



# DeepFake detection algorithm based on improved vision transformer

Young-Jin Heo<sup>1</sup> · Woon-Ha Yeo<sup>1</sup> · Byung-Gyu Kim<sup>1</sup>

Accepted: 7 June 2022 / Published online: 26 July 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

## Abstract

A DeepFake is a manipulated video made with generative deep learning technologies, such as generative adversarial networks or auto encoders that anyone can utilize. With the increase in DeepFakes, classifiers consisting of convolutional neural networks (CNN) that can distinguish them have been actively created. However, CNNs have a problem with overfitting and cannot consider the relation between local regions as global feature of image, resulting in misclassification. In this paper, we propose an efficient vision transformer model for DeepFake detection to extract both local and global features. We combine vector-concatenated CNN feature and patch-based positioning to interact with all positions to specify the artifact region. For the distillation token, the logit is trained using binary cross entropy through the sigmoid function. By adding this distillation, the proposed model is generalized to improve performance. From experiments, the proposed model outperforms the SOTA model by 0.006 AUC and 0.013 f1 score on the DFDC test dataset. For 2,500 fake videos, the proposed model correctly predicts 2,313 as fake, whereas the SOTA model predicts 2,276 in the best performance. With the ensemble method, the proposed model outperformed the SOTA model by 0.01 AUC. For Celeb-DF (v2) dataset, the proposed model achieves a high performance of 0.993 AUC and 0.978 f1 score, respectively.

**Keywords** Deep learning · Deepfake detection · Distillation · Generative adversarial network · Vision transformer

## 1 Introduction

With the development of artificial intelligence (AI) capabilities, deep learning models have been employed in various fields of computer vision [1–4]. Convolutional neural network (CNN) models are used in the fields of image classification, image generation, and object detection and have emerged as state-of-the-art (SOTA) methods that apply various training techniques. Recent generative adversarial network (GAN) face generation models perform very well, including StyleGAN [5], StarGAN [6], and InterfaceGAN [7]. Deepfake videos can easily be made using such networks.

DeepFake is a combination of “deep learning” and “fake” terms, which refers to the technique of changing the source person in a target video. This technology makes the impersonation seem to be performing actions or saying things that they never did or said. Notably, cases of abuse, such as fake news and revenge porn, have emerged as detrimental social issues. Therefore, technologies and datasets for detecting fake videos have been studied.

Deepfakes are created using GANs [8] and variational autoencoders (VAEs) [9]. The most well-known deepfake generation technique, in which two VAE models are trained to generate the faces of a person, only the decoder part is exchanged to create an image, as if only the face part is changed. The encoder part is shared by the two models, and the decoder part is trained separately. The recently created DeepFake detection dataset (DFDC: the world’s largest) [10] was applied to every frame by a Deepfake autoencoder with a morphable mask/neural network face swap to change the landmarks of the face. Face-swapping GANs use the neural talking-heads method and a GAN (e.g., StyleGAN) to generate DeepFake videos. Using the DFDC full dataset [10] released by Facebook AI, the EfficientNetwork-b7 [11] became a SOTA model.

---

✉ Byung-Gyu Kim  
bg.kim@sookmyung.ac.kr

Young-Jin Heo  
yj.heo@ivpl.sookmyung.ac.kr

Woon-Ha Yeo  
wh.yeo@ivpl.sookmyung.ac.kr

<sup>1</sup> Sookmyung Women’s University, Seoul, Republic of Korea

The procedure for finding fake videos is as follows. First, after locating the face in each frame, the image is cropped, and a deep learning model discriminates whether the image is fake. Fake videos can be detected by locating and detecting artificial parts within frames. An unnatural part is identified based on the spatial characteristics inside the frame. However, between frames, unnatural features can be found by their temporal characteristics. Networks for detecting edited parts of a frame mainly use a CNN structure because they do a good job of considering spatial characteristics. Additionally, anomaly discriminator methods, such as k-nearest neighbor and support vector machine (SVM) algorithms, work well. Instead of finding defects in one image, a recurrent neural network (RNN), which is normally used for natural language processing (NLP), can be used to find defects based on the temporal characteristics between frames. After the CNN model extracts the frame features, the RNN or a long short-term memory (LSTM) model determines whether the video is fake.

DeepFake detection requires clarification of goals because the results vary depending on which trait is looked for. Additionally, there is a problem in that the accuracy of real-world data is poor, depending on the trained data. In the DFDC private test dataset [10], most models incorrectly identify real videos as fake. To solve this problem, various training techniques and model innovations are required.

In addition, most current models are CNN-based architectures, but Jeffrey Hinton pointed out that the CNN model does not reflect the relationship of regional characteristics well. Hence, he proposed a capsule network [12] to mitigate the problem. As the CNN model goes to a higher level, more complex features are extracted and classified at the top layer. However, there is a disadvantage in that the positional relationship between simple and complex features cannot be considered because it is calculated as the weighted sum of the lower layers to the top layer. Therefore, for the current study, we use a model of a different structure that can consider the positional relationship of each face part, rather than the usual CNN-based approach.

In this study, we find fake videos using their spatial characteristics by leveraging a transformer model. Most DeepFake discrimination models use CNN-based networks as the SOTA model [11]. We also reveal the reason for this through detailed results analysis and discussion.

Our most relevant contributions are as follows:

- **We adopt an improved vision transformer:** An efficient deep architecture with a vision transformer (ViT) that can predict fake videos is designed.
- **We apply a distillation technique for Deepfake detection:** We apply a distillation method of data-efficient image transformers (DeiT) for Deepfake detection and show the results of various conditions.

- **We combine patch embedding and CNN features:** By combining the EfficientNet and patch features, we consider the advantages of two features and obtain higher area under the curve (AUC) and f1 scores than the SOTA [12] and other recent methods.

In summary, we design a DeepFake detection using a ViT model, which has shown good performance in recent image classifications. We combine CNN and patch-embedding features during the input stage. Also the proposed method uses the distillation technique, and the results show a higher performance than SOTA [11]. Moreover, our method shows better performance for fake videos, and we expect high performance in other test datasets as well.

This paper is organized as follows. Section 2 describes related works, and we introduce a model that considers spatial and temporal characteristics in detail. Section 3 proposes the scheme for the proposed DeepFake detection model. We explain the preprocessing process, the basic network, features combined with CNN, and patch and training processes. Section 4 presents the experimental results and analysis, and Section 5 concludes the paper.

## 2 Related works

Most configured models for DeepFake detection are based on a CNN structure. There are two approaches for discriminating DeepFake videos. One is to exploit unnatural spatial properties within one frame of video as an image unit, and the other is to exploit temporal properties to find unnaturalness between video frames.

Figure 1 represents the methods of DeepFake detection. The model finds artifacts using temporal characteristics and feature points using a CNN and puts sends them to a sequential network (e.g., RNN, LSTM, or GRU) in chronological order.

### 2.1 DeepFake detection using spatial properties

To detect spatial manipulation in the face, Li [13] used CNN models (i.e., VGG16 [14], ResNet50, ResNet101, and ResNet152 [15]). Nguyen proposed a capsule network that can detect various types of Deepfakes [12] by using features pretrained by VGG16 and suggested a capsule-forensic architecture. A classification method using an SVM was proposed by Yang [16], and Guarnera employed K-nearest neighbors and linear discriminant analysis [17]. However, owing to the limitations of CNNs, it is necessary to interact with and compare all parts of an image to detect falsified areas.

Until now, the best CNN model for this purpose was EfficientNet [18] on DFDC dataset. EfficientNet improved

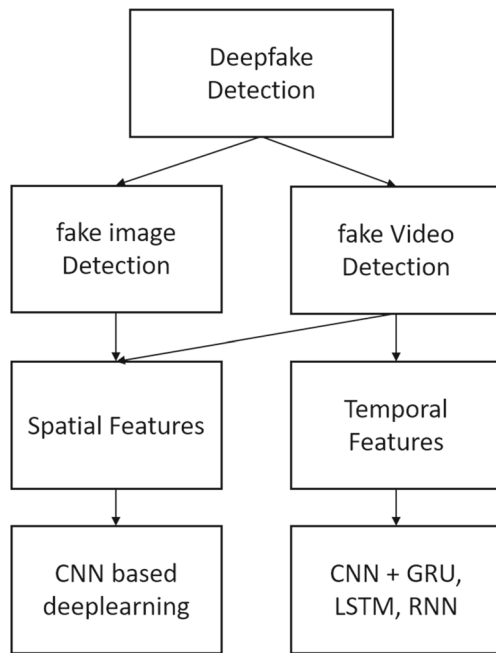


Fig. 1 DeepFake detection methods

the performance by applying several techniques to increase the number of filters by width scaling, the number of layers by depth scaling, and the resolution of the input image by resolution scaling [18]. The SOTA model based on EfficientNet achieved 0.981 AUC using the ensemble technique which was averaged the predictions of multiple trained models for DeepFake detection [11]. However, such models that use spatial characteristics with a two-dimensional (2D) CNN structure cannot correlate features in a distant position with temporal information. This makes it difficult for them to succeed.

Li et al. also proposed a novel image representation (i.e., face X-ray) for detecting forgery in facial images [19]. In this method, the face X-ray of an input face image is used to reveal whether the input image can be decomposed into blended images from different sources. Mitall et al. suggested an audio-visual DeepFake detection method using affective cues [20]. This approach employed a deep learning method inspired by the Siamese network architecture and triplet loss. Using this scheme, they achieved AUC of 0.844 on the DFDC dataset.

Based on this analysis, if we design an improved vision transformer (ViT) model to consider this relation information as a global feature, we are able to expect a performance increment for DeepFake detection task.

## 2.2 DeepFake detection using temporal properties

Figure 2 shows the structure of DeepFake video discrimination using temporal characteristics. Montserrat detected

space–time awkwardness by sending the frames of the video to EfficientNet and each feature from the frames into a gated recurrent unit [21]. Similarly, Güera used a CNN to extract frame-level features and train an RNN that learned to classify fake videos [22]. Unlike previous studies using CNN and RNN networks to determine spatiotemporal properties, de Lima [23] used a three-dimensional (3D) CNN to detect them simultaneously. They employed I3D [24], R3D [25], and MC3 [26] owing to their higher performance.

Using an optical flow-based CNN, Amerini applied an optical flow field to exploit possible inter-frame dissimilarities [27]. However, detection models using temporal properties tend to exhibit poor performance. Amerini [27] used Face2Face, which achieved an 81.61% of accuracy on 120 testing datasets [28]. Montserrat [21] achieved a 91.88% of accuracy on the DFDC test dataset. They successfully extracted the unnatural parts of the inter-frame, but most frames have similar features because almost all of the scenes are the same. Therefore, only temporal feature might be insufficient in those.

## 3 Proposed DeepFake detection algorithm

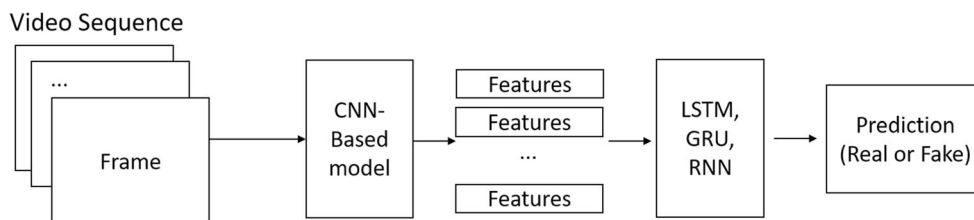
Before introducing our model, we describe its transformer structure and its advantages to DeepFake detection. Transformers are widely used in the natural language processing (NLP) field, but they also show good results in the field of computer vision. The Facebook AI team proposed a method that combined transformer model and distillation method [29]. We employ this model.

### 3.1 ViT for DeepFake detection

The CNN and ViT models have pros and cons in Deepfake detection. First, the CNN cannot learn the relation of different parts of the image. For example, the model cannot find an unnatural relationship between mouth and eyes that is out of synchronization. On the other side, the ViT learns their relationship to each position by assigning an order to patches of the given image. The input is embedded as patches with information on the positioning. These features are connected by Multi-Head Self Attention Layer (MSL) to know which part is fake.

In the Second, the ViT utilizes global information more than the CNN. The CNN uses a convolution filter that extracts crucial edges by filtering the surrounding pixel values regardless of absolute position. The multi-head self-attention layer in ViT makes it possible to embed information globally across the overall image. In [30], ViT have more global information than ResNet at lower layers and uniform representations. The CNN model has no information about the location, only information about

**Fig. 2** DeepFake detection for temporal characteristics



the surrounding pixel values. This characteristic can detect the unnaturalness of the surrounding pixels due to image synthesis. Therefore, we combine a CNN feature and patch embedding to get local and global spatial information.

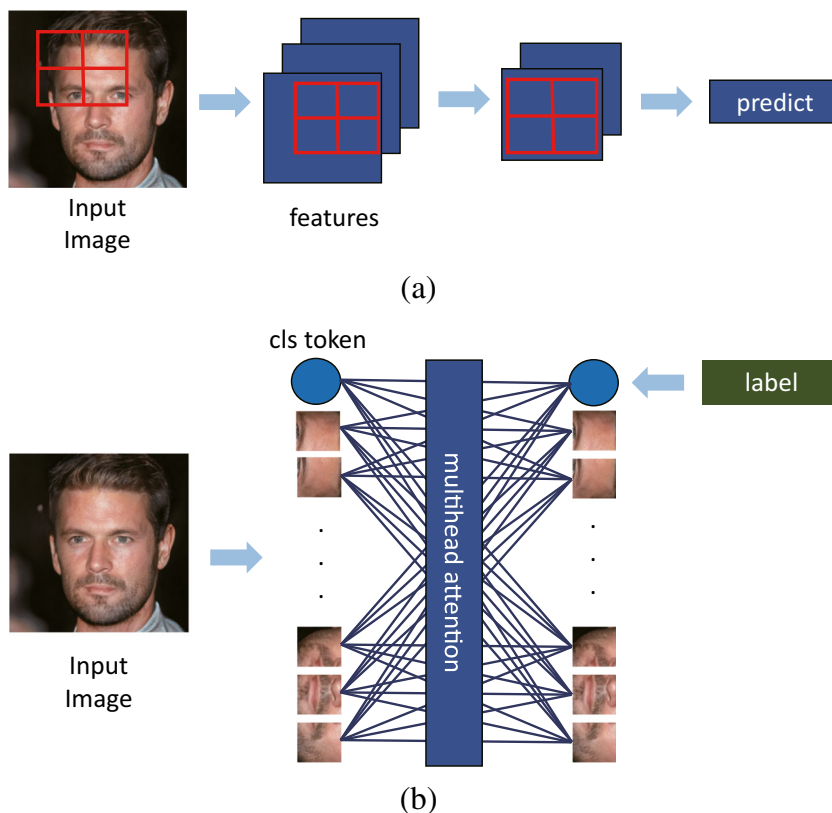
The features of the CNN model are gradually reduced by the CNN kernel through the entire image as input. Figure 3(a) illustrates the process of the CNN structure. Finally, it converges to a single feature and predicts the class of the image. To detect DeepFake images, the CNN model finds anomaly features by searching the entire face from the partial features of the face image.

As shown in Fig. 3(b) regarding the transformer, the cls token interacts with all partial features and interacts with each element to find the deeply related parts. If there are unsuitable features, they affect the specific area involved. Patches with a strong relationship with class tokens appear as active areas. Thus, the most relevant feature with class tokens is an important factor in predicting a DeepFake.

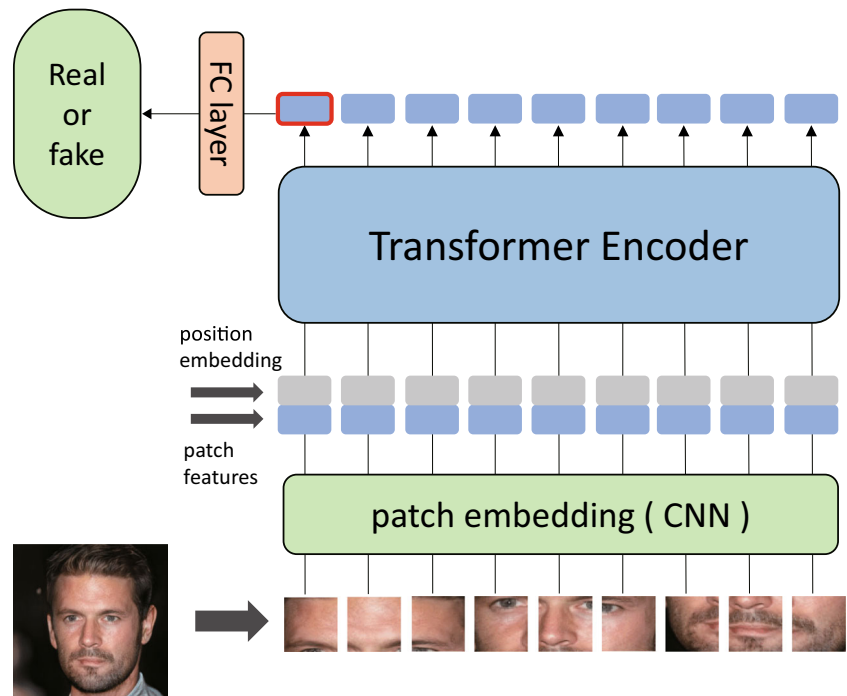
When the model finds the fake part of the face, the interactive weight of the class token is strong. In the real part, all the weights do not bounce and are distributed evenly. For this reason, the transformer has slightly more difficulty finding the real image than does the CNN. However, it has more success in finding a fake image by dividing it into patches and interacting with the class token. On the other hand, the CNN condenses it to a single vector from image features. We consider both CNN features and patch embedding features for their respective advantages.

Figure 4 shows the transformer training process for DeepFake detection. First, we split the face into the desired patch size, and it is patch embedded by one CNN layer using a patch size kernel. Features corresponding to each part of the face are fed to the input of the transformer and interact with each other. Finally, the class token predicts whether the image is real or fake through the fully connected layer.

**Fig. 3** (a) CNN structure process and (b) transformer process on face image



**Fig. 4** ViT for DeepFake detection



Thus, the transformer can detect fake videos by interacting with the unnatural area.

### 3.2 The proposed method

The procedure of the proposed Deepfake detection is shown in Fig. 5. The face is extracted from a video using a multitask cascaded convolutional network (MTCNN) model [31]. Then, a landmark is extracted to proceed with the augmentation, which drops out the face part from the image.

After face extraction from the video and preprocessing, the image enters a deep learning model. We contribute to the deep learning model of the entire process. As the output of the deep learning model, we can determine whether it is real or fake.

Our baseline follows the vision transformer network with a distillation token from DeiT. Input sequences were combined with patch embedding and CNN features. The entire network is shown in Fig. 6. We introduce our baseline model for DeepFake detection in Section 3.2.2 We illustrate that how the input consists in Section 3.2.3 and the

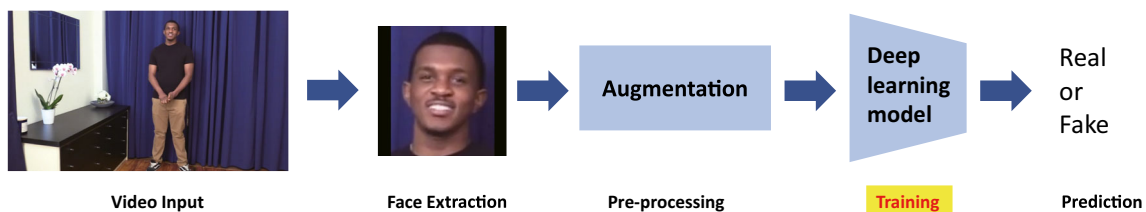
specific distillation training process with teacher network in Section 3.2.4.

#### 3.2.1 Data preprocessing

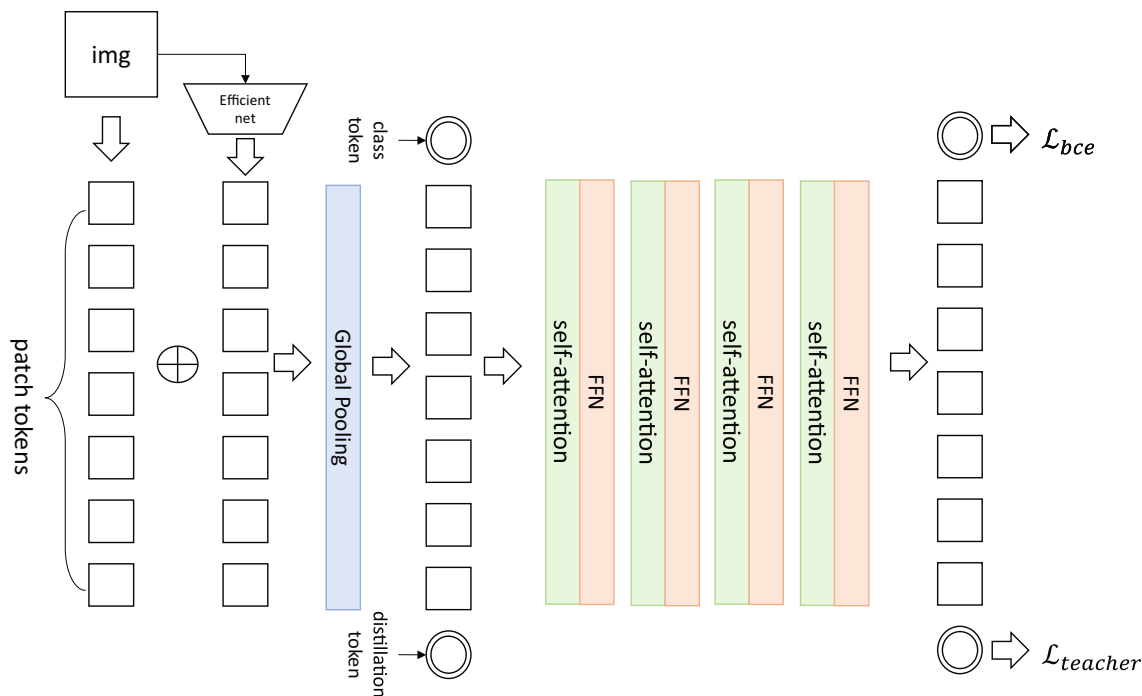
The face from the frames is extracted using the MTCNN model [31]. We generate landmarks in the cropped image and extract the structural similarity difference masks between the real and fake images. We follow [11] for this process. The reason for extracting landmarks is to cut out part of the face to make the model more general. This pre-processing prevents the model from overfitting. For example, the input image is preprocessed, as shown in Fig. 7. Finally, GaussNoise, GaussianBlur, HorizontalFlip, InotropicResize, and ShiftScaleRotate of Albumentations [32] are used for data augmentation.

#### 3.2.2 Basic network architecture

We overview the vision transformer [33] and recognize its efficacy in the field of DeepFake detection. The transformer



**Fig. 5** Proposed DeepFake detection procedure



**Fig. 6** The proposed overall DeepFake detection network. The image is split into patches and passes EfficientNet [18]. We obtained (Batch, N, embedding features) and (Batch, M, embedding features), respectively. These tokens are concatenated through global pooling and fed

to the transformer encoder. The encoder consists of Multi-Head Self Attention and two Gaussian error linear unit (GELU) layers which is feed-forward neural network (FFN). We add a distillation token trained by the teacher network

was originally used for NLP tasks; however, recently, many attempts have been made to apply it to image modeling [34–36]. The vision transformer has an encoder like the bidirectional encoder representations from transformers model, which uses position information and embedding sequences.

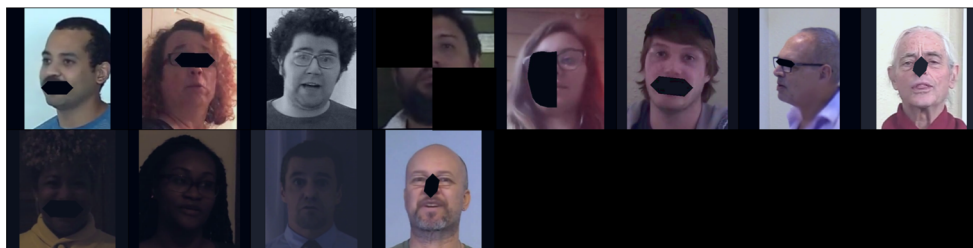
Before the Multi-Head Self Attention Layers (MSLs), the image,  $x \in \mathbb{R}^{(H \times W \times C)}$ , is split into patches,  $x_p \in \mathbb{R}^{(\frac{H}{P} \times \frac{W}{P} \times E)}$ , by learnable embedding, where (H,W) is the resolution of the image, C is the channel, P is the patch size, and E is the number of embedding features. All patches are flattened by linear projection and added to the position embedding equal to  $(\frac{H}{P} \times \frac{W}{P})$ . The transformer encoder consists of a Multi-Head Self Attention and multilayer perceptron (MLP). The MLP contains two layers with GELU non-linearity [33].

The sequences of feature vectors include all parts of the image. An encoder refers to all sequences of split patches. The previous CNN structure focused only on the activated part of the face and could not refer to other distant positions. However, input sequences depend on global information, which can reduce overfitting in transformers. We also find an interesting result in that the transformer makes a relatively fair classification of real and fake videos, rather than being skewed to either side, unlike the previous CNN models.

### 3.2.3 Combination of patch embedding and CNN features

We combine patch embedding and CNN features. Patch embedding determines the features of the patch of the face, and the CNN feature determines the overall features. Two

**Fig. 7** The result of data pre-processing [11]



features are combined and entered through global pooling. If both are considered, the performance is higher than when only single-patch embedding is compared.

Unlike the original input vectors of the vision transformer, we introduce input tokens before feeding the encoder. We define  $\mathbf{Z}_p = (x_p^1 \mathbf{E}, x_p^2 \mathbf{E}, \dots, x_p^N \mathbf{E})$  and  $\mathbf{Z}_f = f(x) = (x_f^1, x_f^2, \dots, x_f^M)$ , where  $x_p$  is a patch,  $\mathbf{E}$  is a learnable embedding,  $N$  is the exponential number of split patches,  $M$  is the number of CNN features, and  $f(\cdot)$  is the CNN model. Thus,  $\mathbf{Z}_f$  is a feature of the CNN model, and we use  $f$  as EfficientNet.

These features are combined as  $\mathbf{Z}_p \oplus \mathbf{Z}_f$  ( $\oplus$  means concatenating features by channels), and global pooling is applied. Min Lin suggested that global average pooling is more interpretable between feature maps and categories [37]. Thus, we represent  $\mathbf{Z}_{p \oplus f} = \text{globalpooling}(\mathbf{Z}_p \oplus \mathbf{Z}_f)$  as input vectors and  $N + M$  to  $N$  (vector number). As a result, we consider not only the main part features of the face but also the correlation of all parts.

The transformer's input features can interact from patch to patch, and the CNN features can interact with the surrounding features. By using this approach, we can obtain better AUC and f1 scores than using only patch embedding or CNN features.

### 3.2.4 Distillation method and teacher network

We have  $x_{class}$  and  $x_{distillation}$  tokens. The class token is trained by the true label value, and the distillation token is trained based on the prediction value of the teacher. To achieve a higher performance than the current SOTA model [11], the teacher is set the same as the SOTA model [11]. If the distillation token is not added, overfitting occurs.

We add class tokens and distillation tokens to input  $\mathbf{Z}_{p \oplus f}$ , and we define the final input,  $\mathbf{Z}_0 = [x_{class}; \mathbf{Z}_{p \oplus f}; x_{distillation}] + \mathbf{E}_{pos}$ , where  $x_{class}$  and  $x_{distillation}$  are tokens for training by the label and teacher network, and  $\mathbf{E}_{pos}$  is the learnable position embedding. Finally, we can define the set train loss as

$$\mathcal{L}_{fake} = \lambda \mathcal{L}_{BCE}([Z_{c_{fake}}], y) + (1 - \lambda) \mathcal{L}_{BCE}(Z_{d_{fake}}, \sigma([Z_{t_{fake}}])), \quad (1)$$

$$\mathcal{L}_{real} = \lambda \mathcal{L}_{BCE}(Z_{c_{real}}, y) + (1 - \lambda) \mathcal{L}_{BCE}(Z_{d_{real}}, \sigma(Z_{t_{real}})), \quad (2)$$

$$\mathcal{L}_{train} = \frac{\mathcal{L}_{fake} + \mathcal{L}_{real}}{2}, \quad (3)$$

where  $Z_{t_{fake}}$  and  $Z_{t_{real}}$  are the logits of the teacher model for fake and real prediction,  $(Z_{d_{fake}}, Z_{d_{real}})$  and  $(Z_{c_{fake}}, Z_{c_{real}})$  are the logits of the distillation tokens and the class

tokens for fake prediction and real prediction, respectively. We set  $\lambda$  by  $\frac{1}{2}$  through experimental analysis and binary cross entropy ( $\mathcal{L}_{BCE}$ ) on the labels,  $y$ , and  $\sigma$  as the sigma function.

In [29] regarding Facebook AI, a distillation method prevents overfitting by expanding the range of weights of labels. Also, when the teacher network is the CNN model, the transformer produces the best results compared with the other models.

Therefore, we chose the teacher network, EfficientNet, which is the state-of-the-art model on the DFDC dataset for DeepFake detection. Each class and distillation token represent the probability that the video is fake. The distillation tokens are used instead of the class tokens when testing. It can be seen that it outperforms when the distillation token is used on the test dataset.

The proposed scheme is efficient in detecting fake videos because we utilize distillation methodology to generalize the model and combine the CNN and patch-embedding features to gather more contextual information.

## 4 Experimental results

Here, we describe the dataset and the details of the parameters. We also compare the proposed scheme with the SOTA model [11], Li [19], Mittal [20] for the DFDC dataset, and I3D [24], R3D [25], MC3 [26] for the Celeb-DF (v2) dataset, which represent the condition of performance measurements. We explain why we used the DFDC and Celeb-DF (v2) datasets in Section 4.1. We describe the parameter setting and configuration environments required in the training process in Section 4.2 and analyze the experimental results in Section 4.3.

### 4.1 Datasets

#### 4.1.1 DFDC Dataset

In a Kaggle competition<sup>1</sup>, the DFDC dataset was previewed [38]. Later, the Facebook AI team opened the full version [10], which is the largest publicly available DeepFake dataset, and it includes approximately 100,000 videos produced by GANs. Figure 8 presents an example.

In a DeepFake dataset survey [10], face-swap datasets were divided into three generations. First-generation datasets, such as DF-TIMIT [39], UADFV [16], and Face-Forensics++DF (FF++DF) [40], have  $10^4 \sim 10^6$  frames and up to 5,000 videos. Second-generation datasets include

<sup>1</sup><https://www.kaggle.com/c/deepfake-detection-challenge>

**Fig. 8** Examples of DFDC Dataset [10]



Celeb-DF [41] and DFDC preview [38]. The DFDC full dataset is third-generation and has 128,154 total videos and 104,500 unique fake videos.

Because the data size is large compared to other datasets, we chose the largest DeepFake dataset and compared its performance to that of the SOTA model [11] on the DFDC full dataset. In the analysis of Dolhansky [10], the submitted best model has an AUC of 0.734 on a private test set. Also, the higher the average precision of the submitted models ([11, 42–45]) on the DFDC dataset, the better the

performance in real videos. Therefore, if the performance is good with the DFDC dataset, we can assume that the results can be generalized to real videos.

#### 4.1.2 Celeb-DF (v2) datasets

The Celeb-DF (v2) dataset contains real and DeepFake synthesized videos having similar visual quality on par with those circulated online [41]. The Celeb-DF (v2) dataset is greatly extended from our previous Celeb-DF (v1), which

**Fig. 9** Examples of Celeb-DF (v2) dataset

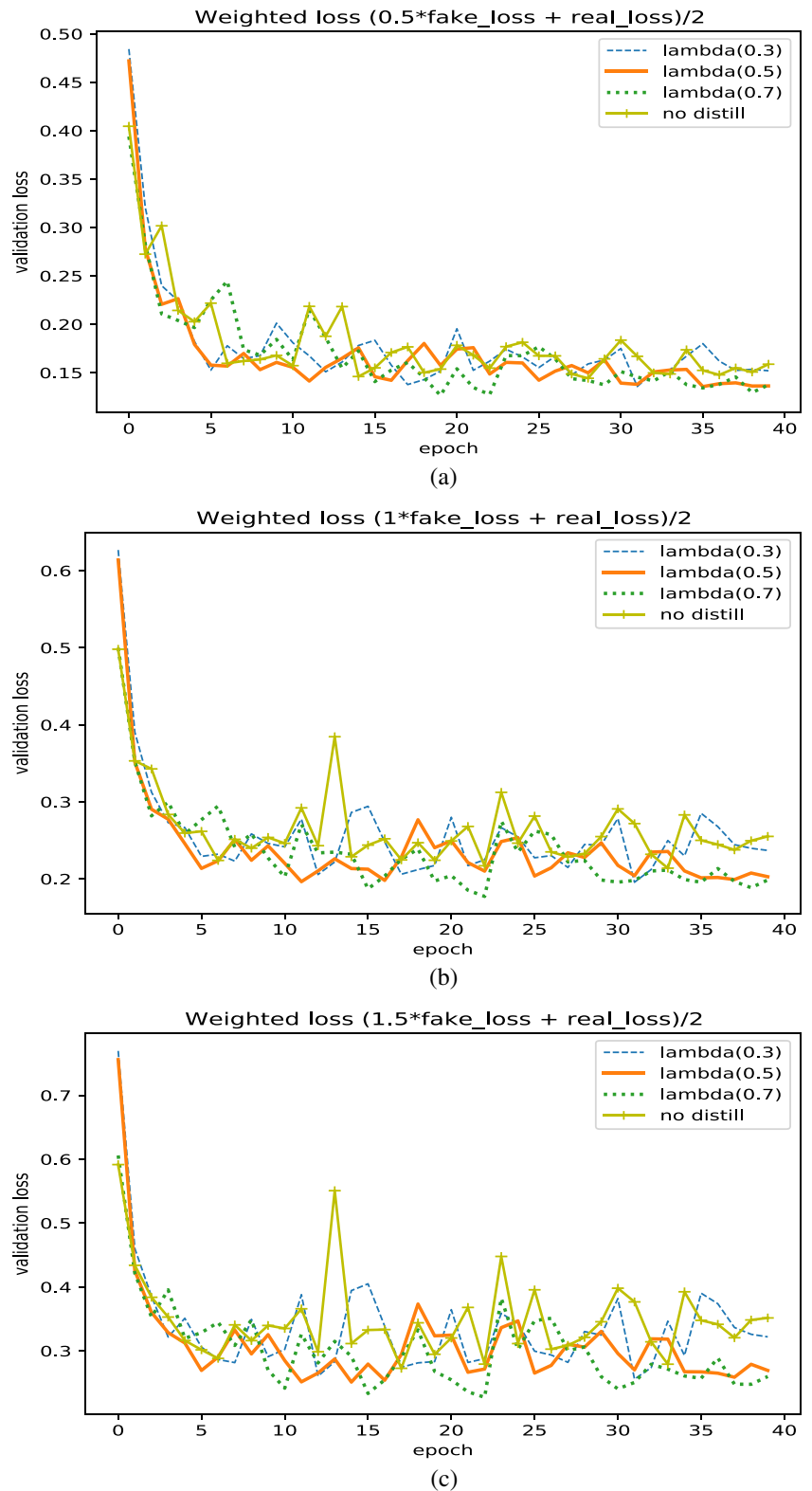




only contains 795 DeepFake videos. To date, Celeb-DF includes 590 original videos collected from YouTube with subjects of different ages, ethnic groups and genders, and

5639 corresponding DeepFake videos. Figure 9 shows some examples of the Celeb-DF (v2) dataset. This is smaller one comparing with the DFDC dataset.

**Fig. 10** Validation loss result according to the weight of fake loss: (a) weight 0.5, (b) weight 1.0, and (c) weight 1.5 on the DFDC dataset



**Table 1** AUC results according to  $\lambda$  values in the test dataset

$\lambda$ values	AUC
0.3	0.976
<b>0.5</b>	<b>0.978</b>
0.7	0.976
no distillation	0.974

## 4.2 Training and testing detail

### 4.2.1 Training detail

We take a pre-processing stage initially. After the pre-processing, images are loaded randomly for training process with their labeled information.

**Pre-processing:** We used a face detector as the MTCNN [31] and cropped all frames to  $384 \times 384$ . We augmented our training data using albumentations [32]. We also cut out and dropped out part of the image, based on [11].

**Training:** The patch size for the embedding features was 32, and the embedding dimension was 1,024. We initialized our transformer and the EfficientNet-B7 model [11] using the pretrained model. We set the transformer to 16 heads and 24 layers, which is identical to the large ViT default model. Additionally, our teacher network used a pre-trained network, EfficientNet-B7 [11] on the DFDC dataset. We used a distillation token only during testing.

**Parameters:** Training and testing were performed on a V100 GPU machine with a batch size of 12 for training. We used a stochastic gradient descent optimizer with an initial learning rate of 0.01 and a differential learning rate reduction policy, which is a step-based method. The training epoch was 40, batches per epoch were 2,500, and it took 2 days on a single V100 GPU.

For classification, we used binary cross entropy for backward values. We tested a publicly available DFDC test dataset of 5,000 videos. The f1 score was measured for comparison with the SOTA model [11] with a 0.55 of threshold value, which was selected through experiments. Also, we will check on the performance at the best threshold value of each method.

### 4.2.2 Testing detail

To detect DeepFake video, we employed the same procedure in the SOTA model [11]. During testing, 32 frames per

video are selected at regular intervals. If an image number with a predicted value larger than 0.8 is 12 frames or more, the predicted values are averaged. If number of images with predicted values smaller than 0.2 is bigger than 90%, the predicted values are averaged. In other case, the average of all predicted values is calculated. As a result, a value from 0 to 1 indicates whether the video is fake or not.

## 4.3 Ablation Study

We show the results of training and testing according to the  $\lambda$  value when applying the distillation method during training. We analyzed the results when the distillation method was not applied and the f1 score experiment result according to the threshold value.

Figure 10 shows that the deviation increases as the weight for fake loss increases. The validation loss was mostly similar, but we can see that the loss was relatively higher than when the distillation technique was not applied.

Table 1 shows the AUC result according to the  $\lambda$  value when training with the distillation method. The highest AUC value was obtained at 0.5 (as shown in boldface), and the lowest AUC value was obtained when no distillation was applied. In order to consider other intervals of values as interpolation, we experimented with three  $\lambda$  values.

Table 2 is the result of the f1 score according to the threshold value ( $\beta$ ), which is the probability of determining an image as fake. The boldface denotes the best performance of each algorithm. The proposed achieved the highest 0.919 f1 score at 0.55 and the highest 0.911 f1 score at 0.4 in the SOTA model [11]. The best threshold value of  $\beta$  has been selected by analyzing the performance (f1 score) as the variation of  $\beta$ . Since the different network structure is used, the best threshold value may be changed due to different training characteristics. When comparing the best performance, the proposed method outperformed the SOTA model [11] by a factor of 0.8% (0.008 of AUC).

Also, we concatenated CNN features and patch embedding to consider local and global information. Without using CNN features, we achieved 0.959 AUC and 0.891 f1 score, respectively. This result describes that the combination of CNN feature with patch embedding is very effective.

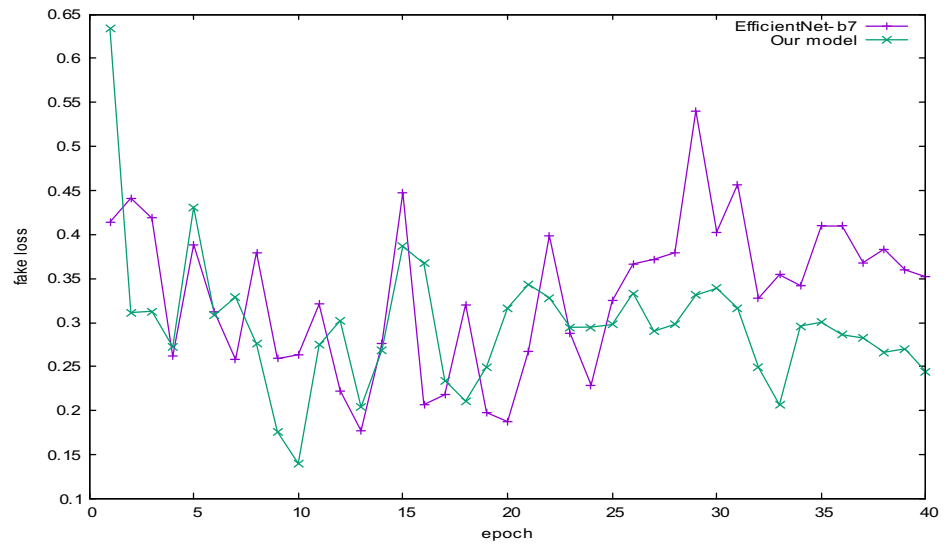
## 4.4 Performance analysis

We compared our model to the SOTA model [11]. We trained the proposed model using the training dataset and

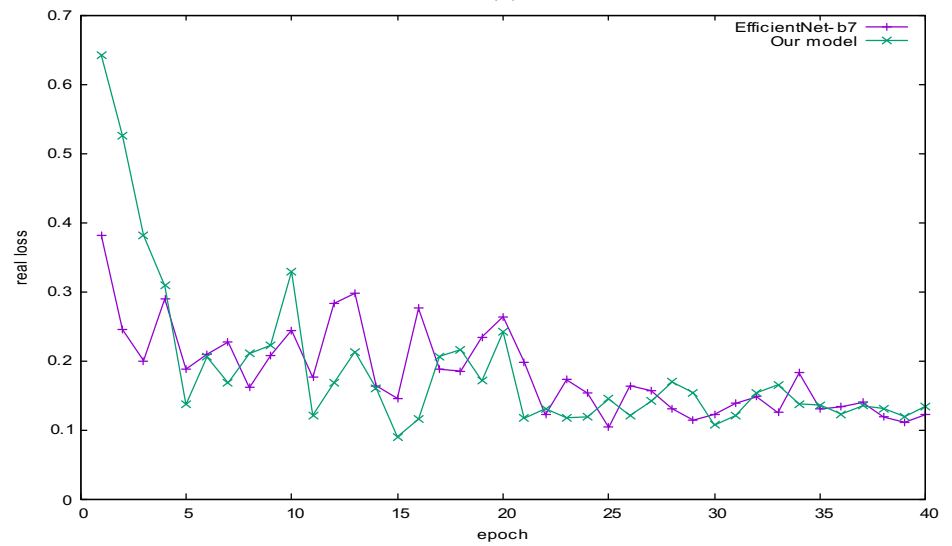
**Table 2** The result of f1 score according to the threshold value in the test dataset

Threshold value ( $\beta$ )	0.3	0.4	0.5	0.55	0.6	0.7	0.8	0.9
SOTA [11]	0.904	<b>0.911</b>	0.908	0.906	0.904	0.901	0.901	0.898
Proposed Method	0.894	0.910	0.917	<b>0.919</b>	0.917	0.916	0.914	0.914

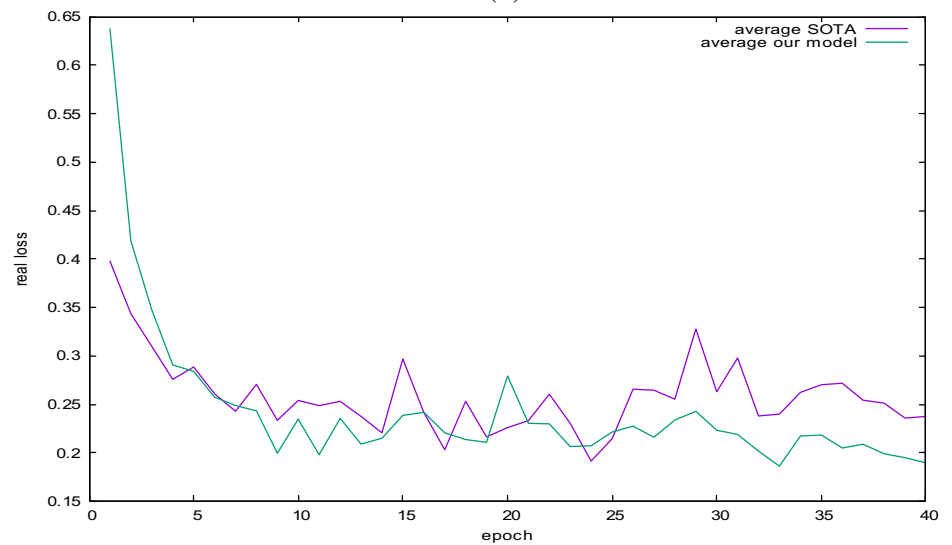
**Fig. 11** Results of loss between SOTA model and our model in validation DFDC dataset: (a) the loss for fake video, (b) the loss for real video, and (c) the average loss



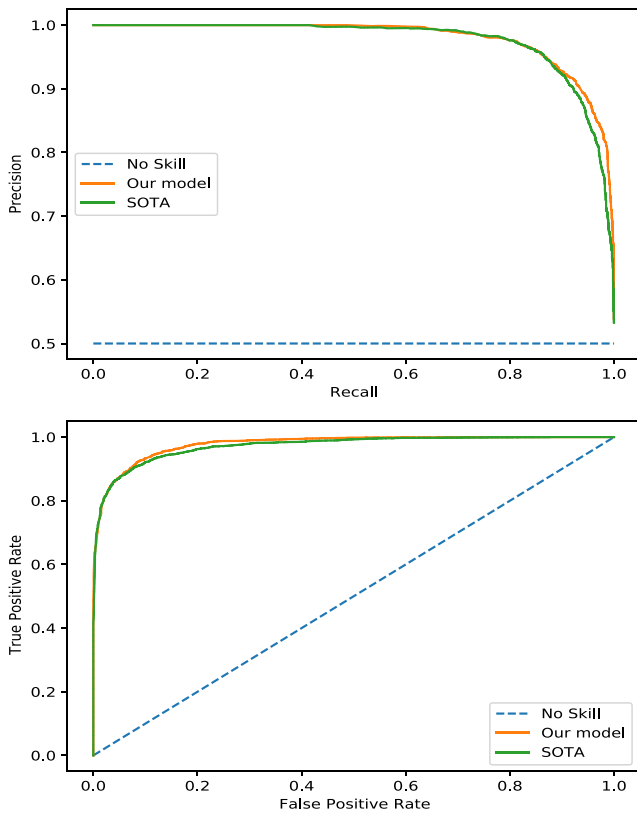
(a)



(b)



(c)



**Fig. 12** Receiver operating characteristic (ROC) and AUC curves. Orange is the curve of our model, and green is the curve of the SOTA model [11]. The proposed model has a larger area than the compared SOTA model [11] on the DFDC dataset

chose the model weights with the lowest loss in the validation set. In Fig. 11, we compare the validation loss to the SOTA model [11] for real and fake videos. The green

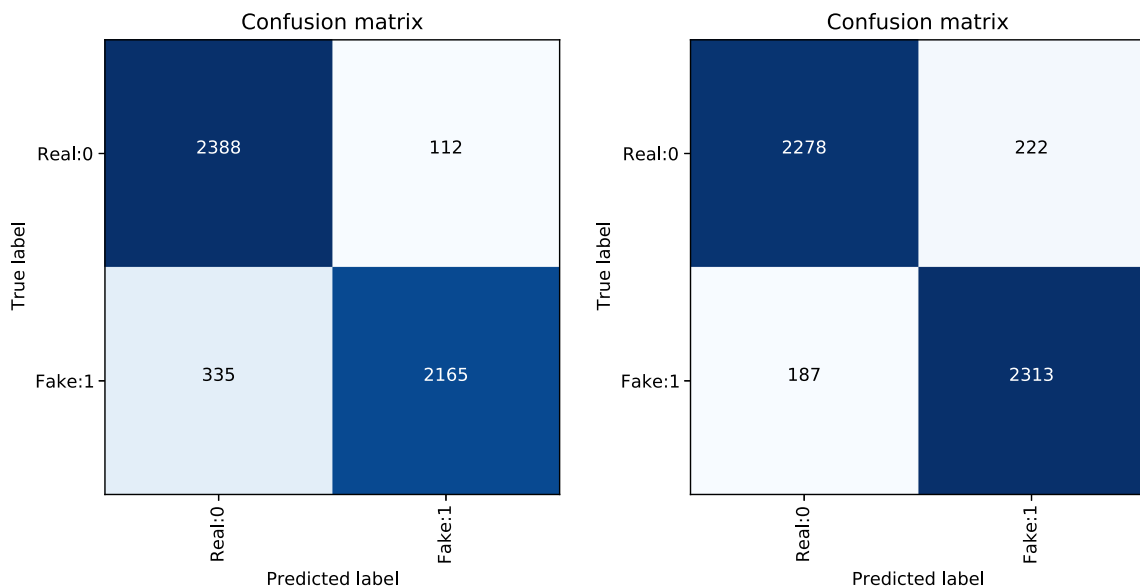
plot indicates our model’s loss, and the purple plot indicates the SOTA model’s [11] loss. There was a slight difference in the loss of the real video, but there was a significant difference in that of the fake video. This graph shows that our model is a more robust classifier for fake videos. Although the real loss was similar, the overall average loss was lower. The validation loss is defined as

$$Log Loss = -\frac{1}{n} \sum_{i=1}^n [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)], \quad (4)$$

where  $n$  is the number of videos being predicted,  $\hat{y}_i$  is the predicted probability of the video being fake,  $y_i$  is one if the video is fake and zero if real. We obtained  $\hat{y}_i$  using distillation tokens.

In addition, the ROC–AUC curve of the proposed model has a larger area (0.978) than that of the SOTA model [11] (0.972) in Fig. 12. This indicates that the proposed classifier is more robust on fake videos because the precision was higher than that of the SOTA model [11], and the recall was close to one.

To verify robustness, a confusion matrix was obtained by setting a threshold of 0.55, which represents the probability of a fake video. It represents the predicted number of videos according to each label in Fig. 13. Top-right, top-left, bottom-right, and bottom-left represent false positive, true negative, true positive, and false negative, respectively. The confusion matrix of the left side is that of the previous SOTA model [11] prediction, and the right side shows our results. We can see that our model clearly predicts fake videos. The false-negative results for each model were 335 and 187, respectively. Thus, the proposed model is robust with fake video detection, and the f1 score of all cases was 0.919 as



**Fig. 13** Confusion matrix from the previous SOTA model which is EfficientNet-B7 [11] (left) and the proposed algorithm (right) on the DFDC dataset when threshold  $\beta=0.55$

**Table 3** AUCs, f1 scores, and the complexity of the recent methods, SOTA, and the proposed model on the DFDC dataset

Methods	AUC	f1 score	# of parameters
Li [19]	0.809	-	-
Mittal [20]	0.844	-	-
Selim (SOTA) [11]	0.972	0.906	67.7M
Proposed model	<b>0.978</b>	<b>0.919</b>	<b>440.2M</b>

shown in boldface, which is higher than the 0.906 of the SOTA model [11] shown in Table 3.

To compare the performance in the best threshold condition, we displayed another confusion matrix in Fig. 14. The best condition of the SOTA model [11] was at  $\beta=0.40$  and  $\beta=0.55$  in the proposed model. For 2,500 fake videos in these conditions, the proposed model correctly predicted 2,313 as fake, whereas the SOTA model predicted 2,276 in number. Also, the false-negative results for each model were 224 and 187, respectively. From this result, we can see that the proposed model is robust with fake video detection.

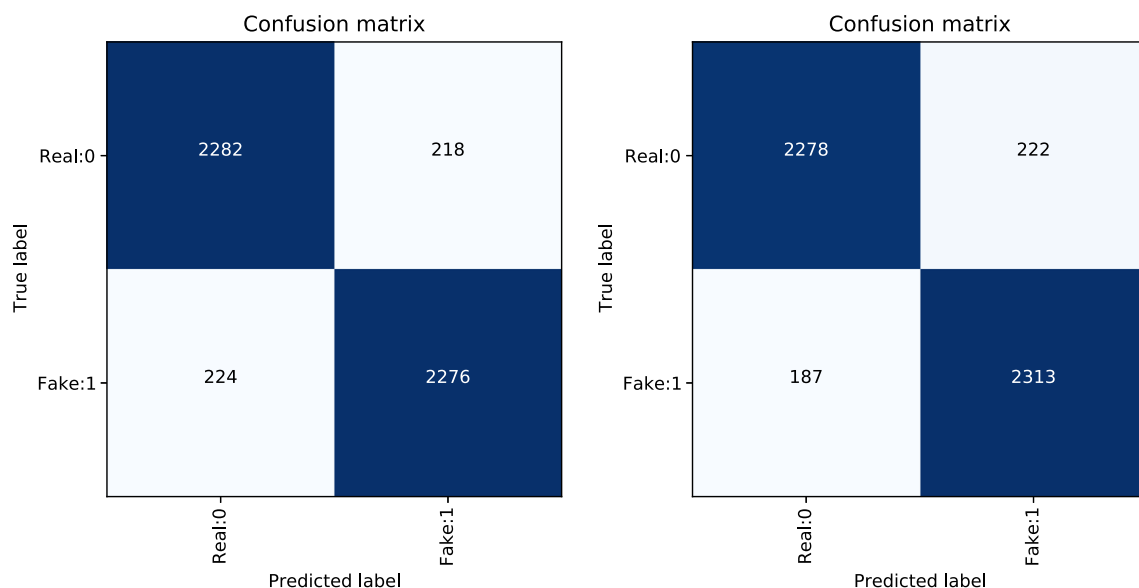
We also compared AUC values to other recent methodologies, such as a scheme based on the face X-ray network [19], a DeepFake detection method using emotion audiovisual affective cues [20], and the SOTA model [11], as shown in Table 3. Generally, DeepFake detection methods focus on manipulation artifacts. However, Li et al. [19] proposed a novel face X-ray image representation, which focuses on blending artifacts. They predicted the boundary of the manipulated face and obtained an AUC score of 0.809 in the DFDC test dataset.

Mittal et al. used audio and visual modalities from within the same video to determine similarity [20]. Their training method was like a Siamese network with facial and speech features. This scheme achieved a 0.844 AUC score for the DFDC test dataset. When compared with these methods, the proposed algorithm yielded a significantly improved AUC score (0.978 AUC score), as shown in Table 3. When comparing to Mittal [20], the proposed method was superior by 0.13 AUC even though the proposed model used only image (facial) feature.

From Table 3, the proposed method is only 0.006 higher in AUC than Selim method (SOTA) [11]. But we observed that the proposed scheme outperformed Selim method (SOTA) [11] by 0.013 f1 score. Also, the proposed method showed much better performance in finding fake video (True Positive). In DeepFake detection task, we think that the detection of fake video is more important than that of real video. In this viewpoint, the proposed scheme is useful enough in DeepFake detection task.

We also compared the ensemble results of the SOTA [11] and the proposed model. We trained five times and tested it by averaging the probability values. As a result, the AUC result of the SOTA model [11] was 0.981, and the proposed model achieved an AUC of 0.982. Thus, the proposed scheme can detect fake videos more robustly than the SOTA model [11].

Additionally, we trained and tested the proposed scheme by same process with single model on the Celeb-DF (v2) dataset. The Celeb-DF (v2) dataset [41] has 590 real videos and 5639 fake videos that is synthesised with high quality. Table 4 shows the AUCs, F1 scores, and the complexity

**Fig. 14** Confusion matrix from the previous SOTA model which is EfficientNet-B7 [11] (best threshold  $\beta=0.40$ ) (left) and the proposed algorithm (best threshold  $\beta=0.55$ ) (right) on the DFDC dataset

**Table 4** AUCs, f1 scores, and the complexity of the recent methods, SOTA, and the proposed model on the Celeb-DF (v2) dataset [41]

Methods	AUC	f1 score	# of parameters
I3D [24]	0.976	-	25.0M
MC3 [26]	0.993	-	11.7M
R3D [25]	0.997	-	33.3M
Proposed model	<b>0.993</b>	<b>0.978</b>	<b>440.2M</b>

of the proposed and recent existing models on the Celeb-DF (v2) dataset [41]. The boldface denotes the results of the proposed algorithm. As in [23], they used 3D CNNs to consider both spatial and temporal information [24, 26], [25]. Except R3D scheme [25], the proposed model gave better AUC. This means that the improved ViT model is able to give good performance for other DeepFake dataset.

Since the proposed scheme uses an improved ViT structure shown in Fig. 6, the computational complexity is inevitably higher than that of ViT. In addition, the proposed scheme uses the feature from EfficientNet [11] together. The complexity of the proposed model is almost 8 ~ 10 times because of modification of ViT as shown in Tables 3 and 4. The transformer utilizes the attention mechanism to compute the correlation crossing on all tokens. This attention module has heavy parameters than CNN structures without attention module. Also, the number of convolution layers should be increased to make higher accuracy in classification task. But a problem is that we can observe the performance saturation (not improved more) although the layer number is increased continuously. To solve this problem, transformer is being widely utilised. With this, we designed the proposed model to improve the detection performance of DeepFake image based on the ViT model.

Despite its high complexity, the proposed algorithm has shown better performance for Deepfake detection by designing distillation techniques and combining CNN features with patch embedding. From Tables 3 and 4, we can see that the proposed scheme achieves better AUC and f1 score on the DFDC and Seleb-DF (v2) datasets. Especially, we observed very high f1 score on the Celeb-DF (v2) dataset [41].

## 5 Conclusion

In this paper, we proposed an improved vision transformer model for DeepFake detection. The proposed scheme is a combination of patch embedding and CNN features utilizing a distillation token based on DeiT. By considering the characteristics of the CNN and the transformer, we verified superior performance over previous results.

We demonstrated the efficiency of the robust vision transformer model compared with EfficientNet as the SOTA

model, which consists of a 2D CNN network. The SOTA obtained an AUC of 0.972, whereas ours obtained 0.978 under the same conditions without an ensemble approach. The proposed scheme produced an f1 score of 0.919, whereas the SOTA model achieved 0.906 under the same threshold condition of 0.55. Furthermore, we observed an AUC improvement of up to 0.17 compared with a recent scheme [19, 20]. With the ensemble method, the proposed model achieved an AUC of 0.982, whereas the SOTA model achieved 0.981 [11]. In addition, we verified 0.993 AUC and 0.978 of f1 score for the Celeb-DF (v2) dataset.

In future work, we will investigate more detailed unnaturalness between frames for DeepFake detection. If the spatial feature is only considered, motion information between adjacent frames of the DeepFake or the synthesized pixel portion within one frame may be missed. Therefore, we will study further a hybrid ViT model which can combine spatial feature with temporal feature, efficiently.

## References

- Choi Y-J, Lee YW, Kim B-G (2021) Group-based bi-directional recurrent wavelet neural networks for video super-resolution, arXiv:2106.07190
- Jeong D, Kim BG, Dong S-Y (2020) Deep joint spatiotemporal network (djstn) for efficient facial expression recognition. *Sensors* 20(7):1936
- Yeo W-H, Heo Y-J, Choi Y-J, Kim B-G (2020) Place classification algorithm based on semantic segmented objects. *Appl Sci* 10(24):9069
- Heo Y-J, Choi Y-J, Lee Y-W, Kim B-G (2021) Deepfake detection scheme based on vision transformer and distillation, arXiv:2104.01353
- Karras T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, pp 4401–4410
- Choi Y, Choi M, Kim M, Ha J-W, Kim S, Choo J (2018) Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: proceedings of the IEEE conference on computer vision and pattern recognition, pp 8789–8797
- Shen Y, Yang C, Tang X, Zhou B (2020) Interfacegan: Interpreting the disentangled face representation learned by gans, *IEEE Transactions on Pattern Analysis and Machine Intelligence*
- Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. *Advances in neural information processing systems*, 27
- Kingma DP, Welling M (2014) Stochastic gradient vb and the variational auto-encoder. In: Second international conference on learning representations, ICLR, vol 19, p 121
- Dolhansky B, Bitton J, Pflaum B, Lu J, Howes R, Wang M, Ferrer CC (2020) The deepfake detection challenge dataset, arXiv preprint arXiv arXiv:2006.07397
- Seferbekov S (2020) [https://github.com/selimsef/dfdc\\_deepfake\\_challenge](https://github.com/selimsef/dfdc_deepfake_challenge). Accessed 24 Jan 2022
- Nguyen HH, Yamagishi Y, Echizen I (2019) Use of a capsule network to detect fake images and videos, arXiv:1910.12467
- Li Y, Lyu S (2019) Exposing deepfake videos by detecting face warping artifacts. In: CVPR Workshops

14. Lui S, Deng W (2015) Very deep convolutional neural network based image classification using small training sample size. In: 2015 3rd IAPR Asian conference on pattern recognition (ACPR), pp 730–734 IEEE
15. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
16. Yang X, Li Y, Lyu S (2019) Exposing deep fakes using inconsistent head poses. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 8261–8265. IEEE
17. Guarnera L, Giudice O, Battiato S (2020) Deepfake detection by analyzing convolutional traces. In: proceedings of the IEEE/CVF Conference on computer vision and pattern recognition workshops, pp 666–667
18. Tan M, Le Q (2019) Efficientnet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. PMLR
19. Li L, Bao J, Zhang T, Yang H, Chen D, Wen F, Guo B (2020) Face x-ray for more general face forgery detection. In: proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, pp 5001–5010
20. Mittal T, Bhattacharya U, Chandra R, Bera A, Manocha D (2020) Emotions don't lie: an audio-visual deepfake detection method using affective cues. In: proceedings of the 28th ACM international conference on multimedia, pp 2823–2832
21. Montserrat DM, Hao H, Yarlagadda SK, Baireddy S, Shao R, Horváth J, Bartusiak E, Yang J, Guera D, Zhu F et al (2020) Deepfakes detection with automatic face weighting. In: proceedings of the IEEE/CVF Conference on computer vision and pattern recognition workshops, pp 668–669
22. Güera D, Delp EJ (2018) Deepfake video detection using recurrent neural networks. In: 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp 1–6. IEEE
23. de Lima O, Franklin S, Basu S, Karwoski B, George A (2020) Deepfake detection using spatiotemporal convolutional networks, arXiv:2006.14749
24. Carreira J, Zisserman A (2017) Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on computer vision and pattern recognition, pp 6299–6308
25. Hara K, Kataoka H, Satoh Y (2017) Learning spatio-temporal features with 3d residual networks for action recognition. In: proceedings of the IEEE International conference on computer vision workshops, pp 3154–3160
26. Tran D, Wang H, Torresani L, Ray J, LeCun Y, Paluri M (2018) A closer look at spatiotemporal convolutions for action recognition. In: proceedings of the IEEE conference on computer vision and pattern recognition, pp 6450–6459
27. Amerini I, Galteri L, Caldelli R, Del Bimbo A (2019) Deepfake video detection through optical flow based cnn. In: proceedings of the IEEE/CVF International conference on computer vision workshops, pp 0–0
28. Thies J, Zollhofer M, Stamminger M, Theobalt C, Nießner M (2016) Face2face: Real-time face capture and reenactment of rgb videos. In: proceedings of the IEEE conference on computer vision and pattern recognition, pp 2387–2395
29. Touvron H, Cord M, Douze M, Massa F, Sablayrolles A, Jégou A (2021) Training data-efficient image transformers & distillation through attention. PMLR
30. Raghu M, Unterthiner T, Kornblith S, Zhang C, Dosovitskiy A (2021) Do vision transformers see like convolutional neural networks?. *Advances in Neural Information Processing Systems*, vol 34
31. Zhang K, Zhang Z, Li Z, Qiao Y (2016) Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters* 23(10):1499–1503
32. Buslaev A, Iglavikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA (2020) Albumentations: fast and flexible image augmentations. *Information* 11(2):125
33. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S et al (2020) An image is worth 16x16 words: Transformers for image recognition at scale, arXiv:2010.11929
34. Girdhar R, Carreira J, Doersch C, Zisserman A (2019) Video action transformer network. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 244–253
35. Neimark D, Bar O, Zohar M, Asselmann D (2021) Video transformer network, arXiv:2102.00719
36. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: Hierarchical vision transformer using shifted windows, *International Conference on Computer Vision (ICCV)*
37. Lin M, Chen Q, Yan S (2013) Network in network, arXiv:1312.4400
38. Dolhansky B, Howes R, Pflaum B, Baram N, Ferrer CC (2019) The deepfake detection challenge (dfdc) preview dataset, arXiv:1910.08854
39. Korshunov P, Marcel S (2018) Deepfakes: a new threat to face recognition? assessment and detection, arXiv:1812.08685
40. Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Nießner M (2019) Faceforensics++: Learning to detect manipulated facial images. In: proceedings of the IEEE/CVF International Conference on Computer Vision, pp 1–11
41. Li Y, Yang X, Sun P, Qi H, Lyu S (2020) Celeb-df: a large-scale challenging dataset for deepfake forensics. In: proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, pp 3207–3216
42. Zhao H, Cui H, Zhou W (2020) <https://github.com/cuihao/le/kaggle-dfdc>. Accessed 24 Jan 2022
43. Davletshin A (2020) <https://github.com/NTech-Lab/deepfake-detection-challenge>. Accessed 24 Jan 2022
44. Shao J, Shi H, Yin Z, Fang Z, Yin G, Chen S, Ning N, Liu Y (2020) <https://github.com/Siyu-C/RobustForensics>. Accessed 24 Jan 2022
45. Howard J, Pan I (2020) <https://github.com/jphdotam/DFDC/>. Accessed 24 Jan 2022

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Young-Jin Heo** received her B. S. degree in the Department of IT Engineering from Sookmyung Women's University, Republic of Korea in 2019, the M. S. degree from Sookmyung Women's University in 2021. In September 2021, she joined the Computer Vision Team, Samsung Electronics Co., Republic of Korea.

Her research interests include Generative Adversarial Network (GAN) techniques, Deepfake detection, and machine learning algorithms for computer vision.



**Woon-Ha Yeo** received the B.S. degree in the Department of IT Engineering from Sookmyung Women's University, Republic of Korea in 2019, and the M.S. degree from Sookmyung Women's University in 2021. She is a Ph.D. candidate in the Department of Convergence Science at Sahmyook University, Republic of Korea.

Her research interests include image and video processing, intelligent video coding algorithm, and terahertz imaging super-resolution.



**Byung-Gyu Kim** (Senior Member, IEEE) received the B.S. degree from Pusan National University, South Korea, in 1996, the M.S. degree from the Korea Advanced Institute of Science and Technology (KAIST), in 1998, and the Ph.D. degree from the Department of Electrical Engineering and Computer Science, KAIST, in 2004.

In March 2004, he joined the Real-Time Multimedia Research Team, Electronics and Telecommunications

Research Institute (ETRI), South Korea, where he was a Senior Researcher. In ETRI, he developed so many real-time video signal processing algorithms and patents and received the Best Paper Award, in 2007. From February 2009 to February 2016, he was an Associate Professor with the Division of Computer Science and Engineering, Sun Moon University, South Korea. In March 2016, he joined the Department of Information Technology (IT) Engineering, Sookmyung Women's University, South Korea, where he is currently a Full Professor. He has published over 250 international journal articles and conference papers, patents in his field. His research interests include image and video signal processing for the content-based image coding, video coding techniques, 3D video signal processing, deep/reinforcement learning algorithm, embedded multimedia systems, and intelligent information system for image signal processing.

Dr. Kim is a Professional Member of ACM and IEICE. He also served or serves on Organizing Committee of CSIP 2011, a Co-Organizer of CICCAT2016/2017, The Seventh International Conference on Advanced Computing, Networking, and Informatics (ICACNI2019), the EAI 13-th International Conference on Wireless Internet Communications Conference (WiCON 2020), and the Program Committee Members of many international conferences. He has received the Special Merit Award for Outstanding Paper from the IEEE Consumer Electronics Society, at IEEE ICCE 2012, the Certification Appreciation Award from the SPIE Optical Engineering, in 2013, and the Best Academic Award from the CIS, in 2014. He has been honored as an IEEE Senior Member, in 2015. He has also received the Excellent Paper Award from the IEEE Consumer Electronics Society, at IEEE ICCE 2021.

He has been serving as a Professional Reviewer in many academic journals, including IEEE, ACM, Elsevier, Springer, Oxford, SPIE, IET, MDPI, IT&T, and so on. In 2007, he has served as an Editorial Board Member for the *International Journal of Soft Computing*, *Recent Patents on Signal Processing*, the *Research Journal of Information Technology*, the *Journal of Convergence Information Technology*, and the *Journal of Engineering and Applied Sciences*. He has been serving as an Associate Editor for *Circuits, Systems and Signal Processing* (Springer), *The Journal of Supercomputing* (Springer), *The Journal of Real-Time Image Processing* (Springer), *Heliyon-Computer Science* (Cell press), and *Applied Sciences* (MDPI). Since March 2018, he has been serving as the Editor-in-Chief for *The Journal of Multimedia Information System* and an Associate Editor for IEEE ACCESS Journal. He is serving as a Topic Editor for *Sensors and Electronics* (MDPI).