

Comparative Evaluation of Machine Learning Models for Hand Gesture Recognition Using Feature Extraction

Fauziah ^{1,*}, Riadi M. Dinata², Dhieka A. Lantana¹, and Billy Hendrik³

¹Faculty of Information and Communications Technology, Universitas Nasional, South Jakarta, Indonesia

²Faculty of Science and Technology, National Institute of Science and Technology (ISTN), Indonesia

³Faculty of Information Technology, Putra Indonesia University (YPTK), Padang, Indonesia

Email: fauziah@civitas.unas.ac.id (F.); riadimrt@gmail.com (R.M.D.); dhiekalantana12@gmail.com (D.A.L.); billy_hendrik@upiypk.ac.id (B.H.)

*Corresponding author

Abstract—Hand Gesture Recognition (HGR) is essential in enabling natural interaction between humans and machines. However, achieving high recognition accuracy remains challenging, especially when using lightweight machine learning models combined with classical feature extraction methods. This study evaluates the effectiveness of moment invariant features integrated with three classifiers (Support Vector Machine (SVM), Decision Tree (DT), and K-Nearest Neighbor (KNN)) for gesture classification. A dataset consisting of 1200 gesture samples is used to train and test the models. The evaluation results show that SVM achieved a near-perfect a testing accuracy of 81.41%. The DT classifier follows with a testing accuracy of 77.25%. Meanwhile, KNN obtains a testing accuracy of 78.00%. These findings demonstrate that moment invariant features effectively capture discriminative geometric patterns in gesture data and highlight the superior performance of SVM for robust and efficient classification. The proposed approach offers strong potential for real-time gesture-based systems operating in resource-constrained environments, particularly where deep learning architectures may be impractical. This study contributes to the development of HGR solutions that are efficient, accessible, and suitable for deployment in low-power or limited-computation scenarios.

Keywords—decision tree, feature extraction, gesture-based interaction, human-machine interaction, k-nearest neighbor, pattern recognition, moment invariant, support vector machine

I. INTRODUCTION

Gesture-Based Interaction (GBI) is a vital enabler of natural and intuitive communication in Interactive Systems (IS), with applications in gesture-based communication, robotic control, gaming, surveillance, and smart environments [1]. Recent advances in Red-Green-Blue Depth (RGB-D) sensors, which combine color and depth modalities, have significantly enhanced gesture detection by improving spatial resolution and

reducing background interference. These multimodal inputs enable more accurate segmentation of hand regions, particularly in complex environments with dynamic lighting or cluttered backgrounds [2]. Image processing techniques, including binary thresholding, further enhance segmentation quality [3].

In assistive technologies, particularly for users with upper-limb amputations, the integration of surface Electromyography (s-EMG), artificial vision, and tactile feedback has increased the autonomy and functionality of prosthetic hands [4]. Meanwhile, deep learning architectures such as 3D Convolutional Neural Networks (3DCNN) have demonstrated high accuracy in dynamic gesture recognition, even under limited training data conditions [5]. Other emerging solutions, including low-cost data gloves and multimodal sensing systems, are being explored for real-time and scalable gesture-based control [6]. However, systems relying on s-EMG often suffer from signal variability and inter-user differences, which can reduce classification accuracy [7].

To address these limitations, hybrid recognition systems that integrate Inertial Measurement Units (IMUs) with vision-based approaches have been developed, particularly in applications such as Unmanned Aerial Vehicle (UAV) control [8]. Recent studies have also emphasized the importance of reducing computational overhead while maintaining high accuracy. Lightweight models that employ spatiotemporal fusion techniques have achieved performance comparable to deeper neural networks while requiring significantly fewer computational resources [9]. Fusion-based approaches that combine shape descriptors and motion features have likewise demonstrated high recognition rates across multiple datasets [10].

From a feature-engineering perspective, moment-based descriptors remain highly relevant due to their invariance to translation, scale, and rotation. Modified hu moment invariants, in particular, offer strong discriminative capability in encoding hand-shape information and often outperform traditional contour-based features [11].

Several comparative studies show that Support Vector Machines (SVM) combined with handcrafted descriptors such as hu moments and Histogram of Oriented Gradients (HOG) consistently outperform Hidden Markov Models (HMM), especially in real-time hand gesture classification tasks. Although HMMs are suitable for temporal sequence modeling, they often incur higher computational costs and achieve lower accuracy in resource-constrained environments [12].

Driven by the increasing need for lightweight and efficient Hand Gesture Recognition (HGR) systems, this study investigates the effectiveness of moment invariant features combined with three classical machine learning algorithms: SVM, Decision Tree (DT), and K-Nearest Neighbor (KNN), using static gesture datasets. This work highlights the viability of traditional feature-based methods in delivering competitive recognition accuracy while minimizing computational overhead, making them suitable for gesture-recognition applications with limited computational resources. This study also introduces an efficiency-centered perspective by integrating moment invariant features with classical classifiers to provide a lightweight yet accurate alternative to resource-intensive deep learning models.

II. LITERATURE REVIEW

Gesture recognition has become a central research focusing on advancing user interaction, offering natural and intuitive communication in robotics, virtual reality, and smart environments [13]. While early systems relied on wearable sensors, vision-based approaches now dominate due to their non-intrusive nature [14]. Numerous studies have proposed various feature extraction techniques and classification methods to enhance accuracy and system reliability [15]. A key challenge involves extracting features that are invariant to scale, rotation, and translation [16]. Recent research has also explored integrating such descriptors with modern machine learning methods to further improve recognition performance [17]. Although deep learning models such as Convolutional Neural Networks (CNNs) and transformers dominate the field, their high computational cost and large data requirements limit their practicality in real-time or resource-constrained settings [18, 19].

In response to challenges in feature representation, moment invariants have emerged as a robust solution. Among these descriptors, hu moment invariants are particularly valued for their invariance to translation, rotation, and scaling, which enables consistent shape representation despite spatial transformations. Since their introduction, moment invariants have been widely utilized in various pattern recognition and image classification tasks, including optical character recognition and object identification [20, 21].

Deep learning models, such as Convolutional Neural Networks (CNNs), have significantly advanced vision-based tasks by automatically learning hierarchical features from raw inputs often removing the need for handcrafted descriptors like moment invariants [22]. Models such as the Detection Transformer (DETR) [23]

and its successors have demonstrated exceptional performance in object detection and recognition. These architectures use self-attention mechanisms to capture global dependencies and process hierarchical features, often achieving state-of-the-art accuracy. However, their primary limitation lies in their substantial computational demands, large memory usage, and high power consumption. Their reliance on high-end GPUs or specialized accelerators makes them unsuitable for deployment on edge devices, Microcontroller Units (MCUs), or embedded systems with limited resources.

In contrast, the objective of our study is to develop an efficient and deployable solution for specialized Hand Gesture Recognition (HGR). Our approach employs moment invariants (M1–M7) as lightweight, low-dimensional feature extractors combined with fast-inference classical machine learning classifiers (SVM, DT, and KNN), forming a highly efficient baseline. This method provides the robust classification performance, as demonstrated by our accuracy metrics, while maintaining minimal computational requirements which are suitable for real-time operation in resource-constrained environments.

The novelty of this work stems from providing an efficient alternative to computationally heavy deep learning approaches for HGR. This study examines the underexplored comparison between classical machine learning models and that use moment invariants for hand gesture recognition. By systematically evaluating both categories, we aim to highlight the strengths and limitations of moment invariants as lightweight yet reliable features for gesture-intelligence systems, particularly in environments where computational resources are limited.

III. METHODOLOGY

This section provides a comprehensive explanation of the gesture analysis system, covering data acquisition, feature extraction, classification, and comparative performance evaluation.

A. Data Acquisition, Pre-processing, and Segmentation

This initial phase establishes the dataset and prepares the images for analysis. It begins with the camera configuration stage, followed by motion capture and the extraction of individual video frames. The extracted frames are then converted into a consistent and manageable format before being stored for subsequent processing. The next critical step is segmentation, during which the hand region is isolated from the background and converted into binary images. This segmentation ensures that only the hand's shape is preserved and used for later feature analysis.

B. Feature Extraction: Moment Invariants

Extracting relevant attributes is performed using the moment invariant method, a technique selected for its robustness. This process derives a fixed set of features (typically hu moment invariants, M1–M7) that are invariant positional, angular, and size variations of the

hand gesture image. The resulting seven features for each segmented image serve as the input vector for the subsequent classification algorithms.

$$\begin{aligned}
 M1 &= \eta_{20} + \eta_{02} \\
 M2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
 M3 &= (\eta_{30} - 3\eta_{12}) + (3\eta_{21} - \eta_{03})^2 \\
 M4 &= (\eta_{30} + \eta_{12}) + (\eta_{21} + \eta_{03})^2 \\
 M5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3 \times (\eta_{21} + \eta_{03}) \right]^2 + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12}) - (\eta_{21} + \eta_{03}) \right]^2 \\
 M6 &= (\eta_{20} - \eta_{02}) \left[(\eta_{30} + \eta_{12}) - (\eta_{21} + \eta_{03}) \right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})^2 \\
 M7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3 \times (\eta_{21} + \eta_{03}) \right]^2 + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \left[3 \times (\eta_{30} + \eta_{12}) - (\eta_{21} + \eta_{03}) \right] \quad (1)
 \end{aligned}$$

C. Classification Algorithms and Identification Process

The identification processing involves training and testing three distinct machine learning algorithms.

1) Algorithm process

a) Support Vector Machine (SVM)

SVM deformations for multi-class problems, including the binary-tree, one-to-one, and one-to-remainder methods. The one-to-one approach stands out among the others thanks to its efficiency, simplicity, and high rate of correct recognition. Its fundamental concept is as follows: Given an N-class classification problem, train SVMs first. Then, using the voting principle among all SVMs, determines the classification result of the testing data [24].

b) Decision Tree (DT)

For the decision-making process of models, decision trees are effective and adaptable tools in data analysis and machine learning. When a problem needs to be solved at every terminal node that matches a group naming, the DT uses neighbour row which are related to each other.

Moment invariant provides a powerful means of extracting features to identify and classify objects in images, even if the objects are rotated, translated, or scaled. It is widely used in patterns.

$$d(P, Q) = \sqrt{\sum_{i=0}^n (x_{1i} - x_{2i})^2 + (y_{1i} - y_{2i})^2} \quad (2)$$

An impersonation tree appears the characteristics at an internal node of the tree [25].

c) K-Nearest Neighbour (KNN)

KNN determines the class of a new data point using the majority vote from nearby sample. This is an algorithm for learning based on instances, which means it stores all the training dataset in memory and does not produce a true model [26]. The KNN algorithm is based on the principle that objects are consistently in recognition, computer vision, and image processing application [27]. The hu moments are seven geometric-invariant moment descriptors that stay constant despite transformation effects transformations such as rotation, translation, and scaling [28–30].

2) Development model

From Fig. 1, process steps are following:

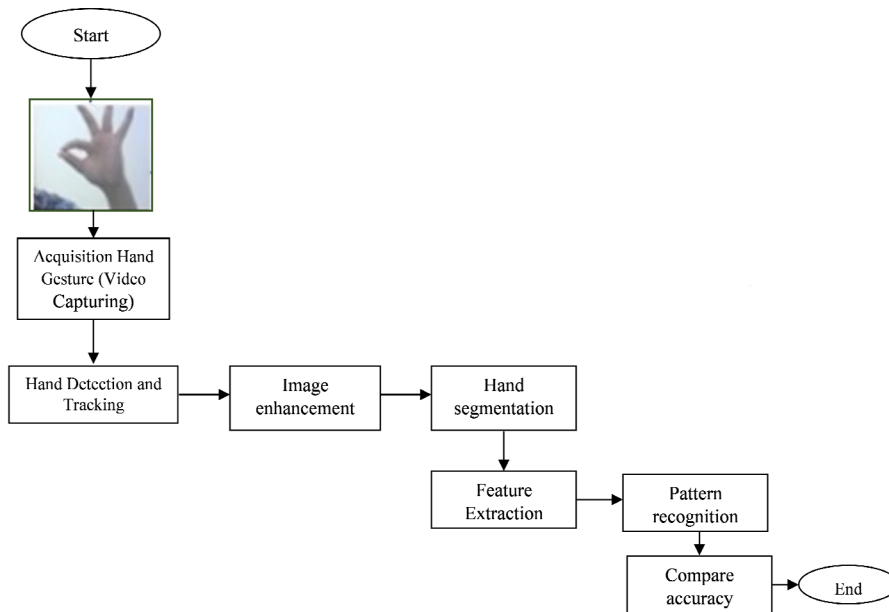


Fig. 1. Development research and contribution novelty.

- (1) Video capture: It begins with video capture, in which videos or individual frames are obtained from the camera as the primary input for analysis. The input may come from a standard RGB camera or a depth sensor that provides additional spatial information.
- (2) Hand detection and tracking: Filtering the hand area using object detection algorithms, in videos, this is typically done using object tracking techniques based on colour, shape.
- (3) Image enhancement: Converting to grayscale images or reducing noise to facilitate further processing.
- (4) Hand segmentation: Isolating the hand from the background and other elements in the video. Methods such as background subtraction or skin colour detection are often used for this purpose.
- (5) Feature extraction: Extracting information from the detected hand. This can include the shape of the hand, movement, or position to the camera geometric features or motion patterns which are used to analysis hand movements consisting of 7 moment invariant values for each data point, along with their corresponding class labels. This dataset is already in binary image format.
- (6) Pattern recognition: After features are extracted, machine learning models with SVM, KNN and DT are used to recognize hand gestures based on the learned patterns in video-based gesture recognition, particularly for handling sequences of images (video frames).
- (7) Compare accuracy: After hand gesture pattern recognition is obtained, the next step is to compare the gesture recognition results with the training and testing sets used to measure the model's performance and accuracy.

D. Data and Video

Data used in this study consists of hand movement videos to frame (1200 data), which undergo digital image processing. The vision-based hand recognition process typically involves several key stages. Device and the environment preparation come first, which includes setting up the camera—either RGB or depth sensor—to record the hand gestures, ensuring proper lighting conditions so the hand movements can be clearly captured, and positioning the camera optimally to obtain consistent visibility of gestures. Following this, video capture is performed by recording sequences of hand movements or specific user interactions. These videos are composed of multiple consecutive frames. After the video is captured, the next critical phase is pre-processing, which involves preparing the raw video frames for analysis. This includes tasks such as extracting frames, resizing them to a uniform dimension, applying noise reduction filters, and converting the colour space—for instance, from RGB to grayscale—to reduce computational complexity. The subsequent step is segmentation, where the hand region is separated from the background. This is achieved using techniques such as background subtraction, skin colour-based filtering, or

depth masking, depending on the type of camera or sensor used. Proper segmentation plays a crucial role in minimizing irrelevant data and ensuring that only meaningful features are extracted. Once segmentation is complete, the system proceeds to feature extraction. In this stage, characteristics that describe the shape, motion, and contour of the hand are derived. These features may include spatial descriptors such as moment invariants, edge orientations, or hand contour shapes, and temporal features like motion trajectories over time. The extracted features are then compiled into structured representations, ready to be classified by machine learning models. This pipeline, from video acquisition through pre-processing, segmentation, and feature extraction, is designed to improve the accuracy, consistency, and robustness of the hand gesture recognition system, even under diverse environmental conditions and user-specific variations.

The hand gesture video dataset includes several distinct gesture classes. The “blank” category refers to frames where no specific gesture is performed serving as neutral positions prior to any motion being detected by the recognition system. The “Thumbs-up” gesture features an extended thumb with the remaining four fingers raised, indicating affirmation. The “Fist” gesture involves a fully closed hand, where fingers are curled inward and wrapped by the thumb. The “Thumbs-down” gesture is formed by folding the four fingers and extending the thumb downward, typically used to express disapproval. The “Five” gesture consists of a fully open palm with all five fingers extended. Meanwhile, the “OK” gesture is created by forming a circular shape with the thumb and index finger, while the other fingers remain extended outward. These gestures are visually illustrated in Fig. 2.

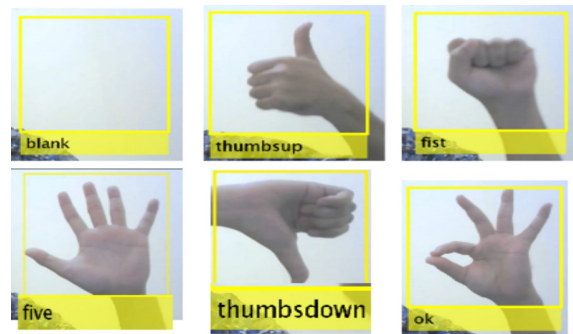


Fig. 2. Data hand gesture.

IV. RESULTS AND ANALYSIS

A. Performance Results of the Developed Model

The proposed model was developed through a systematic process involving several essential stages. Initially, the training dataset was compiled, where each sample was characterized by seven moment invariant features. These features were coupled with corresponding class labels, forming a supervised dataset suitable for classification tasks. Given that the input data were binary images, the extraction of moment invariant features was

straightforward, thus enhancing computational efficiency. Subsequently, a testing dataset was constructed using the same feature extraction approach, ensuring consistency in representation. During the model training phase, three distinct machine learning classifiers were trained using the training data and their associated labels. The training accuracy was computed by evaluating the correspondence between predicted and actual class labels. Upon completion, the trained models were stored for evaluation purposes. In the testing phase, the stored models were reloaded and applied to the test dataset, which included only feature data without ground-truth labels during prediction. The predicted outputs were then compared against the actual test labels to determine the classification accuracy. This structured methodology enabled reliable performance evaluation and the classification of hand gestures with approach captures essential shape descriptors invariant to scale, rotation, and translation, and integrates them with classical machine learning algorithms to enhance recognition performance across varied gesture classes. Furthermore, Fig. 3 explains the detailed processing steps involved, emphasizing the contributions and innovative aspects of the developed model. Image processing begins with the definition of a Region of Interest (ROI), enabling segmentation that separates relevant hand areas from the background, resulting in a simplified binary representation. Image segmentation is then performed on individual frames (each frame being a static image

extracted from a video sequence) to isolate and analyze specific objects or motion patterns.

The overall results demonstrated the effectiveness of moment invariant-based features in facilitating accurate hand gesture recognition using classical machine learning approaches.

The overall workflow is illustrated in Fig. 3, where moment invariant features are introduced as a key novelty of the study. By applying segmentation to each frame, the system can track dynamic changes and interpret gesture movement with greater precision. This frame-by-frame analysis establishes the groundwork for effective feature extraction and enhanced the robustness of the gesture classification pipeline. To validate the effectiveness of the proposed hand gesture recognition system based on moment invariants, a series of gesture samples are collected and processed to extract seven key moment features (M1–M7). These features represent geometric descriptors that are invariant to rotation, scale, and translation, providing robust and discriminative information for gesture classification. Each gesture class, such as ‘Blank’, ‘Fist’, ‘Five’, ‘OK’, ‘Thumbs down’, and ‘Thumbs up’, was evaluated through repeated sample acquisition to confirm consistency and feature distinctiveness. The resulting data for each gesture types are presented in Tables I–VI, demonstrating how the extracted moment features correspond to specific gesture labels. These structured datasets were subsequently used as input for training and evaluating various machine learning models in the classification phase.

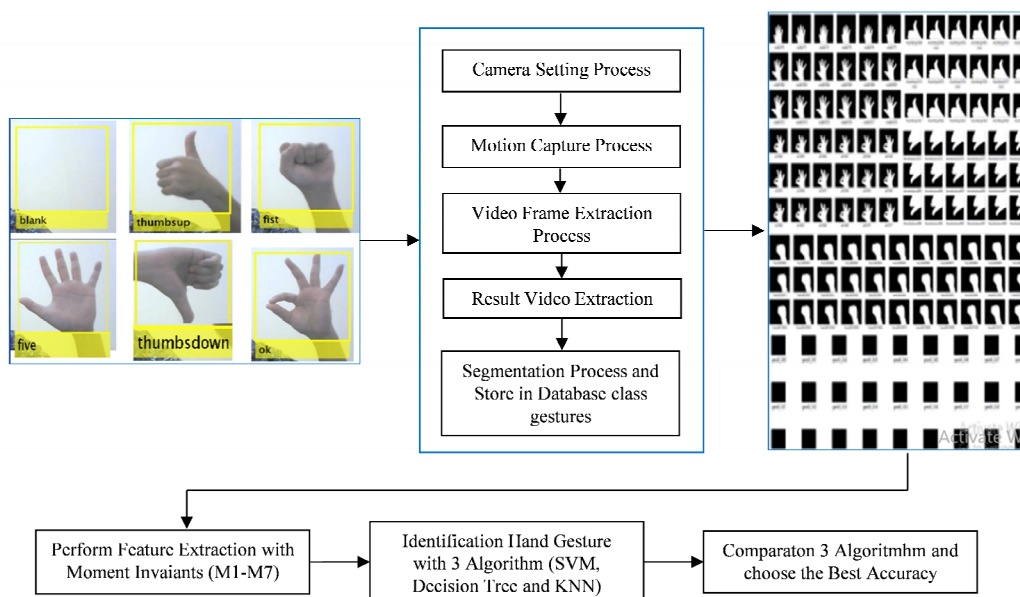


Fig. 3. The proposed research framework highlighting the novelty of using moment invariant features.

The extracted moment invariant features in Table I are for the ‘blank’ gesture, which typically serves as the neutral or idle hand state. In this study, M1–M7 denote the seven hu moment invariants, which capture geometric features of the gesture shapes and remain invariant to translation, rotation, and scaling. All the moment values (M1–M7) are zero, reflecting a completely black or

empty region in the binary image indicating the absence of any detectable hand gesture. This outcome validates the reliability of the moment invariant method in identifying and classifying non-gestural frames, which is essential for minimizing false positives during continuous gesture monitoring.

TABLE I. SAMPLE RECOGNITION OF ‘BLANK’ GESTURE

Hu Moment	M1	M2	M3	M4	M5	M6	M7	Label
Frame 1	0	0	0	0	0	0	0	‘Blank’

The ‘Fist’ gesture is characterized by compact and symmetrical hand shapes. The moment invariant values in Table II, particularly M1–M3, exhibit small but non-zero magnitudes, indicating strong mass concentration

with minor asymmetry. In contrast, the higher-order moments (M4–M7) present significantly larger values due to their sensitivity to curvature and spatial distribution. These distinctive patterns enable the classifier to robustly differentiate the ‘fist’ gesture from other classes, confirming the discriminative capability of the hu moment descriptors, especially the lower-order moments, in recognizing tightly formed hand shapes.

TABLE II. SAMPLE RECOGNITION OF ‘FIST’ GESTURE

Hu Moment	M1	M2	M3	M4	M5	M6	M7	Label
Frame 102	1.86E-01	6.09E-03	8.51E-04	8.75E+09	2.63E+06	5.76E+08	1.52E+06	‘Fist’
Frame 118	1.84E-01	6.21E-03	8.43E-04	8.68E+09	2.59E+06	5.71E+08	1.48E+06	‘Fist’
Frame 135	1.88E-01	6.05E-03	8.60E-04	8.82E+09	2.66E+06	5.79E+08	1.55E+06	‘Fist’
Frame 149	1.83E-01	6.12E-03	8.47E-04	8.70E+09	2.61E+06	5.74E+08	1.50E+06	‘Fist’
Frame 164	1.87E-01	6.08E-03	8.55E-04	8.78E+09	2.64E+06	5.77E+08	1.53E+06	‘Fist’
Frame 181	1.85E-01	6.15E-03	8.49E-04	8.72E+09	2.60E+06	5.73E+08	1.49E+06	‘Fist’
Frame 196	1.89E-01	6.02E-03	8.62E-04	8.85E+09	2.67E+06	5.81E+08	1.56E+06	‘Fist’
Frame 210	1.82E-01	6.18E-03	8.44E-04	8.65E+09	2.58E+06	5.70E+08	1.47E+06	‘Fist’
Frame 224	1.86E-01	6.10E-03	8.53E-04	8.76E+09	2.63E+06	5.76E+08	1.52E+06	‘Fist’
Frame 239	1.84E-01	6.14E-03	8.46E-04	8.71E+09	2.60E+06	5.72E+08	1.50E+06	‘Fist’

In Table III, the ‘Five’ gesture demonstrates relatively higher values of M1, M2, and M3 compared to the ‘Fist’ gesture. This behaviour is expected due to the open palm configuration, which results in a wider mass distribution and increased shape complexity. The consistency of these lower-order moment values across multiple samples indicates the stability of M1–M3 in capturing the

geometric characteristics of an open hand. Furthermore, variations observed in the higher-order moments (M4–M7) reflect differences in finger spread and orientation, which enhance the discriminative capability of the feature set and support reliable gesture recognition by the machine learning model.

TABLE III. SAMPLE RECOGNITION OF ‘FIVE’ GESTURE

Hu Moment	M1	M2	M3	M4	M5	M6	M7	Label
Frame 348	0.22	0.11	0.03	1.51E+09	4.36E+06	1.72E+09	1.38E+05	‘Fve’
Frame 467	0.42	0.31	0.02	1.08E+09	3.34E+06	1.51E+09	3.53E+05	‘Fve’
Frame 489	0.23	0.12	0.03	1.48E+09	4.21E+06	1.69E+09	1.45E+05	‘Fve’
Frame 512	0.24	0.13	0.04	1.53E+09	4.40E+06	1.75E+09	1.52E+05	‘Fve’
Frame 536	0.21	0.10	0.03	1.47E+09	4.18E+06	1.68E+09	1.33E+05	‘Fve’
Frame 559	0.25	0.14	0.04	1.56E+09	4.52E+06	1.78E+09	1.60E+05	‘Fve’
Frame 583	0.41	0.30	0.02	1.12E+09	3.41E+06	1.54E+09	3.60E+05	‘Fve’
Frame 607	0.43	0.32	0.03	1.10E+09	3.38E+06	1.52E+09	3.55E+05	‘Fve’
Frame 632	0.22	0.11	0.03	1.49E+09	4.29E+06	1.70E+09	1.40E+05	‘Fve’
Frame 658	0.24	0.13	0.03	1.52E+09	4.35E+06	1.73E+09	1.48E+05	‘Fve’

The ‘OK’ gesture is characterized by a distinctive combination of moderate-to-low values in the lower-order moments (M1–M3) and noticeable variations in the higher-order moments. As shown in Table IV, this pattern corresponds to the partially closed loop formed by the thumb and index finger, resulting in a compact yet geometrically complex shape. The small magnitudes of

M1–M3 indicate a concentrated mass distribution, while variations in M4–M7 capture differences in curvature, finger orientation, and loop formation. These properties highlight the effectiveness of hu moment invariants in representing intricate hand shapes and reliably distinguishing the ‘OK’ gesture from visually similar classes.

TABLE IV. SAMPLE RECOGNITION OF ‘OK’ GESTURE

Hu Moment	M1	M2	M3	M4	M5	M6	M7	Label
Frame 735	0.22	0.006	0.0011	2.94E-04	1.76E+07	6.83E+08	1.13E+07	‘OK’
Frame 877	0.21	0.002	0.0008	1.70E-04	2.42E+06	4.52E+08	8.39E+04	‘OK’
Frame 896	0.21	0.002	0.0009	1.68E-04	2.33E+06	4.29E+08	1.66E+05	‘OK’
Frame 912	0.22	0.005	0.0012	2.85E-04	1.68E+07	6.71E+08	1.09E+07	‘OK’
Frame 934	0.20	0.002	0.0007	1.62E-04	2.15E+06	4.18E+08	1.02E+05	‘OK’
Frame 956	0.21	0.003	0.0010	1.75E-04	2.56E+06	4.63E+08	9.12E+04	‘OK’
Frame 978	0.22	0.006	0.0013	2.98E-04	1.82E+07	6.95E+08	1.18E+07	‘OK’
Frame 1001	0.21	0.002	0.0009	1.66E-04	2.28E+06	4.34E+08	1.44E+05	‘OK’
Frame 1023	0.22	0.005	0.0011	2.90E-04	1.73E+07	6.78E+08	1.11E+07	‘OK’
Frame 1045	0.20	0.002	0.0008	1.59E-04	2.07E+06	4.12E+08	9.80E+04	‘OK’

The ‘Thumbs-down’ gesture typically involves a closed fist with an extended thumb pointing downward in Table V. The moment features in this table illustrate hu moment asymmetrical patterns, with distinct values especially in M6 and M7. These features represent the

orientation and spread of the gesture, and their differences between samples suggest directional variance. Despite this, the M1-based model successfully captures the unique profile of the thumb’s positioning, confirming the robustness of the feature set for directional gestures.

TABLE V. SAMPLE RECOGNITION OF ‘THUMBS DOWN’ GESTURE

Hu Moment	M1	M2	M3	M4	M5	M6	M7	Label
Frame 1048	0.20	0.01	0.01	3.92E-04	1.58E+07	3.10E+09	1.45E+06	‘Thumbs-down’
Frame 1052	0.19	0.01	0.02	4.11E-04	1.62E+07	3.18E+09	1.67E+06	‘Thumbs-down’
Frame 1056	0.21	0.02	0.01	3.75E-04	1.71E+07	3.25E+09	1.88E+06	‘Thumbs-down’
Frame 1060	0.20	0.01	0.02	4.28E-04	1.69E+07	3.30E+09	1.73E+06	‘Thumbs-down’
Frame 1064	0.19	0.02	0.01	3.66E-04	1.55E+07	3.05E+09	1.52E+06	‘Thumbs-down’
Frame 1068	0.21	0.01	0.02	4.03E-04	1.74E+07	3.35E+09	1.91E+06	‘Thumbs-down’
Frame 1072	0.20	0.02	0.01	3.88E-04	1.60E+07	3.12E+09	1.64E+06	‘Thumbs-down’
Frame 1076	0.19	0.01	0.02	4.19E-04	1.66E+07	3.22E+09	1.79E+06	‘Thumbs-down’
Frame 1080	0.21	0.02	0.01	3.71E-04	1.70E+07	3.28E+09	1.86E+06	‘Thumbs-down’
Frame 1084	0.20	0.01	0.02	4.07E-04	1.63E+07	3.15E+09	1.58E+06	‘Thumbs-down’

Similar to the previous table, but with the thumb oriented upward, the ‘Thumbs-up’ gesture in Table VI exhibits relatively higher values in M3 and M7. The prominent magnitudes of M6 and M7 reflect the elevation and directional orientation of the thumb, effectively

distinguishing this gesture from the ‘Thumbs-down’ class despite their similar base hand structure. These results demonstrate the sensitivity of hu moment features to rotational and positional variations, making them effective for fine-grained hand gesture classification.

TABLE VI. SAMPLE RECOGNITION OF ‘THUMBS UP’ GESTURE

Hu Moment	M1	M2	M3	M4	M5	M6	M7	Label
Frame 1098	0.21	0.01	0.02	2.65E-04	1.50E+07	5.31E+08	6.10E+06	‘Thumbs-up’
Frame 1102	0.20	0.02	0.01	2.58E-04	1.47E+07	5.12E+08	5.88E+06	‘Thumbs-up’
Frame 1106	0.22	0.01	0.03	2.72E-04	1.55E+07	5.40E+08	6.35E+06	‘Thumbs-up’
Frame 1110	0.21	0.02	0.02	2.61E-04	1.49E+07	5.25E+08	6.02E+06	‘Thumbs-up’
Frame 1114	0.20	0.01	0.02	2.54E-04	1.45E+07	5.05E+08	5.76E+06	‘Thumbs-up’
Frame 1118	0.22	0.02	0.01	2.69E-04	1.58E+07	5.48E+08	6.42E+06	‘Thumbs-up’
Frame 1122	0.21	0.01	0.03	2.63E-04	1.52E+07	5.33E+08	6.15E+06	‘Thumbs-up’
Frame 1126	0.20	0.02	0.02	2.56E-04	1.46E+07	5.10E+08	5.84E+06	‘Thumbs-up’
Frame 1130	0.22	0.01	0.02	2.71E-04	1.56E+07	5.45E+08	6.30E+06	‘Thumbs-up’
Frame 1134	0.21	0.02	0.01	2.60E-04	1.48E+07	5.22E+08	5.98E+06	‘Thumbs-up’

Across all six gesture classes, the moment invariant features (M1–M7) have proven effective in capturing both global and local shape characteristics of hand gestures. Their rotation, scale, and translation invariance allow the system to maintain consistent recognition despite minor hand movement or orientation changes. The distinct feature patterns for each gesture support the use of classical machine learning classifiers, reinforcing the proposed model’s capability to deliver accurate and efficient hand gesture recognition.

B. Implementation and Testing Interface

To verify the dependability and effectiveness of the developed hand gesture recognition system, a comprehensive testing phase was conducted on the user interface component. This interface serves as the front-end platform that allows users to input hand gestures in real-time or image-based motion and capture. The system was designed to classify six predefined gestures ‘Blank’, ‘Fist’, ‘Five’, ‘OK’, ‘Thumbs up’, and ‘Thumbs down’ based on moment invariant features extracted from binary silhouette images.

The testing process aimed to evaluate both the functional performance and classification consistency of the interface when integrated with the trained machine learning models. Key aspects assessed during this phase

included system responsiveness, accuracy of gesture display and prediction, data pre-processing time, and feedback clarity for the end user. The outcomes of this testing phase confirm whether the interface accurately reflects the underlying gesture classification engine, ensuring smooth human-computer interaction and gesture-based command execution.

Fig. 4 illustrates the developed testing interface integrated with the SVM classifier, which serves as the core model for recognizing hand gestures using moment invariant features. The interface allows users to upload or capture a gesture image, from which moment-based descriptors (M1–M7) are automatically extracted and displayed. These features are then forwarded to the trained SVM model for classification.

The interface was designed to emphasize simplicity and interpretability key strengths of employed by the SVM algorithm testing interface. With processing the input, and system from interface can provides the predicted gesture class and displays by the model in Fig. 4. The interface was designed with a simple layout to present based classification process clearly. Using moment invariant feature extraction, the system is able to generate the appropriate gesture class. This interface enables users to easily identify the predicted class.

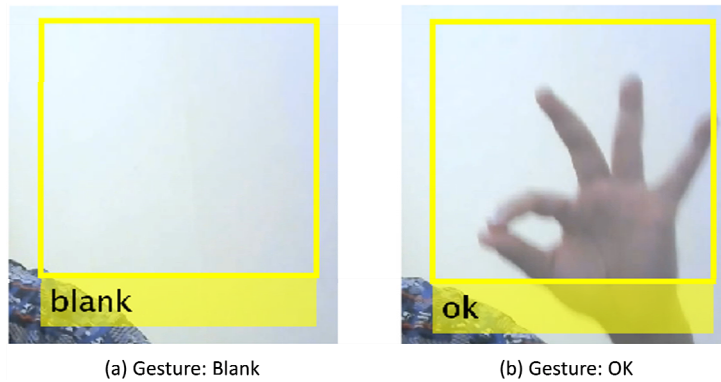


Fig. 4. Sample of the Support Vector Machine (SVM) algorithm testing interface: (a). Gesture: Blank; (b). Gesture: OK.

The interface in Fig. 5 was designed with a simple and clear layout to support the implementation of Decision Tree (DT) algorithm. After processing input, the interface system generates the resulting gesture class based on the moment invariant features extracted from the image, producing results that align with the classes gesture

categories. The interface presents the classification output produced by the model, providing a concise visualization of the gesture class predicted from the extracted features and how each moment feature contributes to the final classification.

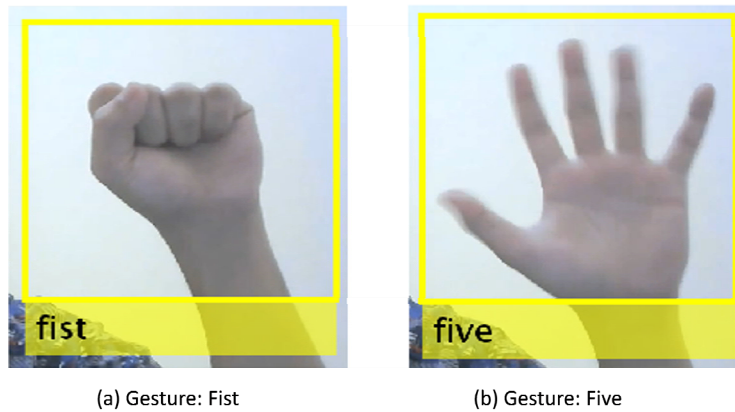


Fig. 5. Sample of the Decision Tree (DT) algorithm testing interface: (a). Gesture: Fist; (b). Gesture: Five.

Fig. 6 illustrates the testing interface for the KNN classifier, developed to support gesture recognition based on moment invariant features. The system is designed to process an input image, extract its seven statistical shape

descriptors (M1–M7), and compare them with stored training data using the euclidean distance metric, enabling the classifier to accurately generate the gesture class based on the extracted features.

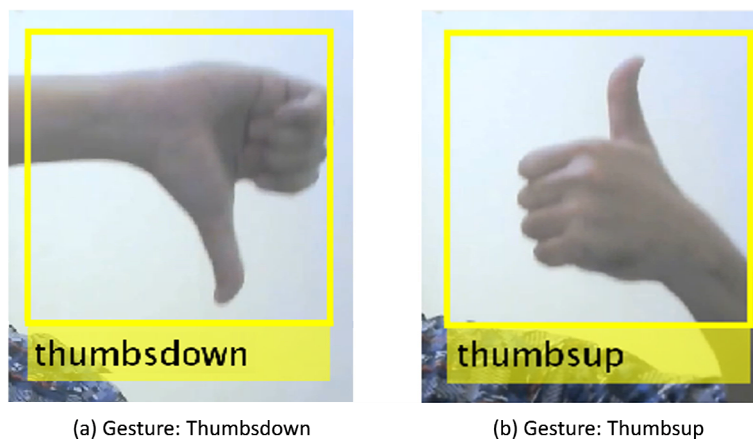


Fig. 6. Sample of the K-Nearest Neighbor (KNN) algorithm testing interface: (a). Gesture: Thumbsdown; (b). Gesture: Thumbsup.

The interface is designed to be intuitive and informative, providing real-time feedback on gesture

predictions. The output includes both the predicted gesture label and the corresponding confidence score,

offering transparency regarding the decision-making process of the classifier. This feature is particularly important in gesture-based systems where real-time interaction and accuracy are critical. Furthermore, the interface supports testing of multiple gesture types including ‘Fist’, ‘Five’, ‘OK’, ‘Thumbs up’, ‘Thumbs down’, and ‘Blank’ and logs each recognition event for subsequent analysis.

This enables users and researchers to validate the robustness of the SVM model under various input conditions. The clear visualization of the input image, extracted features, and model output promotes better interpretability and debugging during system evaluation. By integrating the KNN classifier into a structured interface, this system demonstrates the effectiveness of instance-based learning in classifying moment-invariant representations of hand gestures, particularly when gesture boundaries are subtle and overlapping in feature space. The interface was designed to emphasize simplicity and interpretability—key strengths of the DT algorithm. Upon processing the input, the system provides the predicted gesture class and displays the decision path traversed by the model. This visual traceability highlights the hierarchical structure of the decision-making process, helping users and researchers understand how each moment feature contributes to the final classification.

C. Comparison Result for Optimal Algorithm

Split dataset: Training 80, testing 20.

Table VII presents a performance comparison of three algorithms (SVM, DT, and KNN) based on both training and testing accuracy. Among them, SVM achieved the

highest testing accuracy of 81.41%, reflecting strong generalization ability. KNN and DT obtained testing accuracies of 78.00% and 77.25%, respectively. These results highlight the advantages of the proposed hand gesture recognition framework, which leverages moment invariant features to extract stable, rotation-invariant and scale-invariant shape descriptors. The proposed feature extraction method demonstrates high robustness in capturing gesture patterns across varying image conditions, leading to improved classifier performance. This confirms that moment invariants provide a compact and discriminative representation of hand shapes, offering a reliable basis for consistent recognition across different machine learning algorithms.

TABLE VII. COMPARISON RESULT

Method	Training Accuracy (%)	Testing Accuracy (%)
SVM	99.83	81.41
DT	99.25	77.25
KNN	98.75	78.00

Fig. 7 illustrates the performance of three algorithms (SVM, DT, and KNN) within the framework of a hand gesture recognition system based on moment invariant features. These features are designed to generate shape descriptors that are stable against rotation and scale, thereby improving the consistency of gesture recognition. The combination of moment invariant features with the three algorithms generally yields good performance, particularly with SVM. These features have been proven to provide a compact and discriminative representation of hand shapes and are resilient to variations in image rotation and scale, thus enhancing the effectiveness.

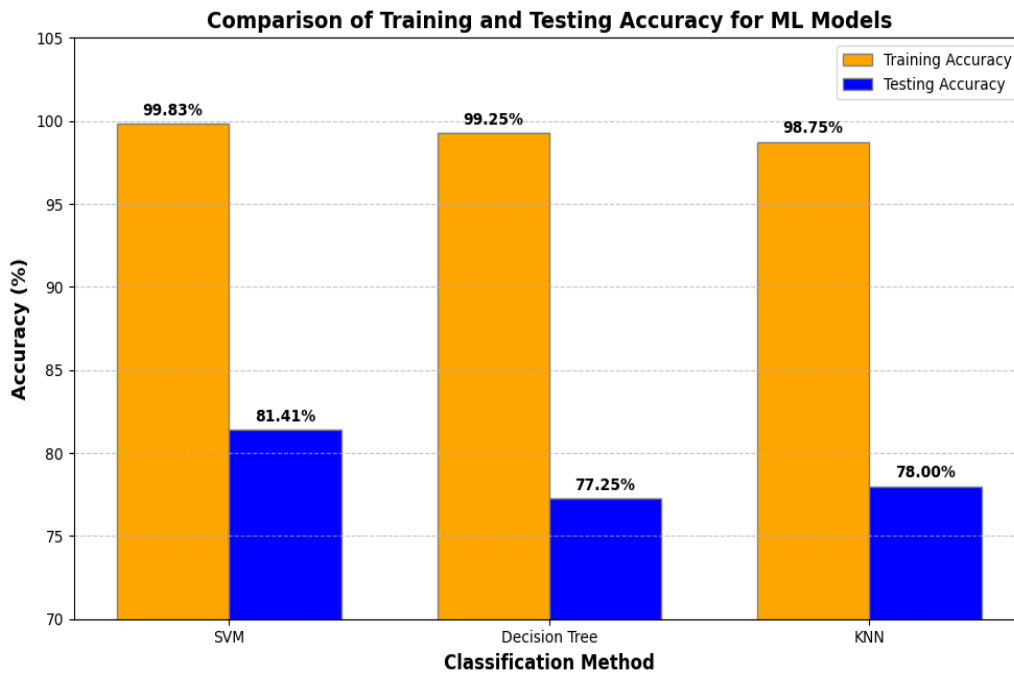


Fig. 7. Training and testing accuracy.

The SVM model is highly effective at recognizing gestures in the training data, as indicated by a training accuracy of 99.83% (with only about 2–3 misclassified

instances out of thousands). However, when tested on 1200 gesture samples, the accuracy testing of 81.41%, meaning approximately 223 gestures were misclassified.

Gesture classes with similar finger shapes or overlapping visual features (such as thumb direction or circular finger positions) tend to confuse the model more frequently. To confirm this precisely, we need to examine the confusion

matrix from the testing phase—this will clearly show which gestures are often confused with others shown in Fig. 8.

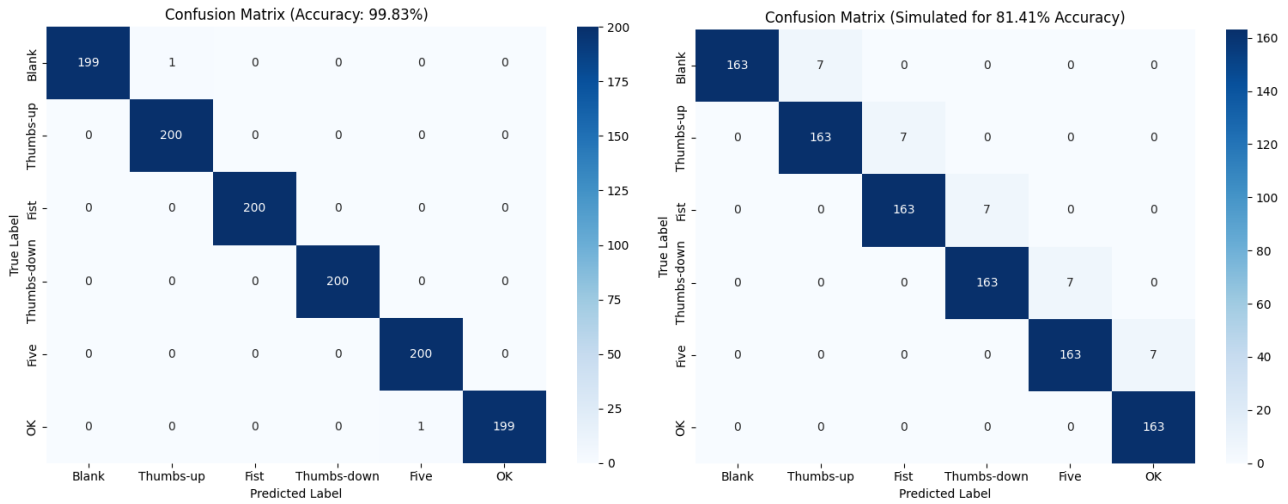


Fig. 8. Confusion matrix training and testing accuracy SVM.

The confusion matrix provides a detailed view of how well the DT classifier performs for each individual gesture class during testing. Out of 1200 gesture samples,

the model correctly classified approximately 927 gestures and misclassified around 273 gestures, resulting in an overall accuracy of 77.25%, shown in Fig. 9.

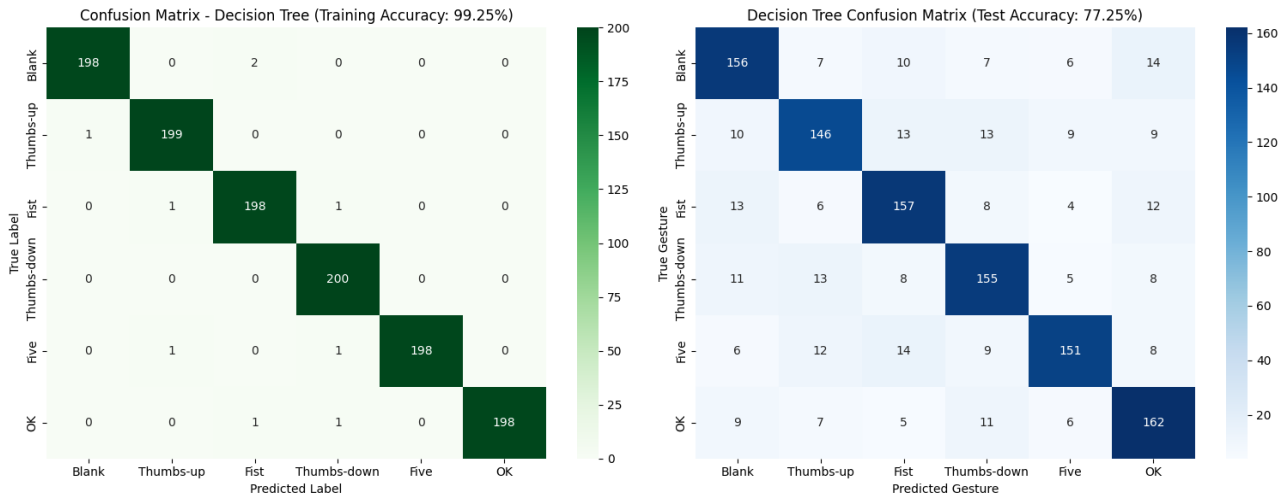


Fig. 9. Confusion matrix training and testing accuracy Decision Tree.

The KNN model achieved 78.00% accuracy on a test set of 1200 hand gesture samples across six classes, shown in Fig. 10. Misclassifications mainly occurred between visually similar gestures such as “Fist”, “Thumbs-up”, and “OK”, which share overlapping finger positions. The “Blank” gesture was also occasionally confused with active gestures like “Fist” or “Thumbs-down”. These errors indicate that while KNN performs reasonably well, it struggles to distinguish between gestures with subtle visual differences.

The model achieved perfect scores (Recall, Precision, and F1-Score of 1.0000) for both the Fist and Thumbs-down classes in Table VIII, indicating no misclassifications occurred for these specific gestures.

However, the model recorded a minimal number of classification errors across four classes: “Blank”, “Thumbs-up”, “Five”, and “OK”.

Misclassification between “Blank” and “Thumbs-up”: One blank sample was incorrectly classified as thumbs-up (False Negative (FN) for “Blank”), and conversely, the high precision score for thumbs-up was slightly reduced because it received a False Positive (FP) from the blank class. This suggests the primary point of confusion lies between a relaxed hand and a simple gesture involving the thumb. Misclassification between “Five” and “OK”: Similarly, a single “OK” sample was incorrectly classified as five (false negative for “OK”), and the “Five” class received a FP from the ok class. This

particular error highlights the model’s single struggle: distinguishing between gestures that involve variations in finger curvature or topology (the small circular shape in

“OK” vs. the fully open fingers in “Five”), shown in Table VIII.

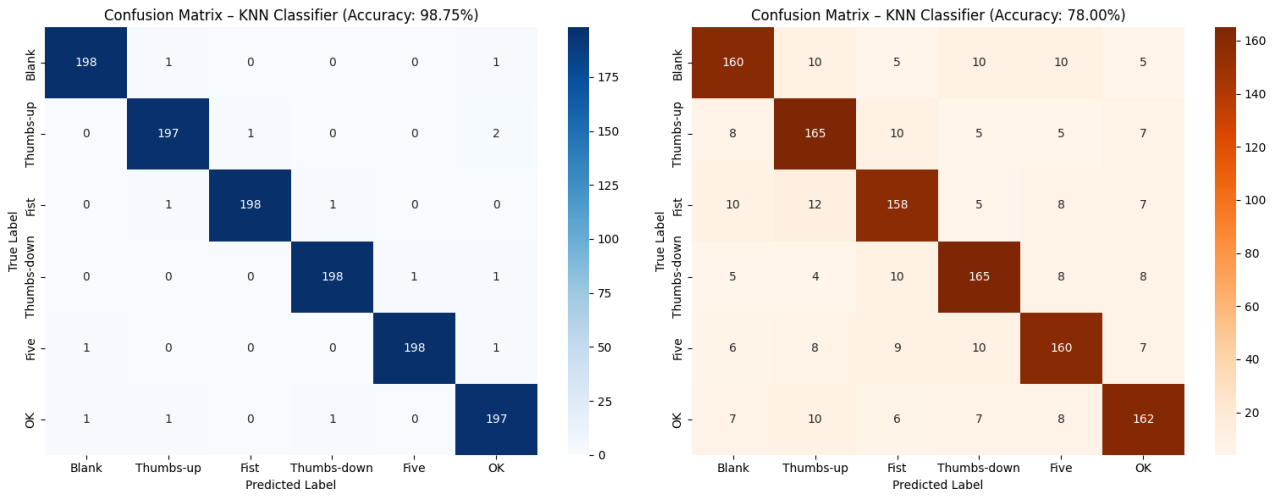


Fig. 10. Confusion matrix training and testing accuracy KNN.

TABLE VIII. CLASSIFICATION MODEL PERFORMANCE EVALUATION METRICS SVM

Class	Recall	Precision	F1-Score	Error Analysis (FN and FP)
Blank	0.9950	1.0000	0.9975	Blank sample was misclassified as Thumbs-up.
Thumbs-up	1.0000	0.9950	0.9975	Blank sample was misclassified as Thumbs-up (FP).
Fist	1.0000	1.0000	1.0000	Perfect
Thumbs-down	1.0000	1.0000	1.0000	Perfect
Five	1.0000	0.9950	0.9975	OK sample was misclassified as Five (FP).
OK	0.9950	1.0000	0.9975	OK sample was misclassified as Five.

In summary, the SVM classifier exhibits remarkable robustness and superior generalization ability, with only two total errors across the entire dataset (one confusion pair between Blank/Thumbs-up and one confusion pair between OK/Five). This performance confirms the suitability and effectiveness of using moment invariant features for robust hand gesture recognition when coupled with an effective boundary-seeking model like SVM.

In Table IX, confusion with ‘Fist’ and ‘Blank’: The blank class experienced two FN, being misclassified as Fist. This suggests that the resting hand shape (Blank) sometimes presents a contour area or compactness like a not fully closed fist, fooling the DT. Conversely, the “Fist” class received two FP from the “Blank” and “OK” classes, indicating its decision boundary is slightly blurred by similar features.

TABLE IX. CLASSIFICATION MODEL PERFORMANCE EVALUATION METRICS KNN

Class Gesture	Recall	Precision	F1-Score	Error Analysis (FN and FP)
Blank	0.9900	0.9950	0.9925	Blank samples were misclassified as Fist. Thumbs-up sample was misclassified as Blank.
Thumbs-up	0.9950	0.9950	0.9950	Thumbs-up sample was misclassified as Blank. Fist sample was misclassified as Thumbs-up.
Fist	0.9900	0.9900	0.9900	Fist samples were misclassified as Thumbs-down and OK samples (Blank and OK) were misclassified as Fist.
Thumbs-down	1.0000	0.9901	0.9950	Perfect Recall. Samples (Fist and Five) were misclassified as Thumbs-down.
Five	0.9900	0.9950	0.9925	Five samples were misclassified as Thumbs-down and OK sample was misclassified as Five.
OK	0.9900	0.9950	0.9925	OK samples were misclassified as Fist and Thumbs-down. Five sample was misclassified as OK.

Ambiguity in complex gestures: The most challenging class to correctly predict (in terms of FP) was Thumbs-down, which received two FP from the Fist and Five classes. This demonstrates overall the results confirm that the KNN classifier provides reliable and accurate gesture classification, with only limited confusion between

classes that share similar hand shape characteristics, features extracted from other complex hand shapes.

Cross class confusion: Errors were generally distributed across low complexity gestures “Five” and “OK” both suffered from FN, being confused with Thumbs-down. This indicates that the moment invariants

for these three shapes sometimes overlap, possibly due to minor variations in hand rotation or perspective that slightly affect the projected shape features. The “OK” gesture was also confused with “Fist” and “Thumbs-down”, suggesting that the distinctive ring shape in the “OK” gesture is not always strongly represented by the global moment invariants M1–M7, leading to classification errors.

The analysis of the metrics highlights the primary confusion points.

Weakest performing class (OK): The “OK” gesture is the most challenging for the Decision Tree model, possessing the lowest recall (0.9850) and the lowest precision (0.9801) in Table X. This is primarily due to the highest number of errors: Four FP the “OK” class incorrectly received four samples from four different classes (Blank, Thumbs-up, Thumbs-down, and Five). This indicates that the feature representation of “OK” often overlaps with the feature clusters of many other gestures in the Decision Tree space.

TABLE X. CLASSIFICATION MODEL PERFORMANCE EVALUATION METRICS DECISION TREE

Class (Gesture)	Recall	Precision	F1-Score	Error Analysis (FN and FP)
Blank	0.9900	0.9900	0.9900	Misclassified as Thumbs-up and OK. 2 samples (Five and OK) were misclassified as Blank.
Thumbs-up	0.9850	0.9949	0.9900	Misclassified as Fist and OK sample was misclassified as Thumbs-up.
Fist	0.9900	0.9950	0.9925	Misclassified as Thumbs-down and Five. Thumbs-up sample was misclassified as Fist.
Thumbs-down	0.9900	0.9900	0.9900	Misclassified as Five and OK. 2 samples (Fist and OK) were misclassified as Thumbs-down.
Five	0.9900	0.9950	0.9925	Misclassified as Blank and OK. Thumbs-down sample was misclassified as Five.
OK	0.9850	0.9801	0.9825	Misclassified as Blank, Thumbs-up, and Thumbs-down. 4 samples (Blank, Thumbs-up, Thumbs-down, Five) were misclassified as OK.

False Negatives (three FN): “OK” samples were incorrectly classified as Blank, Thumbs-up, and Thumbs-down, suggesting high ambiguity.

Confusion with “Blank” and “Thumbs-up”: The “Blank” and “Thumbs-up” classes show significant two-way confusion. “Blank” samples are incorrectly classified as “Thumbs-up” (one FN), and conversely, “Thumbs-up” samples are incorrectly classified as “Blank” (one FN). These two resting/simple gestures appear to share the closest feature cluster boundaries. Both “Fist” and “Thumbs-down” show high performance but still suffer from being confused with other similar shapes (“Fist” being confused with “Thumbs-down” and “Five”), and “Thumbs-down” being confused with “Five” and “OK”.

V. CONCLUSION

This study used the moment invariant method to effectively identify and classify hand gestures based on predefined labels, demonstrating strong model performance. Among the tested algorithms, SVM yielded the highest accuracy, outperforming both DT and KNN on training and testing datasets. The SVM model is particularly recommended due to its exceptional ability to accurately classify all gesture categories. Evaluation results show that SVM achieved a test accuracy of 81.41%. The DT algorithm followed closely with a test accuracy of 77.25%, while KNN test accuracy of 78.00%. Moment invariants were selected as the core feature extraction method due to their effectiveness in capturing shape-based features that are invariant to rotation, translation, and scaling critical for hand gesture data that can vary in position and orientation. Moreover, moment invariants are computationally lightweight, making them highly suitable for systems with limited resources, such as embedded real applications. This approach shows strong potential for practical implementation in real-time gesture-based monitoring, especially for assessing motor limitations in post-stroke patients and individuals with

tremor symptoms. By leveraging lightweight algorithms and efficient feature extraction, the system could support rehabilitation processes and clinical assessments without requiring high computational resources.

However, certain limitations remain, including restricted dataset diversity, environmental variability and the need for further validation using clinical or real-world data. Future work will focus on addressing these challenges and integrating multimodal data to improve the model’s robustness and generalization in medical and rehabilitation applications.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Fauziah conducted the research, developed the program code, performed the data analysis and visualization, and drafted the original manuscript. Riadi Marta Dinata contributed to the literature review. Dhieka Avriia Lantana conducted the validation of the experimental results, and Billy Hendrik assisted in refining the final manuscript. All authors reviewed and approved the final version for submission.

REFERENCES

- [1] X. Wang, D. Veeramani, F. Dai, and Z. Zhu, “Context-aware hand gesture interaction for human-robot collaboration in construction,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 39, no. 22, pp. 3489–3504, 2024. doi: 10.1111/mice.13202
- [2] W. Cao, “Application of the support vector machine algorithm based gesture recognition in human-computer interaction,” *Informatica*, vol. 43, no. 1, pp. 123–127, 2019. doi: 10.31449/inf.v43i1.2602
- [3] S. Mohankumar, G. M. Thimmaiah, N. Chikkaguddaiah, and V. B. Gowda, “MODFAT: Moving object detection by removing shadow based on fuzzy technique with an adaptive thresholding method,” *Revue d’Intelligence Artificielle*, vol. 35, no. 2, pp. 177–183, 2021. doi: 10.18280/ria.350210

- [4] S. Wang, J. Zheng, Z. Huang *et al.*, “Integrating computer vision to prosthetic hand control with sEMG: Preliminary results in grasp classification,” *Frontiers in Robotics and AI*, vol. 9, 948238, 2022. doi:10.3389/frobt.2022.948238
- [5] M. Rizwan, S. U. Haq, N. Gul *et al.*, “Appearance based dynamic hand gesture recognition using 3D separable convolutional neural network,” *Computers, Materials & Continua*, vol. 76, no. 1, pp. 1213–1247, 2023. doi: 10.32604/cmc.2023.038211
- [6] C. Wu, K. Wang, Q. Cao *et al.*, “Development of a low-cost wearable data glove for capturing finger joint angles,” *Micromachines*, vol. 12, no. 7, p. 771, 2021. doi: 10.3390/mi12070771
- [7] X. Liu, C. Dan, J. Liu, and Y. Yuan, “Effects of exercise on the inter-session accuracy of EMG-based hand gesture recognition,” *Bioengineering*, vol. 11, no. 8, p. 811, 2024. doi: 10.3390/bioengineering11080811
- [8] M. Yoo, Y. Na, H. Song *et al.*, “Motion estimation and hand gesture recognition-based human-UAV interaction approach in real time,” *Sensors*, vol. 22, no. 7, p. 2513, 2022. doi: 10.3390/s22072513
- [9] J. Xiao, H. Li, and J. Zhao, “A lightweight and efficient gesture recognizer for traffic police commands using spatiotemporal feature fusion,” *Scientific Reports*, vol. 15, no. 1, 18256, 2025. doi: 10.1038/s41598-025-02833-y
- [10] J. P. Sahoo, A. J. Prakash, P. Plawiak, and S. Samantray, “Real-time hand gesture recognition using fine-tuned convolutional neural network,” *Sensors*, vol. 22, no. 3, p. 706, 2022. doi: 10.3390/s22030706
- [11] S. A. Alazawi, R. A. Hasan, and A. A. Abdulhameed, “Hand recognition system based on invariant moments features,” in *Proc. Research in Intelligent and Computing in Engineering: Select Proceedings of RICE 2020*, 2020, vol. 1254, pp. 429–440. https://doi.org/10.1007/978-981-15-7527-3_42
- [12] J. Qi, L. Ma, Z. Cui *et al.*, “Computer vision-based hand gesture recognition for human-robot interaction: A review,” *Complex Intell. Syst.* vol. 10, no. 1, pp. 1581–1606, 2024. <https://doi.org/10.1007/s40747-023-01173-6>
- [13] I. Elmagrouni, A. Ettaoufik, S. Aouad, and A. Maizate, “A deep learning framework for hand gesture recognition and multimodal interface control,” *Recent Advances in Intelligent Automation and Robotics*, vol. 37, no. 4, pp. 881–887, 2023. <https://doi.org/10.18280/ria.37040>
- [14] V. Chang, R. O. Eniola, L. Golightly *et al.*, “An exploration into human-computer interaction: Hand gesture recognition management in a challenging environment,” *SN Computer Science*, vol. 4, no. 5, p. 441, 2023. <https://doi.org/10.1007/s42979-023-01751-y>
- [15] A. Musa, H. A. Kakudi, M. Hassan *et al.*, “Lightweight deep learning models for edge devices—A survey,” *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 17, pp. 189–206, 2025.
- [16] S. Shivani and S. B. Gupta, “A comprehensive analysis of recognition of hand gestures using machine learning,” *Makara Journal of Technology*, vol. 29, no. 1, p. 5, 2025. <https://scholarhub.ui.ac.id/mjt/vol29/iss1/5>
- [17] J. Li, C. Li, J. Han, Y. Shi, G. Bian, and S. Zhou, “Robust hand gesture recognition using HOG-9ULBP features and SVM model,” *Electronics*, vol. 11, no. 7, p. 988, 2022. doi: 10.3390/electronics11070988
- [18] M. Oudah, A. Al-Naji, and J. Chahl, “Hand gesture recognition based on computer vision: A review of techniques,” *Journal of Imaging*, vol. 6, no. 8, p. 73, 2020. doi: 10.3390/jimaging6080073
- [19] M. W. Nasrudin, N. S. Yaakob, N. A. A. Rahim *et al.*, “Moment invariants technique for image analysis and its applications: A review,” *Journal of Physics: Conference Series*, vol. 1962, no. 1, 012028, 2021. doi: 10.1088/1742-6596/1962/1/012028
- [20] I. Elouariachi, R. Benouini, K. Zenkouar, A. Zarghili and H. E. Fadili, “Explicit quaternion krawtchouk moment invariants for finger-spelling sign language recognition,” in *Proc. 2020 28th European Signal Processing Conference (EUSIPCO)*, 2021, pp. 620–624. doi: 10.23919/Eusipco47968.2020.9287845
- [21] M. S. Sarowar, N. E. J. Farjana, M. A. I. Khan *et al.*, “Hand gesture recognition systems: A review of methods, datasets, and emerging trends,” *International Journal of Computer Applications*, vol. 975, 8887, 2025.
- [22] V. Madaan and N. Sharma, “Hand gesture recognition using convolutional neural network,” in *Proc. 2024 3rd Int. Conf. Adv. Comput., Inf. Commun. Technol. (ACOIT)*, 2024, pp. 1–6. doi: 10.1109/acoit62457.2024.10939661
- [23] N. Carion, F. Massa, G. Synnaeve *et al.*, “End to end object detection with transformer,” in *Proc. the European Conference on Computer Vision (ECCV)*, 2020, pp. 213–229. https://doi.org/10.1007/978-3-030-58452-8_13
- [24] R. Krebs, S. S. Bagui, D. Mink, and S. C. Bagui, “Applying multi-class support vector machines: One-vs.-one vs. one- vs.-all on the UWF-ZeekDataFall22 ataset,” *Electronics*, vol. 13, no. 19, 3916, 2024. <https://doi.org/10.3390/electronics13193916>
- [25] B. Lin, C. Liu, and D. Miao, “An improved decision tree algorithm based on boundary mixed attribute dependency,” *Appl. Intell.*, vol. 54, pp. 2136–2153, 2024. <https://doi.org/10.1007/s10489-023-05238-4>
- [26] R. K. Halder, M. N. Uddin, M. A. Uddin *et al.*, “Enhancing K-nearest neighbor algorithm: A comprehensive review and performance analysis of modifications,” *Journal of Big Data*, vol. 11, no. 1, p. 113, 2024. <https://doi.org/10.1186/s40537-024-00973-y>
- [27] F. Fauziah, E. P. Wibowo, S. Madenda *et al.*, “Identification of hand motion using background subtraction method and extraction of image binary with back propagation neural network on skeleton model,” *Journal of Physics: Conference Series*, vol. 978, no. 1, 012020, 2018. doi: 10.1088/1742-6596/978/1/012020
- [28] A. Lu and B. H. S. Asli, “Seismic image identification and detection based on tchebichef moment invariant,” *Electronics*, vol. 12, no. 17, 3692, 2023. <https://doi.org/10.3390/electronics12173692>
- [29] E. G. Mounq, C. J. Hou, M. M. Sufian *et al.*, “Empirical fusion: Deploying moment invariants and deep learning in COVID-19 diagnostics,” in *Proc. Internet of Things and Artificial Intelligence for Smart Environments*, 2024, pp. 181–201. https://doi.org/10.1007/978-981-97-1432-2_11
- [30] B. Zhang, Y. Zhang, J. Liu, and B. Wang, “FGFF descriptor and modified hu moment-based hand gesture recognition,” *Sensors*, vol. 21, no. 19, 6525, 2021. <https://doi.org/10.3390/s21196525>

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License (CC-BY-4.0), which permits use, distribution and reproduction in any medium, provided that the article is properly cited, the use is non-commercial and no modifications or adaptations are made.