ASIC-Based Facial Emotion Recognition System for Human-Computer Interaction

Khoo Wing Jian[1], Lee Mun Feng[1], Nabihah Ahmad[1,2,*], Chessda Uttraphan[1,2], Sundararajan Ananiah Durai[3], Warsuzarina Mat Jubadi[1,2]

[1] Department of Electronics Engineering, Faculty of Electrical and Electronics Engineering, Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400 Batu Pahat, Johor, Malaysia
[2] VLSI and Embedded Technology (VEST) Focus Group, Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400 Batu Pahat, Johor, Malaysia
[3] School of Electronics Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu 600127, India

ABSTRACT

Keywords:

Artificial intelligence (AI); application specific integrated circuit; convolution neutral network (CNN); deep learning; facial emotion recognition system (FER)

In the era of technology, Artificial Intelligence (AI) plays a vital role in human daily life, especially AI machines. But there are some limitations when AI machines communicate with the humans such as recognition accuracy and recognition speed. In this study, the Facial Emotion Recognition (FER) System that used Convolution Neural Network (CNN) technique is developed using Application Specific Integrated Circuit (ASIC) implementation. The free online software – Google Colab is used to train the CNN deep learning model and generate the weight and bias values that are used during the designing of the CNN model in Verilog Hardware Description Language (HDL). The facial emotion expression images undergo the trained CNN deep learning model to classify into seven basic universal emotions such as happy, sad, angry, disgust, surprise, fear and neutral. Functional verification, logic synthesis and physical synthesis are carried out using the Electronic Design Automation (EDA) tool - Synopsys with 32nm technology. The final layout of the FER system had an area of $3101.20\ \mu m^2$ with a total power consumption of $871.05\ \mu W$. This system is achieving 92% of recognition accuracy and used $133767ns$ to recognize an image and classify it according to the emotion class. The proposed design with a compact size and low power consumption can be beneficial for various automation applications such as human-computer interaction systems.

1. Introduction

Emotion plays a vital role in the interaction and communication between human beings in daily life. Emotion as a non-verbal communication helps human beings to understand one another during verbal communication. There are seven common emotions such as anger, disgust, fear, happiness, sadness, surprise and neutral which were introduced by Ozdemir *et al.,* [1]. In the era of technology,

emotion is very important when humans need to communicate with the computer in the form of digitalized.

Human beings can understand emotion more simply however emotion detection is a difficult task for a computer that does not have a brain like human beings. The computerized FER system that used machine learning was proposed by many researchers. Machine learning is a branch of Artificial intelligence (AI) that enables computers to learn from data and make decisions with little human involvement as proposed by Sanjana *et al.,* [2]. Recently, FER system has been applied in various sectors like the security sector, healthcare sector, transportation sector and others. Ulusoy *et al.,* [3] concluded that FER system could be used to find people who had a probability of having bipolar disorder because of their genetics. Ajay *et al.*, [4] proposed that the FER system could be further improved to become a driver drowsiness system to detect the driver's condition by tracking the facial emotion expressions.

Krizhevsky *et al.*, [5] introduced the Convolution neural network (CNN) as one of the popular deep learning models in machine learning that performed well in some fields. Xiao *et al.*, [6] gave the examples like image classification, recognition, detection and segmentation. Wong *et al.*, [7] presented that the CNN model builds by multi-layers, which consist of convolution layers, subsampling layers and fully connected layers. Choudhari *et al.,* [8] said each layer is designated to extract different features from the images. Face recognition that used CNN technique had been proposed by researchers Yan *et al.,* [9]; Coşkun *et al.,* [10]; Ding *et al.,* [11] and Zhao *et al.,* [12]. However, the CNN model has many parameters and computations which occupy many resources and a large amount of memory for the application. Cheang *et al.,* [13] gave some examples of the work and research about the FER system that is managed by software but Wong *et al.*, [7] found out that the implementation of CNN in software is a slow process. Considering this problem, some methods had been proposed in previous research to develop the FER system using hardware implementation. The FPGA-based lightweight convolutional neural network (CNN) accelerator has been proposed by Kim *et al.,* [14] and used the quantization method to reduce the size of hardware resources occupied by the model. The FER system has been introduced with the help of high-level language by Khan *et al.,* [15]; Phan-Xuan *et al.,* [16] and Qiao *et al.,* [17]. CNN model has been developed and trained to generate the weight and bias for further use in FPGA implementation.

In this project, a FER system via ASIC implementation is presented to further improve the development of the FER system by using the trained CNN model. The generated weight and bias from the trained CNN model are used during the development of the system using ASIC implementation. This system can be used to recognize the input facial emotion expression image and classified according to the emotions class. Recognition accuracy and recognition speed of the proposed system are recorded. The design parameters such as timing, total area and power consumption has also been analysed in this project.

2. Methodology

Figure 1 shows the operation flow of the designed system which recognizes facial emotion expressions. The FER system consists of several distinct phase:

    i.   Training the CNN model and obtaining weight and bias values by utilizing Google Colab and capturing a photo
    ii.   Providing facial expressions as input
    iii. Recognition and interpreting the facial expressions
    iv. Producing the output result.

In the initial phase, the CNN model was trained using the FER 2013 database, which served as input and was implemented in Google Colab. In this study, the source code of Facial Emotion Recognition using the CNN technique is taken from researchers [18,19]. Python code is written using the free online software – Google Colab to develop and train the CNN model using the FER-2013 database. This is because it is very complicated to directly train the CNN model using the Verilog HDL module as the training process requires a lot of hardware resources. On the other hand, the FER 2013 database contained a total of 35,887 28,709 expression image samples, encompassing happy, anger, sad, disgust, surprise, fear and neutral while the 28,709 image samples were picked to training process and 7,178 were reserved for validation process. Each grayscale image in the dataset had dimensions of 48x48 pixels.

Moving to the second phase, a proposed method was implemented to perform facial emotion recognition using a CNN architecture. This architecture incorporated Convolutional Layers and Max Pooling Layers as essential components, enabling the extraction of features from input images for the development of the CNN model used in expression recognition and classification.

During the third phase, the system became capable of detecting and classifying human facial expressions. Finally, in the fourth phase, the facial expression images obtained during the testing process were employed by the system to determine the corresponding human emotion.
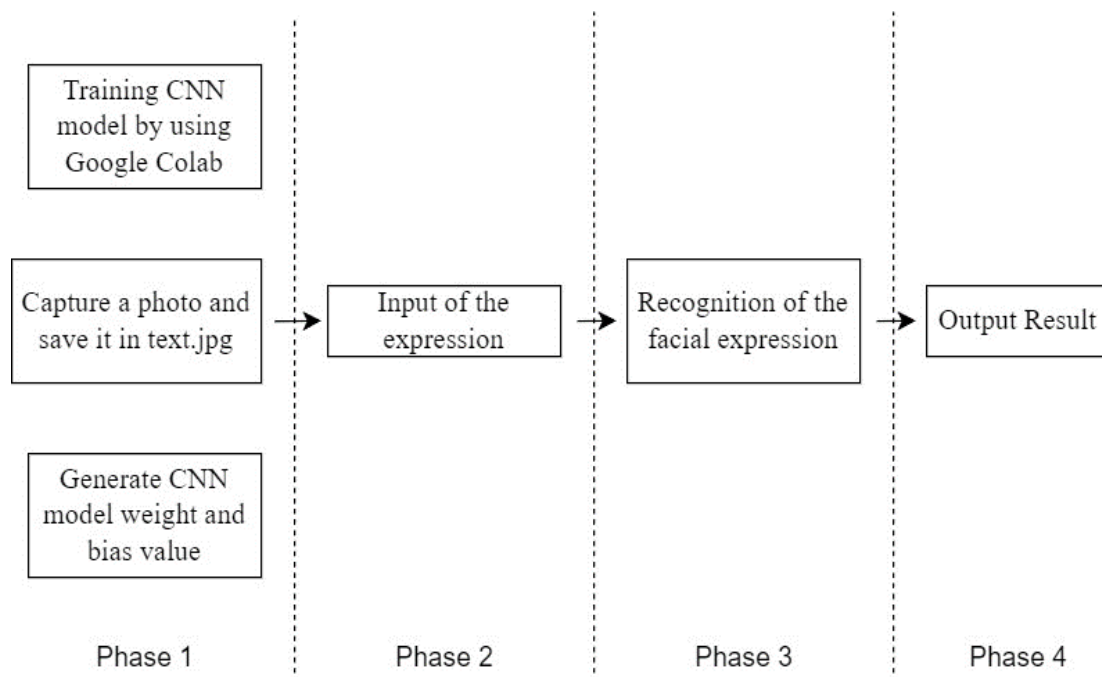

Fig. 1. Operation flow of facial emotion recognition system

Other than that, the pixel values from captured image will be extracted and stored in the text file for the input of the Verilog HDL module to check the functionality of the Verilog HDL module. This is because Verilog HDL code cannot read images directly.

The CNN model used in this study consists of 4 convolution layers, 3 max-pooling layers with relu activation, 2 fully connected layer and 1 comparator. Each layer of output shape and trainable parameters are provided in Table 1. The facial emotion recognition model is a sequential model with a total of 13 layers. It starts with two convolutional layers, followed by three max-pooling layers and a dropout layer. Then, there are two more convolutional layers and another max-pooling layer. A second dropout layer is applied before the output is flattened and passed through two fully connected layers. The model has a total of 2,345,607 trainable parameters. The convolutional layers

extract features from the input images, while the max-pooling layers reduce spatial dimensions. The fully connected layers transform the flattened output into the final prediction. Dropout layers are used for regularization to prevent overfitting. The model architecture suggests a deep learning approach to capture facial emotion features and achieve accurate emotion recognition.

Table 1
Summaries of each layer of CNN model

Layer	Output Shape	Trainable parameter
Conv2d	(46,46,32)	320
Conv2d_1	(44,44,64)	18496
Max_pooling2d	(22,22,64)	0
dropout	(22,22,64)	0
Conv2d_2	(20,20,128)	73856
Max_pooling2d_1	(10,10,128)	0
Conv2d_3	(8,8,128)	147583
Max_pooling2d_2	(4,4,128)	0
Dropout_1	(4,4,128)	0
Flatten	(2048)	0
Dense	(1024)	2098176
Droput_2	(1024)	0
Dense_1	(7)	7175

Figure 2 shows the project design flow. Verilog HDL modules are designed and configured according to the structure of the CNN model. The weight and bias extracted from the trained CNN model in Google Colab that are stored in the text files are used during the development of the CNN model in the Verilog HDL module. Testbench is prepared to test the functionality of the design system. This process is called functional verification and is carried out using the Synopsys VCS.

When all of the functionalities meet, the Synopsys Design Compiler was used and load the Top-level module. It is used to compile the Register Transfer Level (RTL) logic design with design constraints such as clock period, input and output constraints. Timing, area and power are analysed and reported in this stage. The gate-level netlist is also generated and stored in ddc.format.

After that, the gate-level netlist file is loaded into Synopsys IC Compiler. A floorplan is created and then specify the die area, pin arrangement and power network. The placement of the cells within the die area has also been optimized. The Clock Tree Synthesis (CTS) is used to create a buffer tree for the clock network, which allows a single clock signal to drive several flip-flops without weakening the signal. The final step in the physical design process is to route the design. The design layout was performed and verified using Design Rule Check (DRC) and Layout Versus Schematic (LVS). The command verify_zrt_route is used to check any design rule violations (DRC) while the command verify_lvs is used to ensure that there are no shorts or open nets in the design layout. The overall status report was generated including timing, area and power.
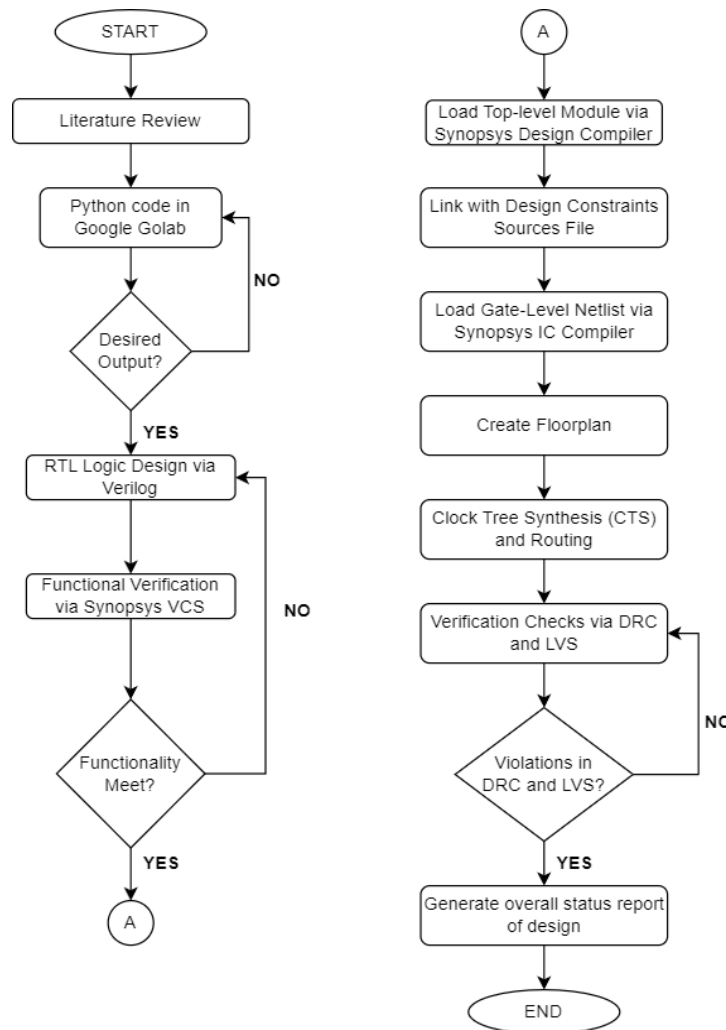
Fig. 2. Project design methodology

3. Results

3.1 Training of CNN Model using Google Colab

Figure 3 shows the simulation result from the Google Colab coding. The coding constructs the CNN model to train and test the FER system. The higher accuracy of the trained CNN model is 92% after training 70 epochs.

```
Epoch 68/70
448/448 [==============================] - 15s 33ms/step - loss: 0.2296 - accuracy: 0.9180 - val_loss: 1.3924 - val_accuracy: 0.6251
Epoch 69/70
448/448 [==============================] - 15s 34ms/step - loss: 0.2272 - accuracy: 0.9194 - val_loss: 1.3985 - val_accuracy: 0.6247
Epoch 70/70
448/448 [==============================] - 16s 35ms/step - loss: 0.2238 - accuracy: 0.9210 - val_loss: 1.4044 - val_accuracy: 0.6258
```

Fig. 3. Recognition accuracy of trained CNN model

To verify the effectiveness of the CNN model in recognizing human facial emotions, a system was developed to capture photos using the webcam within the Google Colab environment. The implementation involved a combination of JavaScript and Python code snippets. Through the utilization of JavaScript, Python and Google Colab integration, a function named 'take_photo()' was

created, enabling users to capture a photo, save it as a JPEG image file and obtain the corresponding filename. The code interacts with JavaScript in the Colab notebook through the utilization of functions such as 'display()' and 'eval_js()', while the 'base64' module is employed to decode and store the captured image data. The process of capturing a photo and presenting the resulting emotion through a bar chart is visually depicted in Figure 4.
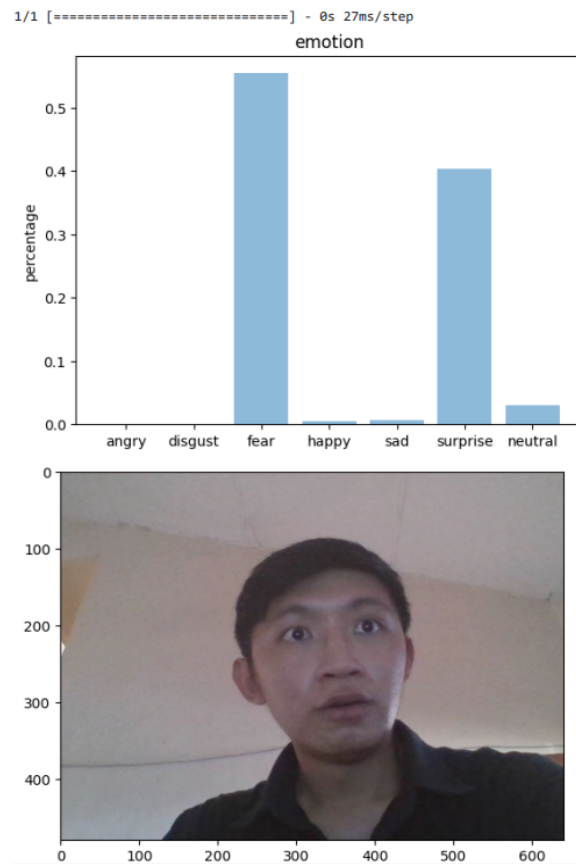
Fig. 4. Capture photo and the output emotion
represent in bar chart

Once the photo is captured, the model is stored and the weights and biases from the TensorFlow-trained CNN model are exported. The code then loads a pre-existing model and proceeds to iterate through its layers, extracting the corresponding weights and biases. Subsequently, these weights and biases are reshaped and individually saved as text files for each layer. This methodology ensures the preservation and potential reuse of the acquired parameters, facilitating additional analysis, transfer learning or implementation of the trained model.

3.2 Functional Verification using Synopsys VCS

Figure 5 shows the output waveforms for the top-level module in a clock period of 10ns after functional verification using Synopsys VCS. The input data has undergone the CNN operation and has been classified into the emotion class according to the facial emotion expression of the input image. In $133767ns$, the finish signal shows 1 which indicates that the result of the 'emotion_out' is valid and represents the emotion class of the input image.
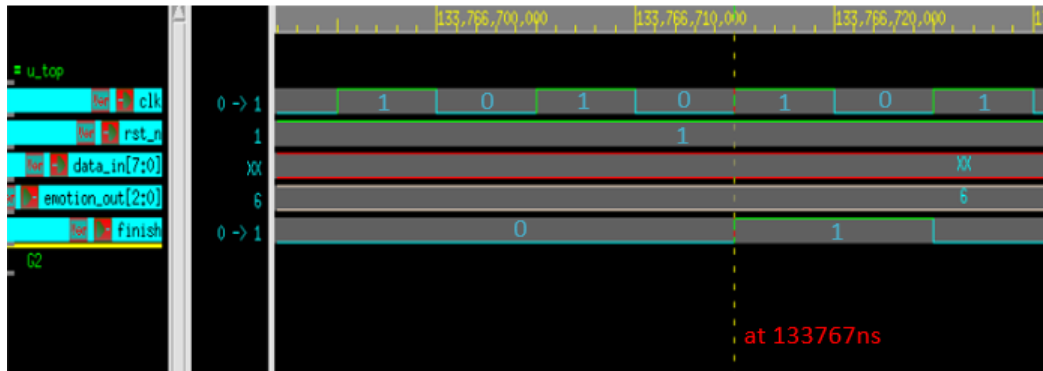
Fig. 5. Simulation waveform of the output layer

According to Table 2, the output of 'emotion_out' displays the hexadecimal number 6, which represents the detected emotion class 6, specifically the fear expression.

Table 2
Truth table of output waveform

Expression	No represents (Score)	No. bit for waveform
Angry	0	000
Disgust	1	001
Neutral	2	010
Happy	3	011
Sad	4	100
Surprise	5	101
Fear	6	110

3.3 Post-Place and Route

The post-place and route results were generated in Synopsys IC Compiler to create the floorplan, carry out CTS and routing for the chip design. Table 3 shows the timing, area and power analysis obtained after the physical design using IC Compiler. The total power consumption was $871.05\mu W$. The total cell leakage power and total dynamic power in physical synthesis are $570.57\mu W$ and $300.48\mu W$ respectively. The power consumption of the ASIC-based human facial recognition system is directly affected by the clock frequency at which it operates. A higher clock frequency leads to increased switching activities, resulting in higher dynamic power consumption. In the case of a clock frequency of 100 MHz, corresponding to a clock period of 10 ns in the system, the faster processing and response times offered by higher frequencies come at the cost of increased power consumption. The final chip area is $3101.20\mu m^2$. The timing analysis also shows that no timing violation occurs in this stage.

Table 3
Summaries results of timing area and power analysis in IC compiler

Design Matrix	Physical Synthesis
Timing Slack (ns)	0.53
Clock Frequency (MHz)	100
Total Area (μm^2)	3101.20
Total Power Consumption (μW)	871.05

Figure 6 shows the final layout for the chip design. The resulting layout was then verified through DRC and LVS to ensure that it do not have any violations.
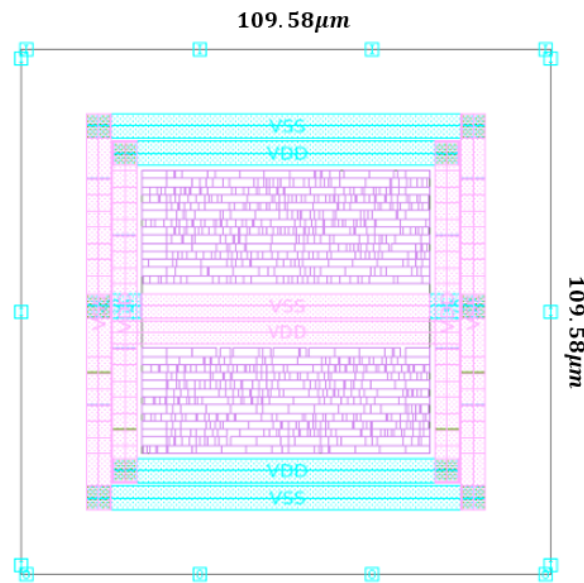
109.58μm

109.58μm

Fig. 6. Final layout of the FER system

Based on Figure 7 and Figure 8, no design rule violations and LVS violations were reported.

```
Total number of DRCs = 0
Total number of antenna violations = no antenna rules defined
Total number of voltage-area violations = no voltage-areas defined
Total number of tie to rail violations = not checked
Total number of tie to rail directly violations = not checked

1
```

Fig. 7. DRC report for the final layout design

```
** Total Floating ports are 0.
** Total Floating Nets are 0.
** Total SHORT Nets are 0.
** Total OPEN Nets are 0.
** Total Electrical Equivalent Error are 0.
** Total Must Joint Error are 0.

-- LVS END : --
Elapsed =    0:00:00, CPU =    0:00:00
Update error cell ...
1
```

Fig. 8. LVS report for the final layout design

4. Discussion

This study showed the overall process of designing and developing a FER system. By utilizing four convolutional layers, the CNN model developed in Google Colab achieved an impressive recognition accuracy of 92%. On the other hand, the simulation waveform shows that the FER system can recognize the image in $133767\ ns$. The performance of the proposed system is also analysed in terms of area and power consumption. The area of the FER system obtained after the physical synthesis stage is $3101.20\mu m^2$ while the power consumption is $871.05\mu W$ with the clock frequency is

$100MHz$. A FER system that is small, low power with acceptable recognition accuracy and high recognition speed able to be designed using ASIC implementation.

5. Conclusions

This paper introduces an ASIC-based FER system utilizing CNN technology. During the CNN model training on Google Colab, the FER-2013 dataset is employed, resulting in a commendable recognition accuracy of 92%. Subsequently, the physical design demonstrates a reduction in total area and power consumption compared to the logical synthesis stage outcomes. The FER system exhibits great potential in transforming facial expression recognition due to its remarkable precision, rapid recognition speeds and enhanced efficiency, enabling applications in emotion detection, human-computer interaction and biometric systems. For future advancements, the training and testing process of the CNN model could be optimized by employing the Verilog HDL module, eliminating the need for transfer learning from the CNN model trained on Google Colab. Additionally, augmenting the number of convolution layers could effectively extract more features from the database, thereby further enhancing recognition accuracy.

References
[1] Ozdemir, Mehmet Akif, Berkay Elagoz, Aysegul Alaybeyoglu, Reza Sadighzadeh and Aydin Akan. "Real time emotion recognition from facial expressions using CNN architecture." In *2019 medical technologies congress (tiptekno)*, pp. 1-4. IEEE, 2019. https://doi.org/10.1109/TIPTEKNO.2019.8895215

[2] Sanjana, S., S. Sanjana, V. R. Shriya, Gururaj Vaishnavi and K. Ashwini. "A review on various methodologies used for vehicle classification, helmet detection and number plate recognition." *Evolutionary Intelligence* 14, no. 2 (2021): 979-987. https://doi.org/10.1007/s12065-020-00493-7

[3] Ulusoy, Selen Işık, Şeref Abdurrahman Gülseren, Nermin Özkan and Cüneyt Bilen. "Facial emotion recognition deficits in patients with bipolar disorder and their healthy parents." *General Hospital Psychiatry* 65 (2020): 9-14. https://doi.org/10.1016/j.genhosppsych.2020.04.008

[4] Ajay, B. S. and Madhav Rao. "Binary neural network based real time emotion detection on an edge computing device to detect passenger anomaly." In *2021 34th International Conference on VLSI Design and 2021 20th International Conference on Embedded Systems (VLSID)*, pp. 175-180. IEEE, 2021. https://doi.org/10.1109/VLSID51830.2021.00035

[5] Krizhevsky, Alex, Ilya Sutskever and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks." *Communications of the ACM* 60, no. 6 (2017): 84-90. https://doi.org/10.1145/3065386

[6] Xiao, Rui, Junsheng Shi and Chao Zhang. "FPGA implementation of CNN for handwritten digit recognition." In *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, vol. 1, pp. 1128-1133. IEEE, 2020. https://doi.org/10.1109/ITNEC48623.2020.9085002

[7] Wong, Y. C. and Y. Q. Lee. "Design and Development of Deep Learning Convolutional Neural Network on an Field Programmable Gate Array." *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 10, no. 4 (2018): 25-29.

[8] Choudhari, Onkar, Marisha Chopade, Sourabh Chopde, Swarali Dabhadkar and V. Ingale. "Hardware accelerator: implementation of CNN on FPGA for digit recognition." In *2020 24th International Symposium on VLSI Design and Test (VDAT)*, pp. 1-6. IEEE, 2020. https://doi.org/10.1109/VDAT50263.2020.9190274

[9] Yan, Kewen, Shaohui Huang, Yaoxian Song, Wei Liu and Neng Fan. "Face recognition based on convolution neural network." In *2017 36th Chinese Control Conference (CCC)*, pp. 4077-4081. IEEE, 2017. https://doi.org/10.23919/ChiCC.2017.8027997

[10] Coşkun, Musab, Ayşegül Uçar, Özal Yildirim and Yakup Demir. "Face recognition based on convolutional neural network." In *2017 international conference on modern electrical and energy systems (MEES)*, pp. 376-379. IEEE, 2017. https://doi.org/10.1109/MEES.2017.8248937

[11] Ding, Chunhui, Tianlong Bao, Saleem Karmoshi and Ming Zhu. "Low-resolution face recognition via convolutional neural network." In *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)*, pp. 1157-1161. IEEE, 2017. https://doi.org/10.1109/ICCSN.2017.8230292

[12] Zhao, Guodong, Wei Wei, Xiaofei Xie, Shida Fan and Kai Sun. "An FPGA-based BNN real-time facial emotion recognition algorithm." In *2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pp. 20-24. IEEE, 2022. https://doi.org/10.1109/ICAICA54878.2022.9844526

[13] Cheang, Kah Ying and Siti Hawa Ruslan. "Performance Optimization of Face Detection Algorithm." *Evolution in Electrical and Electronic Engineering* 1, no. 1 (2020): 145-152.

[14] Kim, Jaemyung, Jin-Ku Kang and Yongwoo Kim. "A resource efficient integer-arithmetic-only FPGA-based CNN accelerator for real-time facial emotion recognition." *IEEE Access* 9 (2021): 104367-104381. https://doi.org/10.1109/ACCESS.2021.3099075

[15] Khan, Nizamuddin, Ajay Vikram Singh and Rajeev Agrawal. "Enhancing feature extraction technique through spatial deep learning model for facial emotion detection." *Annals of Emerging Technologies in Computing (AETiC)* 7, no. 2 (2023): 9-22. https://doi.org/10.33166/AETiC.2023.02.002

[16] Phan-Xuan, Hanh, Thuong Le-Tien and Sy Nguyen-Tan. "FPGA platform applied for facial expression recognition system using convolutional neural networks." *Procedia computer science* 151 (2019): 651-658. https://doi.org/10.1016/j.procs.2019.04.087

[17] Qiao, Shijie and Jie Ma. "Fpga implementation of face recognition system based on convolution neural network." In *2018 Chinese Automation Congress (CAC)*, pp. 2430-2434. IEEE, 2018. https://doi.org/10.1109/CAC.2018.8623662

[18] Komalck. "Komalck/Facial-Emotion-Recognition." *GitHub*. (2020). https://github.com/komalck/FACIAL-EMOTION-RECOGNITION

[19] Boaaaang. "CNN Implementation in Verilog." *GitHub*. (2021). https://github.com/boaaaang/CNN-Implementation-in-Verilog

Name of Author	Email
Khoo Wing Jian	khoowingjian@gmail.com
Lee Mun Feng	mun.feng99@gmail.com
Nabihah binti Ahmad	nabihah@uthm.edu.my
C. Uttraphan	chessda@uthm.edu.my
S. Ananiah Durai	ananiahdurai.s@vit.ac.in
Warsuzarina Mat Jubadi	suzarina@uthm.edu.my