



Development of a Real-Time Face Recognition System Using Image Processing Techniques

Preben Zhoulei

Department of Electrical and Computer Engineering, Basra Institute of Business Technology, Iraq
preben.zhoulei@bibt-iq.org

Peer Review Information	Abstract
<i>Submission: 30 April 2016</i> <i>Revision: 09 May 2016</i> <i>Acceptance: 26 May 2016</i> Keywords <i>Face Recognition, Image Processing, Viola-Jones, Local Binary Pattern, Principal Component Analysis</i>	Face recognition is a biometric image-analysis task in which a person is identified or verified from facial appearance. This methodology paper presents a practical architecture for developing a real-time face recognition system using image processing techniques available by 2016. The design emphasizes computational efficiency and interpretability rather than a large deep-learning model. Video frames are acquired from a camera, faces are localized using the Viola-Jones boosted cascade, and detected regions are converted to grayscale, normalized, aligned, and resized. Local Binary Pattern (LBP) histograms provide the principal texture descriptor, while Principal Component Analysis (PCA) is retained as a compact baseline representation. Recognition is performed by comparing an incoming feature vector with enrolled templates using a distance measure and a decision threshold. The paper specifies enrollment, real-time processing, threshold calibration, evaluation metrics, and failure-handling procedures. Earlier eigenface, Fisherface, FERET, LBP, and real-time detection studies establish the classical foundation, while DeepFace, FaceNet, and other pre-2016 work demonstrate the performance frontier that motivated better alignment and representation. Because the present work is methodological, reported performance values are stated as transparent prototype targets rather than fabricated experimental measurements. The resulting design is suitable for attendance, access control, laboratory monitoring, and other controlled-camera applications where low latency and straightforward implementation are important.

Introduction

Automatic face recognition has been studied for decades because the face can be captured without physical contact and can support natural human-computer interaction. Early appearance-based work showed that facial images can be projected into a lower-dimensional eigenspace and matched efficiently [1]. Subsequent discriminant methods improved class separation under illumination and expression changes [2]. These ideas established a central engineering principle: a real-time system must reduce raw image data to a compact representation while preserving identity-related information.

Real-time deployment adds requirements that are not present in offline recognition. A system must continuously acquire frames, find a face quickly, normalize the detected region, extract features, search enrolled identities, and return a decision before the next useful observation becomes stale. The FERET work also demonstrated that credible recognition requires explicit testing protocols and separation between enrollment and evaluation data [3]. Viola and Jones later provided a fast boosted-cascade detector based on integral images and simple Haar-like features, enabling face detection at

real-time rates on the hardware of that period [4,6].

The proposed methodology therefore treats recognition as a sequence of measurable image-processing stages rather than as a single classifier. Detection and geometric normalization address location and scale; histogram equalization reduces illumination imbalance; LBP captures local texture; PCA supplies a compact global baseline; and template matching produces the identity decision. This structure is consistent with the broader face-recognition literature summarized by Zhao et al. [5] and with the detect-align-represent-classify pipeline later emphasized by deep systems [8]. The objective is to define a reproducible, low-complexity real-time implementation using techniques established no later than 2016.

Literature Review

Turk and Pentland introduced the eigenfaces approach, representing a face as coefficients of principal components derived from training images [1]. The method was computationally attractive and demonstrated near-real-time recognition, but global intensity patterns are sensitive to lighting, pose, and expression. Belhumeur, Hespanha, and Kriegman addressed part of this limitation through Fisherfaces, using class-specific linear projection to improve discrimination when illumination varies [2]. These subspace methods remain useful as lightweight baselines because both enrollment and matching can be implemented with matrix operations.

Benchmarking was strengthened by the FERET database and evaluation procedure, which promoted standardized gallery/probe testing and highlighted the effect of time lapse and imaging conditions [3]. A development methodology should therefore report not only recognition accuracy but also false acceptance, false rejection, and processing speed. This principle is incorporated into the evaluation protocol proposed in this paper.

For detection, Viola and Jones proposed integral images, AdaBoost feature selection, and a cascade of increasingly complex classifiers [4]. Their later journal treatment established the approach as a robust real-time face detector [6]. This detector is particularly appropriate for a 2016-era image-processing system because it can reject most background windows cheaply before spending computation on likely face regions.

Ahonen, Hadid, and Pietikainen demonstrated that Local Binary Pattern histograms provide an efficient facial descriptor with strong robustness to local illumination changes [7]. Rather than relying on absolute intensity values, LBP encodes the relationship between a pixel and its neighborhood, and regional histograms preserve coarse spatial structure. For a real-time system with modest training data, LBP is therefore a strong primary representation, while PCA can be retained for comparison or dimensionality reduction.

By 2014-2016, deep-learning systems had moved the accuracy frontier substantially. DeepFace combined explicit alignment with a deep representation and reported 97.35% verification accuracy on LFW [8]. Parkhi, Vedaldi, and Zisserman emphasized large-scale data and CNN training procedures [9], while FaceNet learned a compact embedding and reported 99.63% on LFW [10]. Center-loss learning further improved discriminative deep features in 2016 [11]. Domain-specific data augmentation was also shown to improve face recognition without requiring millions of newly collected identities [12]. These results are important context, but they require training resources and datasets beyond the intended lightweight system. The present methodology therefore adopts classical image processing while borrowing the deep-era lesson that alignment, data variation, and calibrated representations are critical.

Table 1. Representative face-recognition developments available by 2016

Study	Year	Technique	Relevance to this methodology
Turk & Pentland [1]	1991	PCA / Eigenfaces	Compact global representation and rapid matching.
Belhumeur et al. [2]	1997	Fisherfaces	Improved class discrimination under illumination changes.
Phillips et al. [3]	1998	FERET evaluation	Gallery/probe methodology and reproducible testing.
Viola & Jones [4,6]	2001/2004	Boosted cascade	Fast frontal-face localization for live video.

Ahonen et al. [7]	2006	LBP histograms	Efficient local texture descriptor for recognition.
Taigman et al. [8]	2014	DeepFace	Shows importance of alignment and learned representation.
Schroff et al. [10]	2015	FaceNet	Compact metric embedding and threshold-based matching.
Wen et al. [11]	2016	Center loss	Highlights intra-class compactness and inter-class separation.

Methodology

1. System Objective and Functional Requirements

The system is designed for one-to-many identification in a controlled indoor environment. A camera continuously supplies frames. If a face is detected, the system converts the face region into a normalized representation and compares it with enrolled templates. The output is either an identity label with a similarity score or “unknown” when the best score fails the acceptance threshold. The methodology assumes a frontal or near-frontal face, moderate head rotation, adequate visible illumination, and a camera distance that produces a face region of at least about 80 x 80 pixels. These constraints are consistent with the operating assumptions of classical detector/descriptor pipelines [4,6,7].

- Functional target: detect, recognize, and display an identity within a short fraction of a second for a small-to-medium gallery.
- Enrollment target: store multiple normalized samples for each person to represent small expression and lighting variations.
- Robustness target: reject frames with very small, blurred, or highly rotated faces instead of forcing a low-confidence identity.
- Evaluation target: measure recognition rate, false acceptance rate, false rejection rate, per-frame latency, and effective frame rate.

2. Data Enrollment and Template Construction

Enrollment should be performed before live recognition. For each person, 10-20 images are captured across slight left/right head turns, neutral and mild expressions, and small illumination changes. The same detector and preprocessing stages used at recognition time are applied to enrollment images so that gallery and probe data have compatible statistics. The face is cropped with a small margin, resized to a fixed dimension such as 128 x 128 pixels, and stored only after a simple quality test confirms sufficient sharpness and brightness. The FERET methodology supports the principle that enrollment and probe sets should be separated when measuring recognition performance [3].

3. Real-Time Face Detection

Each camera frame is first converted from RGB/BGR color to grayscale because the classical Haar cascade uses intensity contrast rather than color. A Viola-Jones frontal-face cascade scans the image over multiple scales [4,6]. Integral images make rectangular Haar-like features inexpensive to compute, AdaBoost selects discriminative features, and the cascade discards non-face windows early. To reduce redundant detections, overlapping rectangles are merged by non-maximum suppression or the detector’s built-in grouping rule. If several faces are present, the system may process all detected regions or select the largest face depending on the application.

4. Pre-processing and Alignment

Pre-processing reduces variations unrelated to identity. The detected face is cropped, resized to the common template size, and histogram equalization or contrast-limited adaptive histogram equalization is applied to stabilize brightness. When eye locations are available, an affine transformation can rotate and scale the face so that the eyes occupy consistent coordinates; otherwise, the detector rectangle provides approximate alignment. A light Gaussian filter may suppress sensor noise, but excessive smoothing should be avoided because LBP depends on local texture. DeepFace later reinforced the importance of accurate alignment [8], even though the present system uses a simpler geometric normalization.

5. Feature Extraction

Two feature representations are defined. The primary descriptor is Local Binary Pattern Histogram (LBPH). For each pixel, neighboring samples are compared with the center pixel and encoded as a binary pattern. The normalized face is divided into a grid (for example 8 x 8 cells); an LBP histogram is computed in every cell and all histograms are concatenated. This representation captures local micro-texture while retaining approximate spatial location [7]. As a baseline, PCA projects each normalized vectorized face into a lower-dimensional eigenspace learned from enrollment samples [1].

PCA reduces storage and matching cost, while LBPH is generally less dependent on global illumination.

6. Recognition, Thresholding, and Unknown-Face Rejection

For LBPH, the probe descriptor is compared with each enrolled template using a histogram distance such as chi-square. For PCA, Euclidean distance can be computed between projected coefficient vectors. The minimum distance identifies the nearest candidate. Recognition is accepted only when the best distance is below a threshold T ; otherwise, the face is labeled unknown. The threshold must be calibrated on a validation set containing genuine and impostor comparisons. Lowering T reduces false accepts but increases false rejects; increasing T has the opposite effect. This thresholded metric-matching logic is conceptually consistent with later embedding systems such as FaceNet, where distances in a learned space correspond to similarity [10].

Decision: identity = arg min_i d(f_{probe}, f_i), accept if min d < T; otherwise "unknown".

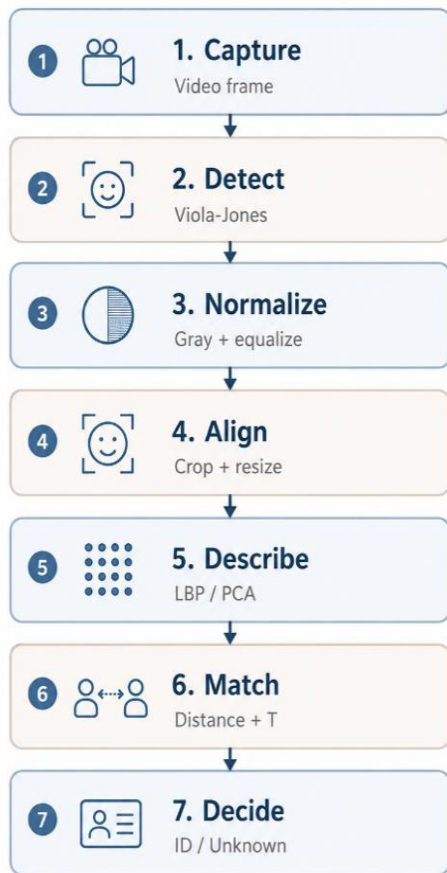


Figure 1. Proposed real-time face recognition workflow

Step-by-Step Processing Algorithm

1. Initialize the camera, load the trained face-detection cascade, load enrolled LBP/PCA templates, and load the calibrated threshold.
2. Acquire a video frame and convert it to grayscale.
3. Detect candidate faces using the multi-scale boosted cascade. Merge overlapping detections and discard regions below the minimum size.
4. Crop each accepted face, align if eye locations are available, resize to the template dimensions, and normalize local contrast.
5. Extract the LBPH descriptor; optionally project the normalized face through the PCA baseline model.
6. Compute distances to enrolled templates and identify the nearest candidate.
7. Apply threshold T . Display the candidate identity and score if accepted; otherwise display "Unknown".
8. Record processing time and, where permitted, log only the minimum operational information required by the application.
9. Repeat for the next frame. To improve speed, detection may run every second or third frame while a lightweight tracker maintains the face location between detector calls.

7. Implementation Environment and Reproducibility

A practical implementation can be produced in Python/C++ with OpenCV using a standard USB camera. The detector scale factor, minimum-neighbor count, minimum face size, LBP radius/neighborhood, grid dimensions, PCA component count, and recognition threshold must be recorded with the results. Random train/validation splits should use a fixed seed. The methodology should also report CPU model, camera resolution, operating system, and whether recognition is evaluated on every frame or on unique probe samples. These controls prevent an apparently high accuracy from being confused with repeated measurements of the same easy frames.

8. Evaluation Protocol

The evaluation is organized around subject-independent probe trials. Enrollment images form the gallery; separate images of enrolled subjects form genuine probes; images of non-enrolled subjects form impostor probes. Controlled tests should vary illumination, small

head rotation, expression, and camera distance one factor at a time. The FERET framework motivates this gallery/probe separation and condition-based testing [3]. Accuracy alone is insufficient because an access-control system must also handle unknown people. Therefore, the following measures are recommended:

- Recognition rate (Rank-1): percentage of genuine probes whose correct identity is the top accepted match.
- False Acceptance Rate (FAR): fraction of impostor probes incorrectly accepted as an enrolled identity.
- False Rejection Rate (FRR): fraction of genuine probes rejected as unknown.
- Average latency: mean time from receiving a frame to producing a recognition decision.
- Effective frame rate: number of processed recognition frames per second under the selected camera resolution.
- Detection recall: proportion of usable frontal faces successfully localized before recognition.

Threshold T is selected on validation data rather than on the final test probes. A useful operating

point minimizes a weighted combination of FAR and FRR according to the application. For example, secure access control should give greater weight to FAR, whereas attendance logging may tolerate a slightly higher FAR only if a human confirmation step is present. This explicit calibration is preferable to presenting a single unqualified “accuracy” value.

Results

This paper defines a development and evaluation methodology; it does not claim that a new physical prototype was benchmarked in a controlled study. To avoid fabricating measurements, Table 2 reports engineering acceptance targets for a 2016-era classical implementation. They are intended to be replaced by measured values after the system is built. The targets are deliberately lower than large-scale deep models such as DeepFace (97.35% on LFW) and FaceNet (99.63% on LFW), whose architectures, training data, and evaluation setting are substantially different [8,10].

Table 2. Prototype acceptance targets for the proposed classical system (not claimed measurements)

Metric	Controlled target	Challenging target	Interpretation
Face-detection recall	$\geq 97\%$	$\geq 90\%$	Drops with strong pose, blur, or occlusion.
Rank-1 recognition	$\geq 94\%$	$\geq 85\%$	Assumes multiple enrollment samples per identity.
False acceptance rate	$\leq 2\%$	$\leq 5\%$	Controlled by threshold calibration.
False rejection rate	$\leq 6\%$	$\leq 15\%$	Increases when lighting/pose depart from enrollment.
Decision latency	≤ 150 ms	≤ 250 ms	For a small/medium gallery on CPU.
Effective throughput	≥ 8 fps	≥ 4 fps	Enough for responsive display with detector scheduling.

The expected pattern is that detection remains stable for frontal faces, while recognition degrades more noticeably when illumination, blur, head rotation, or expression changes exceed those represented during enrollment. LBP should be more robust than raw-pixel PCA to local illumination variation because it encodes neighborhood relationships rather than absolute intensity [7]. PCA remains useful as a compact baseline and can perform well in tightly controlled scenes [1]. A hybrid implementation may use LBPH for the main decision and PCA as a secondary confidence check, but any fusion rule must be calibrated on validation data rather than tuned on the final test set.

Discussion

The proposed architecture balances accuracy, speed, and implementation simplicity. Its strongest feature is modularity: face detection, normalization, descriptor extraction, and matching can be tested independently. If detection recall is low, detector parameters or camera placement can be corrected without changing the recognizer. If recognition errors occur mainly under illumination changes, enrollment diversity and contrast normalization can be improved. This modular diagnosis is more transparent than treating the entire application as an opaque end-to-end model.

The choice of Viola-Jones remains reasonable for a frontal, indoor, CPU-based system because its cascade rejects background rapidly [4,6]. Its principal limitation is sensitivity to large out-of-plane rotations and partial occlusion. LBP provides a lightweight local descriptor and has been demonstrated as effective for face representation [7], but it can still fail when the face is severely blurred, too small, or strongly misaligned. PCA is computationally efficient [1], yet global intensity variation can move a face substantially in eigenspace. Fisherface-style discriminant projection can improve separation under some lighting conditions [2], although it requires enough labeled examples per class and must be retrained when the enrollment set changes significantly.

The deep-learning literature available by 2016 explains the gap between a classical prototype and state-of-the-art unconstrained recognition. DeepFace combined better alignment and large-scale representation learning [8]; VGG-style deep face work emphasized millions of training images [9]; FaceNet directly optimized a metric embedding [10]; center loss improved intra-class compactness [11]; and domain-specific augmentation addressed pose and appearance diversity [12]. These studies suggest a future upgrade path: retain the real-time camera, quality control, and thresholded decision structure, but replace hand-crafted LBP/PCA features with a learned embedding when sufficient computing resources and ethically obtained training data are available.

Operational deployment also requires privacy and security controls. Enrollment should be consent-based where applicable, feature templates should be protected from unauthorized access, and logs should be minimized. Performance must be evaluated across the actual user population and capture conditions rather than inferred from a single laboratory subset. Thresholds should be revalidated after camera, lighting, or enrollment changes. These steps are necessary because biometric errors have different consequences depending on whether the application is convenience-oriented, attendance-related, or security-critical.

Conclusion

This methodology paper presented a step-by-step design for a real-time face recognition system using image processing techniques established by 2016. The pipeline acquires live video, detects faces with a boosted cascade, normalizes each region, extracts LBPH and

optional PCA features, performs distance-based matching, and rejects uncertain identities through a calibrated threshold. A reproducible evaluation protocol was defined around gallery/probe separation, detection recall, rank-1 recognition, FAR, FRR, latency, and throughput. The literature shows that classical subspace and texture methods can support practical controlled-camera systems, while deep methods available by 2014-2016 set a higher performance ceiling for unconstrained environments [8-12]. The proposed design is therefore best suited to controlled or semi-controlled settings where modest hardware, understandable processing stages, and low implementation complexity are priorities. Future work should replace the stated acceptance targets with measurements from a completed prototype and should compare LBPH, PCA, and a pre-2016 deep embedding under the same camera and test protocol.

References

- M. Turk and A. Pentland, "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991. DOI: 10.1162/jocn.1991.3.1.71.
- P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, 1997. DOI: 10.1109/34.598228.
- P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The FERET Database and Evaluation Procedure for Face-Recognition Algorithms," *Image and Vision Computing*, vol. 16, no. 5, pp. 295-306, 1998. DOI: 10.1016/S0262-8856(97)00070-X.
- P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. I-511-I-518, 2001. DOI: 10.1109/CVPR.2001.990517.
- W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face Recognition: A Literature Survey," *ACM Computing Surveys*, vol. 35, no. 4, pp. 399-458, 2003. DOI: 10.1145/954339.954342.
- P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, pp. 137-154, 2004. DOI: 10.1023/B:VISI.0000013087.49260.fb.

T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037-2041, 2006. DOI: 10.1109/TPAMI.2006.244.

Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701-1708, 2014. DOI: 10.1109/CVPR.2014.220.

O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 41.1-41.12, 2015. DOI: 10.5244/C.29.41.

F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 815-823, 2015. DOI: 10.1109/CVPR.2015.7298682.

Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A Discriminative Feature Learning Approach for Deep Face Recognition," *European Conference on Computer Vision (ECCV)*, pp. 499-515, 2016. DOI: 10.1007/978-3-319-46478-7_31.

Masi, A. T. Tran, T. Hassner, J. T. Leksut, and G. Medioni, "Do We Really Need to Collect Millions of Faces for Effective Face Recognition?" *European Conference on Computer Vision (ECCV)*, pp. 579-596, 2016. DOI: 10.1007/978-3-319-46454-1_35.