p-ISSN 2301–4156 | e-ISSN 2460–5719

Comparisons of Expression Phases Using Local Binary Pattern Histograms for Microexpression Recognition

Ulla Delfana Rosiani[1], Priska Choirina [2], Yessy Nindi Pratiwi [3], Septiar Enggar Sukmana[4]

[1,3,4] Informatics Engineering Study Program, Department of Information Technology, Malang State Polytechnic, Malang, 65141, INDONESIA, (tel.: 0341-404424; fax:0341-404420, email: [1]rosiani@polinema.ac.id, [3]yessynindi@gmail.com, [4]enggar@polinema.ac.id)
[2] Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Raden Rahmat, Malang 65163, INDONESIA (tel.: 0341- 399099; fax: -; email: [2]priska_choirina@uniramalang.ac.id)

[Accepted: 20 July 2023, Revised: 18 September 2023]
Corresponding Author: Ulla Delfana Rosiani

ABSTRACT — Microexpression is an emotional representation occurring spontaneously and cannot be controlled consciously. It is temporary (short duration) with subtle movements, making it difficult to detect with the naked eye. Microexpressions' muscle movements are generated in only a few small areas of the face, so observation of specific areas results in faster computation time and provides important information compared to observation of the entire face. This research proposes reducing the observation area and phase for microexpression recognition. The observed areas in the Chinese Academy of Science Micro-Expressions (CASME II) dataset are left and right eyebrows, right and left eyes, and mouth. The observation phase of microexpressions included analyzing the comparison in the onset to offset phase ("fullOAO") and in the onset, apex, and offset phase ("OAO"). Feature extraction was performed using a simple local binary patterns histogram (LBPH) method, which can represent local features in the facial area. The best result of the proposed method was the "fullOAO" phase with an accuracy of 96.8% (using support vector machine-radial basis function, SVM-RBF) and an average computation time of 0.192 ms per frame and 10.473 ms per video. In "OAO" phase type, an accuracy of 87.7% was achieved with a computation time of 0.159 ms per frame and 0.576 ms per video. The difference in accuracy and computation time between the two-phase types occurs because the number of frames in "fullOAO" type is greater than in "OAO", resulting in a different amount of processing time and feature extraction data. However, the 9% decrease in accuracy does not significantly affect the accuracy since the accuracy rate is still relatively good, above 80%. Furthermore, the correct measurement for computation time was the time taken to process each frame in the input video. Therefore, the proposed method can produce fast computation time and relatively accurate recognition.

KEYWORDS — Microexpression Recognition, Expression Phase, LBPH, Microexpression.

I. INTRODUCTION

Human facial expression is an important factor in social communication. Communication typically encompasses both verbal and nonverbal forms of expression [1]. Nonverbal communication is characterized by facial movements, commonly referred to as facial expressions. Facial expression is one of the techniques that is suitable to determine the mental state, attitude, and intention in humans [2]. In general, facial expressions encompass a variety of emotional states, such as sadness, disgust, fear, anger, surprise, and happiness [3]. Within the field of psychology, facial expressions are categorized into two distinct types: macroexpression and microexpression [4]. Facial expressions that can be easily observed using the naked eye can be referred to as macroexpressions. Macroexpressions are characterized by distinct movement in the facial muscles, so they can be easily analyzed [5].

In contrast to macroexpression, microexpression refers to facial muscle movements that exhibit subtle and spontaneous motion. Microexpression typically lasts less than 200 ms [6], making it difficult to hide even for highly skilled actors [7]. Therefore, microexpression is regarded as a concrete manifestation of human emotions [8]. Microexpression has three primary characteristics: short duration, smooth movement, and difficulty in hiding expression [9]. Microexpression analysis has proven to be highly beneficial in high-risk situations, such as assessments for depression recovery, negotiations with individuals involved in acts of terrorism, or criminal investigations [7]. Hence, microexpression analysis plays an important role in interpreting the genuine emotions of a person that can be implemented in everyday life [10].

A microexpression is a dynamic facial movement characterized by the subsequent progression of expression phases: neutral-onset-apex-offset-neutral. The initial condition is a neutral phase, and then when a stimulant condition is introduced, facial movements lead to the onset phase [11]. The onset phase shows the initial expression condition with the facial muscles starting to contract. Then, the apex phase refers to the state in which the facial muscles exhibit optimal movement, facilitating the execution of analysis during this phase [3]. Subsequently, the offset phase refers to the state where the facial muscles begin shifting to a neutral position.

Microexpression recognition is a relatively new area of study that has experienced significant growth in recent years. The existing body of microexpression study encompasses a diverse range of methodologies that seek to achieve automated recognition of microexpression. There are several methods commonly employed in facial expression recognition, such as principal component analysis (PCA) [12], local binary pattern (LBP) [13], and linear discriminant analysis (LDA) [14]. These methods have been successfully applied with relatively satisfactory accuracy results. The outcomes are dependent upon multiple factors. Apart from the selection of feature extraction methods, the selection of feature types and classification methods are crucial factors in achieving a high level of accuracy.

p-ISSN 2301–4156 | e-ISSN 2460–5719

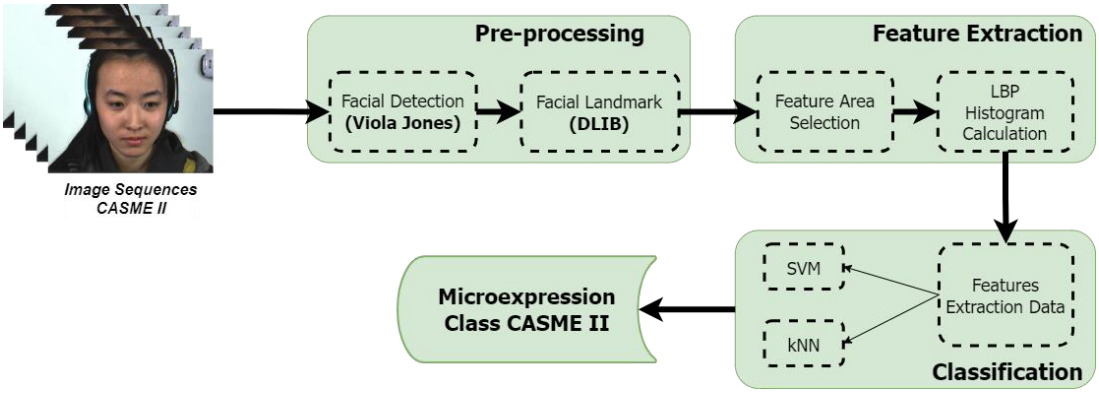

Figure 1. Workflow of the proposed study.

The PCA method has been applied to general expression recognition to show facial expressions [15]. However, it is not suitable for illumination variation [16]. Therefore, a development was undertaken, i.e., the LBP method. This method requires shorter computational time and better tolerance to illumination [13]. One of the developments in the LBP method is the local binary pattern histogram (LBPH). The LBPH method is a combination of the LBP method algorithm and histogram, which is one of the methods suitable for describing textures. In addition, the LBPH method can represent local features in the facial area. It supports facial expression recognition that does not occur in large facial areas. The LBPH method is also a relatively simple method because it involves only a few simple calculation operations, such as binaries and histograms. Consequently, it requires less computation time.

Previous studies have adopted the LBP-method modification technique, i.e., LBP-three orthogonal plane (LBP-TOP) as a feature extraction method [17], [18]. This method allows the texture of a digital image to be described by dividing the image into small sections of features to be extracted. The features consist of binary patterns describing the neighborhood of pixels in the analyzed facial area. The features are then converted into a single feature histogram that forms a representation of the image. The weakness of the study, nevertheless, is the comparatively lengthy computation procedure. The reason is that the observations were conducted on the entirety of the facial area. Consequently, a substantial amount of feature data was produced through a process that required a significant amount of time.

Feature extraction is an important stage in microexpression recognition. The system in microexpression recognition produces good accuracy when the extracted features can reduce within-class variation and maximize between-class variation. A good feature representation can ensure the recognition process runs efficiently with accurate results [19]. In microexpression recognition, muscle movements only appear on some small facial areas. The dominant movements are found in two parts of the face, namely in the eyebrow and mouth areas [20]. Meanwhile, according to other studies, observing facial microexpressions in certain areas requires a shorter computation time, and it is more accurate compared to observing the entire face [21]. Detection of facial component areas can be carried out in various ways, one of which is adopting the landmark point method in the formation of facial component areas [22]. By observing specific areas, the duration of microexpression recognition is shorter compared to observing the entire face [23]. On the other hand, accuracy depends on several things, including selecting the utilized feature types [1].

Based on the problems described, this study proposes a comparative analysis of microexpression phases, namely "fullOAO" and "OAO", on selected facial components. The primary distinction between the two categories of expression phases lies in the number of frames used during the analysis process. The "fullOAO" expression phase is the selected frame from the onset to offset phase, while the "OAO" phase refers to the sole utilization of framing only during the onset, apex, and offset phases. The expression phase comparison aims to determine the performance results of recognition accuracy against the use of a frame based on the expression phase on the dataset. Using the selected frame can provide lower processing time compared to using the entire frame on the video clip of the dataset. The selected facial areas for microexpression recognition are the left eyebrow, right eyebrow, right eye, left eye, and mouth. Based on previous studies [21], [20], these areas were selected as facial areas for feature extraction. The application of limited facial areas is expected to speed up the recognition process and produce representative feature extraction data for the classification process.

II. METHODOLOGY

In a real-time microexpression recognition system, a fast response with precise accuracy is required to identify an expression gesture. Therefore, fast processing time and precise accuracy are inseparable in building a real-time microexpression recognition system. Microexpression recognition requires a balance between accuracy and processing time. In general, the faster a system can recognize and interpret the facial microexpression, the better the response quality of the system. However, to improve accuracy, more complex and sophisticated analysis methods are often used, resulting in longer processing time.

This study proposes the use of the LBPH method for feature extraction in the recognition of microexpressions on human faces. Figure 1 depicts the proposed framework in detail. This study has three main stages: preprocessing, feature extraction, and classification, as shown in Figure 1. This study aims to recognize microexpressions in video image sequences of the Chinese Academy of Science Micro-expressions (CASME II) database. The proposed study contribution is the reduction of the observation area and the comparative analysis of the expression phases, i.e., the onset to offset phase ("fullOAO") and only the onset, apex, and offset phase (OAO). The components of the facial area are the left eyebrow, right eyebrow, right eye, left eye, and mouth [21], [20]. By employing the LBPH method in the feature extraction process,

p-ISSN 2301–4156 | e-ISSN 2460–5719

local features in the motion of selected facial areas can be represented. The LBPH method generates a feature that serves to classify images combined from LBP and histograms. The LBPH method is one of the latest techniques from LBP to modify the performance in terms of recognition, including image recognition.

A. CASME II DATASET BASED ON EXPRESSION PHASE

CASME II dataset is widely used in facial expression analysis and microexpression recognition fields [24]. This Dataset contains video clips of spontaneous microexpressions, which are very brief facial expressions occurring involuntarily and can reveal true emotions. The CASME II dataset consists of 247 video sequences taken from 26 subjects, including both men and women. Each video clip is typically 5 to 10 s long and is recorded at a rate of 200 frames per second (fps) with a spatial resolution of approximately 640 × 480 pixels on the face area. This dataset includes seven basic emotions (expression classes): happiness, sadness, surprise, fear, disgust, anger, and contempt. The data details of each video clip in CASME II, such as file name, number of frames, expression label, and frame expression phase (onset, apex, and offset), are available in file .xls in the dataset.

This study utilizes four selected expression classes: disgust, sadness, surprise, and happiness. The four expression labels are selected based on the different number of videos from each expression class in CASME II. The difference in the number of videos is due to the difficulty of obtaining these expressions, causing an uneven distribution of video clips [25]. Thus, the available CASME II dataset only provides five microexpression classes: disgust, fear, happiness, sadness, and surprise. The five expression classes have an unbalanced number of videos. Therefore, the model's performance also became unbalanced, resulting in data evaluation biased towards most of the data. Therefore, in this study, four expression classes were manually selected by referring to the file details available in CASME II.

Manual selection was carried out by considering almost the same number of frames between onset to offset for each video, as shown in Table I. Frame images are used only at frame onset to offset because the purpose of this study is to conduct a comparative analysis of expression phases onset to offset ("fullOAO") and only at phase onset, apex, and offset (OAO), using selected facial component areas, namely the left eyebrow, right eyebrow, right eye, left eye, and mouth areas. For example, the data of Happiness (17_EP01_15) has a number of frames in the video of 223 images. The number of frames between onset to offset in the "fullOAO" phase in the video is 42 image frames, while for the "OAO" phase, there are three image frames because it only consists of frame onset, apex, and offset.

TABLE I
INFORMATION ON THE NUMBER OF FRAMES IN EACH EXPRESSION CLASS

No.	Class of Expression	Total of Videos	Number of Overall Frames	Total of Frames	
				fullOAO	*OAO*
1	Disgust	7	2,071	269	21
2	Happiness	7	1,529	319	21
3	Sadness	7	2,788	281	21
4	Surprise	6	1,517	286	18
Total of frames		27	7,905	1,155	81

B. PREPROCESSING

The preprocessing stage consists of two main processes: face detection and location search of facial components, such as eyebrows, eyes, and mouth. In the face location search phase, the Viola-Jones method was applied, while for the facial component search, the face landmark method from Dlib [26] was employed.

1) FACE DETECTION USING VIOLA JONES

At this stage, the input data was in the form of a video. Each frame of video was then extracted, and image sequences were generated. Each image is subjected to a grayscaling process throughout the image, which serves to convert the color image (RGB) into a grayscale image. Subsequently, the Viola-Jones method was employed to conduct the facial recognition procedure on every image sequence extracted. This method is widely referred to as the Haar cascade classifier [27]. This method has a relatively fast computation time because in searching for an object, this method only searches for the center of the area where the object is most likely to be found. In addition, the object search results exhibit a notable degree of precision and demonstrate efficiency in detecting faces [22]. The following is an explanation of the approach process to detecting facial objects.

1. Data training was carried out to form a model with two types of data: positive and negative. In this case, positive data is facial image data, while negative data is non-image facial data. This process aims to study the differences between the objects to be detected.
2. Determining Haar-like features by subtracting the average value of pixels in dark and light areas. The feature is considered to be present if the difference in values is at the specified threshold value.
3. Integral image was used to determine the presence or absence of Haar-like features in the processed image.
4. The integral image at (x, y) contains the sum of pixels to the left of and above, as in (1).

$$II(x,y) = \sum_{x' \le x, y' \le y} i(x', y') \quad (1)$$

where $II(x,y)$ is the integral image and $i(x', y')$ is the original image.

5. Equations (2) and (3), $s(x,y)$ is the cumulative row accumulation of $s(x,-1) = 0$ and $II(-1,y) = 0$. The integral image was formulated in a single trajectory passing the original image.

$$s(x,y) = s(x,y-1) + i(x,y) \quad (2)$$

$$II(x,y) = II(x-1,y) + s(x,y) \quad (3)$$

6. A Cascade classifier was utilized to connect multiple features by combining the classification process in a cascaded manner to be more efficient.
7. If the face had been located, then the location of the face was marked with a region of interest (ROI).

2) FACIAL LANDMARK WITH DLIB

A facial landmark is the detection of facial components on the basis of prominent facial points, which serves to form the facial structure as a marker of the facial components. Facial landmarks are able to precisely detect facial components such as the eyebrow, eye, nose, and mouth areas. This study applies a Dlib tool using a regression tree to localize pre-trained 68 points [26]. After the ROI was formed from the results of the previous face detection process, the next step was to determine the feature points for the facial component search. The

p-ISSN 2301–4156 | e-ISSN 2460–5719

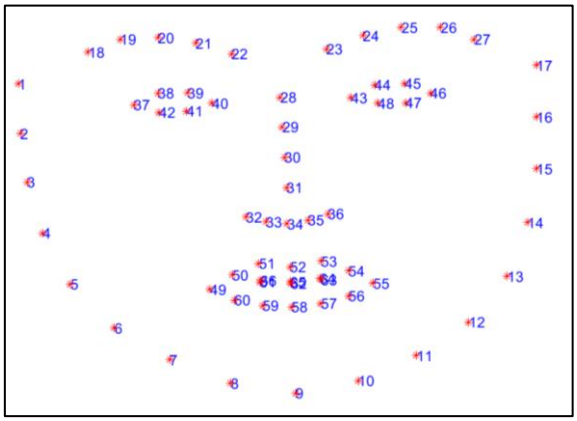

Figure 2. Dlib's 68-point structure for area division of facial components.

application of 68 points by Dlib was performed on the ROI area that had been determined as the face, as shown in Figure 2 [26]. The objective is to minimize errors in facial component detection. The regression tree on the Dlib works using a regressor, which is to make predictions based on features. These features include pixel intensity values calculated from I and dynamic indexing of the estimated recurrence that is taking place. This study uses landmark points only in the eyebrow, eye, and mouth areas, as shown in Figure 2. The use of points is carried out to mark the features of the facial components as follows: right eyebrow (18th to 22nd points), left eyebrow (23rd to 27th points), right eye (37th to 41st points), left eye (43rd to 47th points), and mouth (49th, 55th, 52nd and 58th points). After these points were determined, the detected components were cropped and resized to 100 × 50 pixels on each facial component, as shown in Figure 3.

C. FEATURE EXTRACTION WITH LBPH

LBPH is a method used in computer vision that is applied for object recognition. This method involves breaking the image into smaller areas and calculating the LBP for each area. The LBP is a binary code that represents the intensity relationship of a pixel to its neighboring pixels. By combining the histograms and LBPs of all the areas in all the images, a feature vector is generated. These feature vectors are used for image classification using machine learning algorithms such as k-nearest neighbor (KNN) or support vector machine (SVM). The LBPH method is known for its simplicity [28], [29], efficiency, and robustness in illumination changes. Therefore, this method is popular for microexpression recognition.

Figure 4 illustrates the process of the LBPH method applied in this study. For example, in the first step, the input image used is the right eyebrow image. The size of the image is $I_{w,h}$= 58 × 55. The notation I represents an image matrix with a width (w) of 58 pixels and a height (h) of 55 pixels. The LBP method uses a grayscale input image to reduce calculation complexity and focus on the texture patterns in the image. Then, the second step is to divide the image into blocks of pixels non-overlapping. This study uses a 3 × 3 block division that results in five predefined facial feature areas. The 3 × 3 block size division aims to represent more detailed features in capturing local patterns in the image. The LBP method can provide a better feature representation by considering the neighboring pixels in the block. Microexpression motion has subtle movement characteristics. The smaller the part of the block processed, the more detailed the resulting features. The LBP method with 3 × 3 blocks uses the center value as the threshold for its neighboring pixels.

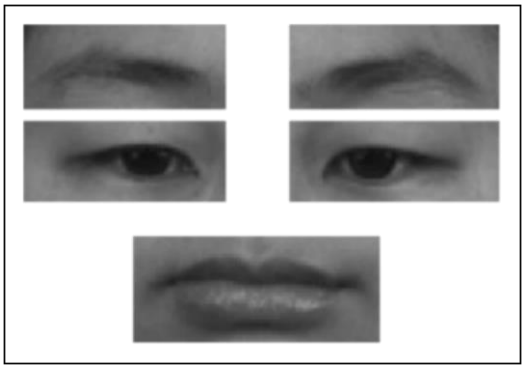

Figure 3. Results of facial component cutting for feature extraction.

$$LBP_{P,R} = \sum_{n=0}^{P-1} S(x_n - x_c)2^n, \;\; S(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (4)$$

In (4), the notation (P, R) is the sampling point (number of neighbors) or point P at radius R. The x_n notation is the value of the neighboring pixel, while x_c is the value of the center pixel. The 2^n notation is the weight value of the sum of the powers of n. $S(x)$ is the determination of the threshold value at $S(x_n - x_c)$. If the value of x is greater than or equal to zero, $S(x)$ is 1. Otherwise, $S(x)$ is 0.

After the block division process in the image, the LBP calculation in this study uses the value $P = 8$ with a radius $R = 1$ ($LBP_{8,1}$). The LBP calculation starts from the matrix $(0,0)$ or the upper left area. The division of blocks does not occur evenly throughout the image, so it is necessary to add pixels with a zero value (zero padding) around the edge of the image so that there is no loss of movement in the image. The 3 × 3 matrix is converted into a binary matrix by comparing the center value (x_c) with its neighboring values in a clockwise direction. If the intensity value of a neighboring pixel is greater than or equal to that of the center pixel, the binary value is 1. If the opposite is true, the binary value is 0. After the binary matrix is formed, the next process is to multiply it with the weight of 2^n, i.e., $2^8 = 2^0 + 2^1 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7$. Next, there was the accumulation of the result of multiplying the binary matrix and the 2^n weight of all pixels in the block, thus forming a new center value representing the block.

All blocks in the image were processed according to their order, thus forming a new matrix of LBP calculation results. The LBP matrix of all facial components is represented in a histogram. This entire process is illustrated in Figure 5(a). The histogram serves to represent the frequency distribution of local binary patterns in the image. Each bin on the histogram represents one local binary pattern, while the height of the bin indicates the number of occurrences of that pattern. Equation (5) is a formula for forming the histogram of each facial feature component. The notation $S(x)$ is a vector of the number of occurrences of categorized pixels according to the value of the bin. The notation n is the frequency of occurrence; in $LBP_{8.1}$, is 256 in total, with range values of 0 to 255.

$$H_i = \sum_{x_c \in i_{(x,y)}} f\left\{LBP_{P,R} = \sum_{n=0}^{P-1} S(x)\right\} \quad (5)$$

$$S(x) = \begin{cases} 1, & if\ y\ is\ true \\ 0, & if\ y\ is\ false \end{cases}, \;\; i = 0,1,\dots,n-1$$

$$h_i = \frac{n_i}{n}, i = 0,1,\dots,L-1. \quad (6)$$

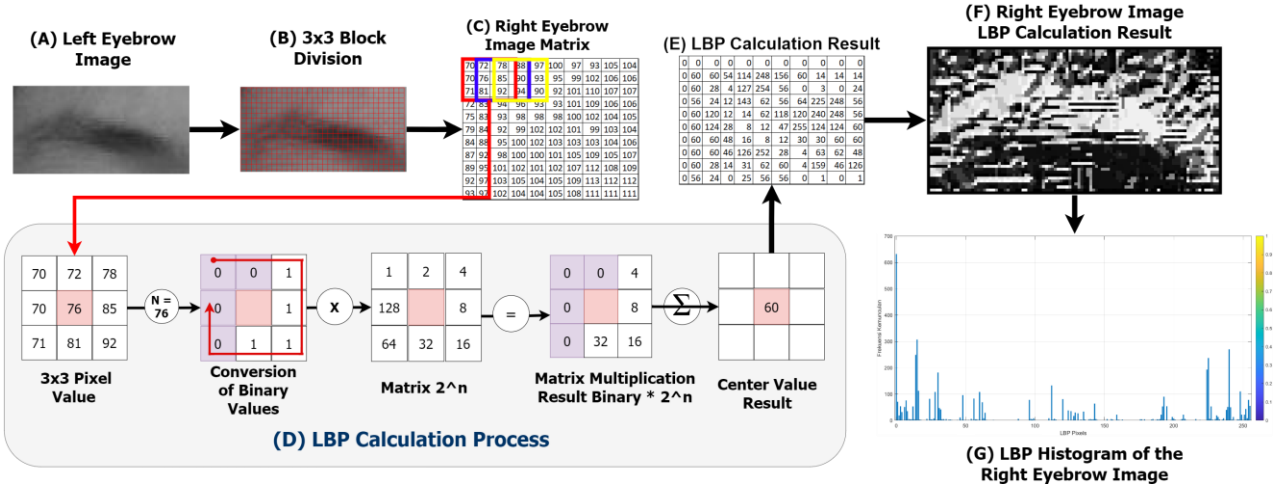

Figure 4. Workflow of the LBPH method on the *grayscale* image of the right eyebrow.

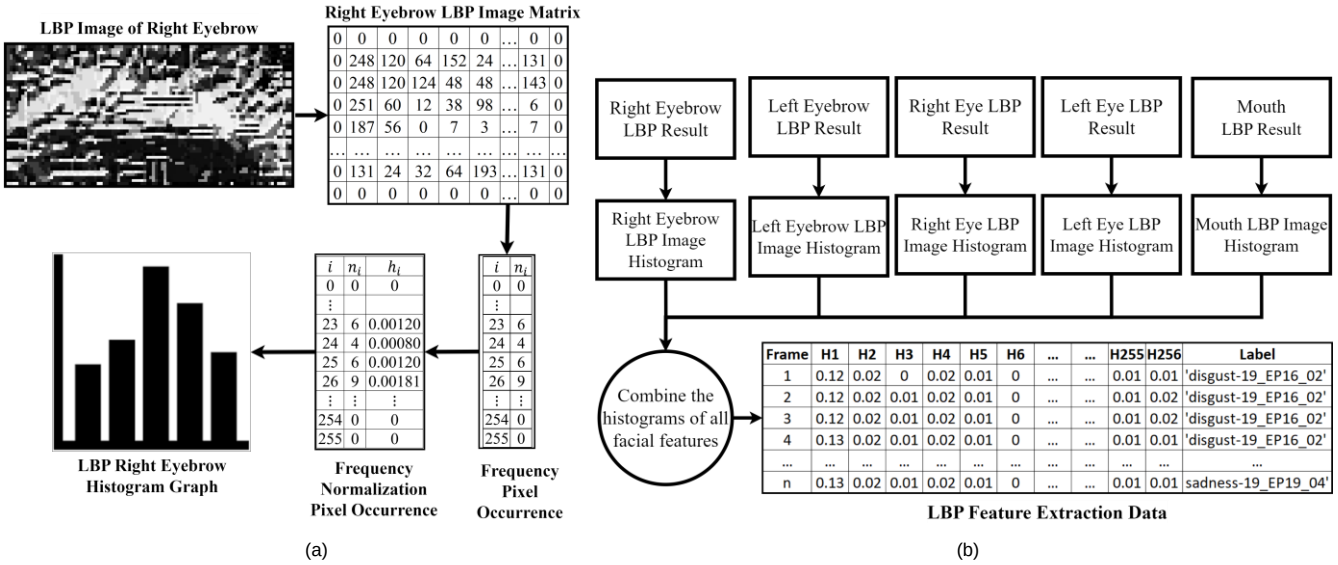

Figure 5. Conversion to histogram (a) the stage of converting the LBP matrix into a histogram, (b) the histogram of all feature area components.

The histogram calculation on each facial component begins with determining the histogram bin according to the LBP pattern. The present study employs an 8-bit pattern known as LBP ($LBP_{8,1}$), resulting in a total of 256 histogram bins. The number of bins corresponds to the pixels in the image, which is between 0 and 255, totaling 256. Furthermore, the frequency of occurrence of each LBP value in the image is calculated by creating a matrix whose size corresponds to the number of histogram bins, i.e., 0 to 255. LBP image histogram data has a wide range of values, so it can provide inaccurate predictions in the classification process. Therefore, a data normalization process is required. In the histogram data normalization in (6), the value of n_i is the number of pixels of gray degree, n is the number of all pixels in the image, and L is the number of bins in the histogram.

$$H_{i,j} = \sum_{x_c \in R_j} f\{LBP_{P,R} = \sum_{n=0}^{P-1} S(x)\},\ j = 0, 1, \ldots, m-1. \quad (7)$$

The next process is merging all histogram data from each facial component. The histogram data merging was carried out by summing the frequency of occurrence according to the value of the bin. Equation (7) shows the histogram formula to store the entire area of the facial component of each image marked with the notation m area in each frame. The notation $H_{i,j}$ is the value i of (5) which indicates the order of the facial area components of the histogram j. The regional histograms (per area) were combined to create a global description of the image. The merging of all histograms of the component face areas is illustrated in Figure 5(b).

The merged histogram data shown in Figure 5(b) is the feature extraction data for the classification process. In this study, analysis was conducted in comparing microexpression phase data, namely "fullOAO" (image data from frame onset to offset) and "OAO" (image data were only frame onset-apex-onset). The feature extraction data information has three types of attributes, namely frame, histogram data, and the label of each data, as shown in Figure 5(b) in the feature extraction data section. The main attribute as feature data is the result of histogram data with bin 0 to 255 with frequency of occurrence with normalized value.

D. FEATURE REDUCTION (CUR MATRIX DECOMPOSITION)

The histogram of each facial image component results in a large dimension, which is illustrated in Figure 5(b). Each frame in the image produces 256 features, so it is necessary to eliminate uninformative features for the data classification process. This feature reduction aims to eliminate irrelevant features or redundancies while retaining the important information contained in the dataset. The CUR matrix decomposition technique can obtain a more efficient data representation of the previous feature data by retaining the most

p-ISSN 2301–4156 | e-ISSN 2460–5719

important subset of columns and rows. It can be beneficial in the condition that the original feature data have large row dimensions because this technique can reduce computational complexity and data storage requirements.

This study performs the feature reduction process using the CUR matrix decomposition method. The method can select features that show strong statistical influence, or can have a large influence, from the matrix data. This technique breaks the input matrix into three smaller matrices: *C*, *U*, and *R*. The following are the steps of feature reduction with CUR matrix decomposition.

1. Column selection (*C*), which was to select a number of columns from the original matrix (*A*) that would form the matrix by the random selection method by the system.
2. Row selection (*R*), which was to select a number of rows to form a matrix *R* with selection based on the importance of the rows (currently all data frames in the feature extraction data are considered important).
3. Center matrix formation (*U*), i.e., selecting a subset of columns from the feature extraction matrix. In selecting the subset in this study, random sampling was employed.
4. Weight calculation was performed after obtaining the matrices *C*, *U*, and *R*. Weights were calculated for each selected row and column using (8) and (9).

$$W_{C(i,i)} = \frac{1}{sqrt\left(\left\|col_{i(A)}\right\|_2 \times \left\|col_{i(C)}\right\|_2\right)} \quad (8)$$

$$W_{R(i,i)} = \frac{1}{sqrt\left(\left\|row_{i(A)}\right\|_2 \times \left\|row_{i(R)}\right\|\right)} \quad (9)$$

where $W_{C(i,i)}$ are the weights for the selected columns in matrix *C*and $W_{R(i,i)}$ are the selected weights for the selected rows in matrix *C*and *R*. The purpose of weight calculation is to determine the relative contribution of each row and selected column of the input matrix.

5. The approximation of the input matrix is performed after obtaining the *C*, *U*, *R* matrices and weights. The approximation of the input matrix *A'* can be calculated by combining the three matrices, using (10).

$$A' = C \times W_C \times U \times W_R \times R \quad (10)$$

where W_C and W_R are the diagonal weight matrices for the selected columns and rows of the input matrix approximation.

E. MICROEXPRESSION CLASSIFICATION USING KNN AND SVM

1) K-NEAREST NEIGHBORS (KNN)

This study implements the KNN method to model the data. The principle of this method is that every point in adjacent data will be in the same label/class. That is, the KNN method can classify new data based on the similarity of the available models [30]. This method is a type of actuation-based learning where the functions are local proximity values. In this study, the classification stage compares three distance calculation methods: Euclidean, Manhattan, and Chebyshev. The following are the steps to calculate the KNN.

$$Euclidean(a,b) = \sqrt{\sum_{i=1}^{n}(x_{ai} - y_{bi})^2} \quad (11)$$

$$Manhattan(a,b) = \sum_{i=1}^{n}|x_{ai} - y_{bi}| \quad (12)$$

$$Chebyshev(a,b) = \max_i(|x_{ai} - y_{bi}|) \quad (13)$$

where a is the center data cluster, b is the data in the feature attribute, i is the notation for each data in n, n indicates the number of data, x_{ai} denotes the data in the center cluster up to k, while y_{bi} is each data object up to k.

The Euclidean distance method is considered the foundation of the similarity or dissimilarity method. In this study, (11) is used to calculate the Euclidean distance between corresponding elements of two vector spaces. The Chebyshev distance calculation is performed using the largest difference of the two-input data, as in (13). Meanwhile, the Manhattan distance method is another method to measure the distance between two vectors, as shown in (12).

2) SUPPORT VECTOR MACHINE (SVM)

The SVM method is used in the study because it proves to be successful in statistical learning, data optimization modeling, object detection, as well as microexpression recognition. Mathematically, classification modeling in SVM has a better concept than other classification techniques. The model in SVM can solve both linear and nonlinear classification-related problems. In this study, functions with linear, polynomial, radial basis function (RBF), and sigmoid functions are applied to SVM. The mathematical equations of the four kernels are listed in (14) to (16).

$$Linear\ K(x_i, x_j) = x_i^T x_j + c \quad (14)$$

$$polynomial\ K(x_i, x_j) = (c + x_i^T x_2)^b \quad (15)$$

$$RBF\ K(x_i, x_j) = \exp\left(\frac{\|x_i - x_j\|}{2S^2}\right) \quad (16)$$

where x_i and x_j are input vectors, $K()$ is the kernel function, c is a constant, and S is the bandwidth parameter that determines the descent rate. metric similarity as the samples moves away from each other. The notation x_i^T is the transposition value of the weight vector x, while the notation b is the value of the polynomial degree.

III. RESULT AND DISCUSSION

A. DATA SETUP

This study compares the accuracy of microexpression recognition in the two types of expression phases presented in Table I. The "fullOAO" phase type uses all frames in the phase from onset to offset, while the "OAO" type only uses three frames, consisting of onset, apex, and offset frames. This study utilizes 27 selected videos from CASME II with four data labels. Further description is presented in Table I. The purpose of video selection is to consider the number of frames from each onset to offset phase, to avoid imbalance in the amount of data. In this study, microexpression recognition is performed using five facial components: the left eyebrow, right eyebrow, left eye, right eye, and mouth. The result of the LPBH method is the combination of the number of occurrence frequencies on five facial components that have the form of a data series of combinations of the number of pixels with a value of 0 to 255. By using five facial components, it can be concluded that the feature extraction data is high-dimensional.

Table II presents information on the number of feature dimensions in "fullOAO" and "OAO" of 256 features in each frame. The histogram process in feature extraction produces a distribution of 0-valued and equal-valued data in some pixels. It has an impact on the classification process, such as data redundancy, so a data reduction process is required using the

p-ISSN 2301–4156 | e-ISSN 2460–5719

TABLE II
NUMBER OF MICROEXPRESSION PHASE DATA COMPARISON FRAMES

No.	Phase Type	Total Frame	Dimension Feature	Reduction of Feature Dimensions
1	fullOAO	1,155	256	35
2	OAO	82	256	35

CUR matrix decomposition technique. A description of the number of frames in the data reduction, phase type, and expression class is presented in Table II. The utilization of this technique results in a smaller number of features, i.e., 35 data. However, it is considered to have more effective representation by retaining the most important or significant features.

B. MEASUREMENT OF MICROEXPRESSION RECOGNITION PERFORMANCE

In this study, SVM and KNN were selected as the methods for experimental classification. The selection of the kernel function for SVM and the distance measurement method is critical to the performance of each method. Experiments were conducted on CASME II using the LBPH feature extraction method. SVM and KNN classifiers are used to evaluate the proposed methods using multi-subject analysis, leave-one-out cross-validation (LOSOCV), and 10-cross validation. LOSOCV is a cross-validation method used to estimate the performance of a model or algorithm on a limited (small number of) dataset. The use of both analysis techniques aims to obtain the most accurate estimation of model performance. In validating the effectiveness of the proposed method in LOSOCV, the image sequence of one subject was treated as testing data and the remaining frame images were treated as training data. The process was repeated for k times, where k indicates the number of subjects in the dataset. Then, the average recognition results of all subjects were calculated to form the final recognition accuracy.

In 10-fold cross validation, the data set was divided into ten parts. Then, nine parts of the data were taken as training data in turn and one part as test data. The correct rate was obtained from each test and the average value of the correct rate was used to estimate the accuracy of an algorithm. As the last stage, this study conducted 10-fold cross-validation and obtained the average value as the highest accuracy. In this study, two commonly used cross-validation methods are used to evaluate the ability of classification methods with SVM and KNN methods for recognition.

1) TESTING OF CLASSIFICATION WITH COMPARISON OF K VALUES AND DISTANCE METRICS ON KNN

The KNN test was conducted using the samples listed in Table III. The distance calculation in KNN uses three comparison methods: Euclidean, Manhattan, and Chebyshev. The determination of the k value in KNN depends on the input data. A high value of k will reduce the effect of noise but will form the boundaries of each classification result. This study uses three kinds of odd values for k, namely 1, 3, and 5. Since the number of data labels is even, this study uses odd values of k.

The calculation results of Euclidean, Manhattan, and Chebyshev distances against KNN accuracy are shown in Table IV. On "fullOAO" data with LOSOCV analysis and 10-cross validation, the average result is above 90% at each k value. The highest accuracy result using the value of k = 5 and the calculation of Manhattan distance, with LOSOCV and 10-cross validation accuracy of 96.6%. In "fullOAO." the different values of k do not affect the accuracy rate because at each value of k in Table III, the accuracy generated by each method is above 90%. In the "OAO" phase type data, the use of larger k values resulted in a smaller accuracy in each analysis. The highest accuracy value occurs at k = 1 with Manhattan distance measurement with a value of 87.7%. Manhattan corresponds well to geometric data, where the distance between points can be measured based on the absolute difference between their coordinates in each dimension.

TABLE III
COMPARISON OF DISTANCE METRICS OF KNN IN "FULLOAO" AND "OAO PHASES

KNN - fullOAO						
k	*LOSOCV (%)*			*10-Cross Validation (%)*		
	EC	*MH*	*CS*	*EC*	*MH*	*CS*
1	95.8	95.9	94.3	95.5	95.6	94
3	96.2	95.5	94.3	96.1	95.4	94.2
5	96.1	96.6	95	96.3	96.6	96.3
KNN - OAO						
k	*LOSOCV (%)*			*10-Cross Validation (%)*		
	EC	*MH*	*CS*	*EC*	*MH*	*CS*
1	84.0	87.7	81.5	85.2	87.7	80.2
3	81.5	84.0	81.5	79.0	85.2	77.8
5	76.5	79.0	74.1	65.4	77.8	67.9

Note: EC = Euclidean, MH = Manhattan, CS = Chebyshev

TABLE IV
COMPARISON OF KERNEL SVM IN "OAO" AND "FULLOAO" PHASE

Kernel	fullOAO		OAO	
	LOSOCV (%)	*10-Cross (%)*	*LOSOCV (%)*	*10-Cross (%)*
Linear	53.9	53.7	64.2	63
Polynomial	96.3	96.3	81.5	81.5
RBF	96.7	96.8	74.1	71.6
Sigmoid	13.6	15.6	39.5	46.9

2) CLASSIFICATION MEASUREMENT WITH KERNEL SVM COMPARISON

Testing with SVM was carried out using the same sample as described earlier. In the SVM classification process, three important parameters need to be selected and adjusted. The first parameter is the kernel. This study compares four commonly used kernels in SVM: linear, polynomial, RBF, and sigmoid. The second parameter is the penalty coefficient, *C*, which is the fault tolerance value. The value can minimize training error and model complexity. The higher C value indicates that the error cannot be, and it is easy to over-fit. The smaller the C value, the easier it is to under-fit [31]. Then, the third parameter is the gamma parameter, which is the parameter of the RBF function when selected as the kernel. The RBF width value would affect the range of action of each support vector corresponding to Gauss, thus affecting the ability in general. Table IV presents the analysis results with two types of phases, the "full AOA" and "OAO" phases, using LOSOCV and 10-cross validation. The "fullOAO" data with kernel RBF in both learning methods produced the highest values of 96.7% and 96.8%. The SVM method with kernel RBF can separate high-dimensional data that is difficult to separate with linear methods. Unlike the previous data type, the "OAO" kernel polynomial produced the highest accuracy, which was 81.5% in both methods. It happens because the kernel polynomial can capture more complex

p-ISSN 2301–4156 | e-ISSN 2460–5719

relationships between features compared to other kernel methods.

C. TESTING OF COMPUTATION TIME OF MICROEXPRESSION RECOGNITION

In addition to testing the accuracy performance, this study tested the computation time of the proposed algorithm (see Table V). The computation time of the algorithm is analyzed on an 11th Gen Intel® Core™ i3-1115G4 @ 3,00 GHz processor with 8 GB of random-access memory (RAM). The computation time required to extract features on each frame averages 0.192 ms for fullOAO and 0.159 ms for OAO. Computation time on the expression phase type results in almost the same value because basically every frame is processed with the same algorithm. Unlike the per-frame testing, the per-video testing produces a different time. In the "fullOAO" phase type, an average computation time of 10.473 ms was generated, while in OAO it was 0.576 ms.

The time difference between the two types of phases is due to the difference in the number of frames processed. Technically, the number of frames in "fullOAO" is larger than in the "OAO" phase type (see Table I), thus impacting the processing time of each video.

The implementation of microexpression recognition systems is generally carried out in real-time. As a result, it requires fast computation time to process each frame. Basically, the characteristics of microexpressions are fast in duration, with small and subtle movement intensity, making them difficult to identify by the ordinary human eye. From this statement, it can be concluded that the microexpression recognition system requires fast computation time to process each frame in the input video. In this study, the "OAO" phase type, especially in the video test, resulted in a computation time 20 times faster than "fullOAO". This is due to the different number of frames processed in the two types of frames. In the "OAO" phase type, the frame processed is the core frame (microexpression phase: onset, apex, and offset), while in "fullOAO", the frame processed is from onset to offset frames, so it takes a longer time.

D. RESULTS OF PERFORMANCE ACCURACY AND COMPUTATION TIME TOWARDS EXPRESSION PHASE TYPES

In the previous section, the classification stage was carried out by comparing KNN and SVM methods with various combinations of kernels and distance measurements. Based on KNN and SVM classification testing, it has been observed that the "fullOAO" phase type yields an accuracy that is nearly equivalent to that of SVM (96.8%) and KNN (96.6%). In contrast, in the "OAO" phase type, a marginal decrease in accuracy is observed when compared to the "fullOAO" phase, at 81.5% for SVM and 87.7% for KNN. The decrease in accuracy occurs due to the different amounts of data from each type of expression phase. In "fullAOA", the number of frames is 1,115, while in "OAO" there are 82 frames (see Table II). Based on the findings of the accuracy performance results, it can be inferred that an increase in the number of *frames* analyzed leads to a corresponding improvement in the achieved accuracy level.

In order to evaluate the performance of the algorithm proposed in this study, this section presents a comparative analysis with prior studies employing diverse methodologies, as depicted in Table VI. All of the studies included in Table VI utilized the CASME II dataset and employ the LOSOCV

TABLE V
COMPARISON OF COMPUTATION TIME OF EACH PHASE TYPE

No.	Testing Type	fullOAO (ms)	OAO (ms)
1	Average per frame	0.192	0.159
2	Average per video	10.473	0.576

TABLE VI
COMPARISON OF ACCURACY WITH OTHER STUDIES WITH LOSOCV

Study	Feature Extraction	Classifier	Performance (%)
[32]	LBP-TOP	SMO	68.24
[33]	Bi-WOOF	SVM	58.85
[34]	OFF-ApexNet	CNN	88.28
This study	LBPH	SVM	96.80

classification technique, employing various classification methods, such as sequential minimal optimization (SMO), convolutional neural network (CNN), and SVM. According to the findings presented in Table VI, the previous study reported the highest accuracy of 68.24% using the LBP-TOP and SMO techniques [32]. Additionally, employing bi-weighted oriented optical flow (Bi-WOOF) in conjunction with SVM resulted in an accuracy of 58.85% [33]. The reference cited in [34] demonstrated superior accuracy compared to previous studies utilizing OFF-ApexNet and CNN, achieving an accuracy rate of 88.28%. In addition, the algorithm proposed in this study utilizes LBPH and SVM, which results in an accuracy of 96.8%.

The accuracy level of microexpression recognition is very important because accurate results are needed to understand the feelings and emotions in each input expression data. This study presents a proposed feature extraction method utilizing LBPH, which achieves a final accuracy of 87.7% in KNN and 96.8% in SVM. The difference in accuracy is relatively small, standing at 9%, with the minimum accuracy deemed satisfactory surpassing 80%. Furthermore, the feature extraction data utilized as input data for classification is identical; thus, the difference in accuracy does not significantly affect the microexpression recognition. The aforementioned outcomes can be classified as satisfactory because the proposed method has a strong performance in recognizing microexpressions on facial areas. The microexpression recognition process involves real-time processing within a brief timeframe. In addition to satisfactory accuracy results, it is imperative to take into account the processing time required for recognition.

Table VII summarizes the processing time on each frame from previous studies [17], [18] and the proposed study. This section provides no definitive conclusions regarding the difference in computation time observed in each study. It is due to the different specifications of the tools used in each study and the preprocessing stages applied. In [18], tensor unfolding with graphics processing unit (GPU) was utilized, and it resulted in a computation time of 107.39 ms, while [17] used an LBP-TOP-based GPU with the CUDA platform, which provided a computation time of 2.98 ms. In both studies, preprocessing steps are required, including face detection from each frame, face alignment, face cropping, and phase detection of onset to offset. During the processing phase, some of these stages require a considerable amount of time. In contrast, the present study demonstrates that the processing time for each frame of the entire selected facial components is 0.175 ms. The computational time of the proposed study results in a faster process compared to previous studies [17], [18]. Nevertheless,

TABLE VII
COMPUTATIONAL PERFORMANCE FOR THIS STUDY AND PREVIOUS STUDIES

Study	Feature extraction	Preprocessing	Time per Frame (ms)
[17]	LBP-TOP	Face detection, face cropping, face alignment, and onset-offset detection	2.980
[18]	LBP-TOP	Face detection, face alignment, face cropping, and onset-offset detection	107.390
This study	LBPH	Facial landmarks detection, facial component cropping	0.175

the proposed study does not incorporate an automated phase onset to offset detection process, which has been shown to require less computational time in comparison to previous studies.

IV. CONCLUSION

Microexpression recognition requires a balance between accuracy and processing time. Typically, the higher the speed at which a system can recognize and interpret facial microexpression s, the enhanced the quality of the system's response. However, in order to improve accuracy, more complex methods and sophisticated analysis methods are frequently implemented, resulting in longer processing times. This study proposes a simple method, LBPH, for the feature extraction process. A comparison of expression phases in microexpression based on core phases (onset, apex, and offset) was applied in this study. The proposed method is expected to provide fast computation time and precise microexpression recognition. In this study, a comparison of the phase types of microexpression, namely "fullOAO" (phases onset to offset) and "OAO" (phases onset, apex, and offset) with the LBPH feature extraction method was performed.

The experiments in this study utilized one of the microexpression datasets, CASME II. In this study, SVM and KNN were selected as methods for experimental classification. The selection of kernel function for SVM and the distance measurement method is critical to the performance of each method. In the multi-subject analysis, LOSOCV and 10-cross validation were used to validate the effectiveness of the method proposed in LOSOCV, with the image sequence from one subject treated as the testing data and the remaining frame images as the training data.

The highest accuracy achieved in "fullOAO" (SVM-RBF) is 96.8%, whereas the accuracy in "OAO" (KNN-Manhattan) is 87.7%. The difference in accuracy in the expression phase type is inversely proportional to the computation time, which is 20 times faster for "OAO" than for "fullOAO". The execution time for the "OAO" is 0.576 ms, whereas the "fullOAO" requires a longer execution time of 10.473 ms. The difference in computation time is due to the different number of frames processed in the two types of expression phases. Technically, the "fullOAO" type contains a greater number of frames than the "OAO" type; consequently, the processing time is also distinct.

The observed decline of 9% in accuracy across the two types of phases has a relatively minimal impact, as the accuracy remains above 80% in both types of expression phases. Moreover, the difference in the computation time of the two types of expression phases depends on the number of frames of each input video. Therefore, an appropriate measurement for computation time is the time required to process each frame in the input video. The present study demonstrates that the processing time of each frame at "fullOAO" is 0.192ms faster than previous studies. Therefore, it can be concluded that the proposed algorithm achieves fast computation time with accurate microexpression recognition.

CONFLICT OF INTEREST

The authors guarantee that there are no conflicts of interest, either in specific circumstances or personal interests, that impact the presentation or interpretation of the study results.

AUTHOR CONTRIBUTION

Conceptualization, Ulla Delfana Rosiani and Priska Choirina; methodology, Ulla Delfana Rosiani and Yessy Nindi Pratiwi; software, Yessy Nindi Pratiwi and Septiar Enggar Sukmana; validation, Ulla Delfana Rosiani and Priska Choirina; formal analysis, Septiar Enggar Sukmana.

REFERENCES

[1] L. Zhou, X. Shao, and Q. Mao, "A Survey of Micro-Expression Recognition," *Image, Vis. Comput.*, Vol. 105, pp. 1–11, Jan. 2021, doi: 10.1016/j.imavis.2020.104043.

[2] P. Ekman, "Lie Catching and Microexpressions," in *The Philosophy of Deception*, C. Martin, Ed., New York, USA: Oxford University Press, 2009.

[3] P. Ekman and W.V. Friesen, *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Palo Alto, USA: Consulting Psychologists Press, 1978.

[4] M. Peng, Z. Wu, Z. Zhang, and T. Chen, "From Macro to Micro Expression Recognition: Deep Learning on Small Datasets Using Transfer Learning," *2018 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG 2018)*, 2018, pp. 657–661, doi: 10.1109/FG.2018.00103.

[5] C.H. Yap, C. Kendrick, and M.H. Yap, "SAMM Long Videos: A Spontaneous Facial Micro- and Macro-Expressions Dataset," 2019, *arXiv:1911.01519*.

[6] P. Ekman and W.V. Friesen, "Nonverbal Leakage and Clues to Deception," *Psychiatry*, Vol. 32, No. 1, pp. 88–106, 1969, doi: 10.1080/00332747.1969.11023575.

[7] H.-X. Xie, L. Lo, H.-H. Shuai, and W.-H. Cheng, "AU-assisted Graph Attention Convolutional Network for Micro-Expression Recognition," *Proc. 28th ACM Int. Conf. Multimed.*, 2020, pp. 2871–2880, doi: 10.1145/3394171.3414012.

[8] J. Ma, H. Tang, W.-L. Zheng, and B.-L. Lu, "Emotion Recognition Using Multimodal Residual LSTM Network," *Proc. 27th ACM Int. Conf. Multimed.*, 2019, pp. 176–183, doi: 10.1145/3343031.3350871.

[9] I.P. Adegun and H.B. Vadapalli, "Facial Micro-Expression Recognition: A Machine Learning Approach," *Sci. Afr.*, Vol. 8, pp. 1–14, Jul. 2020, doi: 10.1016/j.sciaf.2020.e00465.

[10] Y. Zhu, Z. Chen, and F. Wu, "Multimodal Deep Denoise Framework for Affective Video Content Analysis," *Proc. 27th ACM Int. Conf. Multimed.*, 2019, pp. 130–138.

[11] A.M. Buhari *et al.*, "FACS-Based Graph Features for Real-Time Micro-Expression Recognition," *J. Imaging*, Vol. 6, No. 12, pp. 1–20, Dec. 2020, doi: 10.3390/jimaging6120130.

[12] S.-J. Wang *et al.*, "Micro-Expression Recognition Using Robust Principal Component Analysis and Local Spatiotemporal Directional Features," in *Computer Vision - ECCV 2014 Workshops*, L. Agapito, M.M. Bronstein, and C. Rother, Eds., Cham, Switzerland: Springer International Publishing, 2015, pp. 325–338, doi: 10.1007/978-3-319-16178-5_23.

[13] Y. Wang, J. See, R. C.-W. Phan, and Y.-H. Oh, "LBP with Six Intersection Points: Reducing Redundant Information in LBP-TOP for Micro-expression Recognition," in *Computer Vision – ACCV 2014*, D. Cremers, I. Reid, H. Saito, and M.-H. Yang, Eds., Cham, Switzerland: Springer International Publishing, 2015, pp. 525–537, doi: 10.1007/978-3-319-16865-4_34.

[14] P. Zhang *et al.*, "Micro-Expression Recognition System," *Optik*, Vol. 127, No. 3, pp. 1395–1400, Feb. 2016, doi: 10.1016/j.ijleo.2015.10.217.

[15] M.N. Patil, B. Iyer, and R. Arya, "Performance Evaluation of PCA and ICA Algorithm for Facial Expression Recognition Application," in *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*, SocProS 2015, Vol. 1, M. Pant *et al.*, Eds., Singapore, Singapore: Springer, 2016, pp. 965–976, doi: 10.1007/978-981-10-0448-3_81.

[16] W.-L. Chao, J.-J. Ding, and J.-Z. Liu, "Facial Expression Recognition Based on Improved Local Binary Pattern and Class-Regularized Locality Preserving Projection," *Signal Process.*, Vol. 117, pp. 1–10, Dec. 2015, doi: 10.1016/j.sigpro.2015.04.007.

[17] X.R. Soh, V.M. Baskaran, A.M. Buhari, and R.C.-W. Phan, "A Real Time Micro-Expression Detection System with LBP-TOP on a Many-Core Processor," *2017 Asia-Pacific Signal, Inf. Process. Assoc. Annu. Summit, Conf. (APSIPA ASC)*, 2017, pp. 309–315, doi: 10.1109/APSIPA.2017.8282041.

[18] X. Hong, Y. Xu, and G. Zhao, "LBP-TOP: A Tensor Unfolding Revisit," in *Computer Vision – ACCV 2016 Workshops*, C.-S. Chen, J. Lu, and K.-K. Ma, Eds., Cham, Switzerland: Springer International Publishing, 2017, pp. 513–527, doi: 10.1007/978-3-319-54407-6_34.

[19] N. Samadiani *et al.*, "A Review on Automatic Facial Expression Recognition Systems Assisted by Multimodal Sensor Data," *Sens.*, Vol. 19, No. 8, pp. 1–27, Apr. 2019, doi: 10.3390/s19081863.

[20] P. Choirina, U.D. Rosiani, and I.M. Fitriani, "Pengenalan Ekspresi Mikro Wajah Berdasarkan Point Feature Tracking Menggunakan Fase Apex pada Database Ekspresi Mikro," *Edu Komputika J.*, Vol. 9, No. 1, pp. 28–36, Jun. 2022, doi: 10.15294/edukomputika.v9i1.56600.

[21] U.D. Rosiani *et al.*, "A Novel Approach on Motion Estimation for Micro-Expression Recognition Using Phase Only Correlation with All Block Search (POC-ABS)," *Int. J. Intell. Eng., Syst.*, Vol. 13, No. 6, pp. 546–559, 2020, doi: 10.22266/ijies2020.1231.48.

[22] U.D. Rosiani, P. Choirina, S. Sumpeno, and M. Hery P., "Menuju Pengenalan Ekspresi Mikro: Pendeteksian Komponen Wajah Menggunakan Discriminative Response Map Fitting," *J. Nas. Tek. Elekt., Teknol. Inf.*, Vol. 7, No. 2, pp. 204–211, May 2018, doi: 10.22146/jnteti.v7i2.424.

[23] S.P.T. Reddy, S.T. Karri, S.R. Dubey, and S. Mukherjee, "Spontaneous Facial Micro-Expression Recognition Using 3D Spatiotemporal Convolutional Neural Networks," *2019 Int. Joint Conf. Neural Netw. (IJCNN)*, 2019, pp. 1–8, doi: 10.1109/IJCNN.2019.8852419.

[24] W.-J. Yan *et al.*, "CASME II: An Improved Spontaneous Micro-Expression Database and the Baseline Evaluation," *PLoS ONE*, Vol. 9, No. 1, pp. 1–8, Jan. 2014, doi: 10.1371/journal.pone.0086041.

[25] "CASME II Database," [Online], http://casme.psych.ac.cn/casme/e2, access date: 26-Jun-2023.

[26] M. Xu, D. Chen, and G. Zhou, "Real-Time Face Recognition Based on Dlib," in *Innovative Computing*, C.-T. Yang, Y. Pei, and J.-W. Chang, Eds., Singapore, Singapore: Springer, 2020, pp. 1451–1459, doi: 10.1007/978-981-15-5959-4_177.

[27] T.Q. Vinh and N.T.N. Anh, "Real-Time Face Mask Detector Using YOLOv3 Algorithm and Haar Cascade Classifier," *2020 Int. Conf. Adv. Comput., Appl. (ACOMP)*, 2020, pp. 146–149, doi: 10.1109/ACOMP50827.2020.00029.

[28] N.M. Parsania, K.H. Solanki, and A.R. Mehta. (3-4 Sep. 2021). Innovative Approach for Fingerprint Recognition Using LBP and PCA Algorithms. Presented at 1st Int. Conf. Adv. Inf. Technol., Commun. (IC-AITC), [Online], https://www.youtube.com/watch?v=8x3vsNyfLg0

[29] V. Esmaeili, M.M. Feghhi, and S.O. Shahdi, "Micro-Expression Recognition Using Histogram of Image Gradient Orientation on Diagonal Planes," *2021 5th Int. Conf. Pattern Recognit., Image Anal. (IPRIA)*, 2021, pp. 1–5, doi: 10.1109/IPRIA53572.2021.9483500.

[30] D. Rahmawati *et al.*, "The Design of Facial Expression Detection System to Determine the Level of Customer Satisfaction Using K-Nearest Neighbor Method," *MATEC Web Conf.*, Vol. 372, 2022, pp. 1–5, doi: 10.1051/matecconf/202237206002.

[31] H. Pan *et al.*, "Hierarchical Support Vector Machine for Facial Micro-Expression Recognition," *Multimed. Tools, Appl.*, Vol. 79, pp. 31451–31465, Nov. 2020, doi: 10.1007/s11042-020-09475-4.

[32] A.K. Davison, W. Merghani, and M.H. Yap, "Objective Classes for Micro-Facial Expression Recognition," *J. Imag.*, Vol. 4, No. 10, pp. 1–13, Oct. 2018, doi: 10.3390/jimaging4100119.

[33] S.-T. Liong, J. See, K. Wong, and R C.-W. Phan, "Less is More: Micro-Expression Recognition from Video Using Apex Frame," *Signal Process. Image Commun.*, Vol. 62, pp. 82–92, Mar. 2018, doi: 10.1016/j.image.2017.11.006.

[34] Y.S. Gan *et al.*, "OFF-ApexNet on Micro-Expression Recognition System," *Signal Process. Image Commun.*, Vol. 74, pp. 129–139, May 2019, doi: 10.1016/j.image.2019.02.005.