Journal of Electronics, Electromedical Engineering, and Medical Informatics
Multidisciplinary: Rapid Review: Open Access Journal Vol. 7, No. 2, April 2025, pp: 341-351; eISSN: 2656-8632

Manuscript received December 8, 2024; Revised February 10, 2025; Accepted March 1, 2025; date of publication March 4, 2025
Digital Object Identifier (**DOI**): https://doi.org/10.35882/jeeemi.v7i2.645
Copyright © 2025 by the authors. This work is an open-access article and licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0).
How to cite: Salamet Nur Himawan, Adi Suheryadi, Kurnia Adi Cahyanto, Filemon Sitanggang and Kiki Adi Pamungkas, "Comparative Analysis of Texture Based and Geometric Feature Extraction Techniques for Facial Paralysis Classification", Journal of Electronics, Electromedical Engineering, and Medical Informatics, vol. 7, no. 1, pp. 341-351, January 2025.

Comparative Analysis of Texture Based and Geometric Feature Extraction Techniques for Facial Paralysis Classification

Salamet Nur Himawan, Adi Suheryadi, Kurnia Adi Cahyanto, Filemon Sitanggang and Kiki Adi Pamungkas

Informatic Department, Politeknik Negeri Indramayu, Indonesia

Corresponding author: Salamet Nur Himawan (e-mail: snhimawan@polindra.ac.id).
This work was supported by Center of Research and Community Service (P3M), Politeknik Negeri Indramayu under Kegiatan PUKTI 2024.

ABSTRACT Facial paralysis significantly affects a person's ability to communicate and perform essential functions. Facial paralysis classification plays a vital role in the diagnosis and monitoring of facial disorders. Traditional diagnostic methods often rely on subjective evaluations, leading to inconsistent outcomes. The aim of this study is to evaluate and compare various feature extraction techniques to enhance the accuracy and efficiency of facial paralysis classification. The primary contribution of this research lies in its comprehensive analysis of texture-based (Local Binary Patterns, Histogram of Oriented Gradients, Gabor filters) and geometric feature extraction methods, providing insights into their respective strengths and limitations for facial paralysis detection. This study utilizes the YouTube Facial Palsy (YFP) dataset, comprising annotated images of paralyzed and non-paralyzed faces. Preprocessing included resizing images to 128x128 pixels to standardize inputs. Feature extraction methods were applied to the dataset, and the extracted features were classified using machine learning algorithms, including Support Vector Machines (SVM), Random Forest (RF), and k-Nearest Neighbors (KNN). Model performance was evaluated using accuracy, precision, recall, and F1-score metrics. The best-performing method achieved an accuracy of 85% using HOG features combined with KNN. The findings highlight that texture-based methods, particularly HOG, excel in capturing subtle asymmetries, while geometric features offer computational efficiency and interpretability with fewer extracted features. This study underscores the importance of selecting suitable feature extraction methods based on task requirements, and emphasizes the potential of hybrid approaches to leverage the strengths of different methods. Future research should explore advanced geometric descriptors and integrate hybrid models to enhance clinical applicability.

INDEX TERMS Facial Paralysis, Feature Extraction, Support Vector Machine, Random Forest, K-Nearest Neighbors, Local Binary Pattern, Histogram Of Oriented Gradients, Gabor Filters

I. INTRODUCTION

Facial paralysis is a severe neurological disorder characterized by the complete absence of voluntary muscle activity on one side of the face, which leads to significant physical and psychological consequences [1]. It may be caused by a variety of factors, including cerebrovascular accidents, physical injuries, infections, and Bell's palsy [2], [3]. Facial paralysis classification is a critical task in medical imaging, as early detection can aid in diagnosing conditions such as stroke, Bell's palsy, and other neurological disorders.

In this case, the sensitivity of the model to capture the above symptoms is very necessary considering that this disease is related to the nervous system, which is certainly connected to daily human activities. Medical solutions by visiting or consulting with a neurologist are indeed the right actions to help patients in making an early diagnosis.

Traditional methods for classifying facial paralysis have traditionally relied on subjective grading systems and manual measurement techniques. The House-Brackmann scale and The Burres-Fisch system is widely used as a grading system to assess the severity of facial nerve dysfunction. The House-Brackmann scale assigns six levels to the severity of facial nerve impairment based on the observed facial movements [4]. The Burres-Fisch system is a research-based system that relies on linear measurements of displacement of reference points on the face [5]. The limitations of traditional methods, such as subjectivity, lack of precision, labor-intensiveness, and susceptibility to human error, highlight the necessity for the development of more objective and reliable diagnostic techniques.

To overcome these limitations, various computational approaches have been explored to automate facial paralysis classification, ranging from traditional handcrafted features to modern machine learning and deep learning techniques. Traditional computational methods for facial paralysis classification primarily relied on geometric and image subtraction techniques to assess facial asymmetry. These methods focused on extracting structural information such as distances, angles, and ratios between key facial landmarks to identify deviations caused by paralysis.

Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), Gabor filters, and Geometric features have been widely applied for face-related analysis, including the detection and classification of facial paralysis. LBP is particularly effective in capturing local texture patterns by encoding pixel intensity differences into binary descriptors, making it robust to grayscale changes and efficient for computational tasks. Several studies have shown the effectiveness of LBP in facial recognition [6], [7], showcasing its ability to detect fine-grained texture variations relevant for asymmetry analysis in facial paralysis. Similarly, HOG has proven to be a powerful tool for capturing structural and gradient information [8][9], highlighting its potential for identifying facial asymmetries and patterns indicative of paralysis.

Gabor filters, known for their ability to capture multi-scale and multi-orientation texture features, have been extensively used in facial analysis to enhance the representation of fine texture details, as shown in the work of Verma [10] and Ou [11]. On the other hand, Geometric features focus on the spatial arrangement and relationships between facial landmarks, effectively capturing asymmetries caused by paralysis. Wang et al utilized geometric measurements to assess facial symmetry, demonstrating their utility in paralysis detection [3]. Other research has been developed in the process of detecting facial paralysis using geometric features [12], [13].

Recent advancements in machine learning have enabled these feature extraction methods to be integrated with powerful classifiers, such as Logistic Regression, Support Vector Machines (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN), significantly improving the accuracy of image classification tasks. Logistic Regression has been extensively used for its interpretability and efficiency in binary classification tasks [14], [15]. SVM excels in handling high-dimensional data and has been widely applied in texture-based and geometric classification tasks [15], [16], [17]. Random Forest demonstrates its strength in handling diverse feature sets and noise, making it a reliable choice for image classification tasks [18], [19], [20]. KNN, with its simplicity and adaptability to non-linear data, has also been successfully applied in facial image recognition and classification [21], [22], [23].

Recently, deep learning models such as convolutional neural networks (CNNs) have demonstrated remarkable performance in automated detection and classification tasks [24], [25]. Unlike traditional approaches that require manual feature extraction, CNNs automatically learn hierarchical features directly from raw image data. This feature learning capability reduces the reliance on handcrafted features and improves classification performance [26], [27]. However, deep learning models often require large annotated datasets and significant computational resources, which may limit their feasibility in certain applications [27], [28].

Feature extraction plays a crucial role in the process of facial paralysis classification by capturing information about texture, shape, and structural asymmetries within an image. This study builds on these previous approaches, conducting a comparative analysis of feature extraction techniques. Texture-based methods include LBP, HOG, Gabor and Geometric methods to evaluate their effectiveness in facial paralysis classification. Texture-based methods are computationally intensive and sensitivity to parameter tuning make them less practical for real-time applications and Geometric methods are limited in capturing fine-grained texture details and heavily depend on the accuracy of landmark detection. Current studies predominantly focus on either texture-based or Geometric methods, but few explore the integration of these complementary approaches. By comparing Texture-based methods and Geometric methods, this study aims to identify optimal feature extraction strategies that balance performance, and computational efficiency.

This study provides a comprehensive comparison of texture-based methods, including Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), Gabor filters, and Geometric features. Additionally, the study highlights the trade-offs between computational efficiency and classification accuracy, emphasizing the potential of hybrid models that integrate texture-based and geometric features to achieve both high performance and practicality. Furthermore, this research identifies opportunities for developing feature extraction techniques that balance accuracy and computational efficiency, making them suitable for future real-world applications. The study not only advances the understanding of feature extraction techniques but also provides a foundation for developing accurate, efficient, and scalable diagnostic tools for facial paralysis, paving the way for future innovation in hybrid feature extraction approaches.

II. METHODOLOGY

The methodology follows a structured pipeline for feature extraction and classification of facial paralysis, as shown in FIGURE 1. It begins with data collection and data preprocessing. Next, feature extraction uses LBP, HOG, Gabor filters, and geometric feature to identify key facial patterns. The classification model is trained using machine learning techniques, LR, SVM, RF and KNN, followed by evaluation using accuracy, precision, recall and F1-score metrics. Beginning with data collection, where images are sourced from public datasets and the YouTube Facial Palsy

(YFP) database. The next step, preprocessing, involves face detection, resizing and grayscaling. Following this, feature extraction employs techniques such as LBP, HOG, Gabor filters, and geometric features to capture texture, gradient, frequency, and spatial details of the facial images. The classification phase utilizes LR, SVM, RF and KNN, trained and tested to differentiate between healthy and paralyzed facial images. Finally, the system undergoes evaluation using metrics accuracy, precision, recall, and F1-Score to assess performance and reliability.

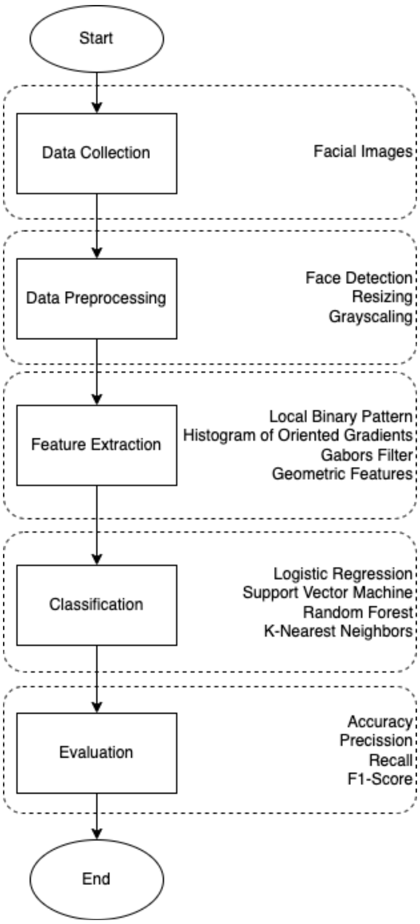

FIGURE 1. Methodology for feature extraction and classification of facial paralysis

A. DATA COLLECTION

The dataset used in this research for facial paralysis detection consists of publicly available images sourced from the internet as well as the YouTube Facial Palsy (YFP) database [29]. The YFP dataset, specifically curated for facial paralysis studies, includes labeled images of individuals with varying degrees of facial paralysis and healthy subjects. This dataset ensures a wide range of conditions, from mild to severe paralysis, making it ideal for robust and comprehensive classification. The publicly available images were carefully selected to supplement the YFP database, ensuring diversity in lighting, angles, and facial expressions, which mimic real-world conditions. The YFP dataset can be accessed through the following URL: https://sites.google.com/view/yfp-database.

The dataset includes two classes, such as healthy (non-paralysis) and unhealthy (paralysis). The images are captured under different conditions, mimicking real-life scenarios. This variability challenges the models to generalize effectively, making the classification system more reliable and applicable in practical settings. The inclusion of publicly available data alongside the YFP dataset enhances the study’s scope, paving the way for the development of a scalable and effective system for facial paralysis detection.

B. DATA PREPROCESSING

Preprocessing plays a critical role in preparing raw image data for effective analysis and classification. The process begins with face detection, dlib are employed to locate and isolate the facial region of interest from the background. This ensures that only relevant portions of the image are processed, reducing noise and improving focus on facial features critical for paralysis detection.

To further prepare the data, each detected facial region is resized to a fixed resolution, typically 128x128 pixels. This step ensures that all images have consistent dimensions, Grayscaling is then applied to the resized images to convert them into single-channel data, reducing computational complexity while retaining essential texture and structural information.. These preprocessing steps collectively ensure that the dataset is clean, consistent, and optimized for feature extraction and classification.

C. FEATURE EXTRACTION

1. LOCAL BINARY PATTERN

Local Binary Pattern (LBP) captures local texture features by encoding binary patterns around each pixel based on intensity comparisons. This method involves assigning a binary value (0 or 1) based on whether the neighboring pixel's intensity is less than or greater than the center pixel's intensity. The resulting binary pattern is then converted into a decimal number, which serves as the LBP code for that pixel. The LBP at a given pixel (x, y) is defined as Eq. (1) [30]. I_p is the intensity of the center pixel, I_q is the intensity of the neighboring pixel and P is the number of neighbors. ALGORITHM 1 presents the basic LBP algorithm.

$$LBP(x,y) = \sum_{p=0}^{P-1} g\left(I_q - I_p\right) 2^p \quad (1)$$

ALGORITHM 1. Local Binary Patterns (LBP)

Require: Grayscale image I
Ensure: LBP matrix *LBP(I)*
Initialize an empty matrix *LBP(I)* with same dimension as I
for each pixel p in I **do**
  Define a 3×3 neighborhood around pixel p with p as the center
  Initialize $binaryPattern \leftarrow []$
  for each neighboring pixel q in the 3×3 neighborhood **do**
    if intensity of q ≥intensity of p **then**
      Append 1 to $binaryPattern$
    else
      Append 0 to $binaryPattern$

end if
end for
Concatenate $binaryPattern$ to form a binary sequence
Concatenate $binaryPattern$ to a decimal value, $LBPCode$
Assign $LBPCode$ to $LBP(I)[p]$
end for
return $LBP(I)$

2. HISTOGRAM OF ORIENTED GRADIENTS

HOG extracts gradient information by binning orientations in localized regions, capturing detailed edge and shape features. HOG divides the image into small cells and computes the gradient magnitude and orientation for each pixel. These gradient orientations are then quantized into bins to create a histogram for each cell. By normalizing the histograms over larger spatial blocks, HOG improves robustness to changes in lighting and contrast, ensuring consistent performance across varied conditions. The gradient at a pixel is calculated as Eq. (2) and Eq. (3) with Eq. (4) and Eq. (5) [31]. G_x is the gradien in x direction, G_y is the gradien in y direction, $M$is the magnitude and θ is the orientation. ALGORITHM 2 presents the basic HOG algorithm.

$$M = \sqrt{G_x^2 + G_y^2} \tag{2}$$

$$\theta = \tan^{-1}\frac{G_y}{G_x} \tag{3}$$

with,

$$G_x = I(x+1, y) - I(x-1, y) \tag{4}$$

$$G_y = I(x, y+1) - I(x, y-1) \tag{5}$$

ALGORITHM 2. Histogram of Oriented Gradients (HOG)

Require: Grayscale image *I*
Ensure: HOG feature vector *HOG(I)*
Define $cellSize$ and $blockSize$
Define number of bins, $numBins$
Initialize an empty list *HOG(I)* to store feature vectors
for each cell in *I* **do**
  Compute the gradient magnitude and orientation for each pixel
  Quantize orientations into $numBins$
  Accumulate magnitudes into bins to form the cell histogram
  Append cell histogram to *HOG(I)*
end for
for each block in *I* **do**
  Concatenate histograms of all cells in the block
  Normalize the concatenated histogram
  Append the normalized block histogram to $HOG(I)$
end for
return $HOG(I)$

3. GABORS FILTER

Gabor filters analyze images across multiple scales and orientations, capturing detailed texture and frequency information. A Gabor filter is defined Eq. (6) [32]. x' and y' is rotated coordinates, λ is the wavelength of sinusoidal component, θ is the orientations of Gabor filter, ψ is the phase offset, γ is the aspect ratio, and σ is the standard deviation of Gaussian envelope. ALGORITHM 3 presents the basic Gabor filter algorithm.

$$G(x, y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda} + \psi\right) \tag{6}$$

ALGORITHM 3. Gabor Filter

Require: Grayscale image *I*
Ensure: Gabor feature matrix *Gabor(I)*
Define $numScale$ and $num\ Orientations$ for the Gabor filters
Initialize an empty list *Gabor(I)* to store feature responses
for each scale *s* in $numScale$ **do**
  for each block in *I* **do**
    Create Gabor filter $G(s, \theta)$ with scale s and orientation θ
    Apply $G(s, \theta)$ to *I* to obtain response $R(s, \theta)$
    Append $R(s, \theta)$ to $Gabor(I)$
  end for
end for
return $Gabor(I)$

4. GEOMETRIC FEATURES

Using facial landmarks, geometric features quantify distances and angles between key points to detect structural asymmetries. ALGORITHM 4 presents the basic HOG algorithm. Distance (d) and angle (θ) can be calculated using equation Eq. (7) and Eq. (8).

$$d(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{7}$$

$$\theta = \tan^{-1}\left(\frac{y_i - y_j}{x_i - x_j}\right) \tag{8}$$

ALGORITHM 4. Geometric Feature

Require: Image *I* with detected facial landmarks
Ensure: Geometric feature *Geo(I)*
Identify key facial landmarks
Initialize an empty list *Geo(I)* to store geometric feature
Calculate distance between landmarks:
  Eye Distance ← Distance between left and right eye centers
  Mouth Width ← Distance between left and right mouth corners
  Nose to Mouth ← Distance between nose tip to mouth center
  Chin to mouth ← Distance between chin to mouth center
Calculate geometric ratios and angles:
  Mouth to Eye Ratio ← Mouth Width / Eye Distance
  Nose to Mouth / Chin to Mouth Ratio ← Nose to Mouth / Chin to Mouth
  Eye Mouth Angle ← Angle between each eye and mouth corners
Concatenate all calculated distances, ratios and angles into $Geo(I)$
return $Geo(I)$

D. CLASSIFICATION

This study implements four algorithms that are frequently implemented in the field of machine learning. Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN) were used to classify facial paralysis based on the extracted features.

1. LOGISTIC REGRESSION (LR)

Logistic Regression is supervised learning algorithm for binary classification tasks. It models the probability of a data point belonging to a particular class using a logistic function. Logistic Regression assumes a linear relationship between the input features and the log-odds of the target class. The probability of a sample belonging to the positive class ($P(y = 1|x)$) is given by Eq. (9) and Eq. (10) [33]. z is the liniear combination of input features, X is the value of the feature, θ is the bias term, and b is the coefficient of the feature

$$P(y = 1|x) = \sigma(z) = \frac{1}{1+e^{-z}} \tag{9}$$

with,

$$z = X \cdot \theta + b \tag{10}$$

The model is trained by minimizing the binary cross-entropy loss Eq. (11) [34]. L is the binary cross-entropy loss, n is the total number of samples, y_i is the actual class label, σ is the predicted of probability of the positive class

$$L = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log\left(\sigma(z_i)\right) + (1 - y_i)\log\left(1 - \sigma(z_i)\right)\right] \tag{11}$$

2. SUPPORT VECTOR MACHINE (SVM)

Support Vector Machine (SVM) is a powerful supervised learning algorithm that aims to find the optimal hyperplane that maximizes the margin between data points of different classes. The margin is defined as the distance between the hyperplane and the nearest data points from each class, known as support vectors. SVM solves the following optimization problem Eq. (12) [35]. w is the wight vector, C is the regularization parameter controlling the trade-off between maximizing the margin and minimizing classification errors, x_i is the feature vector for the *i-th* sample, Φ is the mapping function to transform data into a higher-dimensional feature space, b is bias term, ξ_i is slack variable allowing misclassification, and y is the class label.

$$\min_{w,b,\xi} \frac{1}{2}|w|^2 + C\sum_{i=1}^{n}\xi_i$$
$$\text{subject to: } y_i(w \cdot \phi(x_i) + b) \geq 1 - \xi_i,\ \xi_i \geq 0 \tag{12}$$

When the data is not linearly separable in the input space, SVM uses kernel functions to map the data into a higher-dimensional feature space, where a linear hyperplane can be constructed. The following are four basic kernels, liear kernel Eq. (13), Polynomial Kernel Eq. (14), Sigmoid Kernel Eq. (15) and Radial Basis Function (RBF) Eq. (16) [36]. x is the input feature, K is the kernels function, r is the coefficient or bias term added to the kernel function, d is the degree of the polynomial in the Polynomial kernel, and γ is the scaling parameter that controls the influence of the input data.

Linear

$$K(x_i, x_j) = x_i \cdot x_j \tag{13}$$

Polynomial

$$K(x_i, x_j) = (\gamma \cdot x_i \cdot x_j + r)^d \tag{14}$$

Sigmoid

$$K(x_i, x_j) = \tanh(\gamma \cdot x_i \cdot x_j + r) \tag{15}$$

Radial Basis Function (RBF)

$$K(x_i, x_j) = \exp(-\gamma|x_i - x_j|^2) \tag{16}$$

3. RANDOM FOREST (RF)

Random Forest is an ensemble learning method that constructs multiple decision trees and aggregates their outputs to improve classification accuracy and robustness. It operates by combining predictions from individual trees, each trained on a different bootstrap sample of the training data, reducing overfitting and enhancing generalization. During the construction of each tree, nodes are split based on the Gini Impurity criterion, which measures the homogeneity of the labels in a dataset. The Gini Impurity at a node t ($G(t)$) is calculated as Eq. (17) [37]:

$$[G(t) = 1 - \textstyle\sum_{i=1}^{c} p_i^2] \tag{17}$$

where p_i is the proportion of samples belonging to class i at node t. This ensures that splits maximize the purity of child nodes, improving the discriminative power of the individual trees. For classification, Random Forest uses ensemble prediction, where each tree in the forest votes for a class label, and the final prediction $\hat{y}$ is determined by majority voting Eq. (18):

$$[\hat{y} = \text{mode}(y_1, y_2, \dots, y_{n_{trees}})] \tag{18}$$

4. K-NEAREST NEIGHBORS (KNN)

K-Nearest Neighbors (KNN) is a non-parametric, instance-based learning algorithm used for classification tasks. It identifies the k-nearest data points to a given test sample in the feature space and predicts its class based on the majority class of its neighbors. The choice of distance metric plays a crucial role in determining the performance of KNN, as it defines how "closeness" is measured between data points. Commonly used metrics include Euclidean Distance, which calculates the straight-line distance between two points. The following are four distance metrics Euclidian Distance Eq. (19), Manhattan Distance Eq. (20), Minkowski Distance Eq. (21), and Hamming Distance Eq. (22) [36]. $d(x_i, x_j)$ is the distance between two point, x_i is the i-th data point, x_j is the j-th data point, x_{ik} is the value of the k-th feature for the i-th data point, x_{jk} is the value of the k-th feature for the j-th data point, n is the number of features and p is the parameter controlling the metric.

Euclidian Distance

$$d(x_i, x_j) = \sqrt{\textstyle\sum_{k=1}^{n}(x_{ik} - x_{jk})^2} \tag{19}$$

Manhattan Distance

$$d(x_i, x_j) = \textstyle\sum_{k=1}^{n}|x_{ik} - x_{jk}| \tag{20}$$

Minkowski Distance

$$d(x_i, x_j) = \left(\sum_{k=1}^{n} |x_{ik} - x_{jk}|^p\right)^{\frac{1}{p}} \quad (21)$$

Hamming Distance

$$d(x_i, x_j) = \sum_{k=1}^{n} 1(x_{ik} \neq x_{jk}) \quad (22)$$

E. EVALUATION

In this study, the classification model was evaluated using accuracy, recall, F1-score, and precision. It can be calculated using True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN). The following is the calculation formula.

1. ACCURACY
   Accuracy measures the overall correctness of the model by calculating the percentage of both true positive and true negative predictions out of all predictions. Accuracy can be calculated in Eq. (23) as follows.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (23)$$

2. PRECISION
   Precision shows how reliable the model's positive predictions are by indicating the percentage of correctly identified paralysis cases out of all cases predicted as positive. Precision can be calculated in Eq. (24) as follows.

$$Preccision = \frac{TP}{TP + FP} \quad (24)$$

3. RECALL
   Recall focuses on sensitivity, measuring the percentage of actual paralysis cases that were correctly detected. Recall can be calculated in Eq. (25) as follows.

$$Recall = \frac{TP}{TP + FN} \quad (25)$$

4. F1-SCORE
   F1-score combines precision and recall into a single balanced metric. F1-Score can be calculated in Eq. (26) as follows.

$$F1 - Score = \frac{Precission * Recall}{Precission + Recall} \quad (26)$$

III. RESULT

A. FEATURE EXTRACTION

In this study, various feature extraction techniques were employed to enhance the analysis and representation of input data. These techniques include LBP, HOG, Gabor filter, and geometric features. LBP is employed for capturing local texture patterns. HOG is used to extract shape and gradient-based features. The Gabor filter, known for its spatial and frequency-selective capabilities, captures texture and orientation details. Geometric features, derived from spatial and structural attributes, emphasize shape characteristics. The specific parameters used in each technique, which determine their configuration and effectiveness, are detailed in TABLE 1. These parameters were selected to optimize the performance of the feature extraction process for the given dataset.

TABLE I
Parameters of each technique

Feature Extraction	Key Parameters	Values
LBP	Radius	1
	Neighbors	8
	Grid Size	8×8
HOG	Cell Size	8×8
	Block Size	2×2
	Orientations	9
Gabor	Kernel Size	31×31
	Sigma	4
	Frequency	8
Geometric	Landmark Model	dlib facial landmark [38]
	Distance	Eye, Mouth, Nose and Chin
	Angle	Eye, Mouth and Nose

Feature extraction is performed on images with a size of 128x128 using LBP, HOG, Gabor filter and geometric features according to the parameters provided in TABLE 1. The results of feature extraction for each technique are shown in TABLE 2. The LBP technique extracts the fewest features, as it only extracts local textures in an image. Conversely, the Gabor filter extracts the most features, as it extracts texture and frequency-based features. HOG extracts gradient and edge orientation features, while geometric extracts structural features by calculating the distances and angles between facial landmarks.

TABLE 2
Number of features

Feature Extraction	Number of Features
LBP	10
HOG	810
Gabor	1684
Geometric	13

B. Classification

In this study, classification was performed to determine facial paralysis from the images. After feature extraction, the extracted features were used as inputs for the classification process. The images were labelled, with label 1 representing paralyzed faces and label 0 representing non-paralyzed faces. Classification was conducted using LR, SVM, RF, and KNN methods. In this study, hyperparameter tuning was conducted using the grid space search approach to optimize the classification performance of LR, SVM, RF, and KNN classifiers. The search space every method are presented in TABLE 3.

TABLE 3
Hyperparameter tuning

Clasifier	Hyperparameter	Grid Search Space	Description
LR	*C*	[0.01, 0.1, 1, 10]	Regularization parameter
SVM	Kernel	Linear and Radial Basis Function	The function used to compute the kernel matrix for classification
	C	[0.1, 1, 10]	Regularization parameter
RF	N_estimators	[50, 100, 200]	Number of trees in the forest

	Max_depth	[5, 10, 20]	Maximum depth of the tree
	Min_samples_split	[2, 5, 10]	Minimum number of samples required to split a node
KNN	K_neighbors	[3, 5, 7]	Number of nearest neighbors
	Weights	Uniform and Distance	Weight function used in prediction

Each method was evaluated using 5-fold cross-validation with accuracy as the evaluation metric. The process involved splitting the dataset into five subsets, where the model was trained on four folds and validated on the remaining fold iteratively. The metric results presented in the FIGURE 2 to FIGURE 5 highlight the comparative performance of feature extraction methods LBP, HOG, Gabor, and Geometric features when combined with LR, KNN, RF and SVM.

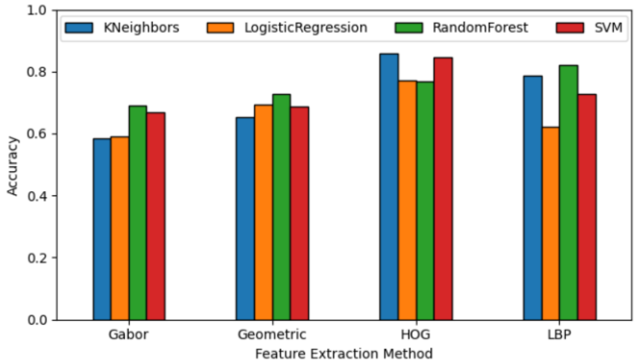

FIGURE 2. Model Accuracy

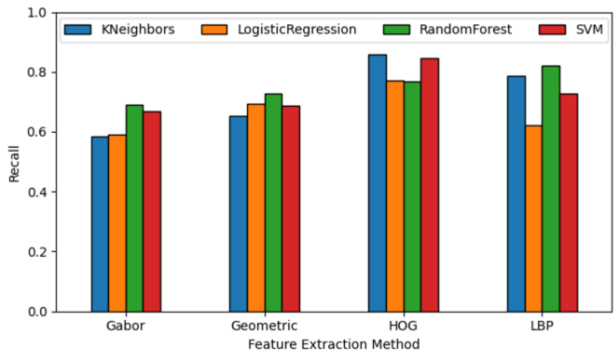

FIGURE 3. Model Recall

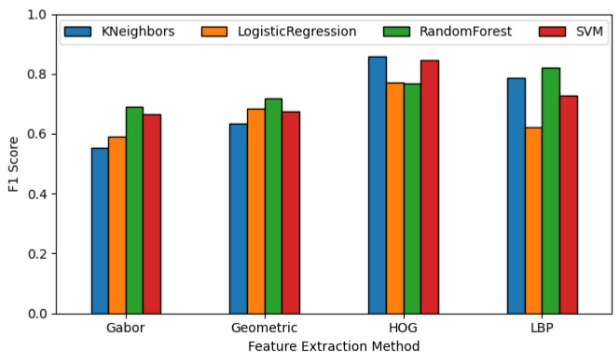

FIGURE 4. Model F1 Score

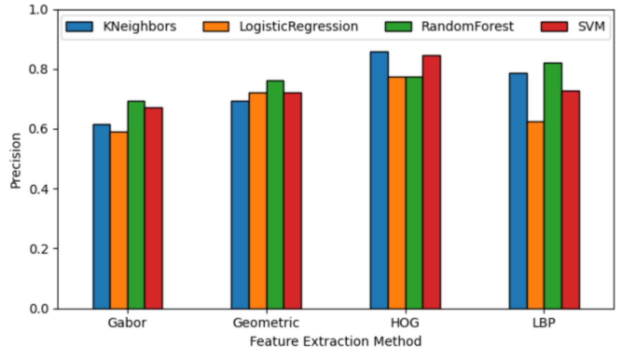

FIGURE 5. Model Precision

TABLE 4 shows the top 10 accuracy provides evaluation of the classification performance for each feature extraction method LBP, HOG, Gabor, and Geometric features combined with LR, KNN, RF and SVM.

TABLE 4
The top 10 Accuracy

No	Feature Extraction	Classifier	Accuracy
1	HOG	KNN	85.96%
2	HOG	SVM	84.52%
3	LBP	RF	82.07%
4	LBP	KNN	78.69%
5	HOG	LR	77.15%
6	HOG	RF	76.84%
7	Geometric	RF	72.85%
8	LBP	SVM	72.64%
9	Geometric	LR	69.36%
10	Gabor	RF	69.05%

IV. DISCUSSION

The number of features extracted by each method varies significantly. LBP extracted 10 features and provides a representation of local texture patterns. HOG extracted 810 features and provides representation of structural and edge details. Gabor filters extracted the largest set of 1,684 features and captured multi-scale and multi-orientation texture information. Geometric features, with only 13 features, focus on structural relationships of distances and angles between landmarks.

The extracted features used as input for machine learning models, including LR, SVM, RF, and KNN. Each classifier was evaluated based on key performance metrics: accuracy, precision, recall, and F1-score. The result reveals that all methods achieved similar metrics, including accuracy, precision, recall, and F1-score, despite significant differences in the number of extracted features. LBP, with only 10 features, and Geometric features, with 13, provided compact representations that performed comparably to HOG and Gabor, which extracted 810 and 1,684 features, respectively. This indicates that feature relevance, rather than sheer quantity, is key to achieving high classification performance.

The results highlight that methods with fewer features can still achieve competitive results. Using fewer features in a machine learning model can significantly enhance computational efficiency. Each feature represents a dimension in the data, and reducing the number of dimensions directly impacts the computational resources required for data processing, training, and inference.

According to the result, HOG combined with KNN achieved the highest scores across all metrics. This indicates that HOG effectively captures the most relevant features for facial paralysis classification. By focusing on gradient orientations and edge information. The high scores across metrics indicate that the features extracted by HOG are not only detailed but also highly discriminative, thereby allowing the classifiers to make precise predictions.

Geometric features show good potential for facial paralysis classification due to their simplicity and efficiency. With only 13 features extracted, they achieved competitive results compared to more complex methods like HOG and Gabor. Geometric features extracts structural features by calculating the distances and angles. These features effectively capture structural asymmetries, making them highly relevant for this task. Additionally, there are many ways to expand geometric features, such as using ratios, higher-order angles, or spatial configurations, which could further improve performance.

In facial paralysis detection, it is generally preferable to have false positives over false negatives. A false positive, where the system mistakenly identifies paralysis in a healthy individual, may cause some unnecessary follow-up evaluations, but it does not carry immediate health risks. Conversely, a false negative, where actual facial paralysis goes undetected, could lead to serious consequences. Facial paralysis can be an early indicator of underlying conditions such as stroke or Bell's palsy, which require prompt medical attention. Missing these cases may delay crucial interventions, potentially worsening patient outcomes.

TABLE 5
The top 10 recall

No	Feature Extraction	Classifier	Recall
1	HOG	KNN	85.96%
2	HOG	SVM	84.53%
3	LBP	RF	82.07%
4	LBP	KNN	78.69%
5	HOG	LR	77.15%
6	HOG	RF	76.84%
7	Geometric	RF	72.85%
8	LBP	SVM	72.64%
9	Geometric	LR	69.36%
10	Gabor	RF	69.06%

TABLE 5 displaying the top 10 recall values highlights the effectiveness of various feature extraction methods and classifiers in correctly identifying true positive cases of facial paralysis. Among the results, HOG combined with KNN achieves the highest recall values, indicating its strength in capturing detailed structural patterns that enhance the detection of facial asymmetries. LBP, despite its smaller feature set, also demonstrates competitive recall values across multiple classifiers, showcasing its efficiency in identifying relevant patterns in local textures. Geometric features, while computationally lightweight, rank among the top methods due to their focus on facial asymmetries, which are key indicators of paralysis. The table underscores the importance of feature selection and classifier pairing, as methods like Gabor, though rich in detail, occasionally rank lower in recall due to feature redundancy or noise. These results emphasize the need to balance feature complexity and classifier sensitivity to achieve optimal recall, ensuring reliable detection of facial paralysis in diverse scenarios.

TABLE 6 presents a comparative of various methodologies for similar studies, focusing on different feature extraction techniques, classifiers, datasets, and performance metrics. Several prior studies have explored different approaches, such as Kim [39], which used a linear regression model for facial landmark detection and an SVM with a linear kernel, achieving 88.9% accuracy on a private dataset of 36 subjects. Arora [40] employed facial landmark features with cascade regression and an SVM, obtaining 76.87% accuracy on the Stroke Faces dataset. Tan [41] introduced the FNPARCELMCCNN method, which achieved 85.5% accuracy on the YouTube Facial Palsy (YFP) database. Meanwhile, Liu [42] applied a PHCNN-LSTM model, reaching 94.81% accuracy using the YouTube Facial Palsy Database and the Extended Cohn-Kanade Database. Lastly, Sajid [43] implemented a CNN-based approach (VGG-16) trained on an augmented dataset of 2000 images, achieving 92.6% accuracy, 92.91% precision, 93.14% sensitivity, and an F1 score of 93%.

In comparison, this study experimented with multiple feature extraction techniques on the YouTube Facial Palsy (YFP) database, with HOG-KNN and LBP-RF both achieving 85.96% accuracy and recall, slightly outperforming Tan [41]. The LBP method alone resulted in 82.07% accuracy and recall, while Geometric-RF and Gabor-RF performed lower, with 72.85% and 69.05% accuracy, respectively. These results indicate that traditional feature extraction techniques like HOG and LBP can still achieve competitive performance, though deep learning-based methods such as CNN (VGG-16) and PHCNN-LSTM consistently outperform classical approaches by exceeding 92% accuracy. This highlights the potential of deep learning for more robust facial paralysis classification.

Geometric-RF showed lower performance, achieving 72.85% accuracy, likely due to its limited feature representation compared to other methods. In this study, the geometric approach used only 13 features, which may not have been sufficient to capture the full complexity of facial paralysis variations. One potential improvement is expanding the geometric feature set by incorporating higher-order statistical features, such as curvature analysis, displacement vectors over time, or dynamic movement trajectories of facial landmarks. Another approach is to combine geometric features with texture-based methods or deep learning embeddings to enhance feature richness. Feature selection techniques, such as Principal Component Analysis (PCA) or Feature Fusion Networks, could also optimize the most discriminative geometric attributes for classification.

TABLE 6
Comparison of similar studies

Author	Methodology	Dataset	Performance
Kim [39]	Linear regression model for facial landmark detection and SVM with linear kernel for classification	Private dataset of 36 subjects (23 noral−13 palsy patients) performing 3 motions	88.9% classification accuracy
Arora [40]	Facial landmark features with cascade regression and SVM	Stroke faces dataset of 1024 images and 1081 images of healthy faces	76.87% accuracy
Tan [41]	FNPARCELMCCNN method	YouTube Facial Palsy (YFP) database	85.5% accuracy
Liu [42]	PHCNN-LSTM	YouTube Facial Palsy Database Extended CohnKanade Database	Accuracy PHCNN-LSTM 0.9481%
Sajid [43]	CNN (VGG-16)	92.6% accuracy 92.91% precision 93.14% sensitivity 93% F1 Score	Dataset from online sources augmented to 2000 images
This Study	HOG-KNN	YouTube Facial Palsy (YFP) database	85.96% Accuracy 85.96% Recall
This Study	LBP-RF	YouTube Facial Palsy (YFP) database	85.96% Accuracy 85.96% Recall
This Study	LBP	YouTube Facial Palsy (YFP) database	82.07% Accuracy 82.07% Recall
This Study	Geometric-RF	YouTube Facial Palsy (YFP) database	72.85% Accuracy 72.85% Recall
This Study	Gabor-RF	YouTube Facial Palsy (YFP) database	69.05% Accuracy 69.06% Recall

The primary limitation of the current study is its reliance on handcrafted feature extraction methods, which require extensive tuning and may not generalize well across diverse datasets. While techniques like HOG and LBP are computationally efficient, they are limited in capturing complex, high-level features present in more intricate datasets. Another weakness is the lack of testing on larger, more diverse datasets, which could better validate the model's generalizability.

This study has implications for the field of facial paralysis classification. This comparative analysis reveal distinct strengths and trade-offs among the feature extraction techniques LBP, HOG, Gabor filters, and Geometric features for facial paralysis classification. HOG demonstrated the highest overall performance, excelling in capturing detailed gradient and structural information, which translated to good accuracy, precision, recall, and F1-score across classifiers. The analysis underscores that while HOG provides more detailed feature representation, lightweight methods like LBP and Geometric features offer efficient alternatives, especially when computational resources are limited. Geometric features stand out as an area with significant potential for expansion. Future work could explore advanced geometric descriptors, such as curvature, symmetry ratios, or dynamic geometric features captured from videos. These findings highlight the importance of aligning feature extraction techniques with application requirements, and the potential for hybrid approaches to combine their strengths for improved classification performance.

V. CONCLUSION

This study provides a comprehensive comparative analysis of feature extraction techniques for facial paralysis classification, focusing on LBP, HOG, Gabor filters, and Geometric features. The evaluation highlights that HOG delivers best performance with an accuracy 85% due to its ability to capture detailed texture and structural information, LBP and Geometric features stand out for their computational efficiency, achieving competitive metrics with significantly fewer features. LBP, employing 10 features, achieves an accuracy of 82%. In comparison, the geometric features, which utilize only 13 features, attain an accuracy of 72%. Future research should expand geometric features and explore hybrid approaches that integrate the strengths of different methods, potentially enhancing both performance and scalability for practical use in facial paralysis detection and related medical imaging applications.

ACKNOWLEDGMENT

This research is funded by Center of Research and Community Service (P3M), Politeknik Negeri Indramayu under Kegiatan PUKTI 2024.

REFERENCES

[1] P. R. Sharma, R. M. Zuker, and G. H. Borschel, "Perspectives in the reconstruction of paediatric facial paralysis," *Curr Opin Otolaryngol Head Neck Surg*, vol. 23, no. 6, pp. 470–479, Dec. 2015, doi: 10.1097/MOO.0000000000000211.

[2] S. E. Coulson, G. R. Croxson, R. D. Adams, and N. J. O'dwyer, "Reliability of the 'Sydney,' 'Sunnybrook,' and 'House Brackmann' Facial Grading Systems to Assess Voluntary Movement and Synkinesis after Facial Nerve Paralysis," *Otolaryngology–Head and Neck Surgery*, vol. 132, no. 4, pp. 543–549, Apr. 2005, doi: 10.1016/j.otohns.2005.01.027.

[3] T. Wang, S. Zhang, J. Dong, L. Liu, and H. Yu, "Automatic evaluation of the degree of facial nerve paralysis," *Multimed Tools Appl*, vol. 75, no. 19, pp. 11893–11908, Oct. 2016, doi: 10.1007/s11042-015-2696-0.

[4] J. W. House and D. E. Brackmann, "Facial Nerve Grading System," *Otolaryngology–Head and Neck Surgery*, vol. 93, no. 2, pp. 146–147, Apr. 1985, doi: 10.1177/019459988509300202.

[5] B. G. Ross, G. Fradet, and J. M. Nedzelski, "Development of a Sensitive Clinical Facial Grading System," *Otolaryngology–Head and Neck Surgery*, vol. 114, no. 3, pp. 380–386, Mar. 1996, doi: 10.1016/S0194-59989670206-1.

[6] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Trans Pattern Anal Mach Intell*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006, doi: 10.1109/TPAMI.2006.244.

[7] A. Günay Yılmaz, U. Turhal, and V. Nabiyev, "Face presentation attack detection performances of facial regions with multi-block LBP features," *Multimed Tools Appl*, vol. 82, no. 26, pp. 40039–40063, Nov. 2023, doi: 10.1007/s11042-023-14453-7.

[8] M. M. Bouchene, "Bayesian optimization of histogram of oriented gradients (HOG) parameters for facial recognition," *J Supercomput*, vol. 80, no. 14, pp. 20118–20149, Sep. 2024, doi: 10.1007/s11227-024-06259-7.

[9] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, IEEE, pp. 886–893. doi: 10.1109/CVPR.2005.177.

[10] K. Verma and A. Khunteta, "Facial expression recognition using Gabor filter and multi-layer artificial neural network," in *2017 International Conference on Information, Communication, Instrumentation and Control (ICICIC)*, IEEE, Aug. 2017, pp. 1–5. doi: 10.1109/ICOMICON.2017.8279123.

[11] J. Ou, X.-B. Bai, Y. Pei, L. Ma, and W. Liu, "Automatic Facial Expression Recognition Using Gabor Filter and Expression Analysis," in *2010 Second International Conference on Computer Modeling and Simulation*, IEEE, Jan. 2010, pp. 215–218. doi: 10.1109/ICCMS.2010.45.

[12] G. S. Parra-Dominguez, C. H. Garcia-Capulin, and R. E. Sanchez-Yanez, "Automatic Facial Palsy Diagnosis as a Classification Problem Using Regional Information Extracted from a Photograph," *Diagnostics*, vol. 12, no. 7, p. 1528, Jun. 2022, doi: 10.3390/diagnostics12071528.

[13] G. S. Parra-Dominguez, R. E. Sanchez-Yanez, and C. H. Garcia-Capulin, "Facial Paralysis Detection on Images Using Key Point Analysis," *Applied Sciences*, vol. 11, no. 5, p. 2435, Mar. 2021, doi: 10.3390/app11052435.

[14] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*. Wiley, 2013. doi: 10.1002/9781118548387.

[15] Nazish, S. I. Ullah, A. Salam, W. Ullah, and M. Imad, "COVID-19 Lung Image Classification Based on Logistic Regression and Support Vector Machine," 2021, pp. 13–23. doi: 10.1007/978-3-030-77246-8_2.

[16] C. Cortes and V. Vapnik, "Support-vector networks," *Mach Learn*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.

[17] R. Ankita, Ch. U. Kumari, Mr. Mohd. J. Mehdi, N. Tejashwini, and D. T. Pavani, "Lung Cancer Image- Feature Extraction and Classification using GLCM and SVM Classifier," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 11, pp. 2211–2215, Sep. 2019, doi: 10.35940/ijitee.K2044.0981119.

[18] L. Breiman, "Random forests," *Mach Learn*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[19] P. Mekha and N. Teeyasuksaet, "Image Classification of Rice Leaf Diseases Using Random Forest Algorithm," in *2021 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering*, IEEE, Mar. 2021, pp. 165–169. doi: 10.1109/ECTIDAMTNCON51128.2021.9425696.

[20] B. Xu, Y. Ye, and L. Nie, "An improved random forest classifier for image classification," in *2012 IEEE International Conference on Information and Automation*, IEEE, Jun. 2012, pp. 795–800. doi: 10.1109/ICInfA.2012.6246927.

[21] S. Sharma and H. Amhia, "Human facial expressions recognition using artificial intelligence K-Nearest Neighbor (KNN) Algorithm," in *2022 6th International Conference on Devices, Circuits and Systems (ICDCS)*, IEEE, Apr. 2022, pp. 107–111. doi: 10.1109/ICDCS54290.2022.9780832.

[22] N. S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression," *Am Stat*, vol. 46, no. 3, pp. 175–185, Aug. 1992, doi: 10.1080/00031305.1992.10475879.

[23] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans Inf Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967, doi: 10.1109/TIT.1967.1053964.

[24] T. C. ten Harkel, G. de Jong, H. A. M. Marres, K. J. A. O. Ingels, C. M. Speksnijder, and T. J. J. Maal, "Automatic grading of patients with a unilateral facial paralysis based on the Sunnybrook Facial Grading System - A deep learning study based on a convolutional neural network," *Am J Otolaryngol*, vol. 44, no. 3, p. 103810, May 2023, doi: 10.1016/j.amjoto.2023.103810.

[25] D. G. Kim, S. K. Seo, H. J. Lee, and K. S. Tae, "Deep Learning-Based Evaluation of Facial Paralysis Recognition Accuracy Using East Asian and Western Facial Datasets," *Journal of Rehabilitation Welfare Engineering & Assistive Technology*, vol. 17, no. 1, pp. 48–54, Feb. 2023, doi: 10.21288/resko.2021.17.1.48.

[26] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja, "Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues," *Array*, vol. 10, p. 100057, Jul. 2021, doi: 10.1016/j.array.2021.100057.

[27] S. F. Ahmed *et al.*, "Deep learning modelling techniques: current progress, applications, advantages, and challenges," *Artif Intell Rev*, vol. 56, no. 11, pp. 13521–13617, Nov. 2023, doi: 10.1007/s10462-023-10466-8.

[28] N. Thompson, K. Greenewald, K. Lee, and G. F. Manso, "The Computational Limits of Deep Learning," in *Ninth Computing within Limits 2023*, LIMITS. doi: 10.21428/bf6fb269.1f033948.

[29] G.-S. J. Hsu, J.-H. Kang, and W.-F. Huang, "Deep Hierarchical Network With Line Segment Learning for Quantitative Analysis of Facial Palsy," *IEEE Access*, vol. 7, pp. 4833–4842, 2019, doi: 10.1109/ACCESS.2018.2884969.

[30] A. Hadid, "The Local Binary Pattern Approach and its Applications to Face Analysis," in *2008 First Workshops on Image Processing Theory, Tools and Applications*, IEEE, Nov. 2008, pp. 1–9. doi: 10.1109/IPTA.2008.4743795.

[31] L. Zhang, W. Zhou, J. Li, J. Li, and X. Lou, "Histogram of Oriented Gradients Feature Extraction Without Normalization," in *2020 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, IEEE, Dec. 2020, pp. 252–255. doi: 10.1109/APCCAS50809.2020.9301715.

[32] R. Mehrotra, K. R. Namuduri, and N. Ranganathan, "Gabor filter-based edge detection," *Pattern Recognit*, vol. 25, no. 12, pp. 1479–1494, Dec. 1992, doi: 10.1016/0031-3203(92)90121-X.
[33] S. Le Cessie and J. C. Van Houwelingen, "Ridge Estimators in Logistic Regression," *Appl Stat*, vol. 41, no. 1, p. 191, 1992, doi: 10.2307/2347628.
[34] J. Terven, D. M. Cordova-Esparza, A. Ramirez-Pedraza, E. A. Chavez-Urbiola, and J. A. Romero-Gonzalez, "Loss Functions and Metrics in Deep Learning," Jul. 2023.
[35] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO Algorithm for SVM Classifier Design," *Neural Comput*, vol. 13, no. 3, pp. 637–649, Mar. 2001, doi: 10.1162/089976601300014493.
[36] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," Jan. 2012.
[37] L. Breiman, "Random Forests," *Mach Learn*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
[38] D. King, "Dlib-ml: A Machine Learning Toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, Jul. 2009, doi: 10.1145/1577069.1755843.
[39] H. Kim, S. Kim, Y. Kim, and K. Park, "A Smartphone-Based Automatic Diagnosis System for Facial Nerve Palsy," *Sensors*, vol. 15, no. 10, pp. 26756–26768, Oct. 2015, doi: 10.3390/s151026756.
[40] A. Arora, A. Sinha, K. Bhansali, R. Goel, I. Sharma, and A. Jayal, "SVM and Logistic Regression for Facial Palsy Detection Utilizing Facial Landmark Features," in *Proceedings of the 2022 Fourteenth International Conference on Contemporary Computing*, New York, NY, USA: ACM, Aug. 2022, pp. 43–48. doi: 10.1145/3549206.3549216.
[41] X. Tan, J. Yang, and J. Cao, "Facial Nerve Paralysis Assessment based on Regularized Correntropy Criterion SSELM ^{vc} and Cascade CNN," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, IEEE, Oct. 2021, pp. 1043–1047. doi: 10.1109/IEEECONF53345.2021.9723091.
[42] X. Liu, Y. Xia, H. Yu, J. Dong, M. Jian, and T. D. Pham, "Region Based Parallel Hierarchy Convolutional Neural Network for Automatic Facial Nerve Paralysis Evaluation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 10, pp. 2325–2332, Oct. 2020, doi: 10.1109/TNSRE.2020.3021410.
[43] M. Sajid, T. Shafique, M. J. A. Baig, I. Riaz, S. Amin, and S. Manzoor, "Automatic Grading of Palsy Using Asymmetrical Facial Features: A Study Complemented by New Solutions," *Symmetry (Basel)*, vol. 10, no. 7, p. 242, Jun. 2018, doi: 10.3390/sym10070242.

AUTHORS BIOGRAPHY

SALAMET NUR HIMAWAN is a lecturer in the informatics engineering department at Politeknik Negeri Indramayu, Indonesia. He received a bachelor of science degree in Physics from Bogor Agricultural University and He received a Master of Science degree in Computational Science from Bandung Institute of Technology. His research interests are mainly focused on data science and computer vision, with an emphasis on machine learning, deep learning and their applications in real-world problem. He actively participates as a member of the societies and journal editor.

ADI SUHERYADI is a lecturer in the informatics engineering department at Politeknik Negeri Indramayu. He received a bachelor's degree in electrical engineering from Institut Teknologi Bandung, Indonesia, and a master's degree in informatics from Universitas Telkom, Bandung, Indonesia. His research interests include image processing, computer vision, and pattern recognition. He actively participates as a member of the societies, a journal reviewer, and an editor in several journals.

KURNIA ADI CAHYANTO is a lecturer in in the informatics engineering department at Politeknik Negeri Indramayu. His research focused on image processing. He graduated from the department of electrical engineering, Muhamadiyah University of Surakarta in 2008. After that, he persued a master's degree in Informatic Engineering in Dian Nuswantoro University, Semarang. His research field involved is data mining and image processing.

FILEMON SITANGGANG is a assistant lecturer in the department of informatic engineering, Politeknik Negeri Indramayu. His work focused on software development on image processing, machine learning, and mobile android development. He graduated from department of informatic engineering, Politeknik Negeri Indramayu in 2024. After that, he pursued carrier as teacher at BPK PENABUR as information and communication technology teacher, and as part time software development to a private company.

KIKI ADI PAMUNGKAS is an assistant lecturer in the Department of Informatics Engineering at Politeknik Negeri Indramayu. His work focuses on machine learning, web development, and API development. Currently, he is a final-year student in the Department of Informatics Engineering at Politeknik Negeri Indramayu and is participating in an internship program at PT. Pratama Solusi Teknologi as a Full Stack Developer.