Received August 15, 2021, accepted August 28, 2021, date of publication September 8, 2021, date of current version September 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3110859

Exposing Fake Faces Through Deep Neural Networks Combining Content and Trace Feature Extractors

EUNJI KIM[1] AND SUNGZOON CHO[2,3]

[1]School of Business Administration, Chung-Ang University, Seoul, Dongjak-gu 06974, Republic of Korea
[2]Department of Industrial Engineering, Seoul National University, Seoul, Gwanak-gu 08826, Republic of Korea
[3]Institute for Industrial Systems Innovation, Seoul National University, Seoul, Gwanak-gu 08826, Republic of Korea

Corresponding author: Sungzoon Cho (zoon@snu.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) funded by the Korean Government (MSIT) under Grant 2021R1G1A1093263.

ABSTRACT With the breakthrough of computer vision and deep learning, there has been a surge of realistic-looking fake face media manipulated by AI such as DeepFake or Face2Face that manipulate facial identities or expressions. The fake faces were mostly created for fun, but abuse has caused social unrest. For example, some celebrities have become victims of fake pornography made by DeepFake. There are also growing concerns about fake political speech videos created by Face2Face. To maintain individual privacy as well as social, political, and international security, it is imperative to develop models that detect fake faces in media. Previous research can be divided into general-purpose image forensics and face image forensics. While the former has been studied for several decades and focuses on extracting hand-crafted features of traces left in the image after manipulation, the latter is based on convolutional neural networks mainly inspired by object detection models specialized to extract images' content features. This paper proposes a hybrid face forensics framework based on a convolutional neural network combining the two forensics approaches to enhance the manipulation detection performance. To validate the proposed framework, we used a public Face2Face dataset and a custom DeepFake dataset collected on our own. Experimental results using the two datasets showed that the proposed model is more accurate and robust at various video compression rates compared to the previous methods. Throughout class activation map visualization, the proposed framework provided information on which face parts are considered important and revealed the tempering traces invisible to naked eyes.

INDEX TERMS Convolutional neural networks, DeepFake, Face2Face, fake face detection, fake face image forensics, multi-channel constrained convolution, transfer learning.

I. INTRODUCTION

Given advances in computer vision and deep learning, fake face media aiming at impersonating target subjects has surged. In June 2019, for example, Mark Zuckerberg became the newest victim of AI-manipulated media [1]. The forged media may be uploaded on social media to propagate fake information, which may have serious moral, ethical, and legal implications. For example, DeepFake has been abused to make fake pornography by putting a victim's face on a naked body, which then spreads on the Internet. This raises significant social issues and concerns. The victims can be celebrities, politicians, or even ordinary people. Another example would be political abuse to threaten international security or interfere in elections. With the Face2Face algorithm and a commodity webcam, anyone can manipulate politicians' speeches by changing their facial expressions in videos. In other applications like biometrics, fake faces may fool an access control system [2], [3]. To protect privacy and security, fake face detection is a very imperative task in image forensics.

Anyone can generate realistic-looking fake faces by utilizing state-of-the-art face manipulation tools such as DeepFake [4] or Face2Face [5]. Fig. 1 shows the examples of genuine and fake faces, which are almost indistinguishable with the bare eye. DeepFake replaces a person's face in the

The associate editor coordinating the review of this manuscript and approving it for publication was Junchi Yan.

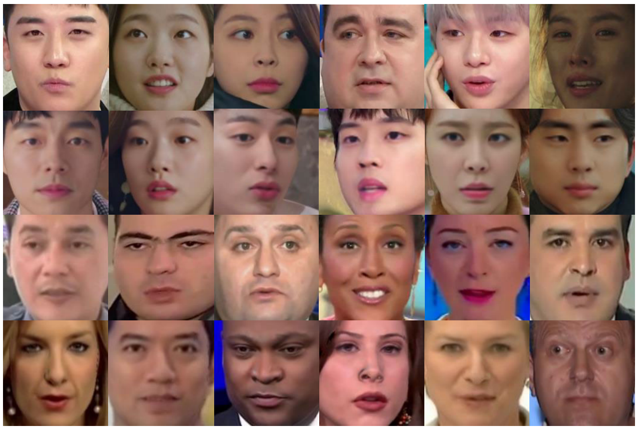


FIGURE 1. Examples of genuine and fake facial images. The second and fourth rows are fake faces manipulated by DeepFake and Face2Face, respectively. Faces on the first and third rows are genuine.

original media with the target person's likeness while maintaining the original facial expression. Face2Face transfers the expression of a source media to a target media while maintaining the original facial identity. Both tools use deep learning and computer vision techniques to generate an image containing the desired manipulation result. The manipulated face is pasted and rendered onto the original image. Then, seamless blending [6] is applied, which adjusts image characteristics such as color, contrast, and brightness to make a realistic manipulation result.

As the need for fake face detection arises, facial image forensics research has been actively conducted in recent years. Most face forensics approaches leverage deep learning, especially convolutional neural networks (CNNs) originally architected for the object recognition task such as AlexNet [7], VGG19 [8], ResNet [9], and Xception [10]. These models have a number of stacked layers for hierarchically extracting content features from an image: it first extract low-level content features (e.g., edge, mesh patterns, text) and mixes them to detect and classify object-specific features (e.g., dog faces, birds' legs) [11]. So, for fake detection, the whole layers except the classification part are borrowed for transfer learning. Among the object detection CNNs, Xception [10] showed the most powerful performance [12]–[14].

Classical general-purpose image forensics focus on detecting manipulation traces left in the images after tempering operations, regardless of what content is contained in the image. The tempering operations includes resampling and resizing [15]–[19], median filtering [20]–[23], and contrast enhancement [24]–[26]. Thus, the forensic features are designed to suppress an image's contents and to describe a large number of different dependencies among neighboring pixels. For feature extraction, hand-crafted [27] or model-driven [28], [29] approaches are incorporated with machine learning classifiers [30]. Recently, [31] proposed an end-to-end CNN-based solution including constrained convolution able to jointly suppress an image's content and then extract forensic features.

In this paper, we propose a face forensics model that mashes up the conventional image forensic approach and the fake face image forensic approach. The proposed model is a type of convolutional neural networks containing two types of feature extractors to simultaneously extract content features and trace features from a face image. The former feature extractor is trained by transferring and fine-tuning the feature extractor of a pre-trained object recognition model. Thus, the extracted features are specialized to represent various contents in a face. The latter feature extractor is based on the local relationship between neighboring pixels, by first applying the multi-channel constrained convolution — an extended version of the single-channel constrained convolution proposed in [31] — to the input image to obtain the content-excluded image and extract the features hierarchically. When the content is excluded, the color and contrast of the original image disappear, leaving only the outlines and some traces (as in Fig. 3). To verify the fake face detection performance of the proposed model, we conducted experiments with facial image datasets manipulated by DeepFake and Face2Face algorithms. As a result, the proposed model showed higher detection accuracy and robustness at various video compression levels than the existing baseline models. Furthermore, to better understand the proposed model, we visualized the hierarchical feature representation extracted from content-independent and content-dependent feature extractors. We also visualized class activation maps (CAMs) [32] to find the visual clues that the two feature extractors rely on to detect a fake face.

The remainder of this paper is organized as follows: Section II presents related work, and Section III introduces the proposed model. In Section IV, the experimental results are presented and discussed. Finally, Section V concludes the paper.

II. RELATED WORK

A. REVIEW OF FACE MANIPULATION

Facial images reveal facial expressions and identities, and its manipulation techniques can be divided into two types: facial expression manipulation and facial identity conversion. Facial expression manipulation studies started with lip motion synthesizing. The first lib motion synthesizing work is presented by [33] to automatically create a video of a person with generated mouth movements. [34] uses high-quality 3D face capturing techniques to alter the mouth motions in a video so that it matches the new audio track of a dubber. [35] demonstrated the first real-time expression transfer for facial reenactment by tracking the source and target faces captured by an RGB-D camera. Furthermore, they reconstructed a 3D model to apply the source facial expression to the target. [5] also proposed an advanced real-time facial reenactment system, called Face2Face, which is capable of altering facial movements in video streams (e.g., YouTube). Face2Face tracks facial expressions of both the source and target videos to reenact the mouth interior of the target video that best matches the expression of the source by warping and produce an accurate fit. The synthesized target face is convincingly re-rendered on top of the corresponding video stream so that it seamlessly blends with the real-world illumination. With the advance of deep learning, [36] proposed a technique that feeds audio input to a recurrent neural network to generate a mouth texture sequence and synthesize a photo-realistic lip-synced video of Obama.

Facial identity conversion techniques have been proposed as far back as 2004 by [37] with fully automatic techniques described a decade ago in [38]. These methods were originally offered in response to privacy preservation concerns by obfuscating the identities of subjects in images where the original face cannot be swapped into the designated target identity. Since then, many applications seem to come from recreation or entertainment [39]–[41]. [42] presented one of the first automatic face swap methods, which replace the source face with the target face by cropping and pasting. [43] presented a similar but more advanced system that replaces the face while preserving the original expressions. After replacing the face, Poisson image editing [6] blends facial tone, illumination, and tempering region boundary. Recently, the remarkable development of AI and deep learning has enhanced face manipulation quality. [44] used CNNs by framing the identity swapping problem in terms of style transfer [45]. DeepFake [4], open-source identity-swapping software based on convolutional autoencoders, was released in December 2017, lowering the hurdles of face manipulation. By using deep convolutional autoencoder, the identity of the original face is swapped to a specific target person designated by the user while preserving the facial expression. DeepFake splices the identity-converted face image onto the convex hull region made of facial landmarks in the original image. Poisson image editing [6] adjusts the boundary, shape, and illumination of the output image to make the tampered result look more realistic.

In this paper, we used two AI-manipulated face image datasets for experiment: one generated by DeepFake [4] that converts facial identity in the original image, and the other generated by Face2Face [5] that manipulates facial expressions.

B. REVIEW OF IMAGE FORENSICS

Here, we review general-purpose image forensics and facial image forensics. General-purpose image forensics aims to ensure the authenticity of an image by detecting the traces of tempering operations such as resampling and resizing [15]–[19], median filtering [20]–[23], contrast enhancement [24]–[26], multiple jpeg compression [46]–[49]. When an image is manipulated by using an image tempering operation, statistical or physical traces remain, even if they are not visible to the naked eye. Thus, the general-purpose forensic methods focus on detecting manipulation traces left in the image after tempering operations, regardless of what content is contained in the image. Thus, the forensic features are designed to suppress an image's contents and to describe a large number of different dependencies among neighboring pixels. Early methods are driven by handcrafted features that capture statistical, physical, or sensor defects that occur during tempering operation [27]. The most popular feature extraction method is the spatial rich model (SRM) [28], [29] originally proposed to extract steganalytic features and successfully performed the general-purpose image manipulation detection [30]. Recently, several CNN-based solutions have been proposed in general-purpose image forensics [31], [50]–[52]. Among them, [31] proposed a constrained convolution able to jointly suppress an image's content. This can be viewed as an adaptive generalization of previous hand-designed feature extraction methods from grayscale images. Their MISLNet architecture includes the constrained convolution, adaptively learn manipulation trace features by back-propagation, and can perform as well as, or slightly better than, the SRM by learning the feature extractor better than human-designed SRMs in general-purpose image forensics.

Recently, facial image forensics specialized in facial media have emerged due to the recent advances in AI-driven face manipulation techniques as described in the previous subsection. The goal of the facial image forensics is to identify the authenticity of a face in a given image. The dominant approach is to utilize CNNs architected for object recognition, such as AlexNet [7], VGG19 [8], Inception [53], and ResNet [9]. Unlike the general-purpose forensics approaches that focus on detecting manipulation traces left in the image, the object recognition CNNs hierarchically extract content features from an image: it first extracts low-level content features (e.g., edge, mesh patterns, text) and mixes them to detect object-specific features (e.g., dog faces, birds' legs) [11]. Initially, [54] proposed a model combining VGG19 and AlexNet to detect morphed face images from print-scanned images that are generated by combining two different images from two subjects. [13] conducted a comparative experiment of several object recognition CNNs using their own dataset called 'Fake Faces in the Wild'. [12] also conducted a comparison study by creating a dataset named 'FaceForensics' containing fake face images manipulated by Face2Face. [14] further developed the dataset by adding additional images manipulated by DeepFake, FaceSwap, and NeuralTextures. Custom CNN architectures are also proposed for specific goals. For compactness, [55] proposed two compact CNN architectures, Meso-4 and MesoInception-4, borrowing the structure of Inception and evaluated them on their own DeepFake dataset and [12]'s Face2Face dataset. For capturing local noise residuals and camera characteristics, [56] combined a GoogLeNet [53] and a patch-based triplet network. To cope with the imbalanced dataset, [57] proposed a CNN called MANFA designed for grayscale image and integrated it with XGBoost and AdaBoost.

III. PROPOSED METHOD

A. OVERALL FRAMEWORK

The proposed fake face detection framework consists of face detection, face alignment and extraction, and authenticity

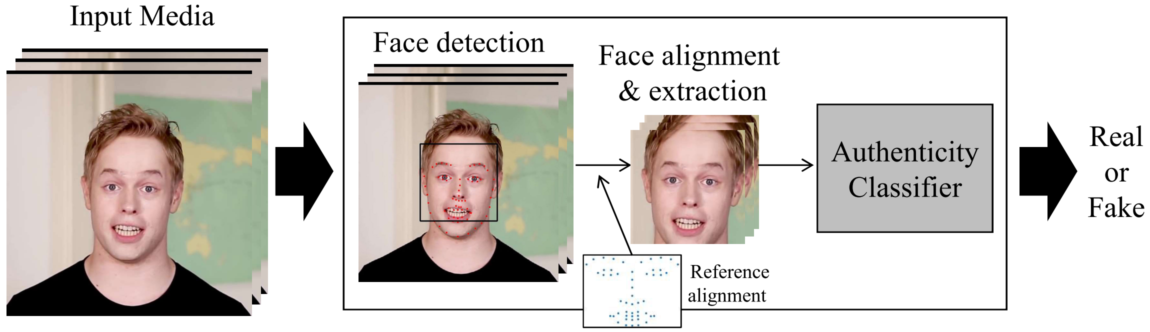


FIGURE 2. The proposed fake face detection framework.

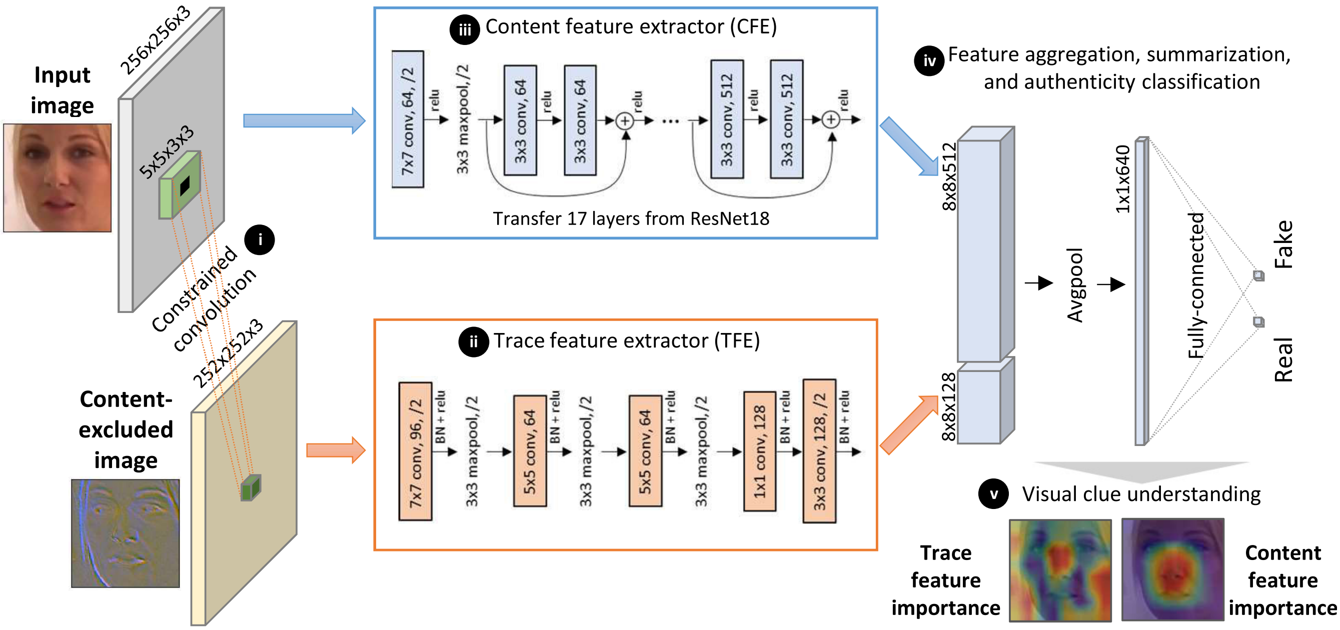


FIGURE 3. The proposed face authenticity classifier combines content feature extractor (CFE) and trace feature extractor (TFE). A convolution is depicted by a square containing its detail in the two feature extractors. For example, the first convolution in the CFE has 7 × 7 convolutional filter with stride 2 and outputs 64 feature maps.

classification. The overall framework is depicted in Fig. 2. Given an input image, the facial region is detected by using a state-of-the-art face detection model. Specifically, we adopted a neural facial landmark detection model [58] that automatically localizes the 68 fiducial facial landmark points around facial components and facial contours such as eyes, mouth, and chin. Among those points, we use only 51 points excluding 17 points from chin because facial manipulation is performed inside the inner facial region. Then, we align the face to fit the reference alignment because faces appearing in media are rarely frontal or unrotated. We apply the affine transformation on the image by finding the one-to-one mapping from the extracted landmark points to the reference alignment points. Through affine transformation, rotated or profile faces can be aligned according to the reference alignment, which helps to enhance the fake face detection performance. Finally, we crop the facial region from the image and feed it to the facial authenticity classifier. The width and height of the input facial image are set to 256. We describe the authenticity classifier in the next subsection (Section III-B).

B. FACE AUTHENTICITY CLASSIFIER

The proposed CNN-based face authenticity classifier consists of (i) content exclusion using multi-channel constrained convolution, (ii) content feature extractor (CFE), (iii) trace feature extractor (TFE), (iv) feature aggregation, summarization, and authenticity classification, and (v) visual clue understanding as depicted in Fig. 3. The details are presented in the following subsections.

1) CONTENT EXCLUSION USING MULTI-CHANNEL CONSTRAINED CONVOLUTION

We apply the multi-channel constrained convolution to the input image to obtain the content-excluded image that will be fed to the TFE. In the content-excluded image as shown in Fig. 3, information such as color, shades, or contrast that describe the facial content disappears. This leaves only trace information such as contour or minor texture differences.

The constrained convolution is originally proposed in [31]. However, it is designed for single-channel grayscale input images and can only be applied to them. Thus, we extended the single-channel constrained convolution to the multi-channel constrained convolution with the following constraints for colored input images.

$$\begin{cases} \boldsymbol{w}_k(0,0,c) = -\dfrac{1}{3} & \text{for } c = 0, 1, 2, \\ \sum_{m\neq 0, n\neq 0, c} \boldsymbol{w}_k(m,n,c) = 1, \end{cases} \tag{1}$$

where $\boldsymbol{w}$ is the convolutional filter, $\boldsymbol{w}_k(0,0,c)$ is the center of the filter weight for the RGB channels, and c and k are the input and output channel indices. A single-channel constrained convolution was applied to the grayscale image ($c = 0$), but we extended it to the color image ($c = 0,\ 1,\ 2$). As depicted in the input image part of Fig. 3, the central value of the constrained convolutional filter (black) is $-1/3$ for each channel, while the sum of the filter values in the remaining part (green) is 1.

Similar to the single-channel constrained convolution, the multi-channel version of the constrained convolution excludes content information in images. To see this, we define a new convolutional filter $\tilde{\boldsymbol{w}}_k$ as

$$\tilde{\boldsymbol{w}}_k(m,n,c) = \begin{cases} \boldsymbol{w}_k(m,n,c), & \text{if } (m,n) \neq (0,0), \\ 0, & \text{if } (m,n) = (0,0). \end{cases} \tag{2}$$

The center value of the filter $\tilde{\boldsymbol{w}}_k$ is 0 and the remaining values are the same as the filter $\boldsymbol{w}_k$ for all channels. We also define a new impulse filter $\boldsymbol{\delta}$ as

$$\boldsymbol{\delta}(m,n,c) = \begin{cases} 0, & \text{if } (m,n) \neq (0,0), \\ \dfrac{1}{3}, & \text{if } (m,n) = (0,0), \end{cases} \tag{3}$$

for input RGB channels $c = 0,\ 1,\ 2$. $\boldsymbol{\delta}$ represents an impulse filter with central values $1/3$ for all input channels while it is 0 elsewhere. Thus, when applying this filter to the input image, the resulting feature map is a grayscale version of the original image. Using the two newly defined filters, the multi-channel convolution can be expressed as $\boldsymbol{w}_k = \tilde{\boldsymbol{w}}_k - \boldsymbol{\delta}$. For an input image $\boldsymbol{I}$, we can represent the k-th feature map of the k-th constrained convolution filter, $\boldsymbol{h}_k$, as

$$\boldsymbol{h}_k = \boldsymbol{I} * \boldsymbol{w}_k \tag{4}$$

$$= \boldsymbol{I} * (\tilde{\boldsymbol{w}}_k - \boldsymbol{\delta}) \tag{5}$$

$$= \boldsymbol{I} * \tilde{\boldsymbol{w}}_k - \boldsymbol{I} * \boldsymbol{\delta} \tag{6}$$

$$= \boldsymbol{I} * \tilde{\boldsymbol{w}}_k - \frac{1}{3}\sum_c \boldsymbol{I}(:,:,c). \tag{7}$$

The latter part in (7) can be viewed as the grayscale version of the original image. Thus, it can be seen that the feature map generated from colored input image through the multi-channel constrained convolution can be expressed as a residual as the single-channel constrained convolution as in [31]; the feature map created by the new convolution minus the grayscale image of the original input.

C. CONTENT FEATURE EXTRACTOR

From input image, the CFE extracts content features for authenticity classification. The CFE is transferred from a ResNet-18, which is one of the residual network architectures in [9] and pre-trained for object recognition. Our CFE is fine-tuned so that it can extract features describing contents in a face image such as facial tones, eyes, and wrinkles. In this model, we used 17 layers before the last fully-connected layer of the ResNet-18. Therefore, the CFE consists of a 7×7 convolutional layer with ReLU activation and max-pooling, followed by 8 residual blocks. As depicted in Fig. 3, the residual block output is the sum of the output from the block and the input from the "shortcut connection". The residual structure are easier to optimize due to the shortcut connections and can gain accuracy from considerably increased depth.

D. TRACE FEATURE EXTRACTOR

From the content-excluded image obtained by applying multi-channel constrained convolution in Section III-B1 to the input image, the CFE extracts content features for authenticity classification. The TFE is designed to be simple than the CFE described in the previous subsection (Section III-C) since the content-excluded image contains less information than the original image. We borrowed the first to forth convolutional layers of the TFE in Fig. 3 from the middle part of the architecture proposed in [31]. The last convolution contains 128 filters of size 3 × 3 and stride of 2. This is designed to output 128 feature maps of size 8 × 8, which is the same size as the last feature maps of CFE.

E. FEATURE AGGREGATION, SUMMARIZATION, AND AUTHENTICITY CLASSIFICATION

Here, the content features and trace features extracted from CFE and TFE, respectively, are concatenated along the channel axis. After 8 × 8 average pooling, the 640-dimensional feature vector is obtained and fed to the fully-connected layer. We apply softmax to the output of fully-connected layer to classify the authenticity of the input face image.

F. VISUAL CLUE UNDERSTANDING

To understand the visual clues, the model relies on to detect a fake face, we utilized CAMs [32]. Through CAM, it is possible to visualize which part of the input image is used to determine authenticity. Global average pooling in the model outputs the spatial average of the feature maps at the last convolutional layer. A weighted sum of these values is used to determine the authenticity of the input image. The CAM score is obtained by the weighted sum of the feature maps of the last convolutional layer before global average pooling. By visualizing CAM, we can understand which part of the input image is important for determining authenticity. We can investigate the important locations of the content and trace features and can deeply understand the face image manipulation techniques. For example, the input in Fig. 3 is manipulated by face2face that reenacts the facial expression. From the provided visual clues, we can understand that the model used the content features from the mouth interior region and the trace features from the central and the outer area of the face for the authenticity classification.

IV. EXPERIMENTAL RESULTS

We first provide dataset descriptions. Then, we present the baselines, the training details, and evaluation metrics.

A. DATASETS

For the experiments, we used two fake face datasets that were created by using Face2Face [5] and DeepFake [4] techniques. The details for each dataset are as follows.

1) Face2Face

The FaceForensics dataset [12] consists of 1,004 pairs of fake and original videos where the fake video was generated by the Face2Face [5] approach. The data was collected from YouTube with a minimum resolution criteria larger than 480p. To extract video sequences that contain a face for more than 300 consecutive frames, the Viola-Jones face detector [59] was used. The Face2Face technique re-renders the face in the target video with possibly different expressions from the source video. This dataset is already split into 704 pairs for training, 150 pairs for validation, and 150 pairs for testing. As in [12], we extracted 10 images from each video.

2) DeepFake

We created our own DeepFake dataset as no such dataset existed at the time of the experiment. First, we collected videos of 67 identities from various platforms (e.g., YouTube, TV shows, and movies) that satisfy a minimum resolution of 480p and a minimum facial region size of 300 × 300. We trained convolutional autoencoder models that swap two identities by using the code from [4]. To obtain realistic-looking fake faces, we selected identity pairs that have the same gender, facial tone, and lineament. By using a GTX 1080 Ti GPU, it took a few days to train a model to achieve realistic swapping results. Forged videos were created by swapping the original face in a video with the output face from the autoencoder model. Consequently, 47 realistically-forged videos were manually chosen among the outputs. All faces have been extracted and aligned by using a pre-trained facial landmark detection neural network [58]. It has been manually reviewed to remove the wrong face detection results. The dataset is split into 80 videos for training, 17 videos for validation, and 17 videos for testing with stratified sampling. We extracted 100 facial images per video.

B. BASELINE MODELS

The baseline models for performance comparison are (i) SRM+SVM [29], (ii) Meso-4 [55], (iii) MesoInception-4 [55], and (iv) M-MISLNet, which replace the single-channel constrained convolution in the original MISLNet proposed in [31] by the multi-channel constrained convolution proposed in this paper in addition to (iv) ResNet18 [9]. We briefly describe all the approaches.

1) SRM+SVM

The hand-crafted 5,514 steganalytical features are obtained by a color rich model [29] that is aware of the underlying spatial alignment of the color filter array. Traces of the dependencies between color channels and among adjacent pixels are summarized by high-order statistics in the form of 3D co-occurrences. These features are then used to train a linear support vector machine (SVM).

2) MESO-4

This architecture is proposed in [55] and consists of four consecutive blocks of convolutional and pooling layers followed by a fully-connected layer.

3) MesoInception-4

This architecture is also proposed in [55]. The first and second blocks in the Meso-4 are replaced by inception modules proposed in [60].

4) M-MISLNet

The original MISLNet is proposed in [31] and consists of one single-channel constrained convolutional layer, four consecutive blocks of convolution and pooling, and then three fully-connected layers. We replaced the single-channel constrained convolution in this model to the multi-channel constrained convolution proposed in this paper for color input images.

5) ResNet18

ResNet-18 proposed in [9] is pre-trained by the ImageNet dataset [61] for object recognition. We transferred the first 17 layers of the network, added a fully-connected layer with randomly initialized parameters, and fine-tuned for our forensic task.

C. TRAINING DETAILS

We ran our experiments using an Intel Xeon(R) E5-2630 @2.20GHz CPU, a Nvidia GeForce GTX 1080 Ti GPU, and 128GB RAM. The weights optimization of the network was achieved with batches of 32 images of size 256 × 256 × 3 using Adam [62] optimizer with learning rate $1e-4$, $\beta_1 = 0.5$, and $\beta_2 = 0.999$. The loss function is set to minimize the binary cross-entropy error, $L(x, y; f) = -y \log f(x) - (1-y)\log(1-f(x))$, where $x \in \mathbb{R}^{256\times256\times3}$ and $y \in \{0, 1\}$ are the input image and its corresponding label, respectively, and $f(\cdot) : \mathbb{R}^{256\times256\times3} \rightarrow \{0, 1\}$ is the network. We trained each network until the validation does not decrease in 5 consecutive epochs or the epoch becomes 100.

D. PERFORMANCE EVALUATION

We utilized accuracy as the performance measure. The accuracy is measured as a ratio of correctly predicted observations to total observations, which is the most intuitive performance measure. The robustness of a model should

TABLE 1. Fake face detection accuracy and F1 score.

	Dataset	DeepFake			Face2Face		
	Compression level	0	23	40	0	23	40
Accuracy	SRM+SVM [29]	96.56	77.02	65.56	98.80	71.59	59.15
	Meso-4 [55]	99.07	95.77	76.31	98.81	88.18	76.17
	MesoInception-4 [55]	96.81	93.28	71.86	96.30	84.89	75.95
	M-MISLNet [31]	99.76	96.79	76.82	99.26	92.38	80.91
	ResNet18 [9]	99.95	96.11	73.83	99.42	96.45	84.83
	Proposed	**99.96**	**97.14**	**77.94**	**99.46**	**97.49**	**85.80**
F1 score	SRM+SVM [29]	93.27	77.34	56.23	99.80	71.37	57.08
	Meso-4 [55]	98.69	94.58	72.64	98.80	87.63	75.60
	MesoInception-4 [55]	95.94	91.25	68.36	96.27	85.27	77.13
	M-MISLNet [31]	99.70	95.78	68.21	99.26	92.38	80.77
	ResNet18 [9]	99.94	94.88	66.64	**99.42**	96.44	84.84
	Proposed	**99.99**	**96.64**	**73.26**	99.41	**97.22**	**85.93**

also be measured to account for practical applications of the model. When images and videos are uploaded to social networks or the Internet, compression and resizing are routinely carried out, which laundry manipulation traces from the data [12]. To evaluated the robustness with different video compression levels, the Face2Face and DeepFake datasets were compressed using the H.264 codec. We chose the same compression rate with H.264 codec as in [12]: 0 (lossless compression), 23 (light compression), and 40 (strong compression). The compression examples are represented in Fig. 4.

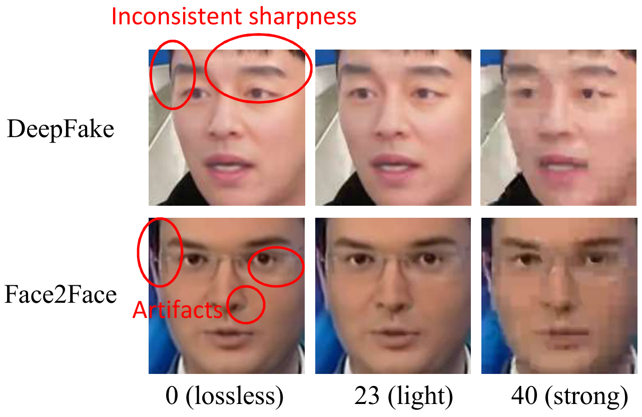


FIGURE 4. Video compression results of forged media using H.264 codec. DeepFake shows inconsistent sharpness along the borderline of the facial region whereas Face2Face contains some artifacts. The compression laundries the manipulation traces from the data.

TABLE 2. Elapsed time for training (unit: min).

Model	DeepFake	Face2Face
SRM+SVM [29]	1,421.30	2,498.43
Meso-4 [55]	16.46	58.21
MesoInception-4 [55]	18.47	69.97
M-MISLNet [31]	7.56	26.75
ResNet18 [9]	15.42	42.40
Proposed	16.83	38.22

E. EXPERIMENTAL RESULTS

Table 1 summarized the fake face detection accuracy and F1 score, respectively. We compared the proposed and the baseline models on DeepFake and Face2Face datasets. We trained 10 models per dataset and compression level and averaged their accuracy. Since the baseline models has no F1 scores reported by the original papers, we also trained and evaluated them together. The highest accuracy and F1 score for each data set and compression level is shown in bold. The proposed model showed the highest accuracy and F1 score at all compression levels, except one case, Face2Face dataset with lossless compression. In this case, however, the best performance model, ResNet18, shows a very small F1 score difference of 0.01. In the lossless compression setting, all models showed very high accuracy and F1 score. As the compression ratio increases, however, the accuracy and F1 score decrease, particularly for manually designed features (SRM+SVM). Comparing DeepFake and Face2Face cases, DeepFake's accuracy and F1 score are greatly reduced. This is because, as the compression level increases, the boundary part of the swapping region is blended, which makes it harder to be detected. Comparing M-MISLNet and ResNet18, M-MISLNet performed well in the DeepFake dataset and ResNet18 performed well in the Face2Face dataset.

We summarized the elapsed time for training in Table 2. The SRM+SVM method took much longer training time than other CNN-based models that took tens of minutes to train. In the SRM+SVM method, the SVM took about ten minutes. However, the feature extraction time reaches an average of 7.4 seconds per image. As a result, it took about 24 hours for DeepFake and 42 hours for Face2Face to extract features of all images. Thus, hand-designed feature extraction methods such as [28], [29] seems to be difficult to apply to a large amount of data, which will benefit from the adaptive feature extraction method.

Fig. 5 are visualizations of the hierarchical feature representation of (a) M-MISLNet, (b) ResNet18, (c) CFE of the proposed model, and (d) TFE of the proposed model. Feature maps created by sequentially applying convolution are presented from left to right. For each feature map, the most activated 4, 9, 16, 25, and 36 channels were selected and visu-

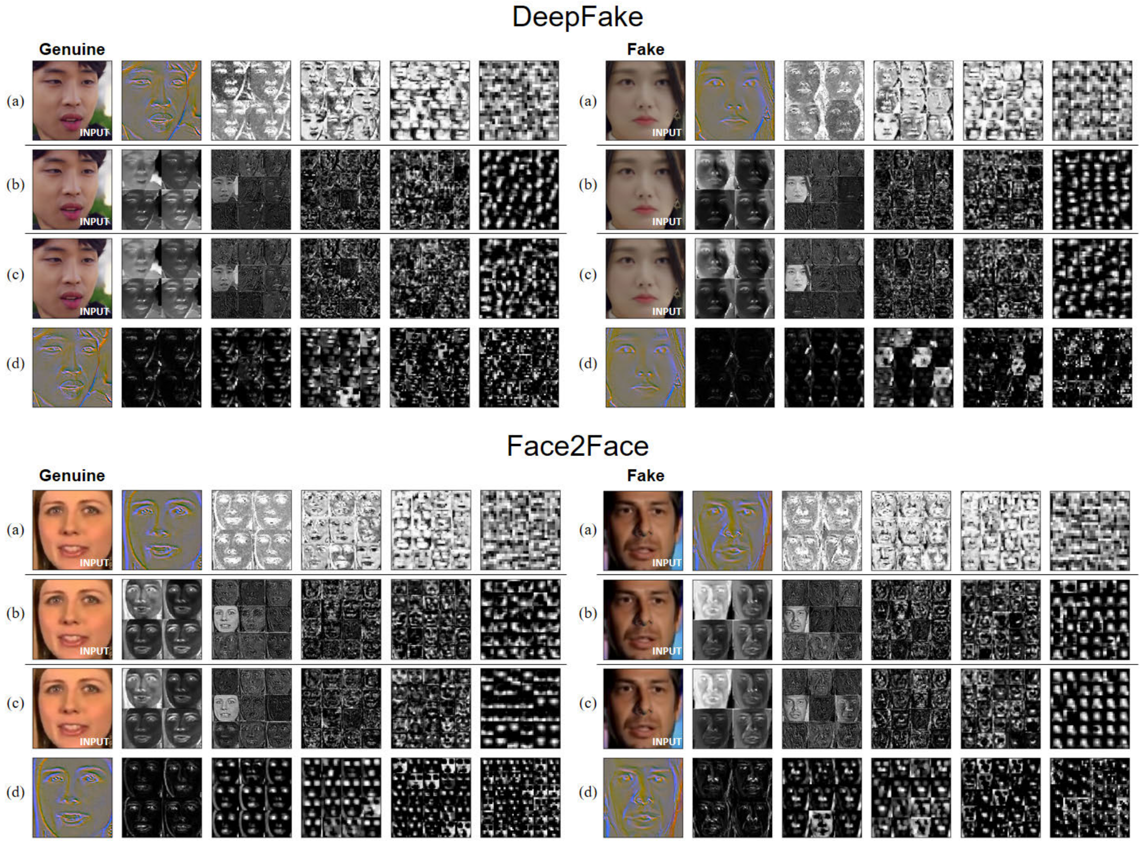


FIGURE 5. Hierarchical feature representation of DeepFake and Face2Face data. (a) M-MISLNet, (b) ResNet18, (c) CFE of the proposed model, and (d) TFE of the proposed model. The right part is for genuine, and the left part is for fake input images. The CFE and TFE extract features similar to ResNet18 and M-MISLNet, respectively.

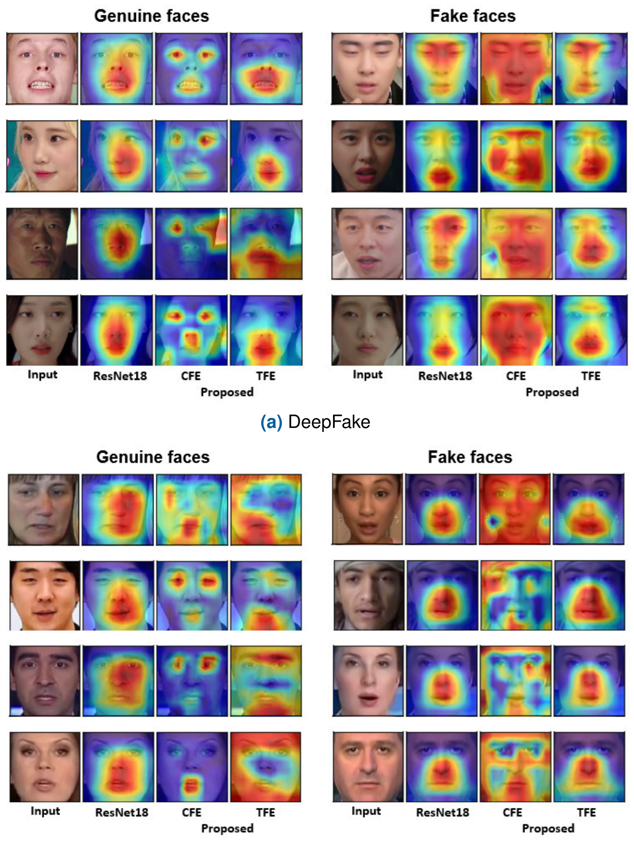
alized (the size of feature maps were adjusted). M-MISLNet applies multi-channel constrained convolution on the input image to get the content-excluded image and extract the features sequentially. Note that the content-excluded image has no color, texture, and contrast information, leaving only the outlines and some traces. ResNet18 is specialized in extracting content features from a large amount of ImageNet database. While M-MISLNet extracts features focused on the shape of eyebrows, lips, and outlines, ResNet18 extracts texture, contrast, and lighting information.

The proposed model is a combination of the aforementioned two methods. In Fig. 5, (c) and (d) are the feature maps made by CFE and TFE, respectively. Comparing ResNet18 and CFE, the feature map of the lower layer close to the input image is very similar to each other. On the other hand, the feature map of the higher layer is slightly different from each other. Nevertheless, the overall feature extraction results are very similar. The content-excluded image placed under the input image of the proposed model is extracted by constrained convolution as in M-MISLNet and is very similar to M-MISLNet. However, the two hierarchical feature map representations after the content-excluded images show totally different appearances.

The TFE in the proposed model extracts the contour features that describe the eyebrows, lips, and facial lines. In contrast, the features extracted by M-MISLNet represent facial area features in addition to them. This is because the content features are already extracted by a pre-trained CFE and the TFE focuses on the difference between neighboring pixels and extracts outlines or traces.

Fig. 6 is the CAMs [32] to understand the visual clues the model relies on to detect a fake face manipulated by DeepFake and Face2Face, respectively. Through CAM, it is possible to visualize which part of the input image is used to determine authenticity. In ResNet18 and the proposed model, global average pooling outputs the spatial average of the feature map of each unit at the last convolutional layer. A weighted sum of these values is used to generate the final output. CAM is obtained by a weighted sum of the feature maps of the last convolutional layer before global average pooling. We visualized ResNet18 and the proposed model in a structure where CAM can be calculated.

Comparing ResNet with the proposed model, we can see that the CAM of ResNet is similar to that of CFE. The TFE locally extracts features from the eyes and mouth in real cases and globally extracts features from the overall facial region in fake cases.



(a) DeepFake

(b) Face2Face

FIGURE 6. Class activation map (CAM) visualizations. Through CAM, it can be seen which part of the input image is used to determine authenticity.

Furthermore, the visual clues for detecting manipulation are different between DeepFake and Face2Face. For a fake image manipulated by DeepFake (Fig. 6a), the features extracted from the entire facial region were important. For a fake image manipulated by Face2Face (Fig. 6b), the features extracted from around the mouth region were important. Since DeepFake is a method of forging facial identity, the entire face is manipulated. On the other hand, Face2Face mainly manipulates around the mouth region. Thus, the CAM visualization results are reasonable.

V. CONCLUSION

In this paper, we proposed a novel deep learning model to perform fake face media forensics. The model extracts both content features and trace features simultaneously to detect fake faces. We also proposed a multi-channel constrained convolution — a generalization of the single-channel constrained convolution in [31] — to obtain the content-excluded image from color input images. Trace features are extracted from the content-excluded image while content features are extracted from the input image by fine-tuning a pre-trained ResNet-18 model.

Experiments were conducted to verify the performance of the proposed model using two datasets manipulated by Face2Face and DeepFake. The proposed model showed the highest accuracy at various video compression levels when compared to the baseline models, confirming its robustness. We compared the input image with the content-excluded image obtained by the proposed multi-channel constrained convolution. Therefore, the visually identifiable information like facial tone, contrast, and skin texture disappeared after the constrained convolution. By visualizing the hierarchical feature map representation, the content features and the trace features show a totally different appearance. Furthermore, visualizing the CAM and comparing important parts used for authenticity classification confirms that the proposed model can learn the different characteristics of fake manipulation methods.

The contributions of this paper are as follows: (i) We proposed a new CNN architecture capable of detecting tampered face images by combining two types of feature extractors. The extracted content and trace features can help discriminate the authenticity of faces in media. (ii) We proposed a multi-channel constrained convolution for color input images, which is a generalized version of a single-channel constrained convolution proposed in [31] for grayscale image. (iii) We conducted experiments of the proposed model and baseline models with DeepFake and Face2Face datasets. (iv) Performance is analyzed on manipulated videos compressed at various quality levels to account for the typical processing encountered when the video is uploaded on the Internet. This is a very challenging situation since low-level manipulation traces can get lost after compression. (v) To understand how the model works, we visualized the hierarchical feature representations from the two feature extractors and the visual clues of face manipulation detection using CAM.

Further studies should examine models robust to video compression. This is because most models, including the proposed model, show a decrease in accuracy as the video compression rate increases. We can borrow recurrent neural networks to apply the proposed framework to video inputs. Furthermore, we expect the proposed multi-channel constrained convolution can be exploited for general-purpose image forensics with color images.

REFERENCES

[1] *Facebook Wants to Stay 'Neutral' on Deepfakes. Congress Might Force it to Act*. Accessed: Jun. 14, 2019. [Online]. Available: https://www.vox.com/future-perfect/2019/6/13/18677574/facebook-zuckerberg-deepfakes-congress-house-hearing

[2] A. K. Jain, A. Ross, and S. Pankanti, "Biometrics: A tool for information security," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 2, pp. 125–143, Jun. 2006.

[3] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 4–20, Jan. 2004.

[4] *DeepFake Github Repository*. Accessed: Jun. 14, 2019. [Online]. Available: https://github.com/deepfakes/faceswap

[5] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, "Face2Face: Real-time face capture and reenactment of RGB videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2387–2395.

[6] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 313–318, Jul. 2003.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[8] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[10] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.

[11] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2014, pp. 818–833.

[12] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics: A large-scale video dataset for forgery detection in human faces," 2018, *arXiv:1803.09179*. [Online]. Available: http://arxiv.org/abs/1803.09179

[13] A. Khodabakhsh, R. Ramachandra, K. Raja, P. Wasnik, and C. Busch, "Fake face detection methods: Can they be generalized?" in *Proc. Int. Conf. Biometrics Special Interest Group (BIOSIG)*, Sep. 2018, pp. 1–6.

[14] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, "FaceForensics++: Learning to detect manipulated facial images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1–11.

[15] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 758–767, Feb. 2005.

[16] M. Kirchner, "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *Proc. 10th ACM Workshop Multimedia Secur. (MM Sec)*, 2008, pp. 11–20.

[17] N. Dalgaard, C. Mosquera, and F. Perez-Gonzalez, "On the role of differentiation for resampling detection," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 1753–1756.

[18] X. Feng, I. J. Cox, and G. Doërr, "Normalized energy density-based forensic detection of resampled images," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 536–545, Jun. 2012.

[19] B. Mahdian and S. Saic, "Blind authentication using periodic properties of interpolation," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 3, pp. 529–538, Sep. 2008.

[20] M. Kirchner and J. Fridrich, "On detection of median filtering in digital images," *Proc. SPIE*, vol. 7541, Jan. 2010, Art. no. 754110.

[21] X. Kang, M. C. Stamm, A. Peng, and K. J. R. Liu, "Robust median filtering forensics using an autoregressive model," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 9, pp. 1456–1468, Sep. 2013.

[22] G. Cao, Y. Zhao, R. Ni, L. Yu, and H. Tian, "Forensic detection of median filtering in digital images," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2010, pp. 89–94.

[23] C. Chen and J. Ni, "Median filtering detection using edge based prediction matrix," in *Proc. Int. Work. Digital Watermarking*, 2011, pp. 361–375.

[24] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 492–506, Sep. 2010.

[25] H. Yao, S. Wang, and X. Zhang, "Detect piecewise linear contrast enhancement and estimate parameters using spectral analysis of image histogram," in *Proc. IET Int. Commun. Conf. Wireless Mobile Comput. (CCWMC)*, 2009, pp. 94–97.

[26] M. Stamm and K. J. R. Liu, "Blind forensics of contrast enhancement in digital images," in *Proc. 15th IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 3112–3115.

[27] R. Böhme and M. Kirchner, "Counter-forensics: Attacking image forensics," in *Digital Image Forensics*. New York, NY, USA: Springer, 2013, pp. 327–366.

[28] J. Fridrich and J. Kodovský, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.

[29] M. Goljan and J. Fridrich, "CFA-aware features for steganalysis of color images," *Proc. SPIE*, vol. 9409, Feb. 2015, Art. no. 94090V.

[30] X. Qiu, H. Li, W. Luo, and J. Huang, "A universal image forensic strategy based on steganalytic model," in *Proc. 2nd ACM Workshop Inf. Hiding Multimedia Secur. (IH MMSec)*, 2014, pp. 165–170.

[31] B. Bayar and M. C. Stamm, "Constrained convolutional neural networks: Image manipulation detection," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2691–2706, Nov. 2018.

[32] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.

[33] C. Bregler, M. Covell, and M. Slaney, "Video rewrite: Driving visual speech with audio," in *Proc. 24th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1997, pp. 353–360.

[34] P. Garrido, L. Valgaerts, H. Sarmadi, I. Steiner, K. Varanasi, P. Perez, and C. Theobalt, "VDub: Modifying face video of actors for plausible visual alignment to a dubbed audio track," in *Comput Graph Forum*, vol. 34. Chichester, U.K.: The Eurographs Association, 2015, pp. 193–204.

[35] J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt, "Real-time expression transfer for facial reenactment," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–183, Nov. 2015.

[36] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing Obama: Learning lip sync from audio," *ACM Trans. Graph.*, vol. 36, no. 4, p. 95, Jul. 2017.

[37] V. Blanz, K. Scherbaum, T. Vetter, and H.-P. Seidel, "Exchanging faces in images," in *Computer Graphics Forum*, vol. 23. 2004, pp. 669–676.

[38] D. Bitouk, N. Kumar, S. Dhillon, P. Belhumeur, and S. K. Nayar, "Face swapping: Automatically replacing faces in photographs," *ACM Trans. Graph.*, vol. 27, no. 3, p. 39, 2008.

[39] I. Kemelmacher-Shlizerman, "Transfiguring portraits," *ACM Trans. Graph.*, vol. 35, no. 4, p. 94, 2016.

[40] L. Wolf, Z. Freund, and S. Avidan, "An eye for an eye: A single camera gaze-replacement method," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 817–824.

[41] O. Alexander, M. Rogers, W. Lambeth, M. Chiang, and P. Debevec, "Creating a photoreal digital actor: The digital emily project," in *Proc. Conf. Vis. Media Prod.*, Nov. 2009, pp. 176–187.

[42] K. Dale, K. Sunkavalli, M. K. Johnson, D. Vlasic, W. Matusik, and H. Pfister, "Video face replacement," *ACM Trans. Graph.*, vol. 30, no. 6, p. 130, 2011.

[43] P. Garrido, L. Valgaerts, O. Rehmsen, T. Thormaehlen, P. Perez, and C. Theobalt, "Automatic face reenactment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4217–4224.

[44] I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3697–3705.

[45] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.

[46] T. Bianchi and A. Piva, "Detection of non-aligned double JPEG compression with estimation of primary compression parameters," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 1929–1932.

[47] T. Bianchi and A. Piva, "Image forgery localization via block-grained analysis of JPEG artifacts," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1003–1017, Jun. 2012.

[48] R. Neelamani, R. de Queiroz, Z. Fan, S. Dash, and R. G. Baraniuk, "JPEG compression history estimation for color images," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1365–1378, Jun. 2006.

[49] Z. Qu, W. Luo, and J. Huang, "A convolutive mixing model for shifted double JPEG compression with application to passive image authentication," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2008, pp. 1661–1664.

[50] D. Cozzolino, G. Poggi, and L. Verdoliva, "Recasting residual-based local descriptors as convolutional neural networks: An application to image forgery detection," in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur.*, Jun. 2017, pp. 159–164.

[51] J.-Y. Sun, S.-W. Kim, S.-W. Lee, and S.-J. Ko, "A novel contrast enhancement forensics based on convolutional neural networks," *Signal Process., Image Commun.*, vol. 63, pp. 149–160, Apr. 2018.

[52] M. Huh, A. Liu, A. Owens, and A. A. Efros, "Fighting fake news: Image splice detection via learned self-consistency," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 101–117.

[53] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[54] R. Raghavendra, K. B. Raja, S. Venkatesh, and C. Busch, "Transferable deep-CNN features for detecting digital and print-scanned morphed face images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1822–1830.

[55] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: A compact facial video forgery detection network," 2018, *arXiv:1809.00888*. [Online]. Available: http://arxiv.org/abs/1809.00888

[56] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Two-stream neural networks for tampered face detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1831–1839.

[57] L. M. Dang, S. I. Hassan, S. Im, and H. Moon, "Face image manipulation detection based on a convolutional neural network," *Expert Syst. Appl.*, vol. 129, pp. 156–168, Sep. 2019.

[58] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial Landmarks)," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017.

[59] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Dec. 2001.

[60] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[61] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, C. A. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[62] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

EUNJI KIM received the B.S. degree in mathematical sciences and the Ph.D. degree in industrial engineering from Seoul National University, Seoul, South Korea, in 2012 and 2019, respectively. She is currently an Assistant Professor with the School of Business Administration, Chung-Ang University, Seoul. She was a Staff Researcher at Samsung Advanced Institute of Technology (SAIT). Her current research interests include data mining, deep learning, and their applications in various areas, such as manufacturing, finance, and media.

SUNGZOON CHO is currently a Professor with the Department of Industrial Engineering, College of Engineering, Seoul National University, Seoul, South Korea. He has published over 100 papers in various journals and proceedings. He also holds a U.S. patent and a Korean patent concerned with keystroke-based user authentication. His research interests include neural networks, pattern recognition, data mining, and their applications in various areas, such as response modeling and keystroke-based authentication.