IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 3, June 2026, pp. 2238∼2246
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i3.pp2238-2246

Fine-tuning convolutional neural network for artificial intelligence generated image detection enhancement

Steven Vincent Hendrawan, Moeljono Widjaja, Alethea Suryadibrata
Department of Informatics, Faculty of Engineering and Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia

Article Info

Article history:

Received Feb 5, 2025
Revised Mar 6, 2026
Accepted Apr 21, 2026

Keywords:

Artificial intelligence generated images
EfficientNet
Image detection
Machine learning
Xception

ABSTRACT

The relationship between art and technology has changed how people engage with creativity, leading to the industrialization of the field. Various digital media have been utilized in the endeavour of art creation, such as artificial intelligence (AI) generation for images. The utilization of AI-generated art has yielded negative reactions due to its exploitative nature on pre-existing artworks without the creator's consent, which raises plagiarism concerns. This research utilized convolutional neural network (CNN) to help detect such images to reduce public concerns on the abuse of AI images. The algorithm is proposed to detect such images as it involves spatial convolution within two-dimensional spaces, matching the nature of images. The model was developed from pre-existing architectures, namely EfficientNetB1 and Xception, which was pre-trained on ImageNet classification task with the modification of inclusion or exclusion of dropout in the top layer. After assessing the models, removing top layer dropout from EfficientNetB1 model improved it to reach the F1-score of 97.66% compared to 97.44% in the base model and Xception with a dropout layer yields lower F1-score of 95.56% compared to 97.07% in the base model.

Corresponding Author:

Alethea Suryadibrata
Department of Informatics, Faculty of Engineering and Informatics, Universitas Multimedia Nusantara
Tangerang, Indonesia
Email: alethea.suryadibrata@lecturer.umn.ac.id

1. INTRODUCTION

Technology and art are two fields that have been inseparable in human lives. Recent digitalization of media has increased the affordability of artworks to be enjoyed by humans. The access to human-made works can now be done with the assistance of the internet [1]. This is due to the art and design being effective instruments in conveying messages with the intent of information and communication [2]. Art and design are built on concepts that allow humans to perceive information more effectively within the scope of information communication technology (ICT). The involvement of art and design within ICT allows technologies to be more accessible as it assists users in navigating the usage of technologies in order to perform certain activities [3]. Meanwhile, ICT also supports the digitalization of art by acting as a medium for developing, preserving, and sharing works for the public, allowing new methods and reach in accessibility and appreciation of art [4]. That realization allows design to be an essential intermediary component for human-computer interaction [5].

The relationship has led to a rise of motivation for entrepreneurship in the art field, forming a connection that subsequently developed into the creative industry [6]. Humans utilize this advancement to develop ideas, creations, and innovations in the form of intellectual property. Professionals who intend to create a legacy through art acknowledge the importance of digitalization in their work and success, as it records the process and results of their work for a wider audience [7]. The need for art and design has proved its role to be enjoyed not only for entertainment purposes but also as a medium of conveying messages due to the interactive nature of contemporary art [8]. The interest, coupled with the rise of the creative industry, increased the need to pursue the knowledge of art and design academically [9].

The development of technology in art has also allowed artists in their endeavours to create art efficiently as well as experiments that result in the creation of contemporary arts [10]. Current artworks have been done via multitudes of mediums, such as application design, 3D sculpted art modeled within augmented or virtual reality spaces, as well as the recent increase usage of artificial intelligence (AI) for art generations. Despite the fact that the concept has existed since 1973, the increase in generative adversarial networks usage in 2014 has increased the popularity and research of image generation [11].

The rise of AI-generated art, or known as AI art, has received the attention of society, with some people having a positive outlook in terms of technological development while others seem to harbour a negative stigma over the topic of its socioeconomic and copyright issues [12]. This is due to the fact that current generated works have been done by stochastic reshuffling of pre-existing works [13]. Reuse of those works is done without the creator's consent and is even used within a commercial level, which leads to the concern of professionals and aspiring artists about their endangered livelihood due to the existence of AI art [14]. Considering the nature of AI art's creation process, the generated work plagiarizes original contents used to train the model. Plagiarism as is has raised various ethical concerns such as commercial copyright and academic integrity [15]. This affects the state of the industry and academia for art and design, which creates the need to distinguish AI art from human work.

There were limited works on the topic prior, as the need to detect AI-generated works is still recent. However, there have been a few researches and projects that started investigating the subject. A project dubbed as Synthbuster utilized spectral analysis with generic training to the cross-difference filter and achieved the accuracy of 87.2% [16]. Another research tuned the ImageNet pre-trained model EfficientNetB4 to the common objects in context (COCO)-Stuff dataset, which yielded an average accuracy of 92% through different datasets [17]. However, neither of the previous researches explored differing learning rates in the development of learning rates to find the better rate as the hyperparameter. This research aims to explore the usage of convolutional neural network (CNN) in detecting AI-generated images as well as the impact of learning rates to enhance the detection of such images.

This research utilized a deep learning algorithm to detect AI-generated images, which are known for their capabilities to speed up the detection process, as well as the easily implementable techniques available [18]. The aforementioned deep learning algorithm for this research is the CNN due to its ability to recognize patterns within images with a considerable reduction of neurons necessary by spatial convolution within two-dimensional spaces [19]. This makes the CNN capable of extracting features within the input that act as building blocks for higher-level features in deeper layers, which are then used to decipher global features in the deepest layer [20]. The algorithm has been proven to be more robust in recognizing complex patterns, even distorted images [21]. These patterns would be apparent between the stochastically randomized AI-generated works and the consistent strokes of human-made works and imagery depictions of reality.

This research favours the CNN algorithm as it is fitting for the nature of the images. CNN is more focused on the relations between pixels in an image instead of the grand image, as the grand image may be similar for both image types. For comparison, vision transformers (ViT) have been proven to be able to detect certain objects better than CNNs in an image [22]. However, what is being dealt with the AI recreation of images are patterns, which ViT are inferior compared to CNNs, proven by a research comparing the two algorithms for digital holography where ViT yields less performance as it tries to look globally inside the image [23]. This makes ViT models less ideal in the practical use, as model should be accessible to public.

The CNN architectures utilized in this research are EfficientNetB1 and Xception, which yielded similar results on the ImageNet dataset on which the models were trained prior. The EfficientNet models utilized mobile inverted-bottleneck architectures built of convolution layers, batch normalization, and activation layers. The architecture, as the name suggests, aimed to achieve better efficiency with fewer parameters and operations [24]. Xception, on the other hand, utilizes depth-wise separable convolutions and batch normalization to improve parameters similar to InceptionV3, on which Xception was improved from, and aims to introduce this component in further developments of CNN algorithms [25]. Batch normalization acts as one of the regularization methods that can prevent overfitting for the model and allows the model to be more stable [19]. This research is conducted with the goal of enhancing a method that can be implemented in detecting AI-generated images. Hence, it aims to detect AI-generated images and human-made images, as it will assist people in ensuring fair attribution and compensation of creation, be it academic (grades) or commercial (monetary).

2. METHOD

This research first established its foundation by defining the problem of AI-generated image detection and selecting the CNN models EfficientNetB1 and Xception. Following the cementing of these core topics, the work was then followed by conducting the research, which was defined into seven phases: data collection, data preprocessing, data splitting, model training, model evaluation, hyperparameter tuning, and result analysis. This section will further elaborate on the processes involved in each step. The previously mentioned research process is illustrated in Figure 1 for visual clarity.

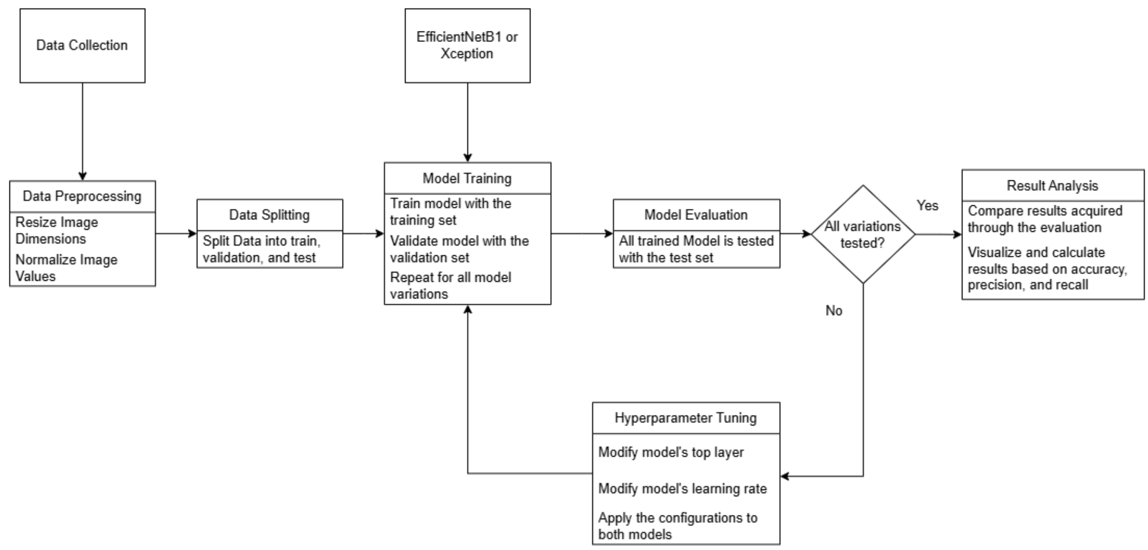


Figure 1. Proposed research workflow

This research was conducted on the AI or human art classification dataset publicly available on Kaggle. The dataset was selected as it had been pre-labeled with 10,330 AI-generated images and 8,288 human-created images. The dataset was majorly sourced from the AI or Not competition in Hugging Face, which hadn't mentioned the source of the generated images [26]. However, there are images sourced from the image sharing site, Pinterest [27]. While specific images sourced from the site hadn't been identified, the addition should provide some variety that helps generalize the model with modern creations. Data are loaded into batches of 32, resulting in 582 batches. The loaded data are then resized into 256×256 dimensions and its values normalized within 0 to 1 on three colour channels to match the pre-trained models requirements. A few sample images available in the dataset utilized in this work are illustrated in Figure 2.

The resized data are then split into a training set consisting of 80% of the data, a validation set consisting of 10% of the data, and a test set containing the remaining 10%. The validation set is used to determine hyperparameters within the training process [28]. In this research, the validation set was utilized in validating the training process' cross-entropy to determine the epochs needed for the training to prevent model overfitting.

The data was then used to train the aforementioned models with rebuilding of the top layers. EfficientNetB1 utilizes a dropout in its top layer. On the other hand, Xception allows the fully-connected and dropout after global pooling to be optional [25]. Due to this, the research compared the two approaches of top layers on both models. Instead of the 1000 classes of ImageNet, the predictor was made of a fully-connected layer with two neurons representing the two classes of images: non-AI-generated and AI-generated. The layer utilized the softmax activation function, which is compatible with the global pooling on the top layers [29].

The model was also optimized with the adaptive moment estimation (Adam) estimator using six different learning rates, which are [1.0, 0.1, 0.01, 0.001, 0.0001, and 0.00001] to determine the learning rate that yields better metrics for each model. The optimizer used for the training, Adam, is capable of adjusting the learning rate for each parameter of the model. This predates the reason for the election of the learning rate, as there are significant differences in the learning rate values. These parameter variations will all be utilized in the training of the model.

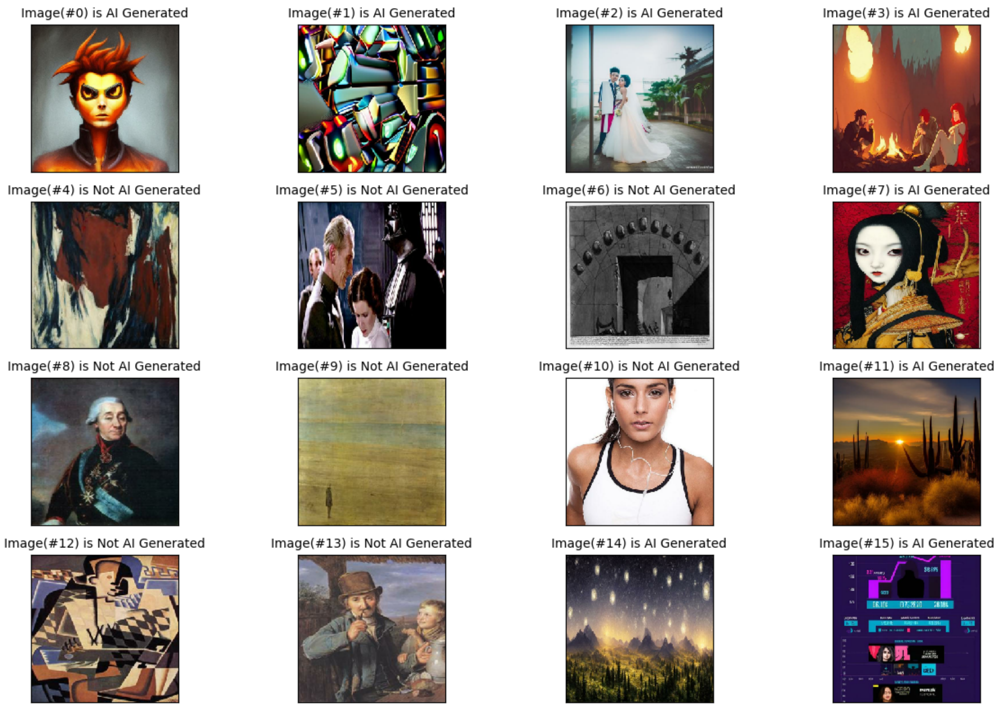


Figure 2. Sample dataset images

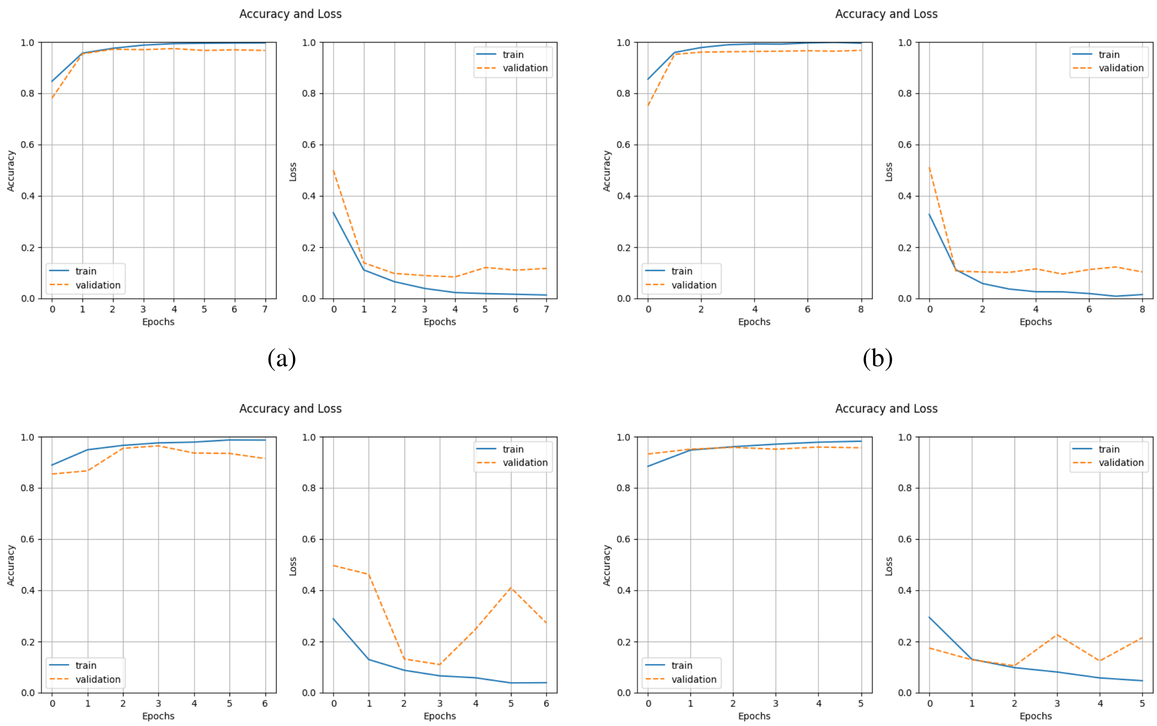
3. RESULTS AND DISCUSSION

This section covers the results of the model training phase and the subsequent phases. The training process is done by training the model to the train set and then validating it with the validation set. The validation results acts as the initial assessor whether the model still generalizes well or has overfit to the train set. Early stopping was used to determine and rollback to the optimal epoch of each model variant which is when the training and validation results converged best. The deterministic parameter for the convergence is the validation loss, as the model will keep training itself to increase its accuracy to the train data. However, this should also apply to the validation set in order for the model to generalize well. The loss would not only calculate the correct predictions but also the difference between the probability distributions of the prediction and actual labels, making it more comprehensive than accuracy alone. This process is done repeatedly on every configuration to both models to check and compare how they performed. The training phase of each model yielded the results depicted in Table 1.

The validation metrics achieved during the model training process in Table 1 provides information whether a model is able to generalize well throughout the dataset instead of overfitting to the training set. It is visible from the model training results that the validation metrics peaked on 10^{-4} learning rate for both variations of the EfficientNetB1 model and 10^{-3} learning rate for both variations of the Xception model and between said variations, the models excluding top layer dropout seem to be advantageous compared to their dropout alternatives. That being said, all learning rates are able to learn data somewhat except for 1.0 learning rate that failed for all model variants. The development for the highlighted learning rates (Figure 3) of each models are shown in Figures 3(a) and 3(b) for both EfficientNetB1 models as well as Figures 3(c) and 3(d) for Xception models.

Table 1. Model training metrics

Model	Learning rate	Train accuracy (%)	Train loss	Validation accuracy (%)	Validation loss
Without dropout					
EfficientNetB1	10^{-5}	98.76	0.0391	96.23	0.1015
	$\mathbf{10^{-4}}$	**99.09**	**0.0251**	**96.34**	**0.0944**
	10^{-3}	98.21	0.0441	95.91	0.1020
	10^{-2}	93.98	0.1422	88.63	0.2461
	10^{-1}	89.69	0.2562	88.25	0.2741
	1.0	51.60	0.7556	50.92	0.6937
Xception (Base model)	10^{-5}	98.40	0.0539	94.50	0.1409
	10^{-4}	99.51	0.0142	96.07	0.1474
	$\mathbf{10^{-3}}$	**97.49**	**0.0653**	**96.34**	**0.1093**
	10^{-2}	91.08	0.2040	91.16	0.2251
	10^{-1}	89.92	0.2396	87.77	0.3612
	1.0	50.95	0.7117	55.93	0.6877
With dropout					
EfficientNetB1 (Base model)	10^{-5}	98.21	0.0551	96.66	0.0948
	$\mathbf{10^{-4}}$	**99.29**	**0.0223**	**97.36**	**0.0831**
	10^{-3}	98.04	0.0507	97.14	0.0963
	10^{-2}	95.46	0.1094	94.94	0.1484
	10^{-1}	58.49	1.5483	57.81	0.8434
	1.0	53.63	0.7326	54.20	0.7041
Xception	10^{-5}	98.20	0.0607	94.29	0.1459
	10^{-4}	90.58	0.2319	94.34	0.1580
	$\mathbf{10^{-3}}$	**95.99**	**0.0973**	**95.74**	**0.1048**
	10^{-2}	92.29	0.1778	93.27	0.1695
	10^{-1}	53.59	1.2624	55.71	0.6928
	1.0	52.88	0.7010	55.87	0.6869



(a) (b)

(c) (d)

Figure 3. Model training metric development, (a) base EfficientNetB1 model, (b) EfficientNetB1 model without top layer dropout, (c) base Xception model, and (d) Xception model with top layer dropout

The amount of epochs is determined thanks to the utilization of the validation set where early stopping is initiated accordingly by the development of the validation loss. EfficientNetB1 generally requires less training time compared to Xception. However, EfficientNetB1 with 10^{-5} learning rate required over twice epochs compared to the other learning rates which subsequently results in the dilation of its training time while Xception itself doesn't suffer from this condition as significantly. The models are then evaluated to the test set to observe how well they generalize outside the training and validation sets.

The model achieved from the training process is then evaluated to the test set by predicting the class of the data within it. The predicted and actual classes were mapped into a confusion matrix, which subsequently provided the information necessary to compute the model's accuracy, precision, recall, and f1-score. The evaluation results showed that both models achieved peak performance at the learning rate of 10^{-4} without dropout and 10^{-3} with dropout, the former indicating superior overall metrics. It is also visible that EfficientNetB1 without dropout trained with the learning rate of 10^{-4} is a victor within all models due to the model achieving best metrics during both training process and evaluation process. On top of that, EfficientNetB1 models generally performed better compared to their Xception counterparts on both metrics and time consumed. The complete metrics of the model testing are shown in Table 2.

The best hyperparameter configurations of each variant of the models are then utilized to train them using the recommended dimensions to see whether there is a significant impact from the difference in dimensions within the data as the research utilized different image resolution than the recommended resolution. Both models, with the recommended input dimensions, produced similar training results to those of the proposed dimensions used in this research. There is, however, a significant difference in the training time especially for Xception models, which reached over 100 seconds per epoch difference due to the relatively bigger data dimensional difference compared to EfficientNetB1's data dimensional difference, which allowed its difference to be barely about 5 seconds. This result is consistent with the statement that EfficientNet models are developed with the aim of efficiency in [29].

The evaluation results show that both models without achieved the best performance for this task when trained with 10^{-4} learning rate, insinuating that between the six proposed rates, 10^{-4} is the most appropriate learning rate for the task. This is in tandem with Krichen's idea that a lower learning rate may be beneficial to ensure the model converges effectively [19]. However, it is important to not put the rate too low as it may converge too early as the training utilized early stopping and that might result in the training stopping before it reaches a better converge point, which is visible when the models are trained with 10^{-5} learning rate compared to their 10^{-4} rate counterparts.

The models reached better results without a top layer dropout when the models are trained with 10^{-5} to 10^{-3} learning rate, yet adding top layer dropout results in improvement on 10^{-2} learning rate. This implies that the model starts to overfit on that learning rate. However, the dropout layer broke the model when trained with higher learning rates, rendering the impact of dropout for those variants inconclusive. This may point out that with 10^{-5} to 10^{-3} learning rate, batch normalization and convolutional dropouts are sufficient to prevent overfitting of the model, as the model is able to converge stably. While on higher learning rates, top dropout layer is justifiable to use as the model overfits more quickly as it learns faster.

Based on the results of the learning rate and regularization, hyperparameters of the model training significantly impact the model's performance. The necessity of certain regularizers depends on the other parameters of the model training. Considering the model's complexity being one of the smaller ones, especially EfficientNetB1 which is much simpler EfficientNet models. The result aligns with Krichen's statement that dropouts may be necessary in deeper models and other regularization methods to be sufficient in more shallow models [19]. Both architectures being robust enough that dropouts are less warranted is also supported by Zhao's idea that batch normalization improves generalization to maximize the architecture's performance [20].

Furthermore, in both proposed and innate recommended dimensions, it is concludable that the models are applicable to different dimensions and achieve pleasing results regardless. There are a multitude of versions of the EfficientNet model for varying image sizes, with the EfficientNetB1 utilizing the most similar dimensions to the data used in this research. In comparison to the prior research with EfficientNetB4, while the EfficientNetB4 performed better in certain datasets, EfficientNetB1's structure is miniature compared to EfficientNetB4, allowing it to be more suitable to use within relatively constrained environments while still providing accurate results.

Table 2. Model evaluation metrics on test set

Model	Learning rate	Accuracy (%)	Precision (1) (%)	Recall (1) (%)	Precision (0) (%)	Recall (0) (%)	F1-score (%)	Inference time
				Without dropout				
EfficientNetB1	10^{-5}	97.09	96.92	97.86	97.31	96.13	97.05	18.12s
	$\mathbf{10^{-4}}$	**97.68**	**98.51**	**97.26**	**96.69**	**98.20**	**97.66**	**17.69s**
	10^{-3}	96.66	95.95	98.17	97.61	94.75	96.60	18.66s
	10^{-2}	87.66	91.97	84.80	83.30	91.10	87.51	17.75s
	10^{-1}	89.06	88.80	90.70	89.37	87.23	89.01	19.01s
	1.0	53.18	68.47	33.40	47.27	79.52	52.10	17.83s
Xception	10^{-5}	94.94	95.66	95.48	93.97	94.21	94.83	18.09s
(Base model)	$\mathbf{10^{-4}}$	**97.09**	**96.94**	**97.71**	**97.27**	**96.36**	**97.07**	**17.69s**
	10^{-3}	96.34	95.44	98.18	97.57	93.96	96.26	17.99s
	10^{-2}	91.65	92.14	92.94	91.02	90.02	91.53	18.59s
	10^{-1}	86.37	83.11	93.69	87.78	77.86	85.30	17.84s
	1.0	54.90	54.90	100	0	0	35.44	18.80s
				With dropout				
EfficientNetB1	10^{-5}	95.96	94.96	97.85	97.26	93.65	95.90	17.55s
(Base model)	10^{-4}	97.14	97.48	97.39	96.72	96.84	97.11	18.41s
	$\mathbf{10^{-3}}$	**97.47**	**96.71**	**98.71**	**98.42**	**95.98**	**97.44**	**18.24s**
	10^{-2}	95.15	93.06	98.53	98.07	91.03	95.07	17.80s
	10^{-1}	55.55	55.55	100	0	0	35.71	18.17s
	1.0	59.54	58.44	99.71	94.64	6.62	43.03	17.69s
				With dropout				
Xception	10^{-5}	95.31	96.22	95.18	94.24	95.48	95.28	18.53s
	10^{-4}	93.80	90.63	99.02	98.64	87.36	93.65	18.00s
	$\mathbf{10^{-3}}$	**95.64**	**95.35**	**96.91**	**96.01**	**94.02**	**95.56**	**17.94s**
	10^{-2}	92.94	95.05	91.74	90.63	94.36	92.91	18.75s
	10^{-1}	54.90	54.90	100	0	0	35.44	18.34s
	1.0	54.20	54.20	100	0	0	35.15	20.08s

4. CONCLUSION

A CNN model to detect AI-generated images has been created, which classifies two classes as AI-generated images and non-AI-generated images. The EfficientNetB1 model, which proposed elimination of dropout in the top layer and trained with a learning rate of 10^{-4} reached a precision of 97.68%, precision of 97.60%, recall of 97.73% and a score of F1 of 97.66% in the test set. However, the base Xception model performed better than the model that incorporates a dropout in the top layer. Training the model on a learning rate of 10^{-4} yielded 97.09% precision, 97.11% precision, 97.04% recall, and F1-score of 97.07%. Aside from the superior results with EfficientNetB1, the model also requires less training time, which made it the preferred architecture for this task. Despite the aforementioned differences, both models were able to perform well in this task. The results show that the models are best trained without dropout of the top layer but with a learning rate of 10^{-4}, and both models yield better metrics compared to previous works. In the future, the research in this area can be improved by adding more specific classes regarding the creation medium on both human-made and AI model generation as well as developing a model which is capable of evaluating images on specific parts that are considered AI-generated and the reason behind the detection.

ACKNOWLEDGMENTS

The authors acknowledge the support provided by Universitas Multimedia Nusantara during this research.

FUNDING INFORMATION

The authors state no funding is involved in this work.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Steven Vincent Hendrawan	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Moeljono Widjaja		✓		✓	✓		✓			✓	✓	✓	✓	
Alethea Suryadibrata	✓			✓	✓			✓		✓	✓	✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal Analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject Administration
Fu : **Fu**nding Acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data used in this work was obtained from the AI or human art classification dataset, which can be accessed through Kaggle at https://www.kaggle.com/datasets/kausthubkannan/ai-and-human-art-classification.

REFERENCES

[1] T. Sahu, A. Tyagi, S. Kumar, and A. Mittal, "Classification and aesthetic evaluation of paintings and artworks," in *2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, Dec. 2017, pp. 179–183, doi: 10.1109/SITIS.2017.39.

[2] G. Lughi, "Digital media and contemporary art," *Mimesis Journal*, vol. 3, no. 2, pp. 43–52, 2014, doi: 10.4000/mimesis.686.

[3] B. D. R. Mariño, M. J. R. FóRtiz, M. V. H. Torres, and H. M. Haddad, "Accessibility and activity-centered design for ICT users: ACCESIBILITIC ontology," *IEEE Access*, vol. 6, pp. 60655-60665, 2018, doi: 10.1109/ACCESS.2018.2875869.

[4] C. Cai, "Virtual reality enables the dissemination and preservation of early works of art," *Applied Mathematics and Nonlinear Sciences*, vol. 9, no. 1, Jan. 2024, doi: 10.2478/amns-2024-0728.

[5] A. Pitale and A. Bhumgara, "Human computer interaction strategies — designing the user interface," in *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Nov. 2019, pp. 752–758, doi: 10.1109/ICSSIT46314.2019.8987819.

[6] M. Abbasi, P. Vassilopoulou, and L. Stergioulas, "Technology roadmap for the creative industries," *Creative Industries Journal*, vol. 10, no. 1, pp. 40–58, Jan. 2017, doi: 10.1080/17510694.2016.1247627.

[7] D. Habsary, A. Kurniawan, and I. Bulan, "Digitalization of arts," in *Proceedings of the Tenth International Conference on Languages and Arts (ICLA 2021)*, 2021, pp. 246–250, doi: 10.2991/assehr.k.211129.039.

[8] X. Kang, W. Chen, and J. Kang, "Art in the age of social media: Interaction behavior analysis of Instagram art accounts," *Informatics*, vol. 6, no. 4, Dec. 2019, doi: 10.3390/informatics6040052.

[9] A. Sarkar, "Exploring perspectives on the impact of artificial intelligence on the creativity of knowledge work: beyond mechanised plagiarism and stochastic Parrots," in *Proceedings of the 2nd Annual Meeting of the Symposium on Human-Computer Interaction for Work*, Jun. 2023, pp. 1–17, doi: 10.1145/3596671.3597650.

[10] K. Düzenli and N. Z. Perdahçı, "The role of digitalization in today's art: a perspective from NFT and artificial intelligenc," *Journal of Arts*, vol. 7, no. 1, pp. 43–59, Feb. 2024, doi: 10.31566/arts.2291.

[11] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: an overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, Jan. 2018, doi: 10.1109/MSP.2017.2765202.

[12] S. Bosonogov and A. Suvorova, "Perception of AI-generated art: text analysis of online discussions," *Journal of Mathematical Sciences*, vol. 285, no. 1, pp. 1–13, Oct. 2024, doi: 10.1007/s10958-024-07418-0.

[13] A. Daniele and Y. Z. Song, "AI + Art = Human," in *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, Jan. 2019, pp. 155–161, doi: 10.1145/3306618.3314233.

[14] J. Oppenlaender, J. Silvennoinen, V. Paananen, and A. Visuri, "Perceptions and realities of text-to-image generation," in *26th International Academic Mindtrek Conference*, Oct. 2023, pp. 279–288, doi: 10.1145/3616961.3616978.

[15] P. I. Fusch, L. R. Ness, J. M. Booker, and G. E. Fusch, "The ethical implications of plagiarism and ghostwriting in an open society," *Journal of Social Change*, vol. 9, no. 1, Jan. 2017, doi: 10.5590/JOSC.2017.09.1.04.

[16] Q. Bammey, "Synthbuster: towards detection of diffusion model generated images," *IEEE Open Journal of Signal Processing*, vol. 5, pp. 1–9, 2024, doi: 10.1109/OJSP.2023.3337714.

[17] S. S. Baraheem and T. V. Nguyen, "AI vs. AI: can AI detect AI-generated images?," *Journal of Imaging*, vol. 9, no. 10, Sep. 2023, doi: 10.3390/jimaging9100199.

[18] N. H. Shabrina, R. A. Lika, and S. Indarti, "Deep learning models for automatic identification of plant-parasitic nematode," *Artificial Intelligence in Agriculture*, vol. 7, pp. 1–12, Mar. 2023, doi: 10.1016/j.aiia.2022.12.002.

[19] M. Krichen, "Convolutional neural networks: a survey," *Computers*, vol. 12, no. 8, Jul. 2023, doi: 10.3390/computers12080151.

[20] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, Mar. 2024, doi: 10.1007/s10462-024-10721-6.

[21] A. Archilles and A. Wicaksana, “Vision: a web service for face recognition using convolutional network,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 3, Jun. 2020, doi: 10.12928/telkomnika.v18i3.14790.

[22] P. Mittal, B. Sharma, and D. P. Yadav, “Comparative analysis between CNN and ViT using brain MRI dataset,” in *2024 Eighth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, Dec. 2024, pp. 290–295, doi: 10.1109/PDGC64653.2024.10984339.

[23] S. Cuenat and R. Couturier, “Convolutional neural network (CNN) vs vision transformer (ViT) for digital holography,” in *2022 2nd International Conference on Computer, Control and Robotics (ICCCR)*, Mar. 2022, pp. 235–240, doi: 10.1109/ICCCR54399.2022.9790134.

[24] M. Tan and Q. Le, “EfficientNet: rethinking model scaling for convolutional neural networks,” in *Proceedings of the 36th International Conference on Machine Learning*, May 2019, pp. 6105–6114.

[25] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 1800–1807, doi: 10.1109/CVPR.2017.195.

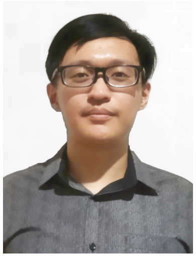
[26] Hugging Face, ”Datasets: competitions/aiornot,” huggingface.co. Accessed: Apr. 10, 2024. [Online]. Available: https://huggingface.co/datasets/competitions/aiornot

[27] K. Kannan, ”AI and human art classification,” kaggle.com. Accessed: Apr. 4, 2024. [Online]. Available: https://www.kaggle.com/datasets/kausthubkannan/ai-and-human-art-classification

[28] R. K. Pandey, A. Kumar, and A. Mandal, “A robust deep structured prediction model for petroleum reservoir characterization using pressure transient test data,” *Petroleum Research*, vol. 7, no. 2, pp. 204–219, Jun. 2022, doi: 10.1016/j.ptlrs.2021.09.003.

[29] M. Lin, Q. Chen, and S. Yan, “Network in network,” 2013, *arXiv:1312.4400*.

BIOGRAPHIES OF AUTHORS



Steven Vincent Hendrawan earned his Bachelor of Informatics from Universitas Multimedia Nusantara, Indonesia in 2024. During his studies, he pursued a non-degree semester through a student exchange program during his bachelor studies at the University of Leeds, United Kingdom. This paper documented what he worked on for his bachelor’s thesis. His area of interest is machine learning and virtual reality development. He can be contacted at email: steven.vincent2002@gmail.com.

Moeljono Widjaja received a doctoral degree in Electrical Engineering from Monash University (Australia). He obtained a bachelor degree in Electrical Engineering from Wayne State University (USA) in 1992 and a master degree in Electrical Engineering from the Ohio State University (USA) in 1994. He is an assistant professor at the Department of Informatics, Universitas Multimedia Nusantara. His main research interests are AI, simulation/modeling, and optimization. He developed a fuzzy controller for an inverted pendulum system and a fuzzy-based bidding strategy for generators in an electricity market. He has been working on intelligent energy management systems. He is a professional member of ACM and IEEE. He can be contacted at email: moeljono.widjaja@umn.ac.id.

Alethea Suryadibrata holds a Master of Engineering degree from Silla University, South Korea (2017), and earned her Bachelor of Informatics from Universitas Multimedia Nusantara (UMN), Indonesia, in 2014. Her research expertise lies in the field of AI, with a particular focus on clustering methods and machine learning applications in biomedical imaging and game-based learning. She is currently a lecturer in the Department of Informatics at UMN. She can be contacted at email: alethea.suryadibrata@lecturer.umn.ac.id.