International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Pragamatic Approach for Real Time Face Tracking

Dr. Ranganatha S.[1], Dr. Thirthe Gowda M. T.[2*], Dr. Shivashankara S.[3], Dr. Ravi P.[4]

Submitted: 18/10/2023 **Revised**: 07/12/2023 **Accepted**: 16/12/2023

Abstract: Live detection and tracking of face(s) is a challenging task in the field of video processing. This paper proposes the detection and live tracking of a single or multiple faces using live video sequences from a webcam with few technical challenges such as partial occlusion, different light intensity levels, and when camera faces different angles. In today's reality, video analysis is an important task that can help in difficult times. Real time face detection and tracking is one such part of video analysis which is tremendously in use. Today's productive cameras and high efficiency computers have made the researchers to develop better algorithms for processing the video sequences. Every now and then, system vision and the things we can achieve using this is spoken for versatility in the field of computer vision. Proposed algorithm detect and track the faces very efficiently even when there are constraints from external factors.

Keywords: *Live Detection and Tracking, Video Sequences, Technical Challenges, Video Analysis, Algorithm*

1. Introduction

On everyday premise preparing a video is turning out to be a lot simpler that is a direct result of the apparatuses accessible in the cutting edge world. This paper has one such arrangement of calculations among which one distinguishes the face and one tracks the identified face. Developing in the field of science has driven the world needing such an innovation where the framework must distinguish the face and track the face on ongoing premise than from the recordings which are put away in the database. Constant applications have colossal extension in the realm of science and innovation. Enormous experimentation in the field of video investigation has been done, yet at the same time our reconnaissance frameworks need programmed video examination. We utilize the facial highlights for the identification purposes and build up the focuses utilizing LDA (Linear Discriminant analysis) and begin following those focuses for additional following.

Each calculation utilized with the end goal of video investigation consistently consider the facial element which are typically the district of premium. We cannot have datasets or anything, for example, databases for this calculation. All the sources of information given to the calculation is new and veritable. For any human and computer to connect with one another, discovery and following of a face and its facial component is significant. Various papers related to this are accessible which tells about various advancements. This paper focuses more on LDA than PCA (Principal Component Analysis). LDA gives us preferred outcomes over PCA.

1.1. TASK ILLUSTRATION AND ORGANIZATION OF THE PAPER

Each time, proposed algorithm goes with a new input video sequence from the camera as the task is to detect and track the face on real time basis. At first, it extracts the facial feature from the input video. In the end, we are interested in classification than representation; so, in these scenarios we wish to find the subspace that will map the sample vectors in the same class that to in a single spot of the feature representation along with variety of classes as far apart from one to other. LDA is one of the familiar DA (Discriminate Analysis) method that achieves this goal. After this, detected face(s) are tracked further.

2. Literature Review

Security in biometric systems has risen due to law enforced applications, and the range of surveillance and access control. This fact is highlighted in the work published in literature [1], [2]. Viola-Jones [3] method utilized Haar-like features during face detection from the video frame. This method performs reliably only for frontal faces. It requires suitable modifications for detecting the faces that are having different poses. To overcome the drawbacks, Jatin Chatrath et al. [4] have used a variant of the algorithm which was explained in literature [3].

Comaniciu and Meer [5] have proposed a technique for analyzing the difficult multimodal feature space and delineating unstructured groups in it. Here, the key module

[1]*Department of Computer Science and Engineering, Government Engineering College, Kushalnagar, Karnataka, India*
E-mail: ranganath38@yahoo.co.in
[2]*Department of Computer Science and Engineering, Government Engineering College, Hassan, Karnataka, India*
E-mail: thirthegowda.gech@gmail.com
[3]*Department of Computer Science and Engineering, Government Engineering College, KR.Pet, Mandya, Karnataka, India*
E-mail: shivashankar.research@gmail.com
[4]*Department of Computer Science and Engineering, Mysore College of Engineering and Management, Mysore, Karnataka, India*
E-mail: ravipbympr@gmail.com
**Corresponding Author*

is an old recognition algorithm, the mean shift [6]. The main limitation of mean shift algorithm is that the window size is petty, which can influence the modes to become merged. It performs well under stable and complex conditions. But, it yields poor results if the environment changes suddenly. The mean shift method was revised to tackle with vigorously varying color distributions derived from the frame sequences of video. The revised method called Continuously Adaptive Mean Shift (CAMSHIFT) [7] overcomes the problems posed by mean shift and can be used for tracking a moving person face. Based on CAMSHIFT, Ranganatha S et al. [8], [9] have proposed novel approaches for detection and tracking of human face(s) in video successions. The proposed approaches tackle various technical challenges like occlusion, pose, expression, illumination etc. Huang and Lin [10] have proposed a system called multi-CAMSHIFT to detect and track multi-view faces. The proposed system uses color and shape characteristics for solving the tracking problems. It utilizes the tracker to get the contender portions by delineating the concerned probability distribution. The performance of the proposed system is further enhanced by means of a multi-resolution framework. The PCA and SVM (Support Vector Machine) are fused to get a module in order to detect and recognize multi-view faces. The proposed system is computationally efficient; capable of working in complex background and perform tracking in real-time. The experimental results show that the proposed system can be applied for tracking and recognition of multi-view faces. It can also be applied to surveillance and face guard systems.

Nijad Al-Najdawi et al. [11] have developed an automated novel algorithm for tracking the objects in real-time security systems. The proposed algorithm used an extended version of KLT algorithm [12], [13] for feature extraction and tracking. The work in literature [14] explore detection and tracking of faces using S-PCA & KLT method. This work also addressed the problem of hand tracking in high-speed videos. Shi and Tomasi [15] have proposed a feature choosing method, as well as a method for detecting occlusions, disocclusions and irrelevant features. These methods are part of a new tracking method, which prolongs the former Newton-Raphson type of searching method. It is capable of performing under affine transformation. The proposed method was verified by conducting experiments on actual images and a number of simulations. Furthermore, there are several works in literature [16], [17], [18], [19], [20], [21] for detection and tracking of human face(s) that are based on KLT. The works just quoted are evaluated using customary, novel and standard performance measurement metrics.

M. Kim et al. [22] have addressed the problem of tracking and recognition of faces in real time. The authors explained their work with the aid of video successions containing the noises. In this method, the tracker adaptively builds a target model by reacting to the changes in the appearance, which is typical to the settings in the video. In order to fit the cross-center-biased properties in real world video sequences, Cheung and Po [23] have proposed a novel method called cross diamond search algorithm. Maciej Smiatacz [24] demonstrated the exhaustive tests of linear transformations; the methods are Eigenfaces, Fisherfaces, Laplacianfaces and Margin faces. As per the work published in literature [25], Fisherface algorithm do better than Eigenface does. Further, based on the parameters in normal and noisy environments, Ravi Kumar et al. [26] have combined three algorithms to compare the performance of their proposed tracking method. The literature presented till now helped to develop the method that is proposed in this paper.

3. Methodology

In this part of the paper, we will depict the working of the incorporated calculations alongside how could we accomplished our ultimate objective. Let us discover how the blend of two calculation will push us to effectively distinguish a face and track the face all through the video until the face is available inside the extent of the window size of the video player.

3.1 FACE DETECTION

Highlighting the location of a face in the video sequence is the most significant advancement of the proposed work. Once after the calculation begins running in matlab, it get to the webcam of the PC and a window gets open normally called as video player which will have different alternatives. The video player holds up till the face is identified in any portion of the casing. When a face is distinguished, our first calculation sets up the track to focus on Region of Interest (ROI) taken from the face. Utilizing Linear Discriminant Analysis (LDA) technique, the dimensionality in the network is lessened. This is the way the track focuses are set up on the face. After this, a rectangular bbox (bounding box) is set up around the face and a check is given to it. The aforementioned process is repeated until the player is closed.

ALGORITHM 1. Detection of face

1. Initialize the video device (camera, input video source).
2. Initialize the face detector, and KLT object tracker.
3. Get the frame and find frame-size information. Start a video player instance.
4. Iterate the frames till the face is detected successfully and loop until the player is closed.

5. Display bounding boxes and tracked points. Repeat until the process halts.

3.2 Tracking the Detected Face

For this situation, we have utilized the calculation of KLT. We have characterized a tracker class in this calculation. The tracker portion of the calculation will get established due to the points from the calculation. It will check the bounding box esteem and the followed focuses in the event that the worth matches, at that point it will erase the bounding box and supplant the bouncing box. It will get the bounding box from each face in the followed focuses.

When the bounding box is set up around the face or the face's followed points, KLT will keep the tally of that case. Until the face is available inside the extent of the camera, the course face detector will keep the followed points unblemished; however, once after the face leaves the extent of the camera, the bounding box will get erased which is the inbuilt usefulness of the KLT tracker class. Henceforth the tracker class of the KLT will be holding up till the following arrangement of track points shows up to it. Getting the bounding box points from Algorithm 1 is an important step for the tracker class to begin its function.

ALGORITHM 2. Tracking of a detected face

1. AddDetections(tracker, I, bboxes)

   ('tracker' is the MultiObjectTrackerKLT object, 'I' is the current frame, and bboxes is an M-by-4 array of [x, y, w, h] bounding boxes).

2. Check if the detection belongs to one of the existing faces.
3. If the box is matched, then release the box and replace it with new box.
4. Re-detect the points then.
5. Finally, determine the faces that are no longer tracked.

3.3 PYRAMIDAL LUCAS-KANADE FEATURE TRACKER

It is one of the most powerful algorithms in case of optical flow [2] that is used for feature tracking. It provides enough accuracy and robustness to carry out the intended work. Considering a point on the first image I as u = (ux, uy), the purpose of feature tracking is to find v = u + d in the next image J, then I(u) and J(v) are very much similar to each other. Here, d is the displacement vector at velocity x which can also be called as optical flow at x. It is very much needed to define the similarity in notion in a 2D neighborhood sense. Let ωx and ωy are two integers. Then the residual function is minimized by d vector

$$\in (d) = \in (d_x, d_y)$$

where,

$$\in (d_x, d_y) = \sum_{x=u_x-w_x}^{u_x+w_x} \sum_{y=u_y-w_y}^{u_y+w_y} \Big(I(x,y) - J\big(x+d_x, y+d_y\big)\Big)^2$$

We can see how the similarity function is measured on the basis of an image neighborhood size (2ωx + 1) × (2ωy +1). This neighborhood is called by the name of integration window. Functional and Performance Requirements:

The algorithm is designed in such a way that it meets the expectations of the practical feature tracking algorithm. Requirements considered include,

a) The algorithm should be accurate,

b) It should be robust and

c) Computationally it must be inexpensive.

4. Results and Analysis

A study is needed to reflect the test results in an organized manner. It will likewise offer us a chance to break down the work did. It's a record which will have all the information acquired from the test in a much-sorted manner. It will also tell the distinctive working conditions in which the calculation is tried, and alongside that it will likewise give us a portrayal of the environment where the study was led.

Table 1 shows the results which are tabulated from the experiment conducted. On an average, the algorithm will detect single or multiple faces within a time span of 15 seconds. The proposed algorithm is tested for a variety of conditions; overall, we saw it giving best results for all of the test conditions.

Let us consider the study more effectively by viewing the snapshots.

4.1 Face Position #1

Fig.1 shows the detection of a face when it is facing the camera at 0^0 angle. Here, the proposed algorithm took 8 seconds for detection. Eight seconds is more; usually the algorithm takes less than or equal to 2-3 seconds. The snapshot in Fig.1 was taken in a room having good amount of light. A live video sequence was generated from the camera of the computational device, in our case the laptop.

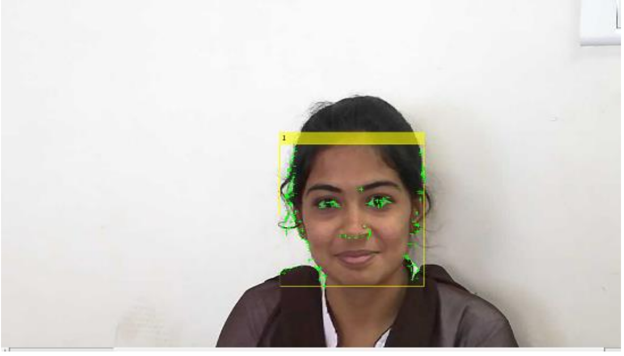

Fig.1. Face detection while facing the camera directly.

Table.1. Summary of the Experimental Results.

No. of Faces	No. of Frames	Time (Sec)	Face Detected	Face Position	Partial Occlusion
1	6	8	1	Front	No
1	18	2	1	Left	No
1	83	55	1	Right	No
1	2	9	1	Left	Yes
1	4	8	1	Right	Yes
2	8	11	2	Front	No
2	4, 200	11, 21	2	Front, Left	No
2	4, 148	12, 18	2	Front, Right	No
2	4, 210	6, 25	2	Front, Left	Yes
2	220, 6	27, 8	2	Left, Front	Yes
3	6	13	3	Front	No
3	6, 341, 735	9, 21, 36	3	Front	Yes
4	6	9	4	Front	No
4	3(2 persons), 177, 222	9 (2 persons), 16, 20	4	Front	Yes
5	9	3	5	Front	No
5	3(2 persons), 400, 600, 620	9 (2 persons), 12, 24, 35	5	Front	Yes

4.2 FACE POSITION #2

Fig.2 shows the detection of a face in the normal environment condition like light intensity. The best part of this detection is that the face is partly covered with a cloth. Even after facing the problem of occlusion, proposed algorithm detects the face successfully and further tracks it in real time. In this case, the time taken to detect the face is little higher as the number of frames were many, and the range is huge.

Fig.2. Face detection with occlusion at an angle of 0^0.

4.3 FACE POSITION #3

Fig.3 is a case where the face is turned left approximately at an angle of 90^0 to the camera. Even with the constraint of angle, the face was successfully detected in less than 8 seconds. This is the case wherein which it didn't include any sort of occlusion.

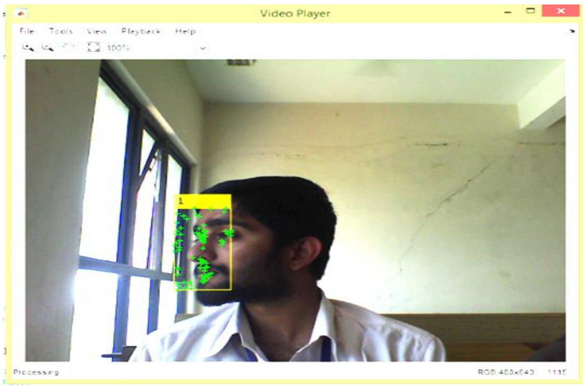

Fig.3. Face detection with no occlusion and with an angle of 90^0.

4.4 FACE POSITION #4

Fig.4 is similar to Fig.3 with a difference in the angle facing the camera. The face in Fig.4 is facing the camera at an angle of 270^0.

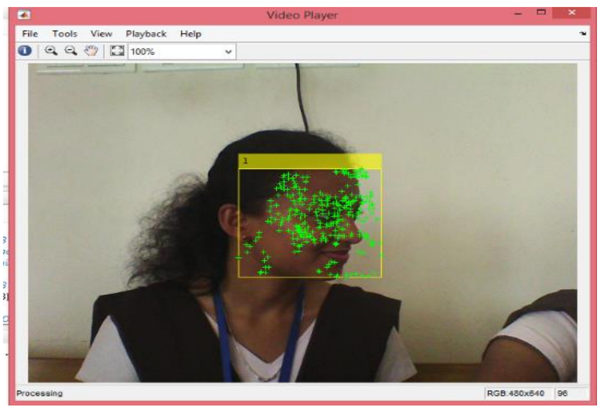

Fig.4. Face detection with no occlusion and with an angle of 270^0 .

4.5 FACE POSITION #5

There is some similarity between Fig.5 and Fig.1; however, two faces were introduced to webcam in Fig.5 instead of 1 as in Fig.1.

The faces were detected successfully as the room’s light condition was normal and no occlusion. The faces were at an angle of 0^0 and were detected and tracked until the track points were within the scope of the camera's vision.

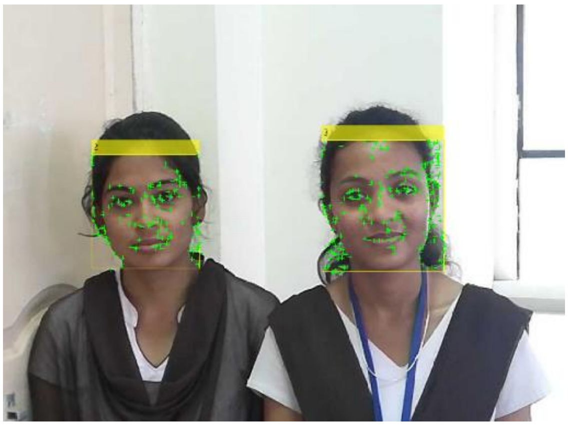

Fig.5. Face detection with no occlusion and with an angle of 0^0.

4.6 FACE POSITION #6

In Fig.6, we can see that the face is half covered with a cloth which can be termed as partial occlusion. Even under partial occlusion, proposed algorithm has successfully detected the faces in minimum time with the faces facing the camera at an angle of 0^0.

Fig.6. Face detection with occlusion and with an angle of 0^0.

4.7 FACE POSITION #7

Fig.7 and Fig.3 are similar in nature except that Fig.7 houses two faces. The faces were detected and tracked successfully in a very short period of time; here, the faces faced the challenge posed by light intensity.

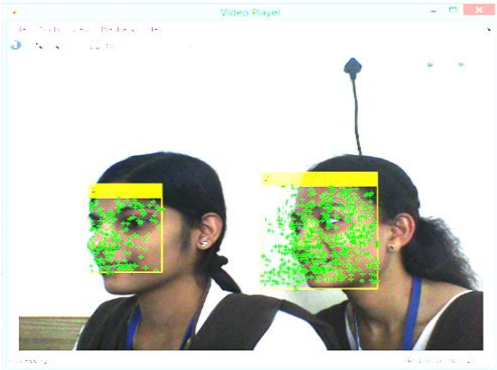

Fig.7. Face detection with no occlusion and with an angle of 90^0.

4.8 FACE POSITION #8

Fig.8 is a case wherein the distance between the faces is kept varied. Proposed algorithm detected and tracked all the faces that are facing the camera at an angle of 0^0, and are at different places.

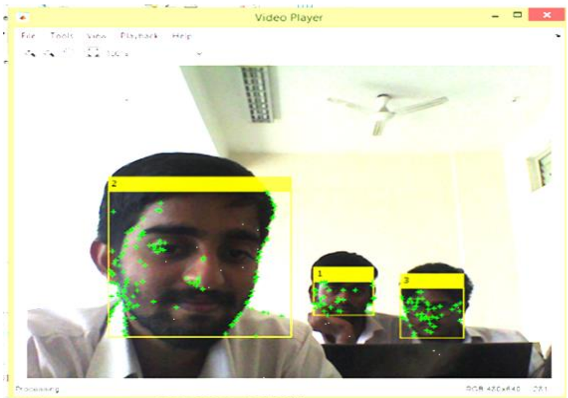

Fig.8. Face detection with no occlusion and with an angle of 0^0.

4.9 FACE POSITION #9

Fig.9 comprises detection and tracking of three faces when partial occlusion is on play. In this case, small number of frames were used, and the proposed algorithm took minimum time to accomplish the intended work.

Fig.9. Face detection with occlusion and with an angle of 0^0.

4.10 FACE POSITION #10 & #11

In Fig.10 and Fig.11, three faces are facing the camera at an angle of 90^0 and 270^0 respectively, and all the three faces were successfully detected and tracked within less time and with normal light intensity. Occlusion was not part of both the figures as can be seen.

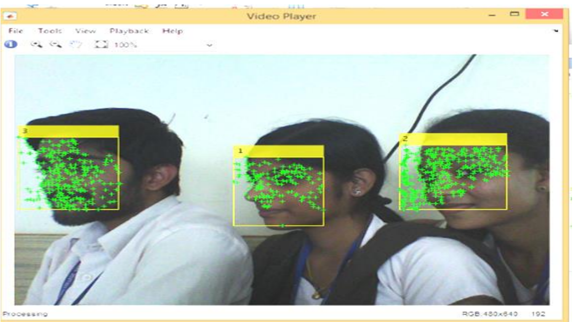

Fig.10. Face detection with no occlusion and an angle of 90^0.

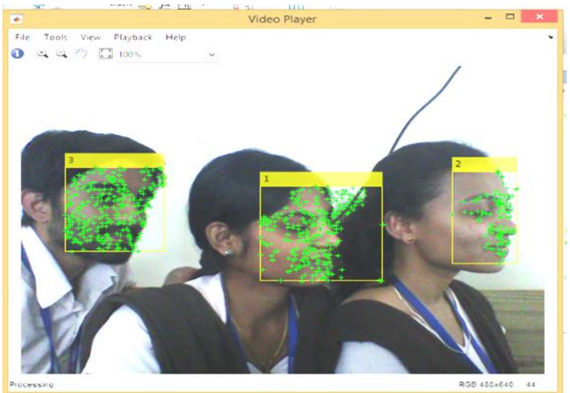

Fig.11. Face detection with no occlusion and an angle of 270^0.

4.11 FACE POSITION #12

In Fig.12, one can observe that the algorithm has successfully detected all the four faces from the live video sequence and faces are facing the camera at an angle 0^0. Environmental condition of the place where the test was conducted was normal. The algorithm didn't take much time to detect the face in the sequence of video.

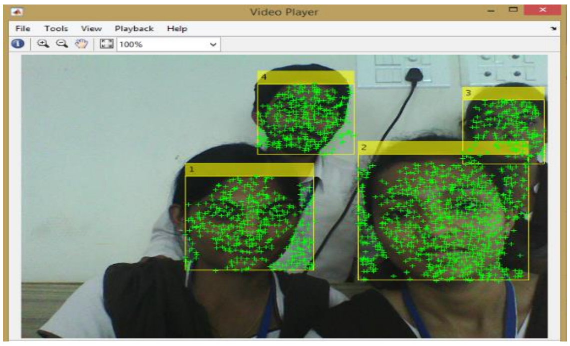

Fig.12. Face detection with no occlusion and with an angle of 0^0.

4.12 FACE POSITION #13

Fig.13 is an example for detection and tracking of faces under the problem of occlusion. The proposed algorithm has detected and tracked multiple faces, and ignored the odds present in the live video sequence.

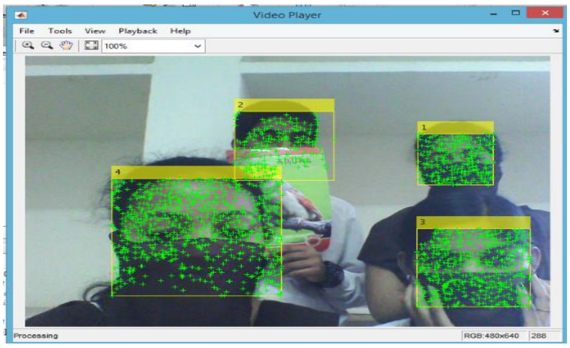

Fig.13. Face detection with occlusion and with an angle of 0^0.

4.13 FACE POSITION #14 & #15

In Fig.14 and Fig.15, faces are facing the camera at an angle of 90^0 and 270^0 respectively, and the algorithm has succeeded in the detection and tracking of all the four faces present in the live video sequence.

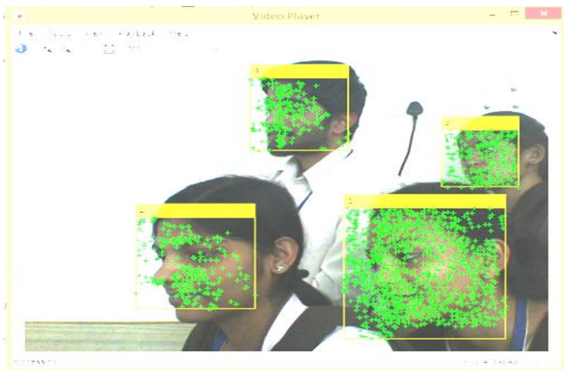

Fig.14. Face detection with no occlusion and an angle of 90^0.

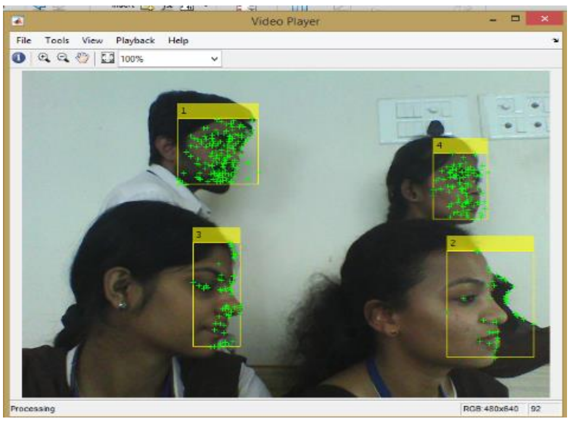

Fig.15. Face detection with no occlusion and an angle of 270^0.

4.14 FACE POSITION #16 & #17

Fig.16 and Fig.17 is a case wherein five faces are introduced into the algorithm. In Fig.16, faces are introduced without occlusion; whereas in Fig.17, faces are introduced with occlusion. As can be seen in figures, proposed algorithm has detected and tracked all the five faces in real time.

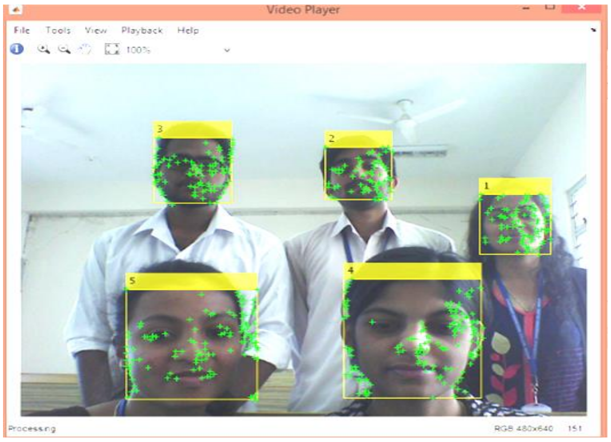

Fig.16. Face detection with no occlusion and with an angle of 0^0.

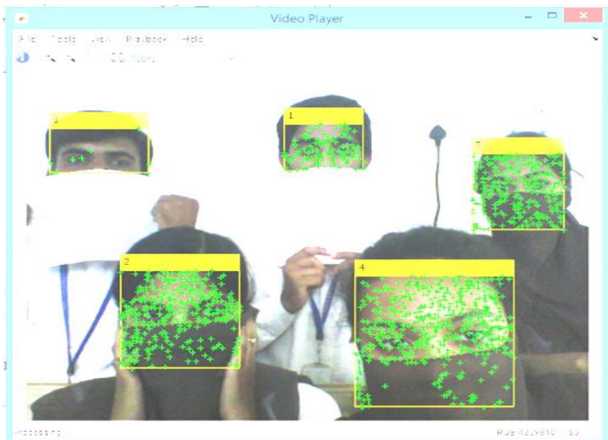

Fig.17. Face detection with occlusion and with an angle of 0^0.

Multiple faces were detected at the same instance. This shows how efficiently the algorithm works when it is subjected to different test conditions. Along with this the algorithm was also tested at different light intensities. Even then the algorithm worked efficiently with no odds in the end result.

Table 2 shows the results which are tabulated from the experiments conducted and tested on the proposed system with other existing algorithms. On an average, the proposed algorithm will detect single or multiple faces within a time span of 15 seconds, optical flow will detect faces within 17 seconds, CAMSHIFT and KLT will detect faces between 19-20 seconds. The comparison results of the algorithms is as shown in Fig.18.

Table.2. Summary of the Results comparison.

Frame number	Proposed	Optical Flow	CAMSHIFT	KLT
6	1	1	1	1
18	1	1	1	1
83	1	1	0	1
2	1	1	1	1
4	1	1	0	0
8	2	2	2	2
4, 200	2	2	1	1
4, 148	2	2	2	2
4, 210	2	2	2	2
220, 6	2	1	1	2
6	3	3	2	2
6, 341, 735	3	3	2	3
6	4	3	3	3
3 , 177, 222	4	3	3	3
9	5	5	5	5
3, 400, 600, 620	5	5	5	5

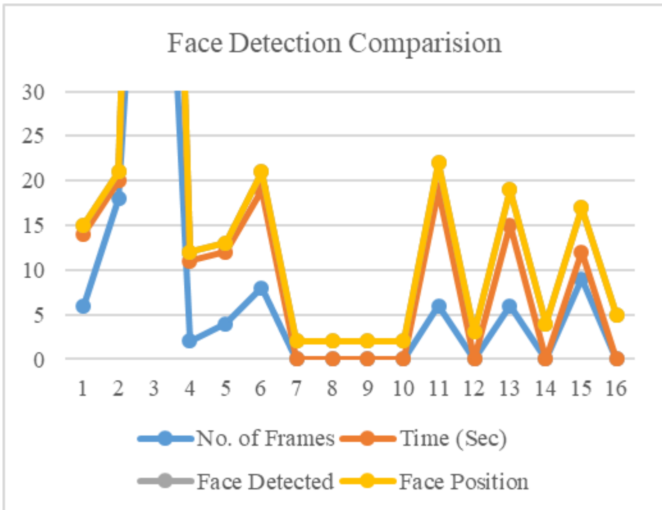

Fig.18. Face detection experimental results comparison.

Table 3 shows accuracy results of all the four algorithms. During experimentation, more than 750 frames were randomly selected for each algorithm, and more than 2100 faces were detected by each algorithm. The time taken by the algorithms is also depicted in the table. Optical flow achieves better results than KLT and CAMSHIFT; proposed algorithm performs better than optical flow, and achieves 95% accuracy. Fig.19 comprises the comparison of accuracy results on different parameters.

Table.3. Comparison of different algorithms accuracy results.

Method	Num. of Frames Tested	Time in Sec	Total num. of Actual Faces	Num. of Faces Detected	Accuracy (in %)
Optical Flow	789	332	2470	2300	93.1
CAMSHIFT	810	350	2359	2130	90.3
KLT	775	310	2374	2172	91.5
Proposed	805	325	2553	2431	95.2

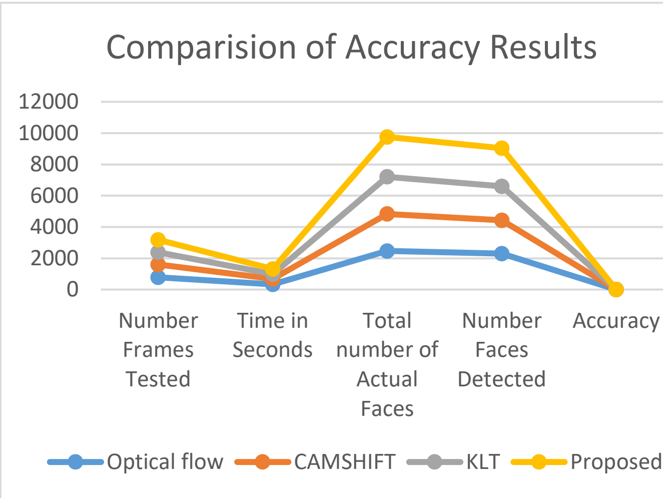

Fig.19. Comparison results of different algorithms with different parameters.

Table 4 shows the efficiency analysis of algorithms with different parameters. As per results, it is evident that the consistency of proposed algorithm for face detection in real-time video sequences is definitely better compared to optical flow, CAMSHIFT and KLT approaches.

Table.4. Performance analysis of the proposed model.

Method	Accuracy (in %)	F1 Score (in %)	Precision (in %)	Recall (in %)
Optical flow	93.10	92.90	92	92.10
CAMSHIFT	90.30	90	88	89.20
KLT	91.50	90.70	89.10	89.10
Proposed	95.20	94.70	94	94.10

The results of the several algorithms utilizing different measures are shown in Figure 20. Accuracy is one of the most widely used criteria to evaluate efficiency. The accuracy of the proposed method's confusion matrix is shown in Fig.21 below. The confusion matrix for the accuracy of KLT, CAMTSHIFT, and Optical Flow algorithms are shown in Fig.22, Fig.23, and Fig.24 respectively.

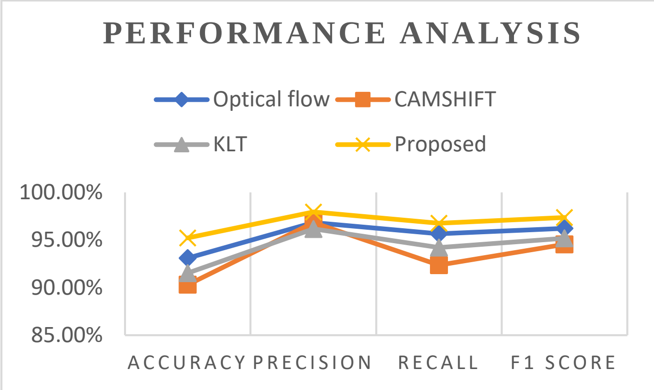

Fig.20. performance efficiency analysis of various methods with different parameters.

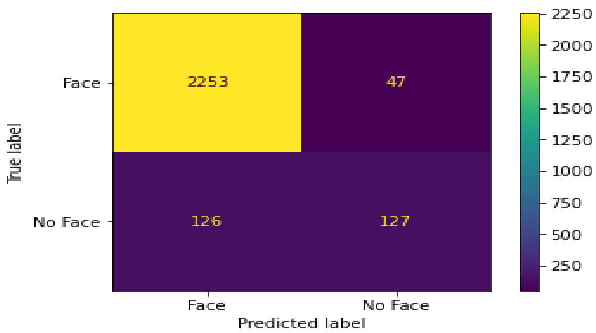

Fig.21. confusion matrix to assess the suggested method accuracy.

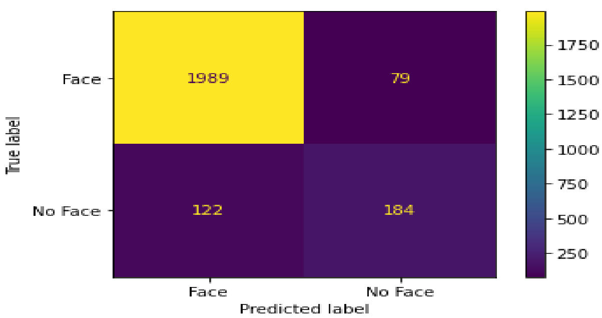

Fig.22. confusion matrix to assess the KLT method accuracy.

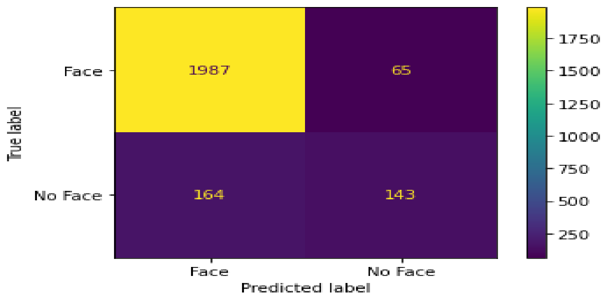

Fig.23. confusion matrix to evaluate the accuracy of the CAMSHIFT technique.

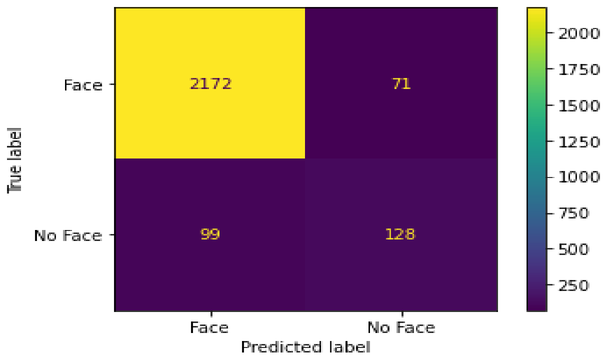

Fig.24. confusion matrix to evaluate the accuracy of the Optical Flow technique.

5. Conclusion and Future Work

The task on continuous face discovery and following had allowed us a chance to learn about numerous famous strategies utilized in the field of video analysis. In addition to this, we realized that the precision of the framework will improve extraordinarily with the consolidation of at least two procedures. The calculation is hearty, invariant to commotion and has been exhibited with the assistance of MATLAB as it is a logical programming language that gives solid scientific and numerical help for the execution of the propelled calculation.

In future, we may fuse the idea of IOT and can apply the idea of AI to accomplish face acknowledgment. Taking numerous contributions from distinctive vision frameworks, we can accomplish more significant standards of acknowledgment.

References

[1] Ranganatha S, Dr. Y P Gowramma, "Face Recognition Techniques: A Survey", *International Journal for Research in Applied Science and Engineering Technology*, Vol.3, Issue 4, pp.630-635, 2015.

[2] Ranganatha S, Y P Gowramma, "A Comprehensive Survey of Algorithms for Face Tracking in Different Background Video Sequence", *International Journal of Computer Applications*, Vol.181, Issue 27, pp.43-49, 2018. DOI: 10.5120/ijca2018918134

[3] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", in *Procedings. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Kauai, USA, Vol.1, pp.511-518, December 2001. DOI: 10.1109/CVPR.2001.990517

[4] Jatin Chatrath, Pankaj Gupta, Puneet Ahuja, Aryan Goel and Shaifali M. Arora, "Real Time Human Face Detection and Tracking", in *Procedings of IEEE International Conference on Signal Processing and Integrated Networks*, Noida, India, pp.705-710, February 2014. DOI: 10.1109/SPIN.2014.6777046

[5] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach toward Feature Space Analysis", in *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, Vol.24, No.5, pp.603-619, May 2002. DOI: 10.1109/34.1000236

[6] K. Fukunaga and L. D. Hostetler, "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition", in *IEEE Transations on Information Theory*, Vol.21, No.1, pp.32-40, January 1975. DOI: 10.1109/TIT.1975.1055330

[7] G. Bradski, "Computer Vision Face Tracking for Use in a Perceptual User Interface", *Intel Technology Journal*, pp.12-21, 1998.

[8] Ranganatha S and Y P Gowramma, "An Integrated Robust Approach for Fast Face Tracking in Noisy Real-World Videos with Visual Constraints", in *Proc. of IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp.772-776, September 2017. DOI: 10.1109/ICACCI.2017.8125935

[9] Ranganatha S and Y P Gowramma, "Color Based New Algorithm for Detection and Single/Multiple Person Face Tracking in Different Background Video Sequence", *International Journal of Information Technology and Computer Science (IJITCS)*, Vol.10, No.11, pp.39-48, November 2018. DOI: 10.5815/ijitcs.2018.11.04

[10] Han-Pang Huang and Chun-Ting Lin, "Multi-CAMSHIFT for Multi-View Faces Tracking and Recognition", in *Procedings of IEEE International Conference on Robotics and Biomimetics*, Kunming, China, pp.1334-1339, December 2006. DOI: 10.1109/ROBIO.2006.340122

[11] N. Al-Najdawi, S. Tedmori, E. A. Edirisinghe and H. E. Bez, "An Automated Real-Time People Tracking System Based on KLT Features Detection", *International Arab Journal of Information Technology*, Vol.9, No.1, pp.100-107, 2012.

[12] Bruce D. Lucas and Takeo Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", in *Procedings of International Joint Conference on Artificial Intelligence*, Vancouver, BC, Canada, Vol.2, pp.674-679, August 1981.

[13] Carlo Tomasi and Takeo Kanade, "Detection and Tracking of Point Features", *Carnegie Mellon University Technical Report* CMU-CS-91-132, April 1991.

[14] J. Savitha, Dr. A.V. Senthil Kumar, "Face Tracking and Detection using S-PCA & KLT Method", *International Journal of Advance Research in Computer Science and Management Studies*", Vol.2, No.2, pp.224-229, February 2014. ISSN: 2321-7782

[15] Jianbo Shi and Carlo Tomasi, "Good Features to Track", in *Procedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp.593-600, June 1994. DOI: 10.1109/CVPR.1994.323794

[16] Ranganatha S and Y P Gowramma, "A Novel Fused Algorithm for Human Face Tracking in Video Sequences", in *Procedings of IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, pp.1-6, October 2016. DOI: 10.1109/CSITSS.2016.7779430

[17] Ranganatha S and Y P Gowramma, "Development of Robust Multiple Face Tracking Algorithm and Novel Performance Evaluation Metrics for Different Background Video Sequences", *International Journal of Intelligent Systems and Applications (IJISA)*, Vol.10, No.8, pp.19-35, August 2018. DOI: 10.5815/ijisa.2018.08.03

[18] Ranganatha S and Y P Gowramma, "Eigen and HOG Features based Algorithm for Human Face Tracking in Different Background Challenging Video Sequences", *International Journal of Image, Graphics and Signal Processing (IJIGSP)*, Vol.14, No.4, pp. 70-83, 2022. DOI: 10.5815/ijigsp.2022.04.06

[19] Ranganatha S and Y P Gowramma, "Image Training, Corner and FAST Features based Algorithm for Face Tracking in Low Resolution Different Background Challenging Video Sequences", *International Journal of Image, Graphics and Signal Processing (IJIGSP)*, Vol.10, No.8, pp.39-53, August 2018. DOI: 10.5815/ijigsp.2018.08.05

[20] Ranganatha S and Y P Gowramma, "Image Training and LBPH Based Algorithm for Face Tracking in Different Background Video Sequence", *International Journal of Computer Sciences and Engineering (IJCSE)*, Vol.6, No.9, pp.349-354, September 2018. CrossRef-DOI: https://doi.org/10.26438/ijcse/v6i9.349354

[21] Ranganatha S and Y P Gowramma, "Selected Single Face Tracking in Technically Challenging Different Background Video Sequences Using Combined Features", *ICTACT Journal on Image and Video Processing (JIVP)*, Vol.9, No.2, pp.1911-1918, November 2018. DOI: 10.21917/ijivp.2018.0271

[22] Minyoung Kim, Sanjiv Kumar, Vladimir Pavlovic and Henry Rowley, "Face Tracking and Recognition with Visual Constraints in Real-World Videos", in *Procedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1-8, June 2008. DOI: 10.1109/CVPR.2008.4587572

[23] Chun-Ho Cheung and Lai-Man Po, "A Novel Cross-Diamond Search Algorithm for Fast Block Motion Estimation", in *IEEE Transactions. on Circuits and Systems for Video Technology*, Vol.12, No.12, pp.1168-1177, December 2002. DOI: 10.1109/TCSVT.2002.806815.

[24] Maciej Smiatacz, "Eigenfaces, Fisherfaces, Laplacianfaces, Marginfaces - How to Face the Face Verification Task", In: Burduk R., Jackowski K., Kurzynski M., Wozniak M., Zolnierek A. (eds) *Proc. of the 8th International Conference on Computer Recognition Systems CORES 2013. Advances in Intelligent Systems and Computing*, Vol. 226. Springer, Heidelberg. https://doi.org/10.1007/978-3-319-00969-8_18

[25] Delpiah Wahyuningsih, Chandra Kirana, Rahmat Sulaiman, Hamidah, Triwanto, "Comparison of the Performance of Eigenface and Fisherface Algorithm in the Face Recognition Process", in *Proc. of IEEE International Conference on Cyber and IT Service Management*, pp.1-5, November 2019. DOI: 10.1109/CITSM47753.2019.8965345

[26] Ravi Kumar Jatoth, Sanjana Gopisetty, Moiz Hussain, "Performance Analysis of Alpha Beta Filter, Kalman Filter and Meanshift for Object Tracking in Video Sequences", *International Journal of Image, Graphics and Signal Processing (IJIGSP)*, Vol.7, No.3, pp.24-30, February 2015. DOI: 10.5815/ijigsp.2015.03.04