Alexandria Engineering Journal 116 (2025) 351–363

Contents lists available at ScienceDirect

Alexandria Engineering Journal

journal homepage: www.elsevier.com/locate/aej

Original article

MEXFIC: A meta ensemble eXplainable approach for AI-synthesized fake image classification

Md Tanvir Islam [a], Ik Hyun Lee [b,c,*], Ahmed Ibrahim Alzahrani [d], Khan Muhammad [e,*]

[a] Visual Analytics for Knowledge Laboratory (VIS2KNOW Lab), Department of Computer Science and Engineering, Sungkyunkwan University, Suwon, 16419, Republic of Korea
[b] Department of Mechatronics Engineering, Tech University of Korea, Siheung, 15073, Republic of Korea
[c] IKLAB Inc., Seoul, 08513, Republic of Korea
[d] Computer Science Department, Community College, King Saud University, Riyadh, 11437, Saudi Arabia
[e] Visual Analytics for Knowledge Laboratory (VIS2KNOW Lab), Department of Applied Artificial Intelligence, School of Convergence, College of Computing and Informatics, SungkyunKwan University, Seoul, 03063, Republic of Korea

ARTICLE INFO

Keywords:
Fake image classification
AI-synthesized image classification
AI-generated image classification
Image authenticity verification
Smart surveillance
CIFAKE

ABSTRACT

In the evolving landscape of artificial intelligence (AI), differentiating between authentic and artificially generated images poses a significant challenge, primarily due to the rapidly enhancing quality of AI-generated images. This paper systematically evaluates state-of-the-art classification models to distinguish authentic images from those synthetically produced using the CIFAKE dataset. We introduce FakeGPT and PFake, two new test datasets featuring genuine and AI-generated synthetic images with specific keywords paralleling the generation of the CIFAKE dataset. We use the transfer learning technique to train the state-of-the-art classification models on the CIFAKE training set, followed by rigorous evaluation against the CIFAKE, FakeGPT, and PFake test datasets. Further, we explore ensemble approaches, including stacking, voting, bagging, and meta-ensemble learning. The culmination of our extensive research efforts is the Meta Ensemble eXplainable Fake Image Classifier (MEXFIC), which stands out with a notable accuracy of 94% and 96.61% against the Stable Diffusion generated CIFAKE and PFake datasets, respectively. This is a significant improvement over the ConvNextLarge model, achieving the highest accuracy of 92.54% among the state-of-the-art models. Our study showcases the competitive edge of MEXFIC that highlights the necessity for more robust models capable of identifying AI-synthesized images, as evidenced by the performance on the challenging FakeGPT dataset.

1. Introduction

The realm of artificial intelligence (AI) has witnessed remarkable progress over recent years, particularly in the field of image generation [1–3]. Today, AI-generated images exhibit such high levels of quality and realism that they rival human creativity and have also begun to win art competitions [4]. This advancement underscores the transformative potential of AI in various sectors, including digital media, entertainment, and even security. Being able to tell authentic images from fake ones made by AI is very important for many reasons. This capability is crucial for maintaining the integrity of digital content and ensuring the security and privacy of individuals.

Additionally, it helps us maintain trust in the information presented online and prevents the spread of misinformation. For example, an excellent Stable Diffusion Model (SDM) [5,6] can create a fake photo of someone doing something wrong or create an alibi for someone who was not there. Today, we face significant problems with misinformation and fake news. Images made by machines can change what people think [7,8].

The rise of deepfake technology [9] has added another layer of complexity to verifying image authenticity. We risk basing our beliefs and decisions on manipulated or fabricated information without distinguishing between real and fake images. On the other hand, this capability poses significant risks regarding misinformation, with potential applications in creating deceptive content ranging from fake news to fraudulent identities. Thus, the importance of developing robust mechanisms to differentiate authentic images from those generated by AI cannot be overstated. In response to these challenges, researchers continuously develop advanced detection algorithms and techniques to discern authentic and AI-generated images accurately [10–12]. It serves the purpose of safeguarding the integrity of digital content and plays a vital role in maintaining trust in digital media.

* Corresponding authors.
E-mail addresses: ihlee@tukorea.ac.kr (I.H. Lee), khan.muhammad@ieee.org (K. Muhammad).

https://doi.org/10.1016/j.aej.2024.12.031
Received 10 June 2024; Received in revised form 12 November 2024; Accepted 9 December 2024
Available online 1 January 2025

Despite the advancements in AI and machine learning algorithms, distinguishing real images from fake ones is challenging [9,13]. One of the primary challenges is the ever-improving quality of AI-generated images, which are becoming increasingly sophisticated and more complex to detect using conventional methods. Additionally, the diversity of generation techniques, including Generative Adversarial Networks (GANs) [14,15], Deepfakes [9], and Variational Autoencoders [16–18], complicates the detection process, requiring versatile and adaptable models to different kinds of image manipulations. Deep learning models have shown promising results in classification-related tasks across the domains. To address this challenge, researchers are exploring various deep learning approaches such as transfer learning [19,20] and multimodal fusion techniques [21,22]. These methods combine the strengths of various detection mechanisms to create more robust and reliable systems for identifying AI-generated images. Moreover, the rapid evolution of deepfake technology necessitates ongoing vigilance and adaptation of detection methods to keep pace with increasingly sophisticated manipulation techniques. This arms race between creators of fake content and developers of detection systems underscores the urgent need for collaborative efforts across academia, industry, and policy-making bodies to mitigate the potential harms associated with AI-generated imagery.

Keeping that in mind, in this paper, we present our research for developing robust CNN-based ensemble models trained using the recently published CIFAKE dataset [23] for detecting real and AI-generated fake images. The CIFAKE dataset represents a significant step forward in this regard. Its balanced compilation of fake and real images provides a comprehensive platform for training and evaluating image authenticity models. As we summerized the workflow of this research in Fig. 1, this paper utilizes the CIFAKE dataset to develop a classification model addressing the abovementioned significant challenges. In addition, we also introduce two new small testing datasets, namely FakeGPT and PFake, to evaluate the models' capability of classifying AI-synthesized fake images from real images. Doing so contributes to the ongoing effort to ensure the veracity of digital imagery, a cornerstone in the digital age where visual content dominates communication and information sharing.

The significance of this paper in the field of identifying AI-generated synthetic images is marked by the following contributions:

- FakeGPT: We introduce a small and highly challenging dataset containing ChatGPT-4o generated synthetic images to challenge the classification models for identifying the ChatGPT-4o generated synthetic images.
- PFake: We generated a new dataset using the Stable Diffusion [6] model for testing and observing the performance of the models on the unknown AI-generated images.
- Benchmarking SOTA Classifiers: We extensively evaluate leading state-of-the-art (SOTA) classification models, setting new benchmarks for real and fake image identification. We benchmark the models against the CIFAKE dataset for classifying AI-synthesized fake images.
- MEXFIC: We develop a Meta Ensemble eXplainable Fake Image Classifier (MEXFIC) using the transfer learning technique with the pre-trained SOTA classifiers. MEXFIC outperforms the existing models by giving up to 94% accuracy on the CIFAKE [23] and 96.61% on our PFake dataset.

The paper is organized as follows: Section 2 presents the related works with a review of previous research works on AI-generated deepfake and general fake image identifications. Section 3 details the datasets we have generated and used for this research. Section 4 outlines the workflow, methods, and techniques we follow to conduct this research, including our proposed framework. In Section 5, we benchmark the SOTA classification models for identifying fake images against the CIFAKE dataset and evaluate our proposed model and different ensemble learning techniques. Section 6 mentions the limitations of our research, while Section 7 concludes our work and suggests future research directions.

2. Related works

Various studies have aimed to identify AI-generated fake images. For instance, Raj et al. propose a compact attention network for robust GAN-generated image detection, outperforming SOTA methods by 5% in cross-category and cross-GAN tests [24]. For deep fake identification, several studies [25–27] have been performed for deep fake detection, particularly for identifying fake faces and utilized different deep learning methods such as ResNet50 [28] and VGG-19 [29]. Similarly, other research [30,31] have also used SOTA CNN models as baselines for fake facial image detection, demonstrating their effectiveness. However, these methods focus mainly on facial images. In contrast, a team developed a generalized fake image detection method using gated hierarchical multi-task learning, achieving over 99% accuracy in closed-set scenarios and surpassing SOTA methods by 4.2% in open-set scenarios [32]. Dong et al. challenge the robustness of frequency spectrum-based fake image detectors by showing that spectral artifacts in GAN-generated images can be mitigated. Their work highlights vulnerabilities in current detection methods and the need for more advanced techniques [33]. Ferreira et al. introduce VIPPrint for forensic analysis of printed documents, including 40,000 images. Their study on deepfake detection and printer source attribution shows a minimum 9% error probability for baseline methods. However, StyleGAN2 [34] images appear genuine after printing and scanning, highlighting the need for advanced detection techniques [35]. Tang et al. use discrete wavelet transform to detect GAN-synthesized images by focusing on spectral correlation. Tested on StyleGAN2 [34] and various GANs, their method is effective and robust against common perturbations [36].

Researchers also used the transfer learning technique and proposed X-Transfer for detecting GAN-generated fake images, employing dual neural networks with interleaved parallel gradient transmission and achieved promising performance [37]. Another team of researchers explores banknote recognition and counterfeit detection through custom models and transfer learning, analyzing the optimal freezing point for classifier performance [19]. Hamid et al. enhance fake image detection with an improved CNN model, comparing six conventional machine learning models and CNN architectures. Their optimized ResNet50 [28], augmented with advanced preprocessing techniques, achieves a remarkable accuracy by marking an 18% performance improvement over other models [20]. A combination of machine learning algorithms to differentiate between tampered and genuine images was also introduced to detect medical image deepfakes. It notably achieves high accuracy in detecting injected tumors, utilizing models like DenseNet [38] and ResNet [28] for feature extraction and refinement [39]. Vora et al. propose techniques using machine learning and knowledge graphs to detect AI-generated content, achieving high accuracy with BERT and CNN models. The research highlights challenges and suggests validation methods [40].

Recent work by Bhinge and Nagpal evaluated the performance of real vs. AI-generated images using the CIFAKE [23] dataset, highlighting EfficientNet_V2_B0's [41] superior accuracy compared to traditional CNN models [42]. Hossain et al. explore CNN and Vision Transformer models to enhance AI-generated image detection using the same CIFAKE dataset. Their CNN model achieved 96.31% accuracy, utilizing Grad-CAM for interpretability, providing insights into model decision processes [43]. Similarly, another research team experiment with VGG19 [29] model training and testing on the same dataset gets 84.24% accuracy [44]. Gupta et al. also employed CNN to improve the detection of AI-generated images using the CIFAKE dataset. Their best model achieved 97.29% validation accuracy; however, the test set accuracy was not reported. Finally, the authors of the CIFAKE dataset proposed a CNN-based model that achieves 92.98% accuracy on the CIFAKE dataset's test set [23].

We reviewed research on detecting and identifying AI-synthetic images, including methods for identifying fake faces, medical images, and general fake images. We noticed that several studies use deep

Table 1
The class distribution of the datasets used in this paper.

Datasets	Trainset		Testset	
	REAL	FAKE	REAL	FAKE
CIFAKE	50,000	50,000	10,000	10,000
FakeGPT	Created for testing		67	67
PFAKE	Created for testing		746	700

learning and transfer learning techniques. Although much research has been conducted in this field, very few studies focus on identifying fake images with general scene types [23,24,40,42–44], as the CIFAKE [23] dataset offers. Also, reviewing the literature, we hardly found comprehensive studies with ensemble methods for AI-synthesized image classification, despite the widespread use of ensemble learning techniques across various domains [45,46]. Identifying this gap, we benchmark SOTA classification models and employ ensemble learning methods in various configurations of hyperparameters in our experiments to identify AI-generated images, particularly with general image scene types. Moreover, we propose an ensemble learning-based model, Meta Ensemble Explainable Fake Image Classifier (MEXFIC), designed to classify AI-generated fake images.

3. Datasets

For training and evaluating the models, we use several datasets as follows:

CIFAKE: A dataset encompassing both actual and synthetic images is essential to construct a model adept at identifying AI-generated imagery. CIFAKE [23] fulfills this requirement by creating AI-generated images derived from the authentic visuals of the CIFAR-10 collection. The CIFAKE dataset contains both training and testing sets. The training partition comprises 100,000 images evenly distributed between the 'FAKE' and 'REAL' labels. The testing counterpart comprises 20,000 images with an equal share for each category.

FakeGPT: The FakeGPT dataset, crafted utilizing ChatGPT-4o, is an evaluation dataset to examine the model's proficiency in recognizing images generated by ChatGPT [1]. The creation of this dataset was guided by specific keywords, mirroring those used for CIFAKE image generation, the details of which are cataloged in Table 1. This ensures a coherent benchmarking standard for model assessment. The purpose of creating this dataset is to evaluate the models that we train against the Stable Diffusion [6] generated CIFAKE dataset. It will determine whether models trained on the CIFAKE dataset can detect fake images generated by ChatGPT-4o.

PFake: The PFake dataset, created to evaluate models trained on the CIFAKE datasets, leverages ChatGPT-4o and Stable Diffusion to generate images for testing AI-synthesized image identification. PFake comprises 1,446 images, divided into synthetic and real categories. Using 746 carefully crafted prompts, synthetic images were generated, each reflecting keywords used to generate the CIFAKE [23] dataset.

The dataset also includes 700 real images from CIFAKE's real class, providing a balanced benchmark for evaluating synthetic images. This mix ensures a comprehensive validation set for model performance assessment. Sample images from each dataset, shown in Fig. 2, highlight the visual similarity between CIFAKE and PFake images, making it challenging to distinguish between fake and real visuals, unlike the more artificial-looking ChatGPT-4o generated images.

[1] https://openai.com/chatgpt/overview

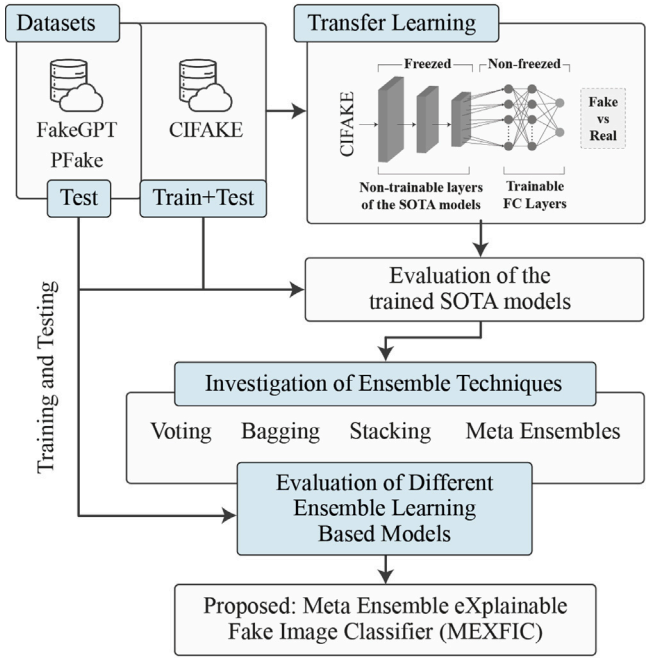


Fig. 1. Workflow diagram showing the research methodology using different datasets. It involves transfer learning with SOTA models, testing ensemble techniques, and proposing the MEXFIC model.

4. Methodology

The full workflow of this research is schematically presented in Fig. 1, which involves rigorous training and testing of SOTA pre-trained neural network models [28,29,38,41,47–51]. These models are then subjected to experiments under different configurations, employing ensemble strategies to enhance prediction accuracy for proposing a final explainable meta-ensemble model as presented in Fig. 3. A comprehensive breakdown of each step in the methodology of our proposed model is explained as follows:

4.1. Ensemble techniques

Ensemble techniques work by aggregating the predictions from different models, reducing variance and bias, or improving predictions. We use three popular ensemble techniques: bagging, voting, and stacking. The final predictions for these techniques are shown in Eqs. (1)–(4).

$$\text{Bagging} = \frac{1}{N}\sum_{i=1}^{N} f_i(x) \tag{1}$$

where N is the number of models in the ensemble, $f_i(x)$ is the prediction of the ith model, and x is the input feature vector.

$$\text{Hard Voting} = \arg\max_{c}\left(\sum_{i=1}^{N} I(p_i(x) = c)\right) \tag{2}$$

Here, c is the class label, $\arg\max_c$ selects the class with the highest votes, N is the number of models, and $I(p_i(x) = c)$ is an indicator function that returns 1 if the ith model predicts class c for instance x, and 0 otherwise.

$$\text{Soft Voting} = \arg\max_{c}\left(\frac{1}{N}\sum_{i=1}^{N} p_i(c|x)\right) \tag{3}$$

Here, N is the number of models, $p_i(c|x)$ denotes the predicted probability of instance x being in class c according to the ith model,

Fig. 2. Sample images from the (a) CIFAKE, (b) FakeGPT, and (c) PFake datasets featuring AI-generated and real images.

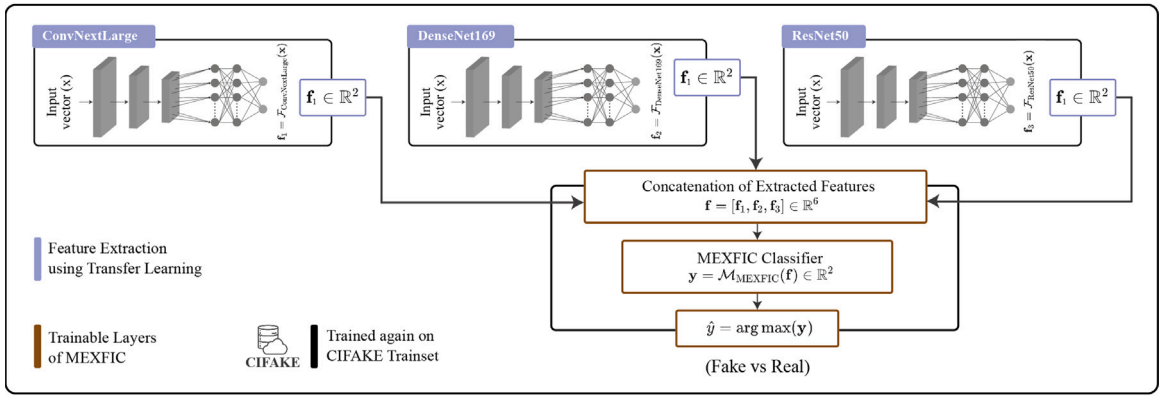


Fig. 3. Architecture of our proposed MEXFIC (Meta Ensemble eXplainable Fake Image Classifier) model. This diagram demonstrates the integration of transfer learning and ensemble techniques across multiple SOTA architectures like ConvNextLarge [47], DenseNet169 [38], and ResNet50 [28], contributing to the model's capability for distinguishing between fake and real images.

and arg max selects the class c with the maximum average probability.

$$\text{Stacking} = f_{\text{meta}}(f_1(x), f_2(x), \ldots, f_N(x)) \tag{4}$$

where f_{meta} is the meta-learner's prediction function based on the outputs $f_1(x), f_2(x), \ldots, f_N(x)$ of the base learners and x is the input image.

4.1.1. Meta ensemble

A meta-ensemble [52] model in deep learning, especially with CNNs, enhances performance and robustness by combining multiple models. Each model, possibly trained on different data subsets or with varied architectures, contributes its unique strengths, resulting in a more generalized and accurate composite prediction. In our study, we train individual CNN-based models using the transfer learning [53,54] technique followed by training the meta-learner for the final prediction as shown in Eq. (4). In our case, we take three models $f_1(x)$, $f_2(x)$, and $f_3(x)$ to use as the baseline of our meta learner and then finally train the f_{meta} against the train set of the CIFAKE dataset.

4.2. Proposed MEXFIC

In this section, we delve deeper into the architecture and training process of the Meta Ensemble eXplainable Fake Image Classifier (MEXFIC), our proposed method for detecting AI-synthesized fake images. MEXFIC is a meta-ensemble learning model incorporating SOTA classifiers: [47], DenseNet169 [38], and ResNet50 [28], based on Eq. (4) explained earlier in Section 4.1.1. In our MEXFIC model, the functions $\mathcal{F}_{\text{ConvNextLarge}}$, $\mathcal{F}_{\text{DenseNet169}}$, and $\mathcal{F}_{\text{ResNet50}}$ correspond to the CIFAKE pre-trained models ConvNextLarge, DenseNet169, and ResNet50, respectively. Therefore, the value of N in Eq. (4) is set to 3, reflecting the integration of these three models as the foundational elements of the MEXFIC architecture.

A detailed architecture of our proposed MEXFIC model is illustrated in Fig. 3. The outputs from ConvNextLarge, DenseNet169, and ResNet50 each have a shape of 2 that is presented in Eqs. (5)–(7). This output shape corresponds to the two classes we aim to classify: real and fake images. In the architecture of MEXFIC, the outputs of these three models are combined, resulting in an aggregated input shape of 6 for the meta-ensemble model, as shown in Eq. (8). This means that MEXFIC receives a concatenated vector of predictions from the three models, which contains rich and diverse insights about the features of the classified images.

$$\mathbf{f}_1 = \mathcal{F}_{\text{ConvNextLarge}}(\mathbf{x}) \quad \text{where} \quad \mathbf{f}_1 \in \mathbb{R}^2 \tag{5}$$

$$\mathbf{f}_2 = \mathcal{F}_{\text{DenseNet169}}(\mathbf{x}) \quad \text{where} \quad \mathbf{f}_2 \in \mathbb{R}^2 \tag{6}$$

$$\mathbf{f}_3 = \mathcal{F}_{\text{ResNet50}}(\mathbf{x}) \quad \text{where} \quad \mathbf{f}_3 \in \mathbb{R}^2 \tag{7}$$

The concatenated features are then passed through the meta-ensemble classifier $\mathcal{M}_{\text{MEXFIC}}$, as follows:

$$\mathbf{y} = \mathcal{M}_{\text{MEXFIC}}(\mathbf{f}) \quad \text{where} \quad \mathbf{y} \in \mathbb{R}^2 \tag{8}$$

For feature extraction of ConvNextLarge [47], DenseNet169 [38], and ResNet50 [28], the feature vectors $\mathbf{f}_1$, $\mathbf{f}_2$, and $\mathbf{f}_3$ are obtained by applying their respective feature extraction functions $\mathcal{F}_{\text{ConvNextLarge}}$, $\mathcal{F}_{\text{DenseNet169}}$, and $\mathcal{F}_{\text{ResNet50}}$ to the input image $\mathbf{x}$ as shown in Eqs. (5)–(7). Each feature vector $\mathbf{f}_i$ is a 2-dimensional vector in $\mathbb{R}^2$ representing the

Algorithm 1 Pseudocode of our proposed MEXFIC and the steps of training.

Require: CIFAKE Dataset (D), SOTA Models,
Split D into Train D_{train}: 80% and Validation D_{val}: 20%
Pretrained Models: M_A, M_B, M_C
Parameters: Batch size $b = 64$, Learning rate $\eta = 0.001$, Image size $s = 256$, Early stop patience = 15 steps, Epochs = 250

Procedure: Develop MEXFIC(D_{train}, D_{val})
class MEXFIC(nn.Module):
  def _init_(self, m_A, m_B, m_C):
    self.m_A = m_A
    self.m_B = m_B
    self.m_C = m_C
    self.classifier = nn.Linear(6, 2)
  def forward(self, x):
    x_1 = self.m_A(x)
    x_2 = self.m_B(x)
    x_3 = self.m_C(x)
    x = torch.cat([x_1, x_2, x_3], dim=1)
    return self.classifier(x)

Procedure: Train MEXFIC(D_{train}, D_{val})
Initialize MEXFIC (M_A, M_B, M_C)
for $epoch = 1$ **to** 250 **do**
  Train model on D_{train}
  Apply learning rate scheduler with gamma = 0.5
  Evaluate model on D_{val}
  Save model if performance improves
  Break if early stopping criterion is met
end for
End Procedure

classification scores for real and fake images. These processes can be combined into a single equation as shown in Eq. (9).

$$\mathbf{f}_i = \mathcal{F}_i(\mathbf{x}) \quad \text{where} \quad \mathbf{f}_i \in \mathbb{R}^2, \\ i \in \{\text{ConvNextLarge, DenseNet169, ResNet50}\} \tag{9}$$

The outputs from the individual models are concatenated to form a single input vector for the MEXFIC classifier as presented in Eq. (10):

$$\mathbf{f} = [\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3] \quad \text{where} \quad \mathbf{f} \in \mathbb{R}^6 \tag{10}$$

Finally, the class with the highest probability is selected as the final prediction, as follows:

$$\hat{y} = \arg\max(\mathbf{y}) \tag{11}$$

The MEXFIC model then processes this six-element input vector through its layers, as shown in Eq. (11). It culminates in a final classifier layer that outputs a shape of 2, corresponding to the two image classes, fake and real. In the case of MEXFIC, we only use a simple dense layer as its layer. Thus, by integrating the capabilities of ConvNextLarge [47], DenseNet169 [38], and ResNet50 [28], MEXFIC leverages a broader spectrum of learned features and decision patterns, enhancing its ability to classify images accurately over the baseline SOTA models.

To further refine and validate our model, we train MEXFIC on the entire CIFAKE dataset, encompassing the training and validation images. This comprehensive training approach helps to ensure that MEXFIC learns from the individual strengths of the SOTA models and optimizes to produce a robust classifier capable of high accuracy in real-world scenarios.

4.3. Computational complexity

The computational complexity of the proposed MEXFIC meta-ensemble model is primarily determined by the complexities of its three base models: ConvNextLarge, DenseNet169, and ResNet50. The overall complexity is the sum of the individual model as presented in Eq. (12) complexities and the operations involved in the ensemble.

For ConvNextLarge, the complexity is approximately $O(N \times D \times F^2 \times K^2)$, where N is the number of layers, D is the number of channels, F is the spatial size of feature maps, and K is the kernel size. DenseNet169 has a complexity of $O(L \times D \times F^2 \times K^2)$, where L represents the number of layers with dense connections, which increases computational demand. ResNet50's complexity is $O(L \times D^2 \times F^2)$, leveraging residual connections to reduce gradient vanishing issues.

The ensemble's total complexity can be expressed as:

$$O_{\text{MEXFIC}} = O_{\text{ConvNextLarge}} + O_{\text{DenseNet169}} + O_{\text{ResNet50}} \tag{12}$$

After obtaining individual model outputs, a classifier with complexity $O(N \times M)$ combines them, where N is the concatenated feature size and M is the output class.

4.4. Performance metrics

We used well-known performance metrics to evaluate the models, such as accuracy, precision, recall, F1-score, and confusion matrix. Besides, we use t-SNE [55], which is an advanced technique used for visualizing high-dimensional data in a lower-dimensional space, making it easier to identify patterns and clusters within complex datasets. This optimization ensures that the low-dimensional representation maintains the original dataset's structure as closely as possible.

$$p_{ij} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k \neq l} \exp(-\|y_k - y_l\|^2)} \tag{13}$$

Here, y_i and y_j represent the low-dimensional embeddings of high-dimensional points. And the k typically refers to the number of nearest neighbors in high-dimensional space, while p_{ij} represents the joint probability that points i and j are neighbors in the low-dimensional space.

4.5. Explainable AI (XAI)

Deep learning models often act as "black boxes", lacking transparency. Explainable AI (XAI) addresses this by making models more understandable. We use Gradient Class Activation Mapping (Grad-CAM) [56] to interpret predictions. Grad-CAM highlights important image areas influencing the model's decision by analyzing gradients in the final convolutional layer, producing a heatmap:

$$L^c_{\text{Grad-CAM}} = \text{ReLU}\left(\sum_k \alpha^c_k A^k\right) \tag{14}$$

Here, α^c_k are weights from global average pooling over the gradients of the feature map A^k for class c. This method visually explains the model's behavior, enabling interpretability.

Table 2
Overview of the parameters, models, and techniques utilized in our research experiments.

Items	Parameters
Models	SOTA Classifiers, Ensemble
Epochs	50, 150, 250
Batch size	8, 16, 32, 64, 128
Learning rate	0.1, 0.01, 0.001, 0.0001
Early stopping	Patience steps = 15
Learning scheduler	Gamma = 0.5

Table 3
Performance evaluation of ImageNet-trained SOTA classification models on the CIFAKE dataset, without additional training or transfer learning techniques. The results indicate that the models require additional fine-tuning to enhance their performance in classifying AI-generated images.

Models	# of Params (Million)	Model size (MB)	GFLOPS	Accuracy	Precision	Recall	F1-score
ConvNextLarge [47]	197.80	754.50	34.36	52.14	52.83	52.14	49.06
DenseNet161 [38]	28.70	110.40	7.73	51.70	51.76	51.70	51.26
DenseNet169 [38]	14.15	54.70	3.36	**58.26**	**58.91**	**58.26**	**57.48**
DenseNet201 [38]	20.01	77.40	4.29	48.04	47.76	48.04	46.37
EfficientNet_B0 [41]	5.30	20.50	0.39	48.10	47.86	48.10	46.62
EfficientNetV2Large [41]	118.50	454.60	56.08	50.00	25.00	50.00	33.33
GoogleNet [48]	6.62	49.70	1.50	46.30	46.24	46.30	46.07
MobileNet_V3 [49]	5.50	21.10	0.22	51.97	52.37	51.97	49.87
ResNet50 [28]	25.60	97.80	4.09	48.16	48.16	48.16	48.15
ResNet101 [28]	44.50	170.50	7.80	51.64	56.83	51.64	40.32
ResNet152 [28]	60.19	203.50	11.51	29.80	45.40	29.80	27.97
ShuffleNet_V2 [50]	1.40	5.30	0.04	50.89	51.31	50.89	46.54
SqueezeNet [51]	1.20	4.70	0.35	49.08	48.86	49.08	46.49
VGG16 [29]	138.40	527.80	15.47	46.86	46.72	46.86	46.27
VGG19 [29]	143.67	548.10	19.63	49.58	49.36	49.58	44.91

Table 4
Performance metrics of the SOTA classification models trained on the CIFAKE dataset using the transfer learning method, with training conducted in two phases at batch sizes of 32 and 64.

Models	Batch size = 32				Batch size = 64			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ConvNextLarge [47]	**89.32**	**90.50**	**89.32**	**89.25**	**92.54**	**92.57**	**92.54**	**92.54**
DenseNet161 [38]	74.69	81.19	74.69	73.30	84.58	84.60	84.58	84.58
DenseNet169 [38]	74.62	81.89	74.62	73.09	85.96	85.97	85.96	85.96
DenseNet201 [38]	71.32	76.58	71.32	69.83	85.23	85.29	85.23	85.22
EfficientNet_B0 [41]	60.98	62.53	60.98	59.73	73.18	73.29	73.18	73.15
EfficientNetV2Large [41]	45.36	44.62	45.36	43.42	67.84	68.05	67.84	67.75
GoogleNet [48]	68.69	70.56	68.69	67.97	70.99	71.01	70.99	70.99
MobileNetV3 [49]	64.96	65.01	64.96	64.92	70.10	70.50	70.10	69.95
ResNet50 [28]	77.39	81.03	77.39	76.70	81.64	81.73	81.64	81.63
ResNet101 [28]	82.28	82.52	82.28	82.25	81.36	82.28	81.36	81.23
ResNet152 [28]	72.84	78.71	72.84	71.37	81.82	82.43	81.82	81.74
ShuffleNetV2 [50]	69.64	74.89	69.64	67.94	81.22	81.24	81.22	81.22
SqueezNet [51]	56.11	73.68	56.11	46.11	55.46	75.40	55.46	44.59
VGG16 [29]	79.68	79.68	79.68	79.67	80.36	80.36	80.36	80.35
VGG19 [29]	78.44	78.45	78.44	78.44	79.62	79.62	79.62	79.61

4.6. Experimental setups

Our research extensively explores a variety of advanced classiers found in current literature. For experimenting with these models we organize our experiments into several distinct phases as follows:

- Phase 1: We first assess the performance of SOTA classifiers using the CIFAKE testset, utilizing the ImageNet pre-trained weights.
- Phase 2: Next, we train and evaluate SOTA classifiers on the CIFAKE dataset to determine which classifiers are most effective at identifying AI-generated fake images. We experiment using batch sizes of 32 and 64, maintaining a consistent learning rate of 0.001 and a training duration of 50 epochs.
- Phase 3: We select the top-performing models from Phase 2 and use them to construct ensemble models through Bagging, Voting, and Stacking techniques. We then test and compare the efficacy of these ensemble models on the CIFAKE dataset.
- Phase 4: We develop and evaluate meta-ensemble learning models using combinations of the top models identified in our earlier studies. Using the Adam optimizer with a learning rate of 0.001 and batch size of 32 and 64, we train these models on 150 epochs.
- Phase 5: Lastly, we conduct ablation studies on the proposed model to examine their inference results and times, aiming to propose the most effective model based on these analyses. We train these models employing the early stopping technique, learning scheduler on 250 epochs.

For all these experimental phases, we employ a consistent set of parameters regarding epochs, batch sizes, and learning rates, incorporating techniques like early stopping and learning rate scheduling, as detailed in Table 2. All experiments are conducted using the PyTorch framework on a high-specification computer setup, which includes a Core i9 processor, NVIDIA RTX 4080 GPU, 64 GB RAM, 1 TB SSD and a 10 TB hard disk.

Table 5
Performance evaluation of individual bagging models trained on three partitions of the CIFAKE dataset, each containing 10,000 images. The models use ConvNextLarge [47] as their base, chosen for its superior accuracy, as shown in Table 4.

Models	Dataset size	Accuracy	Precision	Recall	F1-score
Model 1	32k	92.42	92.44	92.42	92.42
Model 2	32k	92.20	92.21	92.20	92.20
Model 3	32k	92.36	92.36	92.36	92.35

5. Results and discussion

Our extensive experiments began with benchmarking SOTA classifiers and ensemble learning techniques such as Bagging, Voting, and Stacking, culminate in developing meta-ensemble learners. These efforts aim to identify the best meta-ensemble learner for accurately detecting fake images.

5.1. Benchmarks of SOTA models

We first evaluated ImageNet pre-trained SOTA models for classifying AI-synthesized fake images. As shown in Table 3, these models

Table 6
Detailed evaluation of several ensemble models against the CIFAKE test set, utilizing different combinations of top-performing models, including ConvNextLarge, DenseNet169, and ResNet50, as identified earlier in Table 4. This table categorizes the models into bagging and voting (hard) strategies and outlines their respective performance metrics.

Models	ConvNextLarge [47]	DenseNet169 [38]	ResNet50 [28]	Accuracy	Precision	Recall	F1-score
Bagging	✓	×	×	**92.71**	**92.71**	**92.71**	**92.71**
Soft voting 1	✓	✓	✓	83.55	87.02	83.55	83.16
Soft voting 2	×	✓	✓	92.66	92.67	92.66	92.66
Soft voting 3	✓	×	✓	92.56	92.56	92.56	92.56
Hard voting 1	✓	✓	✓	83.55	87.02	83.55	83.16
Hard voting 2	×	✓	✓	77.94	83.37	77.94	77.01
Hard voting 3	✓	×	✓	85.54	88.42	85.54	85.26

Table 7
Meta Ensemble Models while the base models are trained with a batch size of 32 and 64. We chose the top-performing SOTA models as the backbone of these meta-ensemble models and evaluated them against the CIFAKE test. The checkmarks indicate the models used as the backbone of the each meta-ensemble learner.

Models	Baseline models			Batch size	Performance metrics			
	ConvNextLarge [47]	DenseNet169 [38]	ResNet50 [28]		Accuracy	Precision	Recall	F1-score
When the baseline models were trained with a setup of 32 batch size.								
Meta ensemble 1	×	✓	✓	32	78.74	84.01	78.74	77.88
Meta ensemble 2	✓	✓	×	32	87.12	89.39	87.12	86.94
Meta ensemble 3	✓	✓	✓	32	87.88	89.88	87.88	87.73
Meta ensemble 4	×	✓	✓	64	89.32	89.35	89.32	89.31
Meta ensemble 5	✓	✓	×	64	93.58	93.59	93.58	93.58
Meta ensemble 6 (MEXFIC)	✓	✓	✓	64	**93.80**	**93.80**	**93.80**	**93.80**
When the baseline models were trained with a setup of 64 batch size.								
Meta ensemble 7	✓	✓	✓	32	93.00	93.00	93.00	92.99
Meta ensemble 8	✓	✓	✓	64	93.48	93.53	93.48	93.48

failed to perform adequately on this task due to their training on ImageNet features. We also compared the models' number of parameters, size, and GFLOPS to assess efficiency in terms of space and time.

Given the pre-trained models' poor performance, we retrained and tested each model on the CIFAKE dataset. The results, presented in Table 4, show that with a batch size of 32, ConvNextLarge and ResNet101 achieved accuracies of 89.32% and 82.28%, respectively, while other models fell below 80%. With a batch size of 64, ConvNextLarge achieved 92.54% accuracy, and the DenseNet models (DenseNet161, DenseNet169, DenseNet201) achieved around 85%. The ResNet variants such as ResNet50, ResNet101, and ResNet152 reached around 81%.

As shown in Table 3, DenseNet169 and ResNet50 have relatively smaller sizes and fewer parameters, making them efficient. Therefore, we selected ConvNextLarge for its high accuracy and DenseNet169 and ResNet50 for further experiments due to their balance of performance and efficiency. While VGG16 and VGG19 showed near 80% accuracy, we chose ResNet50 for its lower parameter size and ConvNextLarge for superior accuracy, ensuring that top-performing models were selected based on CIFAKE test set performance.

5.2. Results of ensemble learners

As mentioned earlier, in this study, we use three common ensemble techniques, namely Bagging, Voting, and Stacking. Starting with the bagging, we select ConvNextLarge as the baseline model of our Bagging model, as ConvNextLarge achieves the highest accuracy of 92.54% (Table 4) among all the SOTA models. Based on the technique of Bagging, we train the ConvNextLarge model's three different versions with the subsets of the CIFAKE trainset, where each subset consists of 32,000 images, Fake and Real, with each class having 16,000 images. These individual ConvNextLarge models achieve around 92% accuracy while testing on the CIFAKE test set, as shown in Table 5. Using these three versions of the newly trained ConvNextLarge models used as the baseline models of the Bagging model in this experiment and as the evaluation result of our Bagging model stated in Table 6, achieves 92.71% accuracy, precision, recall, and F1-score.

From Table 6, we can learn more about the different variants of the voting techniques, where we take different combinations of the top-performing models. For example, the Soft Voting 1 model with the combination of ConvNextLarge, DenseNet169, and ResNet50 achieves 83.55% accuracy, the lowest compared to the Soft Voting versions 2 and 3. The highest accuracy obtained by the Soft Voting 2 model is 92.66%, which was developed based on the DenseNet169 and ResNet50. On the other hand, the Soft Voting 3, built with a combination of ConvNextLarge and ResNet50, gives the second-highest accuracy of 92.56% among all the voting technique-based models.

Similarly, we combine the same three top-performing SOTA models for the hard voting approach to form our Hard Voting models. The combinations of the baseline models for these hard voting models are indicated in Table 6. Among all the Hard Voting models, Hard Voting 3, which was formed based on ConvNextLarge and ResNet50, gives the highest accuracy of 85.54%, while the combination of DenseNet169 and ResNet50 gives the worst accuracy of 77.94%.

Therefore, comparing the three different ensemble learning methods, we found that the Bagging method achieves the highest accuracy compared to the Voting techniques. Moreover, while comparing the soft and hard majority voting techniques, our experimental results show that the soft voting technique achieves better accuracy than the hard voting technique.

5.3. Results of meta ensemble learners

Our further experimental results with the meta-ensemble learning technique are displayed in Table 7. The table shows that we incorporate our CIFAKE-trained ConvNextLarge, DenseNet169, and ResNet50 models as the backbone of our meta-ensemble learners. Among all the meta-ensemble based models we experimented with in this paper, we found MEXFIC to achieve the highest accuracy of 93.80% accuracy while testing against the CIFAKE test set. For our proposed MEXFIC model, the backbone models used were trained with a batch size of 32, while the MEXFIC model was then trained again with a batch size of 64. The second highest accuracy (93.58%) in this list was achieved by the Meta Ensemble 5, while the backbone models are ConvNextLarge and DenseNet169. Similarly, the other combinations of meta-ensemble learners are detailed in Table 7, with information on their baseline models and their batch sizes. Note that all these models in Table 7 were

Table 8
The ablation study examines performance changes due to variations in batch size (BS), learning rate (LR), and optimizers while training the proposed MEXFIC model. The models were trained with the set of parameter setups mentioned in Table 2.

Models	Param	CIFAKE (Stable Diffusion)				FakeGPT (GPT-generated)				PFake (Stable Diffusion)			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
Proposed MEXFIC	Batch size	Impact of BS when LR = 0.001, optimizer = Adam											
	8	91.91	91.95	91.91	91.91	55.97	56.94	55.97	54.38	94.67	94.84	94.67	94.66
	16	93.52	93.52	93.52	93.51	60.45	60.45	60.45	60.45	95.92	96.00	95.92	95.92
	32	**94.00**	**94.05**	**94.00**	**94.00**	47.01	46.86	47.01	46.34	**96.61**	**96.64**	**96.61**	**96.61**
	64	93.89	93.91	93.89	93.89	**66.25**	**69.28**	**66.25**	**64.87**	96.33	96.40	96.33	96.33
	128	93.60	93.60	93.60	93.60	41.04	40.07	41.04	39.56	96.20	96.26	96.20	96.19
	LR	Impact of varying LR when BS = 32, optimizer = Adam											
	0.1	93.67	93.67	93.67	93.67	**57.46**	**57.98**	**57.46**	**56.77**	95.92	96.02	95.92	95.91
	0.01	93.20	93.22	93.20	93.20	56.72	56.81	56.72	56.56	**96.61**	**96.64**	**96.61**	**96.61**
	0.001	**94.00**	**94.00**	**94.00**	**94.00**	**58.21**	**58.48**	**58.21**	**57.87**	96.33	96.40	96.33	96.33
	0.0001	93.72	93.72	93.72	93.71	55.97	56.29	55.97	55.41	96.13	96.24	96.13	96.12
	Optimizer	Impact of varying optimizer when BS = 32, LR = 0.01											
	Adam	**94.00**	**94.00**	**94.00**	**94.00**	**66.25**	**69.28**	**66.25**	**64.87**	**96.33**	**96.40**	**96.33**	**96.33**
	Adagrad	90.95	90.95	90.95	90.95	58.96	59.31	58.96	58.57	93.22	93.34	93.22	93.21
	AdamW	93.14	93.15	93.14	93.13	59.70	60.29	59.70	59.12	95.37	95.49	95.37	95.36
	RMSprop	91.95	92.03	91.96	91.95	51.49	51.51	51.49	51.36	95.92	95.94	95.92	95.92
	Adadelta	90.59	90.59	90.58	90.58	55.97	56.08	55.97	55.77	93.91	93.98	93.91	93.91
	Adamax	92.73	92.73	92.72	92.72	57.46	57.98	57.46	56.77	95.30	95.34	95.30	95.29
	ASGD	81.04	81.05	81.04	81.04	48.51	48.50	48.51	48.48	80.01	80.02	80.01	80.00
	Rprop	91.28	91.29	91.28	91.28	54.48	54.56	54.48	54.27	94.26	94.31	94.26	94.26

trained in 150 epochs with an early stop patience steps of 15 and not employing the learning scheduler.

As we use the three top-performing SOTA models as the backbone of our proposed MEXFIC model and get the highest accuracy, we further train Meta Ensemble 7 and 8 with batch sizes 32 and 64. In comparison, the 3 backbone models were trained with batch size 64. Remember the MEXFIC model's backbone models were trained with a batch size 32. Therefore, for Meta Ensemble 7 and 8, we train the backbone models with 64 batch sizes to observe whether the performance improves. Our final observation is that among all the meta ensemble learners, our proposed MEXFIC version of meta ensemble learner gives top performance with an accuracy of 93.80%.

5.4. Ablation study

We perform several ablation studies on our proposed MEXFIC to gain insights into the results of varying parameters during training in different steps. The evaluation results of our ablation studies are presented in Table 8.

5.4.1. Impact of batch size

The batch size significantly affects the performance of the MEXFIC model. A batch size of 32 achieves the highest performance metrics across the CIFAKE and PFake datasets, with accuracy, precision, recall, and F1-score all at 94.00% for CIFAKE and 96.61% for PFake. On the FakeGPT dataset, a batch size of 64 yields the best results, with an accuracy of 66.25% and an F1-score of 64.87%. Smaller batch sizes, such as 8, and larger batch sizes, such as 128, result in lower performance across all datasets, indicating that both extremes can negatively impact the model's effectiveness. A batch size of 16 shows moderate performance, better than the extremes but not as effective as 32 or 64.

5.4.2. Impact of learning rate

The learning rate also plays a crucial role in the model's performance. A learning rate of 0.001 provides the best overall performance for the CIFAKE and PFake datasets, with all metrics at 94.00% and 96.33%, respectively. This learning rate also achieves the highest F1-score for the FakeGPT dataset at 58.21%. In contrast, higher (0.1) and lower (0.0001) learning rates result in lower performance, indicating that extremely high or low learning rates are less effective. A learning rate of 0.01 performs moderately but does not match the effectiveness of 0.001.

5.4.3. Impact of optimizer

The choice of optimizer significantly influences the model's performance. The Adam optimizer outperforms all other optimizers across the CIFAKE and PFake datasets, achieving accuracy, precision, recall, and F1-score at 94.00% and 96.33%, respectively. It also shows the best performance for the FakeGPT dataset, with an accuracy of 66.25% and an F1-score of 64.87%. The AdamW optimizer is the next best, particularly on the FakeGPT dataset, achieving an accuracy of 59.70% and an F1-score of 59.70%. Other optimizers, such as Adagrad, Adadelta, ASGD, and Rprop, perform poorly across all datasets, with significantly lower accuracy and F1-scores compared to Adam and AdamW.

The ablation study reveals that the optimal configuration for the MEXFIC model is a batch size of 32, a learning rate of 0.001, and the Adam optimizer, which consistently achieves the highest performance across the CIFAKE, FakeGPT, and PFake datasets, demonstrating the model's robustness and effectiveness under these settings.

5.5. Comparative analysis

We further compared the performance of MEXFIC with recent works [23,24,40,42–44] that primarily focused on identifying AI-generated synthetic images across general scene types.

We displayed the comparison results in Table 9, listing only the recent models tested for identifying general AI-generated synthetic images, as our model's primary goal is also to do so. Comparing the outcomes, we found that MEXFIC gives the highest accuracy of 96.61% among the models [23,24,40,42–44] while testing against both Stable Diffusion-generated CIFAKE and PFake test datasets. Another model [24] that was tested against GAN-generated images jointly with our MEXFIC achieved the second-highest accuracy of 94%.

In Table 10, we compare the performance of the top-performing SOTA classification models and our proposed MEXFIC model across various datasets. The results demonstrate the superior performance of MEXFIC by consistently achieving the highest accuracy in fake image classification across all datasets. For CIFAKE, MEXFIC records an accuracy of 94.00%, outperforming ConvNextLarge, DenseNet169, and ResNet50 which achieved an accuracy of 89.32%, 74.62%, and 77.39%, respectively. Additionally, MEXFIC leads in precision, recall, and F1-score of 94.05%, 94.00%, and 94.00%, respectively, showcasing its superior performance in fake image classification on this dataset. Similarly, while all the models struggled to perform against the FakeGPT

Table 9
Performance comparison of our proposed MEXFIC model against existing models from the literature for classifying general fake images. Here, the term 'General' refers to datasets that are not specific to any particular object or item.

Models	Dataset	Image Types	Approach	Accuracy
2023/Vora et al. [40]	CIFAKE	General	CNN	93.55
2023/Hossain et al. [43]	CIFAKE	General	CNN	96.31
2023/Hayathunnisa et al. [44]	CIFAKE	General	CNN	84.24
2023/Bhinge et al. [42]	CIFAKE	General	CNN	75.26
2024/Raj et al. [24]	GAN-generated	General	CNN	94.00
2024/Bird et al. [23]	CIFAKE	General	CNN	92.98
2024/MEXFIC (Our)	CIFAKE PFAKE	General	**Meta ensemble**	94.00 **96.61**

Table 10
Performance comparison of our proposed MEXFIC with the baseline models used to form the backbone of the MEXFIC based on the accuracy metrics for testing against the CIFAKE, FakeGPT, and Pfake datasets.

Models	CIFAKE (Stable diffusion)				FakeGPT (GPT-generated)				PFake (Stable diffusion)			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ConvNextLarge [47]	89.32	90.50	89.32	89.25	53.73	54.67	53.73	51.29	89.56	90.54	89.56	89.46
DenseNet169 [38]	74.62	81.89	74.62	73.09	57.46	58.28	57.46	56.39	78.42	81.27	78.42	77.78
ResNet50 [28]	77.39	81.03	77.39	76.70	50.00	50.00	50.00	40.73	76.63	83.46	76.63	75.14
MEXFIC (Our)	**94.00**	**94.05**	**94.00**	**94.00**	**58.21**	**58.48**	**58.21**	**57.87**	**96.61**	**96.64**	**96.61**	**96.61**

dataset due to its nature and challenges, MEXFIC still achieved the highest accuracy at 58.21% among all the models. For the PFake dataset, the MEXFIC model maintains its top performance, achieving an accuracy of 96.61%, significantly higher than ConvNextLarge, DenseNet169, and ResNet50. Overall, the MEXFIC model outperforms other SOTA models in fake image classification tasks across different datasets, highlighting its effectiveness and reliability for various applications in image classification.

The inference time in milliseconds (ms) tested with a repetition value of 100 for each model is presented in Table 11. The mean and standard deviation times of the ConvNextLarge model were 55.83 ms and 262.03 ms, while the DenseNet169 took 31.52 ms and 185.35 ms, respectively. On the other hand, among the 4 models, we found ResNet50 to take the lowest mean of 3.71 ms and standard deviation of 0.70 ms inference time. Even though our proposed model, MEXFIC, achieves the highest accuracy, its inference time is also higher than the other models because it incorporates three different models to form a meta-ensemble model. It indicates the need for further improvements of these models regarding accuracy and inference time, which can be a focus for future works in this domain.

5.6. Visual results

The loss curves for training and validating the top-performing SOTA models ConvNextLarge, DenseNet169, ResNet50, and MEXFIC are shown in Fig. 4. As each model was trained with an early stopping technique setup, different models stopped training in different numbers of epochs. For instance, the ConvNextLarge takes more than 100 epochs to converse, while the ResNet50 stopped training only after 25 epochs. The other two models, DenseNet169 and MEXFIC, take more than 50 epochs to stop training. The curves for the ConvNextLarge show a steady decline, suggesting that the model is generalizing well without overfitting. Consequently, the accuracy graph demonstrates a consistent increase in accuracy. The DenseNet169 model's curves reveal a stable and gradual decrease in training loss. In contrast, the validation loss displays considerable fluctuations throughout the epochs. The loss curve of the ResNet50 model shows fluctuations yet an overall downward trend. Ultimately, both the loss curves for the proposed MEXFIC model showed consistent fluctuations during the training and validation phases.

Fig. 5 presents t-SNE visualizations of model classifications on the CIFAKE and PFake datasets, comparing the performance of ConvNextLarge, DenseNet169, ResNet50, and our proposed MEXFIC model. Real images are depicted in green and fake images in red. On the CIFAKE dataset (Fig. 5(a)), the MEXFIC model demonstrates the most distinct

Table 11
Inference time across the models that perform robustly for classifying fake vs. real images.

Models	Mean time ↓	Standard deviation ↓
ConvNextLarge [47]	55.8309	262.0259
DenseNet169 [38]	31.5189	185.3493
ResNet50 [28]	**3.7081**	**0.7045**
MEXFIC (Our)	85.4417	294.8543

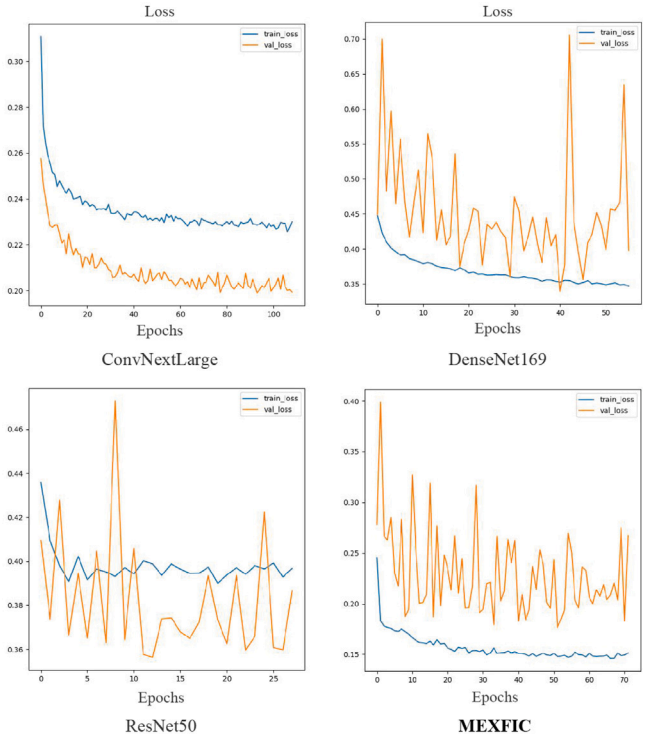
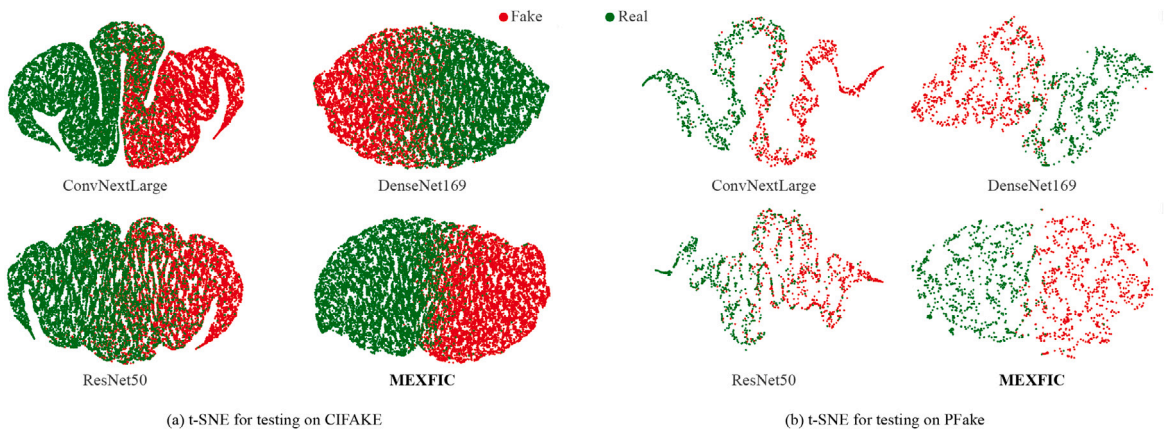
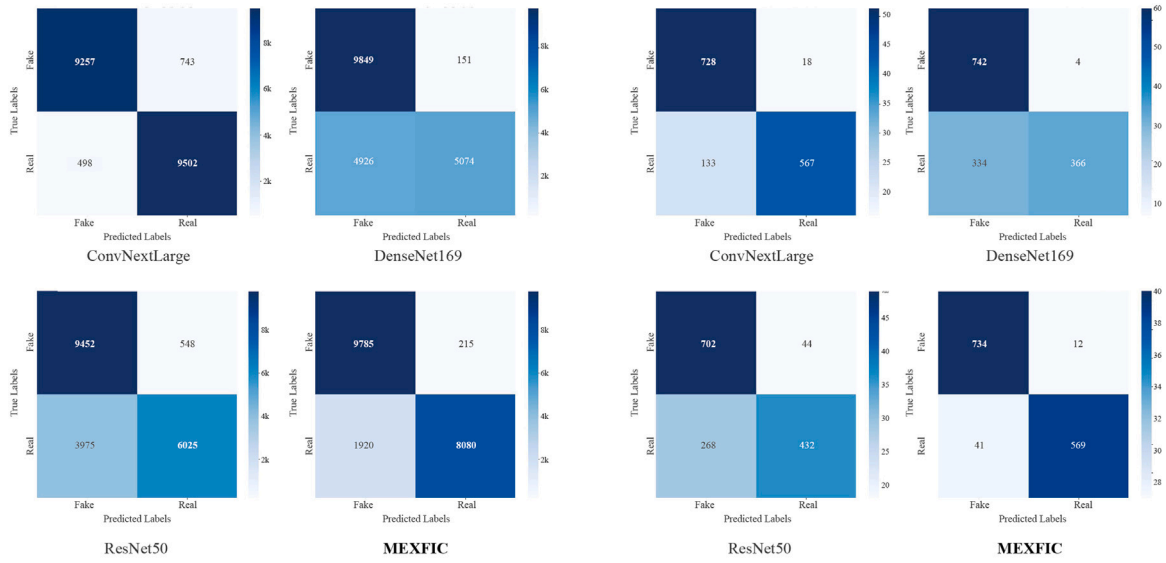


Fig. 4. Training loss curves for the top-performing models and our proposed MEXFIC with the set of parameters listed in Algorithm 1.



(a) t-SNE for testing on CIFAKE (b) t-SNE for testing on PFake

Fig. 5. t-SNE visualization of the top-performing models and the proposed MEXFIC when testing on the CIFAKE (a) and PFake (b) datasets.



(a) Confusion Matrix for testing on CIFAKE (b) Confusion Matrix for testing on PFake

Fig. 6. Confusion Matrices of the top-performing models and the proposed MEXFIC when testing on the CIFAKE (a) and PFake (b) datasets.

separation between real and fake images, with a clear boundary and minimal overlap, indicating superior performance in distinguishing between the two categories. In contrast, ConvNextLarge and ResNet50 show moderate separation with noticeable overlap, suggesting less effective differentiation. DenseNet169 exhibits the least clear separation, with a substantial mixing of real and fake images, indicating poorer performance. On the PFake dataset (Fig. 5(b)), the MEXFIC model again shows superior performance with a clearer separation between real and fake images compared to the other models. ConvNextLarge, DenseNet169, and ResNet50 display moderate separation but overlap more, reflecting challenges in handling PFake. Overall, these t-SNE plots highlight that the MEXFIC model consistently outperforms the baseline models, particularly in its ability to distinctly differentiate between real and fake images across both datasets, underscoring its effectiveness in fake image classification tasks.

Fig. 6 presents confusion matrices for model testing on the CIFAKE and PFake datasets, comparing ConvNextLarge, DenseNet169, ResNet50, and the proposed MEXFIC model. Fig. 6(a) shows the matrices for CIFAKE, and Fig. 6(b) for PFake. On CIFAKE, MEXFIC demonstrates superior performance with 9,785 true positives for fake images and 8,080 for real images, and the lowest false positives and negatives. ConvNextLarge shows moderate performance with 9,257 true positives for fake images and 9,502 for real images, but higher false positives and negatives. ResNet50 and DenseNet169 perform worse, with DenseNet169 having 4,926 true positives for fake images and 5,074 for real images. On PFake, MEXFIC again outperforms, correctly identifying 734 fake and 569 real images, with minimal misclassifications. ConvNextLarge and ResNet50 display higher false positives and negatives, and DenseNet169, while improved, still lags behind MEXFIC. These matrices underscore MEXFIC's superior accuracy and robustness in differentiating real and fake images across both datasets.

Additionally, MEXFIC is explainable like the other SOTA models, which means it is possible to know which features from input images contribute to the model's decision on an output class. In Fig. 7, we present some images using the GradCAM technique to visualize the important features of the input images that contribute the most while inferring these images with the ConvNextLarge, DenseNet169, ResNet50, and MEXFIC models. Fig. 7(a) presents some images from the CIFAKE test set, Fig. 7(b) presents some images from the FakeGPT dataset, and Fig. 7(c) shows the images from the PFake dataset.

6. Limitations and future scopes

In our investigation for benchmarking and proposing models for AI-synthesized fake image classification, we made strides with a meta-ensemble learning approach, which has been underexplored in the

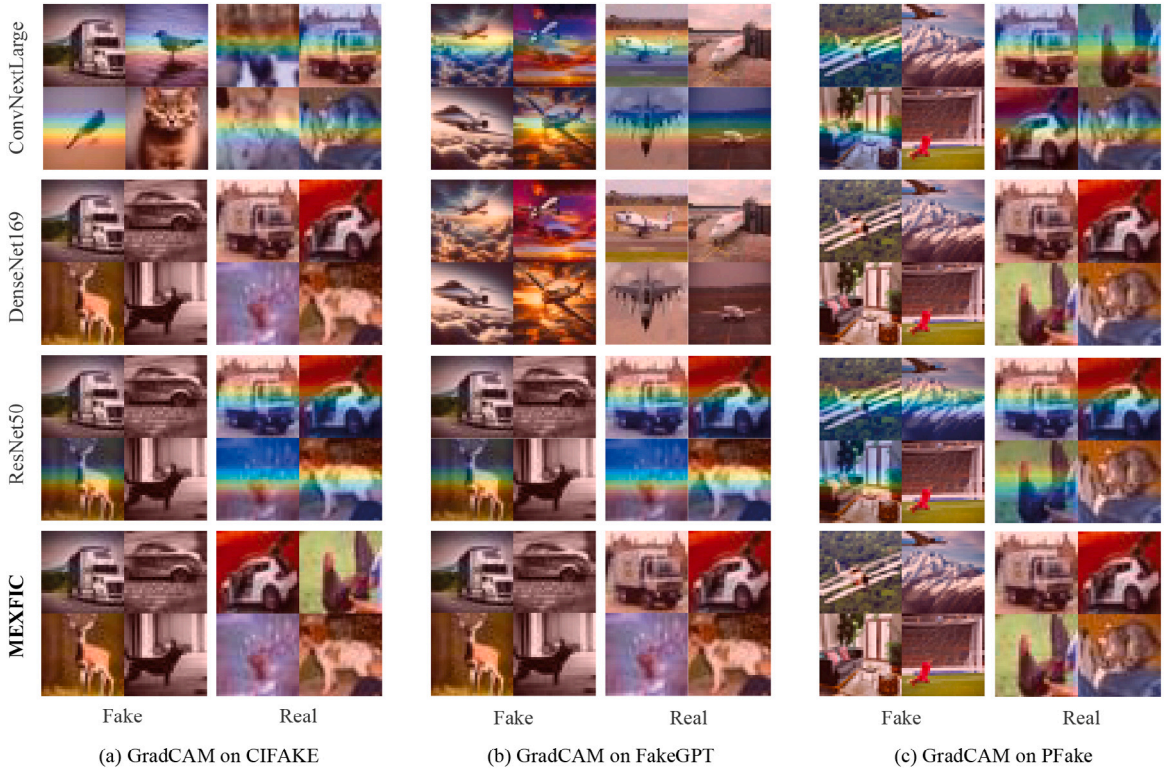


Fig. 7. Grad-CAM visualizations of the top-performing models and the proposed MEXFIC model testing on the CIFAKE (a), FakeGPT (b), and PFake (c) datasets where each row corresponds to a different model, showing their focus areas on fake versus real images.

field. However, our proposed MEXFIC model has a relatively longer inference time compared to other SOTA models, primarily due to its meta-ensemble structure that integrates multiple deep learning models. This increased computational cost may limit its practical applicability in scenarios where real-time or low-latency processing is required. Additionally, our study primarily relied on the CIFAKE dataset for training, which may limit the model's generalizability to other types of AI-generated images. This was evident when testing our model on the ChatGPT-4o generated FakeGPT dataset, where performance varied due to the differences in image characteristics. This reliance on a specific dataset suggests that the model might need further adjustments or retraining to perform effectively on more diverse or unseen datasets.

Several future research directions can be explored to address the limitation of MEXFIC's higher inference time. One promising approach is replacing the current base models with more lightweight architectures, which balance speed and accuracy well. Additionally, applying model pruning and quantization techniques could reduce the model's size and computational requirements without significantly affecting accuracy. Another direction involves using knowledge distillation to create a smaller model replicating the ensemble's performance, thereby reducing inference time while maintaining effectiveness. Furthermore, for furthering the research for AI-synthesized fake image classification, a promising direction is the integration of visual and textual modalities, where we would combine image classification with natural language processing techniques like [57,58] to analyze text descriptions or metadata associated with images. This multimodal approach could prove effective in scenarios like social media or news articles, where misleading text is often paired with AI-generated images. Moreover, creating more robust datasets that incorporate adversarial attacks [59, 60] and various types of noise, potentially generated through diverse noisy dataset generation techniques [3,61], could enhance the models' generalization to handle unknown data types effectively. This approach would not only improve the model's resilience but also ensure its adaptability to real-world scenarios with unpredictable data variations.

7. Conclusion

The rapid advancement of AI technologies has made it increasingly difficult to distinguish between real and fake images, raising concerns about the authenticity of visual information and ethical issues like privacy. Our research addresses this by introducing MEXFIC, a novel model for classifying AI-generated synthetic images using a meta-ensemble learning approach, which is less explored in the field. Although our model shows promise, it has a longer inference time compared to SOTA models. Additionally, reliance on the CIFAKE dataset means findings may not generalize across more diverse data. Future research should explore integrating additional modalities and using a wider variety of GAN-generated datasets to enhance robustness. Our MEXFIC model outperforms existing SOTA methods, achieving 94% accuracy on CIFAKE and 96.61% on the PFake dataset. This improvement is due to the meta-ensemble learning technique, combining the strengths of multiple classifiers. These findings are significant for combating the proliferation of fake images, especially on social media. Future work should focus on developing more sophisticated models to maintain high accuracy without compromising speed and expanding training to diverse GAN-generated datasets. Continued research in this area is vital, and future studies should expand benchmark datasets to include a broader array of AI-synthesized images, ensuring comprehensive evaluation and model robustness.

CRediT authorship contribution statement

Md Tanvir Islam: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Ik Hyun Lee:** Writing – review & editing, Validation, Resources, Methodology, Funding acquisition, Formal analysis. **Ahmed Ibrahim Alzahrani:** Writing – review & editing, Validation, Resources, Investigation, Formal analysis. **Khan Muhammad:** Writing – review & editing, Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis.

Data and code availability

The newly introduced datasets, trained models, and codes are released on GitHub (https://github.com/tanvirnwu/MEXFIC) to support future research in this domain.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Author Khan Muhammad is AE of this journal.

Acknowledgments

This work was supported by the Priority Research Centers Program (2017R1A6A1A03015562) through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Korea government (MOLIT) (No. RS-2021-KA164547, Development of Self-Powered and Wireless Safety Monitoring Technology for Railway Power Supply Systems), and in part by the Gyeonggi-do Regional Research Center (GRRC) Program of Gyeonggi Province, Development of an Intelligent Inspection System and an Autonomous Navigation System for the Transportation of Multi-Material Parts, under Grant GRRC TUKorea2023-B03. This work was also supported by the Researchers Supporting Project number (RSP2025R157), King Saud University, Riyadh, Saudi Arabia.

References

[1] G. Paulin, M. Ivasic-Kos, Review and analysis of synthetic dataset generation methods and techniques for application in computer vision, Artif. Intell. Rev. 56 (9) (2023) 9221–9265.
[2] V.L.T. De Souza, B.A.D. Marques, H.C. Batagelo, J.P. Gois, A review on generative adversarial networks for image generation, Comput. Graph. 114 (2023) 13–25.
[3] M.T. Islam, N. Rahim, S. Anwar, M. Saqib, S. Bakshi, K. Muhammad, Hazespace2m: A dataset for haze aware single image dehazing, in: Proceedings of the 32nd ACM International Conference on Multimedia, 2024, pp. 9155–9164.
[4] K. Roose, An Ai-Generated Picture Won an Art Prize. Artists Aren't Happy - The New York Times, vol. 2, 2022, https://www.nytimes.com/2022/09/02/technology/ai-artificial-intelligence-artists.html. (Accessed 16 February 2024).
[5] F.-A. Croitoru, V. Hondru, R.T. Ionescu, M. Shah, Diffusion models in vision: A survey, IEEE Trans. Pattern Anal. Mach. Intell. 45 (9) (2023) 10850–10869.
[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-resolution image synthesis with latent diffusion models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10684–10695.
[7] G. Pennycook, D.G. Rand, The psychology of fake news, Trends Cogn. Sci. 25 (5) (2021) 388–402.
[8] B. Singh, D.K. Sharma, Predicting image credibility in fake news over social media using multi-modal approach, Neural Comput. Appl. 34 (24) (2022) 21503–21517.
[9] M. Masood, M. Nawaz, K.M. Malik, A. Javed, A. Irtaza, H. Malik, Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward, Appl. Intell. 53 (4) (2023) 3974–4026.
[10] Z. Akhtar, Deepfakes generation and detection: A short survey, J. Imag. 9 (1) (2023) 18.
[11] I. Alam, M.S. Muneer, S.S. Woo, Ugad: universal generative ai detector utilizing frequency fingerprints, in: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, 2024, pp. 4332–4340.
[12] L. Stroebel, M. Llewellyn, T. Hartley, T.S. Ip, M. Ahmed, A systematic literature review on the effectiveness of deepfake detection techniques, J. Cyber Secur. Technol. 7 (2) (2023) 83–113.
[13] L. Guarnera, O. Giudice, F. Guarnera, A. Ortis, G. Puglisi, A. Paratore, L.M. Bui, M. Fontani, D.A. Coccomini, R. Caldelli, et al., The face deepfake detection challenge, J. Imag. 8 (10) (2022) 263.
[14] A. Dash, J. Ye, G. Wang, A review of generative adversarial networks (GANs) and its applications in a wide variety of disciplines: From medical to remote sensing, IEEE Access 12 (2024) 18330–18357, http://dx.doi.org/10.1109/ACCESS.2023.3346273.
[15] T. Chakraborty, U.R. KS, S.M. Naik, M. Panja, B. Manvitha, Ten years of generative adversarial nets (GANs): A survey of the state-of-the-art, Mach. Learn.: Sci. Technol. 5 (1) (2024) 011001.
[16] D.P. Kingma, M. Welling, Auto-encoding variational Bayes, 2022, URL: https://arxiv.org/abs/1312.6114. arXiv:1312.6114.
[17] Y. Akkem, S.K. Biswas, A. Varanasi, A comprehensive review of synthetic data generation in smart farming by using variational autoencoder and generative adversarial network, Eng. Appl. Artif. Intell. 131 (2024) 107881.
[18] I. Cetin, M. Stephens, O. Camara, M.A.G. Ballester, Attri-VAE: Attribute-based interpretable representations of medical images with variational autoencoders, Comput. Med. Imaging Graph. 104 (2023) 102158.
[19] C.G. Pachón, D.M. Ballesteros, D. Renza, Fake banknote recognition using deep learning, Appl. Sci. 11 (3) (2021) 1281.
[20] 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 (1) (2023) 5–15.
[21] J. Wang, H. Mao, H. Li, FMFN: Fine-grained multimodal fusion networks for fake news detection, Appl. Sci. 12 (3) (2022) 1093.
[22] Y. Wu, P. Zhan, Y. Zhang, L. Wang, Z. Xu, Multimodal fusion with co-attention networks for fake news detection, in: Findings of the Association for Computational Linguistics, ACL-IJCNLP 2021, 2021, pp. 2560–2569.
[23] J.J. Bird, A. Lotfi, Cifake: Image classification and explainable identification of ai-generated synthetic images, IEEE Access (2024).
[24] S. Raj, J. Mathew, A. Mondal, Generalized and robust model for GAN-generated image detection, Pattern Recognit. Lett. 182 (2024) 104–110, http://dx.doi.org/10.1016/j.patrec.2024.04.018.
[25] J. Yang, S. Xiao, A. Li, G. Lan, H. Wang, Detecting fake images by identifying potential texture difference, Future Gener. Comput. Syst. 125 (2021) 127–135.
[26] Z. Guo, G. Yang, J. Chen, X. Sun, Fake face detection via adaptive manipulation traces extraction network, Comput. Vis. Image Underst. 204 (2021) 103170.
[27] S. Pashine, S. Mandiya, P. Gupta, R. Sheikh, Deep fake detection: Survey of facial manipulation detection solutions, 2021, arXiv preprint arXiv:2106.12605.
[28] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[29] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2015, URL: https://arxiv.org/abs/1409.1556.
[30] A. Raza, K. Munir, M. Almutairi, A novel deep learning approach for deepfake image detection, Appl. Sci. 12 (19) (2022) 9820.
[31] H.A. Khalil, S.A. Maged, Deepfakes creation and detection using deep learning, in: 2021 International Mobile, Intelligent, and Ubiquitous Computing Conference, MIUCC, IEEE, 2021, pp. 1–4.
[32] Y. Zhou, P. He, W. Li, Y. Cao, X. Jiang, Generalized fake image detection method based on gated hierarchical multi-task learning, IEEE Signal Process. Lett. 30 (2023) 1767–1771.
[33] C. Dong, A. Kumar, E. Liu, Think twice before detecting GAN-generated fake images from their spectral domain imprints, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2022, pp. 7865–7874.
[34] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, T. Aila, Analyzing and improving the image quality of stylegan, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 8110–8119.
[35] A. Ferreira, E. Nowroozi, M. Barni, VIPPrint: Validating synthetic image detection and source linking methods on a large scale dataset of printed documents, J. Imag. 7 (3) (2021) 50.
[36] G. Tang, L. Sun, X. Mao, S. Guo, H. Zhang, X. Wang, Detection of GAN-synthesized image based on discrete wavelet transform, Secur. Commun. Netw. 2021 (2021) 1–10.
[37] L. Zhang, H. Chen, S. Hu, B. Zhu, X. Wu, J. Hu, X. Wang, X-transfer: A transfer learning-based framework for robust gan-generated fake image detection, 2023, arXiv preprint arXiv:2310.04639.
[38] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.
[39] S. Solaiyappan, Y. Wen, Machine learning based medical image deepfake detection: A comparative study, Mach. Learn. Appl. 8 (2022) 100298.
[40] V. Vora, J. Savla, D. Mehta, A. Gawade, R. Mangrulkar, Classification of diverse AI generated content: An in-depth exploration using machine learning and knowledge graphs, 2023, http://dx.doi.org/10.21203/rs.3.rs-3500331/v1.
[41] M. Tan, Q.V. Le, EfficientNet: Rethinking model scaling for convolutional neural networks, 2020, URL: https://arxiv.org/abs/1905.11946.
[42] S. Bhinge, P. Nagpal, Quantifying the performance gap between real and ai-generated synthetic images in computer vision, 2023, http://dx.doi.org/10.2139/ssrn.4594547, Available at SSRN 4594547.
[43] M.Z. Hossain, F.U. Zaman, M.R. Islam, Advancing AI-generated image detection: Enhanced accuracy through CNN and vision transformer models with explainable AI insights, in: 2023 26th International Conference on Computer and Information Technology, ICCIT, IEEE, 2023, pp. 1–6.
[44] V. Hayathunnisa, P. Kuppusamy, A. Manimaran, Art of detection: Custom CNN and VGG19 for accurate real vs fake image identification, in: 2023 6th International Conference on Recent Trends in Advance Computing, ICRTAC, IEEE, 2023, pp. 306–312.

[45] Z. Mian, X. Deng, X. Dong, Y. Tian, T. Cao, K. Chen, T. Al Jaber, A literature review of fault diagnosis based on ensemble learning, Eng. Appl. Artif. Intell. 127 (2024) 107357.
[46] M.J. Shayegan, A brief review and scientometric analysis on ensemble learning methods for handling COVID-19, Heliyon (2024) http://dx.doi.org/10.1016/j.heliyon.2024.e26694.
[47] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, S. Xie, A convnet for the 2020s, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11976–11986.
[48] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.
[49] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, et al., Searching for mobilenetv3, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1314–1324.
[50] N. Ma, X. Zhang, H.-T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient CNN architecture design, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 116–131.
[51] F.N. Iandola, S. Han, M.W. Moskewicz, K. Ashraf, W.J. Dally, K. Keutzer, SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size, 2016, URL: https://arxiv.org/abs/1602.07360.
[52] S. Ha, Y. Yoon, J. Lee, Meta-ensemble learning with a multi-headed model for few-shot problems, ICT Express 9 (5) (2023) 909–914, http://dx.doi.org/10.1016/j.icte.2022.09.001.
[53] L. Torrey, J. Shavlik, Transfer learning, in: Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques, IGI global, 2010, pp. 242–264.
[54] B. Neyshabur, H. Sedghi, C. Zhang, What is being transferred in transfer learning? in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, vol. 33, Curran Associates, Inc., 2020, pp. 512–523.
[55] L. van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (86) (2008) 2579–2605, URL: http://jmlr.org/papers/v9/vandermaaten08a.html.
[56] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.
[57] F. Yan, M. Zhang, B. Wei, K. Ren, W. Jiang, FMC: Multimodal fake news detection based on multi-granularity feature fusion and contrastive learning, Alexandria Engineering Journal 109 (2024) 376–393.
[58] M. Ahammad, A. Sani, K. Rahman, M.T. Islam, M.M.R. Masud, M.M. Hassan, M.A.T. Rony, S.M.N. Alam, M.S.H. Mukta, Roberta-gcn: A novel approach for combating fake news in bangla using advanced language processing and graph convolutional networks, IEEE Access (2024).
[59] S.Y. Khamaiseh, D. Bagagem, A. Al-Alaj, M. Mancino, H.W. Alomari, Adversarial deep learning: A survey on adversarial attacks and defense mechanisms on image classification, IEEE Access 10 (2022) 102266–102291.
[60] D. Gragnaniello, F. Marra, G. Poggi, L. Verdoliva, Analysis of adversarial attacks against CNN-based image forgery detectors, in: 2018 26th European Signal Processing Conference (EUSIPCO), IEEE, 2018, pp. 967–971.
[61] M.T. Islam, I. Alam, S.S. Woo, S. Anwar, I. Lee, K. Muhammad, LoLI-Street: Benchmarking low-light image enhancement and beyond, in: Proceedings of the Asian Conference on Computer Vision, 2024, pp. 1250–1267.