International Journal of Creative and Open Research in Engineering and Management
ISSN: 3108-1754 (Online)
Volume 02 Issue 07 July-2026 | Impact Factor: 3.5

Gesture Controlled Instrument Panel in Augmented Reality Using LabVIEW

Spriha Dibbi[1], Manas Panjwani[2], Roshan Sarmah[3], Thrisha Kumar[4]
Department of Electronics & Instrumentation Engineering
RV College of Engineering
Bengaluru, Karnataka, India

Mentors: Dr. Tabitha Raj and Dr. Venkatesh S
Department of Electronics & Instrumentation Engineering
RV College of Engineering
Bengaluru, India

***Abstract*—This project focuses on developing an augmented reality-based gesture control system using LabVIEW, where hand gestures are captured using a webcam, processed using classical image processing techniques, and interpreted to control system outputs. Multiple intermediate image representations are displayed on the front panel to provide insight into each stage of processing, making the system both functional and educational. Unlike virtual reality, which replaces the real environment entirely, AR preserves the real-world view and augments it with additional data. In gesture-based AR systems, users can see both their physical gestures and the system's interpretation of those gestures simultaneously, leading to more intuitive interaction.**

***Index Terms*—Augmented Reality, Gesture Control, LabVIEW, Image Processing, Human Computer Interaction.**

I. Introduction

Human computer interaction has traditionally depended on physical interfaces such as keyboards, mice, switches, and touchscreens. While these devices are effective for many applications, they impose limitations in environments where physical contact is undesirable, impractical, or unsafe. Situations such as medical environments, industrial automation systems, public interactive displays, and assistive technologies demand alternative interaction methods that are hygienic, intuitive, and accessible.

Gesture recognition provides a promising solution by allowing users to communicate with machines using natural hand movements. Vision-based gesture recognition systems utilize cameras and image processing algorithms to interpret human gestures without the need for wearable sensors or physical contact.

When combined with augmented reality, gesture recognition enables systems where digital feedback is seamlessly integrated into the real-world visual scene. Augmented Reality (AR) refers to the enhancement of real-world environments by overlaying virtual information onto live visual input. Unlike virtual reality, which replaces the real environment entirely, AR preserves the real-world view and augments it with additional data. In gesture-based AR systems, users can see both their physical gestures and the system's interpretation of those gestures simultaneously, leading to more intuitive interaction.

This project focuses on developing an augmented reality-based gesture control system using LabVIEW, where hand gestures are captured using a webcam, processed using classical image processing techniques, and interpreted to control system outputs. Multiple intermediate image representations are displayed on the front panel to provide insight into each stage of processing, making the system both functional and educational.

II. Problem Statement

With the increasing demand for touchless interaction systems, there is a need for reliable and real-time gesture recognition solutions that can operate using simple hardware and without excessive computational requirements. Many modern gesture recognition systems rely on artificial intelligence or deep learning models, which require large datasets, extensive training, and powerful hardware. These requirements can make such systems unsuitable for educational purposes, real-time embedded applications, or environments with limited computational resources.

The problem addressed in this project is the development of a real-time, non-AI-based gesture recognition system that can interpret hand gestures using deterministic image processing techniques. The system must be capable of detecting hand presence and, in theory, recognizing gestures based on finger count. Additionally, the system should provide immediate visual feedback using augmented reality overlays, ensuring an interactive and user-friendly experience.

III. Objectives of the Project

The objectives of this project are outlined below:

- To design a real-time vision-based gesture recognition system using LabVIEW.
- To acquire live video input using a standard webcam.

- To preprocess captured images for efficient and stable analysis.
- To segment the hand region from the background using thresholding techniques.
- To apply morphological operations for noise reduction and image refinement.
- To extract hand features such as area, edges, and skeletal structure.
- To study and partially implement finger-count-based gesture recognition using skeleton images.
- To overlay gesture information directly onto the live camera feed, demonstrating augmented reality.
- To control output indicators based on recognized gestures.
- To analyze system limitations and propose future improvements.

IV. Hardware and Software Requirements

A. Hardware Requirements

The hardware setup for this project is minimal and cost-effective.

- **Webcam:** A standard USB webcam is used to capture live video input. The resolution selected provides sufficient detail for hand detection while maintaining real-time performance.
- **Computer System:** A personal computer with adequate processing capability is required to run LabVIEW and handle continuous image processing tasks. The system does not require specialized hardware such as GPUs, as no AI-based processing is used.

B. Software Requirements

- **LabVIEW:** Used as the primary development environment for implementing the block diagram and front panel.
- **NI Vision Development Module:** A specialized library providing high-level functions for image processing, segmentation, and feature extraction.
- **IMAQ Vision Library:** A collection of VIs used for managing multiple image buffers such as RGB, Grayscale, and Binary images.
- **AR Overlay Module:** A set of functions used to draw non-destructive text and rectangles directly onto the live video feed.

V. System Architecture

The augmented reality-based gesture control system operates in real time using a continuous while loop. Each frame captured from the webcam is processed independently through a structured image processing pipeline.

The process begins with image acquisition, where frames are captured and stored in pre-allocated image buffers using IMAQ Create blocks. These buffers store intermediate images such as raw, grayscale, thresholder, and skeleton images, improving execution efficiency. The captured RGB image is first converted to grayscale to reduce computational complexity. The grayscale image is then processed using range-based thresholding to segment the hand from the background, producing a binary image.

To improve segmentation quality, morphological operations such as opening and hole filling are applied. These operations remove noise and close gaps, resulting in a cleaner binary representation of the hand. The refined binary image is analysed using particle analysis, which extracts features such as area. The largest detected particle is assumed to be the hand. The measured area is compared with a predefined threshold to determine whether an open palm is present. This forms the currently implemented gesture detection logic.

In parallel, the block diagram includes a skeletonization stage for advanced gesture recognition. Skeletonization converts the binary hand image into a one-pixel-wide structural representation while preserving connectivity. This skeleton highlights the central lines of the palm and fingers. The intended gesture logic is based on finger counting using the skeleton image. A horizontal reference line is drawn across the skeleton, and the number of intersections between this line and skeletal branches corresponds to the number of extended fingers. Although the skeleton image is successfully generated, full pixel-level finger counting is only partially implemented due to limitations in express blocks and noise sensitivity.

Finally, augmented reality overlays display system status and recognized gestures directly on the live camera feed, completing the real-time interaction loop. This block diagram provides a clear, modular framework for real-time gesture recognition and serves as a strong foundation for further enhancements.

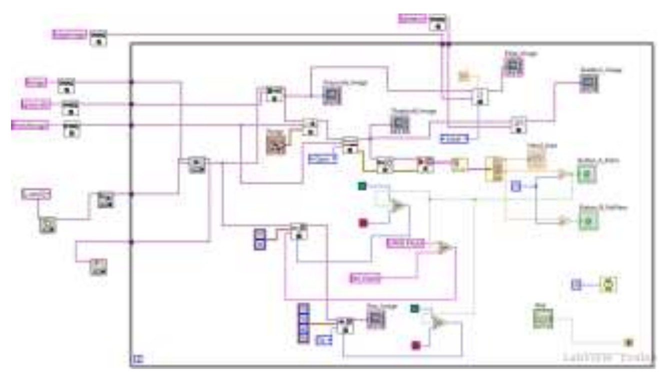

Fig. 1. Block Diagram Of The Architecture

VI. Image Processing Pipeline

The image processing pipeline is the core component of the system. Each stage transforms the image in a specific way, and the outputs of these stages are displayed on the front panel for visualization and analysis.

A. Raw Image Acquisition

The raw image is the unprocessed RGB image captured directly from the webcam. Each frame represents the real-world scene and is continuously updated in real time. This image serves as the base input for all subsequent processing stages. Augmented reality overlays such as gesture status text and indicators are drawn on this image, allowing the user to observe the interaction in real time.

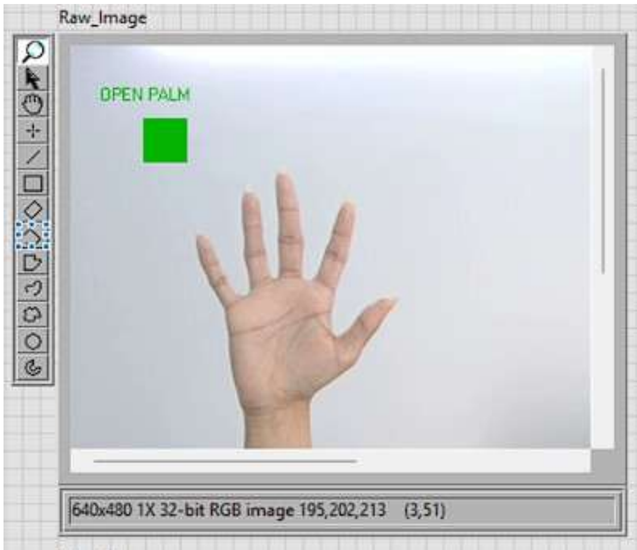

Fig. 2. Raw Image Acquisition

B. Grayscale Conversion

Grayscale conversion reduces the three-channel RGB image into a single-channel intensity image. This step simplifies the data and reduces computational load. Since gesture recognition primarily depends on shape and structure rather than color, grayscale representation is sufficient and more robust to lighting variations. The grayscale image provides a stable foundation for thresholding and segmentation.

The conversion is performed using the standard luminance weighting formula, which aligns with human visual perception:

$$I_{gray}(x, y) = 0.299 \cdot R(x, y) + 0.587 \cdot G(x, y) + 0.114 \cdot B(x, y) \quad (1)$$

where $R(x, y)$, $G(x, y)$, and $B(x, y)$ represent the Red, Green, and Blue intensity values at pixel coordinate (x, y).

Fig. 3. Grayscale Image Conversion

C. Thresholding

Thresholding is used to distinguish the hand from the background based on pixel intensity values. A range-based threshold is applied, where pixels falling within a specified intensity range are classified as foreground, and all other pixels are classified as background. Manual adjustment of threshold values allows the system to adapt to different lighting conditions. The output of this stage is a binary image in which the hand region is clearly highlighted.

Mathematically, a range-based threshold is applied to the grayscale image I_{gray} to generate a binary image $B(x, y)$:

$$B(x, y) = \begin{cases} 1 & \text{if } T_{min} \leq I_{gray}(x, y) \leq T_{max} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where T_{min} and T_{max} represent the lower and upper intensity limits defining the skin tone or hand region.

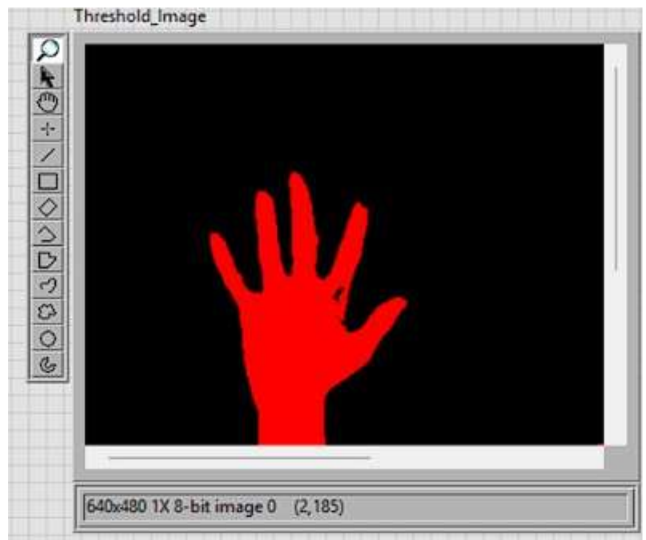

Fig. 4. Threshold Image

D. Morphological Operations

Morphological operations are applied to improve the quality of the thresholder binary image. Noise removal and hole filling ensure that the hand appears as a single continuous object. These operations are crucial for accurate area measurement and skeletonization, as irregularities in the binary image can lead to incorrect feature extraction.

Specifically, the "Opening" operation is used to remove noise. Letting B be the binary image and S be the structuring element (kernel), the opening operation is defined as:

$$B \circ S = (B \ominus S) \oplus S \quad (3)$$

where $\ominus$ denotes erosion and $\oplus$ denotes dilation. Erosion removes small noise particles, while dilation restores the primary shape boundaries.

E. Edge Detection

Edge detection highlights the boundaries of the hand by identifying regions of sharp intensity change. The Sobel operator is used to compute image gradients and extract edges. While the edge image provides useful visual information about hand shape, it is not reliable for gesture classification on its own due to sensitivity to noise and lighting variations.

The Sobel operator estimates the gradient magnitude using two 3×3 convolution kernels, K_x and K_y:

$$K_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad K_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{4}$$

The gradient magnitude G for each pixel is calculated as:

$$G = \sqrt{(I * K_x)^2 + (I * K_y)^2} \tag{5}$$

Fig. 5. Edge Image

F. Skeletonization

Skeletonization reduces the binary hand image to a one-pixel-wide representation that preserves the topological structure of the hand. This process highlights the central lines of the fingers and palm, providing a simplified structural model. Skeletonization forms the basis of the intended finger-count-based gesture recognition approach.

For finger counting, a horizontal reference line is projected across the skeletonized image $S(x, y)$. The number of fingers N_f is estimated by counting the discontinuities (transitions from 0 to 1) along this line:

$$N_f \approx \sum_x \mathbb{I}(S(x, y_{ref}) \neq S(x+1, y_{ref})) \tag{6}$$

where $\mathbb{I}$ is the indicator function. This allows the system to distinguish between different finger-count gestures based on the number of intersections.

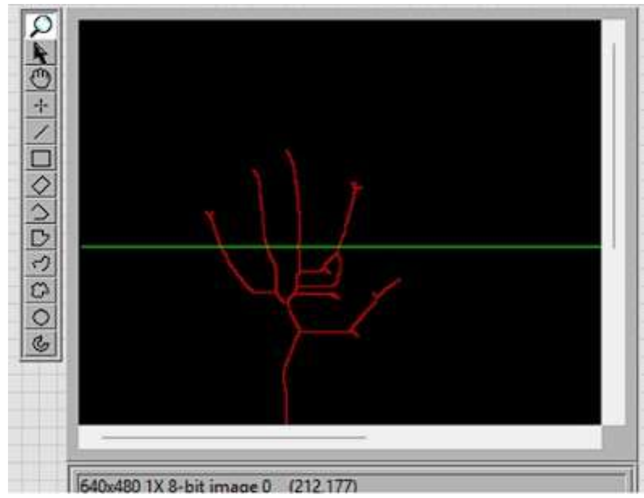

Fig. 6. Skeleton Image

VII. Augmented Reality Implementation

The augmented reality aspect of the system is a defining feature of the project. Augmented reality is achieved by overlaying virtual elements directly onto the live camera feed rather than displaying results separately.

In this system, the raw image captured from the webcam serves as the background layer, while gesture-related information is superimposed in real time. When the system detects a palm based on area analysis, visual indicators such as colored buttons and textual labels are displayed on the live image. These overlays provide immediate feedback to the user, allowing them to understand how the system is interpreting their gestures.

Because the feedback is presented within the same visual context as the real-world scene, the interaction feels natural and intuitive. This approach differentiates the system from conventional image processing applications that display processed images in isolation. By maintaining the real-world view and augmenting it with virtual information, the project satisfies the fundamental criteria of an augmented reality system. The AR implementation also enhances usability and demonstrates the practical application of gesture recognition in interactive environments.

Fig. 7. Open Palm Gesture Case

International Journal of Creative and Open Research in Engineering and Management
ISSN: **3108-1754 (Online)**
Volume 02 Issue 07 July-2026 | Impact Factor: 3.5

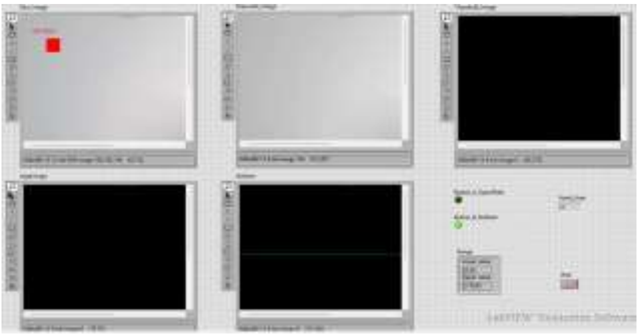

Fig. 8. No Palm Gesture Case

VIII. Results and Outcomes

The augmented reality-based gesture control system was successfully implemented and tested under real-time conditions using a webcam. The system continuously acquired live video input and displayed it on the front panel with augmented reality overlays such as gesture status text and visual indicators. This confirmed the correct integration of real-world video with virtual feedback.

Grayscale conversion produced stable intensity images that were less sensitive to color and lighting variations, improving the reliability of subsequent processing stages. Thresholding successfully segmented the hand from the background under controlled lighting conditions, clearly isolating the hand as a foreground object. Manual threshold adjustment enabled adaptation to varying illumination levels. Morphological operations effectively removed noise and filled gaps within the palm region, resulting in a single continuous hand object. This significantly improved the accuracy of area measurement and skeleton generation.

Area-based analysis reliably detected an open palm; when the hand area exceeded the predefined threshold, the system correctly identified the gesture and activated the corresponding augmented reality indicators and LED output. Edge detection highlighted the outlines of the hand and fingers, providing visual shape information. However, it showed sensitivity to noise and discontinuities, indicating that edge-based methods alone are insufficient for accurate gesture classification.

Skeletonization generated a clear one-pixel-wide representation of the hand, revealing distinct finger branches and a central palm region. Skeleton images supported the theoretical finger-count-based gesture recognition approach. The horizontal reference line intersected multiple skeletal branches corresponding to extended fingers. However, skeleton breaks, background interference, and pixel-level extraction limitations prevented reliable automatic finger counting. Overall, the results confirm reliable palm detection, effective augmented reality visualization, and the feasibility of skeleton-based finger counting with further refinement.

IX. Future Scopes

Future enhancements to the system may include adaptive thresholding techniques to handle varying lighting conditions,

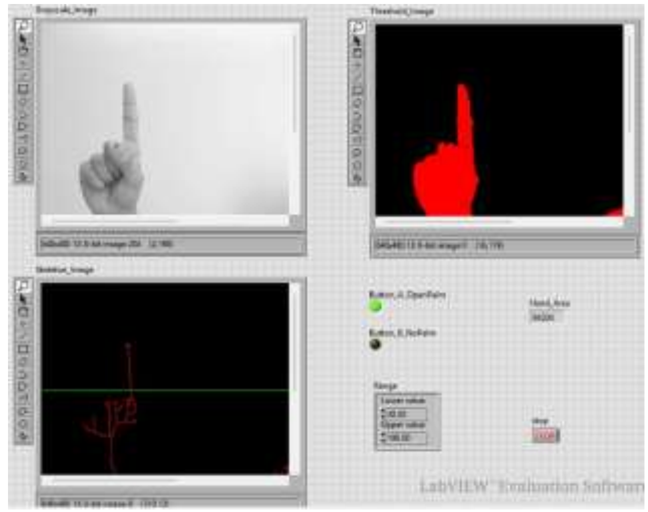

Fig. 9. One Finger Gesture

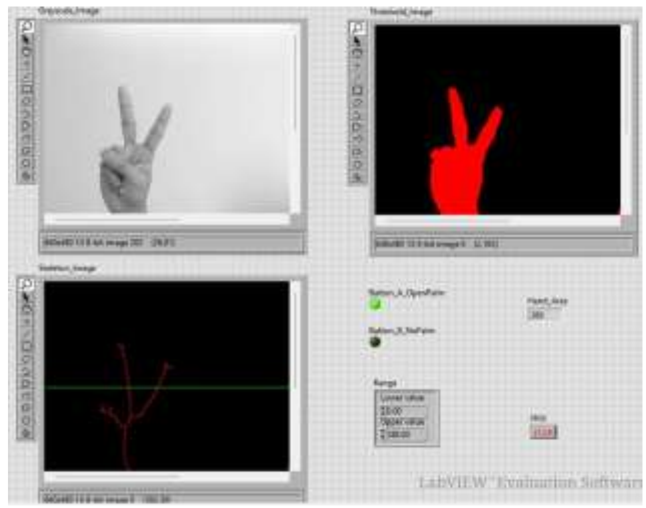

Fig. 10. Two Finger Gesture

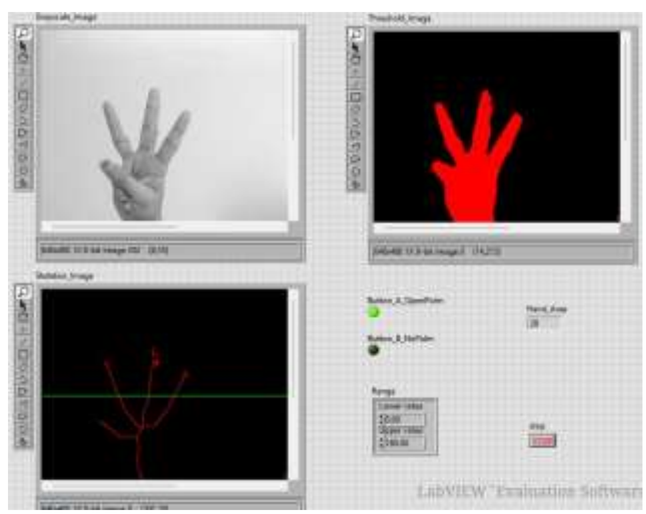

Fig. 11. Three Finger Gesture

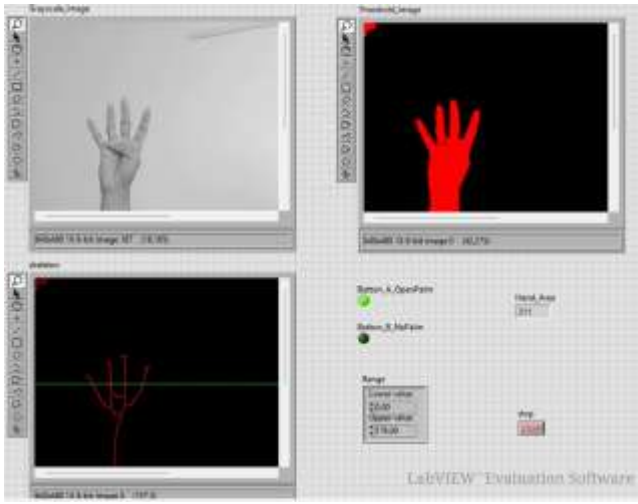

Fig. 12. Four Finger Gesture

improved morphological filtering to produce cleaner skeletons, and region-of-interest-based analysis for more accurate finger counting. Accessing pixel arrays directly rather than relying on express blocks would enable more precise implementation of the finger-counting algorithm.

While this project intentionally avoids AI, future versions could optionally integrate machine learning techniques to improve robustness, allowing comparison between classical and AI-based approaches. The system could also be extended to control physical devices using microcontrollers or data acquisition hardware, expanding its practical applications.

X. Non-AI Approach and Its Advantages

A key design choice in this project is the deliberate exclusion of artificial intelligence, machine learning, and deep learning techniques. All gesture recognition is performed using classical, rule-based image processing methods. This decision offers several advantages, particularly in an educational and real-time application context.

Firstly, the non-AI approach provides complete transparency. Each processing step is explicitly defined and visible in the block diagram, allowing users to understand exactly how the system works. This transparency is especially valuable in academic settings, where understanding underlying principles is more important than achieving maximum accuracy.

Secondly, the computational requirements of the system are significantly lower compared to AI-based approaches. The system can operate in real time on standard hardware without the need for GPUs or extensive memory resources. This makes the system suitable for embedded applications and environments with limited computational capacity.

Thirdly, the absence of AI eliminates the need for training data. AI-based systems require large datasets and extensive training, which can be time-consuming and impractical for small-scale projects. In contrast, the deterministic approach used in this project relies on well-defined rules and thresholds that can be adjusted manually.

Finally, debugging and system modification are easier in a non-AI system. Since each block performs a specific and predictable function, errors can be traced and corrected systematically. This level of control is difficult to achieve in AI-based systems, which often behave as black boxes.

XI. Applications

The augmented reality-based gesture control system developed in this project has a wide range of potential applications in areas where intuitive, touchless, and real-time interaction is required. Since the system relies on classical image processing techniques rather than artificial intelligence, it is particularly suitable for applications that demand transparency, low computational complexity, and predictable behavior.

- **Smart Home and Device Control:** Hand gestures can be mapped to control household appliances such as lights, fans, or media systems. The non-AI approach ensures quick response times and minimal hardware requirements, making it suitable for basic smart home applications.
- **Assistive Technologies:** Gesture-based control systems can assist individuals with physical disabilities by providing an alternative method of interaction. Simple hand gestures can be used to control software interfaces, communication panels, or basic devices, improving accessibility and ease of use.
- **Industrial Automation Interfaces:** In industrial environments, gesture control can be used to operate machines or monitoring panels from a distance. This is especially useful in hazardous areas where direct interaction with control equipment may pose safety risks.
- **Robotics and Mechatronics Control:** Gesture recognition can be used to control robotic systems, robotic arms, or mobile robots through simple hand gestures. The LabVIEW-based implementation allows easy integration with data acquisition hardware and microcontrollers.
- **Educational and Training Applications:** The project is highly suitable for educational purposes, as it demonstrates real-time image processing, gesture recognition, and augmented reality using a graphical programming environment. It can be used as a teaching aid in engineering laboratories and workshops.

XII. Conclusion

This project successfully demonstrates the design and implementation of an augmented reality-based gesture control system using LabVIEW and classical image processing techniques. By combining real-time image acquisition, structured preprocessing, feature extraction, and augmented reality overlays, the system provides an intuitive and interactive user experience. Although finger-count-based gesture recognition is only partially implemented, the theoretical framework and experimental results validate the feasibility of the approach. The project highlights the strengths and limitations of non-AI-based gesture recognition and provides a strong foundation for future research and development.

REFERENCES

[1] C. Santosh Reddy, M. Rout, and S. Banerjee, “Hand Gesture-Based Robot Control Using LabVIEW,” in *Proceedings of the International Conference on Emerging Trends in Engineering and Technology*, 2022.

[2] Y. V. S. Reddy, “LabVIEW Based Hand Gestures Control,” in *Proceedings of the International Conference on Advanced Materials, Manufacturing and Smart Devices (ICAMMSD)*, Atlantis Press, 2025.

[3] “Hand Gesture Recognition System to Control Soft Front Panels,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 3, no. 1, pp. 1–5, 2016.

[4] “Gesture Controlled Robot Using LabVIEW Vision and DAQ Techniques,” *International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering (IJIREEICE)*, vol. 4, no. 5, pp. 230–234, May 2016.

[5] Z. Zhang, “A Flexible New Technique for Camera Calibration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[6] T. Y. Zhang and C. Y. Suen, “A Fast Parallel Algorithm for Thinning Digital Patterns,” *Communications of the ACM*, vol. 27, no. 3, pp. 236–239, 1984.

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

[8] National Instruments, *LabVIEW Vision Development Module User Manual*, National Instruments Corporation.

[9] National Instruments, *IMAQ Vision Concepts Manual*, National Instruments Corporation.

[10] S. K. Patil and P. R. Patil, “Real-Time Hand Gesture Recognition System Using LabVIEW and Vision Assistant,” *International Journal of Computer Applications*, vol. 179, no. 7, pp. 22–27, 2018.

[11] A. R. Jadhav and S. A. Ladhake, “LabVIEW-Based Vision System for Hand Gesture Recognition,” *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 5, no. 6, pp. 1852–1856, 2016.