SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11373

e-ISSN : 2541-2019
p-ISSN : 2541-044X

Super Resolution Generative Adversarial Networks for Image Supervise Learning

Mariska Lupitha[1)*], Handri Santoso[2)],
[1)2)] Universitas Pradita
[1)]mariska.lupitha@student.pradita.ac.id, [2)]handri.santoso@pradita.ac.id

Submitted : Apr 9, 2022 | **Accepted** : Apr 10, 2022 | **Published** : Apr 11, 2022

Abstract: The E-Tilang application system has been widely used to support modern traffic, whereas protocol roads in big cities in Indonesia are already widely used. In principle, the plate number detection tool uses image recognition for detection. Image number plates on vehicles cannot always be read clearly, this is what causes the detection method to be a problem if the image plate number is further processed. The method for processing the plate number image uses deep learning and computer vision methods. For the condition of the image plate number that is not clear, the process of improving the image resolution from low resolution to high resolution is carried out, by applying Generative Adversarial Networks. This method consists of two main parts, namely Generate and Discriminator. Generate serves to generate an image and the Discriminator here is to check the image, can the image plate number be read or not? So that if the image plate number cannot be read, then the process is carried out again to the Generator until it is received by the Discriminator to be read. The process does not end here, the results will be carried out in the next process using Convolutional Neural Networks. Where the process is to detect the plate number image according to the classification of the plate number according to the region. The point is that an unclear image becomes clear by increasing the resolution from low resolution to high resolution so that it is easily read by the Convolutional Neural Network (CNN) algorithm so that the image is easily recognized by the CNN Algorithm. This becomes important in the CNN algorithm process because it gets the processed dataset. To produce a good model, preprocessing of the dataset is carried out. So that the model can detect the image well in terms of model performance.

Keywords: Generative Adversarial Networks, Convolutional Neural Network, Image Plate Number Vehicle, Deep Learning, Computer Vision

INTRODUCTION

Currently, the use of image detection has been widely used in the dynamics of any field. Such as the use of existing detection images, such as the use of Deep Learning algorithms such as the Convolutional Neural Network (Yang et al., 2020). For example, many image detection systems, if you get an object that is not clear in the image, then the detection process will experience an error. Many of today's vehicle license plates do not work properly. Like the use of an opaque plate, the use of a plate is mounted obliquely or protrudes downwards. The purpose of using it varies, so that it is not seen by the police, transportation and others.

If the installed camera is not able to detect the vehicle number plate image, then the results obtained will definitely experience errors. If the Convolutional Neural Network algorithm detects it, then feature extraction is carried out which will then be classified or regressed in detecting the correct vehicle number plates and there are vehicle plates that do not match the detection.

CNN works by utilizing the convolution process by moving a convolution kernel (Jiang et al., 2021) or filter, which is a certain size into an image. After that the information is obtained, then perform a new representation of the multiplication results on each part of the image using convolution or filters. As figure 1, explains how the Convolutional Neural Network works. The purpose of CNN is that the spatial hierarchical structure of elements is studied using back propagation from several building blocks. Building blocks include convolutional layers, composite layers, and interconnected layers. CNN is also a mathematical construction, consisting of three types of layers. Convolutional layer, splice layer, and fully connected (Li et al., 2017) layer.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11373
e-ISSN : 2541-2019
p-ISSN : 2541-044X

The first two layers are convolution and merging layers for feature extraction (Kulkarni et al., 2021). The third layer is the connected layer mapping the extracted features. The last layer is classification.

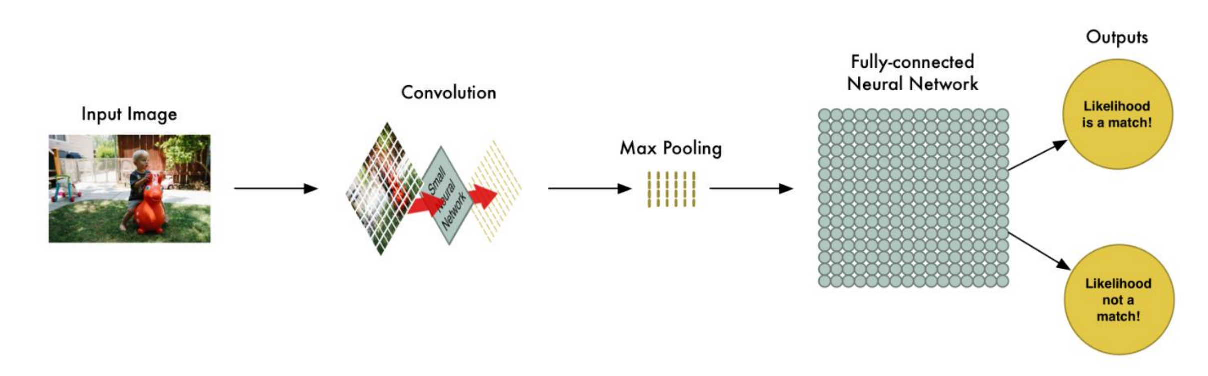

Fig. 1 How Convolutional Neural Networks work
Source : Medium.com

Here are the steps in Convolutional Neural Networks:
1. The image is split and the image is overlapping. The input image is done by splitting as many as 77 small images.
2. Input the image after solving, then input it into the Small Neural Network. Each small image represents a feature of the image. This is what makes CNN have the ability to recognize images or objects. All parts of each thumbnail, the same filter is used. This means that each part of the image will have the same multiplier, the neural network is referred to as weights sharing.
3. Small images are then stored in an array.
4. Downsampling, In step 3, the array is too large, done by reducing the size of the array and using downsampling. Max pooling is done by taking the largest pixel value in each pooling kernel. Reducing the number of parameters, the most important information of the section is still retrieved.
5. Step 1 to step 4, large images into small arrays. The final neural network uses a fully connected (Mukhopadhyay et al., 2022)which is called the Classification step. So that this section can decide which object is appropriate or not.

Fig. 2 Vehicle license plate that looks blurry
Source : Google Image

There is a problem with image detection if the existing image is blurry, this can interfere with the image detection process. Like Fig. 2, the image looks blurry. In this study, there are Research Questions:
1. How to make a blurry image become a clear and clear image?
2. Using deep learning algorithms to detect images?
3. How is the performance of the deep learning algorithm model?

The discussion in this study consists of an introduction related to research. The Literature Review section describes the previous studies in discussing issues related to low resolution and high resolution. The method

*name of corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11373
e-ISSN : 2541-2019
p-ISSN : 2541-044X

section will discuss the proposed method to be used in image detection or object detection. Results and Discussion section, displays the results of the proposed method proposed. The conclusion section is the conclusion of a series of activities in preprocessing problems and subsequent processes.

LITERATURE REVIEW

Several studies related to the Convolutional Neural Network (CNN) are as follows.

Table 1. Previous research topics discussing CNN

Author	Topic	Advantages	Disadvantages
(Suartika E. P et al., 2016)	Image Classification Using Convolutional Neural Network (CNN) on Caltech 101.	Discussion of classification using the CNN method, carried out in detail, explained about feature extraction and classification.	Not discussing dataset preparation, which is really needed so that the dataset is clean and not overfitting.
(Hendriyana & Yazid Hilman Maulana, 2020)	Identification of Types of Wood using Convolutional Neural Network with Mobilenet Architecture	Discussion on wood identification using the CNN method, with the Mobilenet algorithm. Mobilenet includes using light computing but powerful running on mobile devices.	Not discussing dataset preparation, which is really needed so that the dataset is clean and not overfitting.
(Udayana et al., 2021)	Detecting Excessive Daytime Sleepiness with CNN and Commercial Grade EEG	Discussion using CNN method, Pre-processing using Data Normalization, Data Sampling, and Data Acquisition.	The discussion does not use an image dataset.
(Upadhyay et al., 2022)	Coherent convolution neural network based retinal disease detection using optical coherence tomographic images	CNN was used to explore eye disease, based on the synchronous network structure. This model consists of 5 layers, high accuracy is obtained on images with a size of 64x64, the use of VGG16 on pre-trained, with 16 layers. In the sequential model, Block-1 consists of two convolutions a layer with a filter size of 3X3 followed by a maxpooling layer. Block-2 consisting of three layers of convolution with a filter size of 3X3 followed by maxpooling.	Not discussing dataset preparation, which is really needed so that the dataset is clean and not overfitting.
(Sun et al., 2021)	MFBCNNC: Momentum factor biogeography convolutional neural network for COVID-19 detection via chest X-ray images	Use method three convolutional neural networks (LeNet-5, VGG-16, and ResNet-18) as the basic classification model for the detection of COVID-19, Normal, and Pneumonia chest X-ray images. The accuracy of LeNet-5, VGG-16, and ResNet-18 increased by 1.56%, 1.48%, and 0.73% after using biogeography-based optimization to optimize the hyperparameters of the model.	Not discussing dataset preparation, which is really needed so that the dataset is clean and not overfitting
(Muralidharan et al., 2022)	Detection of COVID19 from X-ray images using multiscale Deep Convolutional Neural Network	Detection of covid-19 is carried out using X-Ray which is directly processed using the Convolutional Neural Network..	Does not discuss dataset preparation.
(Matsunobu et	Cloud detection using	This work evaluates the	Not discussing dataset

*name of corresponding author

al., 2021)	convolutional neural networks on remote sensing images	performance of a convolutional neural network (CNN)-based cloud mask (CCM) at 12 geographically and climatically diverse locations across the continental U.S. (CONUS). Performance is largely characterized by the Mathews correlation coefficient (MCC) score.	preparation, which is really needed so that the dataset is clean and not overfitting

This state-of-the-art research uses image preprocessing as an image cleaning dataset (such as image blur, images too small). The image dataset will be improved so that all datasets will become very visually clear. Bad dataset image will affect during training and testing.

METHOD

Image pre-processing is not limited to, resizing, orientation and color correction. Manipulations applied to images create different versions of similar content to expose the model to a wider training set. Random changes in the rotation, brightness, or scale of the input image require the model to consider what the subject of the image looks like in various situations. The image augmentation process is only applied to the training data. Thus, the transformation can be an augmentation in some situations. The image augmentation process can increase the number of datasets for training, thus producing a good dataset when creating a training model.

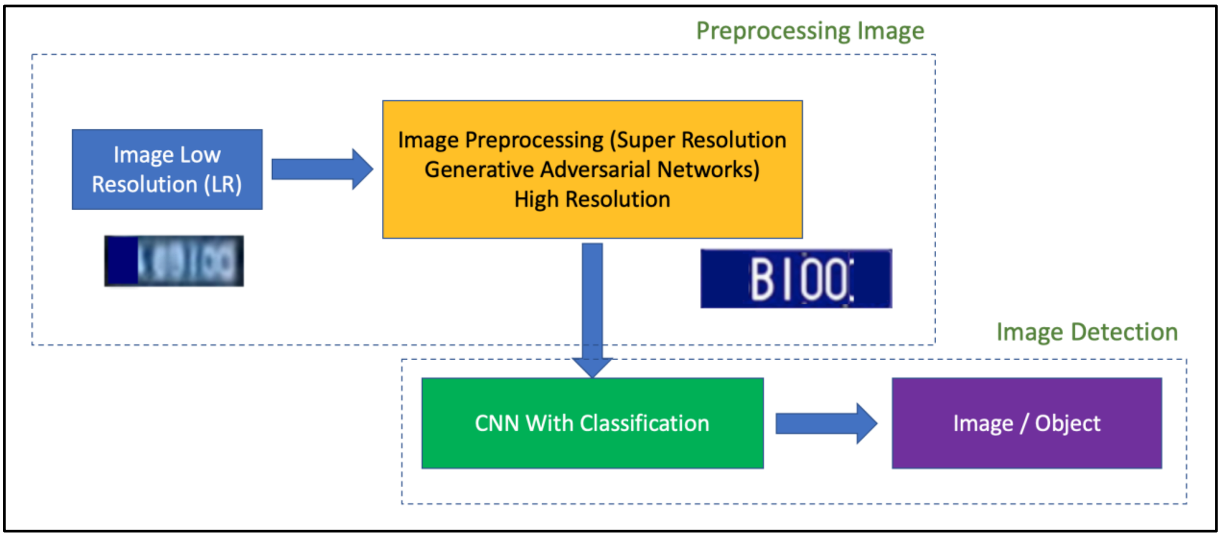

Fig 3. Proposed method for resolving blurry images.
Source : researcher property

Often in processing image datasets do not pay attention to the dataset, for better or for worse. In Machine Learning processing, if the dataset is bad, it will produce a bad model as well. To overcome this problem, suggestions for improving or augmenting images need to be carried out on the dataset, so that the dataset will produce a good model if the image dataset used is good. This study proposes to clean the image dataset, and augmented the image is done so that the image dataset becomes better. The proposed augmented image uses image preprocessing with the Super-Resolution Generative Adversarial Networks (SRGAN) (Moran et al., 2021), (Moran et al., 2021), (Bode et al., 2021) method. Basically, SRGAN is divided into Generator as an image producer, while Discriminator as a differentiator between the generated image and the original image. Therefore, it is necessary to conduct training so that the Discriminator is able to distinguish the image from the generator from the real image dataset.

Loss Function

The loss function (Abu-Srhan et al., 2022), in a neural network calculates the difference between the expected result and the real result. Patch sketches of the natural image of the manifold (red color) and super-finished patch were obtained with MSE (blue color) and GAN (orange color). The solution with MSE (Y. S. Liu et al., 2021) looks too smooth because it is a pixel-wise average of the possible solutions in the pixel space, while the GAN

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11373
e-ISSN : 2541-2019
p-ISSN : 2541-044X

method performs reconstruction towards a natural image manifold resulting in a more convincing perceptual solution. As in Fig 4.

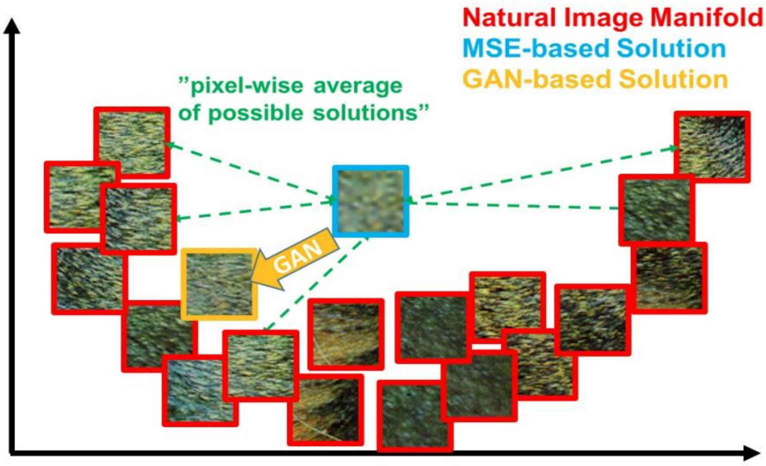

Fig. 4 Illustration of Natural Image Manifold, MSE-based Solution, and GAN-based Solution
Source : (Qi et al., 2018)

Pixel-wise loss function to handle inherent uncertainty and perform high-frequency lost recovery. The detail of texture is that minimizing MSE encourages finding the pixel average of a reasonable solution. Usually, the pixels are too fine and have poor perceptual quality (Qi et al., 2018).

Generator Network
Skip connection is a standard module in the convolution architecture of neural networks (Lin et al., 2021). The use of skip connections provides an alternative path for gradients with back propagation. Experimentally validated and additional paths are useful for model convergence. Pass connections in deep architecture, bypassing multiple layers in the neural network thereby providing output from one layer as input to the next. So it can speed up the process. Generator Network works as follows: Input Low Resolution (LR) performed Conv and Parametric ReLU, with k9n64s1 (Kernel size 9, feature maps 64, stride 1). B residual blocks consist of k3n64s1, k3n64s1 is the result of Conv, BatchNorm, Parametric ReLU. Skip connection 4 times. Next k3n64s1 results from Conv, BatchNorm, Elementwise Sum. k3n256s1 results from Conv, PixelShuffler x2, ReLU, continue to process until k9n3s1. This results in an image that is 4x larger in size. This process uses Convolutional Neural Networks with model VGG16 (16 layers). All Generator Network processes as in Fig. 5.

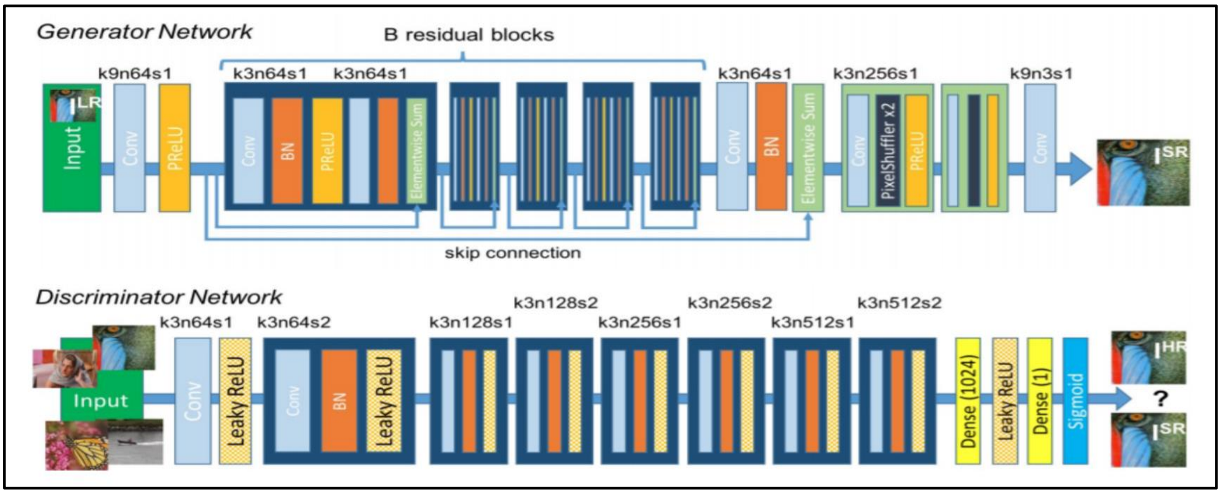

Fig 5. Image dengan Super Resolution Generative Adversarial Network.
Source : (Y. Liu et al., 2019)

Discriminator Network
*name of corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11373
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Input from Generator and input from the real dataset, then Conv process, Leaky ReLU (Y. Liu et al., 2019)is performed, resulting in k3n64s1 (Kernel Size 3, feature map 64, stride 1). Then the Conv, BatchNorm, and Leaky ReLU processes generate k3n64s2. The feature extraction process Conv, BatchNorm, LeakyReLU starting from (Conv, BatchNorm, LeakyReLU) k3n128s1, (Conv, BatchNorm, LeakyReLU) k3n128s2, (Conv, BatchNorm, LeakyReLU) k3n256s1, (Conv, BatchNorm, LeakyReLU) k3n256s2, (Conv, BatchNorm, LeakyReLU) k3n512s1, (Conv, BatchNorm, LeakyReLU) k3n512s2. The following processes are classified, such as Dense (1024), LeakyReLU, Dense (1), and sigmoid. The result is a Super Resolution and High-Resolution image.

RESULT

Generative Adversarial Networks are used to correct blurry, small, and unclear images so that the image improvement process is carried out. This process is known as a Super Resolution image. The results of the devel opment of the Generative Adversarial Networks model for the Generator module are shown in table 1.

Table 1. Model: "Generator"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, None, None, 3)]	0
lambda (Lambda)	(None, None, None, 3)	0
conv2d_block (Conv2DBlock)	(None, None, None, 64)	1856
conv2d_block_1 (Conv2DBlock)	(None, None, None, 64)	4224
rrd_block (RRDBlock)	(None, None, None, 64)	251072
rrd_block_1 (RRDBlock)	(None, None, None, 64)	251072
rrd_block_2 (RRDBlock)	(None, None, None, 64)	251072
rrd_block_3 (RRDBlock)	(None, None, None, 64)	251072
pixel_shuffle_up_sampling (PixelShuffleUpSam	(None, None, None, 64)	147776
pixel_shuffle_up_sampling_1 (PixelShuffleUpS	(None, None, None, 64)	147776
conv2d_block_40 (Conv2DBlock)	(None, None, None, 64)	36992
conv2d_block_41 (Conv2DBlock)	(None, None, None, 3)	1731
activation (Activation)	(None, None, None, 3)	0
lambda_3 (Lambda)	(None, None, None, 3)	0

Total params: 1,344,643
Trainable params: 1,341,571
Non-trainable params: 3,072

Discriminator Network

This model is to compare the results of the model generator with the real dataset, later the results will distinguish between the generator and real data.

Table 2. Model: "Discriminator"

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	(None, 128, 128, 3)]	0
lambda_4 (Lambda)	(None, 128, 128, 3)	0
conv2d_42 (Conv2D)	(None, 128, 128, 32)	896
leaky_re_lu (LeakyReLU)	(None, 128, 128, 32)	0
conv2d_43 (Conv2D)	(None, 64, 64, 32)	9248

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11373

e-ISSN : 2541-2019
p-ISSN : 2541-044X

batch_normalization_36 (BatchNormalization)	(None, 64, 64, 32)	128
leaky_re_lu_1 (LeakyReLU)	(None, 64, 64, 32)	0
conv2d_44 (Conv2D)	(None, 64, 64, 64)	18496
batch_normalization_37 (BatchNormalization)	(None, 64, 64, 64)	256
leaky_re_lu_2 (LeakyReLU)	(None, 64, 64, 64)	0
conv2d_45 (Conv2D)	(None, 32, 32, 64)	36928
batch_normalization_38 (BatchNormalization)	(None, 32, 32, 64)	256
leaky_re_lu_3 (LeakyReLU)	(None, 32, 32, 64)	0
conv2d_46 (Conv2D)	(None, 32, 32, 128)	73856
batch_normalization_39 (BatchNormalization)	(None, 32, 32, 128)	512
leaky_re_lu_4 (LeakyReLU)	(None, 32, 32, 128)	0
conv2d_47 (Conv2D)	(None, 16, 16, 128)	147584
batch_normalization_40 (BatchNormalization)	(None, 16, 16, 128)	512
leaky_re_lu_5 (LeakyReLU)	(None, 16, 16, 128)	0
conv2d_48 (Conv2D)	(None, 16, 16, 256)	295168
batch_normalization_41 (BatchNormalization)	(None, 16, 16, 256)	1024
leaky_re_lu_6 (LeakyReLU)	(None, 16, 16, 256)	0
conv2d_49 (Conv2D)	(None, 8, 8, 256)	590080
batch_normalization_42 (BatchNormalization)	(None, 8, 8, 256)	1024
leaky_re_lu_7 (LeakyReLU)	(None, 8, 8, 256)	0
flatten (Flatten)	(None, 16384)	0
dense (Dense)	(None, 1024)	16778240
leaky_re_lu_8 (LeakyReLU)	(None, 1024)	0
dense_1 (Dense)	(None, 1024)	1049600
leaky_re_lu_9 (LeakyReLU)	(None, 1024)	0
dense_2 (Dense)	(None, 1)	1025

==

Total params: 19,004,833
Trainable params: 19,002,977
Non-trainable params: 1,856

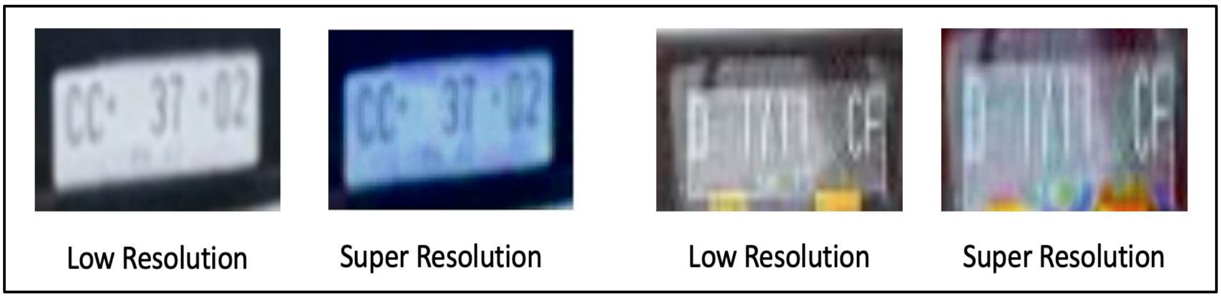

Fig. 6 Result Super Resoultion Generative Adversarial Network
Source : Google Image

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11373
e-ISSN : 2541-2019
p-ISSN : 2541-044X

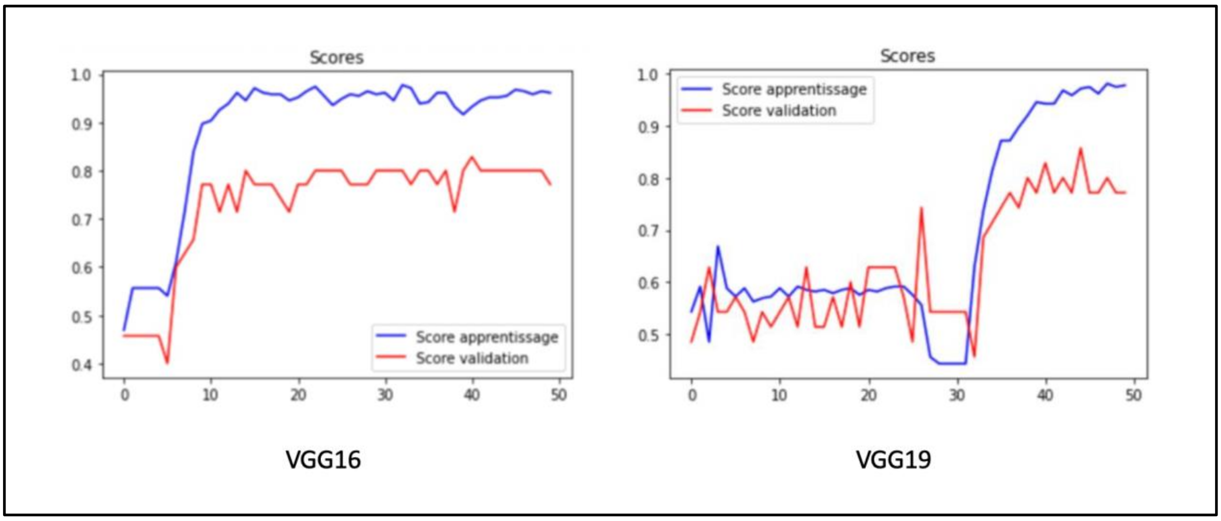

Fig. 7 Performance Convolutional Neural Network
Source : researcher property

The results of the training process on classification using CNN with the VGG16 model can achieve Loss accuracy: 0.65% and Accuracy model 85.90%. The results of the training process with VGG19 can achieve Loss accuracy: 0.65% and Accuracy model reaches 91.5%. From the experiment by comparing the two Convolutional Neural Network (CNN) methods, using VGG19 can produce an increased accuracy score.

DISCUSSIONS

Based on a report from Keras.io, the accuracy range of VGG16 and VGG19 is in the 71% - 90.1% range. VGG16's accuracy is around 85.9%, which means it's still pretty good. While the accuracy of VGG19 is around 91.5%, it means that there is an increase of 1.4% higher than the standard range of Keras.io. This means that there is an increase in accuracy by performing augmented image datasets by preprocessing Super Resolution Generative Adversarial Networks.

CONCLUSION

Based on experiments on Super-Resolution Generative Adversarial Networks research as preprocessing in Image Supervise Learning using the VGG16 and VGG19 Classification Algorithms, the results have improved accuracy. The accuracy of VGG19 reaches 91.5% accuracy which is higher than the standard of hard.io accuracy. This proves that the hypothesis for image preprocessing will improve the accuracy performance of the Image Supervise algorithm with VGG19. Proof of correct accuracy by conducting an experiment with classification with VGG19.

REFERENCES

Abu-Srhan, A., Abushariah, M. A. M., & Al-Kadi, O. S. (2022). The effect of loss function on conditional generative adversarial networks. *Journal of King Saud University - Computer and Information Sciences*, *xxxx*. https://doi.org/10.1016/j.jksuci.2022.02.018

Bode, M., Gauding, M., Lian, Z., Denker, D., Davidovic, M., Kleinheinz, K., Jitsev, J., & Pitsch, H. (2021). Using physics-informed enhanced super-resolution generative adversarial networks for subfilter modeling in turbulent reactive flows. *Proceedings of the Combustion Institute*, *38*(2), 2617–2625. https://doi.org/10.1016/j.proci.2020.06.022

Hendriyana, H., & Yazid Hilman Maulana. (2020). Identification of Types of Wood using Convolutional Neural Network with Mobilenet Architecture. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *4*(1), 70–76. https://doi.org/10.29207/resti.v4i1.1445

Jiang, M., Zhi, M., Wei, L., Yang, X., Zhang, J., Li, Y., Wang, P., Huang, J., & Yang, G. (2021). FA-GAN: Fused attentive generative adversarial networks for MRI image super-resolution. *Computerized Medical Imaging and Graphics*, *92*(April), 101969. https://doi.org/10.1016/j.compmedimag.2021.101969

Kulkarni, U., Meena, S. M., Gurlahosur, S. V., & Bhogar, G. (2021). Quantization Friendly MobileNet (QF-MobileNet) Architecture for Vision Based Applications on Embedded Platforms. *Neural Networks*, *136*,

*name of corresponding author

28–39. https://doi.org/10.1016/j.neunet.2020.12.022

Li, S., Jiang, H., & Pang, W. (2017). Joint multiple fully connected convolutional neural network with extreme learning machine for hepatocellular carcinoma nuclei grading. *Computers in Biology and Medicine*, *84*, 156–167. https://doi.org/10.1016/j.compbiomed.2017.03.017

Lin, C., Zhang, S., You, S., Liu, X., & Zhu, Z. (2021). Real-time foreground object segmentation networks using long and short skip connections. *Information Sciences*, *571*, 543–559. https://doi.org/10.1016/j.ins.2021.01.044

Liu, Y. S., Huang, C. H., & You, S. D. (2021). Estimation of channel MSE for ATSC 3.0 receiver and its applications. *ICT Express*, *xxxx*, 1–5. https://doi.org/10.1016/j.icte.2021.09.001

Liu, Y., Wang, X., Wang, L., & Liu, D. (2019). A modified leaky ReLU scheme (MLRS) for topology optimization with multiple materials. *Applied Mathematics and Computation*, *352*, 188–204. https://doi.org/10.1016/j.amc.2019.01.038

Matsunobu, L. M., Pedro, H. T. C., & Coimbra, C. F. M. (2021). Cloud detection using convolutional neural networks on remote sensing images. *Solar Energy*, *230*(May), 1020–1032. https://doi.org/10.1016/j.solener.2021.10.065

Moran, M. B. H., Faria, M. D. B., Giraldi, G. A., Bastos, L. F., & Conci, A. (2021). Using super-resolution generative adversarial network models and transfer learning to obtain high resolution digital periapical radiographs. *Computers in Biology and Medicine*, *129*(July 2020), 104139. https://doi.org/10.1016/j.compbiomed.2020.104139

Mukhopadhyay, A. K., Majumder, S., & Chakrabarti, I. (2022). Systematic realization of a fully connected deep and convolutional neural network architecture on a field programmable gate array. *Computers and Electrical Engineering*, *97*(October 2020), 107628. https://doi.org/10.1016/j.compeleceng.2021.107628

Muralidharan, N., Gupta, S., Prusty, M. R., & Tripathy, R. K. (2022). Detection of COVID19 from X-ray images using multiscale Deep Convolutional Neural Network. *Applied Soft Computing*, *119*, 108610. https://doi.org/10.1016/j.asoc.2022.108610

Qi, H., Xiao, S., Shi, R., Ward, M. O., Chen, Y., Tu, W., Su, Q., Wang, W., Wang, X., & Zhang, Z. (2018). DEEP MULTI-SCALE VIDEO PREDICTION BEYOND MEAN SQUARE ERROR. In *Nature* (Vol. 388, pp. 539–547).

Suartika E. P, I. W., Wijaya, A. Y., & Soelaiman, R. (2016). Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) Pada Caltech 101. *Jurnal Teknik ITS*, *5*(1), 76. http://repository.its.ac.id/48842/

Sun, J., Li, X., Tang, C., Wang, S. H., & Zhang, Y. D. (2021). MFBCNNC: Momentum factor biogeography convolutional neural network for COVID-19 detection via chest X-ray images[Formula presented]. *Knowledge-Based Systems*, *232*, 107494. https://doi.org/10.1016/j.knosys.2021.107494

Udayana, I. P. A. E. D., Sudarma, M., & Ariyani, N. W. S. (2021). Detecting Excessive Daytime Sleepiness With CNN And Commercial Grade EEG. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, *12*(3), 186. https://doi.org/10.24843/lkjiti.2021.v12.i03.p06

Upadhyay, P. K., Rastogi, S., & Kumar, K. V. (2022). Coherent convolution neural network based retinal disease detection using optical coherence tomographic images. *Journal of King Saud University - Computer and Information Sciences*, *xxxx*. https://doi.org/10.1016/j.jksuci.2021.12.002

Yang, L., Chen, W., Liu, W., Zha, B., & Zhu, L. (2020). Random Noise Attenuation Based on Residual Convolutional Neural Network in Seismic Datasets. *IEEE Access*, *8*(12), 30271–30286. https://doi.org/10.1109/ACCESS.2020.2972464