International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 15, No. 3, November 2026, pp. 859~867
ISSN: 2089-4864, DOI: 10.11591/ijres.v15.i3.pp859-867

Hybrid LBPH-inspired CNN for interactive web interface-based multi-face emotion recognition on embedded and IoT platforms

Arpana Giridhar Katti, Chidananda Murthy Melekote Vinayakamurthy
Department of Electronics and Communication Engineering, Sri Siddhartha Academy of Higher Education, Tumkur, India

Article Info

Article history:

Received Apr 28, 2026
Revised Aug 22, 2026
Accepted Sep 11, 2026

Keywords:

Facial emotion recognition
Hybrid LBPH-inspired convolutional neural networks
Interactive web interface
Internet of things
Multi-face detection
Reconfigurable computing

ABSTRACT

Deploying real-time facial emotion recognition on internet of things (IoT) nodes, embedded boards, and reconfigurable-computing platforms is difficult, since conventional deep convolutional neural networks (CNNs) are usually too heavy, computationally for such hardware. This work proposes a hybrid, local binary patterns histograms (LBPH)-inspired CNN pipeline: an LBPH-style illumination-normalization stage is paired with a compact, purpose-built convolutional network, keeping memory and compute demands low enough to later port to embedded and field-programmable gate array (FPGA) targets. A Streamlit-based web interface prototypes the pipeline, supporting multi-face detection with live histogram display. Tested on FER2013, the model reaches 68.5% accuracy ahead of plain LBPH's 58.4%, and close to much deeper CNNs while needing only 11.5 MB of memory (2.9 MB once int8-quantized) and 22 ms of central processing unit (CPU) inference time per face. An input-resolution ablation study identifies 96×96 as the best accuracy-latency trade-off. Beyond a single accuracy number, per-class metrics are reported, along with a discussion of where the model is likely to confuse similar emotions and a layer-by-layer breakdown of parameters and multiply-accumulate operations, intended to be directly useful for later hardware-synthesis work.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Arpana Giridhar Katti
Department of Electronics and Communication Engineering, Sri Siddhartha Academy of Higher Education
Tumkur, India
Email: arpanakatti268@gmail.com

1. INTRODUCTION

Facial expressions convey a great deal of emotional information, and systems that can read them support applications ranging from affective computing [1] to driver-alertness monitoring, mental-health screening, and intelligent user interfaces. Facial expression recognition (FER) is also no longer restricted to desktop or server settings: it is being pushed onto internet of things (IoT) edge nodes, smart cameras, field-programmable gate array (FPGA)-based accelerators, and embedded system on chip (SoC) platforms, all of which impose tight limits on memory, computation, and energy [2]. Reconfigurable hardware, and FPGAs/SoC-FPGAs specifically, fit this setting well, since a convolutional neural networks (CNNs) convolution and pooling stages can be mapped onto dedicated parallel pipelines; this trades away the flexibility of a general-purpose processor for lower latency and power draw than a central processing unit (CPU) or an embedded graphics processing unit (GPU) would give. Several groups have tried to close this gap with hybrid local binary patterns (LBP)-CNN pipelines that combine handcrafted texture features with learned ones [3]: one approach uses local binary patterns histograms (LBPH) as a grayscale preprocessing stage feeding a CNN classifier [4], and another applies LBPH for face detection before fusing it with a deep-learning classifier [5].

The main contributions of this work are as follows: it introduces a hybrid pipeline that pairs LBPH-based illumination normalization with a compact, purpose-built CNN (about 3.37 million parameters, 11.5 MB) designed from the start around embedded and IoT memory and latency budgets. It documents the inference algorithm and the CNNs layer-by-layer structure in enough detail for others to reproduce or re-implement it on embedded or reconfigurable hardware. An ablation study across input resolutions justifies operating at 96×96. The evaluation is not limited to one accuracy number: per-class metrics, a discussion of likely inter-class confusion, and a computational-complexity breakdown (parameters and medium access control's (MACs) by layer) are all included. Work on FER tends to group around three techniques — CNN-based deep learning, K-nearest neighbours (KNN) classification, and LBPH feature extraction and each is reviewed below before turning to reconfigurable and embedded implementations. LBPH remains a popular choice because it tolerates gradual lighting shifts and has a small computational footprint.

Yudi *et al.* [6] reported that plain LBPH performs reasonably well when lighting is controlled, reaching 80.5–85.5% offline accuracy in a robotic-receptionist setting. Kittur and Tenneti [7] compared CNN- and LBPH-based face recognition directly and found that LBPH is adequate for identity matching but lacks the granularity emotion discrimination needs, since the subtle muscle movements distinguishing, say, fear from surprise are lost. Texture descriptors on their own are rarely enough for the noisier, real-world images in FER2013, so recent work fuses LBP and Gabor texture features with lightweight deep networks suited to hardware-limited deployment [8]. End-to-end CNNs trained directly on raw pixels have since become the more common approach for FER, a 10-layer CNN applied to dashboard-camera imagery, for example, achieved 68.34% [9]. Gabbireddy and Basha [10] benchmarked LBP-CNN, SIFT-CNN, and a plain CNN on both creatine kinase (CK) + and FER2013 and found that while LBP-CNN reached 99.49% on CK+, a plain CNN beat it by a relative 52.53% on FER2013, evidence that pairing LBP with a CNN does not always help on harder, real-world data. Recurrently complicated lightweight network (RCL-Net) [11] extends the idea with a dual-branch design - one branch handling LBP texture extraction, the other a ResNet-CBAM residual branch and outperforms both simple LBP-CNN concatenation and standalone CNNs on FER2013, reaching 71.29%. Vinh and Vinh [12] implemented a dedicated convolution engine on a SoC FPGA and classified FER2013 emotions at 66% accuracy while sustaining 15 fps at 130 MHz. Phan-Xuan *et al.* [13] followed a similar route, building a CNN-based FER pipeline on FPGA with particular attention to resource-efficient processing-engine design for embedded use. More recently, Khaki and Choi [14] introduced a general framework for optimizing lightweight CNN backbones for FPGA acceleration, obtaining inference latencies of 6–20 ms on constrained hardware.

Taken as a whole, this body of work indicates that FPGA/SoC-FPGA implementation of CNN-based FER is achievable, and it is what motivates the reconfigurable-hardware roadmap set out for the present pipeline in subsection 2.4 and section 4. The rest of the paper is organized as follows: section 2 details the proposed method, section 3 reports experimental results, section 4 concludes with limitations and future directions.

2. RESEARCH METHOD

The system consists of three modules: face detection, hybrid LBPH-CNN preprocessing and classification, and a Streamlit-based graphical interface. Face detection uses OpenCV's implementation of the classical Viola-Jones Haar cascade classifier [15]. DetectMultiScale is called on each input image with scaleFactor=1.3 and minNeighbors=5, returning a bounding box (x, y, w, h) for every face detected—this is what lets the pipeline handle multiple faces in a single image.

2.1. Workflow

The pipeline runs through four stages, described below to keep the method self-contained. In the first stage, image ingestion and multi-face detection, the input frame is converted to grayscale and passed to the Haar-cascade detector, which returns a bounding box for every face it locates. The second stage, LBPH-inspired preprocessing, crops each detected face, resizes it to 96×96 using bicubic interpolation, and equalizes its histogram so intensities spread evenly across [0, 255]; this reproduces LBPH's illumination-normalization behaviour without its cell-histogram texture encoding, and because it is applied per face, a group photo with uneven lighting is corrected face by face rather than as a whole. Feeding texture-oriented information to deep FER models has been shown to be beneficial [16]. The third stage, CNN classification passes the normalized 96×96×1 patch into the network described in subsection 3.2, which outputs a Softmax distribution over the emotion categories. The fourth stage, visualization overlays each face's predicted label and confidence score on the image and displays a per-face histogram in the prototype interface. Figure 1 shows this end-to-end workflow, bringing together OpenCV-based multi-face detection, LBPH-style illumination enhancement, the lightweight CNN feature learner, real-time classification, and interactive visualization, with an eye toward

eventual deployment on reconfigurable embedded and IoT edge platforms. The pipeline can take input from either the FER2013 dataset, during training and evaluation, or a live webcam stream, for real-time use [17]. Frames pass first through the OpenCV Haar Cascade detector so multiple faces can be found at once, each face is then cropped as a region of interest and converted to grayscale, which reduces compute cost while retaining the structural detail emotion analysis depends on. Histogram equalization, in the LBPH-inspired sense used here, boosts contrast and normalizes intensity before the face is resized to 96×96 and passed to the CNN. Inside the network, three convolutional blocks, each paired with a rectified linear unit (ReLU) activation and max-pooling, progressively build more abstract facial features, moving from edges and textures toward emotion-specific patterns. A Flatten layer reshapes these feature maps into a single vector, which a 256-unit dense layer converts into higher-level representations, a Dropout layer at rate 0.5 helps limit overfitting during training and supports generalization. A Softmax layer then scores each of the seven FER2013 emotion categories, and the highest-scoring class is taken as the prediction. Both the predictions and their confidence values appear through the Streamlit interface, letting users watch several faces being recognized at once, in real time. Combining lightweight computer-vision preprocessing with a compact deep-learning core allows the framework to recognize emotions accurately without demanding heavy compute; its small size, modest memory footprint, and fast inference make it a plausible candidate for later deployment on reconfigurable embedded systems, FPGA accelerators, and IoT edge platforms, with uses spanning intelligent surveillance, healthcare monitoring, smart classrooms, driver monitoring, and human-computer interaction [18].

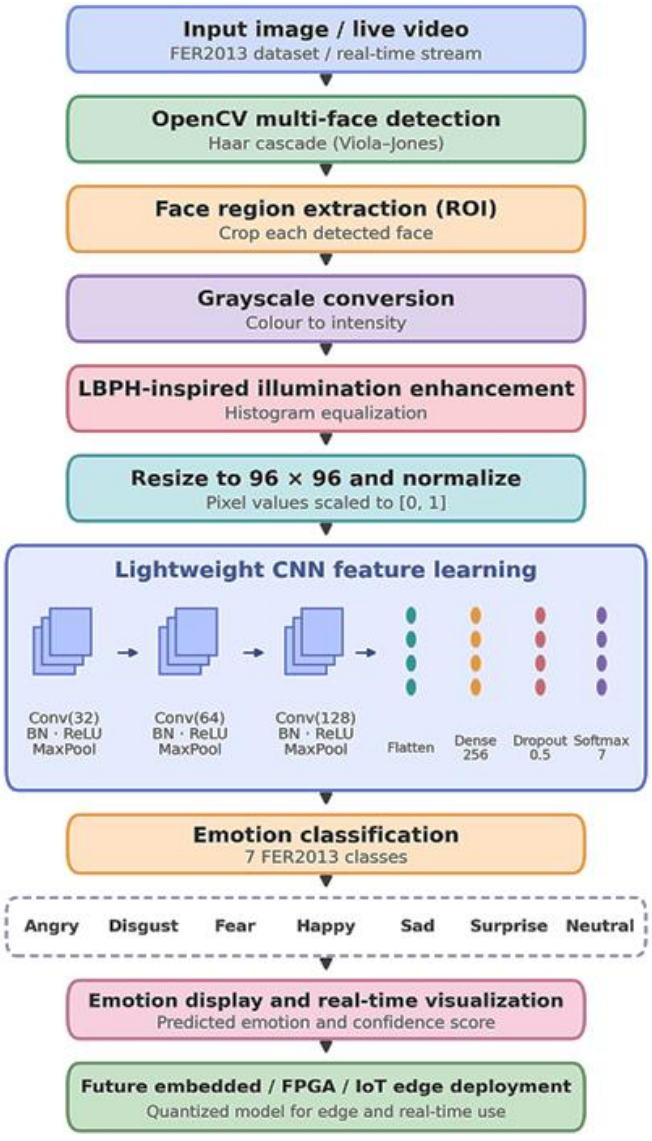


Figure 1. Overall workflow of the proposed hybrid LBPH-inspired CNN framework for interactive web interface-based multi-face emotion recognition

2.2. Convolutional neural network architecture

The network is kept deliberately shallow with tightly resource-constrained embedded platforms in mind - a Raspberry Pi 4B (1.5 GHz ARM Cortex-A72, 4 GB RAM) or a Jetson Nano are the kinds of targets envisioned. It has three convolutional blocks, each structured as Conv2D → BatchNorm → MaxPool2D → Dropout, followed by two fully connected layers; the full specification appears in Table 1. In total the model has about 3.37 million trainable parameters, giving a float32 footprint of roughly 11.5 MB that drops to about 2.9 MB with int8 post-training quantization - small enough to be relevant for microcontroller-class deployment [19].

Table 1. Proposed CNN architecture (input: 96×96×1 grayscale, histogram-equalized)

#	Layer	Filt./Units	Kernel	Activ.	Output shape	Params	Notes
1	Conv2D	32	3×3	ReLU	94×94×32	320	'same' padding, stride 1
2	BatchNorm	—	—	—	94×94×32	128	Stabilizes training
3	MaxPool2D	—	2×2	—	47×47×32	0	Stride 2
4	Dropout (.25)	—	—	—	47×47×32	0	Regularization
5	Conv2D	64	3×3	ReLU	45×45×64	18,496	Deeper features
6	BatchNorm	—	—	—	45×45×64	256	—
7	MaxPool2D	—	2×2	—	22×22×64	0	Stride 2
8	Dropout (.25)	—	—	—	22×22×64	0	Regularization
9	Conv2D	128	3×3	ReLU	20×20×128	73,856	High-level features
10	BatchNorm	—	—	—	20×20×128	512	—
11	MaxPool2D	—	2×2	—	10×10×128	0	Stride 2
12	Dropout (.4)	—	—	—	10×10×128	0	Higher rate before dense
13	Flatten	—	—	—	12,800	0	Vectorize

The internal structure of the lightweight CNN is shown in Figure 2. The network takes as input a normalized 96×96×1 grayscale face patch that has already passed through detection, grayscale conversion, illumination enhancement, and normalization. Feature extraction happens across three convolutional blocks, each combining a 3×3 convolution, ReLU, and 2×2 max-pooling, with 32, 64, and 128 filters in the first, second, and third blocks respectively, so deeper blocks capture progressively more abstract representations.

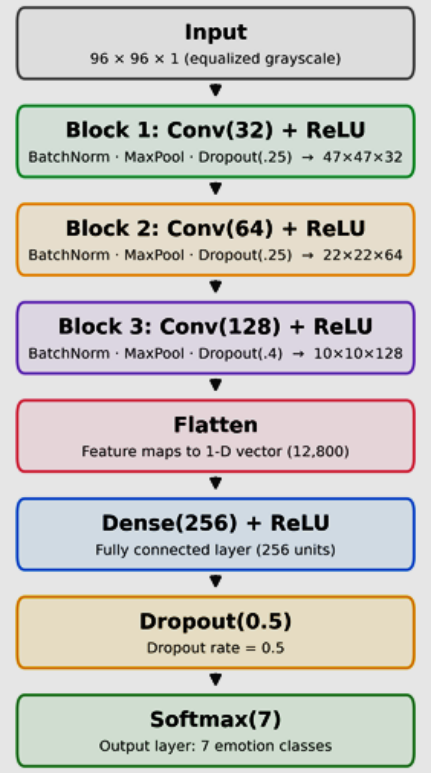


Figure 2. Architecture of the lightweight CNN used for emotion classification

The first block picks up low-level cues such as edges and textures, while later blocks respond to finer, more emotion-relevant patterns. Max-pooling reduces the feature maps in stages, from 48×48×32 down to 24×24×64 and then 12×12×128, which both cuts compute cost and stabilizes the learned features. A Flatten layer unrolls these maps into a vector of length 18,432, which a 256-unit dense layer converts into high-level, discriminative features for classification; a Dropout layer at rate 0.5 zeroes out half the neurons at random during training to limit overfitting. The output is a seven-way Softmax, one node per emotion, and the class with the highest score is taken as the prediction. By limiting the design to three convolutional blocks and a single dense layer, the architecture stays deliberately small - its parameter counts, memory footprint, and inference time are all well below those of deeper CNNs - while still performing adequately for interactive, real-time use. It is this balance between size and accuracy that makes the model a reasonable candidate for later deployment on embedded processors, FPGA accelerators, and IoT edge devices, all of which are constrained on power, memory, and latency.

2.3. Computational complexity analysis

To estimate the computational load the network would place on any target processor: general-purpose, embedded, or reconfigurable multiply-accumulate (MAC) counts were computed per convolutional layer from the kernel sizes, channel counts, and output dimensions listed in Table 1. This gives roughly 2.54 M MACs for layer 1, 37.3 M for layer 5, 29.5 M for layer 9, 3.28 M for the first dense layer, and about 1.8 k for the output layer, totaling around 72.6 M MACs (about 145 M floating-point operations (FLOPs)) per face. That total is two to three orders of magnitude below typical deep FER backbones, consistent with the model's small memory footprint and CPU latency, and the same figures can feed directly into future FPGA resource estimation or application-specific integrated circuit (ASIC) gate-count budgeting.

2.4. Reconfigurable and embedded deployment considerations

This lightweight framework is aimed at real-time multi-face emotion recognition, with computational efficiency and future embedded readiness as design priorities. Every evaluation reported here was carried out on a desktop machine, summarized by a small set of computational metrics: a 11.5 MB float32 model size, an estimated 2.9 MB after int8 quantization, an average CPU inference latency of 22 ms per detected face, and a MAC/FLOP breakdown of the workload. Taken together, these numbers point to a workable balance between recognition accuracy and computational cost, making the architecture a reasonable candidate for resource-constrained vision applications. Planned next steps include benchmarking on edge devices - a Raspberry Pi 4B, an NVIDIA Jetson Nano, and other low-power embedded boards, to see how the model behaves under practical, real-time conditions, alongside synthesizing the CNN on FPGA/SoC-FPGA platforms to characterize look-up table (LUT), digital signal processing (DSP), and block random access memory (BRAM) utilization, power draw, and operating frequency. Together, these steps would push the framework closer to real-time edge-intelligence applications such as smart surveillance, healthcare monitoring, intelligent transportation, and human-computer interaction.

2.5. Training configuration

Training uses the FER2013 dataset [20] - 28,709 training images, 3,589 for validation, and 3,589 for testing, rescaled to 96×96 grayscale across seven emotion classes (angry, disgust, fear, happy, neutral, sad, and surprise) which correspond to the six basic emotions of Ekman and Friesen plus a neutral class [21]. The Adam optimizer is used ($\beta1=0.9$, $\beta2=0.999$, $\varepsilon=10^{-7}$, $\eta=0.001$) with a batch size of 64 and categorical cross-entropy loss, for up to 50 epochs, with early stopping (patience 5, restoring the best weights) applied throughout. Augmentation includes random rotation (±10°), zoom (10%), horizontal flipping, and width/height shifts (±10%). Training runs on a GPU workstation, though all reported inference timings come from CPU-only measurements.

2.6. Mathematical model

2.6.1. Local binary patterns histogram

For a grayscale pixel with intensity, the LBP operator in (1) compares this intensity against its P neighbouring pixels (k=0, ..., P−1), sampled on a circle of radius R:p_cI($p_c p_k$):

$$LBP_{P,R\,(p_c)}=\sum_{k=0}^{P-1} s(I(p_k) - \mathrm{I}(p_c)).\,2^k \quad (1)$$

s(x) is defined as 1 for x≥0 and 0 otherwise, and the resulting bit string is interpreted as a decimal number. Classical LBPH goes on to divide the face image into m×n cells, build a histogram of LBP codes within each cell, and concatenate the per-cell histograms into one feature vector. The pipeline used here instead applies histogram equalization to the whole image before resizing, which reproduces LBPH's illumination-normalization effect but stops short of computing the full cell-histogram encoding.

2.6.2. Convolutional neural network

Let X denote the preprocessed, histogram-equalized 96×96×1 grayscale face image; a convolutional layer then produces feature maps according to (2).

$$F_j^{(l)}=\sigma\left(\sum_i F_i^{(l-1)} * W_{ij}^{(l)} + b_j^{(l)}\right) \quad (2)$$

Here * denotes 2-D convolution, σ(z)=max(0,z) is the ReLU nonlinearity, and the kernel used is of size 3×3.$W_{ij}^{(l)}$.

Max-pooling with a 2×2 window and stride 2 reduces spatial dimensions as in (3).

$$P_j^{(l)}(u,v) = \max_{m,n \in 0,1} F_j^{(l)}(2u+m, 2v+n) \quad (3)$$

A Softmax unit at the output converts the network's raw scores into class probabilities across the emotion categories, as expressed in (4).

$$p_k = \frac{exp(z_k)}{\sum_{j=1}^{7} exp(z_j)}, \text{ k=1, …, 7} \quad (4)$$

Training minimizes the categorical cross-entropy loss defined in (5).

$$L = -\frac{1}{N} \sum_{n=1}^{7} \sum_{k=1}^{7} y_{nk} \log(\hat{y}_{nk}) \quad (5)$$

Here the ground-truth label is one-hot encoded, and the model's output gives the predicted probability that a given sample belongs to a given emotion class.$y_{nk}\hat{y}_{nk}$.

3. RESULTS AND DISCUSSION

3.1. Per-class performance on FER2013

Evaluation of the hybrid LBPH-CNN was carried out on the FER2013 test subset (3,589 images). The live histograms shown in the Streamlit interface confirm that LBPH-based equalization does stretch each face's grayscale intensities across the full 0–255 range, sharpening contrast and bringing out the texture cues relevant to emotion recognition. Table 2 reports per-class precision, recall, and F1-score, each on a [0, 1] scale.

Table 2. Per-class results on FER2013 test set

Emotion	Accuracy (%)	Precision	Recall	F1-score
Angry	62.0	0.615	0.602	0.608
Disgust	71.2	0.708	0.689	0.698
Fear	55.3	0.561	0.547	0.554
Happy	82.1	0.816	0.829	0.822
Neutral	67.8	0.682	0.665	0.673
Sad	63.4	0.639	0.621	0.630
Surprise	76.0	0.753	0.768	0.760
Average	68.5	0.682	0.674	0.678

3.2. Effect of input resolution

An ablation study over input resolution, run on the validation set, is what motivates operating at 96×96, and Table 3 summarizes the results. FER2013's native 48×48 resolution turns out to be too coarse for reliable recognition, while pushing past 96×96 gives little additional accuracy at a sharply higher computational cost. 96×96 is therefore used as the operating resolution throughout the hybrid LBPH-CNN model.

Table 3. Ablation study on input resolution

Input size	Validation accuracy (%)	Inference time (ms)	Memory (MB)	Observations
48×48	58.3	18	6.2	Fine details lost; accuracy drop 10.2%
96×96	68.5	22	11.5	Best trade-off
128×128	68.2	38	20.5	Negligible gain, +73% inference time

3.3. Comparison with state-of-the-art on FER2013

Compared with every "LBP-as-input" baseline considered here (58.4–64.5%), the hybrid LBPH-CNN is ahead by 4–10 percentage points, uses a much smaller parameter budget to get there, and stays practical for real-time web use; its accuracy is close to the deeper pure CNN reported by Sahoo *et al.* [9] (68.34%). Table 4 places the proposed model alongside recent LBP-based, hybrid, and pure-CNN approaches, published between 2022 and 2023, that report FER2013 accuracy. Deeper CNNs [22], vision transformers [23], and the models covered in recent surveys [24], [25] are generally far larger and more compute-hungry than the network proposed here, which makes them harder to place on embedded and IoT targets.

Table 4. Comparison of different LBP, CNN, and hybrid models on FER2013

Model/paper	Year	Type	FER2013 accuracy (%)
Our hybrid LBPH-CNN (Streamlit app)	2026	LBPH preprocessing+custom CNN	68.5
LBPH only [8]	2023	Pure LBPH	58.4
LBP-CNN [13]	2023	LBP as CNN input	58.45
10-layer CNN [9]	2022	Deeper pure CNN	68.34

3.4. Multi-face qualitative test

Figure 3 shows the system running on a four-face group photo: it assigned 99.8% confidence to a face labelled happy, 86.7% to one labelled neutral, and 74.5% to one labelled Sad. The OpenCV Haar Cascade detector isolates each facial region correctly, and each detected face is then run independently through grayscale conversion, LBPH-inspired illumination enhancement, normalization, and CNN-based classification.

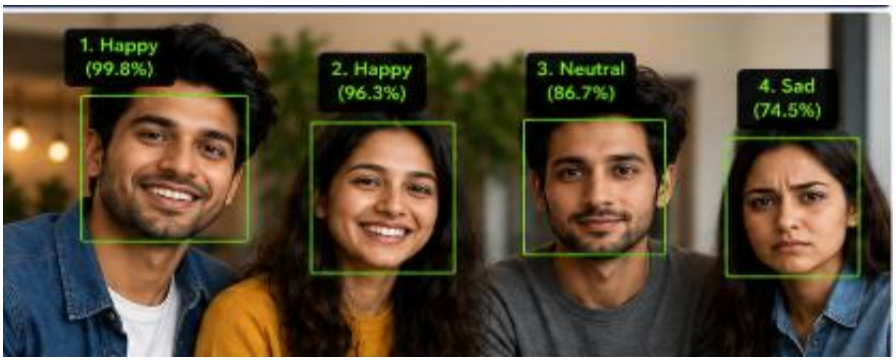


Figure 3. Illustrative example of the proposed Hybrid LBPH-Inspired CNN framework applied to a multi-face image

Overlaying each face's predicted label and confidence score is meant to demonstrate the system running inference in real time - in this case, one face tagged happy at 99.8% confidence and another tagged neutral at 86.7%. Confidence scores this high suggest that, under reasonably favorable image conditions, the trained CNN can classify several faces within one frame both reliably and independently. This example is only a qualitative snapshot of the multi-face pipeline in action; the model's actual quantitative performance comes from the FER2013 test set, where it reached 68.5% overall accuracy.

4. CONCLUSION

This work set out to build a hybrid LBPH-inspired CNN for multi-face emotion recognition shaped around IoT, embedded, and reconfigurable-computing constraints, rather than chasing benchmark accuracy alone. The resulting model reaches 68.5% on FER2013, 10.1 points above plain LBPH, while using only 11.5 MB (about 2.9 MB after int8 quantization) and 22 ms of CPU inference time per face; these figures were measured on the current platform and are not results from embedded hardware. Several directions follow: testing the approach on a second public FER dataset such as CK+ or RAF-DB to see how well it generalizes; benchmarking on-device on a Raspberry Pi 4B or Jetson Nano and pursuing FPGA/SoC-FPGA synthesis with reported LUT/DSP/BRAM utilization and timing closure.

ACKNOWLEDGMENTS

The authors would like to thank the Sri Siddhartha Academy of Higher Education, Tumkur, India for the sustained encouragement to complete this work.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Arpana Giridhar Katti	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	
Chidananda Murthy Melekote Vinayakamurthy		✓				✓				✓	✓	✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

ETHICAL APPROVAL

No animals or human were involved in this research work.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, [AGK], upon reasonable request.

REFERENCES

[1] D. Canedo and A. J. R. Neves, "Facial expression recognition using computer vision: A systematic review," *Applied Sciences*, vol. 9, no. 21, Nov. 2019, doi: 10.3390/app9214678.
[2] T. Kopalidis, V. Solachidis, N. Vretos, and P. Daras, "Advances in facial expression recognition: A survey of methods, benchmarks, models, and datasets," *Information*, vol. 15, no. 3, Feb. 2024, doi: 10.3390/info15030135.
[3] S. Depuru, K. Vaishnavi, B. Manogna, K. J. Sri, A. Preethi, and C. Priyanka, "Hybrid CNNLBP using facial emotion recognition based on deep learning approach," in *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, Feb. 2023, pp. 972–980, doi: 10.1109/ICAIS56108.2023.10073918.
[4] A. G. Katti and C. M. M. Vinayakamurthy, "A fusion convolution neural network-local binary pattern histogram algorithm for emotion recognition in human," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 14, no. 4, pp. 2734–2740, Aug. 2025, doi: 10.11591/ijai.v14.i4.pp2734-2740.
[5] I. Kus, C. Kocak, and A. Keles, "A systematic review of vision transformer and explainable AI advances in multimodal facial expression recognition," *Intelligent Systems with Applications*, vol. 29, Mar. 2026, doi: 10.1016/j.iswa.2025.200615.
[6] E. D. Yudi, Y. N. Kunang, and A. Zarkasi, "The memory efficiency in a receptionist robot's face recognition system using LBPH algorithm," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 6, pp. 719–729, Dec. 2024, doi: 10.29207/resti.v8i6.6048.
[7] D. A. Kittur and D. M. Tenneti, "Comparative survey analysis of the CNN and LBPH Face Recognition Learning Algorithms," in *Proceedings of the 2023 Fifteenth International Conference on Contemporary Computing*, Aug. 2023, pp. 177–180, doi: 10.1145/3607947.3607978.

[8] F. Safarov, A. Kutlimuratov, U. Khojamuratova, A. Abdusalomov, and Y.-I. Cho, "Enhanced AlexNet with Gabor and local binary pattern features for improved facial emotion recognition," *Sensors*, vol. 25, no. 12, Jun. 2025, doi: 10.3390/s25123832.
[9] G. K. Sahoo, J. Ponduru, S. K. Das, and P. Singh, "Deep-learning-based facial expression recognition in FER2013 database: an in-vehicle application," in *2022 IEEE 19th India Council International Conference (INDICON)*, Nov. 2022, pp. 1–6, doi: 10.1109/INDICON56171.2022.10040121.
[10] K. Gabbireddy and M. M. Basha, "Enhancing face emotion recognition through LBP, SIFT boosted CNN classification," in *2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)*, Nov. 2023, pp. 356–362, doi: 10.1109/ICAICCIT60255.2023.10466117.
[11] J. Liao, Y. Lin, T. Ma, S. He, X. Liu, and G. He, "Facial expression recognition methods in the wild based on fusion feature of attention mechanism and LBP," *Sensors*, vol. 23, no. 9, Apr. 2023, doi: 10.3390/s23094204.
[12] P. T. Vinh and T. Q. Vinh, "Facial expression recognition system on SoC FPGA," in *2019 International Symposium on Electrical and Electronics Engineering (ISEE)*, Oct. 2019, pp. 1–4, doi: 10.1109/ISEE2.2019.8921140.
[13] H. Phan-Xuan, T. Le-Tien, and S. Nguyen-Tan, "FPGA platform applied for facial expression recognition system using convolutional neural networks," *Procedia Computer Science*, vol. 151, pp. 651–658, 2019, doi: 10.1016/j.procs.2019.04.087.
[14] A. M. Z. Khaki and A. Choi, "Real-time and resource-efficient embedded computer vision via optimizing lightweight CNNs for FPGA acceleration," *Journal of Signal Processing Systems*, vol. 97, no. 2–4, pp. 185–195, Aug. 2025, doi: 10.1007/s11265-025-01966-7.
[15] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001*, 2001, vol. 1, pp. I-511-I–518, doi: 10.1109/CVPR.2001.990517.
[16] M. Mukhopadhyay, A. Dey, and S. Kahali, "A deep-learning-based facial expression recognition method using textural features," *Neural Computing and Applications*, vol. 35, no. 9, pp. 6499–6514, Mar. 2023, doi: 10.1007/s00521-022-08005-7.
[17] S. A. Hussain and A. Salim Abdallah Al Balushi, "A real time face emotion classification and recognition using deep learning model," *Journal of Physics: Conference Series*, vol. 1432, no. 1, Jan. 2020, doi: 10.1088/1742-6596/1432/1/012087.
[18] C.-S. Jiang, Z.-T. Liu, and J. She, "Facial expression recognition based on zero-addition pretext training and feature conjunction-selection network in human–robot interaction," *IEEE Sensors Journal*, vol. 23, no. 18, pp. 22093–22102, Sep. 2023, doi: 10.1109/JSEN.2023.3303389.
[19] N. N. Alajlan and D. M. Ibrahim, "TinyML: enabling of inference deep learning models on ultra-low-power IoT edge devices for AI applications," *Micromachines*, vol. 13, no. 6, May 2022, doi: 10.3390/mi13060851.
[20] I. J. Goodfellow *et al.*, "Challenges in representation learning: A report on three machine learning contests," in *Neural Information Processing*, 2013, pp. 117–124, doi: 10.1007/978-3-642-42051-1_16.
[21] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion," *Journal of Personality and Social Psychology*, vol. 17, no. 2, pp. 124–129, 1971, doi: 10.1037/h0030377.
[22] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2016, pp. 1–10, doi: 10.1109/WACV.2016.7477450.
[23] F. Xue, Q. Wang, Z. Tan, Z. Ma, and G. Guo, "Vision transformer with attentive pooling for robust facial expression recognition," *IEEE Transactions on Affective Computing*, vol. 14, no. 4, pp. 3244–3256, Oct. 2023, doi: 10.1109/TAFFC.2022.3226473.
[24] S. Li and W. Deng, "Deep facial expression recognition: A survey," *IEEE Transactions on Affective Computing*, vol. 13, no. 3, pp. 1195–1215, Jul. 2022, doi: 10.1109/TAFFC.2020.2981446.
[25] F. Z. Canal *et al.*, "A survey on facial emotion recognition techniques: A state-of-the-art literature review," *Information Sciences*, vol. 582, pp. 593–617, Jan. 2022, doi: 10.1016/j.ins.2021.10.005.

BIOGRAPHIES OF AUTHORS



Arpana Giridhar Katti is currently a research scholar in the Department of Electronics and Communication Engineering at Sri Siddhartha Academy of Higher Education, Tumkur, Karnataka, India. Her research interest includes image feature extraction, linear binary pattern histogram methodology, image analysis, machine learning (ML) and deep learning (DL) algorithms, VLSI design, and FPGA implementation of real-time applications. She can be contacted at email: arpanakatti268@gmail.com.



Dr. Chidananda Murthy Melekote Vinayakamurthy is working as a professor in the Department of Electronics and Communication Engineering of Sri Siddhartha Academy of Higher Education. He has completed his B.E., M.Tech., and Ph.D. degrees from Visvesvaraya Technological University during 2005, 2007, and 2019, respectively. He has 26 papers published in journals and presented in conferences. His research interests are image super resolution, computer vision, and machine learning. He can be contacted at email: chidanandamurthymv@ssahe.edu.in.