# FPGA-Based Real-Time Face Detection with PC-Assisted Recognition Using Tang Nano FPGA

PRAMODH G S | pramodshankar654@gmail.com
PRAVEENA S | praveenskumar900@gmail.com
GURUPRASAD C | guruprasadc391@gmail.com
**NIVEDITHA (Mentor) | nivedithabs@ewit.edu.in**

***Abstract*—*This paper presents the design of an FPGA-based real-time face-detection system in which a Tang Nano FPGA performs camera acquisition and initial image processing, while a personal computer performs face recognition and database matching. The FPGA pipeline comprises pixel acquisition, RGB processing, RGB-to-YCbCr conversion, skin-region thresholding, binary segmentation, noise filtering, candidate-region analysis, bounding-box generation, and face region-of-interest extraction. Detected face information is transferred to a PC through a suitable USB/UART-class interface. The PC application, implemented using Python and OpenCV, performs preprocessing, feature comparison against registered users, known/unknown classification, confidence display, and event logging. The proposed architecture separates deterministic pixel-level processing from computationally intensive recognition, providing a practical hardware-software co-design platform for real-time computer-vision education and prototyping.*

***Keywords:*** *FPGA, Tang Nano, face detection, face recognition, YCbCr, Verilog HDL, OpenCV, hardware acceleration, image processing.*

## 1. INTRODUCTION

Real-time face analysis is an important demonstration problem for digital image processing because it combines continuous sensor input, high data throughput, low-latency decision making, and a need for flexible recognition algorithms. In the proposed system, these requirements are divided between programmable logic and a general-purpose computer. The Tang Nano FPGA continuously receives camera pixels and performs the time-critical detection pipeline, whereas the PC performs face recognition and database operations.

The design is intended to demonstrate practical FPGA image processing rather than to move the complete recognition workload into hardware. The FPGA therefore identifies candidate face regions and generates their coordinates or region-of-interest data. The PC then executes the more computationally intensive recognition stage using Python, OpenCV, NumPy, and a local registered-face database. This partition preserves the parallelism and deterministic timing of FPGA logic while retaining software flexibility for recognition and user-interface functions.

## 2. SYSTEM OBJECTIVES

The main objectives are: (1) to develop a real-time camera-based face-detection system using a Tang Nano FPGA development board; (2) to implement hardware image-processing functions in Verilog HDL; (3) to transfer detected face information from the FPGA to a PC; and (4) to develop a monitoring and recognition application capable of identifying registered persons and indicating unknown persons.

## 3. PROPOSED ARCHITECTURE

The architecture consists of four major subsystems: camera input, Tang Nano FPGA processing, FPGA-to-PC communication, and PC-based face recognition. The end-to-end signal path begins with live image acquisition and terminates in a monitoring interface that displays detection and recognition status.

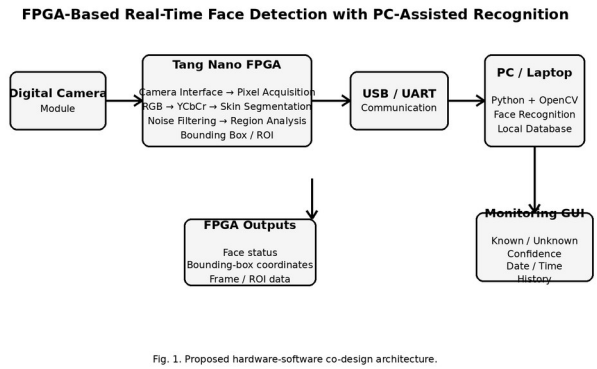



Fig. 1. Proposed FPGA-PC architecture.

### A. Camera Input and Pixel Acquisition

A digital camera module supplies pixel data together with timing information including pixel clock, horizontal synchronization, vertical synchronization, and frame synchronization. FPGA counters identify the current X and Y pixel coordinates, enabling later region localization and bounding-box calculation.

### B. FPGA Image-Processing Pipeline

Incoming RGB pixels are processed as a streaming pipeline. The proposed sequence is RGB extraction, color-space conversion, thresholding, binary image generation, noise removal, region detection, face-candidate analysis, and bounding-box generation. Pipelining allows several processing stages to operate concurrently on different pixels.

## 4. FPGA-BASED FACE DETECTION

### A. RGB-to-YCbCr Conversion

YCbCr separates luminance from chrominance. In the proposed design, the Y component represents luminance, whereas Cb and Cr represent blue- and red-difference chrominance. The chrominance channels are useful for skin-region segmentation because candidate skin pixels can be selected by predefined threshold conditions.

A fixed-point implementation is suitable for FPGA logic because multiplication and addition can be mapped to dedicated arithmetic resources or shift-add structures. The exact coefficients and bit widths should be selected according to the camera format, required numerical accuracy, and the resources available on the selected Tang Nano device.

### B. Segmentation and Noise Filtering

After color conversion, each pixel is classified as a candidate skin/face pixel or background pixel, producing a binary image. Isolated pixels and small unwanted regions caused by lighting variation or background objects are then suppressed using neighbourhood checks, binary filtering, or basic morphological operations.

### C. Region Analysis and Bounding Box

The region-analysis logic evaluates candidate clusters using their minimum and maximum X and Y coordinates, width, height, and area. Regions satisfying predefined face conditions are retained as possible faces. The FPGA then expresses the result as a bounding box (Xmin, Ymin, Xmax, Ymax). The same coordinates can be used to select a face region of interest.

## 5. FPGA-TO-PC COMMUNICATION

The Tang Nano FPGA communicates with the PC using a suitable interface supported by the selected board and prototype configuration. Candidate interfaces include USB-UART, UART, USB communication, or an SPI-to-USB bridge. Data transferred to the PC can include face-detected status, bounding-box coordinates, frame number, detection state, or face ROI data.

Transferring only a detected ROI or coordinates rather than an entire frame reduces communication bandwidth, PC processing load, and recognition latency. The final choice between coordinate-only transfer and ROI transfer depends on how the live camera stream is made available to the PC application.

## 6. PC-ASSISTED FACE RECOGNITION

### A. Registration and Database

The PC application provides a registration facility for storing a person's name, user ID, face images, and derived face-feature information. Multiple samples may be captured for each registered user to improve robustness. The information is stored in a local database or application directory.

### B. Recognition Workflow

For each detected face, the application receives the ROI or corresponding location information, resizes and preprocesses the face, extracts recognition features, and compares them with the registered database. A successful match produces a registered-person result; otherwise the person is labelled unknown. The interface may display person name, user ID, recognition status, confidence, date, and time.

### C. Monitoring Interface

The monitoring GUI is designed to show live camera feed, FPGA connection status, camera status, face-detected state, bounding boxes, detected face images, recognized name, unknown-person indication, confidence, date/time, number of detected faces, and recognition history. Event history can be stored locally for later review.

## 7. HARDWARE AND SOFTWARE REQUIREMENTS

Hardware—Tang Nano FPGA development board; digital camera module; PC or laptop; USB programming cable; FPGA-to-PC communication interface; camera interface circuitry; 5-V/USB power supply; connecting cables; and an optional prototype PCB or interface board.

Software—Verilog HDL; Gowin FPGA Design Software; FPGA synthesis, simulation and programming tools; Python; OpenCV; NumPy; serial communication library; a face-recognition algorithm; and a GUI development framework.

## 8. VERILOG MODULE ORGANIZATION

A modular RTL structure improves testability and integration. The proposed FPGA design comprises a clock generator, camera controller, pixel receiver, coordinate generator, RGB-processing block, color converter, skin-pixel detector, binary image generator, noise filter, region analyzer, bounding-box generator, ROI controller, communication controller, and a top-level controller. Each module can be verified independently before system-level integration.

## 9. FUNCTIONAL WORKFLOW

At startup, the FPGA initializes the camera and begins receiving video pixels. The pipeline performs RGB processing, YCbCr conversion, skin thresholding, binary segmentation, noise filtering, and region analysis. If no valid candidate is detected, streaming continues. When a candidate is found, the FPGA generates a bounding box, extracts or identifies the ROI, and sends the result to the PC. The PC preprocesses the face, compares it with the registered database, displays either a registered-person or unknown-person result, stores the recognition record, and continues monitoring.

## 10. EXPECTED RESULTS AND DISCUSSION

The proposed system is expected to demonstrate continuous real-time camera acquisition on the Tang Nano FPGA and deterministic pixel-level processing through a pipelined architecture. For a detected registered user, the monitoring application is intended to display the recognized person's identity, user ID, recognition status, confidence, FPGA status, camera status, date, and time. For a non-matching face, the application is intended to display an unknown/not-registered indication.

Because the source specification defines the project architecture and expected outputs rather than reporting completed benchmark measurements, this paper does not claim measured frame rate, recognition accuracy, resource utilization, or end-to-end latency. These quantities should be obtained during implementation using synthesis reports and controlled tests under different lighting, face pose, distance, and background conditions.

## 11. ADVANTAGES AND APPLICATIONS

The architecture demonstrates FPGA parallel processing, modular Verilog design, real-time camera input, and a practical hardware-software partition in which the FPGA handles time-critical detection and the PC handles recognition. Potential demonstration applications include smart access monitoring, attendance monitoring, laboratory and office entry monitoring, restricted-area monitoring, human-identification research, FPGA computer-vision education, and hardware-accelerated image-processing laboratories.

## 12. CONCLUSION

This paper described a Tang Nano FPGA-based real-time face-detection system with PC-assisted recognition. The FPGA performs camera acquisition and a hardware pipeline consisting of color conversion, segmentation, filtering, region analysis, bounding-box generation, and ROI selection. The PC performs recognition, user database matching, monitoring, and history logging. The resulting system is a compact hardware-software co-design platform for demonstrating FPGA-based digital image processing and flexible software-based identity recognition.

## REFERENCES


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

[2] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed. Pearson, 2018.

[3] G. Bradski, “The OpenCV Library,” Dr. Dobb’s Journal of Software Tools, 2000.

[4] S. Hauck and A. DeHon, Eds., Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation. Morgan Kaufmann, 2008.

[5] Gowin Semiconductor, “GOWIN FPGA Design Software and Tang Nano device documentation,” vendor documentation.