Applied Mathematics and Computational Intelligence
Volume 11, No.2, Dec 2022 [505-520]

An Android Application for Facial Expression Recognition Using Deep Learning

Siti Haslini Ab Hamid [1], Naajihah Mustapa[2], Muhammad Firdaus Mustapha[3*]

[1] FH Training Center, 16800 Pasir Puteh, Kelantan Malaysia
[2] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Terengganu, 21080 Kuala Terengganu, Terengganu Malaysia
[3] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Kelantan, 18500 Machang, Kelantan Malaysia

[*] Corresponding author: mdfirdaus@uitm.edu.my

Received: 28 September 2022; Accepted: 7 November 2022; Available online (in press): 6 December 2022

ABSTRACT

Facial expression recognition (FER) is one of the most popular research interests in Computer Science and can be applied in many real time applications. Most of the FER systems apply machine learning techniques such as Convolutional Neural Network and Support Vector Machine. Although there are many researches on recognition of facial expression, it is still a challenging task for computer programs and also mobile applications. Therefore, this paper proposed an Android application for FER using deep learning technique that can recognize three facial expressions which are happy, angry and surprise. FER model has been trained by using Personal Image Classifier (PIC) based on facial expression dataset from Kaggle and achieved 95% accuracy. The trained model was later applied for Android application development in MIT App Inventor. The proposed Android application for FER has been successfully built, tested, and uploaded into Google Play Store.

Keywords: Deep learning, Facial Expression, Mobile Application, MobileNet, Personal Image Classifier

1 INTRODUCTION

Facial expression recognition (FER) is becoming one of the most popular research interests in Computer Science and can be applied in many real-world applications such as interactive game design and portable mobile application to automatically insert emotions in chat and assistance systems for autistic people [1]. Research on facial expression detection and recognition received a great deal of attention due to its essential uses in technology in many fields such as virtual reality, psychological studies, medical, and human-computer interaction [2], [3]. A facial expression is made up of one or more movement or facial muscle postures. One disputed theory claims that these movements reveal an individual's emotional condition to observers. Nonverbal communication can also take the shape of facial expressions [4] and it provides information about emotional state [5].

Many of the research in FER systems applied machine learning techniques. Machine learning is a subfield of artificial intelligence (AI) and computer science that focuses on using data and algorithms

to simulate how humans learn without being explicitly programmed and gradually increasing the accuracy of the system [6], [7]. Machine learning can be used to design and program explicit algorithms with high performance output in a variety of computing areas. For example, facial detection, fraud detection, and product recommendation.

The most popular machine learning techniques for FER are Convolutional Neural Network (CNN), Support Vector Machine (SVM) and k-Nearest Neighbors (kNN) because they can provide better recognition accuracy and better classification [8]. In addition, most of the research of FER systems follow the pattern recognition framework which consists of three phases: face detection, facial feature extraction and expression classification [9]. Although there are many researches on FER, it is still a challenging task for computer programs and also mobile applications. Therefore, deep learning technique which is a subset of machine learning was introduced and can be applied in FER because it can learn from vast amount of data to provide better result for FER.

This paper proposes an Android application for FER using deep learning technique and MIT App Inventor. The proposed application is aimed to provide an interactive application for various users from children to adult to practice making the correct facial expression in their communications. FER model was built and trained based on the facial expression dataset from Kaggle using Personal Image Classifier (PIC) which is applied deep learning technique. The model can recognize three facial expressions (happy, angry and surprise) that allow users to practice making a facial expression and play some fun and exciting games. The FER model has been integrated into MIT App Inventor for Android development. MIT App Inventor is an open source and online tool that provides intuitive mobile application development and uses block programming [10].

2 RELATED WORKS ON FACIAL EXPRESSION RECOGNITION

This section includes a literature review related to facial expression recognition (FER). A different approach of techniques was used to recognize facial expression. Table 1 summarized some of the previous researches on FER. For example, Jia et al. [4] found that CNN is able to improve the expression recognition effect and with the help of SVM, it is able to further improve the accuracy. Rahul et al. [11] proposed a hybrid approach for emotion recognition by combining CNN and Recurrent neural networks (RNN) for their research and three datasets was used to be tested, EMOTIC, FER-13 [12], and Facial expression Research Group (FERG) [13] database. FER-13 dataset achieved 94.08% of accuracy rate meanwhile EMOTIC dataset attained 72.64% and the lowest accuracy rate is 68.10% using the FERG dataset. Moreover, Abinaya et al. [2] proposed Hybrid Adaptive Kernel based Extreme Learning Machine (HAKELM) scheme on their research and achieved 95.5% of accuracy, 90.12% of sensitivity, and 95.1% specificity compared to the previous existing algorithm. Minaee et al. [14] has been experimenting on four different datasets by using their own proposed network which is end-to-end deep learning framework based on an attentional convolutional network. From their experiment, by using the FERG dataset, they were able to attain the highest accuracy rate of around 99.3% compared to using other datasets.

Futhermore, Kaviya and Arumugaprakash [15] demonstrated that the proposed CNN can learn facial expression. From the testing phase, the model test accuracy improved to 65% for FER-2103 dataset and 60% for custom datasets. Perveen et al. [5] have successfully achieved 90% of maximum accuracy by applying k-NN algorithm in the classification process. Among the six expressions, the surprise expression achieves a 100% accuracy compared to the other expressions. Meanwhile,

Sarode and Bhatia [16] applied 2D appearance-based local approach for the extraction of intransient facial features and recognition of four facial expressions from grayscale image and achieved 81% accuracy.

Table 1: Summary of researches on facial expression recognition

Author	Year Publication	Research Title	Technique / Tools	Type of Facial expression	Dataset
Jia et al. [4]	2022	Facial Expression Recognition Based on the Ensemble Learning of CNNs	CNN	Angry, disgust, fear, happy, sad, surprise, and neutral	FER2013
Rahul et al. [11]	2022	A New Hybrid Approach for Efficient Emotion Recognition using Deep Learning	CNN and Recurrent Neural Network (RNN)	Angry, disgust, fear, happy, neutral, sad, and surprise	EMOTIC, FER-13, FERG
Abinaya et al. [2]	2021	Classification of Facial Expression Recognition using Machine Learning Algorithms	HAKELM	Normal, happy, sad, surprise, angry and disgust	AT&T, YALE FACE B
Minaee et al. [14]	2021	Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network	End-to-end deep learning framework based on an attentional convolutional network	Angry, disgust, fear, happy, neutral, sad, and surprise	FER2013, CK+, JAFFE, FERG
Kaviya & Arumugapr akash [15]	2020	Group Facial Emotion Analysis System Using Convolutional Neural Network	CNN	Angry, happy, sad, surprise, and neutral	FER-2013, Custom Dataset
Perveen et al. [5]	2016	Facial Expression Recognition Through Machine Learning	SVM, Artificial Neural Network, and kNN	Happy, angry, sad, fear, disgust, and surprise	Real-time/ physical data
Sarode & Bhatia [16]	2010	Facial Expression Recognition	2D appearance-based local approach	Happy, sad, surprise and anger	YALE FACE B & JAFFE

Based on Table 1, some of the techniques applied by researchers to recognize facial expression are CNN, SVM and kNN. The popular facial expressions that can be detected in the previous research are happy, sad, angry and surprise. Many of the researches also can detect disgust, fear and neutral facial expressions. However, this research only trained three facial expressions into FER model which are happy, angry and surprise. Based on Angelica Perez [17], higher accuracy can be achieved when classifying a smaller subset of highly distinguishable expressions such as happy and angry. Sad expression is not included into FER model because it was found to be prone to error [18]. Moreover, most of the researches used online facial expression dataset instead of using real time physical facial expressions dataset for training and testing purposes such as FER2013, YALE FACE B, FERG and JAFFE.

Therefore, this paper proposes an Android application for FER that has two main features which are practicing facial expression and facial expression game by using new machine learning technique which is deep learning. Further explanation of this technique will be discussed in the next section.

3 METHODOLOGY

This section consists of information related to the methodology for this research which are data collection, hardware and software requirements, build facial expression recognition (FER) model, build Android application for FER and upload Android application for FER into Google Play Store.

3.1 Data Collection

The model used in this FER application was trained using a secondary data from the Kaggle website [19]. The micro-expression [20] dataset was chosen for data training and testing. Initially the collected dataset consists of angry, disgust, fear, happy, neutral, sad, and surprise. Due to maintain high accuracy for the FER model, only three facial expressions were selected which are happy, angry, and surprise. The number of images of each expressions exceeded 80 images with a pixel value, 80x80. However, a total of 150 clear and sharp images for three facial expressions were selected. Figure 1 shows some of the selected images from the dataset.

Figure 1: Dataset of facial expressions

3.2 Hardware And Software Requirements

To develop Android application for FER, it requires specific hardware and software. Hardware is used during training dataset to build FER model and also for Android application development. For hardware, this research uses laptop as depicted in Table 2. The low specification hardware will take a longer time to train the data even with a small number of images.

Table 2: Specification of the hardware

Operating System	Windows 10
Processor	Intel(R) Core i7-3517U
RAM	8.00 GB

For software, MIT App Inventor is used to develop the Android application for FER. MIT App Inventor is an intuitive, visual programming environment that allows everyone – even children – to build fully functional apps for Android phones, iPhones, and Android/iOS tablets [10]. It is easy and beginner-friendly software with a graphical user interface (GUI) which allows users to only drag and drop the component provided.

3.3 Build FER Model

To build FER model, this research use web tool which is Personal image classifier (PIC) [21]. PIC is a machine learning tool to build a model based on specific image dataset [22] and it is an extension of the MIT App Inventor. PIC makes use of MobileNet in Tensorflow.js as machine learning backend [23]. MobileNet is a CNN with a smaller model size, less trainable parameter and calculation amount, and suitable for mobile devices. It takes full advantage of its computing resources and improves model accuracy to the greatest extent [24]. The basic framework of MobileNet used for FER model is illustrated in Figure 2. PIC also employed transfer learning in MobileNet that can give a benefit to who would like to train a small set of datasets. The learning process will take a shorter time and will produce more accurate output [25].

In the FER model creation process, there are two main phases involve which are training and testing [26]. Based on hold out cross-validation [27], 60% from the total 150 images were used for training phase and the remaining 40% images were used for testing phase as depicted in Table 3. In training phase, label or type of facial expression for the classification needs to be created first. For this model, three labels were created which are happy, angry, and surprise. Next, facial expression image is needed for each label for training process. The facial expression images can be uploaded by file or captured directly from computer webcam. This research used file upload where 30 facial expression images uploaded into the specific labels (happy, angry and surprise) for training as shown in Table 3. Then, the uploaded images were automatically trained using MobileNet model.

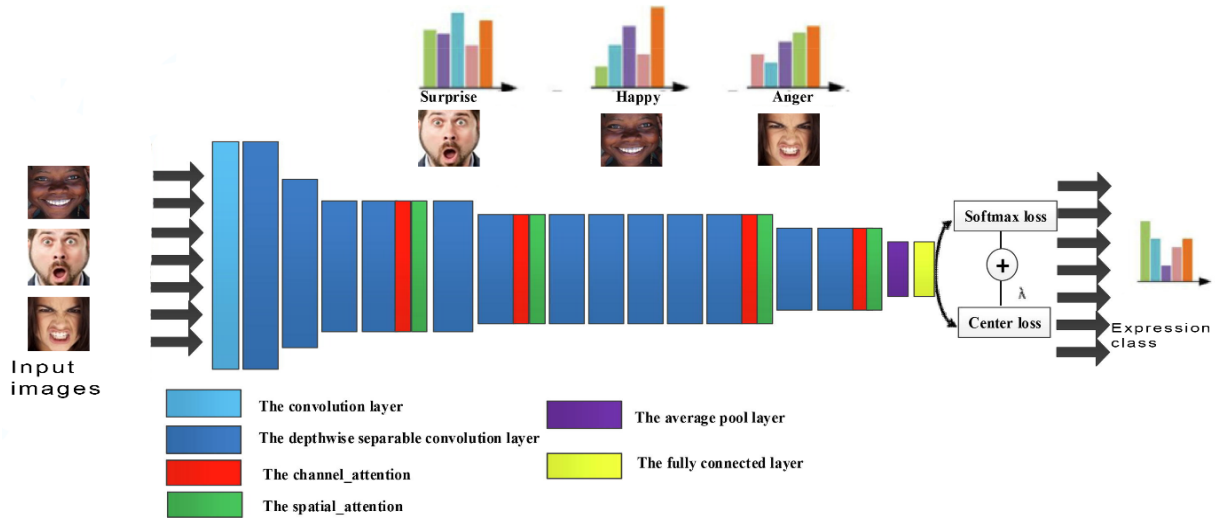


Figure 2: Basic framework of A- MobileNet network model

Table 3: Number of images used for training and testing

Facial expression	Training	Testing	Total
Happy	30	20	50
Angry	30	20	50
Surprise	30	20	50
	90 (60%)	60(40%)	150 (100%)

In testing phase, there is an option either to test FER model using live images which are captured using webcam or images from uploaded file. For this research, 20 images of each facial expression were uploaded for testing. During testing phase, the FER model that has been built will show the confidence level and the classification for each of the testing facial expression images. This phase can be reversed to training phase if the result is not satisfied. If the result is performant enough, the FER model can be downloaded as "model.mdl" and exported as an extension for the use in MIT App Inventor for Android application development [22]. The FER model was able to achieve 95% accuracy where 57 out of 60 testing images were classified with the correct expression.

3.4 Build Android Application for FER

3.4.1 Architecture for Application

Figure 3 shows the summary of architecture of Android application for FER that has been developed. The application has five visible components and four non-visible components. It uses six variables to save the specific data related to facial expression detection and games. It also uses four event handlers and a procedure or function for playing games.

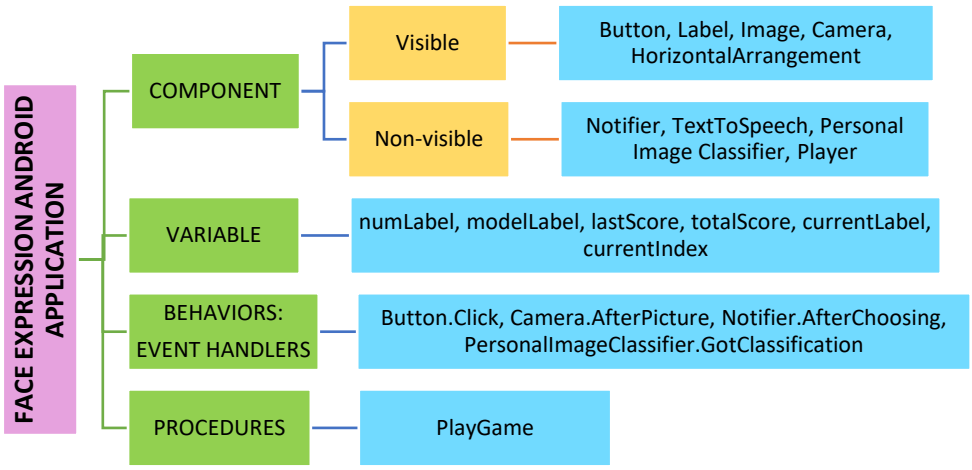
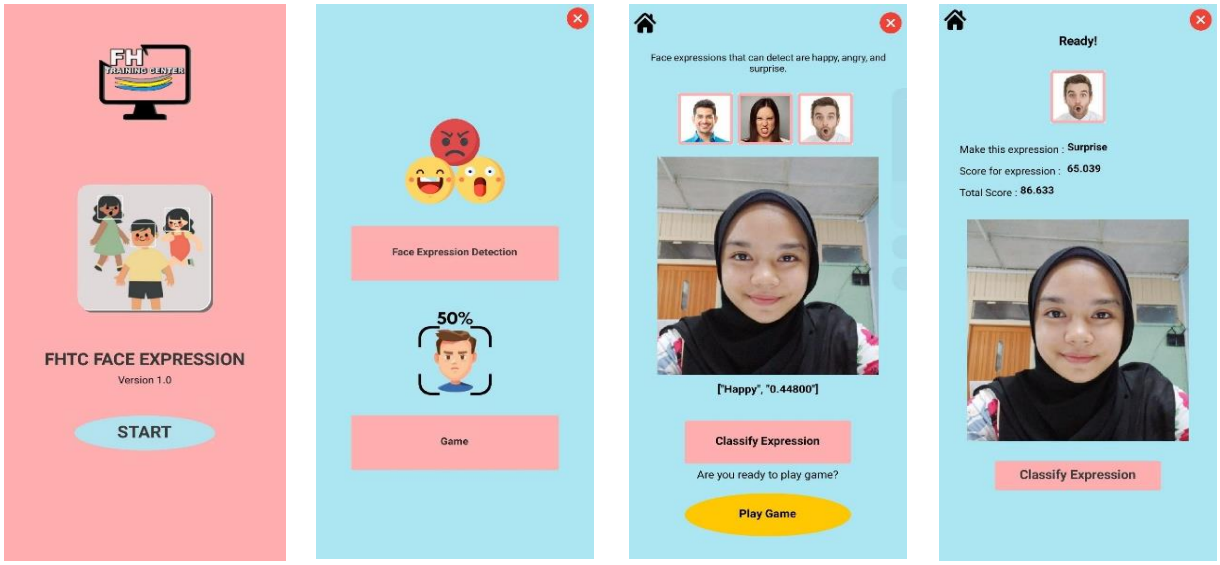


Figure 3: Architecture of facial expressions application

3.4.2 Interface Design for Application

The interface design for this application uses designer tab/ menu in MIT App Inventor. There are four main screens which are splash screen, main menu, face expression detection and face expression game as shown in Figure 4. The first screen of the application or known as Splash screen has a start button as shown in Figure 4(a). To start using the features in the application, user needs to click on the start button and it will open another screen or main menu. The main menu is the second screen which consists of a face expression detection button and a game button as shown in Figure 4(b).



(a) Splash screen (b) Main menu (c) Facial expressions detection (d) Game

Figure 4: Example of interface design

If the user clicks on the face expression detection button, it will open another screen as Figure 4(c). The purpose of this screen is to practice on facial expression before continuing to the face expression game. This screen can display image captured by the phone camera, type of facial expression (happy, angry, surprise) and value for confidence level. The range of the confidence level value is from 0 to 1 [28]. The value of 0 indicates the lowest confidence level or lowest accuracy meanwhile value 1 indicates the highest confidence level or highest accuracy. User needs to click on the Classify Expression button to capture an image of facial expression. When the image is captured, the application can detect and recognize the facial expression.

The fourth screen is a facial expression game as depicted in Figure 4(d). The purpose of this screen is to make three facial expressions start with angry, happy and surprise. The correct facial expression will get a high score while the wrong facial expression will get a low score. The calculation of score is summarized in Figure 5. Based on Figure 5, score for expression is calculated using confidence level value (0-1) got from FER model multiplied by 100. So, the range of score for expression is from 0 to 100. Value 0 means the lowest accuracy of facial expression while 100 is the highest accuracy of facial expression. If the score of expression is equal or greater than 40, the facial expression made by user is correct and the phone will say “This expression is correct”. Otherwise, if the score of expression is below 40, the facial expression is wrong and the phone will say “This expression is incorrect”. For total score calculation, if the total score of expression is equal or greater than 120, the user wins the game. In contrast, if total score of expression is lower than 120, the user loses the game.

Calculation on score for expression (SFE):
SFE = confidence level value x 100
If SFE is equal or greater than 40
    Phone say "This expression is correct."
Else
    Phone say "This expression is wrong."
EndIf

Calculation on total score (TS):
TS = current TS + SFE
If TS is equal or greater than 120
    Notifier print "CONGRATULATION! High expression score. Thanks for playing!"
Else
    Notifier print "TRY AGAIN! Low expression score. Thanks for playing!"
EndIf

Figure 5: Calculation of score for facial expression game

3.4.3 Program codes for Application

FER model that has been converted as a PIC extension is imported into the MIT App Inventor to create the function of facial expression recognition. Figure 6 displays a block coding from the PIC extension. It is shown that when the classifier is ready, the application will notify the user by showing text "Ready" and the phone will also say "Ready".

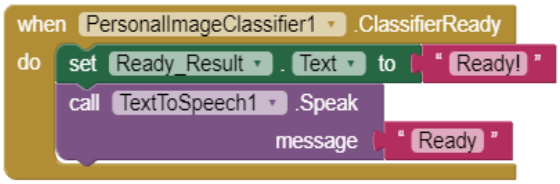


Figure 6: Block coding when classifier ready

Figure 7 shows the block coding when the classify expression button is clicked, it will call the camera to capture an image of facial expression and Figure 8 depicts the use of the PIC extension when it got the classification from the image captured by the user earlier.

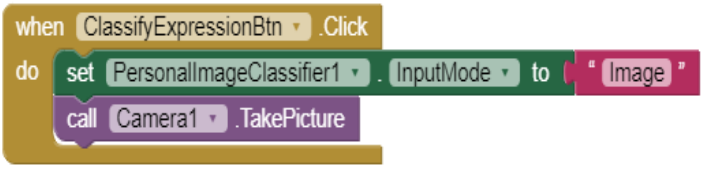


Figure 7: Block coding for classify expression button

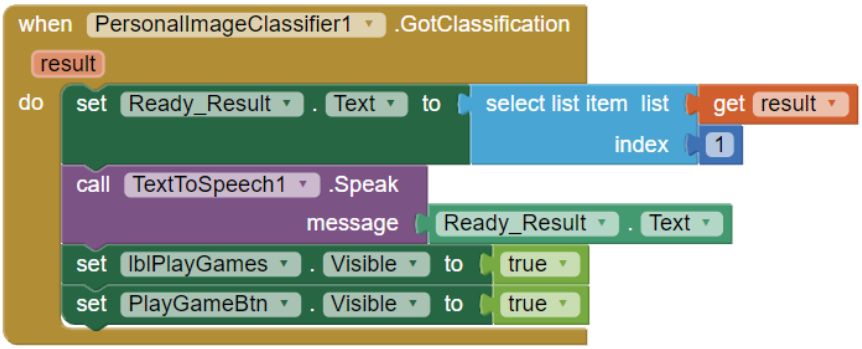


Figure 8: Block coding for facial expression recognition

3.4.4 Live Testing and Application Testing

During the development process, the application can be tested live using the AI companion where the laptop and the Android device (phone or tablet) need to be connected to the similar Wi-Fi. There are few limitations during the live testing that may cause the application not functioning well. For example, the function to exit or close an application cannot be tested during the live testing. So, it is important to do application testing before uploading into the Google Play Store. In order to do application testing, an Android application file (.apk) needs to be produced or built. For both type of testing, several factors have been considered such as phone sizes and phone model.

3.5 Upload Android Application for FER into Google Play Store

Android application for FER has been successfully built and uploaded into Google Play Store. It can be accessed and downloaded into Android phones and tablets by using the link below: https://play.google.com/store/apps/details?id=appinventor.ai_fhtrainingctr.FHTCFaceExpression [29].

4 RESULT AND DISCUSSION

The developed Android application for FER can detect and recognize three facial expressions captured by the camera which are happy, angry and surprise. This section discusses the example of result for two application screens which are facial expression detection screen and facial expression game screen.

4.1 Result for Facial Expression Detection Screen

This section discusses the example of three real facial expressions made by the user in the facial expression detection screen are shown in Figure 9. Based on the first image in Figure 9, the application can detect happy expression made by user with confidence level 0.71094. Based on the explanation of confidence level in section 3.4.2, 0.71094 is considered a high value of confidence level or high accuracy. In the second image, the application can detect angry expression with moderate confidence level 0.49048. For the last image, the application can also detect the surprise expression with the highest confidence level 0.85742.

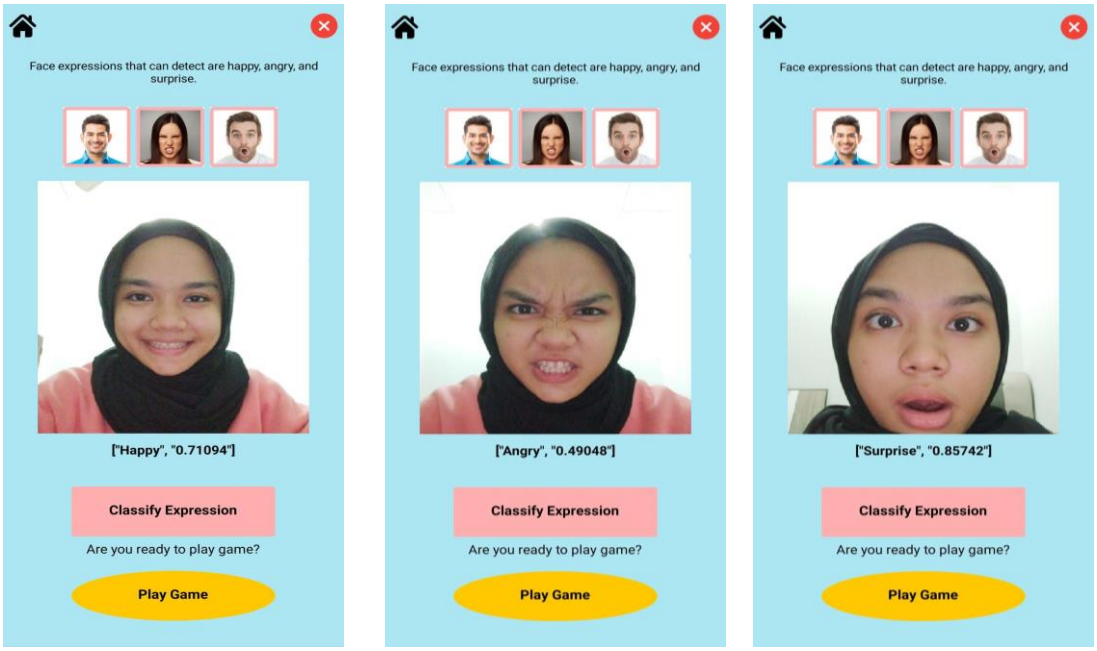


Figure 9: Result for three different type of face expressions

Based on the results obtained, surprise is the easiest facial expression for user to make among three expressions while angry expression is the most difficult for users to make. Some of the previous researches also reported that a substantial decrease in accuracy for angry [18]. There are some reasons that can cause low value of confidence level or low accuracy of result such as image quality and angle of camera when the user takes the image.

4.2 Result for Facial Expression Game Screen

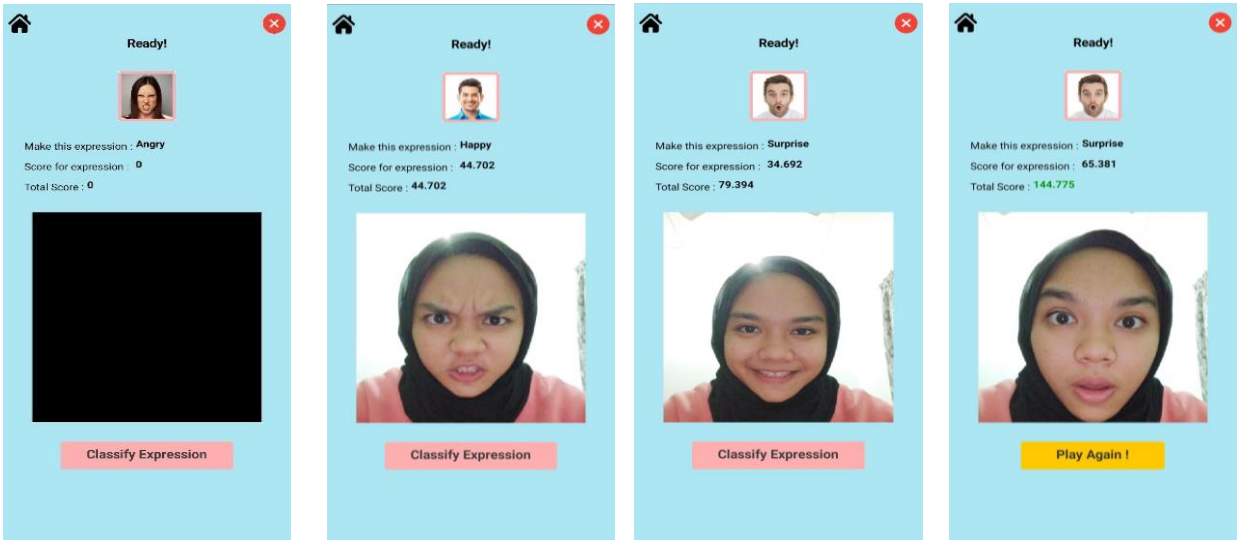
This section discusses the results for facial expression games played by the user. For more understanding, the screen has three main labels as below:

i) Make this expression – display the next face expression to be made by the user

ii) Score for expression – display current score for facial expression captured by camera

iii) Total score – display current total score for facial expression made by user

The example of detail results shown in Figure 10 and 11. Figure 10 displays the flow of the game and result while playing facial expression game. The order of facial expression starts with angry, happy and finally surprise. The calculation on score for expression and total score were discussed in section 3.4.2. Based on Figure 10, user made 2 correct or accurate expressions for angry (44.702) and surprise (65.381) because the scores are above 40. However, user made inaccurate facial expression for happy (34.692) because the score is less than 40. Based on the results obtained for facial expression game, surprise is also the easiest facial expression for user to make compared to other expressions.



(a) Initial interface (b) Result for angry face (c) Result for happy face (d) Result for surprise face

Figure 10: Example of result in face expression game screen while playing game

After completing the facial expression game, the final result will be displayed as shown in Figure 11. The total scores for the three real facial expressions made by the user will be calculated in order to identify whether the user won or lost the game. Based on Figure 11, the left image is example of result when user lost the game with total score is 99.127 which is below 120. On the other hand, the right image shows an example of result when user won the game with total score is 138.672.

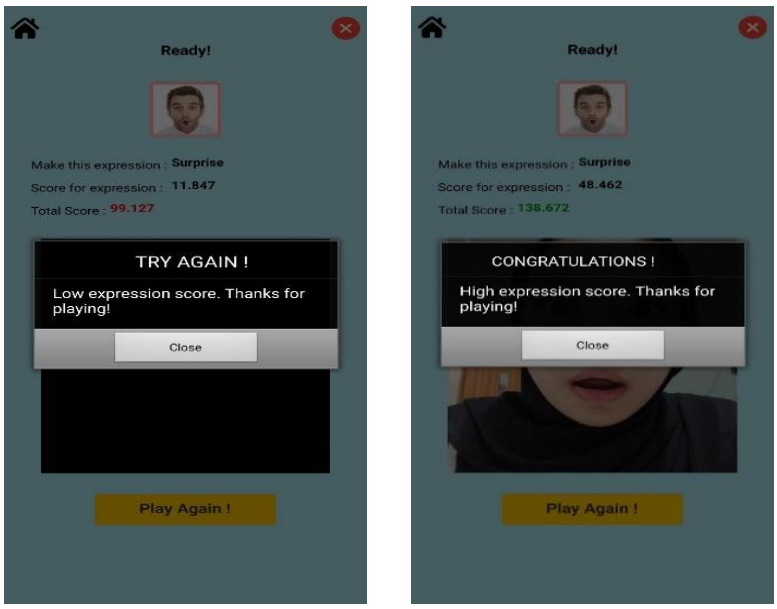


Figure 11: Example of result for completed face expression game

5 CONCLUSION

This research has successfully developed an interactive Android application for facial expression recognition using deep learning technique that is suitable for various users to practice making the correct facial expression in their communication. The result shows that surprise expression is easily made by user with higher confidence level value and score compared to angry and happy expressions. The proposed application has several main features such as:

i) Recognize three facial expressions (happy, angry, surprise)

ii) Display confidence level value as a benchmark for facial expression recognition accuracy

iii) Allow the camera to be front or rear

iv) Provide text and facial expression image as a guideline for user

v) Can be used and played anytime and anywhere via offline

Even though the android application has successfully developed and uploaded into Google Play Store, it has limitation on number of facial expressions to be recognized. It can only recognize three facial expressions which are happy, angry and surprise. Therefore, future work can be done to improve the android application by increasing the number of facial expressions to be recognized such as four popular facial expressions (happy, sad, angry and surprise). The accuracy of the facial recognition model can be improved by increasing the number of train images for each facial expression. Other than that, by adding more types of game that related to the facial expression recognition, it can make the application more fun and valuable.

ACKNOWLEDGEMENT

This work was supported by the Fundamental Research Grant Scheme (FRGS) under the Ministry of Education (MOE) with grant number 600-IRMI/FRGS 5/3 (234/2019).

REFERENCES

[1] L. Zhang and D. Tjondronegoro, "Facial expression recognition using facial movement features," *IEEE Trans. Affect. Comput.*, vol. 2, no. 4, pp. 219–229, 2011, doi: 10.1109/T-AFFC.2011.13.

[2] D. Abinaya, C. Priyanka, M. Rocky Stefinjain, G. K. D. Prasanna Venkatesan, and S. Kamalraj, "Classification of Facial Expression Recognition using Machine Learning Algorithms," *J. Phys. Conf. Ser.*, vol. 1937, no. 1, 2021, doi: 10.1088/1742-6596/1937/1/012001.

[3] S. Ullah, A. Jan, and G. M. Khan, "Facial Expression Recognition Using Machine Learning Techniques," in *7th Int. Conf. Eng. Emerg. Technol. ICEET 2021*, 2021, pp. 27–28, doi: 10.1109/ICEET53442.2021.9659631.

[4] C. Jia, C. L. Li, and Z. Ying, "Facial expression recognition based on the ensemble learning of CNNs," in *ICSPCC 2020 - IEEE Int. Conf. Signal Process. Commun. Comput. Proc.*, 2020, pp. 0–4, doi: 10.1109/ICSPCC50002.2020.9259543.

[5] N. Perveen, N. Ahmad, M. A. Qadoos, B. Khan, R. Khalid, and S. Qadri, "Facial Expression Recognition Through Machine Learning," *Int. J. Sci. Technol. Res.*, vol. 5, no. 3, pp. 91–97, 2016.

[6] J. Alzubi, A. Nayyar, and A. Kumar, "Machine Learning from Theory to Algorithms: An Overview," *J. Phys. Conf. Ser.*, vol. 1142, no. 1, 2018, doi: 10.1088/1742-6596/1142/1/012012.

[7] M. Batta, "Machine Learning Algorithms - A Review," *Int. J. Sci. Res. (IJ*, vol. 9, no. 1, pp. 381–386, 2020, doi: 10.21275/ART20203995.

[8] D. Mehta, M. F. H. Siddiqui, and A. Y. Javaid, "Recognition of emotion intensities using machine learning algorithms: A comparative study," *Sensors (Switzerland)*, vol. 19, no. 8, pp. 1–24, 2019, doi: 10.3390/s19081897.

[9] S. Rajan, P. Chenniappan, S. Devaraj, and N. Madian, "Facial expression recognition techniques: A comprehensive survey," *IET Image Process.*, vol. 13, no. 7, pp. 1031–1040, 2019, doi: 10.1049/iet-ipr.2018.6647.

[10] "About us," *International Cooperative Alliance*, *ICA*, 2020. [Online]. Available: https://www.ica.coop/en/cooperatives/facts-and-figures. [Accessed July 27 , 2022].

[11] M. Rahul, N. Tiwari, R. Shukla, D. Tyagi, and V. Yadav, "A New Hybrid Approach for Efficient Emotion Recognition using Deep Learning," *Int. J. Electr. Electron. Res.*, vol. 10, no. 1, pp. 18–22, 2022, doi: 10.37391/IJEER.100103.

[12] M. Sambare, "FER-2013 | Kaggle," *Kaggle*. 2013. [Online]. Available: https://www.kaggle.com/msambare/fer2013. [Accessed July 27, 2022].

[13] "TuSimple - V7 Open Datasets." [Online]. Available: https://www.v7labs.com/open-datasets/tusimple. [Accessed July 27, 2022].

[14] S. Minaee, M. Minaei, and A. Abdolrashidi, "Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network," *Sensors*, vol. 21, no. 9, p. 3046, Apr. 2021, doi: 10.3390/s21093046.

[15] P. Kaviya and T. Arumugaprakash, "Group Facial Emotion Analysis System Using Convolutional Neural Network," in *Proc. 4th Int. Conf. Trends Electron. Informatics, ICOEI 2020*, 2020, pp. 643–647, doi: 10.1109/ICOEI48184.2020.9143037.

[16] K. R. Kulkarni and S. B. Bagal, "Facial expression recognition," in *Proc. - IEEE Int. Conf. Inf. Process. ICIP 2015*, 2015, pp. 535–539, 2016, doi: 10.1109/INFOP.2015.7489442.

[17] "Recognizing Human Facial Expressions With Machine Learning | ThoughtWorks Arts." [Online]. Available: https://thoughtworksarts.io/blog/recognizing-facial-expressions-machine-learning/. [Accessed August 20, 2022].

[18] T. Küntzler, T. T. A. Höfling, and G. W. Alpers, "Automatic Facial Expression Recognition in Standardized and Non-standardized Emotional Expressions," *Front. Psychol.*, vol. 12, pp. 1–13, 2021, doi: 10.3389/fpsyg.2021.627561.

[19] "Find Open Datasets and Machine Learning Projects | Kaggle," *Kaggle*, 2020. [Online]. Available: https://www.kaggle.com/datasets. [Accessed April 4, 2022].

[20] K. M. Irfan, "Micro_expressions," *Kaggle*, 25-Apr-2022. [Online]. Available: https://www.kaggle.com/datasets/kmirfan/micro-expressions. [Accessed April 4, 2022].

[21] "Personal image classifier," *Personal Image Classifier*. [Online]. Available: https://classifier.appinventor.mit.edu/. [Accessed April 7, 2022].

[22] J. D. Rodríguez-García, J. Moreno-León, M. Román-González, and G. Robles, "LearningML: A tool to foster computational thinking skills through practical artificial intelligence projects," *Rev. Educ. a Distancia*, vol. 20, no. 63, 2020, doi: 10.6018/RED.410121.

[23] D. Tang, "Empowering novices to understand and use machine learning with personalized image classification models, intuitive analysis tools, and MIT App Inventor," M. Eng. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2019, [Online]. Available: https://hdl.handle.net/1721.1/123130.

[24] Y. Nan, J. Ju, Q. Hua, H. Zhang, and B. Wang, "A-MobileNet: An approach of facial expression recognition," *Alexandria Eng. J.*, vol. 61, no. 6, pp. 4435–4444, 2022, doi: 10.1016/j.aej.2021.09.066.

[25] "What Is Transfer Learning? [Examples & Newbie-Friendly Guide]." [Online]. Available: https://www.v7labs.com/blog/transfer-learning-guide. [Accessed August 20, 2022].

[26] P. Refaeilzadeh, L. Tang, H. Liu, L. Angeles, and C. D. Scientist, "Encyclopedia of Database Systems," *Encycl. Database Syst.*, 2020, doi: 10.1007/978-1-4899-7993-3.

[27] Lakshana, "Cross-Validation Techniques in Machine Learning for Better Model," *Analytics Vidhya*. 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/4-ways-to-evaluate-your-machine-learning-model-cross-validation-techniques-with-python-code/. [Accessed August 21, 2022].

[28] J. Grandperrin, "How to use confidence scores in machine learning models," *Towards Data Science*, 2021. [Online]. Available: https://towardsdatascience.com/how-to-use-confidence-scores-in-machine-learning-models-abe9773306fa. [Accessed August 21, 2022].

[29] "FHTC face expression - apps on Google Play," *Google*. [Online]. Available: https://play.google.com/store/apps/details?id=appinventor.ai_fhtrainingctr.FHTCFaceExpression. [Accessed April 20, 2022].