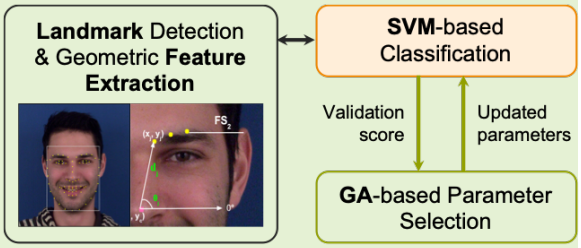
GA-SVM based Facial Emotion Recognition using Facial Geometric Features

Xiao Liu, Xiangyi Cheng, and Kiju Lee, *Member, IEEE*

***Abstract*— This paper presents a facial emotion recognition technique using two newly defined geometric features, landmark curvature and vectorized landmark. These features are extracted from facial landmarks associated with individual components of facial muscle movements. The presented method combines support vector machine (SVM) based classification with a genetic algorithm (GA) for a multi-attribute optimization problem of feature and parameter selection. Experimental evaluations were conducted on the extended Cohn-Kanade dataset (CK+) and the Multimedia Understanding Group (MUG) dataset. For 8-class CK+, 7-class CK+, and 7-class MUG, the validation accuracy was 93.57, 95.58, and 96.29%; and the test accuracy resulted in 95.85, 97.59, and 96.56%, respectively. Overall precision, recall, and F1-score were about 0.97, 0.95, and 0.96. For further evaluation, the presented technique was compared with a convolutional neural network (CNN), one of the widely adopted methods for facial emotion recognition. The presented method showed slightly higher test accuracy than CNN for 8-class CK+ (95.85% (SVM) vs. 95.43% (CNN)) and 7-class CK+ (97.59 vs. 97.34), while the CNN slightly outperformed on the 7-class MUG dataset (96.56 vs. 99.62). Compared to CNN-based approaches, this method employs less complicated models and thus shows potential for real-time machine vision applications in automated systems.**



***Index Terms*— Facial Emotion Recognition, Support Vector Machine, Genetic Algorithm, Facial Geometric Features**

I. Introduction

CAPABILITY of automatically detecting facial expressions can play an important role in various forms of automated technologies designed for direct interaction with the users [1]. Previous studies have demonstrated its practical utilities and potentials in assessing drivers' behavior [2], user engagement in computer games [3], and user preference towards interactive technologies, such as online learning tools [4].

This paper presents an efficient method for facial emotion recognition (FER) via SVM-based classification combined with the GA-based parameter optimization. Two geometric features, landmark curvature (LC) and vectorized landmark (VL), are extracted from the selected facial landmarks and input to the SVM for classification. These landmarks are considered to be closely linked to Action Units (AUs), which refer to facial movements and shape transformations [5], [6]. Traditional Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradient (HOG) can also extract similar information from a face image, but they are more computationally expensive than these geometric features because they use the whole image as an input [7]. To achieve optimal classification performance, the presented method involves a multi-parameter optimization problem: 1) selection of facial landmarks, 2) selection of the regularization parameter of SVM, and 3) relative attributes of the two selected features (VL & LC). A GA-based method is adopted for optimally selecting these parameters.

Algorithm validation employed two publicly available datasets, CK+ [8] and MUG [9]. For the same datasets, a CNN-based FER was also performed for comparison with the presented GA-SVM method. CK+ and MUG used in this study contain well-controlled, full frontal faces. Although there exist more natural datasets (i.e., AffectNet [10], Aff-Wild [11], and Aff-Wild2 [12]), one's facial emotions are highly complicated social behavior which involves significant subjectivity and individual differences in both expression and perception. When a face is not fully visible or shows a subtle expression, its interpretation may vary. In an attempt to focus on algorithm validation and avoid such potential complications, these two traditional datasets with distinct facial emotions are employed.

The presented method involves the following three concurrent processes for achieving optimal landmark selection and SVM parameters, as illustrated in Fig. 1:

- **Landmark detection and geometric feature extraction:** Sixty eight (68) facial landmarks are detected using

X. Liu is with the School of Computing, Informatics, and Decision Systems Engineering at Arizona State University, Tempe, Arizona, USA.
X. Cheng is with Department of Mechanical Engineering at Texas A&M University, College Station, Texas, USA.
K. Lee (Corresponding Author) is with Department of Engineering Technology and Industrial Distribution and Department of Mechanical Engineering at Texas A&M University, College Station, Texas, USA. Email: kiju.lee@tamu.edu
This work was partially done while the authors were at Case Western Reserve University, Cleveland, Ohio, USA.

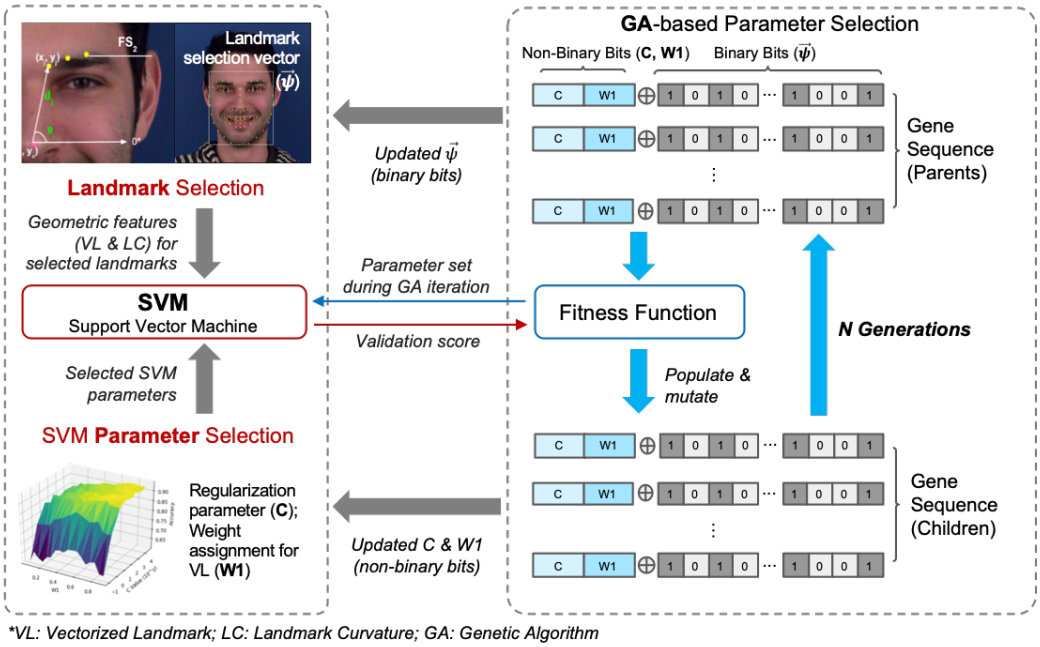


Fig. 1: Overview of the presented FER method: SVM takes two geometric features (VL & LC) extracted from facial landmarks as inputs for classification. To optimize the performance, a GA algorithm is applied to find an optimal subset of these facial landmarks ($\vec{\psi}$) and SVM parameters (C, $W1$). During the GA iterations, interim parameter sets are fed to SVM, which returns a validation score. After N iterations, final results ($\vec{\psi}$, C, $W1$) are used for the input features and parameters of the SVM.

Ensemble of Regression Trees [13] through a machine learning toolkit-Dlib [14]. Landmarks associated with AUs are categorized into facial segments (FSs); and two geometrical features (LC & VL) are extracted from these landmarks.

- **GA-based parameter selection:** This process updates landmark selection ($\vec{\psi}$: landmark selection vector) and two SVM parameters (C: regularization parameter; $W1$ weight for LC) in order to improve FER validation accuracy. Note that $W2 = 1 - W1$, where $W2$ is the weight for VL.
- **SVM-based classification:** SVM takes results (i.e., $\vec{\psi}$, C, & $W1$) from the GA and returns the classification outcome.

All computational processes presented in this paper were carried out in a Ubuntu 18.04 system, with Intel i7-8700 CPU (3.20 GHz) and 16G RAM. Table I defines symbols and acronyms frequently used in this paper.

TABLE I: Nomenclature

Symbols	Definition
FER	Facial emotion recognition
AU	Action unit
FS	Facial segment
LC	Landmark curvature
VL	Vectorized landmark
L_i	Landmark i with its coordinate, (x_i, y_i)
L_c	Geometric center of all landmarks, (x_c, y_c)
FS_i	Facial segment i
$a_0 \cdots a_M$	Coefficients of $f_{\text{FS}i}(x)$
M	Order of the curve fitting
$f_{\text{FS}_i}(x)$	Least squares regression function
κ_i	Curvature at the landmark i
d_i	Euclidean distance of L_i measured from L_c
θ_i	Angle of L_i measured from the horizontal axis.
$\mathbf{M}_{LC}$	Feature matrix for LC
$\mathbf{M}_{VL}$	Feature matrix for VL
W_1	Weight value of the LC feature in the SVM model
W_2	Weight value of the VL feature in the SVM model, such that $W2 = 1 - W1$
C	Regularization parameter of SVM
$\vec{\psi}$	Landmark selection vector
Γ	Gene sequence in the genetic algorithm
p_m	Mutation rate in the genetic algorithm

II. Related Work

Vision-based FER typically involves the following procedures [15]: 1) face detection, 2) feature extraction, and 3) expression classification. The *first* step of FER begins with detecting a face (or multiple faces) within an image. Widely used methods for face detection include, but not limited to, the Viola-Jones algorithm [16], the eigenface techniques [17], and the GA-based methods [18]. The Viola-Jones face detector has advantages of high accuracy and efficiency. Eigenface-based methods reduce the feature dimension while retaining the key features and thus leading to rapid computation. GA-based face detection methods used for both feature selection and parameter tuning also reduce searching costs while increasing detection accuracy [19].

The *second* step is to extract facial features from the detected faces. Handcrafted local image descriptors used for this process include SIFT, Localized Binary Patterns (LBP), LBP variants, geometric descriptors, and Local Directional Patterns (LDP) [20]. There also exist global image descriptors, such as HOG and Gray Level Co-occurrence matrix (GLCM) [21].

The *third* step is to classify the facial features into emotion classes. A convolutional neural network (CNN) extracts such

features directly from the detected faces, and thus performs the second and third steps in a combined manner [22]. Other classifiers, such as SVM [23], Linear Discriminant Analysis (LDA) [24], K-nearest neighbors (KNNs) [25], and Tree-based Learning [26], require extraction of desired features as their inputs. Complicated nonlinear classifiers, such as CNN or SVM with radial basis function kernel (RBF-SVM), likely achieve higher accuracy, but can sometimes result in overfitting and large variance. Linear classifiers, such as linear SVM or LDA, show relatively stable performance with low computational costs compared to the nonlinear methods [27].

Performance on FER is, in fact, highly dependent on the selected dataset as well as the adopted methods for the aforementioned procedures. An algorithm often results in a higher FER accuracy for a specific dataset than that for other datasets. Traditional datasets, such as CK+, MUG, Japanese Female Facial Expression (JAFFE) [28], and MMI Facial Expression Dataset (MMI) [29], include full frontal face images from a single angle view. Each of the images is also cropped, resized, and labeled as one of the distinctive emotions. AffectNet [10], Aff-Wild [11], and Aff-Wild2 [12] are relatively new datasets with more natural face images with larger variations in background, face angles, and sizes as well as races, ages, and genders. Images in these datasets have valence and arousal values, indicating how positive or negative the emotion is and how strong the feeling is. In addition to these values, emotion labels and AUs are also provided in AffectNet and Aff-Wild2. AffectNet includes a small number of cartoon faces and non-face objects.

III. Geometric Features and SVM Classifier

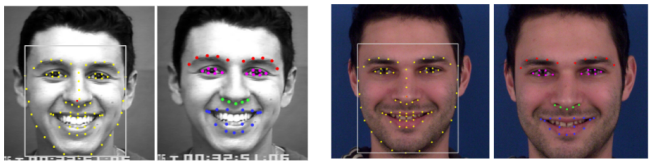
This section presents the procedural method for FER based on a SVM-based classifier.

A. Facial Landmark Detection

As the first step, facial landmarks are automatically detected using the Dlib tool kit. This tool kit enables automated face detection using a single rigid HOG filter trained by Max-Margin Object Detection [14] followed by landmark detection using Ensemble of Regression Trees. The landmark positions are estimated by the cascade regressor derived from a sparse subset of pixel values. This technique produces 68 well-trained landmarks from Labeled Face Parts in the Wild (LFPW) dataset in milliseconds [30]. The detected landmarks include corners of a mouth, eyebrows, eyes, and nose. Fig. 2 shows two example face images with 68 detected landmarks (yellow dots) and the landmarks classified into AU groups shown in different colors.

B. Action Units and Facial Segments

The Facial Action Coding System (FACS) is a system developed for encoding facial movements by distinctive momentary changes [31], [32]. FACS describes facial expressions based on AUs. According to the FACS, 30 AUs (i.e., 12 for upper face & 18 for lower face) are considered anatomically related to the contractions of facial muscles generating facial expressions



(a) (b)

Fig. 2: Detected landmarks (yellow) and AU groups (multicolors); (a) a sample image from CK+ and (b) one from MUG.

[5], [6]. 12 out of 30 AUs can be described with the 68 landmarks which can be automatically detected by using Dlib tool. Instead of attempting to increase the number of detectable AUs – which would inherently increase the processing time, our method focuses on achieving optimal FER performance using these landmarks or even less. As shown in Table. II, these 12 AUs are re-categorized into 16 FSs for more detailed and geometrically well defined facial descriptions.

TABLE II: AUs, facial segments (FSs) with description, and associated landmark numbers for each segment.

AU	FS	Description	Index of Landmarks
AU_1	FS_1	Left inner brow raiser	20-22
	FS_2	Right inner brow raiser	23-25
AU_2	FS_3	Left outer brow raiser	18-20
	FS_4	Right outer brow raiser	25-27
AU_5	FS_5	Left upper lid raiser	37-40
	FS_6	Right upper lid raiser	43-46
AU_7	FS_7	Left lid tightener	37, 40-42
	FS_8	Right lid tightener	43, 46-48
AU_9	FS_9	Nose wrinkler	32-36
AU_{10}	FS_{10}	Upper lid raiser	49-55
AU_{12};AU_{15}	FS_{11}	Left lip corner	49, 61, 68
	FS_{12}	Right lip corner	55, 65, 66
AU_{20};AU_{23}	FS_{13}	Lip stretched/tightener	49, 55-60
AU_{13}	FS_{14}	Left cheek puffer	1-6
	FS_{15}	Right cheek puffer	12-17
AU_{17}	FS_{16}	Chin raiser	7-11

C. Geometric Feature Extraction

SVM takes one or more types of vectorized data as inputs and classifies the data into distinctive classes. For FER, two types of geometric features, LC and VL, are extracted from the facial landmarks. First of all, the pixel location of each landmark is described with respect to the reference frame with its origin located at the upper left corner of the image. It is denoted as $L_i = (x_i, y_i)$, for $i = 1, \cdots, N$, where $N = 68$ in our case. As shown in Table II, each FS is comprised of a unique subset of landmarks, e.g., $\text{FS}_1 = \{L_{20}, L_{21}, L_{22}\}$ and $\text{FS}_2 = \{L_{23}, L_{24}, L_{25}\}$.

For a set of landmarks corresponding to a specific FS_n, least squares regression is applied for curve fitting:

$$f_{\text{FS}_n}(x) = \sum_{k=0}^{M} a_k x^k \tag{1}$$

where linear interpolation is adopted to avoid poor fitting condition. M determines the geometric shape of the curve

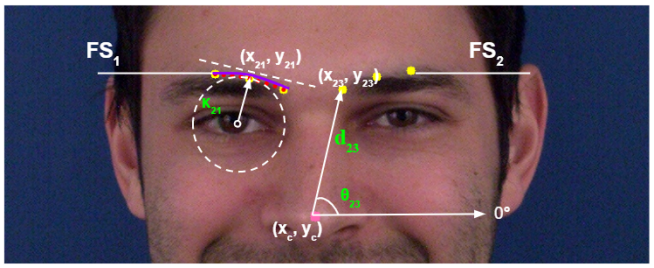


Fig. 3: Feature extraction from given landmarks associated with FS_1 and FS_2: LC feature (κ_{21}) at L_{21}, and VL feature (d_{23}, θ_{23}) at L_{23}.

and thus is expected to affect the training results (See Section IV-B). After curve fitting, the local curvature at each landmark is calculated by

$$\kappa_i = \frac{\mid f''_{\text{FS}_n}(x_i) \mid}{\left\{1 + \left(f'_{\text{FS}_n}(x_i)\right)^2\right\}^{\frac{3}{2}}} \tag{2}$$

where x_i is the x coordinate of L_i that is associated with FS_n, $f'_{\text{FS}_n}(x)$ and $f''_{\text{FS}_n}(x)$ are the first and second derivatives with respect to x, respectively. Fig. 3 shows a purple dashed curve on FS_1. The dashed white circle shown below visualizes how κ_{21} is calculated for $L_{21} \in \text{FS}_1$.

The VL features are simply defined by the polar coordinates of the landmarks described with respect to the geometric center of all landmarks. For N landmarks, the geometric center is defined as $L_c = (x_c, y_c)$, where $x_c = \frac{1}{N}\sum_{j=1}^{N} x_j$ and $y_c = \frac{1}{N}\sum_{j=1}^{N} y_j$. The pair of the Euclidean distance (d_j) and angle measured from the horizontal axis (θ_j) for L_j are calculated as

$$d_j = \sqrt{(x_j - x_c)^2 + (y_j - y_c)^2}; \quad \theta_j = \tan^{-1}\frac{y_j - y_c}{x_j - x_c}.$$

Fig. 3 shows $L_{23} \in \text{FS}_2$ and its VL features, d_{23} and θ_{23}.

D. SVM for Classification

SVM is a powerful tool for both binary and multi-class classification and regression [33]. Although multivariate outliers mining can be used for filtering inconsistent observations from datasets [34], [35], SVM itself is robust against outliers in handling well-crafted datasets, such as CK+ and MUG, and therefore, it is suitable for our FER application. SVM estimates optimal separating hyper-planes among different classes while maximizing the margins between the hyper-planes and closest points of the classes. This SVM optimization problem is typically formulated with a loss function, $\xi(\vec{\omega}; \vec{\alpha}_i, \beta_i)$, as:

$$\min_{\vec{\omega}} = \frac{1}{2}\vec{\omega}^T\vec{\omega} + C\sum_{i=1}^{n}\xi(\vec{\omega}; \vec{\alpha}_i, \beta_i) \tag{3}$$

where $\vec{\alpha}_i$ and β_i represent a training pair containing one group of the training features and its label. L2-loss function, $\xi(\vec{\omega}; \vec{\alpha}_i, \beta_i) = \max(1 - \beta_i(\vec{\omega}^T\vec{\phi}(\vec{\alpha}_i) + \epsilon), 0)^2$, is chosen for multi-classification problems [36]. $\vec{\omega}$ and ϵ determine a linear hyper-plane and $\vec{\phi}$ maps the training feature ($\vec{\alpha}_i$) into a higher dimensional space. $C(> 0)$ is an essential regularization parameter, which controls the trade-off between reducing the error and minimizing the norm of the weights. To determine the attributes of VL and LC features in training, weight factors, W_1 and W_2, are introduced prior to constructing the training pairs with the regularity term, C. The process of tuning these SVM parameters is described in Section IV.

IV. GA-SVM based Parameter Optimization

This section describes GA-based landmark selection and SVM parameter tuning. Except for M in (1) – which simply follows a greedy approach to acquire the optimum, the rest of the parameters, including C, $W1$ (and $W2$), and $\vec{\psi}$, are systematically selected via GA-SVM optimization.

A. Dataset Preparation

CK+ and MUG datasets were used for training and testing the presented FER method. Both datasets are randomly shuffled and divided into 90% and 10%, in which 10% is used for testing. Among the other 90% of the dataset, 10% was used for validation and the rest for training.

CK+ consists of 593 image sequences taken from 123 subjects. Each sequence starts with onset (neutral) and ends with a peak expression, showing the change of one's facial expression from neutral to a certain emotion. From 327 selected labeled sequences, six frames from each sequence were extracted. These six frames include the start frame (neutral) and the last five frames (the labeled emotion). Based on this approach, 1,872 frames, including eight classes of emotion, anger (225), disgust (295), contempt (90), fear (125), happiness (345), neutral (327), sadness (140), and surprise (415) separately, were extracted.

MUG contains 1,462 color image sequences from 86 subjects with different facial expressions. Frames representing the peak expression were extracted and organized. The processed dataset contains 2,658 images in total, including seven classes of emotion, including anger (318), disgust (366), fear (207), happiness (520), neutral (521), sadness (339), and surprise (380). This is a commonly adopted method for data preparation [37], [38].

Since the two datasets have different numbers of emotion classes, i.e. CK+ (8-class) and MUG (7-class), direct comparison between the two may not be desirable. Considering that they both contain seven common emotion classes (except for the contempt group included in CK+), evaluation focused on 7-class FER for both datasets. The results on 8-class FER for the entire CK+ dataset is also presented in this paper for comprehensive evaluation. Fig. 4a shows seven images arbitrarily selected from CK+ (top); the detected faces boxed around with 68 landmarks shown in yellow dots for each (middle); and the cropped and resized face images for better visualization of the detected landmarks (bottom). Fig. 4b shows the same for selected images from MUG.

B. Feature Type & Interpolating Order Determination

Prior to parameter optimization, if using both VL and LC results in better FER performance needs to be evaluated. Since

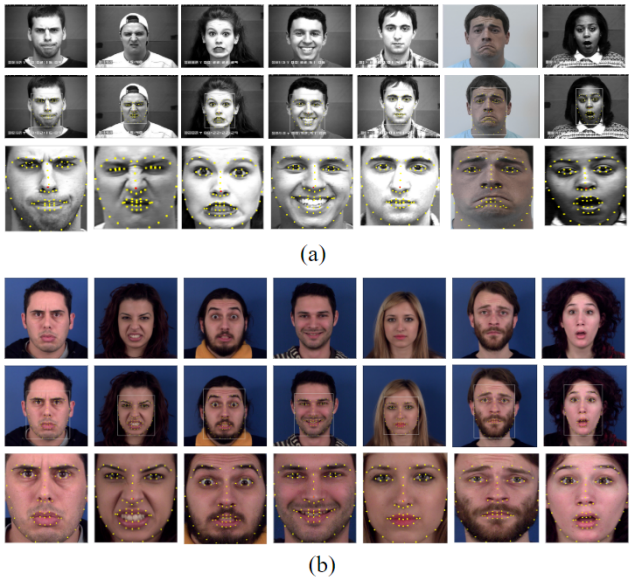


Fig. 4: (a) CK+ examples of seven emotions (i.e., anger, disgust, fear, happy, neutral, sadness, and surprise) (top); landmarks detected (middle); and cropped and re-sized (bottom); (b) MUG examples organized in the same way as (a).

LC features are dependent on M as shown in (1), different values of M are also tested. At this stage, the SVM-related parameters were set to $C = 1$ and $W_1 = W_2 = 0.5$ as default. For both datasets, using both VL and LC with $M = 3$ resulted in the highest accuracy (Table III). When only one of the features was used, the validation accuracy was significantly lower than the case both were used. Using both VL and LC increased the computational time compared to the case using only one of them. It is also found that a value of $M > 3$ lowered the accuracy due to overfitting. The computational time for MUG was relatively high due to its large dataset size compared to CK+, while the same trend in the results was observed.

TABLE III: VL, LC features, and polynomial interpolation order (M) determination

VL	LC	M	CK+ Dataset		MUG Dataset	
			time (sec)	Accuracy	time (sec)	Accuracy
✓	✗	✗	402.33	76.24%	1387.54	82.20%
✗	✓	2	214.17	72.27%	719.11	78.60%
✗	✓	3	213.38	76.57%	719.36	84.09%
✗	✓	4	215.49	74.82%	708.44	77.84%
✓	✓	2	415.48	84.62%	1421.70	88.02%
✓	✓	**3**	**416.36**	**88.01%**	**1399.96**	**88.83%**
✓	✓	4	416.74	85.98%	1406.76	87.48%

C. GA-based Parameter Selection

GA-based parameter optimization is applied for selecting landmarks and SVM parameters. This process determines an optimal subset of the 68 detected landmarks and thus results in using a smaller number of landmarks for SVM. The SVM parameters (W_1, W_2, and C) are also subject to simultaneous optimization. Therefore, this process is a multi-attribute problem involving non-binary SVM-related parameters and a binary facial landmark selection vector ($\vec{\psi}$). A GA-based method is applied to handle this parameter selection problem.

LC and VL features are concatenated into two matrix forms, $\mathbf{M}_{LC}$ and $\mathbf{M}_{VL}$, which are both non-standardized. Knowing that using both LC and VL prominently improves FER performance, the optimization model can be described as

$$\underset{W_1,W_2,C}{\text{argmax}} = \text{score}(C, \mathbf{M}_{VL} \cdot W_1 \oplus \mathbf{M}_{LC} \cdot W_2) \tag{4}$$

where 'score()' returns the validation accuracy from the SVM classifier. '$\oplus$' represents concatenation of two vectors. The search space of C is 10^{λ}, where $\lambda \in \mathbb{R}$. Since $W_1 + W_2 = 1$, only one of them (i.e., W_1, weight for LC) is handled as a variable and its search space is set to $(0, 1)$. Another pivotal parameter subject to optimization is $\vec{\psi}$. It determines which subset of landmarks with associated geometric features will be used for training. Since this process eliminates some of the unnecessary landmarks, computational efficiency can also be improved while achieving higher FER accuracy. $\vec{\psi}$ is a binary bit array, while C and W_1 are non-binary.

The GA-SVM optimization process is described as follows. Consider n initial landmark selection vectors, each denoted as $\vec{\psi}_i$, for $i = 1, \cdots, n$, where all of these vectors are in the solution domain, i.e., binary bit arrays with the size of 68. The gene sequence in GA, denoted as Γ, is a concatenation of the non-binary and binary arrays, such that

$$\Gamma_i^k = (C_i^k \oplus {W_1}_i^k) \oplus \vec{\psi}_i^k \tag{5}$$

representing the i^{th} gene sequence in the k^{th} generation. The GA follows three steps: initialization, selection, and population, where iterations occur on the selection and population phases. In the selection phase, n gene sequences are evaluated by a fitness function with the fitness score calculated by $f(\Gamma)$ in return, which is adopted from (4). At the k^{th} generation, the fitness score of Γ_i^k is represented as:

$$f(\Gamma_i^k) = \text{score}(C_i^k, \langle \vec{\psi}_i^k, \mathbf{M}_{VL} \rangle \cdot {W_1}_i^k \oplus \langle \vec{\psi}_i^k, \mathbf{M}_{LC} \rangle \cdot {W_2}_i^k) \tag{6}$$

where ${W_2}_i^k = 1 - {W_1}_i^k$. The selection process follows the roulette wheel selection rule. After sorting the fitness scores in a non-descending order, probability of choosing individual Γ_i^k for the next generation is calculated as

$$p(\Gamma_i^k) = \frac{f(\Gamma_i^k)}{\sum_{m=1}^{n} f(\Gamma_m^k)} \tag{7}$$

In this case, Γ with a higher fitness score is chosen at a higher probability.

In the population phase, two selected gene sequences go through crossover and mutation. Since Γ has both non-binary and binary bits, single-point crossover for both sides was adopted. Two parent chromosomes transfer their binary bit array ($\vec{\psi}^k$) to the offspring ($\vec{\psi}^{k+1}$), and each parent chromosome offers one of the two non-binary bits (C or W_1) to the offspring chromosome. The mutation rate of $p_m = 4\%$ was used in the population phase in order to prevent from GA selection converging at a local optimum. The population step

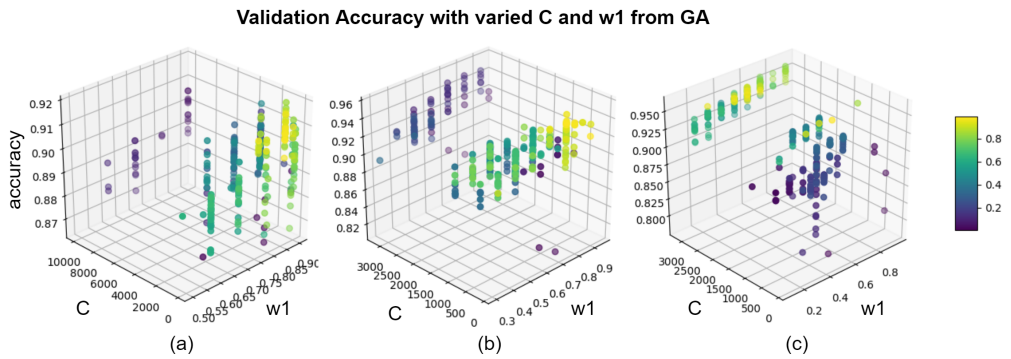


Fig. 5: Feature weight and penalty parameter distribution in GA. (a) CK+ with 8-class; (b) CK+ with 7-class (c) MUG with 7-class.

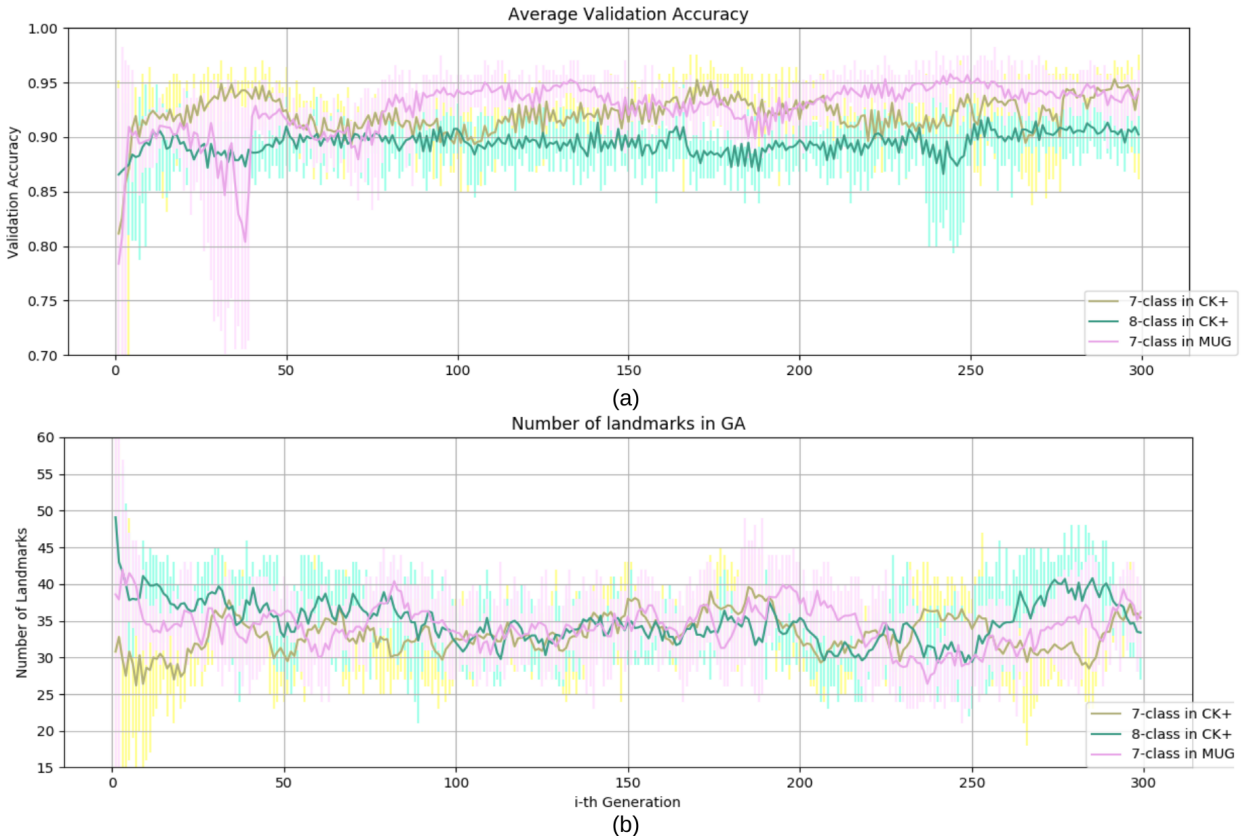


Fig. 6: Results of three experiments conducted on the CK+ and MUG datasets. (a) validation accuracy (with deviation) vs. number of generation; (b) number of landmarks (with deviation) vs. number of generation.

at the k^{th} generation can be represented as:

$$\begin{aligned}
\vec{\psi}_i^{k+1} &= (\vec{\psi}_{(1)}^{k} >< \vec{\psi}_{(2)}^{k}) \cup (\ p_m \circledast \vec{\psi}^{k}) \\
C_i^{k+1} \oplus {W_1}_i^{k+1} &= (C_{(1)}^{k} \oplus {W_1}_{(1)}^{k}) >< (C_{(2)}^{k} \oplus {W_1}_{(2)}^{k}) \\
C_i^{k+1} &= C_i^{k+1} + (\ p_m \circledast C_i^{k+1}) \\
{W_1}_i^{k+1} &= {W_1}_i^{k+1} + (\ p_m \circledast {W_1}_i^{k+1})
\end{aligned} \tag{8}$$

where '><' denotes the mid-point single crossover and '$\circledast$' represents that mutation is applied with p_m. For non-binary mutation, an offset is added to the non-binary bits of Γ after crossover. Therefore, the GA results in $n \times N$ gene sequences after N generations. In our experiment, $N = 300$ and $n = 10$ were selected and search spaces were [10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, $10^{2.5}$, 10^{3}, $10^{3.5}$, 10^{4}] for C, and (0, 1) with step size 0.05 for W_1. A pseudo code of the adopted GA algorithm is shown in Algorithm 1.

D. Results from Parameter Tuning

The distributions of C and W_1 values over 300 generations are shown in Fig. 5. Fig. 5a-b show the parameters selected for CK+ and Fig. 5c shows the results for MUG. The parameters associated with higher validation accuracy (yellow) tended to congregate in certain regions while clear differences in their locations between the two datasets are observed.

The results of the presented GA-SVM method for parameter

Algorithm 1: GA-SVM Optimization

Result: Selection of W_1, C and $\vec{\psi}$;
random initialization in search space;
get genes $\Gamma = (C \oplus W_1) \oplus \vec{\psi}$;
while *error* ***or*** *iteration condition* **do**
  compute and sort fitness scores of Γ;
  perform population based on $p(\Gamma)$;
  update genes Γ;
  if *mutation rate* $> p_m$ **then**
    conduct mutation to Γ;
  end
end

optimization are also presented in terms of the average validation accuracy and the number of selected landmarks over 300 generations. Fig. 6a shows that the average validation accuracy in the first 50 generations was highly unstable. The validation accuracy ranged between 70% and 94% on 7-class MUG (pink); 78% and 94% on 8-class CK+ (green); and 82% and 96% on 7-class CK+ (yellow). The average validation accuracy showed an increasing and converging trend as the iterations proceed. After 300 generations, the average accuracy was around 94% for 7-class MUG, 92% for 8-class CK+, and 93% for 8-class CK+. Since our GA-SVM targets on solving a multi-attribute problem, some fluctuations from generation to generation in the results are expected. Once the SVM-related parameters are set, GA searches through the solution domain of the landmark selection vectors. However, if SVM-related parameters change due to mutation between generations, it can cause a relatively large change in the result.

The number of selected landmarks over 300 generations showed a similar converging behavior. The general tendency of the number of landmarks was decreasing over generations, but the variation in the results remained relatively large in all three cases (Fig. 6b). This can be caused by mutations and/or different landmark selection ($\vec{\psi}$) resulting in the same validation accuracy. For the latter case, all $\vec{\psi}$ resulting in the same validation accuracy may be compared and the one with the least number of landmarks may be selected.

V. Evaluation

This section presents the test accuracy of the optimized GA-SVM method in comparison with CNN, which is one of the widely used methods for FER. Evaluation was performed for the following three cases: 8-class CK+, 7-class CK+, and 7-class MUG.

A. GA-SVM Algorithm Evaluation

An optimal set of parameters was extracted by the best gene sequence (Γ) which resulted in the highest average validation accuracy after 300 iterations. A specific convergence criteria was not employed in this study, because the validation accuracy was used as the fitness score defined in (6). Table IV shows the selected W_1 and C values for each experimental case (8-class CK+, 7-class CK+, and 7-class MUG). The corresponding validation accuracy and test accuracy values are presented in Table IV. The test accuracy values were slightly higher than the validation accuracy in all three cases, indicating that the SVM models are not overfitting by GA-SVM optimization. In addition, the number of landmarks was reduced to 41, 37, and 31, respectively, from the total of 68. Fig. 7 shows the selected landmarks (red) for each case after parameter tuning.

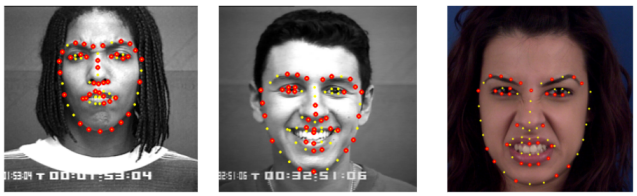
TABLE IV: SVM parameters and their corresponding validation and test accuracy results.

Dataset	W_1	C	Validation Acc.	Test Acc.
CK+ (8)	0.85	$10^{2.5}$	93.57%	**95.85%**
CK+ (7)	0.75	10^{2}	95.58%	**97.59%**
MUG (7)	0.85	10^{3}	96.29%	**96.56%**

TABLE V: Statistical results of the GA-SVM based FER: precision, recall, and F1-score for each emotion class.

Dataset	Emotion	Precision	Recall	F1-score
CK+ (8-class)	Anger	1.00	0.95	0.98
	Disgust	1.00	0.93	0.96
	Fear	1.00	0.92	0.96
	Happy	1.00	1.00	1.00
	Sad	0.93	1.00	0.97
	Surprise	0.97	0.95	0.96
	Neutral	0.84	0.97	0.90
	Contempt	1.00	0.89	0.94
CK+ (7-class)	Anger	1.00	0.96	0.97
	Disgust	1.00	0.97	0.98
	Fear	1.00	1.00	1.00
	Happy	1.00	1.00	1.00
	Sad	1.00	0.86	0.92
	Surprise	0.97	1.00	0.98
	Neutral	0.90	0.88	0.89
MUG (7-class)	Anger	0.97	0.97	0.97
	Disgust	0.97	0.97	0.97
	Fear	0.90	0.90	0.90
	Happy	1.00	0.98	0.99
	Sad	0.97	0.94	0.95
	Surprise	0.92	0.95	0.94
	Neutral	0.98	1.00	0.99

In addition to the validation and test accuracy results, overall classification performance for these three experimental cases were further evaluated in terms of precision, recall, and F1-score to examine if the SVM results in a balanced recurrence for each emotion class [39]. Table V shows the statistical results for these three dataset cases. The average precision of all cases was above 0.97, implying that the percentage of true positives is high and the classification result is balanced.



(a) (b) (c)

Fig. 7: Landmarks selected via the GA-based parameter optimization process: (a) CK+ with 8-class; (b) CK+ with 7-class; (c) MUG with 7-class.

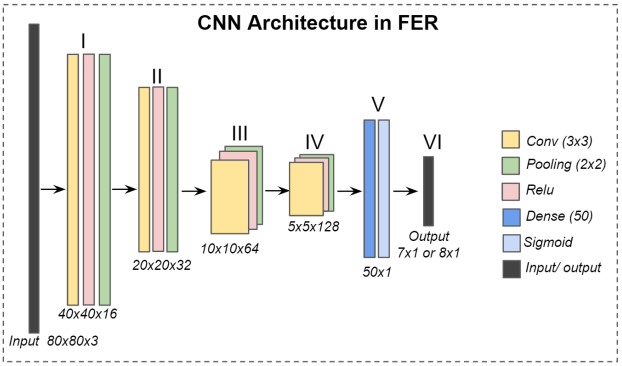


Fig. 8: Proposed CNN architecture for FER: 4 convolutional layers followed by 1 fully connected layer.

The recall value was about 0.95 overall, indicating that the number of false negative classification is low and thus the model is sensitive. The F1-score conveys the balance between the precision and recall values. The average of the F1-score was about 0.96, implying that the classifier is competitive in terms of precision and recall.

B. Comparison with CNN

To compare our results with another commonly used technique, a classic feed-forward CNN was considered. Following the same data preparation procedure adopted for our method, 10% data from each set were randomly selected as the testing set. 90% and 10% of the rest data were selected as the training set and validation set. The architecture of CNN consists of four convolutional layers, one fully connected layer, and one softmax output layer (Fig. 8). The input images are in 80×80 pixels and normalized. The first convolutional layer convolves images with 16 kernels in 3×3 (3-pixel width and 3-pixel height) patch. Its weight tensor has the volume size of $3 \times 3 \times 1$ and 16 output channels to the subsequent convolutional layer. The output features of the first convolution layer then follow a process of convolving training patterns with weight tensors and adding bias. Then, max pooling in size of 2×2 is performed. The image size is reduced to 40×40 before being fed into the next hidden layer. The second convolution layer has 32 kernels for each 3×3 patch. Therefore, the weight tensor has the volume size of $3 \times 3 \times 32$ with 32 output channels. After following the same convolution and max pooling processes, the image size is further reduced to 20×20. The convolutional layer III and IV has 64 and 128 kernels, respectively, for each 3×3 patch.

After four layers of convolution, one fully-connected (local V in Fig. 8) and one softmax output layer (local VI in Fig. 8) are followed. Local V layer contains 50 neurons and local VI layer has 7 or 8 neurons – which is the number of emotion classes. The input of local V is multiplied by a weight tensor, added by a bias, and then applied to a sigmoid function. To avoid overfitting, a dropout is implemented prior to local VI. Cross entropy is used as a loss function and Adam is chosen as an optimizer. Fifty epochs are performed as the result converged within this range. Table VI shows the statistical results using this CNN model. Comparisons between the presented GA-SVM and this CNN method in terms of test accuracy on 8-class CK+, 7-class CK+, and 7-class MUG are shown in Fig. 9. GA-SVM achieved 95.85%, 97.59%, and 96.56% in test accuracy on 8-class CK+, 7-class CK+, and 7-class MUG, respectively. The CNN-based method resulted in 95.43%, 97.34%, and 99.62% on 8-class CK+, 7-class CK+, and 7-class MUG. The presented GA-SVM method resulted in slightly higher test accuracy values for CK+ while CNN outperformed for MUG. Some existing works also report a similar trend in the results [40], [41].

TABLE VI: Statistical report with CNN on precision, recall and F1-score with emotion classes

Dataset	Emotion	Precision	Recall	F1-score
CK+ (8-class)	Anger	0.84	1.00	0.91
	Disgust	0.94	0.97	0.96
	Fear	0.95	1.00	0.97
	Happy	1.00	1.00	1.00
	Sad	1.00	1.00	1.00
	Surprise	0.98	0.98	0.98
	Neutral	0.96	0.76	0.85
	Contempt	0.90	1.00	0.91
CK+ (7-class)	Anger	0.96	1.00	0.98
	Disgust	0.96	0.96	0.96
	Fear	1.00	1.00	1.00
	Happy	1.00	1.00	1.00
	Sad	1.00	0.94	0.97
	Surprise	1.00	0.97	0.99
	Neutral	0.90	0.93	0.91
MUG (7-class)	Anger	1.00	1.00	1.00
	Disgust	1.00	1.00	1.00
	Fear	1.00	0.96	0.98
	Happy	1.00	1.00	1.00
	Sad	1.00	1.00	1.00
	Surprise	0.97	1.00	0.98
	Neutral	1.00	1.00	1.00

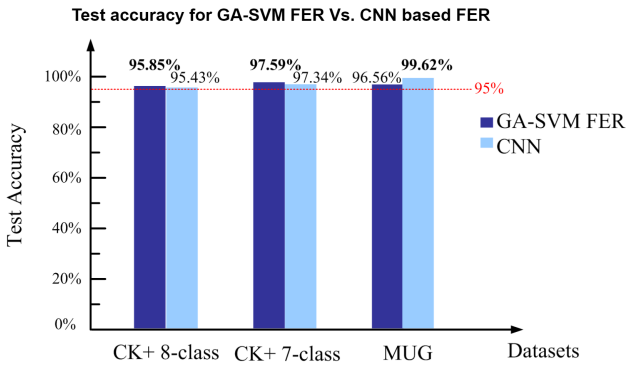


Fig. 9: Test Accuracy Comparison between GA-SVM and CNN on 8-class CK+, 7-class CK+ and 7-class MUG.

C. Comparison with Other Existing Methods

Table VII lists relatively recent work in the field of vision-based FER using traditional datasets, such as CK+, MUG, JAFFE, and MMI, and reports FER performance results in comparison with the presented approach. Existing literature reporting only the cross validation accuracy is not included in this table because the results – regardless if it is K-fold or leave-N-out approach – are likely biased. As shown in Table VII, the Viola-Jones algorithm is commonly used for

face detection [16], [42]–[44]. Some papers do not report the information about the specific face detection algorithm; and there also exist FER algorithms which do not require this process exclusively [40], [41], [45]–[53]. Applications using SVM, LDA, and some network methods require extracted input data fed into the classifiers [41]–[44], [47], [48], [50], [53], while other methods take the images in the datasets directly as the input [16], [40], [46], [49], [51], [52]. The training accuracy for 6-class CK+ varied between 85.42% and 99.33%; 7-class CK+ varied between 95.6% and 96.46%; 8-class CK+ around 95.51%; 6-class MUG varied between 87.65% and 99.3%; 7-class MUG around 99.3%; 6-class JAFFE varied between 84% and 98.8%; 7-class JAFFE around 98.43%; and 6-class MMI varied between 81.5% and 97.55%. One literature also reported 76% accuracy on the combined dataset of MUG, CK+, and JAFFE. The presented method resulted in 95.85% on 8-class CK+ and 97.59% on 7-class CK+, which are higher than these previously reported results. On the other hand, CNN showed a slightly better performance than our method on 7-class MUG, consistent with the result shown in this paper (Section V-B).

Unlike using the traditional datasets, FER using wild datasets involves significant inconsistency and uncertainties. Therefore, deep neural network (DNN) is more broadly adopted for FER with wild datasets because it shows better performance in terms of feature extraction and classification, compared to other traditional machine learning methods, such as SVM and LDA [54]. Also, the features extracted from deep convolutional neural network (DCNN) fed into other traditional machine learning techniques (i.e., SVM, LDA, or KNN) can further improve the FER performance compared to pure CNN in the wild [55]. Most of these algorithms, however, used DNN and its extension for both feature extraction and classification. DCNN with pre-processed images (i.e., face corp, gray scale, and contrast normalization) showed the test accuracy of 76.79% and 77.08% on AffectNet and the Karolinska Directed Emotional Faces (KDEF) dataset [56], respectively [57]. DenseCANet and DenseSANet adopted an attention mechanism based module with DenseNet, which is one of the DCNN architectures [58]. The accuracy of both DenseCANet and DenseSANet was over 60% for the AffectNet dataset.

VI. Conclusion

Two newly defined geometric features extracted from facial landmark are used for SVM-based FER. The presented method adopted a GA-based parameter optimization technique for landmark selection and SVM parameter optimization. The presented GA-SVM method for FER achieved over 95% recognition rate (test accuracy) for the both CK+ and MUG datasets. Unlike many recent approaches, this method does not require superfluous procedural steps nor computationally expensive models. The parameters are tuned autonomously by using a GA to achieve optimal FER performance. Moreover, the method showed consistent and balanced performance over varied machine learning model measurements (Table V), indicating that the proposed model is stable, balanced, and sensitive to actual positive recurrence statistically.

The presented GA-SVM technique can be used for broad machine-vision applications, especially for automated systems. For instance, our preliminary work on real-time FER achieved the processing speed of 6.67 fps using a frontal camera with a 640×480 resolution. This reveals that the presented algorithm is well suitable for interactive applications. Further improvements can be made by adopting different techniques for landmark detection since accurate detection of facial landmark is a critical precondition to successful FER. Moreover, implementing proposed algorithm in 3D faces, considering both frontal and side views of human faces will further broaden the potential utilities of this FER technique.

References

[1] D. R. Faria, M. Vieira, F. C. Faria, and C. Premebida, "Affective facial expressions recognition for human-robot interaction," in *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2017, pp. 805–810.

[2] X. Wang, Y. Liu, F. Wang, J. Wang, L. Liu, and J. Wang, "Feature extraction and dynamic identification of drivers' emotions," *Transportation research part F: traffic psychology and behaviour*, vol. 62, pp. 175–191, 2019.

[3] K. Martin, E. Q. Wang, C. Bain, and M. Worsley, "Computationally augmented ethnography: Emotion tracking and learning in museum games," in *International Conference on Quantitative Ethnography*. Springer, 2019, pp. 141–153.

[4] K. Bahreini, R. Nadolski, and W. Westera, "Towards multimodal emotion recognition in e-learning environments," *Interactive Learning Environments*, vol. 24, no. 3, pp. 590–605, 2016.

[5] Y.-I. Tian, T. Kanade, and J. F. Cohn, "Recognizing action units for facial expression analysis," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 2, pp. 97–115, 2001.

[6] R. Ekman, *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997.

[7] H. Fleyeh and J. Roch, *Benchmark evaluation of HOG descriptors as features for classification of traffic signs*. Högskolan Dalarna, 2013.

[8] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE, 2010, pp. 94–101.

[9] N. Aifanti, C. Papachristou, and A. Delopoulos, "The mug facial expression database," in *11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS 10*. IEEE, 2010, pp. 1–4.

[10] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild," *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 18–31, 2017.

[11] D. Kollias, P. Tzirakis, M. A. Nicolaou, A. Papaioannou, G. Zhao, B. Schuller, I. Kotsia, and S. Zafeiriou, "Deep affect prediction in-the-wild: Aff-wild database and challenge, deep architectures, and beyond," *International Journal of Computer Vision*, vol. 127, no. 6-7, pp. 907–929, 2019.

[12] D. Kollias and S. Zafeiriou, "Expression, affect, action unit recognition: Aff-wild2, multi-task learning and arcface," *arXiv preprint arXiv:1910.04855*, 2019.

[13] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1867–1874.

[14] D. E. King, "Max-margin object detection," *arXiv preprint arXiv:1502.00046*, 2015.

[15] N. Sheikh and M. K. Singhal, "A review of human facial expressions recognition methodologies," 2019.

[16] B. R. Ilyas, B. Mohammed, M. Khaled, A. T. Ahmed, and A. Ihsen, "Facial expression recognition based on dwt feature for deep cnn," in *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2019, pp. 344–348.

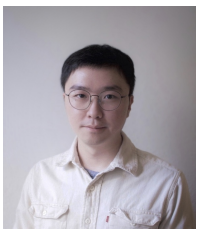
[17] N. K. A. Wirdiani, T. Lattifia, I. K. Supadma, B. J. K. Mahar, D. A. N. Taradhita, and A. Fahmi, "Real-time face recognition with eigenface method," 2019.

TABLE VII: Comparison of Recent FER Approaches based on Frontal Face Data

Dataset	#. Class	Face Detection	Feature Exaction	Classifier	Accuracy (%)	Ref.
MMI CK+	6	Viola-Jones algorithm	Geometric feature and LBP texture feature fused by autoencoder	SVM	93.19 95.01	[44]
MMI CK+	6	Viola-Jones algorithm	Geometric feature and LBP texture feature fused by autoencoder	Self-organzing map (SOM)	97.55 98.95	[44]
FER-2013 [59]	7	N/A	CNN parameters	SVM	97.85	[47]
RaFD [60]	6	Viola-Jones algorithm	Pyramid HOG (PHOG)	GA-LDA	97.33	[42]
CK+	8	N/A	Difference center-symmetric local directional pattern (DCS-LDP)	SVM	95.51	[48]
CK+	6	N/A	Haar Wavelet Transform and Gabor wavelets	SVM	98	[50]
JAFFE CK+ MUG	6	Viola-Jones algorithm	Geometric facial landmarks	CNN	84.00 85.42 89.19	[43]
CK+ MUG JAFFE MMI	6	N/A	Normalized distance signature	Multilayer Perceptron (MLP)	96.3 96.7 94 81.5	[45]
CK+ MUG MMI JAFFE	6	N/A	Grids, triangles on face images and texture features found by salient landmarks	Deep belif network (DBN)	98 99.1 93.1 98.8	[53]
CK+ MUG MMI JAFFE	6	N/A	Distance and texture characteristics among the landmark points	Radial basis function (RBF) network	98.6 99.3 90.9 97.6	[41]
Combination of MUG, CK+ and JAFFE	7	Used, but not specified	Face alignment by landmarks with CNN		76	[46]
JAFFE CK+	7	Viola-Jones algorithm	CNN with histogram equalization (HE) and discrete wavelet transform (DWT)		98.43 96.46	[16]
MUG CK+	7	N/A	CNN-base and hybrid inherited lightweight network HiNet		99.3 95.7	[40]
RaFD CK+ MUG	6	N/A	CNN with 4 convolutional layers, 3 max-pooling, 1 fully-connected, and 1 softmax output layer.		93.33 99.33 87.65	[51]
CK+ FER-2013 FERG [61] JAFFE	6	N/A	End-to-end deep learning framework based on attentional convolutional network		98.0 70.02 99.3 92.8	[52]
FER-2013	7	N/A	CNN		88.9	[49]
CK+ **CK+** **MUG**	**8** **7** **7**	**Viola-Jones algorithm**	**Presented geometric features with GA-SVM optimization**		**95.85** **97.59** **96.56**	-

[18] H. Zhi and S. Liu, "Face recognition based on genetic algorithm," *Journal of Visual Communication and Image Representation*, vol. 58, pp. 495–502, 2019.
[19] Z. Tao, L. Huiling, W. Wenwen, and Y. Xia, "Ga-svm based feature selection and parameter optimization in hospitalization expense modeling," *Applied Soft Computing*, vol. 75, pp. 323–332, 2019.
[20] W. Hayale, P. Negi, and M. Mahoor, "Facial expression recognition using deep siamese neural networks with a supervised loss function," in *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*. IEEE, 2019, pp. 1–7.
[21] K. Chengeta and S. Viriri, "A review of local, holistic and deep learning approaches in facial expressions recognition," in *2019 Conference on Information Communications Technology and Society (ICTAS)*. IEEE, 2019, pp. 1–7.
[22] J. J. Sanchez-Castro, J. C. Rodríguez-Quiñonez, L. R. Ramírez-Hernández, G. Galaviz, D. Hernández-Balbuena, G. Trujillo-Hernández, W. Flores-Fuentes, P. Mercorelli, W. Hernández-Perdomo, O. Sergiyenko *et al.*, "A lean convolutional neural network for vehicle classification," in *2020 IEEE 29th International Symposium on Industrial Electronics (ISIE)*. IEEE, 2020, pp. 1365–1369.
[23] W. Flores-Fuentes, M. Rivas-Lopez, O. Sergiyenko, F. F. Gonzalez-Navarro, J. Rivera-Castillo, D. Hernandez-Balbuena, and J. C. Rodríguez-Quiñonez, "Combined application of power spectrum centroid and support vector machines for measurement improvement in optical scanning systems," *Signal Processing*, vol. 98, pp. 37–51, 2014.
[24] Y. Wei, S. Jia, Q. Wang, and H. Yu, "The application of lda model on user profile," in *2017 3rd International Conference on Economics, Social Science, Arts, Education and Management Engineering (ESSAEME 2017)*. Atlantis Press, 2017.
[25] O. Real-Moreno, M. J. Castro-Toscano, J. C. Rodrízuez-Ouiñonez, D. Hernández-Balbuena, W. Flores-Fuentes, and M. Rivas-Lopez, "Implementing k-nearest neighbor algorithm on scanning aperture for accuracy improvement," in *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2018, pp. 3182–3186.
[26] B. Berthon, C. Marshall, M. Evans, and E. Spezi, "Atlaas: an automatic decision tree-based learning algorithm for advanced image segmentation in positron emission tomography," *Physics in Medicine & Biology*, vol. 61, no. 13, p. 4855, 2016.
[27] K. Chengeta and S. Viriri, "A review of local, holistic and deep learning approaches in facial expressions recognition," in *2019 Conference on Information Communications Technology and Society (ICTAS)*. IEEE, 2019, pp. 1–7.
[28] M. J. Lyons, S. Akamatsu, M. Kamachi, J. Gyoba, and J. Budynek, "The japanese female facial expression (jaffe) database," in *Proceedings of third international conference on automatic face and gesture recognition*, 1998, pp. 14–16.
[29] M. Valstar and M. Pantic, "Induced disgust, happiness and surprise: an addition to the mmi facial expression database," in *Proc. 3rd Intern. Workshop on EMOTION (satellite of LREC): Corpora for Research on Emotion and Affect*. Paris, France, 2010, p. 65.
[30] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 12, pp. 2930–2940, 2013.
[31] E. Friesen and P. Ekman, "Facial action coding system: a technique for the measurement of facial movement," *Palo Alto*, vol. 3, 1978.
[32] J. Hamm, C. G. Kohler, R. C. Gur, and R. Verma, "Automated facial action coding system for dynamic analysis of facial expressions in neuropsychiatric disorders," *Journal of neuroscience methods*, vol. 200, no. 2, pp. 237–256, 2011.
[33] Y.-W. Chang and C.-J. Lin, "Feature ranking using linear svm," in *Causation and Prediction Challenge*, 2008, pp. 53–64.

[34] W. Flores-Fuentes, O. Sergiyenko, F. F. Gonzalez-Navarro, M. Rivas-López, J. C. Rodríguez-Quiñonez, D. Hernández-Balbuena, V. Tyrsa, and L. Lindner, "Multivariate outlier mining and regression feedback for 3d measurement improvement in opto-mechanical system," *Optical and Quantum Electronics*, vol. 48, no. 8, p. 403, 2016.
[35] W. Flores-Fuentes, D. Hernandez-Balbuena, J. C. Rodriguez-Quiñonez, D. Olivas-Ugalde, F. F. González-Navarro, O. Sergiyenko, M. Rivas-López, F. N. Murrieta-Rico, and V. M. Kartashov, "Outlier mining of a vision sensing databasefor svm regression improvement," in *IECON 2015-41st Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2015, pp. 000 208–000 213.
[36] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *Journal of machine learning research*, vol. 9, no. Aug, pp. 1871–1874, 2008.
[37] R. Melaugh, N. Siddique, S. Coleman, and P. Yogarajah, "Facial expression recognition on partial facial sections," in *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*. IEEE, 2019, pp. 193–197.
[38] S. M. González-Lozoya, J. de la Calleja, L. Pellegrin, H. J. Escalante, M. A. Medina, and A. Benitez-Ruiz, "Recognition of facial expressions based on cnn features," *Multimedia Tools and Applications*, pp. 1–21, 2020.
[39] R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval*. ACM press New York, 1999, vol. 463.
[40] M. Verma, S. K. Vipparthi, and G. Singh, "Hinet: Hybrid inherited feature learning network for facial expression recognition," *IEEE Letters of the Computer Society*, vol. 2, no. 4, pp. 36–39, 2019.
[41] A. Barman and P. Dutta, "Facial expression recognition using distance and texture signature relevant features," *Applied Soft Computing*, vol. 77, pp. 88–105, 2019.
[42] H. Boubenna and D. Lee, "Feature selection for facial emotion recognition based on genetic algorithm," in *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*. IEEE, 2016, pp. 511–517.
[43] N. Gopalan, S. Bellamkonda, and V. S. Chaitanya, "Facial expression recognition using geometric landmark points and convolutional neural networks," in *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*. IEEE, 2018, pp. 1149–1153.
[44] A. Majumder, L. Behera, and V. K. Subramanian, "Automatic facial expression recognition system using deep network-based data fusion," *IEEE transactions on cybernetics*, vol. 48, no. 1, pp. 103–114, 2016.
[45] A. Barman and P. Dutta, "Facial expression recognition using distance signature feature," in *Advanced Computational and Communication Paradigms*. Springer, 2018, pp. 155–163.
[46] S. Xu, Y. Cheng, Q. Lin, and J. Allebach, "Emotion recognition using convolutional neural networks," *Electronic Imaging*, vol. 2019, no. 8, pp. 402–1, 2019.
[47] X. Liu and K. Lee, "Optimized facial emotion recognition technique for assessing user experience," in *2018 IEEE Games, Entertainment, Media Conference (GEM)*. IEEE, 2018, pp. 1–9.
[48] Y. Jiao, X. Jia, and J. Zhao, "Facial expression recognition method based on difference center-symmetric local directional pattern," in *2019 International Conference on Computer, Network, Communication and Information Systems (CNCI 2019)*. Atlantis Press, 2019.
[49] N. Meeki, A. Amine, M. A. Boudia, and R. M. Hamou, "Deep learning for non verbal sentiment analysis: Facial emotional expressions," EasyChair, Tech. Rep., 2020.
[50] C. Reddy, U. Reddy, and K. Kishore, "Facial emotion recognition using nlpca and svm." *Traitement du Signal*, vol. 36, no. 1, pp. 13–22, 2019.
[51] A. Fathallah, L. Abdi, and A. Douik, "Facial expression recognition via deep learning," in *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*. IEEE, 2017, pp. 745–750.
[52] S. Minaee and A. Abdolrashidi, "Deep-emotion: Facial expression recognition using attentional convolutional network," *arXiv preprint arXiv:1902.01019*, 2019.
[53] A. Barman and P. Dutta, "Influence of shape and texture features on facial expression recognition," *IET Image Processing*, vol. 13, no. 8, pp. 1349–1363, 2019.
[54] S. Kim and H. Kim, "Deep explanation model for facial expression recognition through facial action coding unit," in *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*. IEEE, 2019, pp. 1–4.
[55] Z. Fei, E. Yang, D. D.-U. Li, S. Butler, W. Ijomah, X. Li, and H. Zhou, "Deep convolution network based emotion analysis towards mental health care," *Neurocomputing*, 2020.
[56] D. Lundqvist, A. Flykt, and A. Öhman, "The karolinska directed emotional faces (kdef)," *CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet*, vol. 91, no. 630, pp. 2–2, 1998.
[57] A. D. Rasamoelina, F. Adjailia, and P. SinČàk, "Deep convolutional neural network for robust facial emotion recognition," in *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE, 2019, pp. 1–6.
[58] C. Wang, K. Lu, J. Xue, and Y. Yan, "Dense attention network for facial expression recognition in the wild," in *Proceedings of the ACM Multimedia Asia on ZZZ*, 2019, pp. 1–6.
[59] P.-L. Carrier, A. Courville, I. J. Goodfellow, M. Mirza, and Y. Bengio, "Fer-2013 face database," *Universit de Montral*, 2013.
[60] J.-Y. Son, J.-H. Lee, J.-Y. Kim, J.-H. Park, and Y.-H. Lee, "Rafd: Resource-aware fault diagnosis system for home environment with smart devices," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 4, pp. 1185–1193, 2012.
[61] D. Aneja, A. Colburn, G. Faigin, L. Shapiro, and B. Mones, "Modeling stylized character expressions via deep learning," in *Asian conference on computer vision*. Springer, 2016, pp. 136–153.



Xiao Liu received his B.S. degree in Mechanical Engineering from Southwest Jiaotong University, Chengdu, China, in 2015, and the M.S. degree in Mechanical Engineering from Case Western Reserve University, Cleveland, Ohio, USA, in 2019, with his thesis on the topic of human-robot interaction under the supervision of Prof. Kiju Lee. He is pursuing his Ph.D. in Computer Science with school of Computing, Informatics, and Decision Systems Engineering at Arizona State University, Tempe, Arizona, USA. His research interests are computer vision, robot learning, and human-robot interaction.

Xiangyi Cheng received her B.S. degree in Mechanical Engineering from China University of Mining and Technology - Beijing, Beijing, China, in 2015. She started her graduate study at Case Western Reserve University, Cleveland, Ohio, in 2015 and transferred to Texas A&M University, College Station, Texas, USA, in 2019. She is currently pursuing her Ph.D. in Mechanical Engineering at Texas A&M University. Her research interests involve serious games for cognitive assessment, computer vision, and human-technology interaction. Her doctoral research focuses on technology-enabled cognitive assessment for older adults.

Kiju Lee received her B.S.E. degree in Electrical and Electronics Engineering from Chung-Ang University, Seoul, Korea, in 2002, and the M.S.E. and Ph.D. in Mechanical Engineering from Johns Hopkins University, Baltimore, Maryland, USA, in 2006 and 2008, respectively. From 2008 to 2019, she was with the faculty of the Department of Mechanical and Aerospace Engineering at Case Western Reserve University, Cleveland, Ohio, USA. Since 2019, she holds a joint faculty position between the Department of Engineering Technology and Industrial Distribution and the Department of Mechanical Engineering at Texas A&M University, College Station, Texas, USA. Kiju Lee is the director of the Adaptive Robotics and Technology (ART) Lab through which she runs various research projects focusing on swarm robotics, human-robot interaction, robotic mechanism design, and tangible games.