УДК 004.93:004.032.26 https://doi.org/10.15407/vidbir2024.52.087

RECOGNITION OF SYNTHESIZED IMAGES USING MODIFIED CONVOLUTIONAL NEURAL NETWORK MODEL VGG16

D. V. Matei [1], I. B. Ivasenko [1,2]

[1] Lviv Polytechnic National University, Lviv;

[2] H. V. Karpenko Physico-Mechanical Institute of the NAS of Ukraine, Lviv

E-mail: ivasenko.iryna@gmail.com

This paper presents a new approach to recognizing synthesized images using transfer learning, specifically the VGG16 model. With the growing prevalence of AI-generated content on social media and the increasing use of synthesized images for fraudulent purposes, the ability to accurately distinguish between real and synthesized images is of utmost importance. The study addresses the limitations of existing image recognition technologies, which often have difficulty when working with high-quality images created by AI. The proposed method uses a custom-made dataset of more than 200 000 images, balanced between AI-generated and real images of several classes, to train the model. By fine-tuning the VGG16 model and unfreezing all layers, this approach achieves great accuracy. Experimental results show that the model achieves an overall accuracy of 97%, compared to 93% accuracy of baseline model, indicating its effectiveness in distinguishing between real and synthesized images. However, shortcomings such as slight overfitting are noted, and suggestions for future improvement include regularization techniques and exploring more advanced architectures and techniques. This research highlights the potential of transfer learning in developing robust solutions for synthesized image recognition.

Keywords: *deep learning, image classification, fraud detection, synthesized image recognition, transfer learning, VGG16, AI-generated content.*

РОЗПІЗНАВАННЯ СИНТЕЗОВАНИХ ЗОБРАЖЕНЬ ЗА ДОПОМОГОЮ МОДИФІКОВАНОЇ МОДЕЛІ ЗГОРТКОВОЇ НЕЙРОМЕРЕЖІ VGG16

Д. В. Матей [1], І. Б. Івасенко [1,2]

[1] Національний університет “Львівська політехніка”;

[2] Фізико-механічний інститут ім. Г. В. Карпенка НАН України, Львів

Запропоновано новий підхід до розпізнавання синтезованих зображень за допомогою трансферного навчання, зокрема, моделі VGG16 з глибокою архітектурою, яка добре зарекомендувала себе в задачах класифікації зображень. З поширенням контенту, створеного штучним інтелектом у соціальних мережах, виникає потреба у вдосконалених методах розпізнавання. Синтезовані зображення використовують не лише для творчих або розважальних цілей, але нерідко вони стають інструментом для шахрайства та поширення дезінформації. Це створює серйозні виклики для технологій безпеки та контролю інформації. Звернено увагу на обмеження існуючих технологій розпізнавання зображень, які часто стикаються з труднощами під час роботи з високоякісними синтезованими зображеннями. Сучасні генеративні моделі, зокрема генеративні змагальні мережі та перетворювачі, здатні створювати надзвичайно реалістичні зображення, через що розпізнати їх від реальних складно. Багато традиційних моделей класифікації не враховують специфіку синтезованих зображень, що обмежує їхню ефективність. Запропонований метод використовує спеціально створений набір даних, який охоплює понад 200 000 реальних зображень та зображень, синтезованих штучним інтелектом таких поширених класів, як тварини, транспортні засоби, рослини, люди, будівлі тощо. Це дає можливість моделі навчатися на різноманітних даних та уникати дисбалансу. Важливим аспектом цього підходу є застосування трансферного навчання, яке уможливлює використання заздалегідь натренованих моделей для розв’язання нових задач. Налаштувавши модель VGG16 та розблокувавши всі її шари, можна досягти високих точності (97%) та продуктивності. Це помітно перевищує показники базової моделі, точність якої ~ 93%. Одна з основних проблем – незначне перенавчання моделі на навчальному наборі даних, що може свідчити про потребу в додаткових методах регуляризації. У подальших дослідженнях доцільно

вивчити складніші архітектури глибокого навчання або використовувати ансамблеве навчання для поліпшення результатів. Виконане дослідження підкреслює значний потенціал технологій трансферного навчання у розробці надійних рішень для задачі розпізнавання синтезованих зображень.

Ключові слова: *глибоке навчання, класифікація зображень, виявлення шахрайства, розпізнавання синтезованих зображень, трансферне навчання, модель VGG16, синтезований штучним інтелектом контент.*

Introduction. With the growing influence of social media and increasing use of synthesized images for deception and fraud, there is a need to develop an effective synthesized image recognition system. Social media has become a platform for the dissemination of synthesized images used for fraud, disinformation and other negative practices. The lack of reliable mechanisms for recognizing synthesized images jeopardizes the security and reliability of information in the online environment.

The problem of recognizing synthesized images is an urgent scientific issue. Existing technologies for recognizing fake images have certain limitations which affect their effectiveness. For example, machine learning algorithms often have difficulty in recognizing high-quality synthesized images created using technologies based on generative adversarial networks. Existing solutions are often not easily accessible to the common user due to their complexity and high cost. Due to the rapid development of AI, simpler and more affordable solutions quickly become obsolete.

Existing solutions. Reviewing current state of the problem of recognition of synthesized images, first we will consider the known methods of image synthesis [1], then go over the most prominent solutions that combat the threat.

Generative adversarial networks are one of the most widely used image generation techniques [2]. This architecture includes a generator and a discriminator that compete: the generator tries to create realistic images, while the discriminator determines whether an image is real or synthesized.

Variational autoencoders are another popular image generation technique [3]. They are based on the concept of the variational Bayesian approach and learn to model a distribution in a vector space.

Deep neural networks, in particular convolutional neural networks, are often used for image generation [4]. These networks can learn hierarchical features and relationships. Transformer-based models, such as GPT (Generative Pre-trained Transformer), have become popular for text and image generation [5]. They utilize attention mechanisms to process context efficiently.

Stylization using neural style transfer is a technique that uses neural networks to transfer the artistic style of one image to another [6]. It allows creation of unique and artistically designed images by changing their characteristics under the influence of a known artistic style.

Deep neural networks are one of the best choices for synthesized image recognition [7]. Models can be trained based on various characteristics such as texture features, object structure, and pixel statistics. Another approach is artifact analysis [8]. It consists of recognizing artifacts and anomalies in pixel distribution which are specific for certain synthesis methods.

File metadata may indicate image processing or synthesis [9]. This approach is to check information about the device on which the image was captured, as well as other parameters. Analyzing anomalies in the frequency space and recognizing unusual characteristics in the frequency properties of an image can help in spotting synthesis [10].

Detection of inconsistencies in the interaction of objects by analyzing the context of an image and the relative position of objects can identify mismatch in the context of the scene [11]. Use of pre-trained models to recognize synthesized images is another option [12].

Use of pre-trained VGG16 model. Convolutional neural networks are the main tool for image analysis in modern machine learning [13]. They consist of convolutional, subsampling, and fully connected layers which allow for automatic feature extraction at different levels of abstraction. Main advantages of convolutional neural networks are the ability to automatically learn to extract important features from images and recognize objects regardless of their size, position, or orientation.

Transfer learning is one of the key methods in modern machine learning that allows us to adapt already trained models for new tasks [14]. It significantly increases the efficiency and productivity of model development. Pre-trained models have already undergone lengthy and computationally expensive training on large datasets. Using these models as a basis, we can quickly adapt them to our specific tasks without having to train from square one. The amount of data required is also smaller.

For us, transfer learning is the optimal choice due to the efficient use of limited time and computing resources, and higher accuracy compared to solutions built from zero.

The VGG16, developed by researchers at the University of Oxford, is one of the most popular models for image classification [15]. It has 16 feedback layers, including 13 convolutional layers and 3 fully connected layers.

The VGG16 model was chosen in favour of its simple architecture, which provides high accuracy. It is easy to modify and customize and is well suited for the task at hand. This model is widely used in academic and industrial research, ensuring decent documentation and support.

Custom model for synthesized image recognition. To train this model, we decided to create a custom dataset that would include a wide range of images. A total of 200428 images were collected. Half of them were sampled from the DiffusionDB dataset from Kaggle [16]. The second half, real images, were collected from Open Images [17]. These real-world images were carefully selected to be of 10 different classes: animals, vehicles, plants, people, buildings, clothing, food, tools, furniture, and sports equipment.

Images were preprocessed by resizing them to 224 by 224 pixels and normalizing RGB values.

This approach to building the dataset was chosen to ensure that the model could generalize well and work with new, unfamiliar data in real-world conditions.

A prototype neural network was built based on a pre-trained VGG16 model.

First, we discarded the top classification layers. We unfroze the weights of all layers of VGG16. This way we will improve the speed and accuracy of training, by using the initial weight values, reducing the number of additional layers we would need to add.

Among these, 13 layers are convolutional [18]. The main operation in these layers is convolution, which is defined as follows [19]:

$$(f \cdot g)(t) = \int_{-\infty}^{\infty} f(\tau) \cdot g(t-\tau)\, d\tau\,. \tag{1}$$

For digital images, this translates into a discrete convolution [19]:

$$(f \cdot g)(i,\ j) = \sum_{m=-k}^{k} \sum_{n=-k}^{k} f(m,\ n) g(i-m,\ j-n)\,, \tag{2}$$

where f is the input image, g is the filter (convolution kernel), and (i, j) are the pixel coordinates.

Remaining layers are max pooling layers [18]. These layers reduce the size of spatial features while preserving important information [19]:

$$P(i,\ j) = \max\left\{x_{m,n} : (m,\ n) \in R(i,\ j)\right\}, \tag{3}$$

where $R(i,\ j)$ is the input region corresponding to the output pixel $(i,\ j)$.

After the base model, we added 4 custom layers [18]. Flatten layer converts the multidimensional output of the base model into a one-dimensional vector. Then a fully connected dense layer with 16 neurons and ReLU activation adds nonlinearity. This can be described as follows [19]:

$$y_i = \sigma\left(\sum_{j=1}^{N} w_{ij} x_j + b_i\right), \tag{4}$$

where y_i is the output signal of the neuron, x_j is the input signal, w_{ij} are the weighting coefficients, b_i is the offset, and σ is the activation function.

The Dropout layer with probability of 0.1 is used to prevent overfitting [19]:

$$h_i^{drop} = h_i \cdot d_i, \tag{5}$$

where h_i is the output of the neuron, d_i is a random variable which takes the value 0 with probability p and 1 with probability $1–p$.

The output layer is a fully connected dense layer with a single neuron and sigmoid activation. The model will be compiled using the Adam optimizer [19]:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_\theta J(\theta_t), \tag{6}$$

where θ are the model parameters, η is the learning rate, and $\nabla_\theta J(\theta_t)$ is the gradient of the loss function.

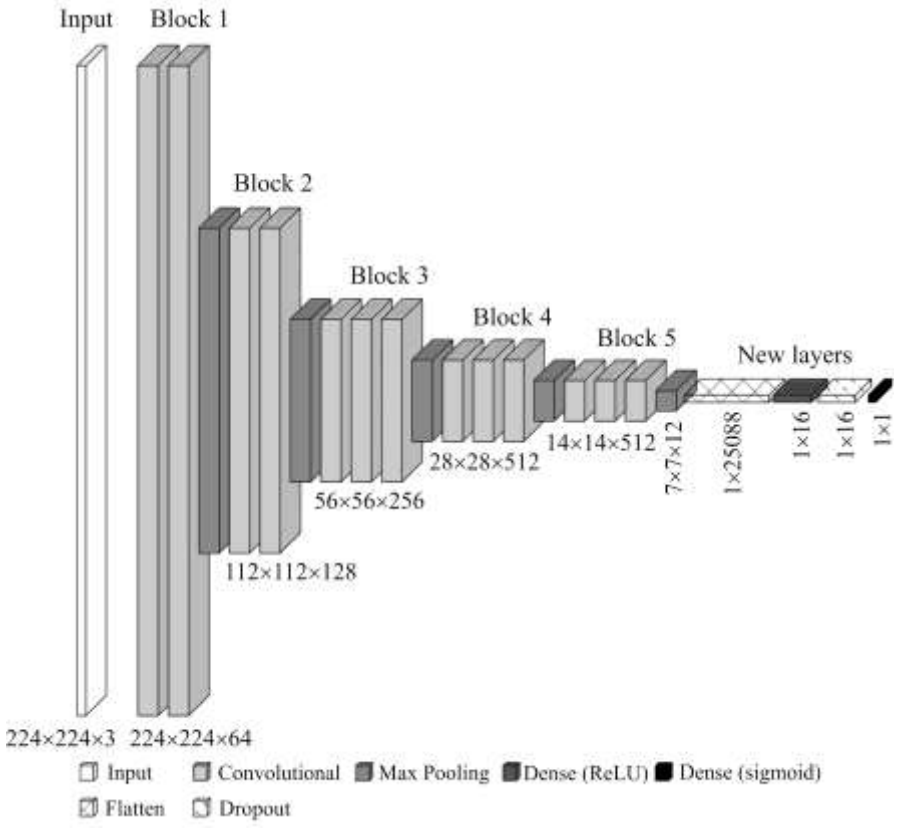


Fig. 1. Custom VGG16 based model diagram.

For the loss function, the binary cross entropy will be used [19]:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log \hat{y}_i + (1-y_i)\log(1-\hat{y}_i)], \quad (7)$$

where y_i are the true labels, $\hat{y}_i$ are the predicted probabilities.

The diagram of custom model is provided in Fig. 1. The model will then be trained on the data described in previous section for 10 epochs with a batch size of 32 and a test split of 0.3.

Experimental results. To determine the efficiency of our model we created a similar model from zero. The baseline model has similar layer structure and require about the same computational power.

Confusion matrix figures presented in Table 1 show that the custom model demonstrates symmetrical performance for both classes.

Table. 1. Comparison of confusion matrices

	Custom model		Baseline model	
	Predicted negative	Predicted positive	Predicted negative	Predicted positive
Real negative	4008	92	3882	218
Real positive	152	3848	336	3664

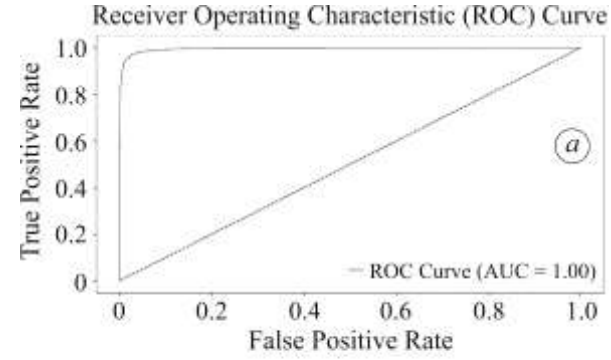
The errors are distributed almost equally between the classes, indicating that the model does not favor one class over the other. Custom model has a noticeably lower number of classification errors, especially for negative (real image) class, indicating better performance so far.

The precision of the custom model of 0.96 to 0.98 indicates a low rate of false positives. Recall of 0.96 to 0.98 indicates a low rate of missed positive cases. F1-score of 0.97 means that the model is balanced between false positives and negatives. The overall accuracy of the custom model is 97% as shown in Table 2. The custom model is again shown to be more consistent and has higher precision and recall value, while baseline model has a lower overall accuracy of 93%.

Table. 2. Comparison of classification reports.

	Custom model		Baseline model	
	Positive	Negative	Positive	Negative
Precision	0.96	0.98	0.92	0.94
Recall	0.98	0.96	0.94	0.92
F1-score	0.97	0.97	0.93	0.93
Accuracy	0.97		0.93	

The ROC curve of the custom model, shown in Fig. 2*a*, indicates high sensitivity and specificity. AUC of 1.00 indicates that the model has an almost perfect ability to distinguish between classes. Baseline model, while having shown decent results, has lower AOC value as shown in Fig. 2*b*.



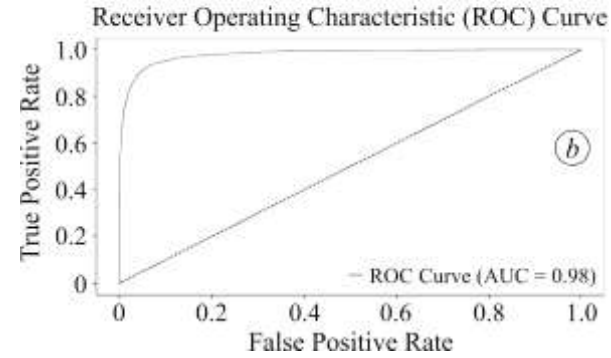


Fig. 2. ROC curve of custom model (*a*) and baseline model (*b*).

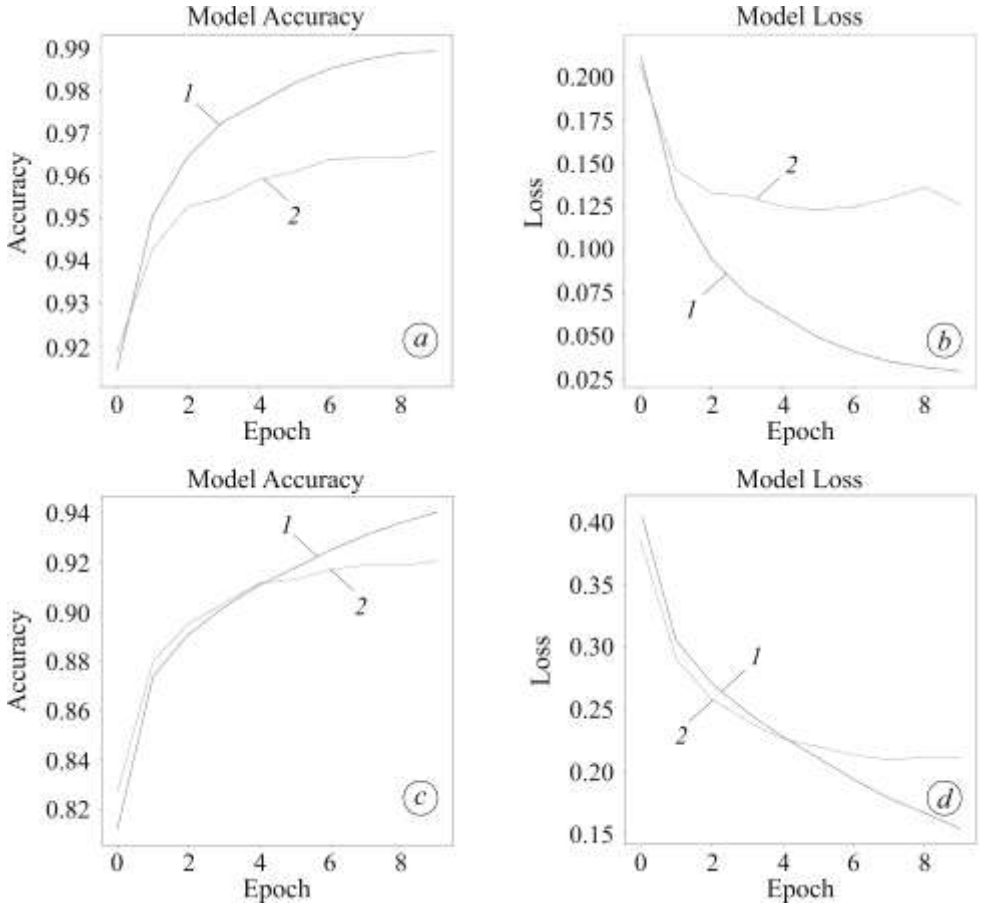


Fig. 3. Accuracy and loss graphs for custom (*a*, *b*) and baseline (*c*, *d*) models: *1* – Train, *2* – Validation.

Fig. 3*a* shows that for custom model the accuracy on the training set reaches almost 99%. However, on the validation set, the accuracy only reaches 96...97%. This indicates a slight overfitting.

Looking at Fig. 3*b*, we can see that losses on the training set decreased significantly during the first few epochs and remained low. The losses on the validation set were decreasing for the first 6 epochs but started fluctuating after. This is confirmation of a slight overfitting.

Baseline model has lower accuracy on both training and validation data, and while having smoother curves as seen in Fig. 3*c*, *d*, also shows signs of slight overfitting.

CONCLUSIONS

In this study, a transfer learning approach was applied to create a solution for synthesized image recognition. The use of VGG16 pre-trained architecture allowed us to create a custom model that demonstrates high accuracy. Comparisons with a baseline model, developed from scratch, showed clear advantages of the custom model in accuracy, F1 score, and the ROC curve, indicating its ability to effectively recognize real and synthesized images with low false positive rates.

The custom model achieved an accuracy of 97% for both classes, indicating its ability to correctly classify most images. It demonstrated a rather symmetrical performance for both classes, which ensures its reliability in different conditions. The AUC of 1.00 confirmed that the model distinguishes classes almost perfectly at different decision thresholds.

However, some shortcomings were found as well. The analysis of the training graphs indicates slight overfitting. The accuracy of the model when using the training data is slightly higher than the accuracy when using the validation data. While the difference of only 2% is acceptable, it still leads to a decrease in the ability of the model to generalize new examples.

These findings indicate new prospects for future research. Using regularization techniques, such as more aggressive dropout or L2 regularization, can help to reduce the risk of overfitting and increase the generalizability of the model. Consideration of other more powerful architectures such as ResNet50 [20] or EfficientNet [21] can further improve model accuracy and reliability. It is also worth trying co-training, where several models are used simultaneously, each specializing in different aspects of the images, which can be especially useful in our case.

1. Wang, L.; Chei, W.; Yang, W.; Yu, F.R. A state-of-the-art review on image synthesis with generative adversarial networks. *IEEE Access.* 2020, 8, 63514–63537. https://doi.org/10.1109/ACCESS.2020.2982224
2. Huang, H.; Yu, P.S.; Wang, C. An introduction to image synthesis with generative adversarial nets. *arXiv*, 2018. [Online]. Available: https://arxiv.org/abs/1803.04469 (accessed 2024-05-24)
3. Huang, H.; Li, Z.; He, R.; Sun, Z.; Tan, T. Introvae: Introspective variational autoencoders for photographic image synthesis. *Adv. Neural Inf. Process. Syst.* **2018**, 31.
4. Thies, J.; Zollhöfer, M.; Nießner, M. Deferred neural rendering: Image synthesis using neural textures. *ACM Trans. Graph.* **2019**, 38(4), 1–12. https://doi.org/10.1145/3306346.3323035
5. Esser, P.; Rombach, R.; Ommer, B. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. https://doi.org/10.1109/CVPR46437.2021.01268
6. Jing, Y.; Yang, Y.; Feng, Z.; Ye, J.; Yu, Y.; Song, M. Neural style transfer: A review. *IEEE Trans. Vis. Comput. Graphics* **2019**, 26(11), 3365–3385. https://doi.org/10.1109/TVCG.2019.2921336
7. Raza, A.; Munir, K.; Almutairi, M. A novel deep learning approach for deepfake image detection. *Appl. Sci.* **2022**, 12 (19), 9820. https://doi.org/10.3390/app12199820
8. Sun, W.; Li, P.; Liang, Y.; Feng, Y.; Zhao, L. Detection of image artifacts using improved cascade region-based CNN for quality assessment of endoscopic images. *Bioengineering.* **2023**, 10(11). https://doi.org/10.3390/bioengineering10111288

9. Makinde, F.L.; Tchamga, M.S.S.; Jafali, J.; Fatumo, S.; Chimusa, E.R.; Mulder, N.; Mazandu, G.K. Reviewing and assessing existing meta-analysis models and tools. *Brief Bioinform.* **2021**, 11(22). https://doi.org/10.1093/bib/bbab324
10. Tao, R.; Zhao, X.; Li, W.; Li, H.-C.; Du, Q. Hyperspectral anomaly detection by fractional Fourier entropy. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2019**, 12(12), 4920–4929. https://doi.org/10.1109/JSTARS.2019.2940278
11. Yang, Z.; Liang, J.; Xu, Y.; Zhang, X.-Y.; He, R. Masked relation learning for deepfake detection. *IEEE Trans. Inf. Forensics Security* **2023**, 18, 1696–1708. https://doi.org/10.1109/TIFS.2023.3249566
12. Dhar, A.; Prima, A.; Likhan, B.; Shemonti, A.; Abida, S. *Detecting deepfake images using deep convolutional neural network*; Brac University, 2021.
13. Bhatt, D.; Patel, C.; Talsania, H.; Patel, J.; Vaghela, R.; Pandya, S.; Modi, K.; Ghayvat, H. CNN variants for computer vision: History, architecture, application, challenges and future scope. *Electronics.* **2021**, 10(20), 2470. https://doi.org/10.3390/electronics10202470
14. Yang, Q.; Zhang, Y.; Dai, W.; Pan, S.J. *Transfer Learning.* Cambridge University Press, 2020. https://doi.org/10.1017/9781139061773
15. Mascarenhas, S.; Agarwal, M. A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for image classification. In *CENTCON 2021, Proceedings of 2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications,* Bengaluru, India, 19-21 November, 2021, pp 96–99. https://doi.org/10.1109/CENTCON52345.2021.9687944
16. Schettler, D. DiffusionDB-2M – Part 0001 to 0100 of 2000, 2023. [Online]. Available: https://www.kaggle.com/datasets/dschettler8845/diffusiondb-2m-part-0001-to-0100-of-2000 (accessed 2024-05-24)
17. Google Research. Open Images Dataset V7, 2020. [Online]. Available: https://storage.googleapis.com/openimages/web/index.html (accessed 2024-05-24)
18. Pang, B.; Nijkamp, E.; Wu, Y.N. Deep learning with TensorFlow: A review. *J. Educ. Behav. Stat.* **2020**, 45(2), 227–248. https://doi.org/10.3102/1076998619872761
19. Dawani, J. *Hands-On Mathematics for Deep Learning: Build a Solid Mathematical Foundation for Training Efficient Deep Neural Networks*; Packt Publishing Ltd, 2020.
20. Kundu, N. *Exploring ResNet50: An in-depth look at the model architecture and code implementation*, 2023. https://medium.com/@nitishkundu1993/exploring-resnet50-an-in-depth-look-at-the-model-architecture-and-code-implementation-d8d8fa67e46f (accessed 2024-05-24)
21. Koonce, B. *Convolutional Neural Networks with Swift for TensorFlow: Image Recognition and Dataset Categorization*; Apress, 2021, 109–123. https://doi.org/10.1007/978-1-4842-6168-2_10

Received 19.06.2024