A Deep Learning System for Recognizing Facial Expression in Real-Time

YU MIAO and HAIWEI DONG, University of Ottawa
JIHAD MOHAMAD AL JAAM, Qatar University
ABDULMOTALEB EL SADDIK, University of Ottawa

This article presents an image-based real-time facial expression recognition system that is able to recognize the facial expressions of several subjects on a webcam at the same time. Our proposed methodology combines a supervised transfer learning strategy and a joint supervision method with center loss, which is crucial for facial tasks. A newly proposed Convolutional Neural Network (CNN) model, MobileNet, which has both accuracy and speed, is deployed in both offline and in a real-time framework that enables fast and accurate real-time output. Evaluations towards two publicly available datasets, JAFFE and CK+, are carried out respectively. The JAFFE dataset reaches an accuracy of 95.24%, while an accuracy of 96.92% is achieved on the 6-class CK+ dataset, which contains only the last frames of image sequences. At last, the average run-time cost for the recognition of the real-time implementation is around 3.57ms/frame on a NVIDIA Quadro K4200 GPU.

CCS Concepts: • **Computing methodologies → Computer vision tasks**; *Transfer learning*; *Real-time simulation*;

Additional Key Words and Phrases: Facial expression recognition, deep learning networks

ACM Reference format:
Yu Miao, Haiwei Dong, Jihad Mohamad Al Jaam, and Abdulmotaleb El Saddik. 2019. A Deep Learning System for Recognizing Facial Expression in Real-Time. *ACM Trans. Multimedia Comput. Commun. Appl.* 15, 2, Article 33 (May 2019), 20 pages.
https://doi.org/10.1145/3311747

1 INTRODUCTION

Children with autism often have difficulty recognizing the emotional state of people around them. For example, it is difficult for them to distinguish a happy face from a fearful face. Robot-assisted treatment for autism often solves the problem. At the introduction stage of the therapy, the robot is introduced to the child and free-plays with him. At the teaching stage, cards with different facial emotions are presented to the child by the human therapist, while the robot demonstrates the corresponding emotion to teach the child how to recognize joy, sadness, fear, and so on. At last, at

This work was made possible by NPRP grant (10-0205-170346) from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

Authors' addresses: Y. Miao and H. Dong, University of Ottawa, 800 King Edward Avenue, Ottawa, ON K1N 6N5, Canada; emails: {ymiao036, hdong}@uottawa.ca; J. Mohamad Al Jaam, Qatar University, Ibn Khaldoon Hall, Doha, Qatar; email: jaam@qu.edu.qa; A. El Saddik, University of Ottawa, 800 King Edward Avenue, Ottawa, ON K1N 6N5, Canada; email: elsaddik@uottawa.ca.

1551-6857/2019/05-ART33 $15.00
https://doi.org/10.1145/3311747

the practice stage, the child uses the face cards from the teaching stage to indicate what emotion he thinks the robot is displaying. This therapy is said to be currently the most effective one towards treating autism [1]. Inspired by this, we decide to design a system that can accomplish estimating human facial expressions in real time that may help assess the emotional state of autistic children.

Emotional voice, gesture, facial expressions, and so on, constitute the factors of human emotions [2]. Facial expressions, among these factors mentioned above, play the most important role in affect analysis [3]. Mehrabian [4] worked out a formula that considered the effect of the spoken message as a whole; facial expressions of the speaker contribute 55 percent, while the vocal part (e.g., vocal intonation) and the verbal part (i.e., spoken words) contribute only about 38 percent and 7 percent, respectively. This condition emphasizes that facial expression is the most significant part of non-verbal communication and the primary modality used to convey emotions.

Automatic facial expression recognition (FER) systems generally receive two kinds of expected input (still images or a sequence of frames) and output one of the seven basic universal emotions (i.e., angry, disgust, fear, happiness, sadness, surprise, and neutral) that were classified by Ekman [5] in 1975. FER has seen advancement in research and development in the past decade due to advances in machine learning, image processing, human cognition [6], areas such as face detection, tracking, and recognition, as well as the recent availability of relatively cheap computational power.

Most of conventional approaches for FER utilize machine-learning classifiers (e.g., neural networks (NN) [7], linear discriminant analysis (LDA) [8], support vector machines (SVM) [9], etc.) to classify the extracted features (e.g., Gabor wavelet coefficients [7, 8], histograms of local binary pattern (LBP) [10], histograms of oriented gradients (HOG) [11, 12], etc.). Because of the need for computational power and programming efforts, few of these can fit the real-time requirement. Recently, the approaches of deep learning have flourished due to inexpensive computational power, and one example, called the convolutional neural network (CNN), has obtained excellent state-of-the-art results in the field of computer vision (e.g., image classification, face recognition, object detection). Also, CNNs have been successfully applied to FER [13, 14] and have shown better results than many conventional methods due to its efficiency in feature learning and representations.

Among numerous CNN models, MobileNet [15], which was proposed in 2017, is a lightweight deep neural network produced by Google. Following a year's development, it has become a basic and popular network structure similar to GoogleNet and ResNet. Its starting point is to construct a lean, lightweight network based on a streamlined architecture that has much smaller size and lower computation complexity than CNN benchmarks (e.g., AlexNet [16], VGG [17]), and it can be used efficiently on mobile and embedded devices that offer limited performance.

Although FER under controlled conditions is already mature and no longer a substantial problem, robust CNN-based FER still remains an unsolved problem in real-life scenarios in spite of CNN's superiority. CNNs achieve high accuracy in face-recognition tasks with high efficiency, requiring the size of labeled training data in the millions as emphasized in Reference [18]. However, the total number of samples in most datasets for FEA is quite small (e.g., hundreds for References [19, 20] and thousands for References [21, 22], which is far from sufficient for training CNN). Another problem is that the evaluation of the state-of-the-art work on specific datasets in the literature are not applied consistently (e.g., in terms of the number of classes of emotions to be evaluated), which sometimes leads to falsely high accuracies.

Data size is one of the most significant factors affecting the performances of CNN [23]. To compensate for the limitation of small datasets, transfer learning based on ImageNet, which is the largest high-quality visual recognition database at present, has become a standard benchmark and been widely applied to large-scale visual tasks such as image classification and object recognition. Because of its intraclass variation, this database can help such data-driven systems improve their performances even with different domain problems, e.g., the medical image domain [24]. Inspired by the work of References [25, 26], we decide to follow the "two-stage" supervised transfer learning based on ImageNet and an auxiliary dataset FER-2013.

For recognition tasks in the computer vision field, particularly for FER in this case, the ideal learned features to be selected should not only contain interclass separation (separable), but also intraclass compactness (discriminatory). The softmax loss has been widely used for optimization of CNNs for its key characteristic of nonnegativity and eliminates the possibility of positive and negative value cancellation. However, the softmax loss only tends to separate the deep features. Thus, Wen et al. [27] in 2016 proposed a new supervision signal called center loss to enhance this discriminatory power that is very crucial for facial tasks. The promising results obtained in their work enlighten us to the idea of using center loss in the FER scenario. Following their work, we develop an applied algorithm that adds center loss to the loss function together with the softmax loss and apply it to the training of our FER model, enhancing the discriminatory power of the proposed system.

In our work, the new CNN model MobileNet [15] is applied to accomplish this FER task, providing a basis for real time. To address the problem of insufficient size of those small facial expression datasets, a two-stage fine-tuning strategy is used in the training process of CNN. In addition, a new supervision signal, center loss, is leveraged jointly with the softmax loss function in optimization for interclass dispersion and intraclass compactness [27]. The proposed system was evaluated on the CK+ and JAFFE datasets and achieved competitive results. In summary, the main contributions of this article are as follows:

(1) A two-stage fine-tuning strategy is used in this work to solve the problem of insufficient data size of facial expression datasets.
(2) An extra center loss is added to the loss function under joint supervision with the softmax loss for enhancing the discriminatory power of the proposed system.
(3) Implementation of a real-time FER system that achieves a smaller run-time cost compared to the literature.

The remainder of this article is organized as follows: In Section 2, a brief review of conventional methods and recent work that is deep-learning-related for facial expression analysis is presented. The details for the methodology used in the training process of CNN are demonstrated in Section 3. Then the datasets used in this work are presented in Section 4. Experiments and evaluations on two datasets showing the effectiveness of the proposed methodology and results are provided in Section 5. In Section 6, a discussion towards the results are elaborated. Section 7 summarizes the merits of this work and mentions the limitations and future work.

2 RELATED WORK

Facial expression analysis (FEA) as a research field was established by Darwin in 1872 [28]. Since then, FER has been an active research topic across a variety of disciplines, such as biology [28], psychology [29], and computer vision [24]. Especially in computer vision, for its impact and prominent potentiality, automatic FER has been growing in an extensive range of applications, e.g., biometric identification, surveillance, and security [30], driver state surveillance [6], and the entertainment industry and virtual reality.

Conventional methods related to facial feature extraction and representation are categorized as appearance-based [11] and geometric-based (shape-based) [31]. Most of the state-of-art work includes the so-called hand-crafted features (e.g., Gabor wavelet coefficients, histograms of LBP, HOG, etc.). These extracted hand-crafted features are then fed into the empirical classifiers [7, 8] to execute the classification so both the computational power and programming efforts are needed for consideration.

One promising advantage of CNN-based method is that it combines the three steps of automatic FEA (face acquisition, facial feature extraction and representation, and facial expression classification) into one single step. The hierarchical structure of CNN—which is a design of local-to-global feature learning [32] composed of diverse convolution, subsampling (e.g., max-pooling, average-pooling), and fully connected layers—contains a strong feature representation capacity and can be a powerful tool in FER [13, 14]. The frameworks of these works include image preprocessing, CNN architecture, training schemes, and evaluation configuration.

Burkert et al. [33] proposed a CNN architecture that does not depend on the hand-crafted features. Four parts comprise this architecture, and the images are first preprocessed automatically through a convolutional layer. The images are then downsampled by the pooling layer in the second part. The next block, called the FeatEx, serves as the fundamental structure in this architecture, which was inspired by GoogleNet. Finally, the extracted features after two concatenated FeatEx blocks are fed into a fully connected layer to perform the classification. The deep features of different layers are visualized to show its validity, and evaluations are conducted with two standard datasets, namely, MMI and CK+. Their experiment on the CK+ dataset, which evaluated seven classes (*angry, disgust, fear, happiness, sadness, surprise, and contempt*), achieved a recognition rate of 99.6%.

Tang et al. [34] presented a framework that followed the convolution routines of Alex Krizhevsky [16] but replaced the last softmax layer with a linear SVM and optimized the loss from the L2-SVM (DLSVM) instead of the cross-entropy loss. The input images were first preprocessed by subtracting the mean value and setting the norm to 100 before being fed into the network for training. Superior results were obtained by using this DLSVM with softmax by evaluating two standard datasets, MNIST and CIFAR-10, as well as one of the largest recent FER datasets: FER-2013 [35]. The performance of the proposed framework won first place in the FER challenge of the ICML 2013 Workshop hosted on Kaggle, with an accuracy of 69.4% for the public validation set and 71.2% for the private test set.

To engage in the image-based static facial expression recognition sub-challenge of the EmotiW2015, Kim et al. [25] constructed a hierarchical committee of multiple CNNs with an ensemble method based on exponentially weighted decision fusion. First, face registration was achieved by four different pipelines, where the VJ detector and the Zhu-Ramanan (Z-R) model were used for face detection, andIntraFace was used for the facial landmarks extraction; finally, the best method was selected by 2D conventional alignment. Following Tang's CNN architecture [34], these approaches trained several CNN candidates varying in the size of kernels (receptive fields) and the number of neurons in the fully connected layer; finally, a decision was made by the ensemble method. In addition, to improve the performance on the SFEW2.0 dataset, a transfer-learning method that used external datasets—namely, FER-2013, the Toronto Face Database (TFD), and the GENKI-4K database were applied to the training process. This configuration defeated other candidates in this challenge by achieving a test accuracy of 61.6% on the SFEW2.0 dataset; the baseline for this dataset was 39.1%.

As an additional candidate in the contest of EmotiW2015, Ng et al. [26] emphasized the importance of using a transfer-learning method, which was a supervised two-stage process. Two representative CNN architectures (AlexNet and VGG-CNN-M-2048) were selected for their trade-offs regarding accuracy and speed. By first fine-tuning the FER-2013 dataset based on the pretrained models from ImageNet and then fine-tuning the SFEW2.0 dataset (the target dataset), the authors reached an accuracy of 55.6% on the test set, which ranked third place in the contest.

Inspired by the architecture of AlexNet and GoogleNet [36], Mollahosseini et al. [37] proposed their own CNN architecture in 2016, which consisted of two conventional CNN modules (one of which included a convolutional layer followed by a max-pooling layer), four *Inception* modules,

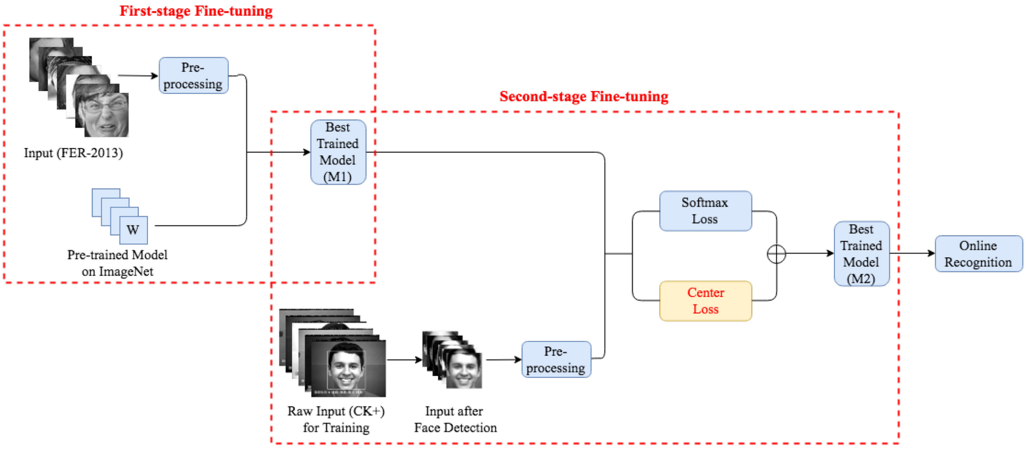


Fig. 1. Overview of the proposed framework. A first stage of fine-tuning is applied to FER-2013 based on the pretrained model from ImageNet. After obtaining the best-trained model, a second stage of fine-tuning is then performed on a specific dataset. An additional center loss is used as a part of the supervision signal together with the softmax loss during optimization. Finally, the best-trained model is selected for online classification. This example involves the CK+ dataset.

and two fully connected layers, having only 25M operations (compared to 100M in AlexNet). Face registration was performed to improve the performance of FER by using the bidirectional warping of the active appearance model (AAM) and a supervised method called IntraFace that adopted the SIFT features to extract 49 facial landmarks. Both subject-independent and cross-database experiments were carried out on seven public standard datasets (MultiPIE, MMI, CK+, DISFA, FERA, SFEW, and FER-2013), and six specific classes (angry, disgust, fear, happiness, sadness, surprise, excluding the neutral and contempt classes) were evaluated on the CK+ dataset.

3 METHOD

In this section, an overview of the proposed methodology is presented that mainly contains two parts, as illustrated in Figure 1. After the image data are prepared, the CNN training scheme—which consists of a transfer-learning strategy and an optimization method combining a newly proposed supervision signal with softmax loss—is applied to a recently proposed CNN model. Finally, the best-trained model is used to perform the online classification.

3.1 Preprocessing

Preprocessing of expression images can vary depending on factors such as the performance of the acquisition equipment or changes in illustration conditions. It is necessary to perform data preprocessing, the general purpose of which is to eliminate noise and to normalize and centralize the gray value of the image to provide a solid foundation for subsequent classification and identification. However, extensive image preprocessing may require large run-time cost, which threatens real-time capability. In our work, we perform a minimal amount of preprocessing while maintaining accuracy.

3.1.1 Face Detection. In our work, a Haar cascade classifier that is based on the object detection classifier proposed by Viola and Jones [38] is adopted for face detection in both our offline and real-time systems. We ignore the non-frontal situation since the main concentration is on the FER part. After loading the pretrained face XML classifier (required for face detection), the input images are loaded and converted into grayscale mode. If the classifier finds the faces, then the four coordinates of the rectangular region of interest (ROI) of the faces are returned. The four vertices are then used to crop the faces and, consequently, irrelevant backgrounds are deleted, as shown in Figure 1.

3.1.2 Data Augmentation. Data augmentation is often employed during the training of the CNNs, since the training process itself incorporates a large quantity of data. In the training scheme of this work, the cropped faces are first distorted with a lightweight library in TensorFlow before feeding them into the CNN. Each cropped face is randomly sampled by one of the distorted bounding boxes. The area of the sampled patch is [0.85, 1] of the original supplied image, and the number of generated images is as high as 100. Furthermore, after rescaling (which will be described below), the images will be randomly flipped horizontally to have two times more data. In this sense, the dataset is ultimately augmented by a factor of up to 200.

3.1.3 Resizing and Normalization. Since the CNN training input must be square, all the cropped images after the data augmentation are rescaled to 48×48 pixels. The reason for the selection of this 48×48 rescale parameter is to remain consistent with the resolution of the FER-2013 dataset. After rescale, the data are normalized into the range of $[-1, 1]$.

3.2 The Framework and Transfer Learning

3.2.1 CNN Structure. The first version of MobileNet (MobileNet V1) is employed as the CNN architecture in both offline and real-time systems of our work, since it focuses both on speed and size, and it is easy to be tuned for resources versus accuracy, as highlighted in this article. The core of the MobileNet V1 is that it decouples standard convolution into a depthwise convolution and a 1×1 pointwise convolution.

All the merits mentioned above contribute to our decision to select this MobileNet structure as the framework of this work. The characteristics of small size, low complexity, and remarkable accuracy enable this FER task to maintain a favorable trade-off between speed and accuracy compared to other popular CNN benchmarks (e.g., AlexNet, GoogleNet, VGG16, SqueezeNet).

3.2.2 Transfer Learning Strategy. One main problem for CNN-based FER is the insufficient size of most of the existing facial expression datasets. The required size of the labeled training data for CNNs to learn and extract features and obtain high accuracies is asked to be in millions (as mentioned in Section 1), while the size of most facial expression datasets is only hundreds or thousands. Training deep models with such limited amount of data is rather challenging, since sometimes it may lead to the problem of overfitting (the model may have poor performance on the test data while attaining rather perfect performance on the training data). In addition, it is time-consuming to train from scratch without taking advantage of the pretrained model. One of the common ways to address this problem is to use inductive transfer learning [39]—the so-called fine-tuning strategy. A common practice is first initializing the network with a set of pretrained weights (and bias) based on a large-scale dataset from one task and retraining these parameters for another new target task. These pretrained weights are adapted for initialization to all layers of the CNN except for the first and the last layer, since the input resolution or the number of classes of the dataset of the new task may vary from the dataset used for pretraining. A hallmark of this approach is the fine-tuning of a small target dataset based on the pretrained models on the ILSVRC-2012 (ImageNet), and many recent works [24] have established the feasibility of this strategy. To further compensate for the small size of the CK+ and JAFFE datasets during fine-tuning (the size of which are both under 1K) and overcome the difference between the target task and the source task, we follow the recent studies of References [25, 26] using the FER-2013 dataset (the size of which is more than 30K). Inspired by the associated transfer-learning schemes, "two-stage" fine-tuning is employed in the training scheme of CNN instead of only performing "one-stage" fine-tuning. This technique is a kind of "coarse-to-fine" training process, as illustrated in Figure 1. To make use of the large FER-2013 dataset, we perform the first stage of fine-tuning following the approach elaborated in Reference [39]. We first fine-tune the relatively small dataset FER-2013 (compared to ImageNet) in the target domain (in FER) by initializing the network with pretrained weights from ILSVRC-2012 in the source domain. Considering the distance between the tasks of FER and object classification (the target and the source), we refer to this first-stage fine-tuning as "coarse" fine-tuning. After obtaining the best-trained models from FER-2013, a second-stage fine-tuning step is applied to the JAFFE and CK+ datasets by transferring these sets of pretrained weights to the network. Since the target and the source tasks are the same in this process, we term it "refined" fine-tuning.

For the two fine-tuning schemes, the last fully connected layers are both replaced by a new classification layer classifying seven classes (the number of classes for CK+ may vary, as elaborated in Section 5). For the "coarse" fine-tuning, the set of weights of the first convolutional layer are randomly initialized from a Gaussian distribution, since the input of the images after preprocessing has a size of 48 × 48, while the original MobileNet V1 is 224 × 224. The initial learning rate of the first-stage fine-tuning is set to 0.001, which is relatively small, to "tune" the pretrained weights of the early layers of the network from ImageNet slightly. For the "refined" fine-tuning, the pretrained weights of the first layer from the FER-2013 are directly set as the initialization at the beginning because of the same input size of 48 × 48. At this stage of fine-tuning, the initial learning rate is set to a relatively large value (e.g., 0.045—see training details in Section 5) to "lock" the weights of early layers (since FER-2013 and the targeted datasets CK+ and JAFFE are in the same domain) and to relearn the high-level features for the specific dataset (CK+ or JAFFE).

3.2.3 *Joint Supervision.*

Inspired by the work of Wen et al. [27], center loss is applied in the training scheme to the discriminatory power. In the training process, center loss acts to reduce the intraclass differences by increasing the distance constraint between the features and its corresponding class center of the samples. The calculation of center loss is given in Equation (1) below:

$$L_{CL} = \frac{1}{2}\sum_{i=1}^{N} \|x_i - c_{y_i}\|_2, \tag{1}$$

where x_i stands for the ith deep features extracted before the final classification layer instead of those classified possibilities after the fully connected layer, and c_{y_i} represents the learned center for the y_ith class. The centers are updated within each iteration using a mini-batch strategy (the size of which is N in Equation (1)) and computed by taking the average of the deep features of the corresponding class. In addition, a hyperparameter α of 0.001 is also employed to control the learning rate of those centers during the update within each iteration:

$$c_{y_i}^{t+1} = c_{y_i}^{t} - \alpha \cdot \Delta c_{y_i}^{t}. \tag{2}$$

Center loss is supervised under collaboration with the conventional softmax loss in our work, closely following the approach of Reference [27]. The total loss used for network optimization is calculated in Equation (3):

$$L = L_S + \lambda \cdot L_{CL} = -\sum_{i=1}^{N} \log \frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{m} e^{W_j^T x_i + b_j}} + \frac{\lambda}{2}\sum_{i=1}^{N} \|x_i - c_{y_i}\|_2. \tag{3}$$

With a hyperparameter λ of 0.001 balancing the two loss functions, the network is optimized using the stochastic gradient descent (SGD) [40] with momentum to stabilize the update and greatly

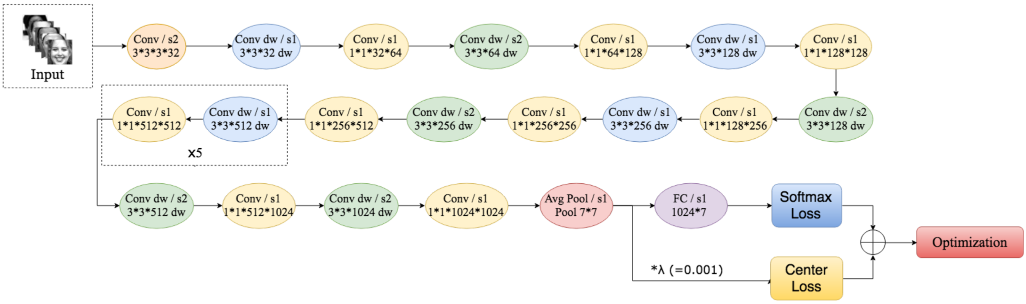


Fig. 2. The proposed structure of this work, using collaborated supervision with center loss. Note that the deep features before the fully connected layer are used for calculating center loss, while those after the fully connected layer are collected for the softmax loss.

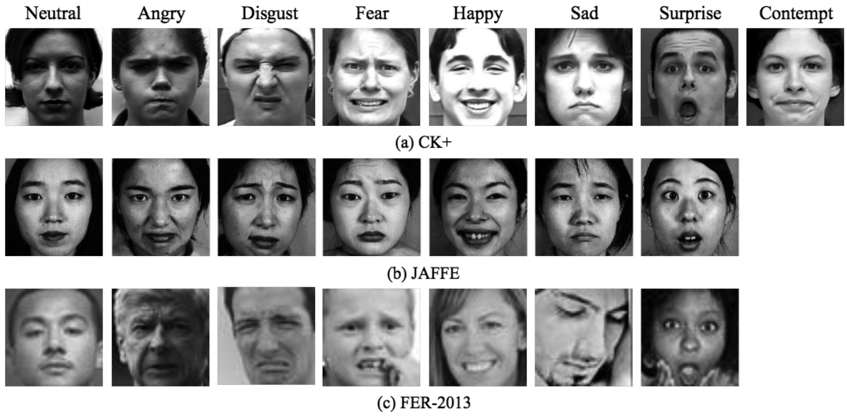
```
ALGORITHM 1: The learning algorithm
Input: Training data $x_i$, initialized parameters $\theta_c$ and weights $W$ of the network, hyperparameter $\lambda$, $\alpha$,
       learning rate and initialize the iteration: $t \leftarrow 0$.
Output: The updated parameters $\theta_c$
while not converge do
    $t \leftarrow t + 1$;
    Compute the total loss by $L^t = L_S^t + \lambda \cdot L_{CL}^t$;
    Compute the backpropagation error by $\frac{\partial L^t}{\partial x_i^t} = \frac{\partial L_S^t}{\partial x_i^t} + \lambda \cdot \frac{\partial L_{CL}^t}{\partial x_i^t}$;
    Update the weights $W$ by $\frac{\partial L^t}{\partial W^t}$;
    Update the parameter $c_j$ by $c_j^{t+1} = c_j^t - \alpha \cdot \Delta c_j^t$;
    Update the parameters $\theta_c$ by $\triangledown \theta_c = \frac{\partial L_S}{\partial \theta_c} + \frac{\partial L_{CL}}{\partial \theta_c}$.
end
```

speed up the convergence. This collaboration of supervision is illustrated in Figure 2, where the deep features extracted after the “Avg Pool” layer are used for calculation of center loss and where those elements extracted after the final fully connected layer are collected to calculate the softmax loss.

The details of the learning process are summarized in Algorithm 1; after initialization, the total loss is first computed during each iteration. Then, the weights and parameters of the network are updated through computing the gradient of backpropagation error.

4 DATASETS

The commonly used existing facial expression datasets are CK+, JAFFE, MMI, SPEW, YaleFace, FER-2013, MultiPIE, TFD, and so on. Among these, the two widely used standard datasets, CK+ [19] and JAFFE [20], are selected in this work for evaluation. Another dataset adopted in this work is FER-2013, which is currently one of the largest facial expression datasets. The examples of basic facial expressions of the three datasets used in this work are shown in Figure 3. The expressions in CK+ and JAFFE datasets are lab-controlled while these in FER2013 are wild-setting. The distribution of every class in each dataset is presented in Table 1, but only in the JAFFE dataset is the number of each class evenly distributed. In addition, the CK+ dataset includes a *contempt* class, while the other two datasets do not.



(a) CK+

(b) JAFFE

(c) FER-2013

Fig. 3. Examples of basic facial expressions of the three datasets. Note that every dataset contains seven basic expressions (*neutral, angry, disgust, fear, happy, sad, surprise*), while CK+ also includes a *contempt* class. The facial expressions of FER-2013 are wild-type, while those in the CK+ and JAFFE datasets are posed and collected in a laboratory-controlled environment.

Table 1. The Distributions of Every Class in Each Dataset

	FER-2013	CK+ (last frame)	JAFFE
Angry	4,593	45	30
Disgust	547	59	30
Fear	5,121	25	31
Happy	8,989	69	31
Sad	6,077	28	31
Surprise	4,002	83	30
Neutral	6,198	327	30
Contempt	0	18	0

Each dataset has seven basic classes, including *angry, disgust, fear, happy, sad, surprise, and neutral.* Especially for CK+, it also contains an extra *contempt* class and the onsets of 327 image sequences make up its *neutral* class.

- The Japanese Female Facial Expression Dataset (JAFFE) [20]
  This dataset consists of seven basic facial expressions (six basic ones and a neutral one). It was collected by ATR Human Information Processing Research Laboratory. The dataset contains 10 Japanese female expressors, each of whom posed $3 \sim 4$ examples of the six basic facial expressions and one for the neutral, resulting in a total of 213 static 256×256 images in the database. All images in the database are well posed, with even illumination, a single imaging background, and no occlusions such as eyeglasses. This database is characterized by relatively subtle emotion expression and presents a more challenging FER task.
- The Extended Cohn-Kanade Dataset (CK+) [19]
  The CK+ dataset, which was released in 2010, is an extension of the Cohn-Kanade (CK) dataset that increased the number of sequences and the number of subjects by 22% and 27%, respectively. The dataset includes 327 image sequences digitized into 640×480 from neutral to peak emotions of both posed and non-posed (spontaneous) expressions with FACS-coded emotion labels for the peak frames. The 123 subjects in this database range

from 18 ~ 50 years of age (81% Euro-American, 13% Afro-American, 6% other ethnicities), 69% of whom are female. In addition to the seven basic facial expressions, another class, *Contempt,* is included in this dataset, resulting in a total of eight classes of facial expressions. Several baseline results and benchmarking protocols for shape and appearance feature tracking, as well as emotion and AU labels, and are also provided in this dataset.

- FER-2013 Dataset [35]
  The FER-2013 dataset was presented in the sub-challenge/competition: Facial Expression Recognition Challenge of Challenges in Representation Learning in the ICML 2013 workshop that was hosted by Kaggle. The dataset itself retrieved from the Internet utilizing the Google image search API consists of 36,887 images where 28,709 are used as training data, 3,589 for public validation, and another 3,589 are used for private testing. It contains 48×48 pixel low-resolution grayscale images across seven basic facial expression classes. Because of the label noise and its various real-world conditions collected from the Internet, the FER-2013 becomes by far one of the largest and also most challenging spontaneous dataset for FER, which has a human recognition rate of around 65 ± 5%.

5 EXPERIMENTS AND EVALUATIONS

In this section, the details for the attained results of this work are provided. Note that the data for CNN training are preprocessed following the procedures elaborated in Section 3.1, while the data for evaluation does not conduct the data augmentation. For evaluation, the images in the JAFFE and CK+ datasets are randomly shuffled, respectively. Then each dataset is split into five groups during the CNN training, with four groups for training and one group for validation. For the FER-2013 dataset, the entire training set (28,709) and the public test set (3,589) are used for training and validation, respectively. The total loss function is optimized during the backpropagation using the stochastic gradient descent (SGD) optimizer with a momentum of 0.9, weight decay of 0.00004 on the model weights, and a batch size of 64 for a mini-batch. The initial learning rate for the first-stage and the second-stage fine-tuning are set to 0.01 and 0.045, respectively. For the "refined" fine-tuning, the initial learning rate is set to be slightly larger than the "coarse" fine-tuning, since we do not intend to heavily alter these already pretrained weights and mainly expect to fine-tune these high-level facial-expression-related features. The learning rate exponentially decreases by 0.94 times every 15 epochs of training. To reduce the occurrence of overfitting, a dropout strategy proposed by Hinton [41] is applied after the "Avg Pool" layer and before the final fully connected layer with a probability of 0.5.

5.1 Offline Experiments

5.1.1 Effects of Two-stage Fine-tuning. To begin with, the first experiment is conducted to show the effectivity of adopting this two-stage fine-tuning training strategy. We compare the results of two scenarios. The first one is directly fine-tuning from the pretrained weights based on ImageNet, and the second one adopts two-stage fine-tuning using the FER-2013 dataset towards both the CK+ and JAFFE datasets.

As for the first-stage fine-tuning on FER-2013 based on ImageNet, this approach obtains an accuracy of 67.03% on the public test set and 68.31% on the private test set. Based on the best pretrained model from FER-2013, second-stage fine-tuning is then conducted on the CK+ and JAFFE datasets. To verify the effectiveness of this two-stage fine-tuning method, we directly fine-tune the CK+ and JAFFE datasets from the ImageNet for comparison. The accuracy of validation on two datasets for comparing the two situations and the improvement in the accuracy rate are shown in Table 2. From Table 2, we can see that the adoption of the two-stage fine-tuning strategy improves the accuracy (3.28 ± 1.64% increase for CK+, and a 21.43% increase for JAFFE) for both datasets. For CK+,

Table 2. Illustration of the Effectiveness of Two-stage Fine-tuning

Dataset	One-stage Fine-tuning	Two-stage Fine-tuning	Accuracy Improvement
CK+	91.80 ± 1.64%	95.08%	**3.28±1.64%**
JAFFE	71.43%	92.86%	**21.43%**

The exact accuracy of the two datasets achieved in both cases and the improvement in the accuracy rate. Note that the accuracy for CK+ takes the six-class situation as the example.

Table 3. An Illustration of the Effectiveness of Joint Supervision

Dataset	Softmax Loss	Softmax Loss + Center Loss	Accuracy Improvement
CK+	95.08%	96.92%	**1.84%**
JAFFE	92.86%	95.24%	**2.38%**

Exact accuracy of the two datasets achieved in both cases and the improvement in the accuracy rate. Note that the accuracy for CK+ takes the six-class situation as the example.

the result of 95.08% has already reached the state-of-art accuracy compared to the literature [12, 42] (even without applying center loss). We take the result of evaluating six basic classes of facial expressions of this dataset (excluding *neutral* and *contempt*) as an example to do the comparison. For the JAFFE dataset, the accuracy of 71.43% is improved to 92.86% after applying this strategy, which can also beat the work of References [42, 43].

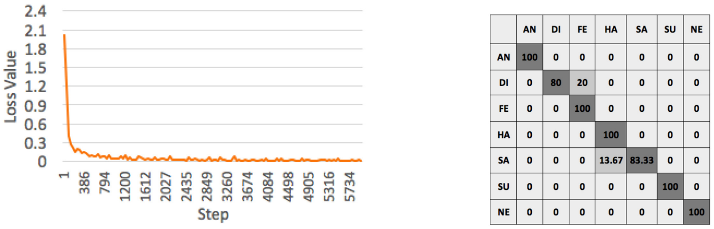
The results presented above do not consider center loss, since its effectiveness will be elaborated in Section 5.1.2. These results suggest that it is advantageous to boost the performance when training relatively small datasets such as JAFFE and CK+ by leveraging a large dataset such as FER-2013, which lies in the same domain.

5.1.2 Effects of Center Loss. To further enhance the discriminatory power of the proposed framework, center loss is employed as one part of the supervision signal. This experiment is conducted to show the superiority of center loss for improving results. Comparisons are carried out regarding both the JAFFE and CK+ datasets, and λ and α for center loss are fixed to 0.001. We compare two scenarios of adopting the joint supervision and the one only using the softmax loss for supervision. The exact accuracy of the two datasets achieved in both cases and the accuracy improvement are reported in Table 3, which illustrates that using center loss as an extra supervision signal can lead to an improvement in accuracy (1.85% for CK+ and 2.38% for JAFFE). After using this joint supervision, an accuracy of 96.92% and 95.24% on CK+ and JAFFE datasets can be obtained, respectively. During training, the total loss (with center loss) converges slightly slower, fluctuates more, and cannot reach the minimum that corresponds to training with only softmax loss. However, this configuration may stimulate the model to continue updating the weights and parameters during the backpropagation for better learning of discriminant deep features. The results presented above involve the supervised fine-tuning method as stated in Section 5.1.1. As stated above, the performance can be effectively enhanced when combining center loss with the conventional softmax loss during the training of the CNN for FER tasks.

5.1.3 Evaluation on JAFFE Dataset. As mentioned in Section 5.1.1, most of the related work that evaluated on the JAFFE dataset utilized conventional machine-learning techniques for customized feature extraction and classification, such as References [12, 14]. The dataset is prepared as stated

Table 4. Performance Comparison with the State-of-the-art Methods on the JAFFE Dataset

Methodology	Validation Accuracy (%)
IVA + HOG + Addaboost & SVM [12]	88.20
LBP + SVM/Adaboost [43]	86.67
Boosted Deep Belief Networks (BDBN) [44]	93
2D-LDA + SVM [45]	94.13
Gaussian Process with Polynomial Kernels & Gaussian RBF [46]	93.43 ~ 95.24
Local Fisher Discriminant Analysis (LFDA) [47]	94.37
Patch-based Gabor Feature + DL2 with SVM [6]	92.93
DCNN + SVM [14]	98.12
Advanced LBP + Tsallis Entropy + NLDA [10]	90.54 (48x48) 94.59 (64x64)
Feature-based Salient Facial Patches [42]	91.80
Proposed Method	**95.24**



(a) The training loss

	AN	DI	FE	HA	SA	SU	NE
AN	100	0	0	0	0	0	0
DI	0	80	20	0	0	0	0
FE	0	0	100	0	0	0	0
HA	0	0	0	100	0	0	0
SA	0	0	0	13.67	83.33	0	0
SU	0	0	0	0	0	100	0
NE	0	0	0	0	0	0	100

(b) Normalized confusion matrix

Fig. 4. The training loss and confusion matrix for the validation set of the JAFFE dataset.

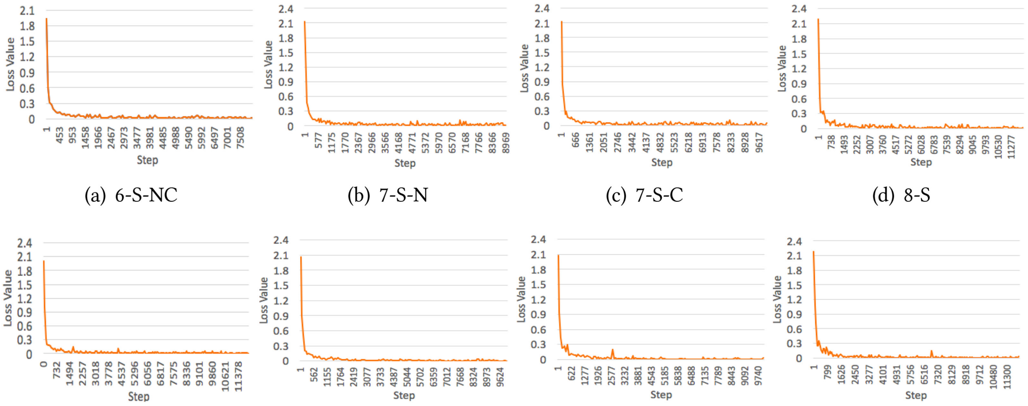
in Section 3.1 before feeding the data into the network for training. As presented in Table 3, the proposed method can ultimately reach an accuracy of 95.24% for evaluating seven classes of this dataset. The results of comparing our proposed model with the literature is shown in Table 4. This model achieves an average precision of 96.19%, recall of 94.76% and an F1-score of 95.47%. The training loss and the confusion matrix for the validation set of the JAFFE dataset are also presented; as shown in Figure 4, the training loss converges quickly, although *disgust* is sometimes confused with *fear*.

Relative to these conventional methods that employ geometric or appearance feature extraction techniques such as those proposed by References [10, 12], our proposed framework not only does not require human effort in feature extraction but also can surpass the maximum accuracy levels of prior work. Examples include the 88.2% accuracy of Reference [12], which used both geometric-based (inter vector angle, IVA) and appearance-based (HOG) feature extraction and the 92.93% accuracy of Reference [6], which employed the patch-based Gabor feature extraction and dense L2 with SVM for classification. Table 4 reveals that although a previous study [14] (which also applied DCNN) achieved an accuracy of 98.12%, surpassing our result slightly, the time cost for classification was much higher than ours (see Section 5.2).

5.1.4 Evaluation on CK+ Dataset. The second dataset to be evaluated is the CK+ dataset, on which eight different evaluation configurations are applied. The number of images in the neutral

Table 5. The Accuracy, Precision, Recall, and F1-score Values (%) of the Small Size (Composed of the Last Frames of Image Sequences) and the Large Size (Composed of the Last Three Frames of Image Sequences) of the CK+ Dataset with Different Evaluation Configurations

Evaluation Setups **(Small Size)**		Accuracy	Precision	Recall	F1-Score
6 Classes		96.92	92.78	95.95	94.34
7 Classes	Neutral_excluded	93.85	93.84	93.51	93.67
	Contempt_excluded	95.38	91.58	91.62	91.60
8 Classes		95.38	94.83	91.38	93.07
Evaluation Setups **(Large Size)**		Accuracy	Precision	Recall	F1-Score
6 Classes		100	100	100	100
7 Classes	Neutral_excluded	100	100	100	100
	Contempt_excluded	98.80 ± 0.40	99.25	99.21	99.23
8 Classes		99.69 ± 0.31	99.17	99.81	99.49



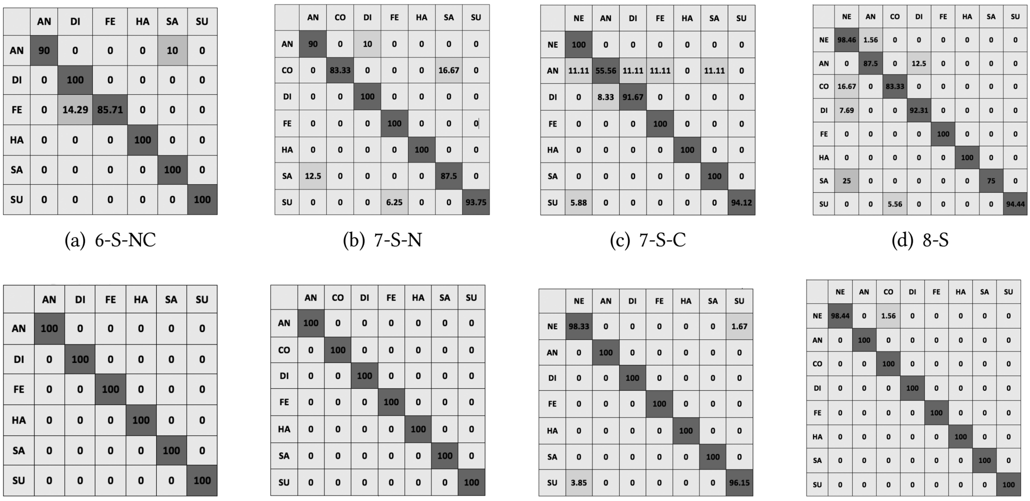
(a) 6-S-NC (b) 7-S-N (c) 7-S-C (d) 8-S

(e) 6-L-NC (f) 7-L-N (g) 7-L-C (h) 8-L

Fig. 5. The training loss of eight different configurations for the CK+ dataset. Note: 6, 7, 8: number of classes evaluated; S: small-size dataset (composed of the last frames of the image sequences); L: large-size dataset (composed of the last three frames of the image sequences); NC: *neutral* and *contempt* excluded; C: *contempt* excluded; N: *neutral* excluded.

class is kept the same for both the small-size and large-size datasets containing the onset frame of each image sequence. The accuracy, precision, recall, and F1-score values of eight situations evaluating the small-size and large-size datasets are reported in Table 5. The training loss and the confusion matrices for eight situations are shown in Figure 5, 6, respectively.

As shown in Table 5, as the number of classes to be evaluated increases, the accuracy drops slightly, since the process is more challenging with only a limited amount of data when the target task becomes more complex. For example, the accuracy of the small-size 6-class dataset can reach 96.92%, while this value drops by 1.54% in the 8-class situation. Note also that the increase in the size of the dataset can greatly help to boost accuracy when comparing the performance on the small-size datasets (which contain only the apex frame of each image sequence) and the large-size datasets (which contain the last three frames of the image sequences for enlarging the data



	AN	DI	FE	HA	SA	SU
AN	90	0	0	0	10	0
DI	0	100	0	0	0	0
FE	0	14.29	85.71	0	0	0
HA	0	0	0	100	0	0
SA	0	0	0	0	100	0
SU	0	0	0	0	0	100

(a) 6-S-NC

	AN	CO	DI	FE	HA	SA	SU
AN	90	0	10	0	0	0	0
CO	0	83.33	0	0	0	16.67	0
DI	0	0	100	0	0	0	0
FE	0	0	0	100	0	0	0
HA	0	0	0	0	100	0	0
SA	12.5	0	0	0	0	87.5	0
SU	0	0	0	6.25	0	0	93.75

(b) 7-S-N

	NE	AN	DI	FE	HA	SA	SU
NE	100	0	0	0	0	0	0
AN	11.11	55.56	11.11	11.11	0	11.11	0
DI	0	8.33	91.67	0	0	0	0
FE	0	0	0	100	0	0	0
HA	0	0	0	0	100	0	0
SA	0	0	0	0	0	100	0
SU	5.88	0	0	0	0	0	94.12

(c) 7-S-C

	NE	AN	CO	DI	FE	HA	SA	SU
NE	98.46	1.56	0	0	0	0	0	0
AN	0	87.5	0	12.5	0	0	0	0
CO	16.67	0	83.33	0	0	0	0	0
DI	7.69	0	0	92.31	0	0	0	0
FE	0	0	0	0	100	0	0	0
HA	0	0	0	0	0	100	0	0
SA	25	0	0	0	0	0	75	0
SU	0	0	5.56	0	0	0	0	94.44

(d) 8-S

	AN	DI	FE	HA	SA	SU
AN	100	0	0	0	0	0
DI	0	100	0	0	0	0
FE	0	0	100	0	0	0
HA	0	0	0	100	0	0
SA	0	0	0	0	100	0
SU	0	0	0	0	0	100

(e) 6-L-NC

	AN	CO	DI	FE	HA	SA	SU
AN	100	0	0	0	0	0	0
CO	0	100	0	0	0	0	0
DI	0	0	100	0	0	0	0
FE	0	0	0	100	0	0	0
HA	0	0	0	0	100	0	0
SA	0	0	0	0	0	100	0
SU	0	0	0	0	0	0	100

(f) 7-L-N

	NE	AN	DI	FE	HA	SA	SU
NE	98.33	0	0	0	0	0	1.67
AN	0	100	0	0	0	0	0
DI	0	0	100	0	0	0	0
FE	0	0	0	100	0	0	0
HA	0	0	0	0	100	0	0
SA	0	0	0	0	0	100	0
SU	3.85	0	0	0	0	0	96.15

(g) 7-L-C

	NE	AN	CO	DI	FE	HA	SA	SU
NE	98.44	0	1.56	0	0	0	0	0
AN	0	100	0	0	0	0	0	0
CO	0	0	100	0	0	0	0	0
DI	0	0	0	100	0	0	0	0
FE	0	0	0	0	100	0	0	0
HA	0	0	0	0	0	100	0	0
SA	0	0	0	0	0	0	100	0
SU	0	0	0	0	0	0	0	100

(h) 8-L

Fig. 6. The confusion matrices of eight different configurations for evaluating the CK+ dataset. Note: 6, 7, 8: number of classes evaluated; S: small-size dataset (composed of the last frames of the image sequences); L: large-size dataset (composed of the last three frames of the image sequences); NC: *neutral* and *contempt* excluded; C: *contempt* excluded; N: *neutral* excluded.

size). For example, using large-size the 8-class dataset provides an improvement of 4.31% over the accuracy of 95.38% for the corresponding small-size dataset.

5.2 Real-time Experiment

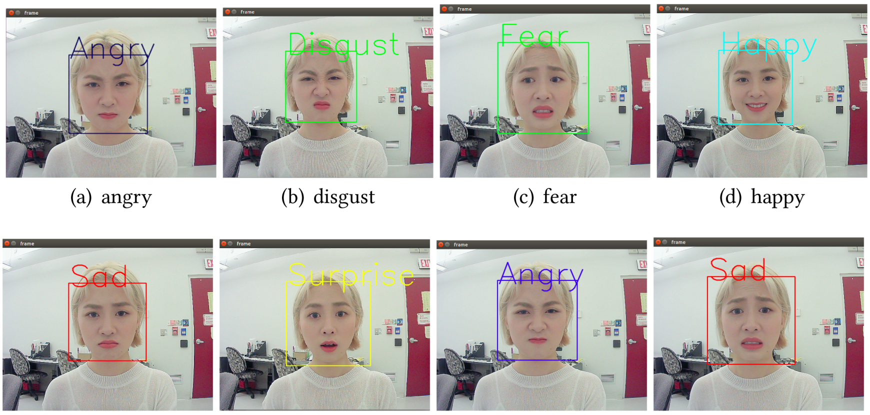
To verify the ability of running the proposed system in real time, we also design an implementation for real-time facial expression recognition from a standard webcam. After the webcam is connected to the network, the faces are preprocessed following the same procedures as described in Section 3.1 (without data augmentation). The data after preprocessing are then fed into the selected best-trained model to perform the classification. The subject is asked to face the camera frontally and display one of the basic facial expressions. The computation time for classifying one single frame is evaluated and results of comparison with the literature are shown in Table 6. Table 6 indicates that our proposed framework can perform classification (with a run-time of only approximately 3.57ms/frame on average) much faster than the conventional classifiers such as References [43, 44] and the one that also used CNN [14]. This implementation can subsequently classify the facial expressions of an arbitrary number of faces simultaneously running in real time, even in non-laboratory-controlled conditions. Selected real-time results are presented in Figure 7.

Note that this computation time includes only the time cost required for the model to perform the classification (disregarding the preprocessing time for the faces). The OpenCV face detection and the data preprocessing module (e.g., resizing, conversion to grayscale) takes approximately 46.93ms/frame and 7.49ms/frame, repectively. Even including these preprocessing modules, the run-time of the complete pipeline for FER is still sufficient for running in real time. Note also that when taking any arbitrary number M of people in a single frame of camera into account, the total time cost for this frame is $46.93 + M \times (7.49 + 3.57)$ms. The proposed framework can successfully distinguish basic facial expressions except for occasional confusion between "angry" and "disgust" and between "fear" and "sadness," as shown in Figures 7(g) and 7(h). It is possible that the subject who presented these posed expressions was not a professional actor; furthermore, these classes

Table 6. The Run-time Cost Comparison Against the State-of-art Methods for Real-time Facial Expression Recognition

Methodology	Classification Time (ms/frame)	System Arrangement
IVA + HOG + Adaboost & SVM [12]	66.7	2.4GHz CPU with no GPU
LBP + SVM/Adaboost [43]	227	Intel i3 2.2GHz CPU
Boosted Deep Belief Networks (BDBN) [44]	210	6-core 2.4GHz PC
2D-LDA + SVM [45]	35.7	Pentium IV with 2.80GHz
DCNN + SVM [14]	140 ~ 145	Tesla K20Xm GPU with compute version 3.5 / CPU (700–900ms)
CNN + OVA Binary Classification [48]	230	Intel Core i7 3.4GHz with a NVIDA GeForce GTX 660
68 Facial Landmarks + Optical Flow + SVM [11]	83.3	2.6GHz Intel Core i5 CPU
Proposed Method	**3.57**	**NVIDIA Quadro K4200 GPU / Intel Xeon (R) E5-1603 v3 2.8GHz * 4 CPU**

Note: This run-time cost only represents the time for classification and does not include the preprocessing module.



(a) angry (b) disgust (c) fear (d) happy

(e) sad (f) surprise (g) disgust but angry (h) fear but sad

Fig. 7. Examples of real-time classification for basic expressions.

tend to be misclassified because of similar geometric and appearance features that are hard even for a human to discern when reviewing the results of the evaluation on the JAFFE and CK+ datasets.

6 DISCUSSION

When evaluating the effectiveness of the two-stage fine-tuning strategy, we find that the JAFFE dataset achieves much lower accuracy (only 71.43%) than other reported methods [6, 44, 45] when fine-tuned only from ImageNet. Moreover, there is hardly any related work that directly used a CNN to fine-tune this database. The reason may be that it is challenging to have the deep model to learn and extract features (in the domain of FER) directly based on the pretrained model obtained in the domain of the object classification (i.e., the target task is much different from the source task), particularly when the size of the target dataset (JAFFE) is very small and the interclass variation of which is nuanced (the subjects are all Asian and the facial expressions of which are very subtle). However, after using an extra auxiliary FER-2013 dataset in the same domain (facial expression)

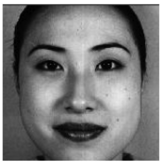


Fig. 8. Example of a mislabeled image in the JAFFE dataset. This image should have been labeled as *happy* but is labeled as *sadness* instead, which eventually leads to the misclassification in the results, as shown in the confusion matrix in Figure 4.

Table 7. A Review of the Evaluation Setups of the State-of-art Methods on CK+ Dataset

Methodology	Evaluation Setup	Validation Accuracy (%)
[49]	7 classes (neutral excluded) Last frame	88.00
[50]	6 classes (neutral & contempt excluded) Last frame	99.10
[33]	7 classes (neutral excluded) Last three frames	99.60
[12]	6 classes (neutral & contempt excluded) Last frame	88.20
[51]	6 classes (neutral & contempt excluded) Last frame	98.30
	8 classes Last frame	96.40
[44]	7 classes (neutral excluded) Last three frames	96.70
[48]	6 classes (neutral & contempt excluded) Last three frames	97.81
[11]	7 classes (contempt excluded) Last frame	98.12
[14]	6 classes (neutral & contempt excluded) Last frame	97.08

These evaluation setups vary in number of classes (from 6 to 8), which classes, and the size of the dataset (small or large) to be evaluated.

containing an enormous amount of data, the proposed method can achieve results superior to those of state-of-the-art methods [10, 47].

From the confusion matrix reported in Figure 4, we notice that it is weird that *sadness* is mixed up with *happy*. When we look into the results, it is eventually because of the mislabeled one in the dataset, as shown in Figure 8. We also find that *disgust* is sometimes confused with *fear*, which may be due to some of the facial expressions in these two classes having similar features (for example, the rise of the eyebrows), especially when the difference between classes is less salient, as in the JAFFE dataset.

In addition, when reviewing the literature on CK+, we find that there is no consistent evaluation configuration with this dataset, as shown in Table 7. First, the number of classes evaluated in the related work varies from six to eight, since there is an additional contempt class in the CK+

dataset to supplement the seven basic facial expressions. Moreover, even with the seven classes, the evaluations differ in whether the *neutral* class or *contempt* class is excluded. Since the CK+ dataset contains 327 image sequences from neutral to peak emotion, another problematic aspect involves the size of the dataset. Some of the work selects only the last frame of each image sequence [11, 12, 14, 49–51] comprising the dataset for training and testing, while other methods choose the last three frames [33, 44, 48] for the purpose of data augmentation. Consequently, it is challenging to make a fair comparison on this dataset using the literature. To address this issue, eight different evaluation configurations for the CK+ dataset are carried out, providing a baseline for a much more comprehensive evaluation comparison with the related work.

In addition, as shown in Table 5, evaluating the small-size 7-class neutral-excluded dataset is more challenging than the contempt-excluded one (the accuracy of the former is only 93.85%, which is 1.53% lower than the latter, and the same situation arises with the average precision, recall, and F1-score). This issue occurs because the number of contempt class is only 18 (only considering the last frame of the image sequences), which is much smaller than that of the neutral class (327), as illustrated in Table 1. Therefore, data imbalance is the biggest factor affecting the performance on small-size datasets when considering the neutral and contempt classes. Moreover, confusion sometimes occurs among the angry, disgust, and fear expressions in the small-size datasets due to the similar geometric and appearance features and the limited number of training samples. The training difficulty for the CNN with small and unbalanced data, particularly in classes with nuanced differences, is illustrated by these examples.

Since most of these facial expression datasets are collected under a controlled laboratory environment, the comparison of classifiers across datasets becomes more reliable than traditional self-classification determining the capability of generalization of classifiers. In this work, we verify this ability of the proposed method by first training with the entire JAFFE dataset (6-class excluding the *neutral* class) and evaluating on the CK+ dataset (6-class excluding the *contempt* and *neutral* classes), and vice versa. Following the same data preprocessing procedure described in Section 3.1, evaluating the CK+ dataset when training with the JAFFE dataset achieves an accuracy of 72.49%, while evaluating the JAFFE dataset when training with the CK+ dataset achieves an accuracy of 50.27 %. Note that cross-dataset evaluation on the JAFFE dataset is much more challenging than on the CK+ dataset due to the characteristics of the database itself. It is difficult to apply the CNN to classify a different dataset, especially when the interclass variation is not dispersed as the source dataset used for training. This factor may be one of the limitations of using the CNN, which can be described as "dataset-sensitive" for classification.

All the offline (training and testing) and real-time experiments in this work are implemented on an NVIDIA Quadro K4200 GPU based on the lightweight Slim library with TensorFlow [52] backend. We use both OpenCV and the Slim library for all image preprocessing, e.g., face detection, random flips, and rescaling.

7 CONCLUSION

In this article, a CNN-based system of estimating basic facial expressions that utilizes a transfer-learning strategy and a joint supervision is proposed, being able to recognize facial expressions of several subjects simultaneously from a webcam in a single-frame-based way. Relative to previous methods, the proposed framework not only can obtain the state-of-art accuracy on JAFFE and CK+ datasets but it also performs the classification much faster than conventional classifiers or even similar CNN-based work as a result of the characteristics of MobileNet and the GPU system. It can be seen that this CNN-based framework for FER is superior to conventional machine-learning methods in that it eliminates much human effort for complex feature extraction and does not require extensive preprocessing procedures while obtaining state-of-the-art results. Although some reported studies outperform our accuracy, those methods either do not provide real-time implementation or incur a much higher run-time cost than our approach.

In our future work, we will consider the influence of head-pose variations, since only frontal faces are taken into account in this work. In addition, efforts may also be tried to leverage the spatial information of video sequences rather than only a single frame to further enhance the system.

REFERENCES

[1] Ognjen Rudovic, Jaeryoung Lee, Miles Dai, Bjorn Schuller, and Rosalind Picard. 2018. Personalized machine learning for robot perception of affect and engagement in autism therapy. Retrieved from *arXiv preprint arXiv:1802.01186*.

[2] Ying Qiu, Yang Liu, Juan Arteaga-Falconi, Haiwei Dong, and Abdulmotaleb El Saddik. 2019. EVM-CNN: Real-time contactless heart rate estimation from facial video. *IEEE Trans. Multimedia* (2019). DOI: 10.1109/TMM.2018.2883866

[3] Abdulmotaleb El Saddik. 2018. Digital twins: The convergence of multimedia technologies. *IEEE MultiMedia* 25, 2 (2018), 87–92.

[4] Albert Mehrabian. 2008. Communication without words. *Communication Theory*, C. David Mortensen (Ed.). Transaction Publishers, New Brunswick, 193–200.

[5] Paul Ekman and Wallace V. Friesen. 2003. *Unmasking the Face: A Guide to Recognizing Emotions from Facial Clues.* ISHK.

[6] Ligang Zhang and Dian Tjondronegoro. 2011. Facial expression recognition using facial movement features. *IEEE Trans. Affect. Comput.* 2, 4 (2011), 219–229.

[7] Zhengyou Zhang, Michael Lyons, Michael Schuster, and Shigeru Akamatsu. 1998. Comparison between geometry-based and Gabor-wavelets-based facial expression recognition using multi-layer perceptron. In *Proceedings of the 3rd International Conference on Face & Gesture Recognition.* 454–459.

[8] Hong-Bo Deng, Lian-Wen Jin, Li-Xin Zhen, Jian-Cheng Huang.2005. A new facial expression recognition method based on local Gabor filter bank and PCA plus LDA. *Int. J. Inform. Technol.* 11, 11 (2005), 86–96.

[9] Feifei Zhang, Qirong Mao, Xiangjun Shen, Yongzhao Zhan, and Ming Dong. 2018. Spatially coherent feature learning for pose-invariant facial expression recognition. *ACM Trans. Multimedia Comput., Commun. Appl.* 14, 1s (2018), 27.

[10] Shu Liao, Wei Fan, Albert C. S. Chung, and Dit-Yan Yeung. 2006. Facial expression recognition using advanced local binary patterns, Tsallis entropies and global appearance features. In *Proceedings of the IEEE International Conference on Image Processing.* 665–668.

[11] Pranav Kumar, S. L. Happy, and Aurobinda Routray. 2016. A real-time robust facial expression recognition system using HOG features. In *Proceedings of the International Conference on Computing, Analytics and Security Trends.* 289–293.

[12] Rahul Islam, Karan Ahuja, Sandip Karmakar, and Ferdous Barbhuiya. 2016. SenTion: A framework for sensing facial expressions. Retrieved from *arXiv preprint arXiv:1608.04489*.

[13] Huei-Fang Yang, Bo-Yao Lin, Kuang-Yu Chang, and Chu-Song Chen. 2018. Joint estimation of age and expression by combining scattering and convolutional networks. *ACM Trans. Multimedia Comput., Commun. Appl.* 14, 1 (2018), 9–1.

[14] Veena Mayya, Radhika M. Pai, and M. M. Manohara Pai. 2016. Automatic facial expression recognition using DCNN. *Procedia Comput. Sci.* 93 (2016), 453–461.

[15] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. Retrieved from *arXiv preprint arXiv:1704.04861*.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, Vol. 1. 1097–1105.

[17] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. Retrieved from *arXiv preprint arXiv:1409.1556*.

[18] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 815–823.

[19] Patrick Lucey, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. 2010. The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops.* 94–101.

[20] Michael Lyons, Shigeru Akamatsu, Miyuki Kamachi, and Jiro Gyoba. 1998. Coding facial expressions with Gabor wavelets. In *Proceedings of the 3rd International Conference on Face & Gesture Recognition.* 200–205.

[21] Maja Pantic, Michel Valstar, Ron Rademaker, and Ludo Maat. 2005. Web-based database for facial expression analysis. In *Proceedings of the IEEE International Conference on Multimedia and Expo.* 5.

[22] Oliver Langner, Ron Dotsch, Gijsbert Bijlstra, Daniel H. J. Wigboldus, Skyler T. Hawk, and A. D. Van Knippenberg. 2010. Presentation and validation of the Radboud Faces Database. *Cognit. Emot.* 24, 8 (2010), 1377–1388.

[23] Yuxiang Jiang, Haiwei Dong, and Abdulmotaleb El Saddik. 2018. Baidu Meizu deep learning competition: Arithmetic operation recognition using end-to-end learning OCR techniques. *IEEE Access* 6 (2018), 60128–60136.

[24] Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway, and Jianming Liang. 2016. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Trans. Medical Imag.* 35, 5 (2016), 1299–1312.

[25] Bo-Kyeong Kim, Jihyeon Roh, Suh-Yeon Dong, and Soo-Young Lee. 2016. Hierarchical committee of deep convolutional neural networks for robust facial expression recognition. *J. Multimod. User Interfaces* 10, 2 (2016), 173–189.

[26] Hong-Wei Ng, Viet Dung Nguyen, Vassilios Vonikakis, and Stefan Winkler. 2015. Deep learning for emotion recognition on small datasets using transfer learning. In *Proceedings of the ACM International Conference on Multimodal Interaction.* 443–449.

[27] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. 2016. A discriminative feature learning approach for deep face recognition. In *Proceedings of the European Conference on Computer Vision.* 499–515.

[28] Charles Darwin and Phillip Prodger. 1998. *The Expression of the Emotions in Man and Animals.* Oxford University Press, USA.

[29] Paul Ekman and Erika L. Rosenberg. 1997. *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS).* Oxford University Press, USA.

[30] Di Huang, Caifeng Shan, Mohsen Ardabilian, Yunhong Wang, and Liming Chen. 2011. Local binary patterns and its application to facial image analysis: a survey. *IEEE Trans. Syst., Man, Cyber., Part C (Appl. Rev.)* 41, 6 (2011), 765–781.

[31] Yongqiang Yao, Di Huang, Xudong Yang, Yunhong Wang, and Liming Chen. 2018. Texture and geometry scattering representation-based facial expression recognition in 2D+3D videos. *ACM Trans. Multimedia Comput., Commun. Appl.* 14, 1s (2018), 18.

[32] Zhiding Yu and Cha Zhang. 2015. Image based static facial expression recognition with multiple deep network learning. In *Proceedings of the ACM International Conference on Multimodal Interaction.* 435–442.

[33] Peter Burkert, Felix Trier, Muhammad Zeshan Afzal, Andreas Dengel, and Marcus Liwicki. 2015. DeXpression: Deep convolutional neural network for expression recognition. Retrieved from *arXiv preprint arXiv:1509.05371.*

[34] Yichuan Tang. 2013. Deep learning using linear support vector machines. Retrieved from *arXiv preprint arXiv:1306.0239.*

[35] Ian J. Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, et al. 2013. Challenges in representation learning: A report on three machine learning contests. In *Proceedings of the International Conference on Neural Information Processing.* 117–124.

[36] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1–9.

[37] Ali Mollahosseini, David Chan, and Mohammad H. Mahoor. 2016. Going deeper in facial expression recognition using deep neural networks. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision.* 1–10.

[38] Paul Viola and Michael J. Jones. 2004. Robust real-time face detection. *Int. J. Comput. Vis.* 57, 2 (2004), 137–154.

[39] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 22, 10 (2010), 1345–1359.

[40] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[41] George E. Dahl, Tara N. Sainath, and Geoffrey E. Hinton. 2013. Improving deep neural networks for LVCSR using rectified linear units and dropout. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing.* 8609–8613.

[42] S. L. Happy and Aurobinda Routray. 2015. Automatic facial expression recognition using features of salient facial patches. *IEEE Trans. Affect. Comput.* 6, 1 (2015), 1–12.

[43] Rohit Verma and Mohamed-Yahia Dabbagh. 2013. Fast facial expression recognition based on local binary patterns. In *Proceedings of the 26th IEEE Canadian Conference on Electrical and Computer Engineering.* 1–4.

[44] Ping Liu, Shizhong Han, Zibo Meng, and Yan Tong. 2014. Facial expression recognition via a boosted deep belief network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1805–1812.

[45] Frank Y. Shih, Chao-Fa Chuang, and Patrick S. P. Wang. 2008. Performance comparisons of facial expression recognition in JAFFE database. *Int. J. Pattern Recog. Artific. Intell.* 22, 3 (2008), 445–459.

[46] Fei Cheng, Jiangsheng Yu, and Huilin Xiong. 2010. Facial expression recognition in JAFFE dataset based on Gaussian process classification. *IEEE Trans. Neural Netw.* 21, 10 (2010), 1685–1690.

[47] Yogachandran Rahulamathavan, Raphael C.-W. Phan, Jonathon A. Chambers, and David J. Parish. 2013. Facial expression recognition in the encrypted domain based on local fisher discriminant analysis. *IEEE Trans. Affect. Comput.* 4, 1 (2013), 83–92.

[48] Andre Teixeira Lopes, Edilson de Aguiar, and Thiago Oliveira-Santos. 2015. A facial expression recognition system using convolutional networks. In *Proceedings of the 28th SIBGRAPI Conference on Graphics, Patterns and Images.* 273–280.

[49] Kamlesh Mistry, Li Zhang, Siew Chin Neoh, Ming Jiang, Alamgir Hossain, and Benoît Lafon. 2014. Intelligent appearance and shape based facial emotion recognition for a humanoid robot. In *Proceedings of the 8th International Conference on Software, Knowledge, Information Management and Applications.* 1–8.

[50] Mundher Al-Shabi, Wooi Ping Cheah, and Tee Connie. 2016. Facial expression recognition using a hybrid CNN-SIFT aggregator. Retrieved from *arXiv preprint arXiv:1608.02833.*

[51] Pooya Khorrami, Thomas Le Paine, and Thomas S. Huang. 2015. Do deep neural networks learn facial action units when doing expression recognition? In *Proceedings of the IEEE International Conference on Computer Vision Workshops.* 19–27.

[52] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. TensorFlow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation.* 265–283.

Received September 2018; revised November 2018; accepted January 2019