iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** XI **Month of publication:** November 2026

DOI: https://doi.org/10.22214/ijraset.2026.85626

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue XI Nov 2026- Available at www.ijraset.com

Design and Simulation of a Face Recognition-Based Smart Door Lock Using Python and OpenCV

Sagar T G[1], Yallaling[2], Yamuna M E[3], Yashavanth M N[4], Dr. G. D. Gurumurthy[5]
Department of Electronics and Communication Engineering, KIT, Tiptur, Karnataka, India

Abstract: ***Keys, access cards and PIN codes can be misplaced, duplicated or handed to others, and an ordinary lock offers the owner no means of judging a person who has not been registered. This paper proposes a software-simulated smart door lock that admits registered people through face recognition and forwards every unregistered visitor to the owner for a decision. A webcam provides live frames to a Haar Cascade detector. Each face found is normalised and compared with a stored set of enrolled people by the Local Binary Patterns Histograms (LBPH) recognizer in OpenCV, and a distance threshold separates familiar faces from strangers. An enrolled person opens a virtual door at once. For a stranger, the program sends the captured image to the owner's Telegram bot together with Allow and Deny buttons, and the door opens only after Allow is pressed; Deny, silence or any fault leaves it locked. All events are recorded in a time-stamped log. The prototype runs in Python on an ordinary laptop, so no Raspberry Pi board or lock hardware is required during testing, while the modular code can later drive a relay. The paper explains the architecture, algorithms, parameter choices, security considerations, expected strengths and weaknesses, and a structured test plan. No measured accuracy is reported, because the prototype has not yet been evaluated.***

Keywords: Smart door lock, face recognition, LBPH, OpenCV, Haar cascade, Telegram bot, access control, remote approval, Python.

I. INTRODUCTION

Deciding who may enter a house, laboratory or office is one of the oldest security tasks. The solutions in common use, namely mechanical keys, keypad codes and proximity cards, all verify possession of a token or a secret rather than the identity of the person. A key can be cast again from a copy, a code can be seen over a shoulder, a card can be lent, and none of these leaves any trace of who actually passed through the door. Biometric traits such as the face, fingerprint and iris are attractive because they are tied to the person and cannot be forgotten at home [11].

A lock that relies on the face alone, however, behaves as if every unfamiliar person were an intruder. Real households and workplaces do not work that way. A courier, a relative on a surprise visit or a repair technician is not enrolled in any database, yet the owner may be glad to let them in, even when the owner is away. A strict system causes inconvenience, whereas a lenient one that accepts anything resembling a face is unsafe. The design presented here takes a middle path: enrolled people pass automatically, and the decision about everyone else is delegated to the owner through a familiar messaging application.

The work is framed as an electronics and communication engineering student prototype. Its purpose is to show how detection, recognition, decision logic, remote messaging and logging can be combined and tested on a laptop before hardware is purchased. It is not offered as a replacement for certified commercial access-control products.

The main contributions of this paper are: (i) a complete processing pipeline with clearly separated modules; (ii) a failsafe approval protocol that uses a request identifier, an owneronly check and a time limit; (iii) a threat analysis that lists attacks against the prototype and the countermeasure for each; and (iv) an evaluation plan with metrics, a threshold-selection procedure and sample-size guidance, so that results can be reported without exaggeration.

The remainder of the paper is organised as follows. Section II states the problem, Section III reviews related work, and Sections IV and V describe the system and its architecture. Section VI covers implementation details, Section VII the security and privacy analysis, Section VIII the requirements, Section IX the operating method, Section X the evaluation plan, and Sections XI to XIII the benefits, limitations, future work and conclusions.

II. PROBLEM STATEMENT AND OBJECTIVES

A conventional installation has four weaknesses. First, credentials can be lost, copied or guessed. Second, the lock cannot distinguish a family member from a stranger, so the owner has no information about a visitor. Third, attempts, whether successful or not, are not recorded. Fourth, building and field-testing a physical lock at the very start of a student project is slow and expensive, and mistakes can damage equipment.

©IJRASET: All Rights are Reserved

A software model that behaves like the intended product allows the recognition and approval logic to be debugged first.
The principal objective is a prototype that detects faces in webcam video, identifies enrolled people and switches a virtual door to the unlocked state for them. The supporting objectives are listed below.

- Forward a photograph of each unfamiliar visitor to the owner's Telegram account with Allow and Deny options.
- Keep the door locked unless the owner explicitly allows entry, and on any error.
- Record each attempt with the time, the identity (or "Unknown") and the outcome.
- Keep the modules independent so that a relay-driven lock can replace the virtual door.
- Define tests and metrics that make an objective evaluation possible.

III. RELATED WORK

A. *Face detection*

Viola and Jones showed that a cascade of boosted classifiers built from simple rectangular features can discard most nonface windows after very little computation, which made realtime detection practical [1]. Lienhart and Maydt enlarged the feature pool with rotated Haar-like features [2]. Trained cascades are bundled with the OpenCV library [3] and frequently serve as the first stage of a recognition chain. A detector only returns the position of a face; it carries no information about whose face it is.

B. *Face recognition*

Turk and Pentland projected face images onto a small set of principal components, the eigenfaces, and identified a person by the nearest neighbour in that subspace [4], [5]. Belhumeur *et al.* later argued that class-specific linear projection (Fisherfaces) copes better with illumination and expression changes than the unsupervised eigenface projection [12]. Both are holistic methods, so a local change such as a shadow can disturb the entire feature vector.
Local Binary Patterns (LBP) describe micro-texture by thresholding the neighbours of each pixel against its centre value; Ojala *et al.* extended the idea to several scales and made it rotation invariant [6]. Ahonen *et al.* split the face into a grid of regions, built an LBP histogram for each, and joined them into one descriptor [7], [8]. As the code depends on the ordering of grey levels and not on their absolute values, a uniform brightness shift leaves it unchanged. The OpenCV LBPH recognizer follows this principle and is used in the present work. Deep embedding networks such as FaceNet map a face to a compact vector whose distances reflect identity [9] and generally outperform hand-crafted descriptors, but they need more data, memory and computation.

Aspect	Eigenfaces	LBPH	Deep embedding
Feature type	Global, linear	Local texture	Learned, non-linear
Lighting tolerance	Low	Moderate	High (with data)
Training effort	Low	Low	High / pre-trained
Compute load	Low	Low	Medium to high
Update with new person	Re-project	Update or retrain	Add embedding

C. *Messaging-based approval*

The Telegram Bot API lets a program send photographs, attach inline buttons to a message and obtain the button press as a callback query [10]. This offers a ready-made mobile interface for owner approval, so that no dedicated phone application has to be written. The present proposal joins face recognition, a virtual door and remote approval in a single demonstrator. It does not claim a new recognition algorithm, nor does it claim to match the verification strength demanded of commercial security products.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue XI Nov 2026- Available at www.ijraset.com

IV. PROPOSED SYSTEM OVERVIEW

The system consists of six functional units: (1) camera and face detection, (2) known-person dataset, (3) LBPH recognition, (4) access decision, (5) Telegram approval, and (6) virtual door with logging. A frame is grabbed, any face in it is located, cropped and normalised, and the recognizer then returns the most similar enrolled label together with a distance d. A threshold τ turns this distance into a known/unknown decision.

When the person is known the virtual door unlocks immediately. When the person is unknown, the program transmits the photograph to the owner and keeps the door locked while it waits. Allow opens the door for that single visit; Deny or silence leaves it locked. Every outcome is written to the log.

V. SYSTEM ARCHITECTURE

Fig. 1 shows the processing chain. Webcam frames go to the Haar Cascade detector; each detected face is cropped, converted to grey scale and resized, and then matched against the trained LBPH model. The distance d is compared with τ, and the result selects either the direct unlock path or the Telegram approval path.

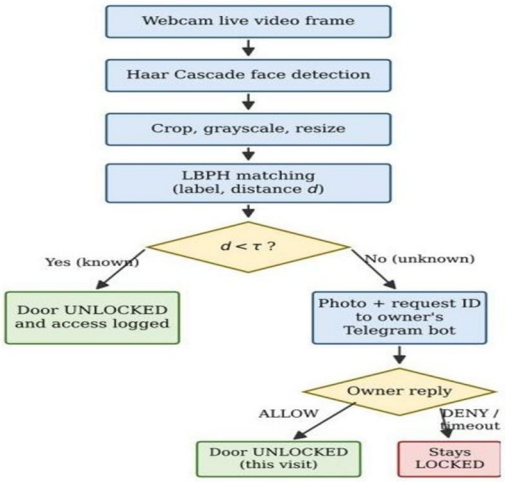


Fig. 1. Proposed processing flow of the smart door lock.

A. Camera and Face Detection Unit

A built-in or USB camera supplies the video. The cascade classifier scans every frame at several scales and returns bounding boxes for frontal faces. Two parameters influence its behaviour: the scale factor, which sets how much the image is shrunk between scans, and the minimum-neighbours count, which suppresses isolated false detections. If no face is found the next frame is read. If several faces appear, only the largest box is processed, because it normally belongs to the person closest to the door. The chosen face is cropped, converted to grey scale and resized to a fixed size. The same preprocessing function is used for enrolment and recognition so that both stages see identical input statistics. Optional histogram equalisation can reduce the effect of poor exposure.

B. Known-Person Dataset

Every authorised person is enrolled by capturing a set of face images with the webcam. The images should differ in head angle, expression and lighting so that the stored histograms reflect conditions at the door; a few tens of images per person is a reasonable starting point. Processed images and their numeric labels are used to train the LBPH model, which is saved to a file. A name-to-label table is kept alongside it. To add a person, new images are captured and the model is updated or retrained. Image files are stored in a folder that only the program account can read.

C. LBPH Recognition Unit

For a pixel with grey value g_c and P sampling neighbours with values g_p, the LBP code is $\mathrm{LBP} = \Sigma\, s(g_p - g_c)\, 2^p$, $p = 0,\ldots,P-1$ (1) where $s(x) = 1$ for $x \geq 0$ and 0 otherwise. The face image is divided into a grid of cells; for each cell a histogram of the LBP codes is computed, and the cell histograms are concatenated into one feature vector. With the common OpenCV settings of radius 1, eight neighbours and an 8×8 grid, the vector contains $64 \times 256 = 16\,384$ bins. A live vector S is compared with each stored

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

vector M using the chi-square distance $d(S,M) = \Sigma i\ (Si - Mi)^2 / (Si + Mi)$ (2) and the nearest stored vector supplies the predicted identity and the distance d. In OpenCV a smaller d indicates a closer match.

D. Access Decision Unit

decision = known if $d < \tau$, otherwise unknown (3) The threshold τ sets the balance between two kinds of error. A low value rejects more impostors but also more genuine users; a high value does the reverse. It must be chosen from test data (Section X). Since accepting a wrong person is more harmful than rejecting a right one, τ should lean toward the strict side. The owner-approval step then gives a wrongly rejected genuine user a second chance. If the cost of a false acceptance is cFA and that of a false rejection is cFR, the operating point can be viewed as minimising $c_{FA} \cdot FAR \cdot p_{imp} + c_{FR} \cdot FRR \cdot p_{gen}$, where $p$imp and $p$gen are the fractions of impostor and genuine attempts; a security product sets cFA well above cFR.

E. Telegram Approval Unit

When the decision is "unknown", the program saves the photograph and sends it to the owner with the Bot API *sendPhoto* method [10]. The message includes an inline keyboard with two buttons, Allow and Deny. The button data contain a unique request identifier (the Bot API limits this field to 64 bytes), so a late press on an old alert cannot open the door for a new visitor. The reply is obtained by long polling with *getUpdates*, which means no public web server is exposed. After a press, the program acknowledges the callback and removes the buttons so that the same alert cannot be answered twice. The following rules keep the step safe.

- Only callbacks from the owner's Telegram chat identifier are accepted; every other sender is ignored.
- The bot token is read from a configuration file or an environment variable, never from the source code.
- Deny, or no reply within the time limit, keeps the door locked.
- Allow unlocks the door for that visit only and does not add the visitor to the dataset.

F. Virtual Door Controller and Logging

The door is modelled by a graphical window showing the camera feed, the recognition result and a LOCKED or UNLOCKED indicator, so that the whole system can be demonstrated without hardware. After a fixed delay the indicator returns to LOCKED. Each attempt is appended to a log as a record of time stamp, recognised name or "Unknown", distance value and outcome (granted, owner-allowed, ownerdenied or timed-out). Storing the distance helps later tuning of τ.

VI. IMPLEMENTATION DETAILS

The software is split into four Python modules that exchange data through plain function calls: *enroll* (capture and preprocess images), *detect* (frame grabbing and face localisation), *recognize* (model loading, prediction and decision) and *notify* (Telegram messaging and callback handling). A small controller, which also draws the graphical interface, owns the door state and the log. Keeping the modules free of global state makes each of them testable on its own with stored images.

A. Control Algorithm

The main loop is summarised below. The state variable prevents repeated alerts while a request is open.

```
load(model, config); door = LOCKED; state = IDLE loop:   frame =
camera.read()   face = detect_largest(frame)  if face is None: continue
(label, d) = recognize(preprocess(face))   if d < tau:       unlock();
log(name[label], d, 'granted')   elif state == IDLE:        rid =
new_request_id()

    send_photo(owner, face, rid); state = PENDING   if state ==
PENDING:         r = poll_reply(rid, timeout)

    if r == 'allow': unlock_once(); log('owner-
allowed')

    else: log('owner-denied' or 'timed-out')
```

Algorithm 1. Control loop (simplified pseudocode).

B. Door States

Fig. 2 presents the states of the door controller. Every error path leads back to the locked condition, and the cool-down state prevents the flood of alerts that would otherwise appear while a stranger stands in front of the camera.

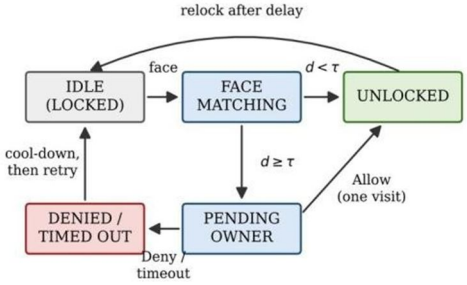


Fig. 2. State diagram of the door controller.

C. Configurable Parameters

Table II lists the parameters that the program reads from a configuration file. The values are starting points for experiments and are not claimed to be optimal; the final report should state the values actually used.

Parameter	Initial value	Role
Face size	100×100 px	Normalised input to LBPH
LBP radius / neighbours	1 / 8	OpenCV default pattern
Grid cells	8 × 8	Spatial histogram layout
Scale factor / min. neighbours	1.1 / 5	Haar detector sensitivity
Threshold τ	from sweep	Known/unknown boundary
Owner reply limit	60 s (tunable)	Wait before timed-out
Relock delay	5 s (tunable)	Return to LOCKED
Alert cool-down	30 s (tunable)	Suppress repeat alerts
Images per person	20–50	Enrolment set size

D. Enrolment Guidelines

The quality of the dataset largely decides how well LBPH performs, so the following practice is recommended.

- Capture images in the same place and at the same camera height as the final installation.
- Include frontal and slightly turned poses, neutral and smiling expressions, and morning and evening light.
- Discard blurred frames and frames in which the detector box cuts off part of the face.
- Keep a few images of each person aside as a validation set for the threshold sweep.

VII. SECURITY AND PRIVACY ANALYSIS

A smart lock is a security device, so a prototype should state openly how it can fail. Table III lists the main threats and the countermeasure proposed for each. The first row is a known weakness that the present version does not remove.

Threat	Effect	Countermeasure
Printed photo or video replay	Impostor accepted as enrolled person	Not mitigated; liveness check is future work; record behaviour in tests
Stolen bot token	Attacker sends or reads messages	Keep token in environment variable; revoke and reissue via BotFather
Reply from other chat	Unauthorised unlock	Accept only owner chat ID

Old button pressed	Door opens for wrong visitor	Unique request ID, one-time use, time limit
Network outage	Owner cannot answer	Door stays locked; event logged; retry
Alert flooding	Owner fatigue, rate limits	Single open request, cool-down
Owner misjudgment	Stranger admitted	Show clear photo and time; owner remains responsible
Log or image theft	Privacy loss	Restrict file access; encrypt in later version

Face images are personal biometric data. Even in a student project they should be stored only as long as needed, kept off shared drives, and collected with the consent of the people enrolled. Data-protection rules such as the EU GDPR [13] and, in India, the Digital Personal Data Protection Act, 2023 [14], treat such data with care, and any deployment beyond a classroom demonstration should be reviewed against the law that applies. Visitors' photographs are sent to a third-party messaging service, which is a further privacy consideration that the owner should accept knowingly.

It should also be noted that presentation-attack resistance is a separate research area with its own test standard [15]. A system without liveness detection cannot meet it, which is why the present design is labelled a prototype.

VIII. HARDWARE AND SOFTWARE REQUIREMENTS

The prototype needs a laptop running the Python program and the graphical interface, a built-in or USB camera placed at the door position, and an internet connection for the Telegram alert and the owner's reply. Local recognition works offline. The owner's smartphone with the Telegram application receives the photo and the two buttons. No microcontroller is needed for the simulation.

Table IV lists the tools.

Tool	Purpose
Python 3.12	Main programming language
OpenCV (with contrib modules)	Camera access, image processing, Haar cascade, LBPH recognizer
NumPy	Array handling for image data
Tkinter	Graphical interface and virtual door display
Telegram Bot API	Photo alert and Allow / Deny callback
CSV / SQLite	Storage of the access log
Webcam (laptop / USB)	Live video capture

The final report should state exact library versions and the threshold chosen, so that the experiment can be repeated.

IX. OPERATING METHODOLOGY

At start-up the program loads the trained model, opens the camera, reads the Telegram settings and shows the door as LOCKED. The loop then reads frames continuously. If no face is detected, no decision is made and the door state is unchanged.

When a face is detected it is preprocessed and matched. If the distance is below τ, the person is treated as known: the door changes to UNLOCKED, the name and time are logged, and the door relocks after the delay. If the distance is above τ, the door stays LOCKED, the photograph is sent to the owner, and the program waits for a reply up to the time limit.

A person standing in front of the camera appears in many consecutive frames, so once an alert has been sent further alerts are suppressed until the request is answered or has timed out, followed by a short cool-down. A decision based on a single frame is also fragile. Requiring the same label in several consecutive frames before an automatic unlock is an inexpensive way to reduce accidental acceptance, at the cost of a slightly longer delay. For faults such as a camera that cannot be opened, a missing model file or loss of internet access, the door remains locked, the event is logged locally, and notification is retried under a controlled policy.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue XI Nov 2026- Available at www.ijraset.com

X. TESTING AND EVALUATION PLAN

No measured performance is claimed in this paper because the evaluation is not finished. The prototype should be tried with the scenarios in Table V, which cover normal operation, difficult lighting, unknown visitors, each owner response and fault conditions. The photo-presentation test is included deliberately: a system without liveness detection is expected to be vulnerable, and recording this behaviour gives an honest account of its limits.

Scenario	Expected behaviour
Known person, normal light	Recognised; door unlocked; logged
Known person, dim or bright light	Result recorded; τ tuned if needed
Known person, glasses or cap	Result recorded; effect on d noted
Unknown person	Door locked; photo sent to owner
Owner presses Allow	Door unlocked for that visit
Owner presses Deny or no reply	Door stays locked; logged
Old alert pressed late	Ignored; door stays locked
Message from another chat	Ignored
No face in frame	Capture continues; no decision
Photo of known person shown	Behaviour observed and recorded
Camera or internet unavailable	Error shown; door stays locked

A. Evaluation Metrics

Let N_{gen} be the number of genuine attempts by enrolled persons and N_{imp} the number of attempts by unknown persons.
With FR genuine attempts wrongly rejected and FA impostor attempts wrongly accepted as known, the false rejection and false acceptance rates are

$$\text{FRR} = FR / N_{\text{gen}} \times 100\ \% \quad (4)$$

$$\text{FAR} = FA / N_{\text{imp}} \times 100\ \% \quad (5)$$

Recognition accuracy is the percentage of all attempts classified correctly. The mean processing time per frame and the mean time between sending an alert and receiving the owner's reply should also be recorded. For the whole system, an additional figure of interest is the owner-alert rate, meaning the share of genuine visits that trigger a Telegram request, since it shows how often a registered user is inconvenienced.

B. Threshold Selection

The threshold τ is swept over a range of values; at each value FAR and FRR are calculated from the logged distances, and the resulting curve shows the trade-off. The value where the two rates are equal (the equal error rate) is a useful single summary for comparing conditions, but because a false acceptance is worse in this application, the final τ should be taken from the part of the curve where FAR is as low as practical. The sweep should use data that were not used for training, for example by keeping some enrolment images aside or by k-fold splitting.

C. Sample Size and Conditions

The reliability of a measured rate depends on the number of trials. As a rule of thumb, if no errors are seen in n independent trials, the 95 % upper confidence bound on the true error rate is about $3/n$, so 100 trials cannot show an error rate below roughly 3 %. The plan therefore proposes several enrolled volunteers, many images and attempts per person, and separate runs in normal, low and strong light. Each reported figure should carry the number of trials and the conditions.

©IJRASET: All Rights are Reserved

XI. EXPECTED BENEFITS AND LIMITATIONS

A. *Expected Benefits*

Entry needs no key, card or PIN that can be carried, lost or shared. Unknown visitors can be accepted or refused by the owner from a phone, so the owner stays in charge even when away. The system uses only a laptop, a webcam and free software, and the time-stamped log makes entries traceable. A new person is added by capturing a few more images, and because recognition runs locally, known users are admitted without waiting for the network. The modular code can later drive a Raspberry Pi relay lock.

Table VI places the proposal beside common access methods. The ratings are qualitative and reflect general properties of each method, not measurements from this work.

Method	Can be lost / lent	Visitor handling	Access record	Extra cost
Mechanical key	Yes	None	No	Low
Keypad PIN	Shared or seen	None	Rarely	Low
RFID card	Yes	None	Possible	Medium
Face only	No	Refuses all	Yes	Low
Proposed	No	Owner decides	Yes	Low

B. *Limitations*

Recognition quality depends on lighting, head pose and image quality, as noted for appearance-based methods in [5] and [8]. The Haar detector performs best on frontal faces and may miss a turned head. Without liveness detection a printed photograph or a replayed video of an enrolled person may be accepted, so the present version suits a prototype and not a highsecurity installation. Owner approval needs an internet connection and an owner who can respond. Finally, this is a software simulation; the behaviour of a real lock, its power supply and the door hardware is outside the scope of this work.

XII. FUTURE ENHANCEMENTS

Several improvements can be considered once the basic prototype has been validated. The virtual door can be replaced by a Raspberry Pi, a relay and an electric strike that are driven by the same modules, with a physical override key for power failure. Liveness detection, for example a blink, headmovement or depth check, would reduce photo and video spoofing. LBPH can be replaced by a deep embedding model [9] for better accuracy under varied conditions. The Telegram interface could let the owner enrol an approved visitor from the chat and send lock or unlock commands. Logs and face data can be stored in encrypted form, with support for more than one door and more than one authorised owner.

XIII. CONCLUSION

This paper proposed a smart door lock in which enrolled people are admitted by face recognition and unknown visitors are referred to the owner through a Telegram bot with Allow and Deny buttons. Haar cascade detection, LBPH recognition and a distance threshold keep the system light enough for a laptop, while fail-safe rules and the owner's final decision limit the consequence of recognition errors. The modular design allows the virtual door to be exchanged for a physical one.

The usefulness of the design depends on good enrolment, a carefully chosen threshold, stable network access and sensible handling of repeated detections. The prototype should be evaluated according to Section X, and real measurements should be reported before any claim is made about accuracy or response time.

XIV. ACKNOWLEDGMENT

The authors thank their project guide and the Head of the Department of Electronics and Communication Engineering, KIT, Tiptur, for their guidance and support.

©IJRASET: All Rights are Reserved

REFERENCES

[1] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Kauai, HI, USA, 2001, pp. I-511–I-518.

[2] R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," in Proc. IEEE Int. Conf. Image Process. (ICIP), Rochester, NY, USA, 2002, pp. I-900–I-903.

[3] G. Bradski, "The OpenCV library," Dr. Dobb's J. Softw. Tools, 2000.

[4] M. Turk and A. Pentland, "Eigenfaces for recognition," J. Cogn. Neurosci., vol. 3, no. 1, pp. 71–86, 1991.

[5] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), Maui, HI, USA, 1991, pp. 586–591.

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

[7] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[8] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in Proc. Eur. Conf. Comput. Vis. (ECCV), Prague, Czech Republic, 2004, pp. 469–481.

[9] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Boston, MA, USA, 2015, pp. 815–823.

[10] Telegram, "Telegram Bot API." [Online]. Available: https://core.telegram.org/bots/api. Accessed: Nov. 3, 2026.

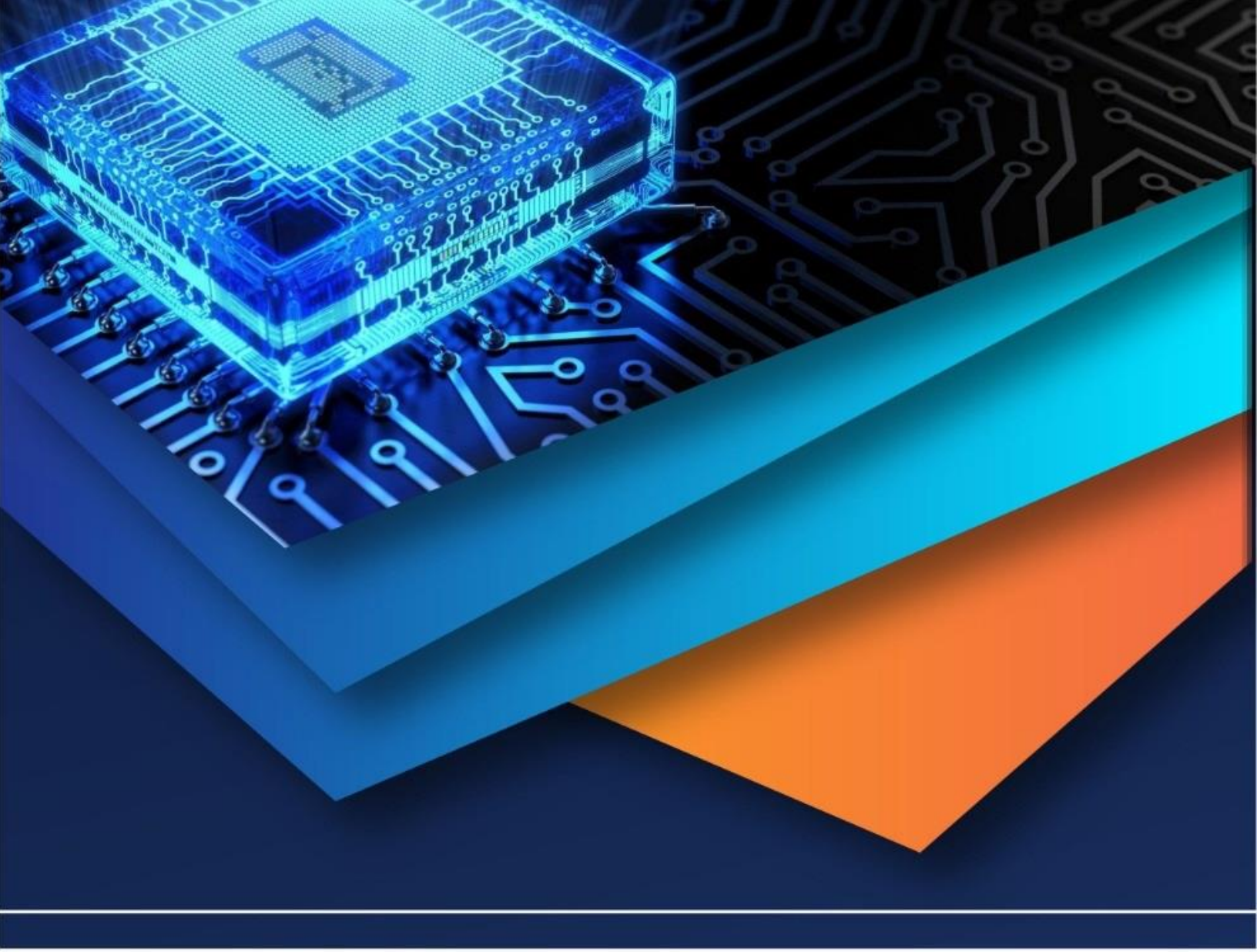
[11] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," IEEE Trans. Circuits Syst. Video Technol., vol. 14, no. 1, pp. 4–20, Jan. 2004.

[12] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 7, pp. 711–720, Jul. 1997.

[13] European Parliament and Council, "Regulation (EU) 2016/679 (General Data Protection Regulation)," Official Journal of the European Union, L 119, 2016.

[14] Government of India, "The Digital Personal Data Protection Act, 2023," Act No. 22 of 2023, The Gazette of India, Aug. 2023.

[15] ISO/IEC 30107-3:2017, "Information technology — Biometric presentation attack detection — Part 3: Testing and reporting," 2017.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)