

RESEARCH ARTICLE



OPEN ACCESS

Received: 28.05.2021

Accepted: 04.07.2021

Published: 21.07.2021

Citation: Nanthini N, Puviarasan N, Aruna P (2021) An Efficient Velocity Estimation Approach for Face Liveness Detection using Sparse Optical Flow Technique. Indian Journal of Science and Technology 14(25): 2128-2136. <https://doi.org/10.17485/IJST/v14i25.942>

* **Corresponding author.**

nanthini0294@gmail.com

Funding: None

Competing Interests: None

Copyright: © 2021 Nanthini et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Published By Indian Society for Education and Environment ([iSee](#))

ISSN

Print: 0974-6846

Electronic: 0974-5645

An Efficient Velocity Estimation Approach for Face Liveness Detection using Sparse Optical Flow Technique

N Nanthini^{1*}, N Puviarasan², P Aruna³

1 Research scholar, Department of CSE, Annamalai University, Chidambaram, India

2 Professor, Department of CIS, Annamalai University, Chidambaram, India

3 Professor, Department of CSE, Annamalai University, Chidambaram, India

Abstract

Objectives: To propose a new liveness detection algorithm using optical flow to ensure the presence of actual live face into a photograph or 2D masks in face recognition biometric security systems. **Methods:** This work proposes an anti-spoofing model namely Sparse Optical Flow Technique with Velocity Estimation Approach (SOFT_VEA). Optical flow is an effective method for tracking objects in motion. It is adapted in this work to capture facial movements and decide the liveness state. The proposed algorithm considers real faces and two kinds of photo imposters. **Findings:** From the input video, the motion information of specific facial landmark points is captured by an optical flow algorithm. Then, the velocity of those landmark points is estimated via Euclidean distance. Based on this calculated velocity, the fake face is discriminated from the real face using a threshold value. The Empirical study shows that the proposed face liveness detection model is effective with an accuracy of 88% and Half Total Error Rate (HTER) of 2.45. **Novelty:** The proposed work is based on real face and photo imposters. The liveness detection algorithm is developed with a novel velocity estimation approach. It is very helpful for biometric security systems.

Keywords: Biometric security system; Liveness detection; Antispoofing; Facial landmarks; Optical flow; Euclidean distance

1 Introduction

Biometric systems for security have developed rapidly in recent years. Every biometric system has to uniquely identify the individual person's identity based on their psychological or behavioural features. Compared with biometric systems like fingerprint, iris, and voice recognition, face recognition is more convenient and effective for users.

Face recognition is a low-cost and highly used biometric system because it requires a simple hardware device like an optical camera and a low computational algorithm⁽¹⁾. These makes face recognition a perfect solution for embedded and mobile devices. But face recognition system fails to differentiate whether the person's face is a 'live' face or a 'not live' face. So, the face recognition system needs 'liveness detection' to protect the system against spoofing. Nowadays it is very easy to spoof a person's face using a photograph or 2D masks⁽²⁾. To find the spoofed faces various methods are implemented

using life signs, textures, motions, and so on. This work focuses on the novel algorithm for detecting facial movements with landmark points and proposed velocity calculation.

Liveness detection on face recognition is performed based on facial movements⁽³⁾. The input is given in video format i.e. sequence of images, in which the system can easily identify the face regions and recognize their movements such as head, eye, and lip movements to find the fake or spoofed faces. The Optical flow field is one of the trajectory algorithm used to track objects in motion⁽⁴⁾. It is done with two approaches, one is dense and the other is sparse. If all the points of the target are considered for tracking, then it is known as optical flow with the dense model. When selected numbers of key points are used to track the target object, then it is known by the sparse optical flow method. Shin et.al, proposed a non-prior training active feature model (NPT-AFM) for tracking objects or persons by extracting feature points for restoring the occlusion of motion information⁽⁵⁾.

In⁽⁶⁾ proposed optical flow of lines to evaluate the motion of the human face for liveness detection. Here, they proposed a quick method for face center detection to classify the typical sources of errors like glasses and facial hair and achieve an error rate of 1%. In⁽⁷⁾ introduced optical flow to find the motion vectors used for recognizing emotions. They used Gautama and VanHulle proposed optical flow algorithm for finding the efficient motion vectors. In⁽⁸⁾ proposed optical flow of lines to evaluate the motion of the human face for liveness detection. The system utilizes a model-based local Gabor decomposition and SVM experts to select the points to form lines. They found the motion depth of 3D face structure varies from 2D face masks. In⁽⁹⁾ experimented using an optical flow algorithm to differentiate the 3D face and 2D face masks. They used optical flow field calculations to distinguish the faces with respect to the threshold values for degree of change. In⁽¹⁰⁾ proposed an optical flow algorithm for debris position and motion detection in image sequences. The parameters used in the method detect in low computational cost than the previous methods. In⁽¹¹⁾ conducted a challenge and response method for real-time liveness detection for security systems. In⁽¹²⁾ proposed an eye and mouth movement combined technique for procuring maximum reliability during face liveness detection. In⁽¹³⁾ proposed a hierarchical neural network, which can fuse image quality cues and motion cues for liveness detection. In⁽¹⁴⁾ proposed an efficient algorithm for Movement Estimation and object tracking in video scenes using Optical Flow and Gabor Features Based Contour Model. In⁽¹⁵⁾ delivered a Video processing system that tracks the gauzes on the video captured by the endoscope. In which, the system tracks the deviate in the size of the gauzes along with challenging illuminations, and in the darker background.

In this work, a sparse optical flow-based liveness detection model is proposed. The proposed method analyses the difference in characteristics of real faces and photograph or 2D masks by detecting facial landmarks for each and every person and calculates the velocity for its optical flow field using an efficient velocity estimation approach.

2 Proposed Work

The proposed face liveness detection model using SOFT_VEA consists of three phases. In the first phase, Facial Landmark Points are marked on the input video frames. Next, the points of interest such as eyes, nose, and mouth points are tracked using the Optical flow algorithm with a sparse approach to finding the motion between consecutive frames in line. Finally, an efficient Velocity estimation approach is utilized to find the impact of sudden and gradual changes in the velocity of landmark points. The impact is measured and using a threshold factor, the fake faces are identified against the real ones. The Proposed block diagram is shown in Figure 1.

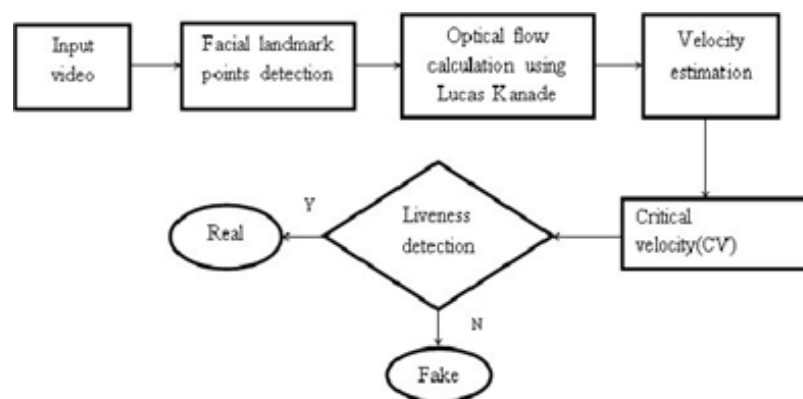


Fig 1. Proposed block diagram of SOFT_VEA based liveness detection model

2.1 Facial Landmark points

For the trajectory operation, video input is converted to a sequence of images and 68 facial landmarks are detected for every frame. To find the facial landmarks, first, the system needs to locate the human face from the input and put a rectangle to it. By using the rectangle coordinates, face landmark points are detected. Figure 2 shows the facial landmark points. Here, from 1 to 68 facial points are detected including the key features like nose, eyes, eyebrows, and mouth.



Fig 2. Facial landmark points

2.2 Sparse Optical Flow Technique

Optical flow is to use the time-domain change and correlation of pixel intensity in image sequences to determine the ‘movement’ of each pixel’s location, that is, to study the relationship between the intensity change by time and the object’s structure and movement in the scene. Optical flow can be found by using two types, sparse optical flow, and dense optical flow. Sparse optical flow selects a set of feature points or pixels such as edges and corners to track its motion. While dense optical flow takes every pixel in each frame to track the motion.

Optical flow for the face is irregular with head motion and irregular facial expressions. The relative motion between the camera and object is of 4 types: rotation, transition, moving front and back, swing. All the other kinds of motions are the combination of these four motions. The representation of each motion varies with varying actions. The optical flow for rotation, transition, and moving front and back is similar for both real face and 2D masks. But the optical flow for swing (i.e. shaking, lowering, and raising head) is different for real and 2D faces. Figure 3 shows the optical flow for swing motion.

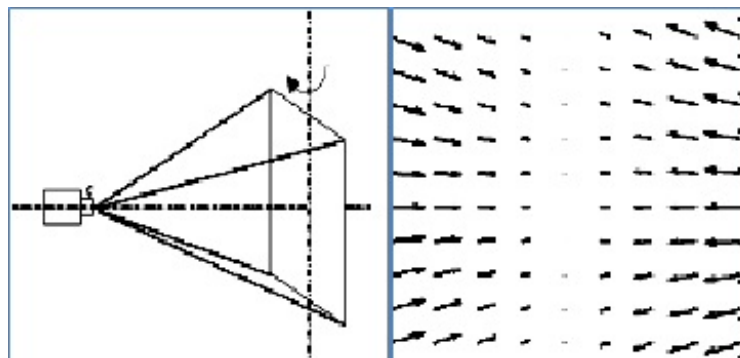


Fig 3. Optical flow generated by the swing motion

Let us assume that, pixel intensities are constant for the optical flow,

$$F(x, y, t) = F(x + \partial x, y + \partial y, t + \partial t) \quad (1)$$

Taylor series is used to expand the RHS of the above equation (1). By eliminating the common terms, optical flow equation is obtained as,

$$\frac{\partial F}{\partial x}a + \frac{\partial F}{\partial y}b + \frac{\partial F}{\partial t} = 0 \quad (2)$$

where a and b are unknown variables.

Solving one equation with two unknown variables cannot be done directly. So Lucas-Kanade method is introduced to solve this problem.

Lucas-Kanade Algorithm:

Lucas-Kanade method is a kind of sparse optical flow that used a corner detection algorithm to select the corner pixels. Instead of corner pixels, the landmark values are given to the Lucas-Kanade algorithm to find the flow of human facial landmarks. The selected landmark values are then passed to the optical flow functions. Assume that, all the neighbouring pixels to the landmark have similar motions. Consider the LK method taken a 3x3 patch from an image that contains 9-pixel values. So 9 equations need to be solved to determine the two unknown variables a and b.

$$\begin{aligned} \partial x(p_1)a + \partial y(p_1)b &= -\partial t(p_1) \\ \partial x(p_2)a + \partial y(p_2)b &= -\partial t(p_2) \\ &\vdots \\ \partial x(p_9)a + \partial y(p_9)b &= -\partial t(p_9) \end{aligned} \quad (3)$$

To solve the above 9 equations, least square fitting needs to be applied. After applying least square fitting algorithm, the equation looks like two equation with two unknowns.

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_i \partial x(p_i)^2 & \sum_i \partial x(p_i) \partial y(p_i) \\ \sum_i \partial y(p_i) \partial x(p_i) & \sum_i \partial y(p_i)^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_i \partial x(p_i) \partial t(p_i) \\ -\sum_i \partial y(p_i) \partial t(p_i) \end{bmatrix} \quad (4)$$

The above equation gives the movement of x and y over time t. Finally, the unknown variables a, and b are calculated by using the above equation. The resultant values from Lucas-Kanade algorithm is used to calculate the optical flow using equation 1.

2.3 Velocity Estimation Approach

Tracking of the image sequences gives the velocity. Velocity measures the distance traveled over change in time, so for the input video, the motion of the human face is calculated from the first frame to the last frame using Sparse Landmark Points where the points from Eyes, Nose, and Mouth regions are selected. Euclidean distance is used to measure the distance traveled by the human face from first to last frames. These specific points are useful in finding the 2D imposter faces and fake photo faces effectively. Thus, the velocity of eyes, nose, and mouth regions from consecutive frames are calculated using Euclidean distance as shown in equation 6.

$$velocity = \frac{Euclidean\ distance}{total\ no.of\ frames} \quad (5)$$

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (6)$$

The average velocity of selected landmark points against each frame is calculated and plotted as a graph. This graph shows the sudden and gradual changes that occur due to the 'shaking' action. For each and every video sample, the Critical Velocity (CV) value is calculated using the proposed formula as follows,

$$CV = mean + \frac{mean}{2} \quad (7)$$

$$mean = \frac{max.velocity}{2} \quad (8)$$

After finding the CV values, the number of peaks in the graph above the CV value is counted to distinguish the real face and masked faces. If the number of peaks is greater than the threshold (th), then the input video is a real face video. If the number of peaks is lesser than or equal then the input video is masked face video. Here, the threshold is assigned as 2.

Algorithms

Algorithm 1: Proposed SOFT_VEA algorithm

Input: Sample video

Output: Live or Masked face prediction

1. avg_velocity[n] = 0

2. Get the window size;

3. **For** each frame k **do**

While (is landmarks found) **do**

 x,y ← landmark points

 a,b ← vector values

 p1.x = p0.x+a;

 p1.y = p0.y+b;

 p0 → old frame;

 p1 → new frame;

if (no landmarks found) **then**

break;

end While

 for i = 0 to len_ p1

dist[i] = dist(p0[i]-p1[i]); / dist refers to Euclidean distance*/*

velocity[i] = dist[i]/ frame_count;

avg_velocity[k] = avg_velocity[k]+velocity[i];

end for

end for

4. plot avg_velocity; /*graph generation*/

5. temp = mean(max(avg_velocity));

6. cv = temp+(temp/2); /* critical velocity calculation*/

7. **for** i = 0 to Total_number_of_frames **do**

 count the peak points greater than cv in graph;

end for

8. **if** count > th

 return “Real face”

else

 return “Fake/Masked face”

end if

End

3 Results and Discussion

3.1 Database

ROSE-YOUTU database is used in this proposed work. It contains around 150 videos of each from 20 different persons of the categories with glass, without glass, photo imposters or 2D masks, photo imposters or 2D masks with eyes and mouth open and video imposters. The length of the one video clip is about 10 to 15 seconds with 30 frames per second.

3.2 Implementation

Proposed work has been implemented by Python 3.6 version along with Anaconda Library. OpenCV (version- 4.2.0) library is imported for handling videos and image files.

3.3 Experimental Results

The experiment is conducted with 60 sample videos of three cases. Case 1 is the real faces of 20 different persons. Case 2 is 20 videos of persons with photo imposters or 2D masks. Case 3 is 20 videos of persons using photo imposter or 2D masks with eye and mouth openings. Figure 4 shows the sample databases of photo imposters, photo imposters with eyes and mouth open, and real face videos used in this work for tracking optical flow.

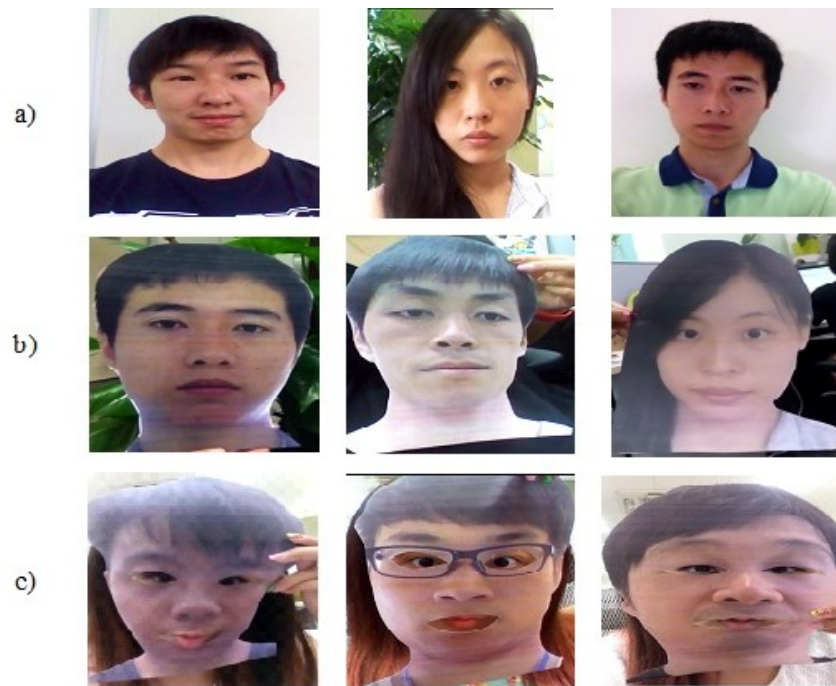


Fig 4. Sample databases of a) case 1, b) case 2, c) case 3

For this work, out of 68 landmark points only 41 points i.e. eyes, nose, and mouth landmark points are selected. In Figure 5, the output screenshots show the proposed optical flow algorithm tracking the facial movements using landmark points on the sample database of all three categories. The calculated velocity using equation 5 of optical flow for each video is represented in terms of a graph with frames as x-axis and velocity as y-axis.

For the real human faces micro facial movements made low intensity value. But, for the 2D masked faces facial movements are purposely made by the humans which cause high intensity values. Velocity increases with increase in intensity, so velocity graph is used to differentiate real faces with 2D masked or fake faces.

Figure 6 shows the optical flow graph correspond to the velocity calculated for each frame in the sample video. If the peaks CV are greater than the assigned th value 2, then it is categorised as fake face videos.

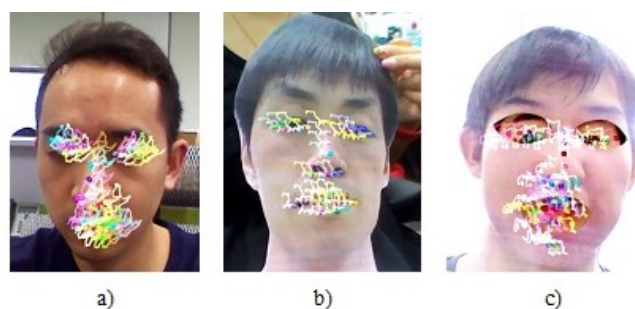


Fig 5. Screenshots of outputs using proposed SOFT_VEA algorithm for a) case 1, b) case 2, c) case 3

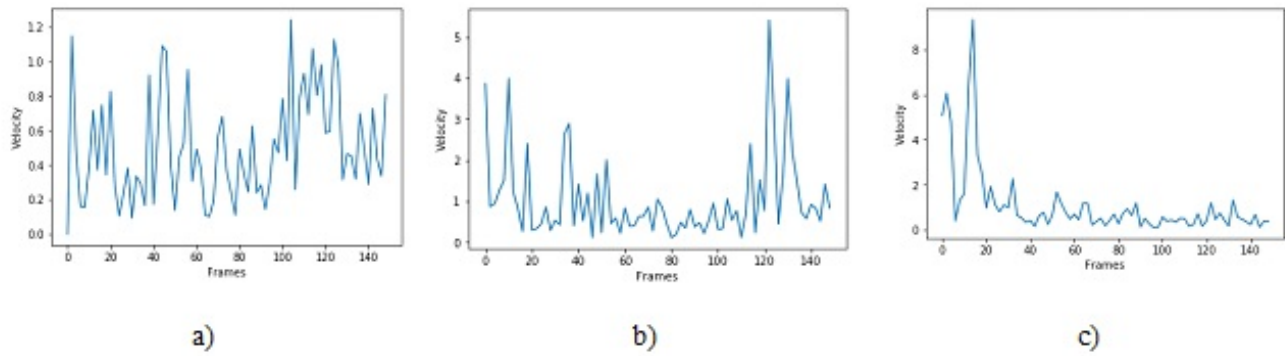


Fig 6. Velocity graph for a) case 1, b) case 2, c) case 3.

Table 1. Experimental results for case 1

| File_name | Critical velocity (cv) | No. of peaks | Inference |
|-----------|------------------------|--------------|-----------|
| Sample2_R | 0.17 | 5 | Real |
| Sample3_R | 3.61 | 2 | Fake |
| Sample4_R | 3.64 | 3 | Real |
| Sample5_R | 3.05 | 3 | Real |
| Sample6_R | 3.88 | 5 | Real |
| Sample7_R | 1.38 | 6 | Real |
| Sample9_R | 0.93 | 9 | Real |

Table 2. Experimental results for case 2

| File_name | Critical velocity (cv) | No. of peaks | Inference |
|------------|------------------------|--------------|-----------|
| Sample2_IP | 1.42 | 2 | Fake |
| Sample3_IP | 0.19 | 5 | Real |
| Sample4_IP | 2.77 | 4 | Real |
| Sample5_IP | 3.02 | 2 | Fake |
| Sample6_IP | 4.06 | 1 | Fake |
| Sample7_IP | 1.09 | 2 | Fake |
| Sample9_IP | 3.33 | 1 | Fake |

Table 3. Experimental results for case 3

| File_name | Critical velocity (cv) | No. of peaks | Inference |
|-------------|------------------------|--------------|-----------|
| Sample2_IPC | 3.92 | 1 | Fake |
| Sample3_IPC | 1.09 | 4 | Real |
| Sample4_IPC | 2.43 | 2 | Fake |
| Sample5_IPC | 2.96 | 1 | Fake |
| Sample6_IPC | 1.21 | 1 | Fake |
| Sample7_IPC | 12.6 | 1 | Fake |
| Sample9_IPC | 2.33 | 2 | Fake |

From [Table 1](#), it is inferred that for samples of real face input video, the proposed SOFT_VEA optical flow algorithm finds 6 as real faces and one as fake faces out of 7 real faces videos with an accuracy of 88%. In [Table 2](#), the experimental results for 7

input videos with photo imposter show that the proposed algorithm finds 5 masked faces as fake faces correctly and 2 masked faces as real faces. So the accuracy for this photo imposter input category is 71%. From Table 3, it is inferred that the proposed optical flow algorithm for inputs of photo imposter with eye and mouth openings finds 6 out of 7 masked faces correctly with an accuracy of 88%.

Half Total Error Rate (HTER) is used to evaluate the detection performances, which is a combination of False Acceptance Rate (FAR) and False Rejection Rate (FRR).

$$HTER = \frac{1}{2}(FAR + FRR) \quad (9)$$

From the above equation, the HTER value for the proposed optical flow algorithm using facial landmark detection is calculated as 2.45.

Table 4. Comparison of HTER value for ROSE-YOUTU database

| Methods | Accuracy (%) | HTER |
|------------------------------------|--------------|------|
| DTLP ⁽¹⁶⁾ | 80 | 6.1 |
| Optical flow fields ⁽⁹⁾ | 83 | 4 |
| Proposed SOFT_VEA | 88 | 2.45 |

DLTP (Dynamic Local Ternary Pattern) is used to analyse the skin textures for finding out the masked faces among real faces⁽¹⁶⁾. The optical flow field methods use the degree of change in facial movements using a traditional optical flow algorithm⁽⁹⁾. In this work, instead of corner detection in a traditional optical flow algorithm, facial landmark points are used and its critical velocity is calculated using Euclidean distance. From Table 4, it is inferred that our proposed SOFT_VEA algorithm has higher accuracy and a lower error rate of 2.45 than existing methods.

4 Conclusion

This work proposed the Sparse Optical Flow Method with Lucas Kanade algorithm using facial landmark detection and determined the real and fake faces using the Velocity Estimation Approach. The proposed model has experimented with three kinds of input videos with persons using real faces, photo imposter, and photo imposter with eye and mouth openings. The performance of the proposed work is evaluated by using accuracy values. The proposed model gives less accuracy for the input of person with photo imposter or 2D masks because it seems to real faces but photo imposter with eye and mouth opening shows the variation in trajectory values for real and masked regions. From the experimental results, it is inferred that the proposed model gives better accuracy of 88% for real face and photo imposter with eye and mouth openings than other state-of-art optical flow methods and lower error rate of 2.45 than existing systems. The future work will consider continuing to optimize the model to further enhance its security performance.

References

- Hasan M, Mahmud SMH, Li XY. Face Anti-Spoofing Using Texture-Based Techniques and Filtering Methods. *Journal of Physics-conference Series*. 2019;p. 1–10. Available from: [10.1088/1742-6596/1229/1/012044](https://doi.org/10.1088/1742-6596/1229/1/012044).
- Bhele S, Mankar V. A Review Paper on Face Recognition Techniques. *International journal of research in computer science engineering and technology*. 2012;1:2278–2311.
- Parmar D, Mehta B. Face Recognition methods and applications. *International Journal of Computer Technology Applications*. 2013;4:84–86. Available from: <https://arxiv.org/abs/1403.0485>.
- Kollreider K, Fronthaler H, Faraj MI, Bigun J. Real-Time Face Detection and Motion Analysis With Application in “Liveness” Assessment. *IEEE Transactions on Information Forensics and Security*. 2007;2(3):548–558. Available from: <https://dx.doi.org/10.1109/tifs.2007.902037>.
- Shin J, Kim S, Kang S, Lee SW, Paik J, Besmaabidi. MongiAbidi Optical flow-based real-time object tracking using non-prior training active feature mode. *Real-Time Imaging*. 2005;11:204–218. Available from: [10.1016/j.rti.2005.03.006](https://doi.org/10.1016/j.rti.2005.03.006).
- Kollreider K, Fronthaler H, Bigun J. Evaluating liveness by face images and the structure tensor. In: Proceedings of 4th IEEE Workshop on Automatic Identification Advanced Technologies AutoID. 2005;p. 75–80. doi:10.1109/AUTOID.2005.20.
- Ahmad R, Naghsh-Nilchi, Roshanzamir M. An Efficient Algorithm for Motion Detection based Facial Expression Recognition using Optical Flow. 2006;14:318–323. Available from: <https://doi.org/10.5281/zenodo.1063056>.
- Kollreider K, Fronthaler H, Bigun J. Non-intrusive liveness detection by face images. *Image and Vision Computing*. 2009;27(3):233–244. Available from: <https://dx.doi.org/10.1016/j.imavis.2007.05.004>.
- Bao W, Li H, Li N, Jiang W. A liveness detection method for face recognition based on optical flow field. *proceedings of International Conference on Image Analysis and Signal Processing*. 2009;p. 233–236. Available from: [10.1109/IASP.2009.5054589](https://doi.org/10.1109/IASP.2009.5054589).
- Fujita K, Hanada T, Kitazawa Y, Kawabe A. A debris image tracking using optical flow algorithm. *Advances in Space Research*. 2012;49(5):1007–1018. Available from: <https://dx.doi.org/10.1016/j.asr.2011.12.010>.

- 11) Singh AK, Joshi P, Nandi GC. Face Recognition with Liveness Detection using Eye and Mouth Movement. In: Proceeding of International Conference on Signal Propagation and Computer Technology (ICSPCT). 2014;p. 592–597. doi:10.1109/ICSPCT.2014.6884911.
- 12) Singh M, Arora AS. A robust anti-spoofing technique for face liveness detection with morphological operations. *Optik*. 2017;139:347–354. Available from: <https://dx.doi.org/10.1016/j.ijleo.2017.04.004>.
- 13) Feng L, Po LM, Li Y, Xu X, Yuan F, Cheung TCH, et al. Integration of image quality and motion cues for face anti-spoofing: A neural network approach. *Journal of Visual Communication and Image Representation*. 2016;38:451–460. Available from: <https://dx.doi.org/10.1016/j.jvcir.2016.03.019>.
- 14) S K, S V. Contour-based object tracking in video scenes through optical flow and gabor features. *Optik*. 2018;157:787–797. Available from: <https://dx.doi.org/10.1016/j.ijleo.2017.11.181>.
- 15) Pragadeeswari CK, Yamuna G, Beham GY. Optical flow detection and tracking of gauzes and cancer cells. *Materials Today: Proceedings*. 2020;33:3558–3563. Available from: <https://dx.doi.org/10.1016/j.matpr.2020.05.525>.
- 16) Parveen S, Ahmad SM, Abbas NH, Adnan WA, Hanafi M, Naeem N. Face liveness detection using dynamic local ternary pattern (DLTP). *First Computing*. 2016;5(2). Available from: <https://doi.org/10.3390/computers5020010>.