2021 The 21st International Conference on Control, Automation and Systems (ICCAS 2021)
Ramada Plaza Hotel, Jeju, Korea, Oct. 12∼15, 2021

A Fast Real-time Facial Expression Classifier Deep Learning-based for Human-robot Interaction

Muhamad Dwisnanto Putro[1*], Duy-Linh Nguyen[2], and Kang-Hyun Jo[3]

[1,2,3]Department of Electrical, Electronic, and Computer Engineering, University of Ulsan, Ulsan, Korea (dputro@islab.ulsan.ac.kr)* Corresponding author

Abstract: Human-robot interaction drives the need for vision technology to recognize user expressions. Convolutional Neural Networks (CNN) has been introduced as a robust facial feature extractor and can overcome classification task. However, it is not supported by efficient computation for real-time applications. The work proposes an efficient CNN architecture to recognize human facial expressions that consist of five stages containing a combination of lightweight convolution operations. It introduces the efficient contextual extractor with a partial transfer module to suppress computational compression. This technique is applied to the mid and high-level features by separating the channel-based input features into two parts. Then it applies sequential convolution to only one part and combines it with the previous separated part. A shuffle channel group is used to exchange the information extracted. The structure of the entire network generates less than a million parameters. The CK+ and KDEF datasets are used as training and test sets to evaluate the performance of the proposed architecture. As a result, the proposed classifier obtains an accuracy that is competitive with other methods. In addition, the efficiency of the classifier has strongly suitable for implementation to edge devices by achieving 43 FPS on a Jetson Nano.

Keywords: Efficient CNN, Facial expression, Human-robot Interaction, Real-time.

1. INTRODUCTION

Face expression is a trend of computer vision work to recognize human facial emotions. This field is part of nonverbal communication that uses facial gestures to show their feelings. There are six expressions as basic human facial expressions, including anger, disgust, fear, happy, surprise, sad [1]. Furthermore, several other expressions can be interpreted as a combination of basic expressions. A different pattern of facial features distinguishes each expression. Therefore, facial information plays an essential role in the decision to predict facial expressions. In addition, the relationship and correlation between facial features can also be used as knowledge for classification. Human-Robot Interaction (HRI) utilizes robot communication with users to ease a task and support its success [2]. It maximizes the performance of the evaluation system of robots. Human expression is an input from the robot to recognize the user's feelings. Humans tend to talk and express their emotions through the face, so the facial expression system supports HRI performance. Moreover, implementation for robotic applications requires a real-time working vision system on portable devices [3]. The lightweight computing system avoids robot delays in receiving input information.

Pattern recognition technology is growing with the introduction of advanced methods of facial feature extraction. Conventional methods explore the extraction of facial color and texture to predict human facial expressions [4]. Even the Local Binary Pattern (LBP) has been applied to extract information from facial gestures [5]. However, these methods still achieve low accuracy due to weak feature extraction. On the other hand, Convolutional Neural Networks (CNN) have emerged as robust feature extraction. It has completed various classification tasks with complex features [6]. CNN explored the performance of neural networks that predict class probabilities by employing kernels of varying sizes and weights. Then the update weight process is applied to optimize the prediction performance by driving the actual value to approach the truth label score. VGG-16 has been present as a reliable backbone benchmark for extracting various object features [7]. This architecture consists of five stages by employing 3 x 3 filters to obtain local information extracted. However, this performance is not balanced with the computing power and the small number of parameters. Thus, it requires a graphics accelerator with large memory to work quickly and achieve real-time performance.

An efficient CNN architecture is needed by a predictor to work optimally on an edge device. The design strategy employs convolution layers and a smaller number of operations than usual. The efficient contextual module is introduced as an extraction module by implementing a partial cross transfer to save computation and parameters. The architecture has the same number of stages as VGG-16 but employs different operations, strategies, and convolution kernels. The proposed architecture encourages the classifier system to be implemented in a practical application for facial expressions shown in Fig. 1. The overall system requires a face detector to generate RoI (Region of Interest) faces at the beginning of the stage. It helps filter out extreme background and focus the classifier's performance on the face area.

Based on previous problems and reviews, the main contributions of the work are summarized as follows:

1. A new real-time face expression classifier is build using the efficient CNN architecture.
2. The efficient contextual extractor with a segment module to suppress computational compression and supports

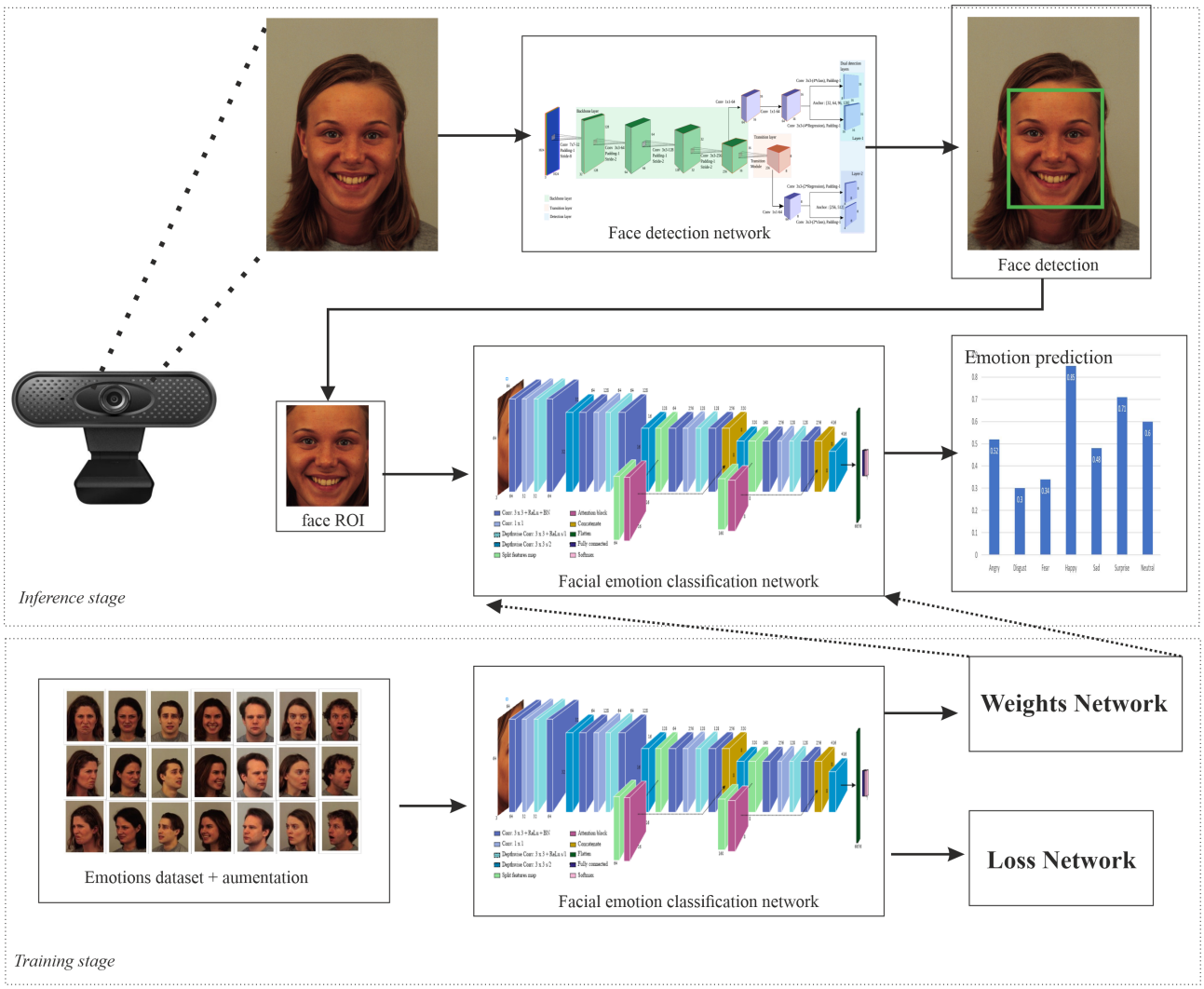

Fig. 1. Real-time face expression detection system. LWFCPU face detector [8] is applied to generate the Region of Interest (RoI) patches of the face.

real-time predictor performance. The proposed classifier obtains competitive performance with other competitors.

2. PROPOSED ARCHITECTURE

The proposed architecture consists of five stages that sequentially extract features. It distinguishes essential facial features to generate a relationship between their components. In addition, it also considers the computational efficiency of the model. Therefore, convolutional operations are minimized for practical application purposes. The proposed model employs two 3 x 3 convolutions at the beginning stage, as shown in Fig. 2. At this stage, it is implemented without reducing the size of the feature map. It assigns a 3 x 3 filter that locally works to extract neighbor features [7]. This kernel effectively captures the essential pixels of the face without generating rich parameters. In order to prevent overfitting at the training stage, ReLU and Batch Normalization are followed for each of these convolution operations [9]. ReLu is applied as an activator to filter out negative pixel information by converting it to zero. It helps prevent the loss of interest features in the next layer. Batch Normalization applies the normalization method of a set of distributed data in each mini-batch. It increases the speed and keeps the stability of the training. Therefore, both components are a supporting method that effectively improves the performance of the convolution operation. On the other hand, convolution with a single filter is applied to shrink the feature map at each stage. Depth-wise convolution saves the number of parameters by ignoring multiples of computing the number of channels.

2.1 Efficient contextual module

Practical applications require a deep learning architecture to produce low computational power. Meanwhile, a large number of 3 x 3 convolutions will generate a large number of parameters from a deep learning model. Therefore, the two stages use these filters sequentially to extract low-level features. At this stage, produce simple features such as lines, squares, and circles. In comparison, the third to fifth stages contain mid and high-level features. Global facial features are visible and identified in these blocks. However, the number of channels of the

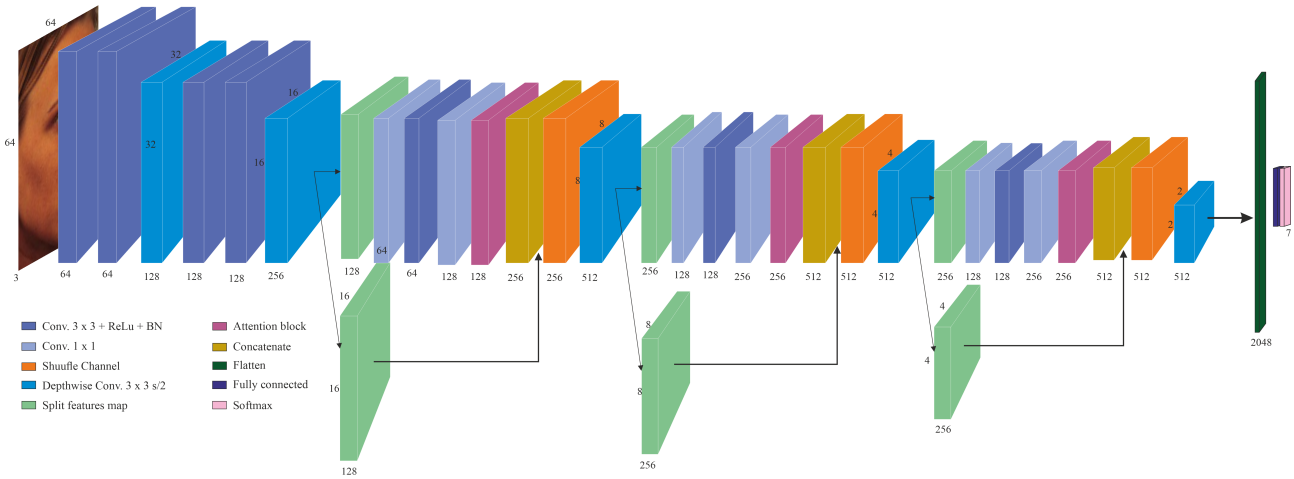

Fig. 2. The proposed architecture of an efficient contextual module. it consists of five stages and applies an enhancement module to produce specific mid and high-level features.

feature map becomes big as the size of the feature map decreases. At this stage, it avoids applying these filters sequentially, which will significantly increase computation. Therefore, an efficient contextual module with a partial transfer is used to replace this convolution operation without reducing the quality of the feature extractor. In the initial step, the input features (x) are divided into two feature maps in a balanced number of channels as expressed:

$$x = [x_{sn-1}, x_{sn}], \tag{1}$$

where sn is the total of the shared feature maps. Then the contextual module $C(\cdot)$ is applied to the partially split input (x_{sn}) by employing the convolution operation followed by the enhancement module, as expressed:

$$C(x_{sn}) = Att(W_3(W_2(W_1 x_{sn} + b_1) + b_2) + b_3). \tag{2}$$

It applies the convolutional bottleneck technique by using 1 x 1 convolution to reduce the channel size at the beginning of the layer. Furthermore, 3 x 3 kernels were employed to extract local features, and 1 x 1 convolution was used to reshaping the original channel size. This extracted feature map and partial feature map (x_{sn-1}) are combined to enrich the information by inserting information from the previous level features. The shuffle technique is applied to the fused layer to exchange channel-based information. It blended the knowledge of the feature map (x_{sn-1}) and extracted layers, as expressed:

$$y = shf(C(x_{sn}) \odot x_{sn-1}. \tag{3}$$

An efficient contextual module with partial transfer efficiently extracts specific features and combines them with feature map information at the previous level. The convolution operation explores a small number of channels, and it can save computational complexity. In addition, the attention module is used to improve the quality of the partially extracted feature map [10].

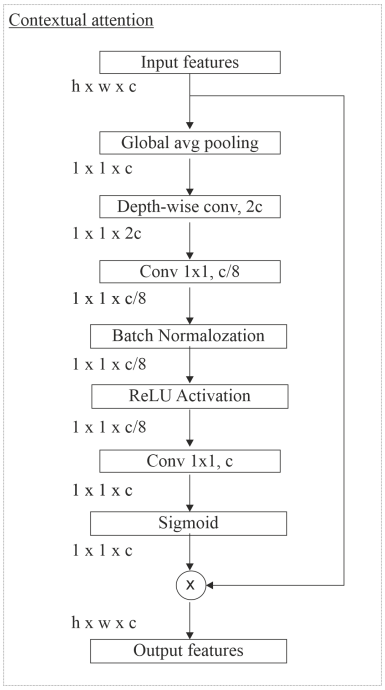

Fig. 3. The attention contextual module.

2.2 Attention contextual module

The proposed attention module is applied to the efficient contextual module to capture useful facial features. It also enhances contextual information from essential components of each local position. The input features of the sequential convolution in contextual module x_{sn} are summarized to obtain the average features for each channel, as expressed as follows:

$$s_{sn} = W_{dw} GAP(x'_{sn}). \tag{4}$$

The vectorization feature is scaled using depth-wise convolution to adjust each of its elements. Furthermore, it employs a sequential squeeze convolution by reducing the number of channels at the beginning of the layer to

Table 1. Evaluation results on KDEF and CK+ dataset.

Architectures	Accuracy (%)
KDEF dataset	
CRC	90.24
PCRC	90.71
RCFN	90.73
RCFN(CPL)	91.11
O-FER [11]	91.42
CCFN	91.60
Multi-Model fusion [12]	93.42
Proposed	**94.03**
CK+ dataset	
CCRNet [6]	98.14
ExpNet+Fusion	98.40
Ding et al	98.60
AM-Net [10]	98.68
Ofodile et al	98.70
MGLN-GRU [13]	99.08
Baseline+STCAM [14]	99.08
Proposed	**99.02**

save the number of parameters, as described:

$$Att_{sn} = \sigma(W_{v2}BN(ReLU(W_{v1}s_{sn}))) \cdot x'_{sn}. \quad (5)$$

Proposed attention generates the probabilities of the selected features by applying sigmoid activation (σ) at the end of the convolution layer. The results of this weight are used to the input features to update useful features for facial expressions and reduce the intensity of trivial features.

2.3 Prediction module

In general, the CNN in classification task model employs a prediction module in the head part to predict the category of each class label. The proposed architecture applies a flattening technique to form a vector of selected features at the ends of the backbone layer. Then fully connected is used to generate vector dimensions that match the number of expression categories. Finally, Softmax activation normalizes each logit score to generate the probabilities for each class label.

3. IMPLEMENTATION SETUP

The proposed network is trained on KDEF and CK+ datasets with several configurations. The augmentation technique is performed on the dataset by applying random brightness, contrast, flip, and rotation. Each dataset is trained at 300 epochs, and each epoch loads the overall instance into 128 mini-batches. It uses an Adam optimizer with 0.9 weight decay and 1e-7 epsilon to boost and stabilize the weight updating performance through the prediction error obtained from the cross-entropy function. It starts with a 0.0001 learning rate and automatically updates if accuracy doesn't increase within 25 epochs.

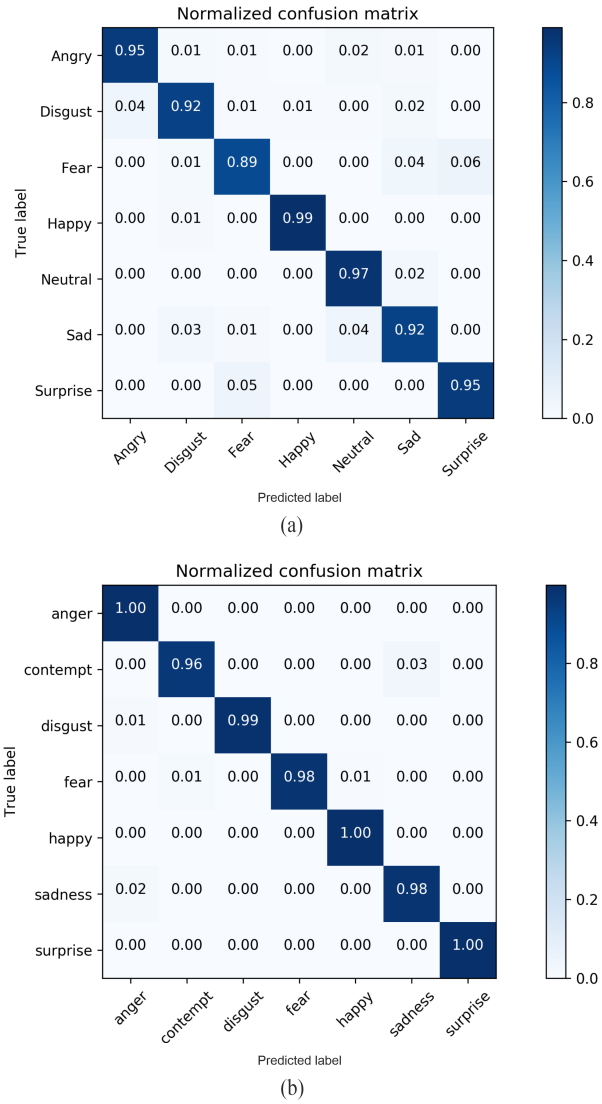

Fig. 4. Confusion matrix of the evaluation results on KDEF (a) and CK+ (b) dataset.

4. EXPERIMENTAL RESULTS

This section evaluates the proposed architecture on several datasets and analyzes the performance of an integrated module in real-time application on a Jetson Nano.

4.1 Evaluation on datasets

4.1.1 KDEF (Karolinska Directed Emotional Faces)

This dataset provides different multi-view face profiles that are suitably used by the real-case classifier. it contains 32-bit RGB images with 562 $\times$ 762 resolution. Neutral, happy, angry, fear, disgust, sadness, and surprise are expressions that are combined with five different angles (full left, half left, straight, half right, and full right profile). The proposed model achieves 94.03% accuracy and outperforms other methods [11, 12], as shown in Table. 1. This performance also shows that the proposed model obtains the best accuracy when predicting "Happy" expressions. It weakly classifies "Fear," which only achieves 89%, as shown by the confusion matrix in Fig. 4(a).

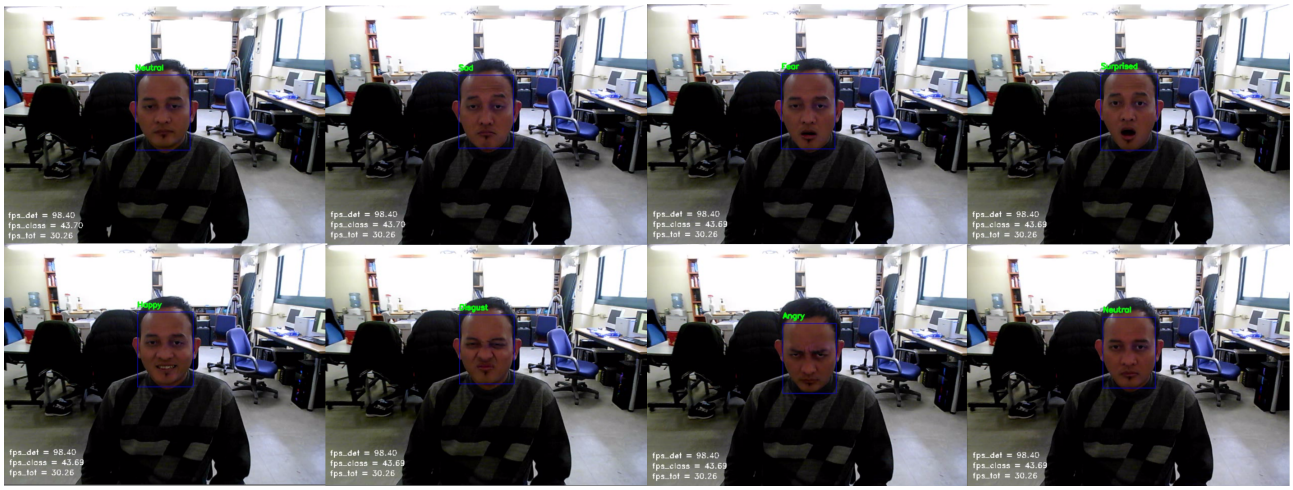

Fig. 5. Qualitative results in a application of Human-robot interaction.

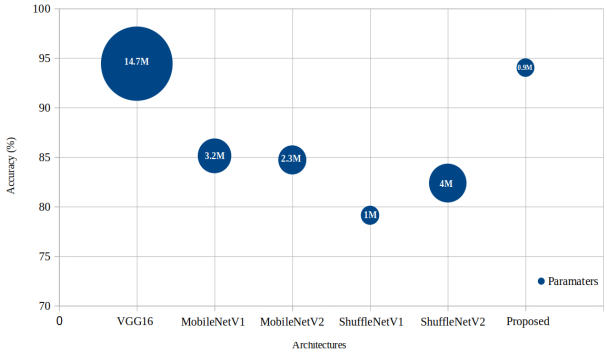

Fig. 6. Trainable parameters and accuracy comparison of the proposed module with benchmark model.

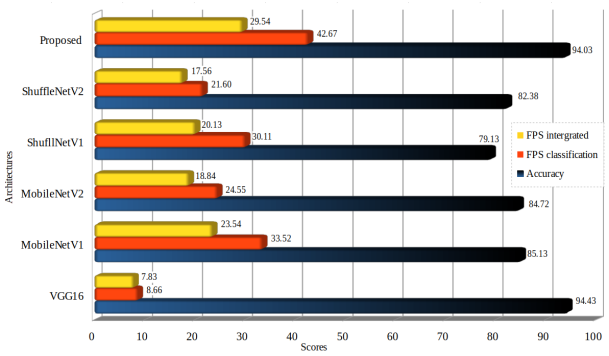

Fig. 7. Speed comparison with other models on a Jetson Nano device.

4.1.2 CK+ (Extended Cohn-Kanade)

This dataset consists of 593 sequential images with 327 labeled instances. In this public dataset, the proposed model takes the last three frames in each sequence and produces a total number of 981 pictures. It contains men and women face that provides seven standard emotions, including anger, contempt, disgust, fear, happiness, sadness, and surprise. Table. 1 shows that the proposed model achieves an accuracy of 99.02%, it has competitive performance from the best competitors [14, 15]. It also indicates that "anger," "happy," and "surprise" get a perfect performance, as shown in Fig. 4(b). On the other hand, "contempt" obtains the lowest accuracy in some instances, incorrectly predicting it as a "sadness" expression.

4.2 Efficiency in real-time application

The KDEF and CK+ datasets contain western and Asian faces without invalid labels, so these datasets are suitable to be used as knowledge bases for the real-case classifier. Even the KDEF dataset provides various facial poses with multi profiles. A trained model is suitably implemented in Human-robot applications, as shown in Fig. 5. The six basic emotions and neutral faces are accurately recognized. Additionally, A real-time application encourages a computer vision method to work quickly. Fig. 6 shows that the proposed module produces a smaller number of parameters than the mobile benchmark models. Although the VGG-16 gets the best accuracy, this backbone generates many parameters and works slowly on a Jetson Nano. In contrast, the proposed module only produces 911K parameters. Furthermore, the runtime efficiency in Fig. 7 shows that it achieved 30 FPS on a Jetson Nano, which outperformed Mobilenet (V1-V2), Sufflenet (V1-V2), and VGG-16. It integrates the classifier module with a face detector [8], as shown in Fig. 1. These results prove that this classifier is feasible to work in real-time and is reliable in classifying facial expressions accurately. Proposed architecture with efficient contextual modules emphasizes parameter and computational savings. In addition, this lightweight model maintains the quality of feature extraction and produces high accuracy to be applied as a classifier in the real-case application.

5. CONCLUSIONS

This paper presents a fast real-time facial expression classifier using a CNN-based efficient contextual module. The efficient module uses the partial transfer approach to split the features map into two parts and extract specific useful features on one segment. It combines previously separated feature maps and shuffles their channels to ex-

change information between feature maps. The attention contextual module enhances important features and reduces trivial information. As a result, the evaluation of two datasets promises that this classifier robust to predict seven expressions and achieve competitive results. Additionally, the proposed classifier can work fast by 30 FPS on a Jetson Nano that integrated with face detector. In future work, the proper loss function can be improved training performance without affecting the speed in the inference stage.

ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the government (MSIT). (No.2020R1A2C200897212)

REFERENCES

[1] P. Ekman, "*Facial expressions of emotion: New findings, new questions*," Psychological Science, vol. 3, no. 1, pp. 34–38, 1992.

[2] M. D. Putro and K. Jo, "*Real-time Face Tracking for Human-Robot Interaction*," 2018 International Conference on Information and Communication Technology Robotics (ICT-ROBOT), 2018, pp. 1-4.

[3] V. Mazzia, A. Khaliq, F. Salvetti and M. Chiaberge, "*Real-Time Apple Detection System Using Embedded Systems With Hardware Accelerators: An Edge AI Application*," in IEEE Access, vol. 8, pp. 9102-9114, 2020.

[4] S. M. Lajevardi and H. R. Wu, "*Facial Expression Recognition in Perceptual Color Space*," in IEEE Transactions on Image Processing, vol. 21, no. 8, pp. 3721-3733, Aug. 2012.

[5] Y. Ding, Q. Zhao, B. Li and X. Yuan, "*Facial Expression Recognition From Image Sequence Based on LBP and Taylor Expansion*," in IEEE Access, vol. 5, pp. 19409-19419, 2017.

[6] Z. Xi, Y. Niu, J. Chen, X. Kan and H. Liu, "*Facial Expression Recognition of Industrial Internet of Things by Parallel Neural Networks Combining Texture Features*," in IEEE Transactions on Industrial Informatics, vol. 17, no. 4, pp. 2784-2793, April 2021.

[7] C. Szegedy et al., "*Going deeper with convolutions*," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 1–9.

[8] M. D. Putro, D. Nguyen and K. Jo, "*Lightweight Convolutional Neural Network for Real-Time Face Detector on CPU Supporting Interaction of Service Robot*," 2020 13th International Conference on Human System Interaction (HSI), 2020, pp. 94-99.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "*Deep residual learning for image recognition*," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[10] J. Li, K. Jin, D. Zhou, N. Kubota, and Z. Ju, "*Attention mechanismbased cnn for facial expression recognition*," Neurocomputing, vol. 411, pp. 340 – 350, 2020.

[11] JunLan Dong, Ling Zhang, YunHua Chen, Wen-Chao Jiang, "*Occlusion expression recognition based on non-convex low-rank double dictionaries and occlusion error model*," Signal Processing: Image Communication, Volume 76, 2019, Pages 81-88, ISSN 0923-5965,

[12] A. Qi, J. Wei and B. Bai, "*Research on Deep Learning Expression Recognition Algorithm Based on Multi-Model Fusion*," 2019 International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), 2019, pp. 288-291.

[13] Mingjing Yu, Huicheng Zheng, Zhifeng Peng, Jiayu Dong, Heran Du, "*Facial expression recognition based on a multi-task global-local network*," Pattern Recognition Letters, Volume 131, 2020, Pages 166-171, ISSN 0167-8655,

[14] W. Chen, D. Zhang, M. Li and D. -J. Lee, "*STCAM: Spatial-Temporal and Channel Attention Module for Dynamic Facial Expression Recognition*," in IEEE Transactions on Affective Computing.