Facial emotion recognition using an ensemble of convolutional neural networks

Oleg Savenko[1,†], Serhiy Balovsyak[2,*,†], Oleksandr Derevyanchuk[2,†], Mykola Ilashchuk[2,†] and Stepan Melnychuk[2,†]

[1] *Khmelnytskyi National University, Instytuts'ka Street 11, 29016 Khmelnytskyi, Ukraine*
[2] *Yuriy Fedkovych Chernivtsi National University, Kotsiubynskoho Street 2, 58012 Chernivtsi, Ukraine*

Abstract

A review of modern developments in the field of automatic emotion recognition is prepared. The relevance of developing computer programs for facial emotion recognition using an ensemble of convolutional neural networks (CNN) is substantiated. A method of increase the accuracy of facial emotion recognition is proposed, which consists in using an ensemble of three CNN. The inputs of CNN #1 receive the initial image, the inputs of CNN #2 receive the contour image, and the inputs of CNN #3 receive the inverted initial image. The developed CNN are designed to recognize seven emotions: Angry, Disgust, Fear, Happy, Neutral, Sad and Surprise. The resulting values of the probability of recognizing emotions are calculated by averaging the outputs of all CNN of the ensemble. The software for recognizing emotions based on face images was developed in Python using the Google Colab cloud platform. In the process of training the CNN, parallel computing using GPU was performed.

Keywords

Facial emotion recognition, ensemble of convolutional neural networks, cloud platform, parallel computing, software, Python.

1. Introduction

Automatic Facial Emotion Recognition (FER) is widely used in various fields of science, technology, education, medicine and business [1, 2]. Taking into account the user's emotions is important for organizing human-machine interaction using adaptive interfaces that change the complexity of tasks or interface design. In educational systems, the emotional state of students is used to increase their motivation during classes and individualize learning, and is taken into account in the process of implementing STEM projects [3]. Facial Emotion Recognition is complemented by other types of image analysis [4]. Determining a person's emotional state by facial expression is implemented by hardware-software systems in which images are read from video cameras. In mobile and embedded systems, microcomputers, such as Raspberry Pi, are often used for image processing. The advantages of computer systems for facial emotion recognition include the availability of hardware and relatively high recognition accuracy.

In general, the sequence of facial emotion recognition consists of four stages. The first stage is the reading of images from digital video cameras. The second stage is the detection of facial images by their spatial localization in rectangular areas, which is performed, in particular, by the Viola-Jones method or using convolutional neural networks (CNN) [5]. The third stage is the normalization of the image of the facial area by scaling it to a standard size, optimizing brightness and contrast. At the fourth stage, FER is implemented using, as a rule, facial features (coordinates of key points) or CNN. The use of CNN allows recognizing emotions with various distortions of images (for example, due to face turns and changes in lighting conditions), but requires a long

AIT&AIS'2025: International Scientific Workshop on Applied Information Technologies and Artificial Intelligence Systems, December 18–19 2025, Chernivtsi, Ukraine
[1,*] Corresponding author.
[†] These authors contributed equally.
savenko_oleg_st@ukr.net (O. Savenko); s.balovsyak@chnu.edu.ua (S. Balovsyak); o.v.derevyanchuk@chnu.edu.ua (O. Derevyanchuk); ilashchuk.mykola.m@chnu.edu.ua (M. Ilashchuk); s.melnychuk@chnu.edu.ua (S. Melnychuk)
0000-0002-4104-745X (O. Savenko); 0000-0002-3253-9006 (S. Balovsyak); 0000-0002-3749-9998 (O. Derevyanchuk); 0009-0002-7996-6176 (M. Ilashchuk); 0009-0001-4325-2342 (S. Melnychuk)

training procedure. As a result of recognition, the emotion that prevails on the person's face is determined. The most commonly recognized 7 types of emotions: Angry, Disgust, Fear, Happy, Sad, Surprise and Neutral [5].

The problem is that existing facial emotion recognition systems have limited accuracy, which is due to the individual characteristics of people, different conditions for obtaining photos, and other factors. When using CNN, the real accuracy of face recognition is often limited by overtraining of neural networks, when at a certain epoch of training the error for the validation dataset begins to increase. Errors in automatic FER systems significantly complicate the practical implementation of such systems, requiring additional checks. Despite the promising potential of CNN, it is difficult to achieve high recognition accuracy for different types of face images using a single CNN.

The aim of the work is to develop methodology and software for facial emotion recognition using an ensemble of convolutional neural networks, which contains three CNN. The first CNN is trained on a dataset of ordinary facial images in a normalized scale. The second CNN is trained on a dataset of image contours, and after training uses image contours as input signals. The third CNN is trained on a dataset of inverted images, and accordingly, after training uses inverted images as input signal. Accordingly, the second CNN reacts to sharp changes in the brightness of the facial image, which are spatially localized in the contour areas (for example, areas of facial features: eyes, nose and mouth). The third CNN significantly responds to areas of facial features, which in the original (non-inverted image) often have low brightness due to shading. Each of the CNN in ensemble has certain characteristics of facial images, so their results complement each other.

The results of emotion recognition are obtained by averaging the outputs of all CNN from ensemble, which allows for the most complete consideration of all characteristics of the facial image and more accurate recognition of emotions. Thus, the results of the study are relevant, as they allow for increasing the accuracy of recognition of a person's emotional state.

2. Related works

A review of the possibilities of using deep learning for automatic facial emotion recognition was provided in the work [6]. It is shown that emotion recognition has a wide scope of application. The possibilities of FER using CNN of various architectures, in particular VGG and ResNet, are described. The possibilities of recognizing emotions based on images, as well as on signals of other types, in particular, sound, are considered [7].

The research [5] is devoted to the practical recognition of 7 basic emotions: Angry, Disgust, Fear, Happy, Sad, Surprise and Neutral (Fig. 1). The recognition is performed based on the CNN model trained on the FER-2013 dataset. The developed CNN architecture contains 6 convolutional layers, and 2 fully connected output layers. Each of the 7 outputs of CNN means the probability that a certain emotion is present in the input image. The quantity of convolutional layers, the sizes of their filters, and the quantity of fully connected layers are chosen to reduce the learning error for both the training and validation datasets. The CNN inputs are fed with face images of a certain size, so the initial images are scaled, as a rule, by interpolation methods [8].

Face detection in images using the Viola-Jones method (using Haar cascades) is described in the work [9]. It is shown that the use of different Haar cascades allows determining the orientation of the face. Facial expression recognition in images using CNN with the A-MobileNet architecture is described in the research [10]. It is shown that the CNN model with the A-MobileNet architecture provides high accuracy in facial expression recognition.

In the research [11] a simplified CNN model with dense-connectivity architecture is described, but it provides accurate FER. This model is designed for integration into learning management systems for the purpose of correcting the educational process. It is shown that the developed CNN model requires less computational resources compared to the known CNN architectures: VGG, ResNet, Xception, EfficientNet, DenseNet [12–14].

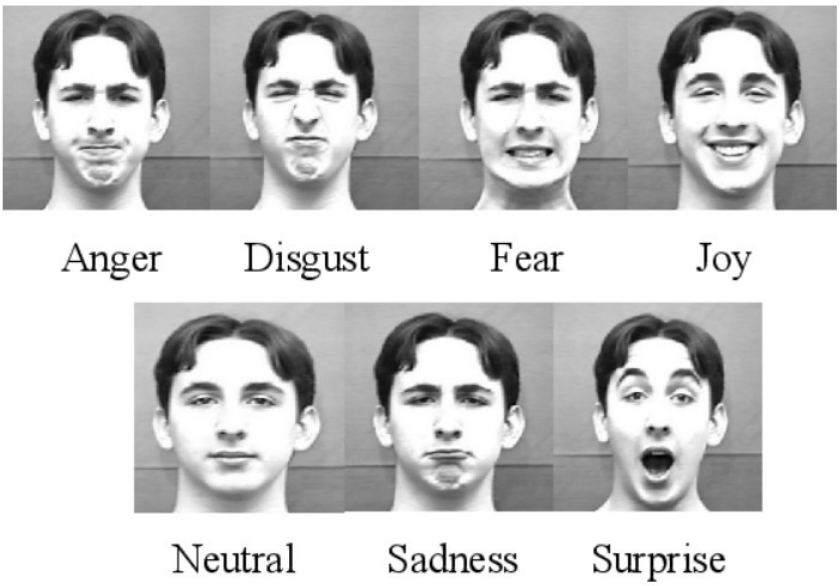

Figure 1: Examples of faces expressing 7 basic emotions [5].

The use of local facial features and Scale Invariant Feature Transform (SIFT) for emotion recognition is described in the work [15]. Recognition of different classes (emotions) was performed using the support vector machine (SVM) method. It is shown that increasing the accuracy of emotion recognition can be achieved by expanding the dataset.

In the work [16], the recognition of 16 emotions in facial images obtained as frames of a video stream was considered. Based on the analysis of the images, the following emotions were determined: Anger, Anxiety, Confidence, Contentment, Courage, Disgust, Excitement, Gratitude, Happiness, Joy, Love, Pride, Relaxation, Resolve, Sadness, and Tranquillity. Emotion recognition in images was performed using recurrent neural networks (RNN). A significant quantity of recognized emotions allows to accurately assess a person's emotional state, but this somewhat complicates the construction of a recognition system and the interpretation of the obtained results.

The use of CNN for facial emotion recognition is described in the research [17]. It has been demonstrated that focusing the attention of CNN on important areas of the face (for example, eyes) allows to increase the accuracy of emotion recognition.

In the work [18] it is shown that filtering face images before processing them with CNN allows increasing the accuracy of facial expression recognition. In particular, the following filtering methods were used: Average filtering, Median filtering, Gaussian filtering, Non local means filtering, Bilateral filtering. A specialized system for FER using CNN is described in the research [19]. However, in almost all cases of using CNN, the question arises of increasing the accuracy of facial emotion recognition.

Thus, the analysis of the reviewed works confirms the relevance of FER using an ensemble of CNN, and also shows the need for software implementation and research of such an emotion recognition system.

3. A proposed method for recognizing emotions using an ensemble of convolutional neural networks

According to the proposed method, to increase the accuracy of facial emotion recognition, not one CNN, but an ensemble of three CNN is used (Fig. 2). The input images for the ensemble are grayscale images $Img = (Img(i, k))$, where $i = 0,..., M-1$; $k = 0,..., N-1$; M, N are the height and width of the image (in pixels). The structure of all CNN #1-#3 is the same, and the difference lies in the input signals. CNN can potentially also process the color images, but for this the channels of the red, green and blue components must be processed in the same way as the brightness channel (image *Img*).

The inputs of the CNN # 1 are fed with the initial image *Img* (as an array of size $M \times N$ elements, the values of which denote the brightness of the corresponding pixels). At the outputs of the CNN # 1, a one-dimensional array $Y = (Y(j))$ is obtained, where $j = 0, ..., Q_Y-1$, $Q_Y = 7$ – the quantity of recognized emotions. The values of the elements of the array $Y(j)$ denote the probability of the appearance of a face with emotion number j at the input of the CNN. The emotions at the outputs of the CNN are numbered as follows: Angry ($j = 0$), Disgust ($j = 1$), Fear ($j = 2$), Happy ($j = 3$), Neutral ($j = 4$), Sad ($j = 5$), Surprise ($j = 6$).

The inputs of the CNN #2 are fed with contour images Cont, which are calculated based on the initial images *Img* using the Sobel method. The outputs of the CNN # 2 obtain a one-dimensional array *Ycont* with the size of Q_Y elements, the values of which indicate the probability of recognizing a certain class (emotion).

The inputs of the CNN #3 are fed with the images Inv, which are calculated by inversion of the initial images *Img*. At the outputs of the CNN # 3, a one-dimensional array Yinv with the size of Q_Y elements is obtained, the values of which also mean the probability of recognizing a certain emotion. The outputs Y, Y_{cont} and Y_{inv} of all CNN of the ensemble are averaged, as a result of which a one-dimensional array YS with the size of Q_Y elements is calculated, the values of which mean the probability of recognizing an emotion (result for the ensemble).

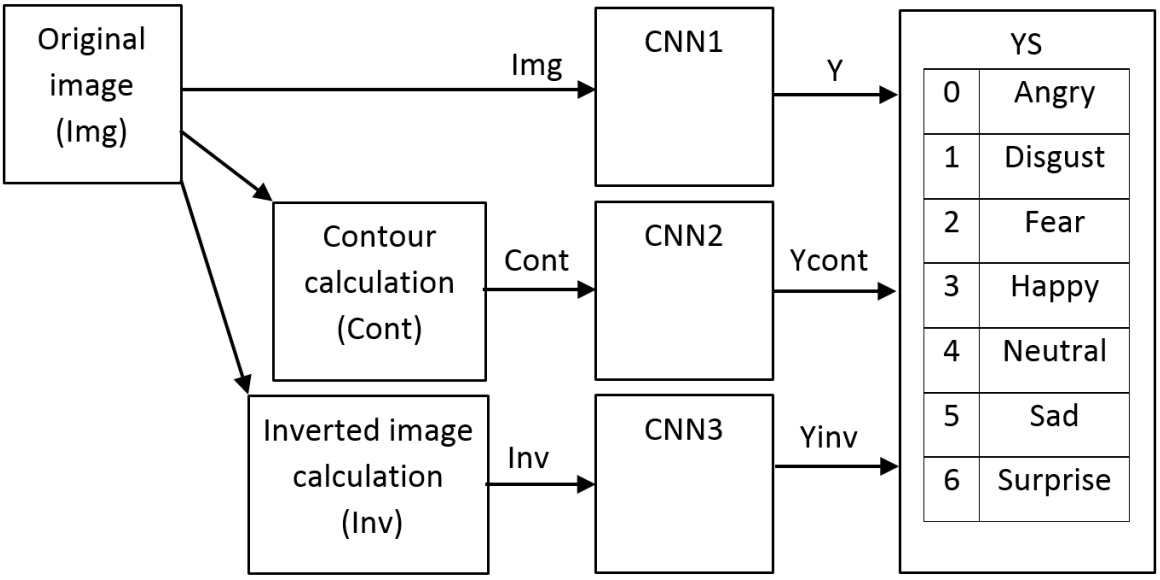

Figure 2: Schema of an ensemble of convolutional neural networks.

The used CNN contains 6 convolutional layers and 2 output fully connected layers. The operation of the convolution layers consists in convolving the initial image *Img* (for the input layer) or feature maps (for the following layers) with the convolution kernel w_C (size $M_w \times N_w$ elements). A separate convolution kernel is used for each feature map.

Mathematically, the convolution operation for the image *Img* is described by the formula

$$I_C(i,k) = \sum_{m=0}^{M_w-1} \sum_{n=0}^{N_w-1} I_{mg}(i-m+m_c, k-n+n_c) \cdot w_c(m,n), \quad (1)$$

where $I_C(i, k)$ is the value of the signal element with indices (i, k) after convolution; $i = 0, ..., M-1$, $k = 0, ..., N-1$; M, N are the height and width of the image (in pixels); m_C is the center of the filter kernel in height; n_C is the center of the filter kernel in width.

Sobel contour extraction is performed by width and height. The calculation of horizontal contours *ContX* is performed by convolution of the initial image *Img* with the filter kernel *wXS*. Similarly, the calculation of vertical contours *ContY* is performed by convolution of the initial image *Img* with the filter kernel *wYS*.

The kernels of the horizontal and vertical filters are described by the formulas

$$w_{XS}=\begin{bmatrix}-1 & -2 & -1\\ 0 & 0 & 0\\ 1 & 2 & 1\end{bmatrix},\quad w_{YS}=\begin{bmatrix}-1 & 0 & 1\\ -2 & 0 & 2\\ -1 & 0 & 1\end{bmatrix}. \tag{2}$$

The resulting contours *Cont* are calculated as the root of the sum of the squares of the horizontal *ContX* and vertical *ContY* contours. After calculating the contours *Cont*, their binarization is not performed, which allows us to accurately take into account the contour intensities for local areas of the image.

The program for FER was developed in Python [20] in the notebook (web shell) Jupyter Notebook using the Google Colab cloud platform. The structure of the CNN (the number of convolutional and fully connected layers, the sizes of their filters) was used the same as in the research [5]. The CNN is first trained on a dataset and then used to recognize emotions in facial images. The CNN model contains 6 convolutional layers, followed by two fully connected layers (with ReLU activation). The output layer of the CNN contains 7 neurons, the output of each of which represents the probability of a certain emotion in the input facial image.

The software implementation of the CNN was performed using the tensorflow and keras libraries. Adding convolutional layers was performed using the “Conv2D” method, batch normalization was implemented using the “BatchNormalization” method. Pooling was performed using the “MaxPooling2D” method, as a result of which the size of the feature maps was reduced by 2 times. Data filtering was performed using the “Dropout” method. Data filtering and batch normalization methods are used to prevent overtraining and to improve the generalization capabilities of the CNN model. In the convolutional layers, filters with a size of 3 × 3 elements and the activation function “relu” were used. The general structure of the CNN is as follows:

1. Convolutional layer #1, contains 32 filters.
2. Batch normalization layer #1.
3. Convolutional layer #2, contains 64 filters.
4. Batch normalization layer #2.
5. Pooling layer #1.
6. Dropout layer #1.
7. Convolutional layer #3, contains 128 filters.
8. Batch normalization layer #3.
9. Convolutional layer #4, contains 128 filters.
10. Batch normalization layer #4.
11. Pooling layer #2.
12. Dropout layer #2.
13. Convolutional layer #5, contains 256 filters.
14. Batch normalization layer #5.
15. Convolutional layer #6, contains 256 filters.
16. Batch normalization layer #6.
17. Pooling layer #3.
18. Dropout layer #3.
19. Dense layer #1, 256 neurons, activation function “relu”.
20. Batch normalization layer #7.
21. Dropout layer #4.
22. Dense layer #2, 7 neurons, activation function “softmax”.

During the training of the CNN, iterator objects were used to read images, which read images in batches (for example, 64 images) and transferred them to the CNN inputs. Due to batch processing, the CNN training time is significantly reduced. The CNN training was performed using the "Adam" method, and during the training process, the loss error was minimized by the "categorical_crossentropy" metric.

Parallelization of calculations during training of the CNN was performed using Graphics Processing Units (GPU) NVIDIA T4 in Google Colab. Due to parallelization of calculations, the training time of the CNN was reduced by an order of magnitude. To avoid overtraining of the CNN and increase the diversity of the dataset, data augmentation was used, which consisted of image displacements in height and width, mirror reflections of the initial images. Prediction of the probabilities of the appearance of emotions in facial images at the output of the CNN (vector Y) is performed using the “predict” method.

4. Experimental results of training convolutional neural networks

The training of the CNN #1-#3 was carried out on the FER-2013 dataset [21], which contains grayscale images of faces (size M × N = 48 × 48 pixels). The faces in the images occupy almost the entire space and are centered. The dataset contains 24400 face images, which are divided into a training dataset (22968 images) and a validation dataset (1432 images) (Fig. 3). The validation dataset was used to prevent overtraining of the CNN.

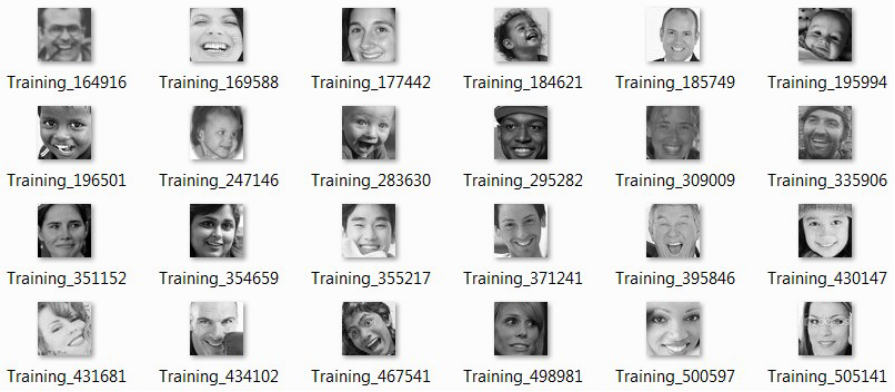

Figure 3: A fragment of the training sample of the FER-2013 dataset, showing an images with the "happy" emotion.

The training of the CNN was performed for 50 epochs, and the training results were evaluated by the Loss and Accuracy metrics separately for the training and validation datasets. For the training sample, the accuracy and loss training parameters were evaluated, and for the validation sample, the val_accuracy and val_loss parameters were evaluated. Training was performed for all CNN of the ensemble, namely for CNN #1 (Fig. 4), CNN #2 (Fig. 5), CNN #3 (Fig. 6). The training time ts for all CNN turned out to be approximately the same. The values of the CNN models were saved in files, which were subsequently used for FER.

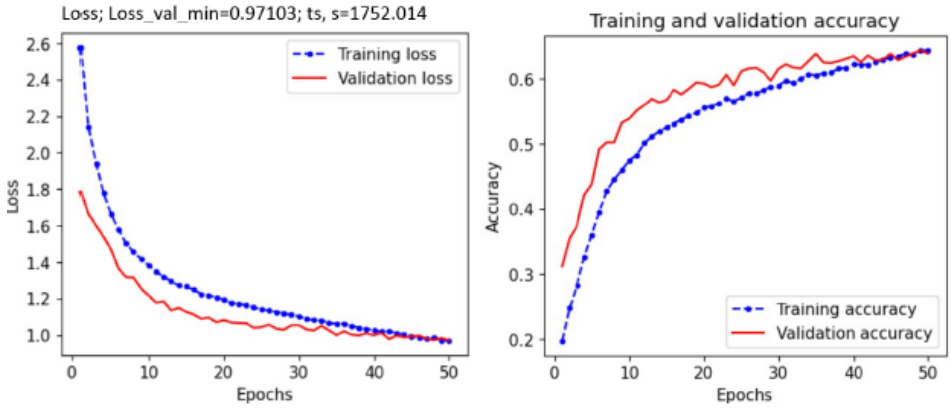

Figure 4: Graphs of training Loss and Accuracy for CNN #1 (initial input images Img), CNN model "model_CNN_50_1" was trained, quantity of epochs – 50, in the last epoch the training parameters: accuracy = 0.6455; loss = 0.9636; val_accuracy = 0.6383; val_loss = 0.9710.

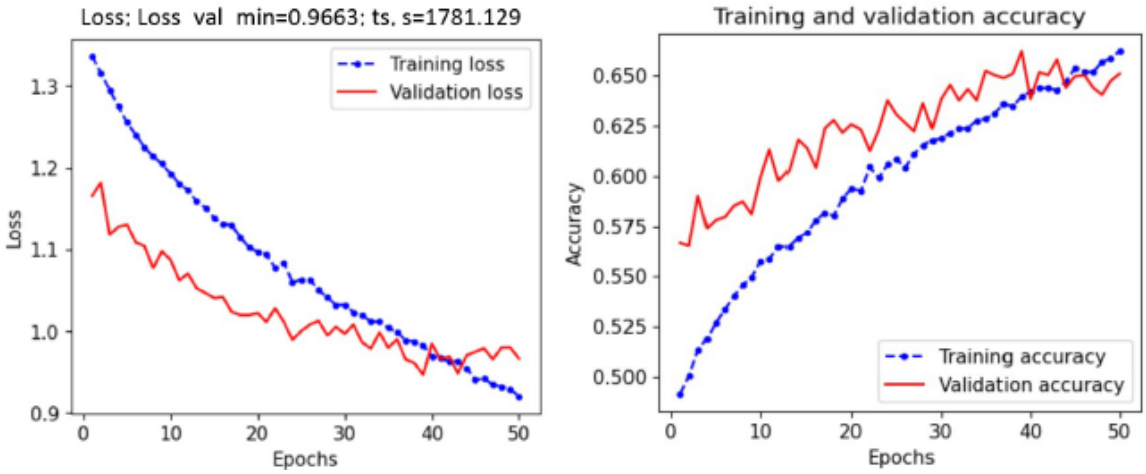

Figure 5: Graphs of training Loss and Accuracy for CNN #2 (input contour images), CNN model "model_CNN_50_cont" was trained, quantity of epochs – 50, training parameters in the last epoch: accuracy = 0.6595; loss = 0.9244; val_accuracy = 0.6508; val_loss = 0.9663.

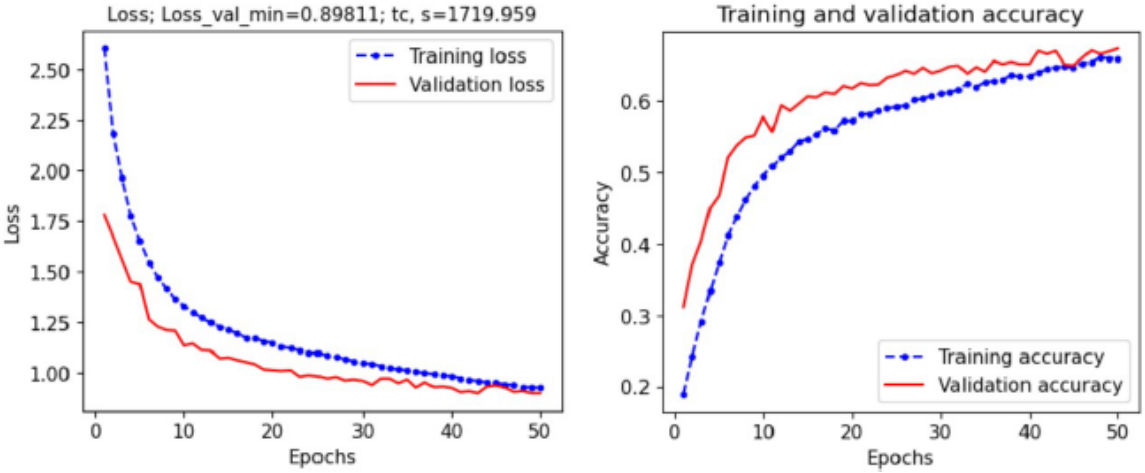

Figure 6: Graphs of training Loss and Accuracy for CNN #3 (inverted input images), the CNN model "model_CNN_50_inv_2" was trained, the number of epochs is 50, in the last epoch the training parameters: accuracy = 0.6528; loss = 0.9337; val_accuracy = 0.6732; val_loss = 0.8981.

According to the val_loss parameter for the validation dataset, the best results were obtained for CNN #3 (val_loss = 0.8981, Fig. 5), and similar results were obtained for other CNN: val_loss = 0.9710 for CNN #1 and val_loss = 0.9663 for CNN #2. According to the val_accuracy parameter, the best results were also obtained for CNN #3 (val_accuracy = 0.6732, Fig. 5). The slight advantage in accuracy for CNN #3 can be explained by the fact that in the inverted images, facial features (in particular, eyes, nose, mouth) have high brightness (compared to the original images).

In all cases, a decrease in the training error for the training and validation datasets was obtained. However, with the number of training epochs close to 50, the error for the validation dataset practically stopped decreasing. Therefore, for more accurate training of the CNN #4 with a larger number of epochs (70 epochs), the training dataset was expanded by image augmentation. However, in previous cases, the image shift was performed in the range of 10% of their height and width, and in this case, the image shift was performed in the range of 15%. As a result the values accuracy = 0.6716 and loss = 0.8971 were obtained for the training dataset, and val_accuracy = 0.6837 and val_loss = 0.9197 were obtained for the validation dataset. Compared with the previous models of the CNN #1-#3, a slight improvement was obtained for the val_accuracy parameter for model #4. Therefore, to obtain higher accuracy of CNN training, it is necessary to obtain augmented images not only by shifting the initial images, but also by rotating them, changing their brightness and contrast.

5. Results of facial emotion recognition

Using the developed ensemble of CNN #1-#3, facial emotion recognition was performed on images of the validation dataset. In most cases, emotions were recognized correctly and reliably for all CNN. However, in some cases, CNN recognized several emotions simultaneously (Fig. 7–Fig.10).

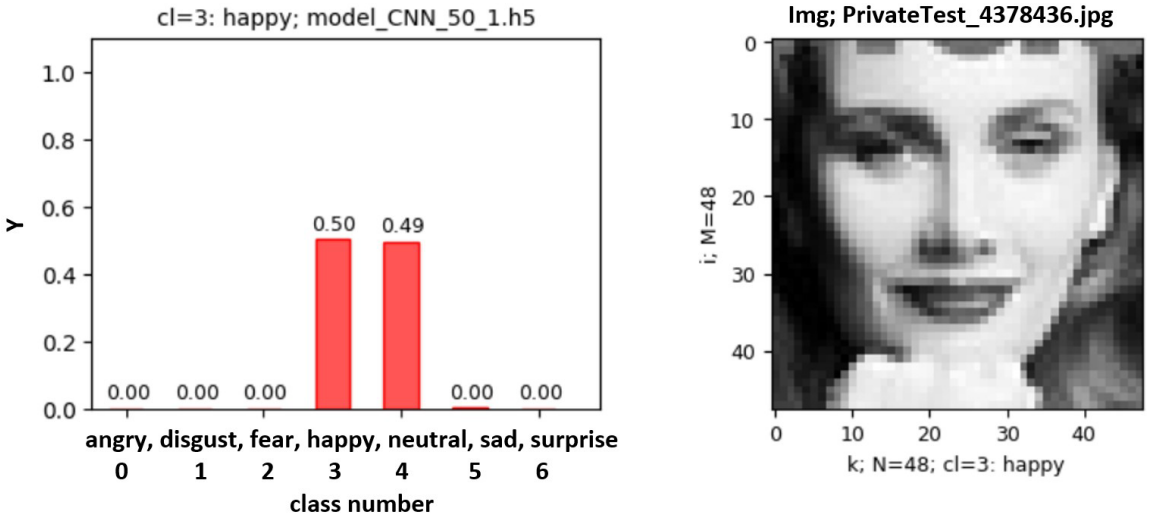

Figure 7: Results of emotion recognition on the initial face image for CNN #1.

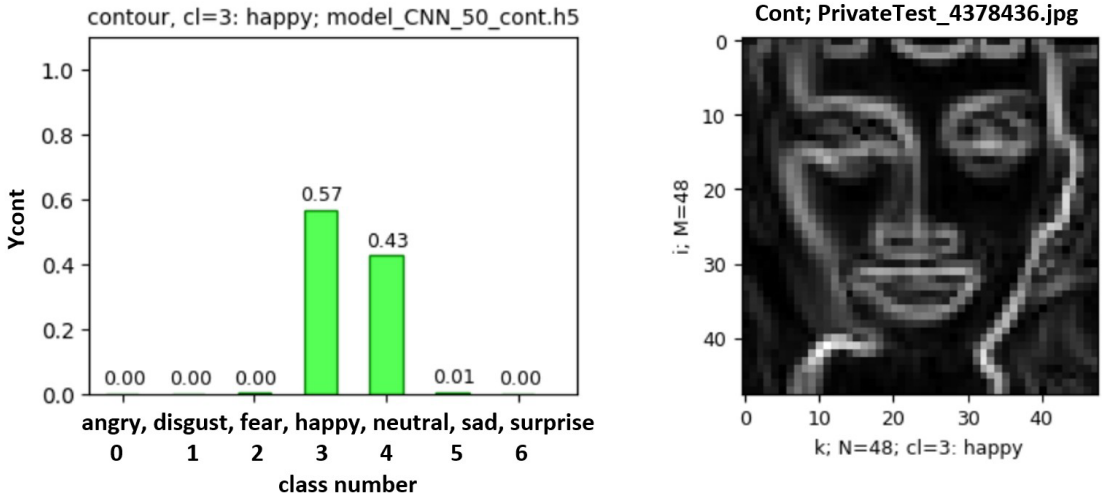

Figure 8: Results of emotion recognition on the image of facial contours for CNN #2.

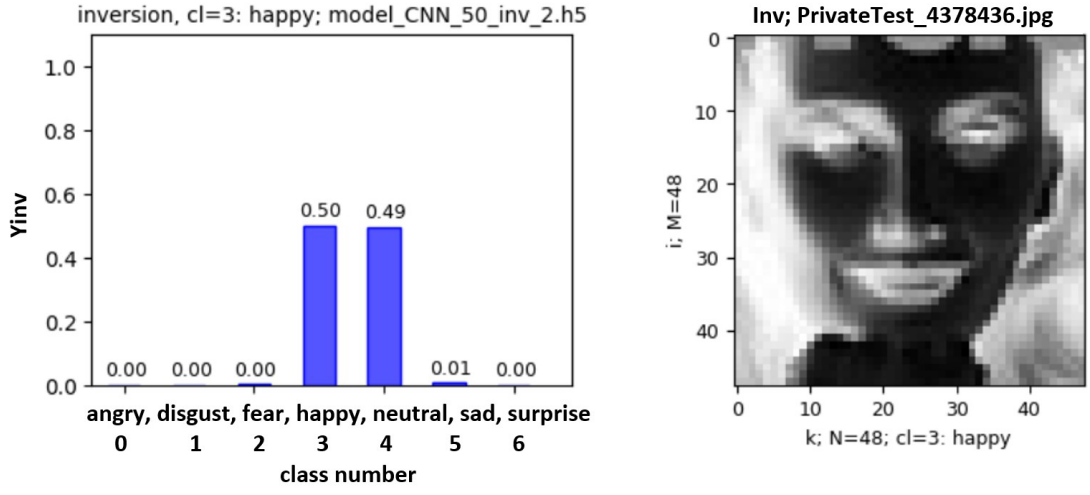

Figure 9: Results of emotion recognition on an inverted face image of CNN #3.

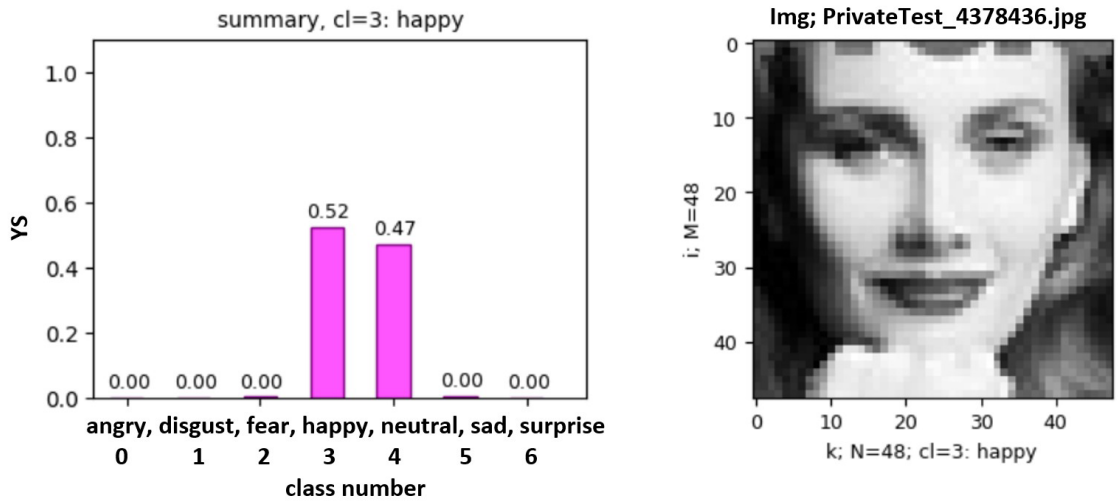

Figure 10: Results of emotion recognition on facial images by the ensemble of CNN #1-#3.

For example, for an image with the correct emotion "happy", CNN #1 (Fig. 7) and CNN #3 (Fig. 9) recognized two emotions with almost the same probability. However, CNN #2, which analyzed the contours of the image, in this case gave more accurate emotion recognition results (Fig. 8). Accordingly, more accurate facial emotion recognition results were obtained by the CNN ensemble, which averages the results of CNN #1-#3 (Fig. 10). In Fig. 7-Fig. 10, the left part shows the CNN outputs (which mean the probabilities of recognizing certain emotions) for the image shown in the right part of the figures. Each figure shows the class number cl for the recognized emotion and its name.

In another case of a difficult-to-analyze image, the CNN #1 (Fig. 11), CNN #2 (Fig. 12) and CNN #3 (Fig. 13) correctly identified the main emotion in the face image, but recognized other emotions. Therefore, the most accurate emotion recognition in this case was performed by the CNN ensemble (Fig. 14), since by averaging the CNN output vectors *Y*, *Ycont* and *Yinv*, the probability of a correctly recognized emotion is high, and the probability of recognizing other emotions has decreased. This is explained by the fact that for a correctly recognized emotion, all CNN determine approximately the same values of the probability of recognition, and for other emotions, different CNN determine different probabilities (which are largely mutually compensated).

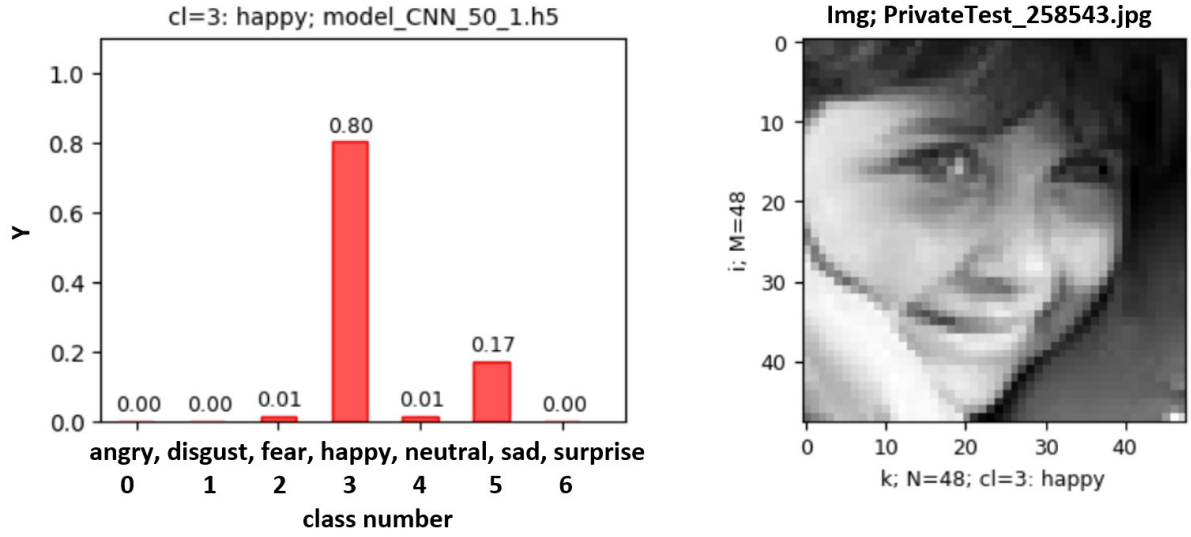

Figure 11: Results of emotion recognition on the initial face image for CNN #1.

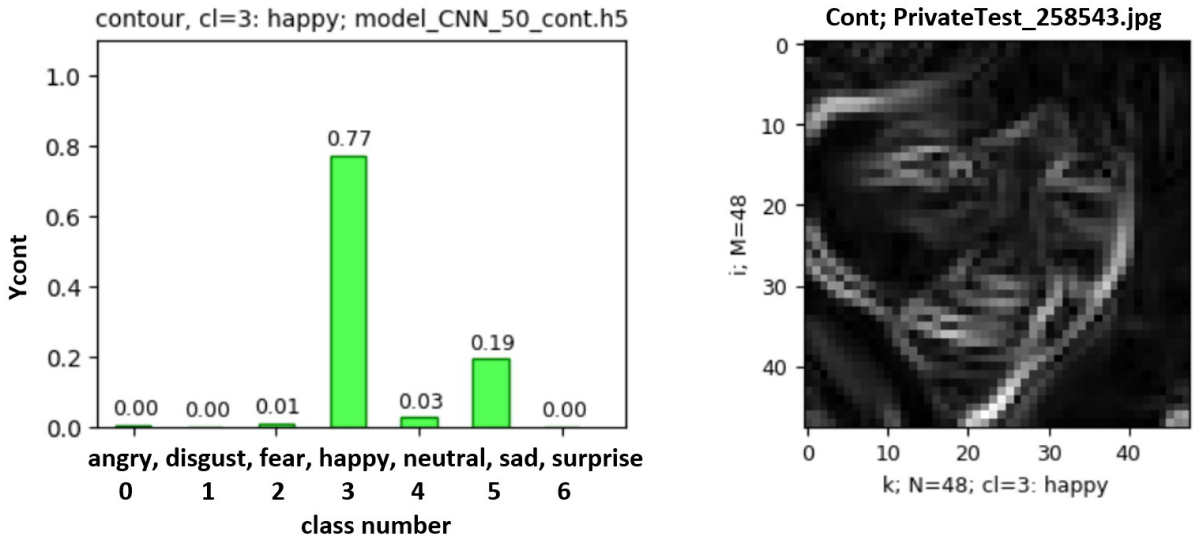

Figure 12: Results of emotion recognition on the image of facial contours for CNN #2.

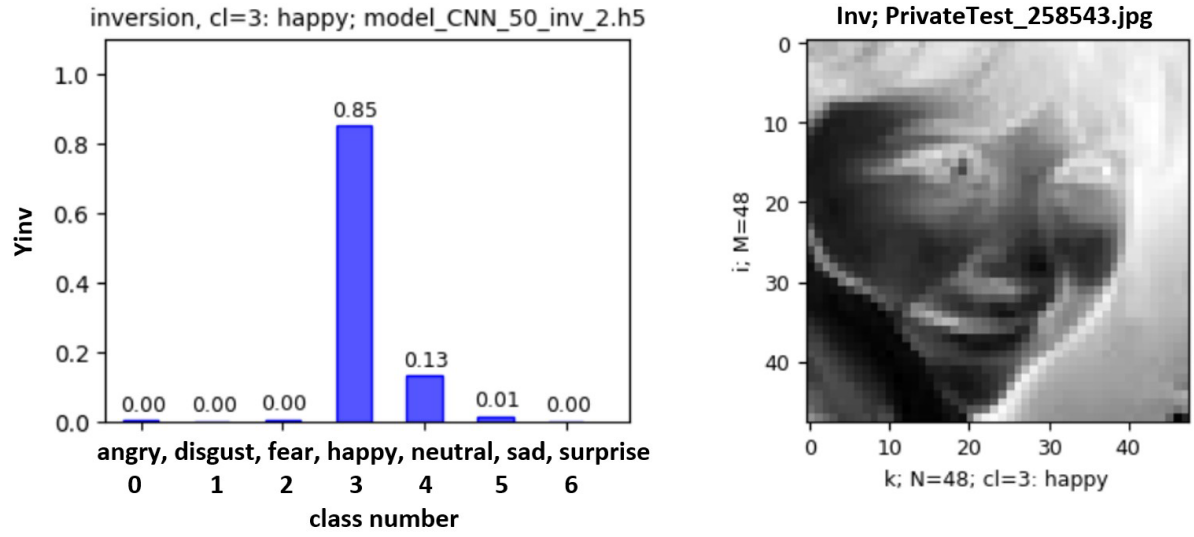

Figure 13: Results of emotion recognition on an inverted face image for CNN #3.

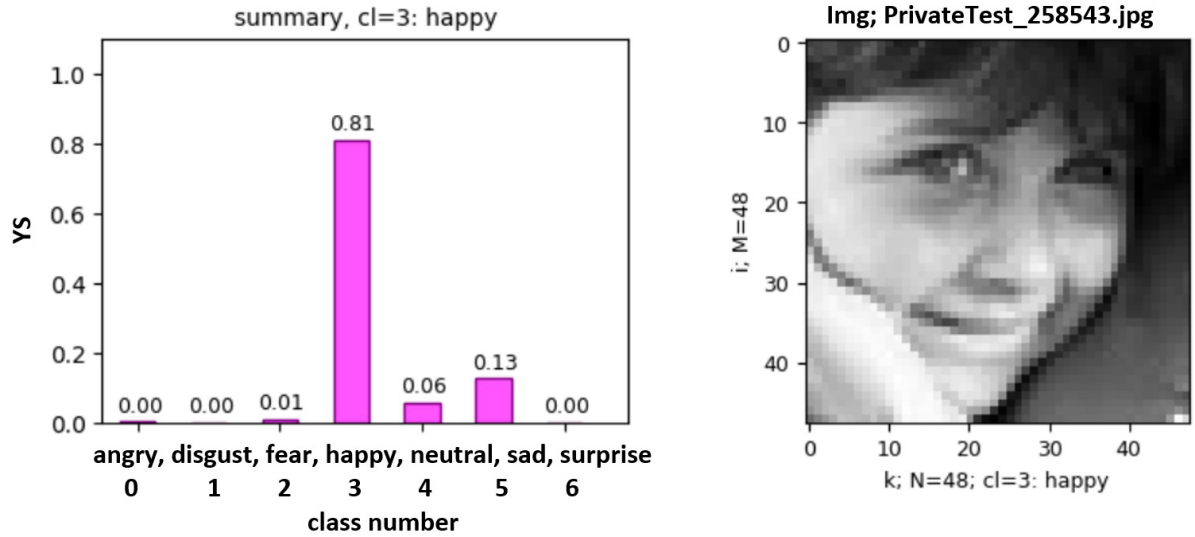

Figure 14: Results of emotion recognition on facial images by the ensemble of CNN #1-#3.

6. Discussion

In order to correctly compare the results when training the ensemble of CNN #1-#3, the same parameters were used as in the research [5]. In comparison with the CNN analogue (Fig. 15), similar results were obtained for the developed CNN #1 in terms of the val_loss and val_accuracy parameters (Fig. 4). However, when using the CNN ensemble, the recognition results are averaged, which allows for more accurate recognition of emotions in the image even when using CNN with the same structure. The effectiveness of the use of the CNN ensemble is explained by the fact that each of its CNN analyzes different image characteristics: CNN #1 analyzes the spatial distribution of brightness of the initial image, CNN #2 processes the contour image (analyzes areas of the image with a sharp change in brightness), and CNN #3 processes inverted images (analyzes predominantly facial features, which in the initial image have predominantly low brightness).

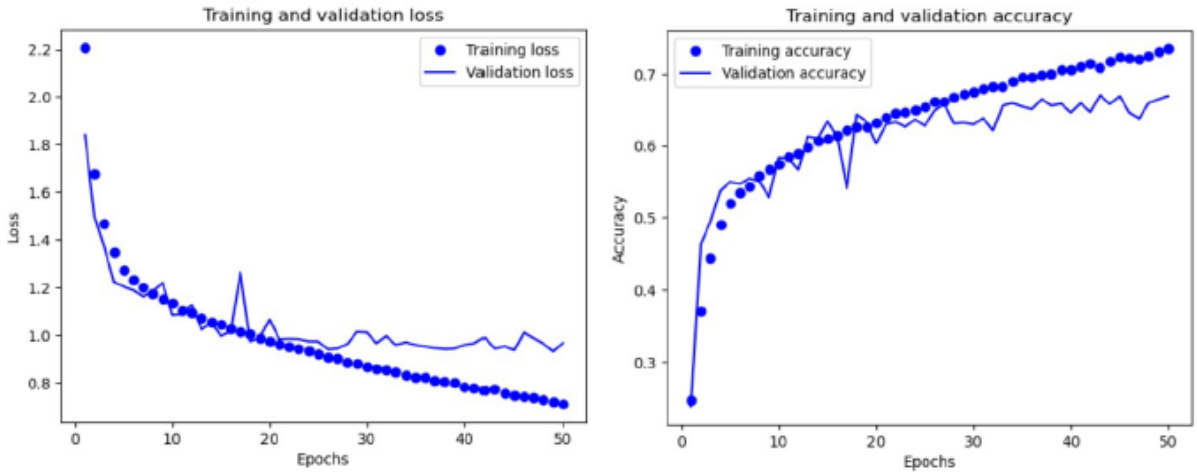

Figure 15: Graphs of the training Loss and Accuracy for the CNN-analogue [5], in the last epoch the training parameters: loss = 0.7126; accuracy = 0.7364; val_loss = 0.9681; val_accuracy = 0.6697.

The developed ensemble of CNN can operate on the basis of a computer or microcomputer hardware platform (e.g., Raspberry Pi), which is especially important in mobile and embedded systems. In the future, the training and validation datasets can be expanded by augmentation (image displacements in height and width, mirror images, changes in brightness and contrast) and the quantity of training epochs can be increased, which will significantly reduce the errors for both the training and validation datasets. The proposed approach to increasing the accuracy of facial emotion recognition by building an ensemble of CNN makes it possible to reduce emotion detection errors not only for the considered CNN with a relatively simple structure, but also for more complex architectures of artificial neural networks.

For example, the use of specialized neural network architectures (FERNet, EmotionNet), which are designed for recognizing emotions in images, is promising. For example, FERNet (facial emotion recognition neural networks) [22] allow for accurate identification of emotions in images even in the case of partial overlap of the face with other objects. However, the accuracy of the functioning of FERNet neural networks largely depends on the training conditions, in particular, on the used dataset and image augmentation. FERNet is often used in combination with other neural networks that perform face or other object detection in images [23].

It is also promising to use other image datasets (e.g., FER-2023 and “facial_expressions”) for training the CNN, in addition to FER-2013. The use of FER-2023 will increase the diversity of the training samples, and the “facial_expressions” dataset contains images with higher resolution. The use of such datasets will potentially increase the training time, but will allow achieving higher accuracy of emotion recognition due to better images detail. The use of facial emotion recognition systems can be effectively complemented by voice recognition systems [7]. Recognized facial emotions are used, in particular, to correct the operation of educational systems.

7. Conclusion

A method has been developed to improve the accuracy of facial emotion recognition, which consists in using an ensemble of three convolutional neural networks. The same structure for all CNN of the ensemble is used, but each CNN receives different input signals. The inputs of CNN #1 are fed with the initial image, the inputs of CNN #2 are fed with the contour image, and the inputs of CNN #3 are fed with the inverted initial image. The image contours are extracted using the Sobel method. At the outputs of the CNN, arrays of values are obtained that indicate the probability of recognizing a certain emotion on the face. The developed CNN are designed to recognize seven emotions: Angry, Disgust, Fear, Happy, Neutral, Sad, Surprise. The resulting values of the probability of recognizing emotions are calculated by averaging the outputs of all the CNN of the ensemble. The used convolutional neural networks contain 6 convolutional layers and 2 output fully connected layers. The software for facial emotion recognition was developed in Python in the notebook (web shell) Jupyter Notebook using the Google Colab cloud platform and using tensorflow and keras libraries.

In the process of training the CNN, parallelization of calculations was performed using Graphics Processing Units (GPU) NVIDIA T4, which allowed to reduce the CNN training time by an order of magnitude. To avoid overtraining the CNN, image augmentation was used. The training of the CNN was carried out on the FER-2013 dataset (dataset for facial emotion recognition), which contains grayscale images of 24400 faces and is divided into training dataset (22968 images) and validation dataset (1432 images). Emotion recognition on images of the validation dataset showed that the recognition error by the ensemble of convolutional neural networks is smaller compared to recognition by individual CNN.

The novelty of the work is the use of an ensemble of three CNN that process the initial images, their contours and inverted images, which allows to increase the accuracy of facial emotion recognition. The developed ensemble of CNN and software can be practically used, in particular, in e-learning systems for correcting the educational process depending on the emotional state of students. The proposed ensemble of convolutional neural networks can be improved by using CNN with a more complex structure and by increasing the dataset augmentation during training, which potentially allows increasing the accuracy of facial emotion recognition.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] F. Z. Canal, T. R. Müller, J. C. Matias, G. G. Scotton, A. R. Junior, E. Pozzebon, A. C. Sobieranski, A survey on facial emotion recognition techniques: A state-of-the-art literature review, Information Sciences 582 (2022) 593–617. doi:10.1016/j.ins.2021.10.005.

[2] Z. Y. Huang, C. C. Chiang, J. H. Chen, Y. C. Chen, H. L. Chung, Y. P. Cai, H. C. Hsu, A study on computer vision for facial emotion recognition, Sci. Rep. 13 (8425) (2023) 1–13. doi:10.1038/s41598-023-35446-4.

[3] S. Balovsyak, O. Derevyanchuk, V. Kovalchuk, H. Kravchenko, Y. Ushenko, Z. Hu, STEM project for vehicle image segmentation using fuzzy logic, IJMECS 16 (2) (2024) 45–57. doi:10.5815/ijmecs.2024.02.04.

[4] Zh. Hu, D. Uhryn, Yu. Ushenko, V. Korolenko, V. Lytvyn, V. Vysotska, System programming of a disease identification model based on medical images, in: Proceeding of the Sixteenth International Conference on Correlation Optics, COR '2023, Proceedings of SPIE, Bellingham, USA, 2024. doi:10.1117/12.3009245.

[5] M. Chahed, Human emotion detection, 2022. URL: https://www.kaggle.com/code/mohamedchahed/human-emotion-detection.

[6] Ch. Liang, J. Dong, A survey of deep learning-based facial expression recognition research, FCIS 5 (2) (2023) 56–60. doi:10.54097/fcis.v5i2.12445.

[7] L. Chyrun, V. Vysotska, S. Tchynetskyi, Yu. Ushenko, D. Uhryn, Information technology for sound analysis and recognition in the metropolis based on machine learning methods, IJISA 16 (6) (2024) 40–72. doi:10.5815/ijisa.2024.06.03.

[8] D. Suresha, H. N. Prakash, Natural Image Super Resolution through modified adaptive bilinear interpolation combined with contra harmonic mean and adaptive median filter, IJIGSP 8 (2) (2016) 1–8. doi:10.5815/ijigsp.2016.02.01.

[9] S. Balovsyak, O. Derevyanchuk, V. Kovalchuk, H. Kravchenko, M. Kozhokar, Face mask recognition by the Viola-Jones method using fuzzy logic, IJIGSP 16 (3) (2024) 39–51. doi:10.5815/ijigsp.2024.03.04.

[10] Y. Nan, J. Ju, Q. Hua, H. Zhang, B. Wang, A-MobileNet: An approach of facial expression recognition, Alex. Eng. J. 61 (2022) 4435–4444. doi:10.1016/j.aej.2021.09.066.

[11] D. Long, T. Tung, T. Dung, A facial expression recognition model using lightweight dense-connectivity neural networks for monitoring online learning activities, IJMECS 14 (6) (2022) 53–64. doi:10.5815/ijmecs.2022.06.05.

[12] A. Geron, Hands-on machine learning with Scikit-learn, Keras, and TensorFlow. O'Reilly Media, Inc., Sebastopol, California, 2019.

[13] Z. Li, F. Liu, W. Yang, S. Peng, J. Zhou, A survey of convolutional neural networks: Analysis, applications, and prospects, IEEE Transactions on Neural Networks and Learning Systems 33 (12) (2022) 6999–7019. doi:10.1109/TNNLS.2021.3084827.

[14] O. Berezsky, P. Liashchynskyi, O. Pitsun, P. Liashchynskyi, M. Berezkyy, Comparison of deep neural network learning algorithms for biomedical image processing, in: Proceedings of the 5th International Conference on Informatics & Data-Driven Medicine, IDDM ’2022, CEUR Workshop Proceedings, Aachen, Germany, 2022, pp. 135–145.

[15] H. Win, Ph. Khine, Z. Nway, Emotion recognition from faces using effective features extraction method, IJIGSP 13 (1) (2021) 50–57. doi:10.5815/ijigsp.2021.01.05.

[16] M. Keinert, S. Pistrosch, A. Mallol-Ragolta, B. Schuller, M. Berking, Facial emotion recognition of 16 distinct emotions from smartphone videos: Comparative study of machine learning and human performance, JMIR 27, (e68942) (2025) 1–17. doi:10.2196/68942.

[17] S. Minaee, M. Minaei, A. Abdolrashidi, Deep-emotion: facial expression recognition using attentional convolutional network, Sensors 21, 3046 (2021) 1–16. doi:10.3390/s21093046.

[18] P. N. R. Bodavarapu, P. Srinivas, An optimized neural network model for facial expression recognition over traditional deep neural networks, IJACSA 12 (7) (2021) 443–451. doi:10.14569/IJACSA.2021.0120751.

[19] R. R. Devaram, A. Cesta, LEMON: A lightweight facial emotion recognition system for assistive robotics based on dilated residual convolutional neural networks, Sensors 22 (3366) (2022) 1–20. doi:10.3390/s22239524.

[20] G. Guillen, Digital image processing with Python and OpenCV, in: G. Guillen (Ed.), Sensor projects with Raspberry Pi, Apress, Berkeley, CA, 2019, pp. 97–140. doi:10.1007/978-1-4842-5299-4_5.

[21] M. Sambare, FER-2013. Learn facial expressions from an image, 2020. URL: https://www.kaggle.com/datasets/msambare/fer2013/data.

[22] A. R. Prasad, A. Rajesh, Occlusion-aware FERNet: an optimized patch-based adaptive residual network with attention mechanism for occlusion-aware facial expression recognition, Soft Comput. 27 (2023) 16401–16427. doi:10.1007/s00500-023-09029-4.

[23] A. I. Pradana, Harsanto, B. B. M. Aboobaider, M. Harsanto, Intelligent surveillance for mask regulation in healthcare using the YOLOv11 algorithm, in: Proceeding of the 6th International Conference Health, Science And Technology, ICOHETECH ’2025, LPPM Universitas Duta Bangsa Surakarta, Surakarta, Indonesia, 2025, pp. 583-593. doi:10.47701/23mc9656.