Journal of Numerical Optimization and Technology Management

Vol. 1, No. 1, 2023

htttps://shmpublisher.com/index.php/jnotm

p-ISSN: xxxx-xxxx
e-ISSN: xxxx-xxxx

.

Deep convolutional generative adversarial networks for data imbalance in convolutional neural networks for facial expression classification

Sugiyarto Surono[1], Choo Wou Onn[2], Almuzhidul Mujhid[3], Nursyiva Irsalinda[1], Goh Khang Wen[2*]
[1]Mathematics Study Program, Universitas Ahmad Dahlan, Indonesia
[2]Faculty Data Science and Information Technology, INTI International University, Nilai Malaysia

Article Info

Article history:

Received June 2023
Revised June 2023
Accepted July 2023

Keywords:

Facial expression recognition
DCGAN
CNN

ABSTRACT

Facial expression recognition technology is a critical direction of emotion computing research, and it is an essential part of human-computer interaction. The facial expression recognition method is a classification method. An excellent classification method and widely used today are the Convolutional Neural Network (CNN). However, there are still shortcomings in accuracy in the CNN method if the available dataset is minimal and imbalanced. There are two ways to overcome this, adding the training data or changing the architecture on CNN. In this research, the researcher uses the method to add to the training dataset using the Deep Convolutional Generative Adversarial Networks (DCGAN) method.

Corresponding Author:

Goh Khang Wen,
Faculty Data Science and Information Technology,
INTI International University,
Nilai Malaysia
Email: khangwen.goh@newinti.edu.my
https://doi.org/10.00000/jnotm.0000.00.00.000

1. INTRODUCTION

Facial expression technology is a critical direction of emotion computing research, is an essential part of human-computer interaction, and has a wide range of applications in medicine, education, business marketing, and other fields [1]–[4]. One method of recognizing facial expressions is to use the classification method. An excellent classification method and widely used today is the Convolutional Neural Network (CNN) [5]–[7]. CNN can input the image directly and get the final classification result without preprocessing the data. By building a neural network model with a certain depth and combining nonlinear operations such as convolution and union, we can realize two crucial functions of mimicking the hierarchical processing of the human brain. It also allows the classification process to be faster and more accurate with smaller image

dimensions [8]. However, the drawback of the Convolutional Neural Network (CNN) method itself requires a vast data set to work appropriately and accurately [9]–[12].

So to overcome the lack of this data set, the researcher proposes a method to overcome the lack of data, namely the Deep Convolutional Generative Adversarial Networks method. Deep Convolutional Generative Adversarial Networks is the development of the Generative Adversarial Networks method, which is a method to increase the amount of image data by adding new image data (synthetic images) and as the standard solution against overfitting [13]–[15]. The synthesized image is obtained by approximating the distribution of the original data and then creating an image from the approximation of the distribution of the original image. In 2014, Ian Goodfellow, then a PhD student at the University of Montreal, discovered Generative Adversarial Networks (GANs). This technique has allowed computers to generate realistic data using not one but two separate neural networks. GANs weren't the first computer programs to create data, but their results and versatility set them apart. GANs have achieved incredible results long ago, thought nearly impossible for artificial systems. The ability to generate real-world quality fake images, convert doodles into photos, or turn video footage of horses into zebra dashes—all without requiring a lot of data [16], [17]. By adding synthetic image data using the Deep Convolutional Generative Adversarial Networks (DCGAN) method, it can overcome the shortcomings of the CNN method [18], [19]. The reason for choosing a thesis on this topic is that the researcher wants to explore the classification of images by increasing the accuracy of the CNN method.

2. METHOD

This section gives several essential definitions and theorems that are being used to support further discussion in the next section. These are facial expression datasets, Deep Convolutional Generative Adversarial Networks (DCGAN) and Convolutional Neural Networks (CNN).

2.1 Deep Convolutional Generative Adversarial Network

DCGAN is a different development method from GAN. The Generator and Discriminator use Deep convolution architecture whose goal is to get a more stable GAN by adding new systems in the GAN architecture. When modelled as Deep Layer Convolution, players G and D can be trained in alternation by decreasing binary crossover entropy.

$$G: R^d \rightarrow R^n$$

Generator G takes any sample $\boldsymbol{z} \in R^d$ from the distribution and then produces a sample $G(\boldsymbol{z})$.

$$\min_G \max_D V(D,G) \coloneqq \mathbb{E}_{x\sim\mu}[logD(x)] + \mathbb{E}_{z\sim\varepsilon}\left[\log\left(1 - D\big(G(z)\big)\right)\right]$$

Deep Convolutional Generative Adversarial Network (GAN) is a framework for estimating the generator model through the adversarial process. There are two models, namely the Generator model and the Discriminator model; both are trained simultaneously. The generator model G will capture the distribution of data, and the discriminator model D, which estimates the probability of the sample coming from the training data of G. The training procedure for G is to maximize the chance of D making a mistake. This framework fits the theory of minimax games with two players [15], [20]. GAN perspective defines the similarity of probability distributions, where two data sets are said to be equal if the samples come from the same (or nearly the same) probability distribution [21]. In an image (original image) consisting of a data set X⊂R^nwhich consists of a sample of the probability distribution μ(with density p(x)), and we we try to find the probability distribution ε (with a density of q(x)) so that is a good approximation of μ. This means that the ε distribution will produce the same or similar image as the original image [22], [23].

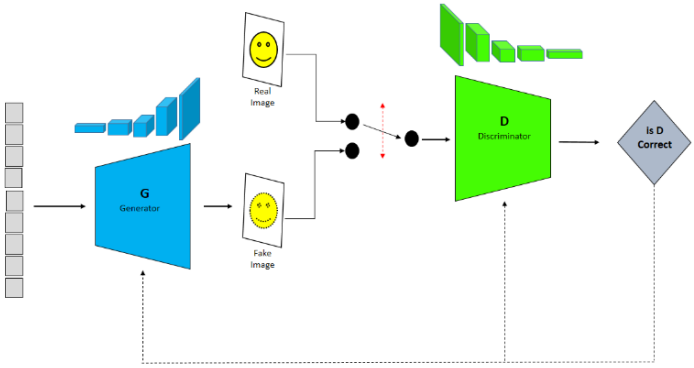

Figure 1. Architecture of DCGAN

2.2 Convolutional Neural Network

Convolutional neural networks(CNN) are the most popular class of models for image recognition and classification tasks nowadays [7], [23]. CNN is the development of a multilayer perceptron (MLP) to process two-dimensional data, which consists of two architectures, namely Convolution and Neural Network [24]. Two things that make CNN different from NN are convolution and backpropagation processes. The CNN method consists of two stages; it begins with image classification using feedforward, then the learning stage with the backpropagation method. Softmax regression is commonly used for classification tasks because it generates a well-performed probability distribution of the outputs [25], [26].

a. Convolutional and Pooling

Kernel or commonly called filter, is a square matrix with dimension n_k×n_k, where n_k is an integer and usually a small number, such as 3 or 5. Kernels or filters are traditionally used in image processing techniques, such as sharpening, blurring and embossing [27]. The pooling layer is additionally called as max-pooling layer or subsampling. Convolution example:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} * \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} = \sum_{i=1}^{3} \sum_{j=1}^{3} a_{ij} k_{ij}$$

b. Activation Function

The activation function is used to output a neuron in the hidden and output layers. The input of the activation function is the operating value of the linear combination of the input values and weights, which can be called net and can be expressed as follows:

$$net = \sum x_i w_i$$

With activation function,

$$f(net) = f(\sum x_i w_i)$$

activation functions we used in CNN and DCGAN

- Sigmoid function

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

$$range = (0,1)$$

- Rectified Linear Unit Function (ReLU)

$$0, if\ x \leq 0$$
$$x, if\ x > 0$$

$$range = [0, \infty)$$

- Gaussian Error Linear Units Function (GELU)

$$GELU(x) = xP(X \leq x) = x\phi(x) = x.\frac{1}{2}\left[1 + erf\left(x/\sqrt{2}\right)\right],$$

$$if\ X \sim \mathcal{N}(0,1)$$

- Softmax Fuction

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

$$range = (-1,1)$$

c. Neural Network

Neural Network (NN) is a method that imitates the working of biological neural networks. The primary component of Neural Networks is a neuron; it serves as a quantifier and nonlinear mapping processor. Between one neuron with another neuron that is connected by a value called weight [28], [29].

- Single Layer

  A single layer network consists of one input layer and one output layer [30]. The way it works is that the nodes of the input layer are directly projected to the output layer of the neurons.

$$output = f\left(\sum_{i=1}^{N} x_i\, w_i + b_i\right)$$
$$x_i = input$$
$$w_i = weight$$
$$b_i = bias$$

- Multi-Layer

  A multiple layer network is a network that consists of one or more hidden layers. This network can solve more complex problems, but it takes longer for the training process.

$$\boldsymbol{o_j} = \boldsymbol{\sigma}\left(\sum_{k=1}^{K} x_k\, w_{k,j} + \beta_j\right)$$
$$\boldsymbol{v_i} = \boldsymbol{\sigma}\left(\sum_{j=1}^{J} o_j\, u_{j,i} + \gamma_i\right)$$
$$\boldsymbol{v_i} = \boldsymbol{\sigma}\left(\sum_{j=1}^{J} \sigma\left(\sum_{k=1}^{K} x_k\, w_{k,j} + \beta_j\right) u_{j,i} + \gamma_i\right)$$

$$o_j = function\ of\ first\ layer$$
$$v_i = function\ of\ second\ layer$$

.

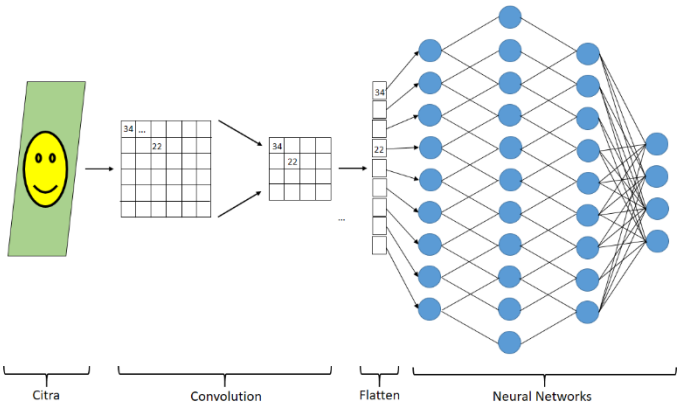

Figure 2. Architecture of CNN

3. RESULTS AND DISCUSSIONS

We build the DCGAN and CNN models in the Google Colab environment with GPU as accelerator hardware. The MMA Facial Expression Dataset dataset for DCGAN is 900 happy images and 820 angry images.

Figure 3. Angry image (left) and happy image (right)

3.1 DCGAN in MMA Facial Expression Dataset

We first take the angry image as data training for our DCGAN because it only has 820 images. Our target is to create 80 imitation images to balance the datasets. The following is the architecture of the DCGAN.

Table 1. Critic or Discriminator architecture in DCGAN

Type	Input Channel	Kernel size	Output Channel	Activation
Convolution Block	3 × 64 × 64	4	64 × 32 × 32	GELU
Convolution Block	64 × 32 × 32	4	128 × 16 × 16	GELU
Convolution Block	128 × 16 × 16	4	256 × 8 × 8	GELU
Convolution Block	256 × 8 × 8	4	512 × 4 × 4	GELU
Convolution Block	512 × 4 × 4	4	2	Sigmoid

Table 2. Generator architecture in DCGAN

Type	Input Channel	Kernel size	Output Channel	Activation
Transpose Convolution	100 × 1 × 1	4	512 × 4 × 4	GELU
Transpose Convolution	512 × 4 × 4	4	256 × 8 × 8	GELU
Transpose Convolution	256 × 8 × 8	4	128 × 16 × 16	GELU
Transpose Convolution	128 × 16 × 16	4	64 × 32 × 32	GELU
Transpose Convolution	64 × 32 × 32	4	3 × 64 × 64	Tanh

In this study of the DCGAN method, we used 300 epochs. and this is the imitation image generated from the DCGAN process in several epochs.

Figure 5. Fake images generated in multiple epochs, 100^th epoch (left), 250^th epoch (left)

As we can see, the image generated in the 250^th epoch is quite good at defining angry expressions. So we can stop and take the imitation images generated at the 300^th epoch.

3.2 Classification CNN

Here we classify two types of data, datasets without augmentation and datasets with augmented DCGAN. In our CNN architecture, we apply the same architecture for both.

a. Resizing image, Giving RGB value, and Rescaling

The first process is to change the image size to $n \times n$ size; in this study, we change the image size to 100×100.

Figure 6. Before resizing (left) and after resizing (right)

After the resizing process, the next step is to convert the image into a matrix 100×100 that has RGB value and rescaled by dividing all the RGB values by 255.

b. Convolution and Pooling

Table 3. architecture of CNN

Type	Input size	Kernel size	Output size	Activation
Convolution	$100 \times 100 \times 3$	$3 \times 3 \times 32$	$98 \times 98 \times 32$	ReLU
Max Pooling	$98 \times 98 \times 32$	$2 \times 2 \times 1$	$49 \times 49 \times 32$	-
Convolution	$49 \times 49 \times 32$	$3 \times 3 \times 32$	$47 \times 47 \times 32$	ReLU
Max Pooling	$47 \times 47 \times 32$	$2 \times 2 \times 1$	$23 \times 23 \times 32$	-
Convolution	$23 \times 23 \times 32$	$3 \times 3 \times 64$	$21 \times 21 \times 64$	ReLU
Max Pooling	$21 \times 21 \times 64$	$2 \times 2 \times 1$	$10 \times 10 \times 64$	-
Flatten	$10 \times 10 \times 64$	-	6400×1	-
Dense Dropout	6400	-	1024	ReLU
Dense	1024	-	2	Softmax

These are the results of the facial expressions classification with two experiments.

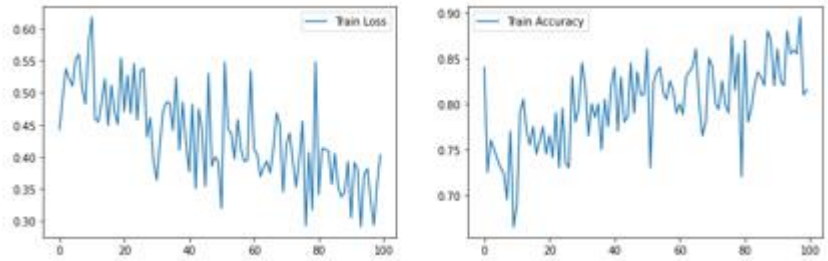

Figure 7. Train loss of CNN (left) and accuracy of CNN (right)

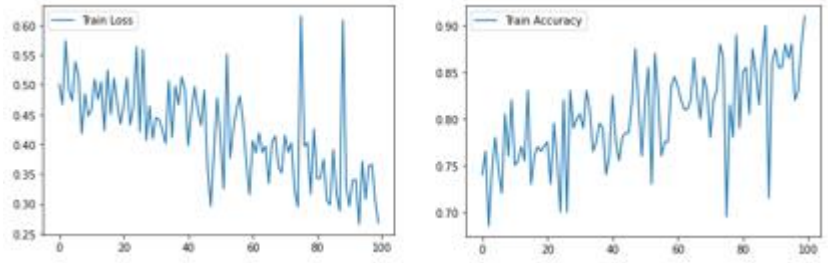

Figure 8. Train loss of CNN with DCGAN data (left) and accuracy of CNN with DCGAN data (right)

It can be seen that the loss from the classification training process decreases, and the classification accuracy increases.

Table 3. Comparison of accuracy and loss of CNN architecture with two kinds of datasets (datasets using traditional augmentation and datasets using DCGAN)

Type	Accuracy	Loss
CNN	81.50%	0.4033
CNN with DCGAN data	91.00%	0.2679

4. CONCLUSION

In this paper, we explored how image augmentation DCGAN can overcome unbalanced training data to increase accuracy in face classification by using the CNN method to obtain better accuracy. While getting imitation images to balance the training data, DCGAN is quite good and solves the data imbalance problem and provides increased accuracy in image classification task by 9,5%.

However, the work still has some limitations. For instance, the number of datasets is limited to emotions and only DCGAN is used in our model and still separated from CNN architecture. Therefore, we consider in future research that we can combine DCGAN and CNN in one architecture so that it can solve data imbalances in any image classification problems.

REFERENCES

[1] M. Wang, P. Tan, X. Zhang, Y. Kang, C. Jin, and J. Cao, "Facial expression recognition based on CNN," in *Journal of Physics: Conference Series*, 2020. doi: 10.1088/1742-6596/1601/5/052027.

[2] H. Kumar, A. Elhance, V. Nagpal, N. Partheeban, K. M. Baalamurugan, and S. Sriramulu, "Facial Expression Recognition System," in *Lecture Notes on Data Engineering and Communications Technologies*, 2021. doi: 10.1007/978-981-16-1866-6_25.

[3] H. Ling, J. Wu, J. Huang, J. Chen, and P. Li, "Attention-based convolutional neural network for deep face recognition," *Multimed. Tools Appl.*, vol. 79, no. 9–10, 2020, doi: 10.1007/s11042-019-08422-2.

[4] Y. M. R. Et.al, "Detection of Tumors From MRI Brain Images Using CNN With Extensive Augmentation," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 6, 2021, doi: 10.17762/turcomat.v12i6.1266.

[5] K. Madineni and P. Vasudevan, "Handwritten Text Recognition Based On CNN Classification," *Int. J. Adv. Sci. Technol.*, vol. 29, no. 9s, 2020.

[6] S. Oh, J. Y. Lee, and D. K. Kim, "The design of CNN architectures for optimal six basic emotion classification using multiple physiological signals," *Sensors (Switzerland)*, vol. 20, no. 3, 2020, doi: 10.3390/s20030866.

[7] S. Ashraf, I. Kadery, A. A. Chowdhury, T. Z. Mahbub, and R. M. Rahman, "Fruit Image Classification Using Convolutional Neural Networks," *Int. J. Softw. Innov.*, vol. 7, no. 4, 2019, doi: 10.4018/IJSI.2019100103.

[8] Y. Guo, Y. Liu, E. M. Bakker, Y. Guo, and M. S. Lew, "CNN-RNN: a large-scale hierarchical image classification framework," *Multimed. Tools Appl.*, vol. 77, no. 8, 2018, doi: 10.1007/s11042-017-5443-x.

[9] X. Zhu, Y. Liu, J. Li, T. Wan, and Z. Qin, "Emotion classification with data augmentation using generative adversarial networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018. doi: 10.1007/978-3-319-93040-4_28.

[10] T. Guo, J. Dong, H. Li, and Y. Gao, "Simple convolutional neural network on image classification," in *2017 IEEE 2nd International Conference on Big Data Analysis, ICBDA 2017*, 2017. doi: 10.1109/ICBDA.2017.8078730.

[11] L. Li, X. Zhu, Y. Hao, S. Wang, X. Gao, and Q. Huang, "A hierarchical CNN-RNN approach for visual emotion classification," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 15, no. 3s, 2019, doi: 10.1145/3359753.

[12] I. Amerini, C. T. Li, and R. Caldelli, "Social Network Identification Through Image Classification with CNN," *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2903876.

[13] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," in *Advances in Neural Information Processing Systems*, 2020.

[14] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.

[15] I. Goodfellow *et al.*, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, 2020, doi: 10.1145/3422622.

[16] Z. J. Xu, R. F. Wang, J. Wang, and D. H. Yu, "Parkinson's Disease Detection Based on Spectrogram-Deep Convolutional Generative Adversarial Network Sample Augmentation," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3037775.

[17] S. K. Venu and S. Ravula, "Evaluation of deep convolutional generative adversarial networks for data augmentation of chest x-ray images," *Futur. Internet*, vol. 13, no. 1, 2021, doi: 10.3390/fi13010008.

[18] S. H. Choi and S. H. Jung, "Similarity analysis of actual fake fingerprints and generated fake fingerprints by DCGAN," *Int. J. Fuzzy Log. Intell. Syst.*, vol. 19, no. 1, 2019, doi: 10.5391/IJFIS.2019.19.1.40.

[19] C. Uzun, M. B. Çolakoğlu, and A. İnceoğlu, "GAN as a generative architectural plan layout tool: A case study for training DCGAN with palladian plans and evaluation of DCGAN outputs," *A/Z ITU J. Fac. Archit.*, vol. 17, no. 2, 2020, doi: 10.5505/itujfa.2020.54037.

[20] A. Mufti, B. Antonelli, and J. Monello, "Conditional GANs for painting generation," 2020. doi: 10.1117/12.2556551.

[21] J. Langr and V. Bok, "Manning | GANs in Action," *GANs in Action: Deep learning with Generative Adversarial Networks*, 2019.

[22] C. Dewi, R. C. Chen, Y. T. Liu, and S. K. Tai, "Synthetic Data generation using DCGAN for improved traffic sign recognition," *Neural Comput. Appl.*, 2021, doi: 10.1007/s00521-021-05982-z.

[23] Y. Pei, Y. Huang, Q. Zou, X. Zhang, and S. Wang, "Effects of Image Degradation and Degradation Removal to CNN-Based Image Classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, 2021, doi: 10.1109/TPAMI.2019.2950923.

[24] J. Qin, W. Pan, X. Xiang, Y. Tan, and G. Hou, "A biological image classification method based on improved CNN," *Ecol. Inform.*, vol. 58, 2020, doi: 10.1016/j.ecoinf.2020.101093.

[25] W. S. Eka Putra, "Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101," *J. Tek. ITS*, vol. 5, no. 1, 2016, doi: 10.12962/j23373539.v5i1.15696.

[26] W. Li, C. Chen, M. Zhang, H. Li, and Q. Du, "Data Augmentation for Hyperspectral Image Classification with Deep CNN," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 4, 2019, doi: 10.1109/LGRS.2018.2878773.

[27] U. Michelucci, *Advanced applied deep learning: Convolutional neural networks and object detection*. 2019. doi: 10.1007/978-1-4842-4976-5.

[28] M. A. Nasichuddin, T. B. Adji, and W. Widyawan, "Performance Improvement Using CNN for Sentiment Analysis," *IJITEE (International J. Inf. Technol. Electr. Eng.*, vol. 2, no. 1, 2018, doi: 10.22146/ijitee.36642.

[29] K. Sankar Raja Sekhar, T. Ranga Babu, G. Prathibha, K. Vijay, and L. Chiau Ming, "Dermoscopic image classification using CNN with Handcrafted features," *J. King Saud Univ. - Sci.*, vol. 33, no. 6, 2021, doi: 10.1016/j.jksus.2021.101550.

[30] L. Fausset, "Fundamentals of Neural Networks: Architecture, Algorithm, and Application," *IEEE Trans. Comput.*, vol. C–18, no. 6, 1994.

.