Efficiency and Accuracy Improvement to the Viola-Jones Object Detection Algorithm Using Machine Learning Methods

J. Skelly
S. Latifi (Instructor)

Abstract: In 2001, Paul Viola and Michael Jones wrote a paper presenting their rapid object detection algorithm using machine learning methods, mainly proposed for the problem of face detection. Now twenty years later, the vast majority of face detection and face recognition algorithms written for modern applications are still based on the original Viola-Jones algorithm. The original algorithm uses a boosted cascade of simple rectangular features to quickly eliminate sub-windows which are determined not to contain a face almost instantly. The remaining sub-windows (those which may possibly contain a face) must pass a series of comparisons and tests involving increasingly complex classifiers for a face to be detected. If any of the tests along the way fail, the entire sub-window is determined not to contain a face. Modern face detection algorithms which are based on Viola-Jones look to improve upon the feature selection process, the problem of redundancy in desirable sub-windows, false positive and false negative detection rates, and training time. This paper discusses the machine learning methods implemented to improve the efficiency and accuracy of the original Viola-Jones algorithm for modern applications.

Keywords: Artificial Neural Network, Convolutional Neural Network, Face Detection, Face Recognition, Machine Learning, Neural Network, Object Detection, Redundancy Reduction, Viola-Jones Algorithm

I. Introduction

In the twenty-first century, face recognition technology is commonplace. Algorithms working to detect and recognize faces are programmed into security and surveillance systems, machines like ATMs and slot machines, and even everyday electronics like cell phones and laptops. According to research conducted by computer vision software company FaceFirst, Inc. [18], face recognition technology has found its way into a wide variety of applications "to make the world safer, smarter, and more convenient." These applications include, but are not limited to, prevention of retail crime, smart advertising, finding missing persons, aiding forensic scientists in investigations, protecting schools from potential threats, tracking attendance at various events, and even diagnosing certain diseases. The human brain is very good at detecting and recognizing faces. Depending on where a person goes to school or work, their brain may encounter hundreds or even thousands of faces each day, and the brain can *detect* each and every one of them. Those which are the most frequently *detected* (i.e., family, friends, coworkers, etc.) are then *recognized* by the brain as family members, friends, or coworkers. This detection and recognition of faces is second nature for the human brain, but getting a computer to accurately detect and recognize faces is a quite complex problem. In the late twentieth century, some algorithms were written to attempt to tackle the problem, but they were generally determined too slow to be useful in most applications.

While face recognition is the task which has the most well-known variety of applications, a face cannot be *recognized* without first being *detected.* Face detection is a sort of precursor to face recognition. Paul Viola and Michael Jones presented their rapid object detection algorithm [5] to the world in 2001 with the goal of solving the face detection problem in mind. The Viola-Jones object detection algorithm has been used for the past two decades to detect faces, and even with the rise of deep learning and artificial neural networks capable of solving a plethora of problems efficiently, Viola-Jones still offers a speed-accuracy tradeoff which keeps it prevalent. At the time when it was introduced, it was 15 times faster than the best competing algorithm (the Rowley-Baluja-Kanade detector) with a near 4% increase in detection accuracy [5]. The algorithm uses a boosted cascade of simple rectangular features (also called Haar-like rectangular features) to quickly eliminate sub-windows which most likely do not contain a face. The remaining sub-windows (those which may possibly contain a face) must pass a series of comparisons and tests for a face to be detected. If any of the tests along the way fail, the entire sub window is determined not to contain a face. The process of rapidly discarding sub-windows which most likely do not contain faces results in the impressive speed of the algorithm.

II. Overview of the Viola-Jones Algorithm

To better understand the operation of the original algorithm, it is important to dive deeper into the four main parts of the algorithm. The four main parts of the algorithm are the Haar-like rectangular features, the integral image, the boosting algorithm, and the cascade.

1) Haar-like Rectangular Features

The Haar-like rectangular features are used to classify a sub-window as either "potentially containing a face" or "certainly not containing a face." These features are classified into three groups: two-rectangle features, three-rectangle features, and four-rectangle features. From Figure 1, features 1 and 2 are classified as two-rectangle features. Feature 1 would likely be used to detect the difference in color in a person's eye and eyebrow area and their forehead. Feature 2 may be used to detect a vertical edge between someone's face and the background behind the face. Feature 3 would be classified as a three-rectangle feature and may be used to detect the difference in color between a person's eyes and the bridge of their nose. Feature 4 is a four-rectangle feature and could be used to detect diagonal differences on various parts of the face. Different combinations, orientations, and sizes of rectangular features like these give the algorithm a different clue as to whether a sub-window contains a face or not. For all these features, the sum of pixels in white rectangles get subtracted from the sum of pixels in gray rectangles to compute differences in darkness or lightness over a given sub-window. Over large numbers of pixels, the number of addition operations would get very large, and the algorithm would run very slow. To avoid this, Viola and Jones introduced to concept of the *integral image.*

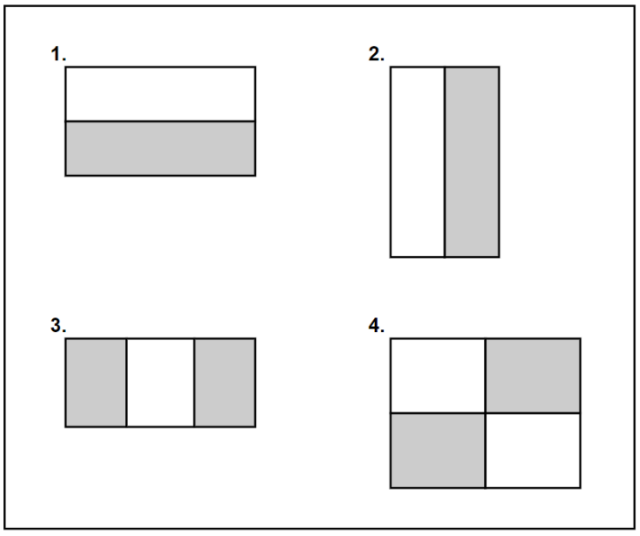


Figure 1: Examples of Haar-like Rectangular Features

2) The Integral Image

The integral image is a clever way of precomputing all the sums of pixel values that could possibly be needed during detection. In the original image, each pixel contains a value which represents the intensity value of that pixel. In a grayscale image, this value would be an integer between 0 (black) and 255 (white). The example original image shown in Figure 2 only contains pixel values which are multiples of 10 so that mental math is not tasking for ease of understanding the example. In the integral image, each pixel contains the value of the sum of all the pixels above and to the left of it, including itself. So, for this example, in the integral image, pixel (1,1) has a value of 240.

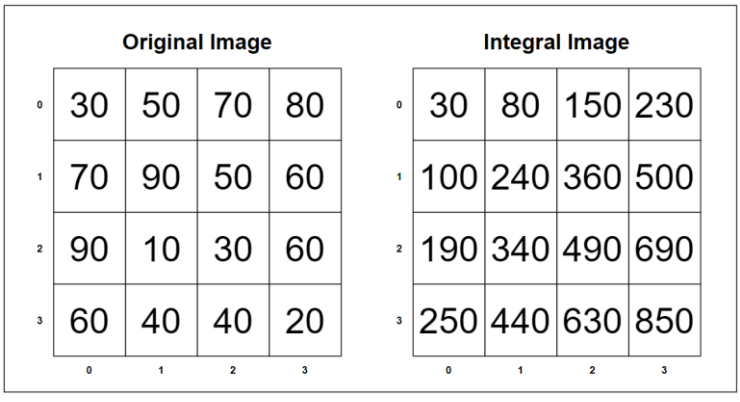


Figure 2: The Concept of the Integral Image

That value is obtained by summing pixels (0,0), (0,1), (1,0), and (1,1) from the original image. This sum is computed as 30 + 50 + 70 + 90 which yields a value of 240. Following the same process, pixel (3,3) is the sum of all the pixel values in the original image. This integral image is computed before any detection is performed so that for each region of the rectangle features, the number of computations is small, and the process is fast.

3) The Boosting Algorithm (AdaBoost)

When training the algorithm, it is important to explore every possible rectangular feature that can fit in a given image sub-window to determine which features are the most critical in detecting faces. Viola and Jones state, "Within any image sub-window the total number of Haar-like features is very large, far larger than the number of pixels. In order to ensure fast classification, the learning process must exclude a large majority of the available features, and focus on a small set of critical features," [5]. This is the motivation for *boosting*, the process of converting "weak learners" into "strong learners" in machine learning. On their own, each of the rectangular Haar-like features are "weak learners" and would do a poor job detecting faces in image sub-windows. Viola-Jones uses an adapted AdaBoost algorithm to select which features are the most critical.

AdaBoost starts with a weak learner and classifies every sample as either a positive or negative sample. Specifically, AdaBoost starts with the weak learner which best classifies each of the sub-windows independent of any other features. In the case of face detection, one example of a weak learner would be feature 1 from Figure 1. Assume this feature is the feature which best classifies all the sub-windows initially. AdaBoost will run this one particular feature of one particular size over each sub-window in the training data set and classify each as a positive example (may contain a face) or a negative example (does not contain a face). It is clear that this feature alone will do a poor job of separating the samples, but assume that it does a *better job* than any of the other features do as they stand alone. The error is computed, and in the next round, emphasis is placed on correctly classifying all those sub-windows which were misclassified in the previous round. A new weak learner is selected to classify all the training data over again, but this time extra emphasis is placed on correctly classifying the previously misclassified sub-windows.

When training is complete, there is a large number of weak learners, each with a different amount of say in whether or not a particular sub-window contains a face. The amount of say is based on how well that feature classified the weighted dataset. The features which have the least amount of say can then be discarded or left out, and only those with the largest amount of say are used to perform the final classification of new images. Discarding the features with the least amount of say results in an insignificant increase in error. This is because certain features do a very poor job of classifying the dataset, so their say in the decision is insignificant. Using this process, Viola and Jones were able to decrease the number of features from 180,000 to only a few hundred with noteworthy accuracy. From [5], "Initial experiments demonstrated that a frontal face classifier constructed from 200 features yields a detection rate of 95% with a false positive rate of 1 in 14084." This accuracy, however, was deemed "not sufficient for many real-world tasks" by

Viola and Jones. The final detector has over 6000 features but is still 15 times faster than the competing algorithms of its time.

4) The Cascade

The final detector takes the form of a cascade, also called a "degenerate decision tree," in which a positive result from the first classifier results in an evaluation of a second classifier, and a positive result from the second classifier results in an evaluation of a third, and so on all down through the cascade. The features present in the classifiers are determined by the AdaBoost process described previously, and each stage of the cascade contains a more complex classifier than the stage preceding it. If at any stage the result is a negative, the entire sub-window is rejected. If a sub-window makes it all the way through the cascade without rejection, then the sub-window is determined to contain a face. This cascade structure is also referred to as "the *attentional* cascade" because it rapidly rejects sub-windows which most likely do not contain a face so that the detector can pay more *attention* to the sub-windows which may contain a face. The rapid rejection of non-face sub-windows results in the impressive speed of the detector.

Results of the Original Algorithm

Viola and Jones trained their detector classifiers on a training set comprised of 4916 faces scaled and aligned to a resolution of 24x24 pixels. The classifiers were each individually trained on 10000 non-face images and 9832 face images (each of the 4916 faces in their original form and mirrored vertically). On a 700 MHz Pentium III processor, the final detector could scan a 384x288 pixel image in about 1/15 of a second, or 15 frames per second. Below is a table from the original Viola-Jones paper showing detection rates of the Viola-Jones algorithm compared with other popular face detection algorithms in 2001. While it is important to note that the number of false detections increases as the detection rate increases, the Viola-Jones algorithm has similar or better accuracy than its predecessors coupled with a massive increase in speed (speed of detectors is not shown in the table).

False detections / Detector	10	31	50	65	78	95	167
Viola-Jones	76.1%	88.4%	91.4%	92.0%	92.1%	92.9%	93.9%
Viola-Jones (voting)	81.1%	89.7%	92.1%	93.1%	93.1%	93.2 %	93.7%
Rowley-Baluja-Kanade	83.2%	86.0%	-	-	-	89.2%	90.1%
Schneiderman-Kanade	-	-	-	94.4%	-	-	-
Roth-Yang-Ahuja	-	-	-	-	(94.8%)	-	-

Figure 3: Table Showing Viola-Jones Detection Rates Compared with Other Algorithms in 2001 [5]

III. Applications Outside of Face Detection

The Viola-Jones algorithm was proposed for, and has been mainly used for, the application of face detection. However, there is a wide variety of applications that the algorithm has been used for in recent years outside of face detection. The 2001 paper [5] introduces and tests the algorithm as a face detection algorithm for the sake of example, and because face detection was a problem that still needed a reasonably fast and efficient solution in 2001. Nonetheless, the algorithm can be applied to different objects using the same features and the same training methodology. For example, the following applications have used the Viola-Jones algorithm:

1. Detection and tracking of locomotive activity of animals in wildlife videos, [15].
2. Emotion recognition to determine success in the learning environment, [3], [14].
3. Drowsiness detection to improve BCI, prevent road accidents, [4], [13].
4. Vehicle counting system for traffic monitoring and surveillance, [11].
5. Hand gesture recognition in real time, [12].

Locomotive Activity Monitoring of Animals

In 2006, a paper was written by UK computer scientists Tilo Burghardt and Janko Calic titled "*Real-time Face Detection and Tracking of Animals*," [15]. The proposed algorithm uses the Viola-Jones object detection algorithm to first detect an animal's face from a wildlife rush. After the face is detected, a different algorithm takes over to track the animal's locomotive behavior. Burghardt and Calic trained their algorithm using 680 images of lions faces and 1000 images which did not contain a lion's face. They were able to obtain a false detection rate of 1 in 10,000 classifications using just 250 features. The final detection rate of the algorithm was 93%. An example of some rectangular Haar-like features used in the animal face detection algorithm and the resulting detected true positive windows are shown in Figure 4 below.

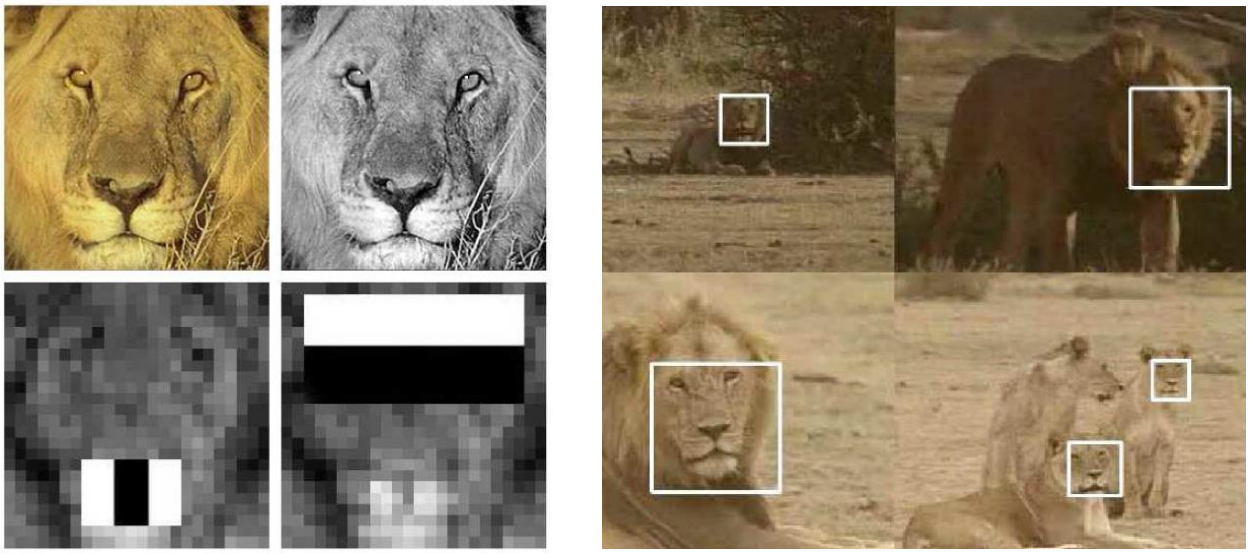


Figure 4: Example of Haar-like Rectangular Features Used to Detect the Face of a Lion [15]

Emotion Recognition in the Learning Environment

In 2018, Indonesian engineers Kartika Candra Kirana et al. wrote two papers titled "*Emotion Recognition using Fisher Face-Based Viola-Jones Algorithm*," [14], and "*Facial Emotion Recognition based on Viola-Jones Algorithm in the Learning Environment,"* [3]. The authors claim that emotion is a good teller of whether a student is having success in the learning environment. For example, the student may look bored or confused if they are having a hard time with the material being presented, and excited or enthusiastic if they understand the material. There are many emotions a person can express through facial features and gestures, but the method discussed in [3] categorizes the faces detected into four categories, namely: interested in subject matter, confused or having difficulty with subject matter, frustrated by subject matter, or pondering new ideas. Interested and pondering are positive emotions, while confused and frustrated are negative emotions. However, the most promising results shown in the paper only test whether a student is bored or interested. These results can be seen below for three different algorithms tested by the authors of the paper, where the proposed Viola-Jones-based method with no Neural Network integration has the best results for the lowest time complexity. In the table TP, TN, FP, and FN stand for True Positive, True Negative, False Positive, and False Negative, respectively.

The Algorithm	Result							
	TP	*TN*	*FP*	*FN*	*accuracy*	*precision*	*recall*	*Time*[c]
Viola-Jones	38	36	14	12	0.74	0.73	0.76	15
Viola-Jones + Neural Network [8]	40	36	14	10	0.76	0.74	0.8	38
Neural Network [1]	26	22	28	24	0.48	0.48	0.52	42

Figure 5: Results of Viola-Jones-Based Method for Emotion Detection [3]

Drowsiness Detection

Two papers with different end goals in mind have been proposed on the topic of drowsiness detection with the use of Viola-Jones. The first, titled "*Real Time Drowsiness Detection using Viola Jones & KLT*," was written by Indian electronics and communications engineers Hilkiya Joseph and Bindhu K. Rajan [13]. The second, titled "*Drowsiness Detection for the Perfection of Brain Computer Interface Using Viola-Jones Algorithm*," was written by Md. Kamrul Hasan et al. from the Department of Electrical and Electronics Engineering at Khulna University of Engineering and Technology in Khulna, Bangladesh [4]. The authors aim to eliminate

contamination in Electroencephalogram (EEG) signals in brain-computer interfaces (BCI). One of the main contributing factors to EEG signal contamination is drowsiness, so the "main target is to determine the level of drowsiness from the patient's EEG signal" to then adjust the BCI accordingly in the best interest of the patient. The authors of [13] aim to design a system which will sound an alarm to alert a drowsy driver as they may doze off on the road to prevent auto accidents from occurring as a result of a drowsy driver. Both algorithms use the Viola-Jones algorithm to first detect the face, and then transition to a state of eye openness monitoring. Figure 6 below shows how the algorithms rate drowsiness based on eye openness level.

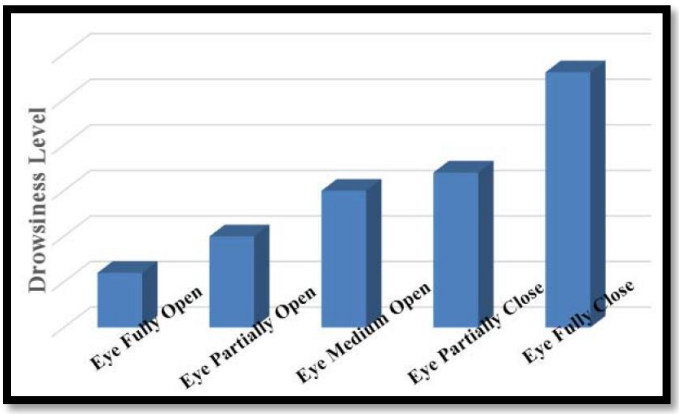


Figure 6: Assumed Drowsiness Level vs. Eye Openness [4]

Vehicle Counting for Traffic Surveillance

Traffic congestion is a problem in many cities in the US and around the world. Engineers in Makassar, Indonesia believe they can solve the problem in their city by using an Intelligent Transportation System (ITS). This ITS would use the Viola-Jones method of object detection to count the number of vehicles on the road and determine the necessary turnover time of traffic lights accordingly to keep the roads from getting too congested. In the experimentation done in [11], Andani et al. achieved 92% maximum detection accuracy by training the algorithm with 150 positive samples and 300 negative samples. The test sample space had 30 total samples. According to the authors, "There are two ways to practice the positive samples; the first is determining bounding boxes on the ROI (Region of Interest) and the second is to crop the car image." The algorithm was written to detect the car front view. Experimental results showed 85% accuracy and 80% accuracy when trained with data from two other databases.

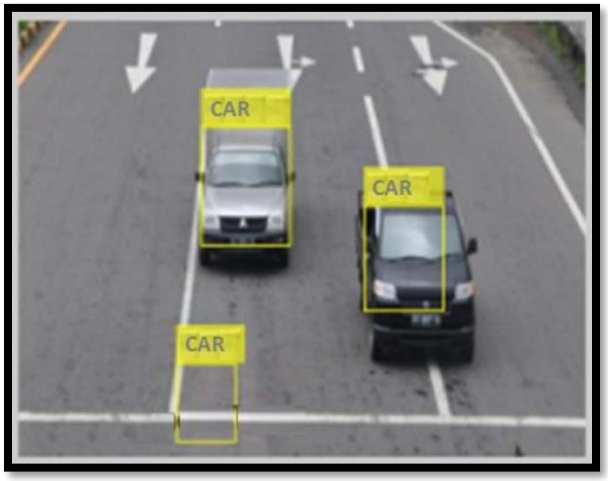


Figure 7: 2 True Positives, 1 False Positive Sample [11]

Hand Gesture Recognition

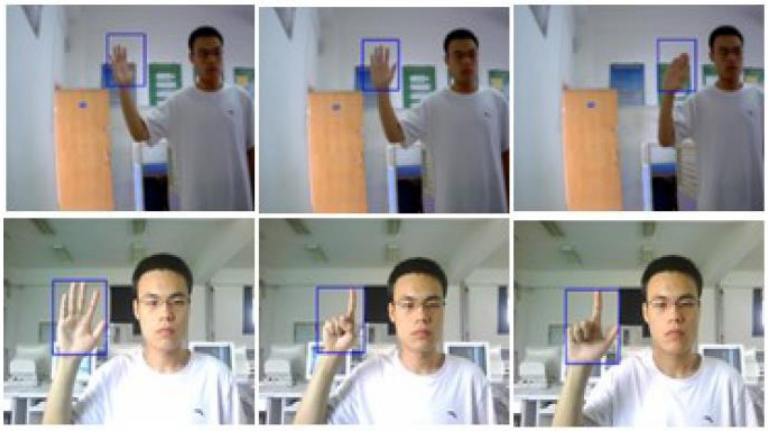


Figure 8: Viola-Jones Used to Detect Region of Interest [12]

Hand gesture recognition is very important in the realm of human-computer interaction (HCI). "Hand gestures are [a] powerful human-to-human communication channel which convey a major part of information transfer in our everyday [lives]. Hand gestures are the natural way of interactions when one person is communicating with another and therefore hand gestures can be treated as a nonverbal form of communication…Hand gesture recognition is of great importance for human computer interaction (HCI) because of its extensive applications in virtual reality and sign language recognition, etc.," [21]. Authors Liu Yun and Zhang Peng of the Qingdao University of Science and Technology in Qingdao, China [12] present a hand gesture recognition system based on the Viola-Jones object detection algorithm and Support Vector Machines (SVMs). The Viola-Jones algorithm in their process is used to accurately locate the hand region, while the SVM is used to classify and extract hand gestures from the region of interest. In Figure 8 above, the blue rectangle is the region determined to contain a hand using the Viola-Jones algorithm. The proposed method also introduced new extended Haar-like features to capture hand gestures that the original set of features was unable to capture. The algorithm was used to determine three different hand gestures, namely: Palm, Index, Lpalm. The hand gestures are shown in Figure 9. The algorithm showed detection rates of 90% or higher for all three hand gestures with the original Haar-like features and showed a near 5% average increase in detection rate when the extended features were introduced.

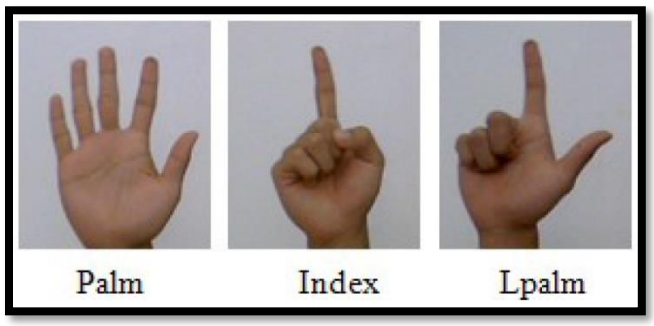


Figure 9: Hand Gestures to be Detected [12]

IV. Improvements to the Algorithm

Now, over two decades after the algorithm was first introduced, Viola-Jones is still cited often and used in modern face detection algorithms, along with a variety of other applications. The paper introducing the algorithm has been cited in 7222 papers and 644 patents [5]. While it performs well and is still the foundation of many modern algorithms, several engineers and

computer scientists in recent years have produced algorithms with even better detection rates and higher speeds by making small tweaks to the original algorithm. These improvements include:

- Application of composite features to reduce false detection rate, [1].
- Reducing redundancy in face sub-window selection, [2].
- Reducing overfitting and training time by pre-selecting features, [7].

Application of Composite Features to Reduce False Detection Rate

Perhaps the biggest problem with the original Viola-Jones algorithm for face detection is the false detection rate. Even with over 6000 features taken into consideration, the algorithm detects non-face sub-windows as faces at a rate that is not negligible. The test data set used to generate the results shown earlier in Figure 3 contains 507 total faces in a set of 130 images. At peak detection rate, the Viola-Jones algorithm detected 167 faces that were not actually faces. That comes out to an average of more than 1 false detection per image, which is not easy to overlook. In a paper written in 2019 for the International Conference on Robots and Intelligent Systems (ICRIS), authors Lu Wen-Yao and Yang Ming discuss, with results, an improved version of the Viola-Jones algorithm which uses composite features. These composite features are an adaptation of the original Haar-like rectangular features used by Viola and Jones. Instead of using just one rectangular feature, Wen-Yao and Ming propose a composite feature, or a vector of simple features as the weak learner in their algorithm. "The weight information of distinguishing features is used to evaluate the validity of basic features in face recognition. Then the [features with the largest weights are] selected to form the composite feature vector," [1].

Table 3 Comprehensive comparison of the two algorithms

Algorithm name	Total face number	Total missing count	Total error count	Total missing rate	Gross error rate
Viola-Jones	1372	189	21	0.138	0.015
Article method		75	1	0.055	0.0007

Figure 8: Table Showing Viola-Jones Improvement with Composite Features [1]

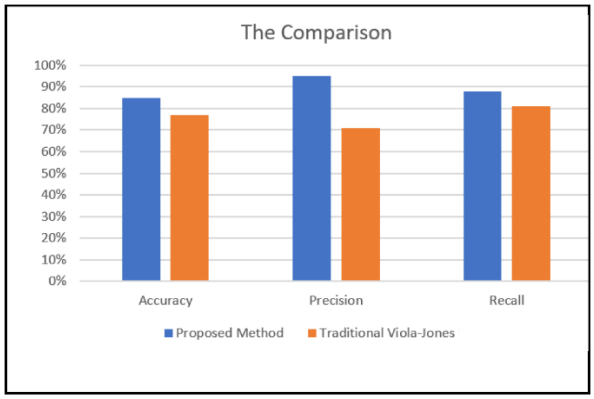
The authors conducted 10 experiments and used 1000 images for experimental data. "By comparing the composite feature method based on [the] Viola-Jones algorithm with the original Viola-Jones algorithm in experiments, the accuracy of [the] textual method used in the face recognition process is illustrated," [1]. From the table of Figure 10 above, the Viola-Jones algorithm fell short in both false positives and false negatives. The proposed method from the text only missed 75 of the 1372 faces in the experimental dataset, while the original Viola-Jones algorithm missed over twice as many. The proposed method also minimized the number of false positives to just 1 over 1000 images, while Viola-Jones counted 21 objects as faces incorrectly.

Reducing Redundancy in Face Sub-Window Detection

Another problem with the original Viola-Jones algorithm is the problem of redundant face detection. Redundancy is a seemingly inescapable issue in face detection. This is because one sub-window which contains a face is almost always contained in a larger sub-window which contains the same face, resulting in the same face being recognized multiple times in different but similar sub-windows. An example of the issue of redundancy can be observed in Figure 11 below. The image set on the left shows multiple instances of the same face being detected using the original Viola-Jones algorithm. The image on the right shows results from a different detection algorithm which proposes a solution to the redundancy problem, where only one window is selected from the many which contain the face. This solution was proposed by Kirana et al. from the Department of Electrical Engineering at Universitas Negeri Malang in Malang, Indonesia [2]. The paper was presented in the 2020 4th International Conference on Vocational Education and Training.

Figure 9: Redundancy Issue in Viola-Jones (left) Contrasted with No Redundancy Issue (right) [2]

The authors of [2] propose a Hill Climbing algorithm which considers all the detected positive sub-windows and chooses a local maximum from the set of sub-windows. From the paper, as a direct result of the redundancy reduction, “there are improvements to the results on precision, recall, and accuracy.” The authors collected 685 random images with data labels on 900 faces. Two tables showing the experimental results from [2] are shown in Figure 12 on the page to follow. The original Viola-Jones algorithm missed 220 of the 900 faces, while the reduced redundancy algorithm only missed 160.The original Viola-Jones algorithm also spotted 225 false positives compared to a remarkably lower number of just 35 false positives from the reduced redundancy algorithm. Overall, the proposed algorithm of [2] is 85% accurate compared to the original algorithm which has an accuracy of 77% on the experimental dataset stated previously. The proposed method also has 95% precision compared to only 71% precision with the traditional Viola-Jones algorithm.



No	Comparison		
	Test	*Proposed Method*	*Traditional Viola-Jones* [2]
1	True Positive	740	520
2	True Negative	25	0
3	False Negative	100	155
4	False Positive	35	225
5	Accuracy	85%	77%
6	Precision	95%	71%
7	Recall	88%	81%

Figure 12: Results from Redundancy Reduction Algorithm Showing Improvements in Accuracy, Precision [2]

Reducing Overfitting, Training Time by Pre-Selecting Features

Before the Viola-Jones algorithm can detect faces, time must be taken to train the classifiers and determine which features are the most critical. To find which features are the most critical, every size and scale of each rectangular feature must be checked. For just a small 24x24 pixel image sub-window, "there are over 180,000 rectangle features…," [5]. As the image size gets larger for different applications of the algorithm, the training time becomes much longer. As stated in a paper written by Simon R. Lang et al., "Viola-Jones draws upon a set of simple, Haar-like image features at all scales and positions. As this set grows rapidly with image size, it can become costly to evaluate and also encourages overfitting of the classifier," [7]. Overfitting occurs when an algorithm fits its training data so well that it cannot handle new test data well. As the in-sample (training) error approaches zero, the out-of-sample (generalization) error should also decrease, but it will reach a minimum. Once the out-of-sample error is minimized, this is where the algorithm should ideally operate. If the in-sample error becomes too small, it is likely that the algorithm has performed a fit on the noise, and the out-of-sample error will increase dramatically. This can become a problem for the Viola-Jones algorithm on higher-resolution images if the images are not normalized and scaled down before training. According to [7], "some of the original Viola-Jones research involved weeks of training per cascade… Being able to pre-select a smaller set of features from which to build classifiers would save significant time."

Lang et al. propose a method of pre-selecting the most critical features which leads to classifiers with improved accuracy and reduces both the effects of overfitting and the training time. The authors discuss two ways that classifiers can be improved by reducing features. The first is known as the "filter" method, which identifies an optimal feature subset "by using abstract measures relevant to important properties of the feature set." The other is known as the "wrapper" method, which identifies an optimal feature subset "by building a classifier from the feature subset and evaluating its performance on actual classification tasks." [7] Since the wrapper method is more reliable at optimizing classification performance, the proposed algorithm uses the wrapper method. It also uses what is known as an EA or evolutionary algorithm to select features and

determine their success or "fitness" based on the number of false positives out of the final stage of the cascade. The process of feature selection is iterative. As features are adjusted and the fitness of each adjustment is measured, the number of stages in the cascade is increased (3 stages initially up to a maximum of 12 stages).

Cascade	Evaluated	Recall	Precision	False Pos. Rate	Stages	Features
Control	1	0.9398	0.0925	0.3931	12	68
Evolved	120	0.9297±0.0022	0.1044±0.0033	0.3867±0.0141	11.33±0.07	76.46±1.05
+ Perf.	26	0.9487±0.0011	0.1054±0.0027	0.3493±0.0090	11.35±0.15	75.46±1.79
+ Perf., - Stages	13	0.9501±0.0015	0.1019±0.0030	0.3605±0.0097	10.85±0.19	71.54±2.32
+ Perf., - Stages, - Feat.	5	0.9493±0.0023	0.0988±0.0003	0.3692±0.0005	10.2±0.2	64±0

Figure 10: Resulting Cascades from Evolutionary Feature Preselection and Their Performance [7]

The experimental methods discussed in [7] resulted in a total of 120 new cascades, 26 of which "have better precision and recall than the control cascade. Some of these cascades also have fewer stages than the control; typically around 1-2 fewer stages." After reducing the subset of 26 cascades even further by only examining those which have fewer stages than the control cascade (the cascade of the original Viola-Jones algorithm), 13 cascades remain, all of which have better performance and fewer stages than the control cascade. Of these 13 cascades, 5 of them also make use of fewer features. Figure 13 above shows a table of results from the paper including the 120 evolved cascades, the 26 with better performance than the control, the 13 with both better performance and fewer stages, and the 5 with better performance, fewer stages, and fewer features. The evolutionary feature preselection process described in [7] shows a decrease in false positive detection rate and an increase in precision over all 120 evolved cascades.

V. Comparison to Other Modern Algorithms

After examining the shortcomings of the Viola-Jones algorithm and the clever ways computer scientists and engineers have improved the algorithm in recent years, a comparison of the 2001 algorithm to modern algorithms is in order. According to computer scientists Kirti Dang and Shanu Sharma of Amity University in Noida, India, the four most common face detection algorithms are:

1. Viola-Jones
2. SMQT Features and SNOW Classifier
3. Neural Network-Based Face Detection
4. Support Vector Machine-Based Face Detection

In their paper [19] titled "*Review and Comparison of Face Detection Algorithms,*" Dang and Sharma analyze the performance of the four algorithms listed above on the basis of precision and recall measurements. **Precision** is the ratio of true positive detections to all positive detections,

while **recall** is the ratio of true positive detections to the number of actual relevant elements in the dataset. An example of relevant elements in a dataset would be the number of actual faces present in a set of 1000 images. If there are 950 faces in 1000 images, the number of relevant elements is 950, and the recall would be the ratio of faces correctly detected to the 950 actual faces in the dataset. The closer precision and recall values are to 1, the more ideal the algorithm is. From Figure 14 below, the neural network-based face detection algorithm has the clear lead in precision performance. However, the recall performance of the NN-based algorithm is very poor. Compared against the other three algorithms overall, Viola-Jones has the best performance results based on the research conducted in [19].

S.No	Face Detection Algorthims	Precision	Recall
1	Viola-Jones face detector	0.27321	0.27321
2	SMQT Features and SNOW Classifier	0.26792	0.26792
3	Neural Network-Based Face Detection	0.339450	0.037582
4	Support Vector Machines-Based face detection	0.01392850	0.00835708

Figure 11: Table of Precision and Recall Measures for Different Face Detection Algorithms [19]

In a different paper [17], "*A Comparison of CNN-based Face and Head Detectors for Real-Time Video Surveillance Applications,*" authors Eric Granger et al. highlight the enhanced performance of CNN-based (Convolutional Neural Network) face detection compared with other algorithms including Viola-Jones. Both the precision measure and the true positive rates of the CNN-based algorithms are superior to the Viola-Jones algorithm. The CNN-based algorithm of [17] showcases a precision measure of 92% with a positive detection rate of 0.93, while the Viola-Jones algorithm has a precision measure of 67% with a positive detection rate of 0.67. These results were obtained testing the algorithms on a dataset called "FDDB" or "Face Detection Dataset and Benchmark" which consists of 5171 labeled faces in images of varying resolution. Per [17] the dataset also includes some challenges, such as "difficult pose angles, out-of-focus faces and low resolution." The authors conclude that despite the large gap in precision and positive detection performance, "even with the fastest CNN architectures, the time complexity is high compared to the Viola-Jones detector." Though not mentioned explicitly, the Viola-Jones algorithm is shown to have far less memory consumption than competing CNN-based algorithms as well. This, along with the superior speed of the Viola-Jones algorithm, can be observed in the table of Figure 15 on the page to follow.

Detector	Time		Memory consumption (GB)
	GFLOPS	FPS	
Viola-Jones [18]	0.6	60.0	0.1
HeadHunter DPM [20]	5.0	1	2.0
SSD[6]	45.8	13.3	0.7
Faster R-CNN [5]	223.9	5.8	2.1
R-FCN 50 [3]	132.1	6.0	2.4
R-FCN 101 [3]	186.6	4.7	3.1
PVANET [13]	40.1	9.0	2.6
Local RCNN [19]	1206.8	0.5	2.1
Yolo 9000 [16]	34.90	19.2	2.1

Figure 12: Time and Memory Complexity of Viola-Jones Compared to CNN-Based Algorithms [19]

While other competing algorithms have been written and perform well for applications where speed is not a necessity, the Viola-Jones algorithm has stood the test of time and remained relevant as a direct result of its robustness and its speed. The original Viola-Jones algorithm operated in 2001 at 15 frames per second, but newer technology and minor adjustments made to the algorithm [20] have boosted the speed of Viola-Jones to 60 frames per second, as shown in Figure 15. Since face detection is often used as a preprocessing step in a broader application, such as facial recognition, facial feature recognition, facial analysis, facial tracking, etc., Viola-Jones has yet to face any real danger of obsoletion.

VI. Conclusion

This paper introduces and explains the operation of the original Viola-Jones rapid object detection algorithm for the application of face detection, along with several applications outside of face detection. It dives into the improvements which have been made to the original algorithm and examines the comparison of the original Viola-Jones algorithm results with the results of the enhanced algorithms. Enhancements to the algorithm which have improved efficiency and accuracy include the introduction of composite features into classifiers, redundancy reduction, and pre-selection of critical features. The Viola-Jones algorithm is also compared to other modern algorithms for face detection, mainly those which use neural networks. Though the neural network-based algorithms show higher detection rates and higher accuracy and precision, the comparatively low time complexity of the Viola-Jones algorithm keeps it relevant in a broad range of applications, even today. Promising detection rates and accuracy are displayed across application of the algorithm to animal face detection, emotion recognition, traffic congestion control via vehicle recognition and tracking, and hand gesture recognition and classification. Work is still being done to improve algorithms attempting to detect drowsiness with hopes to prevent auto accidents.

Face detection is required to further analyze the face for recognition of different faces or facial features. “It is the first step for face recognition, face analysis and detection of other features of the face,” [19]. Of course, in applications where very high detection rates and other metrics are of the essence and where speed is not an issue, the neural network-based algorithms have Viola-Jones beat. However, because of its high frame rate and reasonable accuracy, precision, recall, and detection rates, the Viola-Jones algorithm has remained prominent in the machine learning community for over two decades, and there is little reason to believe that this will change in the years to come.

References

[1] W. Lu and M. Yang, "Face Detection Based on Viola-Jones Algorithm Applying Composite Features," 2019 International Conference on Robots & Intelligent System (ICRIS), Haikou, China, 2019, pp. 82-85, doi: 10.1109/ICRIS.2019.00029.

[2] K. C. Kirana, S. Wibawanto and H. W. Herwanto, "Redundancy Reduction in Face Detection of Viola-Jones using the Hill Climbing Algorithm," 2020 4th International Conference on Vocational Education and Training (ICOVET), Malang, Indonesia, 2020, pp. 139-143, doi: 10.1109/ICOVET50258.2020.9230349.

[3] K. Candra Kirana, S. Wibawanto and H. Wahyu Herwanto, "Facial Emotion Recognition Based on Viola-Jones Algorithm in the Learning Environment," 2018 International Seminar on Application for Technology of Information and Communication, Semarang, Indonesia, 2018, pp. 406-410, doi: 10.1109/ISEMANTIC.2018.8549735.

[4] M. K. Hasan, S. M. Hasnat Ullah, S. S. Gupta and M. Ahmad, "Drowsiness detection for the perfection of brain computer interface using Viola-jones algorithm," 2016 3rd International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), Dhaka, 2016, pp. 1-5, doi: 10.1109/CEEICT.2016.7873106.

[5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[6] Q. Li, U. Niaz and B. Merialdo, "An improved algorithm on Viola-Jones object detector," 2012 10th International Workshop on Content-Based Multimedia Indexing (CBMI), Annecy, France, 2012, pp. 1-6, doi: 10.1109/CBMI.2012.6269796.

[7] S. R. Lang, M. H. Luerssen and D. M. W. Powers, "Evolutionary Feature Preselection for Viola-Jones Classifier Training," 2012 Spring Congress on Engineering and Technology, Xi'an, China, 2012, pp. 1-4, doi: 10.1109/SCET.2012.6342142.

[8] M. Nehru and S. Padmavathi, "Illumination invariant face detection using viola jones algorithm," 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2017, pp. 1-4, doi: 10.1109/ICACCS.2017.8014571.

[9] M. Pound and S. Riley, “Detecting Faces (Viola Jones Algorithm) – Computerphile,” October, 2018. [Online]. Available: https://www.youtube.com/watch?v=uEJ71VlUmMQ. [Accessed March 15, 2021].

[10] Huang, J., Shang, Y. & Chen, H. Improved Viola-Jones face detection algorithm based on HoloLens. J Image Video Proc. 2019, 41 (2019). [Online]. Available: https://jivp-eurasipjournals.springeropen.com/articles/10.1186/s13640-019-0435-6#citeas. [Accessed March 17, 2021]. https://doi.org/10.1186/s13640-019-0435-6.

[11] D. Djamaluddin, T. Indrabulan, Andani, Indrabayu and S. W. Sidehabi, "The simulation of vehicle counting system for traffic surveillance using Viola Jones method," 2014 Makassar International Conference on Electrical Engineering and Informatics (MICEEI), Makassar, Indonesia, 2014, pp. 130-135, doi: 10.1109/MICEEI.2014.7067325.

[12] L. Yun and Z. Peng, "An Automatic Hand Gesture Recognition System Based on Viola-Jones Method and SVMs," 2009 Second International Workshop on Computer Science and Engineering, Qingdao, China, 2009, pp. 72-76, doi: 10.1109/WCSE.2009.769.

[13] H. Joseph and B. K. Rajan, "Real Time Drowsiness Detection using Viola Jones & KLT," 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2020, pp. 583-588, doi: 10.1109/ICOSEC49089.2020.9215255.

[14] K. C. Kirana, S. Wibawanto and H. W. Herwanto, "Emotion Recognition using Fisher Face-based Viola-Jones Algorithm," 2018 5th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Malang, Indonesia, 2018, pp. 173-177, doi: 10.1109/EECSI.2018.8752783.

[15] T. Burghardt and J. Calic, "Real-time Face Detection and Tracking of Animals," 2006 8th Seminar on Neural Network Applications in Electrical Engineering, Belgrade, Serbia, 2006, pp. 27-32, doi: 10.1109/NEUREL.2006.341167.

[16] J. Starmer, "AdaBoost, Clearly Explained," January, 2019. [Online]. Available: https://www.youtube.com/watch?v=LsK-xG1cLYA&t=77s. [Accessed April 10, 2021].

[17] L. T. Nguyen-Meidine, E. Granger, M. Kiran and L. Blais-Morin, "A Comparison of CNN-based Face and Head Detectors for Real-Time Video Surveillance Applications," 2017 The 7th International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, Canada, Nov. 28 to Dec. 1, 2017.

[18] "Amazing Uses for Face Recognition," 2021. [Online]. Available: https://www.facefirst.com/blog/amazing-uses-for-face-recognition-facial-recognition-use-cases/. [Accessed April 27, 2021].

[19] K. Dang and S. Sharma, "Review and comparison of face detection algorithms," *2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence*, 2017, pp. 629-633, doi: 10.1109/CONFLUENCE.2017.7943228.

[20] P. Viola and M. J. Jones. Robust real-time face detection. Int. J. Comput. Vision, 57(2):137–154, May 2004.

[21] Francis, Jobin & Kadan, Anoop. (2014). Significance of Hand Gesture Recognition Systems in Vehicular Automation-A Survey. International Journal of Computer Applications. 99. 50-55. 10.5120/17389-7931.