khan, and M. R. Islam, "Modeling of a high-speed and cost-effective FPV quadcopter for surveillance," in *23rd International Conference on Computer & Information Technology, ICACIT 2020*, Dec. 2020.

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, doi: 10.1109/cvpr.2001.990517.

[5] P. Viola and M. Jones, "Robust real-time face detection," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, p. 747, 2001, doi: 10.1109/ICCV.2001.937709.

[6] D. S. Chen and Z. K. Liu, "Generalized haar-like features for fast face detection," in *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, ICMLC 2007*, 2007, vol. 4, pp. 2131–2135, doi: 10.1109/ICMLC.2007.4370496.

[7] R. Yustiawati *et al.*, "Analyzing of Different Features Using Haar Cascade Classifier," in *Proceedings of 2018 International Conference on Electrical Engineering and Computer Science, ICECOS 2018*, Jan. 2019, pp. 129–134, doi: 10.1109/ICECOS.2018.8605266.

[8] D. Peleshko and K. Soroka, "Research of usage of Haar-like features and AdaBoost algorithm in Viola-Jones method of object detection," *2013 12th International Conference on the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*, Polyana Svalyava, 2013, pp. 284-286.

[9] C. Li, Z. Qi, N. Jia, and J. Wu, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers," in *ICEMI 2017 - Proceedings of IEEE 13th International Conference on Electronic Measurement and Instruments*, Jul. 2017, vol. 2018-January, pp. 483–487, doi: 10.1109/ICEMI.2017.8265863.

[10] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002, doi: 10.1109/TPAMI.2002.1017623.

[11] T. H. Le, "Applying Artificial Neural Networks for Face Recognition," *Adv. Artif. Neural Syst.*, vol. 2011, pp. 1–16, 2011, doi: 10.1155/2011/673016.

[12] A. S. Abstrak, "SISTEM PENGENALAN WAJAH MENGGUNAKAN METODE PRINCIPAL COMPONENT ANALYSIS (PCA) DENGAN ALGORITMA FUZZY C-MEANS (FCM)," *J. Pendidik. Mat.*, vol. 4, no. 2, pp. 58–65, 2015.

[13] Wirtjes and S. Jaceline, "Pengenalan Ekspresi Wajah Menggunakan Convolutional Neural Network (CNN)," *Repos. Institusi USU*, vol. 4, no. 3, pp. 4907–4916, 2019, [Online]. Available: http://repositori.usu.ac.id/handle/123456789/15450.

[14] R. Padilla, C. C. Filho, and M. Costa, "Evaluation of Haar Cascade Classifiers for Face Detection," *J. WASET*, vol. 6, no. 4, pp. 323–326, 2012.

[15] V. Singh, V. Shokeen, and B. Singh, "FACE DETECTION BY HAAR CASCADE CLASSIFIER WITH SIMPLE AND COMPLEX BACKGROUNDS IMAGES USING OPENCV IMPLEMENTATION," *Int. J. Adv. Technol. Eng. Sci.*, vol. 1, no. 12, pp. 33–38, 2013.

[16] W. Sulistiyo, B. Suyanto, I. Hestiningsih, - Mardiyono, and - Sukamto, "Rancang Bangun Prototipe Aplikasi Pengenalan Wajah untuk Sistem Absensi Alternatif dengan Metode Haar Like Feature dan Eigenface," *JTET (Jurnal Tek. Elektro Ter.*, vol. 3, no. 2, Aug. 2014, doi: 10.32497/JTET.V3I2.180.

[17] H. H. Lwin, A. S. Khaing, and H. M. Tun, "Automatic Door Access System Using Face Recognition," *Int. J. Sci. Technol. Res.*, vol. 4, no. 6, pp. 294–299, 2015.

[18] H. Yang and X. A. Wang, "Cascade classifier for face detection," doi: 10.1177/1748301816649073.

[19] I. Gangopadhyay, A. Chatterjee, and I. Das, "Face Detection and Expression Recognition Using Haar Cascade Classifier and Fisherface Algorithm," in *Advances in Intelligent Systems and Computing*, 2019, vol. 922, pp. 1–11, doi: 10.1007/978-981-13-6783-0_1.

[20] J. Jabin, A. M. Chowdhury, E. T. Efaz, M. E. Adnan, and M. R. Islam "An automated agricultural shading for crops with multiple controls," in *2020 International IOT, Electronics & Mechatronics Conference, IEMTRONICS 2020*, Sep. 2020.

[21] M. E. Adnan, N. M. Dastagir, J. Jabin, A. M. Chowdhury, and M. R. Islam, "A cost effective electronic braille for visually impaired individuals," in *5th IEEE Region 10 Humanitarian Technology Conference 2017, R10-HTC 2017*, Feb. 2018, vol. 2018-January, pp. 175–178, doi: 10.1109/R10-HTC.2017.8288932.

[22] J. Jabin, M. E. Adnan, S. S. Mahmud, A. M. Chowdhury, and M. R. Islam, "Low cost 3D printed prosthetic for congenital amputation using flex sensor," in *2019 5th International Conference on Advances in Electrical Engineering, ICAEE 2019*, Sep. 2019, pp. 821–825, doi: 10.1109/ICAEE48663.2019.8975415.

[23] "Cascade Classification — OpenCV 2.4.13.7 documentation." https://docs.opencv.org/2.4/modules/objdetect/doc/cascade_classification.html (accessed Aug. 06, 2020).

[24] "What is Object Detection Video - MATLAB & Simulink." https://www.mathworks.com/videos/what-is-object-detection-1564383482370.html (accessed Aug. 06, 2020).

[25] "S32V Vision and Sensor Fusion Evaluation System | NXP." https://www.nxp.com/design/development-boards/automotive-development-platforms/s32v-mpu-platforms/s32v-vision-and-sensor-fusion-evaluation-system:S32V234EVB (accessed Aug. 06, 2020).