E3S Web of Conferences **687**, 02010 (2026) https://doi.org/10.1051/e3sconf/202668702010
InCASST 2025

Detection of AI-Generated Facial Images Using Convolutional Neural Networks

Anicetus Masdian Rayadi[1], *Hari* Suparwito[1*], and *Anupiya* Nugaliyadde[2]

[1]Informatics Department, Sanata Dharma University, Indonesia
[2]Information Technology Department, Murdoch University, Western Australia

Abstract. In the modern world, the artificial intelligence technology available enables the generation of human faces of which real world counterparts do not exist, and such potential offers a myriad of possibilities. Creativity can illustrate and fabricate work. Granted, technology of this nature can be wielded to serve the purpose of identity fraud, providing misinformation and other seemingly 'immoral' acts. Hence, this study aims to investigate the use of Convolutional Neural Networks (CNN) in composite face images created with 'This person does not exist' and 'real life images' download. Considering the study's focus, the learning rate of 0.0001, sigmoid, 0.4 Dropout, and average pooling for tuning showed the desired learnt outcomes. The results were astounding, the model achieved 99% accuracy on validation and 97% accuracy on the training dataset. This accomplishment was attributed to a face's underlying subtle features, such as its textures, symmetry, and visual interferences. Optimisation was conducted to measure generalisation, needing the model to perform on a new dataset with additional smartphone images. The accuracy was 84% for augmented and real images, with 5 outcomes correct of 6 sample images.

1 Introduction

The internet shows the long history the world has had with the evolution of technology and the ways it alters the way we live, unto this day. From its ability to generate remarkably realistic and professionally taken images to having sites such as 'This Person Does Not Exist,' the internet has truly come a long way. None of this, however, exists without animating a never-ending discourse about the consequences and underlying issues of such new technology, which reach beyond the scope of mere critique and academia. On the one hand, this innovation allows for new realms of entertainment, versatile digital art, and novel app development. On the other hand, it poses the imminent risk of being misused for the purposes of unadulterated misinformation, identity theft, and other illegal actions.

News from Hong Kong mentions the first instances of identity theft via deepfakes [1]. The degree of ease with which identity can be constructed from fake identities indeed underscores the challenges of digital security, privacy, and information authenticity in the contemporary online context. This prompts the areas within artificial intelligence, and within computer science and engineering, that focus on vision and machine learning, to develop

* Corresponding author: shirsj@usd.ac.id

techniques to distinguish computer-generated faces, like those from TPNE, from real photographs of people.

Applying a Convolutional Neural Network (CNN) framework and deep learning to analyse the accuracy of image recognition and classification graphed within the timelines of remains a very promising approach [2][3]. In various studies, the use of CNN methods has been primed to tackle image detection and classification tasks [4][5][6]. Wang et al. proposed a novel CNN detection methodology targeting both visual and ancillary artefacts that are prevalent within AI-generated imagery [7]. Li and Lyu fused CNN deep learning methods in conjunction with frequency analysis to elevate detection efficacies in new and novel ways [8]. Other works have also utilised transfer learning to bolster the model's accuracy when data is scarce [9]. Within heavily contested domains, stagnant and dynamic images are distinguished using CNN methods pertaining to fake image detection [10][11].

The datasets used in this research are the primary hindrance as they are intricate and varied. Images that are captured of a face differ greatly and include a plethora of parameters. Expression, lighting, angle, and the visual field all play a part in the complexity of constructing a working model with flexible and precise classifiers. The variance is important because the learning model needs to trace and recall distinguishable and repetitive classifiers to secure accuracy. In order to evaluate the effectiveness of the CNN algorithm, the study will focus on the images captured on the smartphone itself, as these will differ substantially from what is contained in the public domain datasets, such as those available on Kaggle or GitHub. This is a crucial step in the evaluation of the model's ability to operate on real-life data, which, thanks to the model's shortcomings, will demonstrate discrepancies in quality, illumination, and other parameters that are absent from the model's constructed datasets. The inclusion of such data aims to test the model's robustness and flexibility in realistic and complicated situations.

This study visualises the application of a CNN-based algorithm for distinguishing between face images from TPNE and genuine human pictures, as well as measuring the model's performance on different configurations of hyperparameters.

2 Material and Methods

2.1 Research Workflows

Figure 1 illustrates the research flow. The research phases start with gathering the dataset, then move to data pre-processing to ready the images prior to training the model. Subsequently, the Convolutional Neural Network (CNN) algorithm is employed to create and enhance the classification model. The last phase involves assessing model performance to determine its accuracy and reliability when encountering new data.

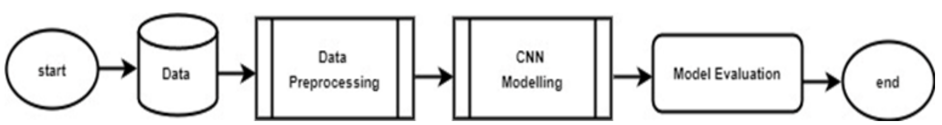


Fig. 1. Research workflow

2.2 Data

The image data used in this study comes from three main sources. The first source is Kaggle (www.kaggle.com/datasets/shirshaka/ai-vs-human-generated-images), which contributed

1,000 randomly selected images. The second source is Github (https://github.com/NVlabs/ffhq-dataset), with a contribution of 6,850 images. The third source is the This Person Does Not Exist (TPNE) website (https://thispersonnotexist.org/), from which we obtained 9,792 images. In total, 17,711 images were collected and ready for analysis. Some examples of images used in this study can be seen in Figures 2 to 4.

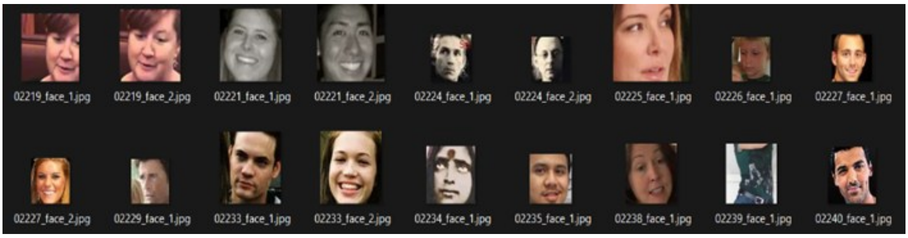


Fig. 2. Sample image data from *kaggle* (www.*kaggle*.com/*datasets*/shirshaka/ai-vs-human-generated-images)

From Kaggle, 1,000 human facial images were selected and labeled as *Human*. These images varied in color composition, with some being full-color and others in black and white.

Fig. 3. Sample image data from *github* (https:// *github*.com/NVlabs/ffhq-dataset)

The second dataset, sourced from GitHub, comprised 6,850 full-color human facial images, all labeled as *Human*.

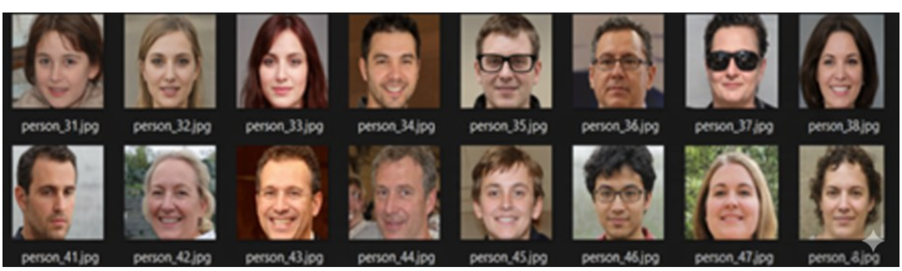


Fig. 4. Sample image data website TPNE (https://thispersonnotexist.org/)

Facial photos sourced from the TPNE website using web scraping tools, and subsequently labelled Fix AI, number 9,792 images. This data set covers images of synthetic faces which were generated using artificial intelligence and more often than not using techniques referred to as Generative Adversarial Networks (GANs). This particular technology is famous and is widely applied in an array of sectors including motion picture, marketing, and the world of digital entertainment as a result of its unique capability of producing images of humans that resemble real people. Although the images appear close to real photographs, AI images also have small distinguishing characteristics, including skin texture and pattern inconsistencies, unnatural facial features and symmetry, blurred and monotonous backgrounds, and extreme facial feature symmetry attributed to the eyes, nose, and mouth which are rarely observed in humans [12]. In sharp contrast, real faces are captured in images without any form of editing and can easily be obtained from an array of sources including instant photographs captured from a camera. These images showcase real faces with natural defects and irregularities, characteristics of which include shapeless, disproportionate, and unsympathetically aligned eyes, mouth, and nose. These images are also marked with real lighting, chaotic backgrounds, and genuine expressions in contrast with the set conditions of the environment. All real face images were properly resized and arranged to eliminate inconsistencies before being used to model the model in this particular study.

2.3 Methodology

After gathering the image data, the next step is to prepare the data. In this case, the ImageDataGenerator included in the Keras library in Python is used. The Keras library streamlines the entire pre-processing phase, data augmentation in particular, burdening the local storage is no longer an issue, as the increased data volumes are managed in the system RAM.

The augmentation procedure consists of a set of transformation parameters identified in other works [13] such as rescaling, rotation, the separation of the dataset used for training, validation, and testing, and the setting of width and height. The overall purpose of these methods is to increase the diversity of the training dataset. Normalisation is a technique used in rescaling, which captures the essence of transforming a range of values to the range of zero to one. In this case, the pixel values within the range of 0 and 255 will be transformed to a range within either of these two-digit values. In context, range scaling is an identification property, which is used to speed up convergence of the model, and training effectiveness as well. Rotation augmentation randomly adjusts the angle of images capturing the objects within a set number of degrees. This addition improves the angle diversity of the dataset so the model can learn to detect objects from a different angle. This means the model becomes less sensitive to certain angles of view, in turn making the model more robust by not depending too much on fixed angles. The parameters width shift range and height shift range move the images horizontally and vertically, respectively. These movements model the interactions of different objects in a scene so that the model does not learn to only focus on central objects. By giving the model multiple shifted images, it learns to ignore the constant features of the images, making the model more robust to variations in the data in turn improving the classification performance [14]. Another important step in pre-processing is setting the colour mode, which determines the colour channel(s) used to load an image. The images in the task may need to be converted to a grayscale image (single channel), RGB image (three colour channels) or RGBA image (four channels with the addition of an alpha channel having the transparency value). This means the colour mode set is relevant to the dataset being used and the architecture design of the neural network.

At the last step, the dataset is divided into three parts: training, validation, and testing. Of the total 17,642 images, 80% or 14,113 images are allocated for training, 15% or 2,646 images for validation and the remaining 5% or 882 images are allocated for testing. Such separation ensures a structured evaluation of the model's learning process.

The training dataset is the model's main source for learning the basic patterns and distinguishing features of the different categories. When the training dataset is sufficiently large, the model is able to learn the different attributes of the categories. This results in a foundation from which the model can make accurate classification decisions from the built representations. The model's performance on the validation set and its ability to incorporate novel information is assessed during training. This subset allows for the early detection of overfitting, defined as the model memorising the training data rather than learning general characteristics. This information is used to make timely adjustments to the model's hyperparameters or training duration. When testing, the model has already undergone training and validation and the purpose now is to provide an unbiased evaluation of the final model. This dataset displays the model's performance of classifying completely foreign data, and thus serves as a reliable measure of the model's practicality and ability to predict in real-world scenarios. This pre-processing pipeline augments the dataset to improve model training, validation, and evaluation, which further assists in developing a robust and adaptable image classifier.

The model is trained in a manner that allows the network to learn the most essential features of the data. The training is set for 10 epochs with the Adam optimiser, a gold standard in deep learning due to its adaptable learning rate and convergence benefits. The training set has 14,113 images, while 2,646 images are for validation, yielding a balanced evaluation of model performance during the learning phase. We have decided to select the CNN parameters to be used in training based on the following considerations: The Adam optimizer was selected because it combines the benefits of both AdaGrad and RMSProp. Adam adapts the learning rate for each parameter, accelerating convergence while stabilizing training. This makes it especially suitable for image datasets with high variability in lighting, symmetry, and texture. Compared to traditional stochastic gradient descent (SGD), Adam requires less manual tuning and performs effectively under sparse gradients, which are often present in facial image analysis. The choice of 128 neurons balances complexity and generalization. More neurons could increase the risk of overfitting, while too few might cause underfitting. Empirical trials showed that 128 neurons were sufficient to capture discriminative features (such as facial irregularities and asymmetries) while maintaining efficiency and preventing overfitting. This configuration yielded the best validation accuracy. After preprocessing, the images were resized to 62×62 pixels and converted into grayscale (single channel). This reduced dimensionality and computational costs while preserving essential facial features. Color information was deemed less critical for distinguishing real and synthetic faces, whereas grayscale emphasized structural and textural cues. The size of 62×62 provided a balance between preserving discriminative detail and ensuring efficient training.

2.4 Modelling

The three subsets within the dataset are training, validation, and testing. Out of the total of 17,642 images, 14,113 images (80%) are used for training, 2,646 images (15%) are used for validation, and the remaining 882 images (5%) are used for testing. This tiered structure allows for systematic evaluation of the model's learning process. The training set is the primary resource used for the model to learn the patterns and features within and between different classes. If the training dataset is large enough, the model is likely to learn the features of all the classes well enough in order to draw enough accurate representations for confident classification decisions. The model needs to be evaluated using a validation set to measure its performance and assess how well it is able to generalise to new, unseen data. Regular evaluation of this subset allows for the detection of overfitting or overtraining, in which the model learns the training data and fails to learn the underlying, general aspects, so that corrective changes to hyperparameter values or training time can be made. Once the training and validation steps are completed, the test set gives a clear-cut evaluation of the final model. This dataset measures how well the model performs on data that it has never seen before, which gives a reliable estimation of its practical usability and prediction accuracy.

This organised pipeline improves the dataset with the intention of enhancing a model's training, validation, and evaluation phases, and consequently, helps build a model with strong and flexible image classification capabilities. Training the model enables a network to learn the basic features of an input. In this case, the Adam optimiser, which is widely used particularly in deep learning because of its learning rate and convergence abilities, performs 10 epochs of training. This dataset contains 14,113 training images and 2,646 validation images which ensures a fair evaluation of the model over the learning period.

The Table 1 shows the various combinations of hyperparameters used to evaluate the performance of the Convolutional Neural Network (CNN) design.

Table 1. The combination of hyperparameter

Parameters	**Values**
Learning rate	0.01, 0.001, 0.0001
Activation function	Sigmoid, Tanh
Dropout	0.2, 0.3, 0.4
Pooling layers	Max pooling, Average pooling

Each model configuration was exhaustively tested to find a suitable model which is capable of high classification accuracy while maintaining cost-effectiveness and general performance. In order to determine the effectiveness of the created models for classifying images as authentic or counterfeit, six models were designed for the specified parameter combinations. The models were compared to determine which one achieved the highest classification accuracy and generalisation over the set of images. To gain insight, all models were trained and validated under the same conditions, which were universally set. The models were built following the same methodology to ensure fairness, order, and precision. This approach clarified how the architecture and training policies influenced the outcome. Models were created using specified parameter combinations to assess their effectiveness in classifying images as authentic or counterfeit.

The aim of this comparative analysis was to determine the best model configuration that attained the maximum classification accuracy and generalization capability. Every model underwent training and validation in uniform conditions to guarantee an equitable and methodical evaluation of how architecture and training parameters influence the overall performance of the Convolutional Neural Network (CNN)

3 Results and Discussion

The best performance of the Convolutional Neural Network (CNN) algorithm was attained by utilizing this parameter setup shown in Table 2.

Table 2. The optimal parameter values

Parameters	**Values**
Learning rate	0.0001
Activation	Sigmoid
Drop out	0.4
Pooling layer	Average pooling 2D
Optimitation	Adam
Fully connected layer	128
Number of features	62x62x1

Throughout the training and validation phases, the model demonstrated a learning trajectory marked by accuracy and loss figures, as shown in Fig. 5 the corresponding graphical representations.

 https://doi.org/10.1051/e3sconf/202668702010

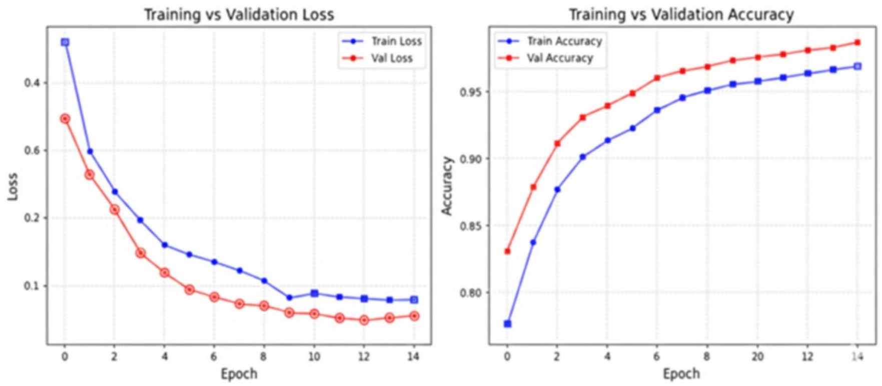


Fig. 5. Loss and Accuracy diagram of Training and Validation process

Confusion matrix obtained from testing dataset is:

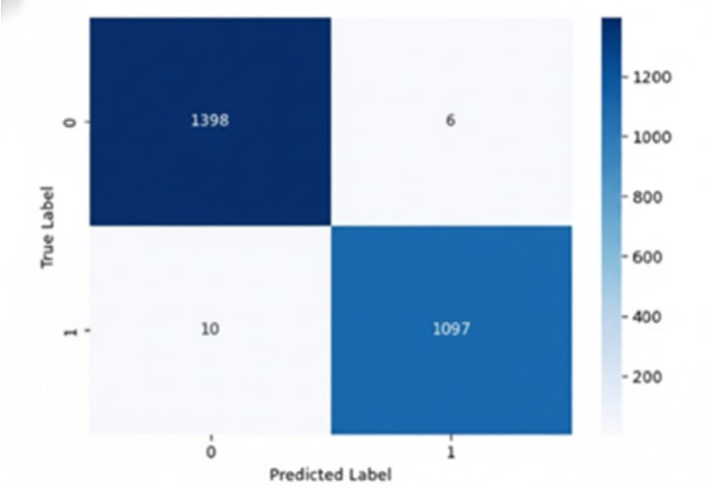


Fig. 6. Confusion matrix of testing dataset

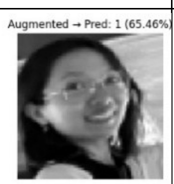
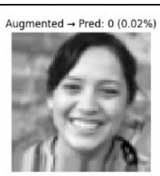
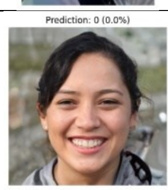
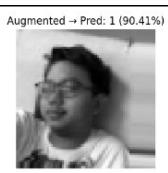
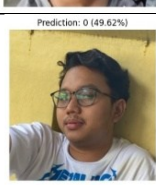
From the results of the confusion matrix in Fig. 6, the CNN model that was built had quite reliable categorisation skills. It achieved 97% training accuracy and 99% validation accuracy within 18 minutes of training, which was quite impressive. This suggests that the model's learning technique is very effective, and it is able to adapt to new, unseen data quite well. The confusion matrix shows the model was able to accurately classify the two target classes of the project, which are “AI Results” and “Human Originals.” In particular, the model successfully identified 1,398 images under the “AI Results” category and 1,097 images under the “Human Originals” category. The model also made some classification errors; 10 images that were supposed to be “Human Original” were identified as “AI Result,” and the same goes for 6 images in the other direction that were taken from the “AI Result” group, which were labelled as “Human Original.” This means that the model is quite functional, but still seems to have a tiny tendency to confuse real human pictures with AI generated photos. This might be due to the more advanced generative models that create artificial images, as they are visually more similar to actual human photos. In some instances, texture, lighting, facial symmetry, deformity, and other descriptors may cause confusion for the model and become misleading.

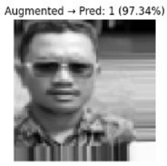
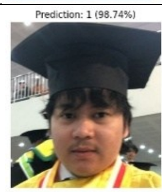
In general, the resulting CNN models continue to be reliably effective for determining the authenticity of facial images. There is a high level of precision with low error rates, and high accuracy is achieved during both training. Execution does not affect the results. Not only does this demonstrate the capability of the model to practically identify AI generated images, but it also highlights the need for further development on certain aspects where there exist minute visual contrasts distinguishing real images from fabricated images.

3.1 Generalisation to a New Data

To enhance the evaluation of the model's generalization capability, the optimized CNN was assessed on a new dataset that varied from the training and validation data. This dataset included enhanced images alongside untouched original images, all captured directly with a mobile phone camera. The addition of these enhanced samples aimed to assess the model's resilience to changes in data. The enhancement procedure included a random rotation of as much as 15%, along with vertical and horizontal displacements of up to 10% of the image size. Table 3 below displays the model's forecasts for both augmented and unprocessed samples, facilitating a direct comparison. Out of the six images evaluated, the model accurately identified five images in both scenarios, resulting in an accuracy rate of 84% for each situation. This indicates that while the model demonstrates strong robustness, classification mistakes can still occur, particularly when synthetic faces exhibit traits closely resembling those of actual human features.

Table 3. Results of the model implementation to a new data

No	Augmented	Original	True Label	Prediction (Augmented)	Prediction (Original)
1			REAL	REAL	REAL
2			FAKE	FAKE	FAKE
3			REAL	REAL	FAKE

4			REAL	REAL	REAL
5			REAL	REAL	REAL
6			REAL	REAL	REAL

Every prediction made by the model comes with a probability score to signify how certain the model is about every outcome, with the lowest score being 0 and the highest being 1. Since this is a binary classification, a decision boundary is required to assign a final label to the input image. For the purposes of this research, the boundary set is 0.5: If the prediction probability is 0.5 or higher, the image is classified to the positive class as "Human"; else, the image is tagged "AI Result." For instance, images with prediction scores from 0.51 to 1.0 will be classified as "Human." In the case of Image 1, which was the original photo of a real person, was classified as "Human" in both the original and enhanced versions. In contrast, Image 3, which is also a real image of a person, was correctly classified as enhanced augmentation, but in the original form, she was wrongly classified as "AI Result." This clearly shows how sensitive the model is to small variations in a set of input features.

The analysis of the updated dataset suggests that the model is less confident when processing the augmented images compared to the original images. "The mean confidence score of un-augmented images was 99.55% while the mean confidence about the augmented counterparts dropped drastically to 65.46%". This fall in confidence regarding the prediction of un-augmented images occurs due to the fact that data augmentation used in the model "distorts" the distribution of input data in a way that may disrupt the interconnections of the learned visual attributes.

Geometric transformations such as the blurring of a class-specific face, rotation, horizontal displacement, and vertical shifting are able to transform "in a more subtle way some crucial face model features the system is trained" to that class. While these transformations are beneficial in increasing the diversity of the dataset and hence increasing model generalisation, they lower the model confidence "by adding examples which deviate too much from the average examples of the training set". Therefore, it is also correct to say that, in spite of the fact the final classification is accurate, the confidence score is likely to be lower.

Alongside this, greater confidence of the model stems from images that closely resemble ones that were a part of the training, which indicates it depends on certain feature patterns from the base data. The model's confidence on augmented samples may indeed drop, but it still manages to hold on to a high rate of accuracy by scoring the majority of images correctly. This indicates that augmentation may irritate the model's confidence, but the model itself does not lack the ability to discriminate. This indicates that it is not only classification accuracy that matters, but also the confidence in the predictions, since that is the manner models will be used in real life, to work with more variations of new inputs.

While the study demonstrated strong performance, it is important to reflect on certain limitations and provide justification for choices made in the scope of this research. We acknowledge that expanding external validation with more images captured from diverse real-world sources such as CCTV, social media, and a wider range of smartphones could further enhance robustness. However, this study intentionally limited its scope to a carefully curated dataset from Kaggle, GitHub, and TPNE, totaling 17,711 images with significant variability in texture, resolution, and background. A smaller set of smartphone images was included to assess cross-domain generalization. While broader validation is desirable, comprehensive collection and annotation of highly diverse "in-the-wild" images requires resources and scope beyond this initial investigation. Importantly, the validation conducted here already demonstrates the model's ability to generalize to new conditions, providing a solid basis for conclusions while leaving expanded testing as a focus for future work. It is also recognized that benchmarking against state-of-the-art models (e.g., XceptionNet, EfficientNet, or ViT architectures) would provide additional context. However, the goal of this study was not to compete in performance rankings but to analyze and demonstrate the effectiveness of a custom CNN through systematic hyperparameter tuning. Incorporating SOTA models would have shifted the focus toward benchmarking rather than exploring CNN performance dynamics. Moreover, such models often require larger computational resources and training pipelines, making fair comparisons beyond the practical scope of this research. Our CNN achieved exceptionally high accuracy (97% training, 99% validation, and strong generalization), showing that the chosen architecture was sufficient to answer the research objectives. Benchmarking with SOTA models remains a meaningful direction for future, large-scale comparative studies.

4 Conclusion

This study employed CNNs to distinguish real human faces from those synthetically created by 'This Person Does Not Exist' (TPNE), to the point where synthetic faces were generated in real-time. The optimised model achieved an astonishing 97% accuracy on the training set while performing 99% accuracy on the validation set, showcasing the robust discriminative capability of the augmented images positive to the model. The overwhelming gains were through hyperparameter tuning such as the low learning rate of 0.0001, the activation function set to sigmoid, dropout set to 0.4, and the inclusion of an average pooling layer. Diagnostics proved the model differentiated features of the images based on textures, asymmetrical shapes, and visophonic anomalies, serving as the focal parameters. When applied to unseen data, the model maintained a 99% accuracy, while the few inaccuracies primarily lay in synthetic images passing off as real faces. This highlights the capability of CNNs in mitigating synthetic face images amidst the deepening challenge brought about by the advancements in imaging. Despite strong performance, several questions remain open such as: As generative adversarial networks (GANs) continue to evolve, synthetic faces increasingly resemble real ones. It is uncertain how well the current CNN architecture will generalize to images generated by next-generation GANs or diffusion models. Future work should explore adaptive or transfer learning approaches to maintain high accuracy across evolving generative techniques. The next one is, the practical deployment in forensic or social media platforms requires lightweight models. Future work should focus on efficient architectures (e.g., MobileNet, EfficientNet) to support real-time detection on mobile and embedded devices.

The authors would like to express their gratitude to the Department of Informatics at Sanata Dharma University for allowing them to use the facilities in the smart computing laboratory so that this study could be completed successfully.

References

1. F. McKenna, AI Deepfakes Merged Scammers Faces On Stolen Identities. Available at: https://frankonfraud.com/ai-deepfakes-merged-scammers-faces-on-stolen-identities/ (2025).
2. K. Azmi, S. Defit, S. Sumijan, Implementasi convolutional neural network (CNN) untuk klasifikasi batik tanah liat sumatera barat. J. Unitek. **16**, 28–40 (2023).
3. E. J. A. Sinaga, G. Oktavia, H. Syahputra, F. Ramadhani, Identifikasi Tanda Tangan Dengan Menggunakan Metode Convolution Neural Network (CNN). J-INTECH. **12**, 138–147 (2024).
4. I. N. G. A. Astawa, M. L. Radhitya, I. W. R. Ardana, F. A. Dwiyanto, Face Images Classification using VGG-CNN. Knowl. Eng. Data Sci. **4**, 49–54 (2021).
5. M. Zamir, N. Ali, A. Naseem, A. Ahmed Frasteen, B. Zafar, M. Assam, E. A. Attia, Face detection & recognition from images & videos based on CNN & Raspberry Pi. Computation. **10**, 148 (2022).
6. R. Wang, et al., FakeSpotter: A Simple yet Robust Baseline for Spotting AI-Synthesized Fake Faces. Available at: http://arxiv.org/abs/1909.06122 (2019).
7. Y. Li, S. Lyu, Exposing DeepFake Videos By Detecting Face Warping Artifacts. Available at: http://arxiv.org/abs/1811.00656 (2018).
8. X. Zhang, S. Karaman, S.-F. Chang, Detecting and Simulating Artifacts in GAN Fake Images. Available at: http://arxiv.org/abs/1907.06515 (2019).
9. Y. Hamid, S. Elyassami, Y. Gulzar, V. R. Balasaraswathi, T. Habuza, S. Wani, An improvised CNN model for fake image detection. Int. J. Inf. Technol. **15**, 5–15 (2023).
10. M. S. M. Altaei, A detection of deep fake in face images using deep learning. Wasit J. Comput. Math. Sci. **1**, 60–71 (2022).
11. T. Karras, S. Laine, T. Aila, A Style-Based Generator Architecture for Generative Adversarial Networks. IEEE Trans. Pattern Anal. Mach. Intell. **43**, 4217–4228 (2021). doi:10.1109/TPAMI.2020.2970919.
12. H. Kaur, N. K. Sandhu, Evaluating the Effectiveness of the Proposed System Using F1 Score, Recall, Accuracy, Precision and Loss Metrics Compared to Prior Techniques. Int. J. Commun. Netw. Inf. Secur. **15**, 368–383 (2023).
13. M. M. Ahsan, M. R. Uddin, M. Farjana, A. N. Sakib, K. A. Momin, S. A. Luna, Image Data collection and implementation of deep learning-based model in detecting Monkeypox disease using modified VGG16. arXiv:2206.01862 (2022).
14. C. Shorten, T. M. Khoshgoftaar, A survey on Image Data Augmentation for Deep Learning. J. Big Data. **6**, 60 (2019). doi:10.1186/s40537-019-0197-0.