Deepfake Face Image Detection

I. Introduction

With the rapid development of deep learning technology, especially the generative adversarial networks (GAN), fake multimedia has become a central problem in the last few years. Using the advanced deep learning tools, like autoencoders (AE) or generative adversarial networks, creating realistic manipulated media assets, such as images and videos, becomes very easy, provided one can access large amounts of data. Fig. 1 shows some manipulated face images from 100k-generated dataset and DeepfaceTIMIT dataset. Such fake images sometimes even cheat our eyes. These manipulated multimedia becomes threats to biometric community and public security. Thereby, more attention need to be paid on fake detection.

Various manipulation methods have been proposed recently. It can be divided into two categories roughly: (1) traditional computer-graphic based method, (2) deep learning based method. For traditional methods, FaceSwap [1] and Face2Face [2] mentioned in paper [3] are two typical computer graphics-based approaches for facial identity manipulation and facial expression manipulation, respectively. For deep learning based methods, DeepFakes [4] uses two autoencoders with a shared encoder to learn the target face. DeepfakeTIMIT [5] database is generated via the open source GAN-based approach [6], which, in turn, was developed from the original autoencoder-based Deepfake algorithm [4], for manipulation. The 100k-generated dataset from Flickr-Face-HQ Dataset [7] is generated by StyleGAN [8]. NeuralTextures [9] is a learning based manipulation method trying to learn the neural texture of target face from the source face.

Based on the various of manipulation methods, a large number of fake face databases are released too, such as 100k-generated dataset [7], FaceForensics++ [3] and DeepfakeTIMIT [5]. The availability of large-scale datasets of DeepFake videos is an enabling factor in the development of DeepFake detection method. The 100k-generated dataset [7] is generated by StyleGAN [8] based on Flickr-Faces-HQ (FFHQ) dataset, which consists of 100,000 high-quality images at 1024x1024 resolution. FaceForensics++ [3] is created by four manipulation methods (FaceSwap [1], Face2Face [2], DeepFakes [4] and NeuralTextures [9]) which generates 1.8 million fake faces. DeepfakeTIMIT [5] is generated by StyleGAN [8] from VidTIMIT [10] database.

Recently, Facebook in collaboration with other companies and academic institutions(e.g., Microsoft, Amazon, MIT) launched a challenge named the Deepfake Detection Challenge (DFDC). They released a preview dataset [11] consisting of 1,131 real videos from 66 paid actors, and 4,119 fake videos. The Google DeepFake detection dataset (DFD) [12] contains 3,068 DeepFake videos generated based on 363 original videos of 28 consented individuals of various genders, ages and ethnic groups. Existing DeepFake video datasets reveal several common visual artifacts, such as low-quality synthesized faces, visible splicing boundaries, color mismatch, visible parts of the original face, and inconsistent synthesized face orientations.

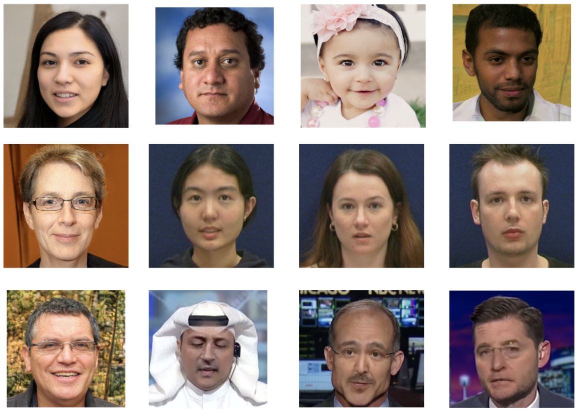

Fig. 1. Samples from several manipulation face databases.

Celeb-DF [13] and DeeperForensics-1.0 [14] are two databases aiming to provide fake videos of better visual qualities. Celeb-DF [13] consists of 590 real videos extracted from Youtube, and 5,639 fake videos, which were created through a refined version of a public DeepFake generation algorithm, improving aspects such as the low resolution of the synthesized faces and colour inconsistencies. DeeperForensics-1.0 [14] represents the largest face forgery detection dataset by far, with 60,000 videos in total, including 50,000 original collected videos and 10, 000 manipulated videos. It constructs a dataset more suitable for real-world face forgery detection by designing this dataset with careful consideration of quality, scale, and diversity.

This creating realistic manipulated faces technology can be used in many applications, like movie productions, photography, video-games and virtual reality. However, it brings bad things too, like creating fake face image to login a photo face based security door, or building fake-news campaigns to manipulate the public opinion. In the long run, it may also reduce trust in journalism, including serious and reliable sources. Actually, this is not a new problem. Image manipulation has been carried out since photography was born, and powerful image/video editing tools, such as Photoshop or the open source software GIMP, have been around for a long time. Using such conventional signal processing methods, images can be easily modified, obtaining realistic results that can fool even a careful observer.

In this paper, we try to extract more representative features of fake face images and cast the forgery detection as a binary classification problem (real/fake). The paper starts with the description of several image features (Section II). Then, Three manipulated face databases are introduced in Section III. Section IV gives a through analysis and comparison of four methods on three databases.

II. Method

This section talks about several image features used for detecting artificial images contents, more specifically, fake faces.

A. Fast Fourier Transform (FFT) based Feature

This method analyzes the characteristics of images on a domain defined by the Fourier transform. Given an image applied with the Fourier transform, the Fourier coefficients indicate the energy distribution of the image over a range of frequencies. The frequency domain analysis have been widely used in various applications [15]–[17], such as signal detection, image denoising and image inpainting, etc. Fig. 2 shows the pipeline of the FFT based detection method. It consists of three steps. First, applying fast Fourier transform (FFT) to the input image. Second, calculating azimuthal average of the spectrum energy. The last step is binary classification. Below we will brief describe the three steps.

Discrete Fourier transform (DFT): Given a gray image of size $M \times N$, let $I(x, y)$ denotes the value of the pixel that locates at (x, y). The DFT of the image is given by:

$$F_{u,v} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x, y) \cdot e^{-j2\pi(ux/M+vy/N)}, \qquad (1)$$

where $u = 0, 1, 2, ..., M - 1$ and $v = 0, 1, 2, ..., N - 1$. The input image $I(x, y)$ is the spatial domain equialent. $F_{u,v}$ represents the image in frequency domain. The image in the spatial and frequency domain is of the same size. The frequency domain coefficients of the image carries information about amplitude and phase over a range of frequencies. In this work, we only use the amplitude information to analyze the images. To decrease the required computations, we employ the FFT to compute DFT. Fig. 3 shows the power spectrum of an input image. Note that the power spectrum is the power of amplitude spectrum.

Azimuthal average computation: After obtain the power spectrum of the image, the azimuthal average can be computed by taking the average power spectrum over each frequency channel. Fig. 4 gives the details about how to compute average of the spectrum energy along azimuth. The left panel is the power spectrum of the input image. The left panel is the corresponding azimuthal average. The yellow circle on the left image represents the spectrum components with the similar frequency. On the left image, the intersection of the yellow straight line and blue curve represents the averaged energy (power spectrum) on the yellow circle. Different from power spectrum, azimuthal average represents the image as a one-dimensional (1D) vector. It reveals the distribution of the power spectrum of the image. Fig. 5 presents several samples (including fake and real images) along with the corresponding azimuthal average. Comparing with the fake images, one can see that the azimuthal averages from real images have different power distribution on high frequency.

Classification: Fake detection is to map the 1D azimuthal average vector to detected results (real or fake). The training process is to learn the mapping function. In the classification, the learned function is used to classify the image by its 1D azimuthal average vector. Some detection results of FFT feature based method are given in Fig. 6.

B. Adaptive Interpolation Method

Taking into account the negative influence bring by the low resolution of images, we utilize the interpolation method to generate the images of high resolution. Then the fake detection metthods are applied to the generated high resolution images. Several classical interpolation techniques such as bilinear, cubic and spline, etc, are widely used in many real time applications. However, these methods do not preserve the spatial details of the source image which leads to annoying artifacts like blurriness, zig-zagging, etc. In this work, we employ an adaptive interpolation method [18] to generate the images of high resolution.

The adaptive interpolation method divede the unknown pixels into several bins depending upon the characteristics of the neighboring pixels (activity level). But instead of finding least square based predictor for each bin, a fixed set of prediction coefficient is defined for prediction of unknown pixels. Different set of prediction parameters are proposed for both edgy and smooth images. Selection of prediction parameter is done on block by block basis instead of image basis.Thus using these fixed set of predictors do not requires any least squares estimation which results into less consumption of computational power.

C. Image Quality Measure (IQM)

Image quality analysis perform outstandingly in image manipulation detection of the forensic field. Image quality assessment has been adopted in face anti-spoofing method and gained a pretty good performance [19], which indicated that the analysis of the general image quality on real face images reveals highly valuable information that may be very efficiently used to discriminate them from fake images.

The goal of the image quality measure is to provide a quantitative score that describes the degree of fidelity or, conversely, the level of distortion of a given test image according to an original distortion-free image. Expected quality differences between real and fake samples may include: degree of sharpness, color and luminance levels, local artifacts, amount of information found in both type of images (entropy), structural distortions or natural appearance.

IQM-18: A total of 18 general image quality features of 25 mentioned in Table 1 of paper [20] are extracted from one image. They are MSE (Mean Squared Error), PSNR (Peak Signal to Noise Ratio), SNR (Signal to Noise Ratio), SC (Structual Content), MD (Maximum Difference), AD (Average

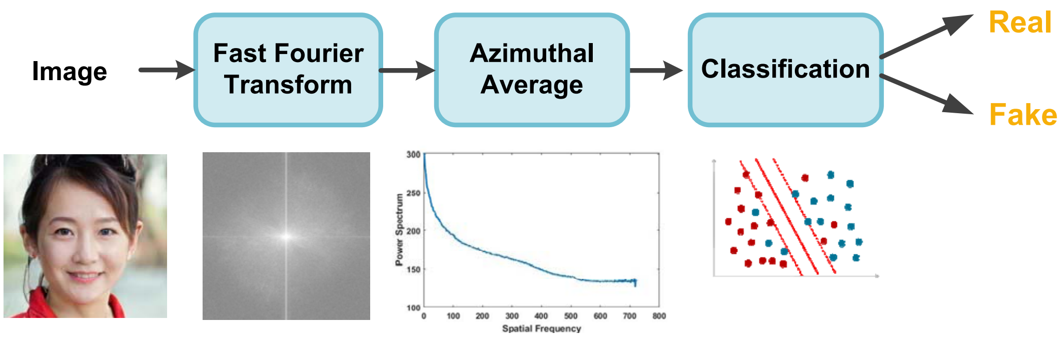

Fig. 2. Overview of the pipeline of the detection method. It contains three steps. First, applying Fast Fourier Transform (FFT) to the input image. Second, calculating azimuthal average of the spectrum energy. The last step is classification.

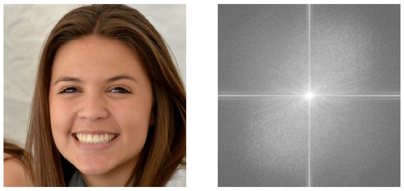

Fig. 3. The outcome of applying FFT to an image.

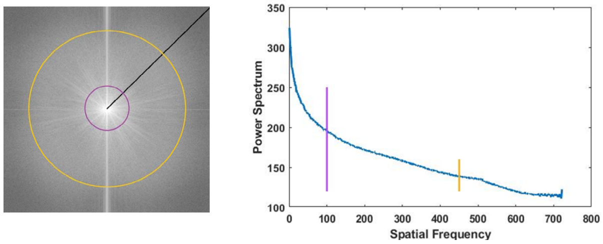

Fig. 4. The details about how to compute azimuthal average by taking the average of the spectrum energy along azimuth. The left panel is the power spectrum of the input image. The left panel is the corresponding azimuthal average.

Difference), NAE (Normalized Absolute Error), RAMD (R-Averaged MD), LMSE (Laplacian MSE), NXC (Normalized Cross-Correlation), MAS (Mean Angle Similarity), MAMS (Mean Angle Magnitude Similarity), SME (Spectral Magnitude Error), GME (Gradient Magnitude Error), GPE (Gradient Phase Error), SSIM (Structural Similarity Index), VIF (Visual Information Fidelity), and HLFI (High-Low Frequently Index).

The following gives the formulas of some measures. Peak Signal to Noise Ratio is calculated in Eqn. (2). Structural Content is computed in Eqn. (3). Mean Angle Similarity is calculated by Eqn. (4).

$$PSNR(I,\hat{I}) = 10log(\frac{max(I^2)}{MSE(I,\hat{I})}) \tag{2}$$

$$SC(I,\hat{I}) = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}(I_{i,j})^2}{N \cdot M \cdot MSE(I,\hat{I})} \tag{3}$$

$$MAS(I,\hat{I}) = 1 - \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}(\alpha_{i,j}) \tag{4}$$

IQM-121: It is another discriminative feature that can be capable of differentiating between genuine and fake faces based on a single frame. Four different features, i.e., specular reflection, blurriness, chromatic moment, and color diversity, are extracted in our work, constituting a 121-dimensional feature vector.

Specular reflection feature is based on the assumption that the illumination is from a single source of uniform color and not over-saturated. Three dimensional features are extracted to represent the specularity intensity distribution by calculating the specular reflection component image. Two types of blurriness features are utilized: one is measured based on the difference between the original input image and its blurred version; one is measured based on the average edge width in the input image. Chromatic moment feature is calculated by computing mean, deviation, and skewness of each channel in HSV (Hue, Saturation, and Value) space of facial images. Color diversity features is extracted by measuring the image color diversity via histogram bin counts of the top 100 most frequently appearing colors and the number of distinct colors appearing in the normalized face images.

A total of 121-dimensional image distortion based feature proposed by et al. Wen [21] are denoted as IQM-121 in this work and are utilized for fake detection. More specifically, according to the Dichromatic Reflection Model [22], light reflectance I of an object at a specific location x can be decomposed into the following diffuse reflection (I_d) and specular reflection (I_s) components:

$$I(x) = I_d + I_s = w_d(x)S(x)E(x) + w_s(x)E(x), \tag{5}$$

where $E(x)$ is the incident light intensity, $w_d(x)$ and $w_s(x)$ are the geometric factors for the diffuse and specular reflections, respectively, and $S(x)$ is the local diffuse reflectance ratio. The formation of fake images intensity $I'(x)$ can be modeled as follows:

$$I'(x) = I'_d + I'_s = F\left(I(x)\right) + w'_s(x)E'(x). \tag{6}$$

Note that Eqns. (5) and (6) only model the reflectance difference between genuine and fake images and do not considered the final image quality after camera capture. Therefore, the total distortion in $I'(x)$ compared to $I(x)$ consists of two parts: distortion in the diffuse reflection component (I'_d) and distortion

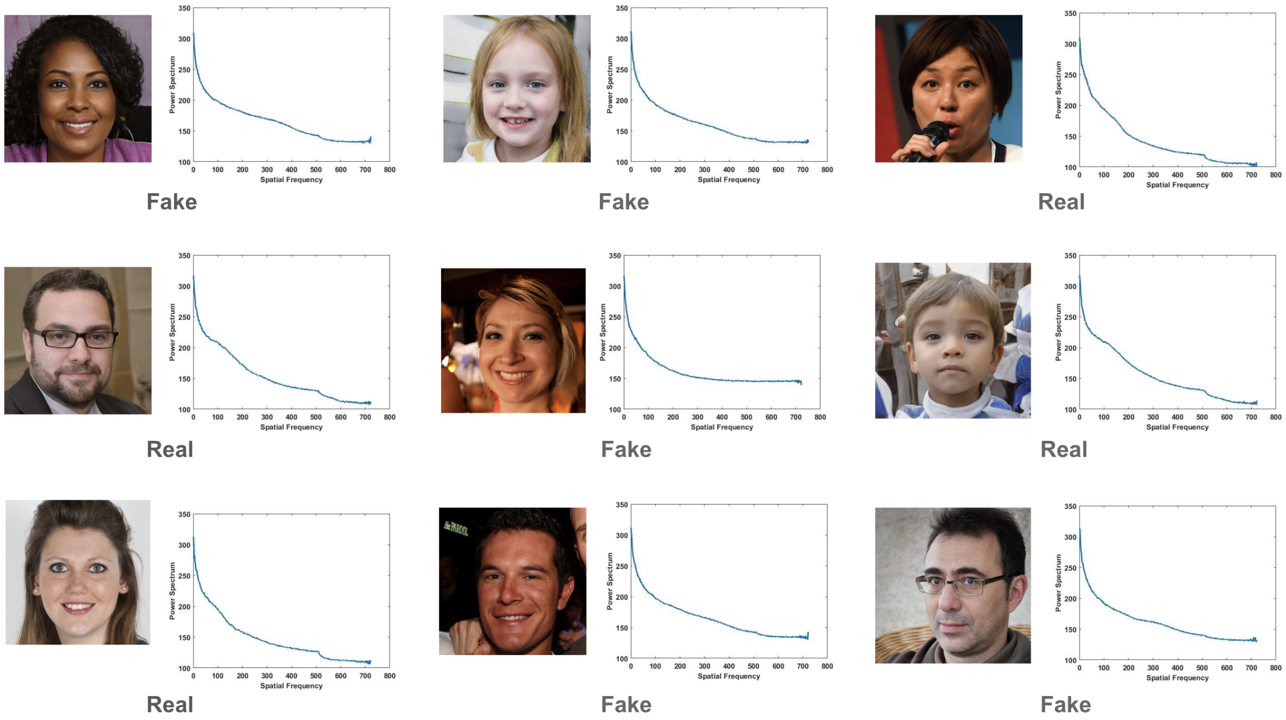

Fig. 5. Samples (including fake and real images) along with the corresponding azimuthal average. Comparing with the fake images, the azimuthal averages from real images have different power distribution on high frequency.

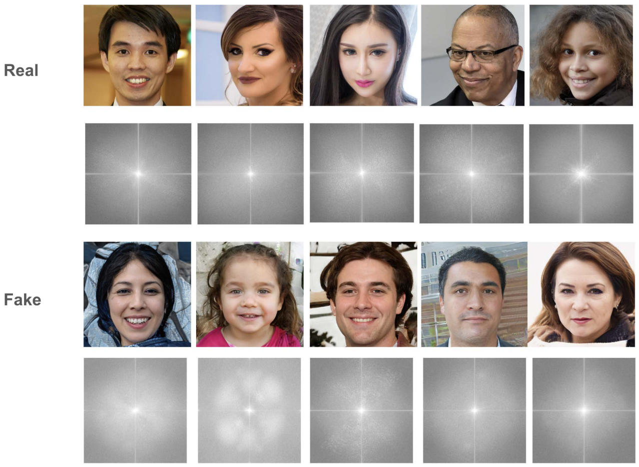

Fig. 6. Some detection results of FFT feature based method.

in the specular reflection component (I'_s). In particular, I'_d is correlated with the original face image $I(x)$, while I'_s is independent of $I(x)$. The distortion function $F(\cdot)$ in the diffuse reflectance component can be modeled as:

$$F(I(x)) = H(GI(x)) \tag{7}$$

where $g(\cdot)$ is a low pass point spread function and $H(\cdot)$ is a histogram transformation function (distorting color intensity). In this work, we utilize Bob toolbox [23] to extract IQM-121 from images.

D. *Steganalysis Feature*

The main idea of steganalysis feature [24] is that a rich model should consist of a large number of diverse submodels. The author [24] assembled a rich model of the noise component as a union of many diverse submodels. These submodels are formed by joint distributions of neighboring samples from quantized image noise residuals obtained using linear and nonlinear high-pass filters, which consider various types of relationships among neighboring samples of noise residuals obtained by linear and nonlinear filters with compact supports. The final rich model contains 106 submodels and the feature gets a total of 34,671 dimension.

E. *Natural Scene Statistics based Features*

Natural image mean the real image captured by regular cameras. The natural undistorted image shows certain statistical properties. The presence of distortions in natural images alters the natural statistical properties of images, thereby rendering them and their statistics unnatural. Natural scene statistics (NSS) models seek to capture these statistical properties of natural scenes that hold across different contents. In this work, we selected seven types of NSS feature from well known no-reference image quality assessment (NR-IQA) algorithms: (1) Spatial and spectral entropy feature [25], (2) BRISQUE feature [26], (3) BLIINDS-II feature [27], (4) DIIVINE feature [28], (5) Curvelet feature [29], (6) NIQE feature [30] and (7) TMIQA features [31].

1) *Spatial and Spectral Entropy Feature*

Natural photographic images are highly structured in the sense that their pixels exhibit strong dependencies in space and frequency. These dependencies carry important information about the visual scene. The spatial entropy is a function of the probability distribution of the local pixel values, and the spectral entropy is a function of the probability distribution of the local discrete cosine transform (DCT) coefficients. The entropy features are highly sensitive to the degrees and types of image distortion. Spatial and spectral entropy [25] from local image blocks on the block spatial scale responses and the block DCT coefficients, are calculated as features. The input image is decomposed into 3 scales(low, middle and high) yielding 3 scale responses. For three scales, the combined final feature size is 12.

2) *BRISQUE Feature*

The approach of Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) [26] is a NSS based distortion-generic blind/no-reference (NR) image quality assessment model in spatial domain. It is based on the assumption that statistical regularities of natural images are disturbed when image distortions are introduced. It uses scene statistics of locally normalized luminance coefficients to quantify possible losses of “naturalness” due to the presence of distortions in the image. The feature are derived from the empirical distribution of locally normalized luminances and products of locally normalized luminances under a spatial natural scene statistic model. Given an input image, locally normalized luminances are computed via local mean subtraction and divisive normalization. The size of final feature is 36.

3) *BLIINDS-II Feature*

BLIINDS-II [27], BLind Image Integrity Notator using DCT Statistics-II, is an efficient blind/ no-reference image quality assessment algorithm. It uses a NSS model of DCT coefficients. First, the input image is partitioned into equally sized 5*5 blocks. And then a local 2-D DCT is computed on each of these blocks. Four types of model-based features are computed: Shape parameter, coefficient of frequency variation, energy subband ratio measure, orientation feature. The size of final feature is 24.

4) *DIIVINE Feature*

DIIVINE [28], Distortion Identification-based Image Verity and INtegrity Evaluation index, is a NSS based QA algorithm that assesses the quality of a distorted image without a reference image. The input image goes through a wavelet decomposition using a steerable pyramid decomposition, over two scales and six orientations. The resulting decomposition results in 12 subbands across orientations and scales. The obtained subband coefficients are then utilized to extract a series of statistical features, stacked to form a vector, which is a statistical description of the distortion in the image. The dimension of extracted feature is 88.

5) *Curvelet Feature*

Curvelet Feature [29] is an intermediate-level image feature which is extracted from the curvelet image transform. It captures regularities arising in low-level NSS models in a localized way, and consequently capture perceptual image distortions in a content independent way. The result shows that it is sensitive to the presence and severity of image distortion. The input image is divided into blocks of size 256*256, and curvelet feature is extracted from each block, yielding a set of feature vectors. Then, the mean feature vectors are calculated to create the final feature vector which contains 12 dimension.

6) *NIQE Feature*

Natural Image Quality Evaluator (NIQE) feature [30] is based on the construction of a quality aware collection of statistical measures. It is similar to the feature used in BRISQUE [26]. The input image is partitioned into 96*96 image patches. Specific NSS features are then computed from the coefficients of each patch. Four parameters are computed along the four orientations which yields 16 parameters. Combined with the two parameters computed from original coefficients, it yields 18 overall features. All features are computed at two scales to capture multiscale behavior, by low pass filtering and downsampling by a factor of 2, yielding a final feature set of size 36, extracted from each patch.

7) *TMIQA Feature*

TMIQA [31], called Topic Model based Image Quality Assessment, is also based on the hypothesis that distorted images have certain latent characteristics that differ from those of ’natural’ or ’pristine’ images. It is a quality-aware, NSS based BRISQUE [26] feature, which applies a topic model on image patches represented in a suitable quality-aware space, and then examining the topic distributions for each image. Given an input image, it is divided into overlapping patches of size 64*64, with an overlap of 8*8 between neighboring patches, and local BRISQUE features are computed from each patch. This gives a set of 36 features per patch.

F. *Learning-based Features*

Xception [32] is a CNN based deep network trained on ImageNet [33]. FaceForensics++ [3] transfer it to the forgery detection task by replacing the final fully connected layer with two outputs. We use the well-trained network for our classification.

III. Dataset

This section talks about a few manipulation face databases used in our experiments, such as 100k-generated dataset from Flickr-Face-HQ Dataset [7], FaceForensics++ [3] database by four manipulation methods and DeepfakeTIMIT [5] generated from VidTIMIT [10].

A. *100k-generated dataset from Flickr-Face-HQ Dataset by StyleGAN*

The 100k-generated dataset [7] is generated by a deep network called StyleGAN [8] based on Flickr-Faces-HQ (FFHQ) dataset, which consists of 100,000 high-quality images at 1024x1024 resolution. The 100k-generated dataset is a pretty huge manipulation face dataset with large number of different identities and images.

Flickr-Faces-HQ (FFHQ) [8] is a high-quality image dataset of human faces, originally created as a benchmark for Generative Adversarial Networks (GAN). It is crawled from Flickr and automatically aligned and cropped using dlib. The dataset consists of 70,000 high-quality images at 1024x1024 resolution and contains considerable variation in terms of age, viewpoint, lighting, ethnicity and image background, and also has much better coverage of accessories such as eyeglasses, sunglasses, hats, etc.

StyleGAN is a style-based generator architecture for GAN to expose some ways to control the image synthesis process. The generator starts from a learned constant input and adjusts the "style" of the image at each convolution layer based on the latent code, therefore directly controlling the strength of image features at different scales. Combined with noise injected directly into the network, this architectural change leads to automatic, unsupervised separation of high-level attributes (e.g., pose, identity) from stochastic variation (e.g., freckles, hair) in the generated images, and enable intuitive scale-specific mixing and interpolation operations. The 100k-generated faces has the characteristics of both source images. Figure 7 gives some samples. The real images are from FFHQ dataset, and the fake images are from 100k-generated dataset.

B. *FaceForensics++*

FaceForensics++ [3] is a manipulation face video dataset. It contains 1000 original videos downloaded from youtube under uncontrolled environment and 4000 manipulation videos by four facial manipulation methods (Face2Face [2], FaceSwap [1], DeepFakes [4], NeuralTextures [9]). In them, Face2Face [1] and NeuralTextures [9] are two facial expression manipulation methods and FaceSwap [1] and DeepFakes [4] are two facial identity manipulation methods. Facial expression manipulation method enables the transfer of facial expressions of one person to another person. Identity manipulation is to replace the face of a person with the face of another person.

FaceSwap [1] and Face2Face [2] are two computer graphics-based approaches. DeepFakes [4] and NeuralTextures [9] are two learning-based methods. FaceSwap [1] tries to fit a 3D template model using blendshapes by back-projecting to the target image by minimizing the difference between the projected shape and the localized landmarks. DeepFakes [4] is based on two autoencoders with a shared encoder, in which one is trained to reconstruct training images of the source and the other one is trained to reconstruct the target face. The well-trained encoder and decoder on source faces are used on target face.

Face2Face [2] first reconstructs identity and tracks the expressions, then generates the reenactment video outputs by transferring the source expression parameters of each frame to the target video. NeuralTextures [9] uses the original video data to learn a neural texture of the target person.

By using these four methods, a total of 1.8 million manipulation images are generated from all 4000 manipulation videos with pristine (i.e., real) sources and target ground truth to enable supervised learning. It is over an order of magnitude larger than comparable publicly available forgery datasets. An automated benchmark based on this database for forgery detection are proposed in realistic scenario, i.e., with random compression and dimensions.

C. *DeepfakeTIMIT*

DeepfakeTIMIT [5] is the first publicly available set of Deepfake videos generated from videos of VidTIMIT [10] database. It uses the open source GAN-based approach [6], which, in turn, was developed from the original autoencoder-based Deepfake algorithm [4].

VidTIMIT [10] is comprised of video and corresponding audio recordings of 43 people, reciting short sentences. The sentences were chosen from the test section of the TIMIT corpus. One video is generated for one sentence. Each person says ten sentences resulting 10 videos per person. The first two sentences for all persons are the same, with the remaining eight generally different for each person. The recording was done in an office (controlled) environment with people facing camera and reciting predetermined short phrases using a broadcast quality digital video camera. The video of each person is stored as a numbered sequence of JPEG images with a resolution of 512 x 384 pixels. 90% quality setting was used during the creation of the JPEG images.

When creating the DeepfakeTIMIT [5] database, 16 similar looking pairs of people are manually selected from VidTIMIT database. Subjects in the same pair have similar prominent visual features, e.g., mustaches or hairs styles. GAN-based face-swapping algorithm is used to generate videos with swapped faces from subject one to subject two and visa versa. For each of 32 subjects, two different models are trained and two versions of the videos are generated: lower quality (LQ) with 64 x 64 input/output size model and higher quality (HQ) with 128 x 128 size model. The blending techniques are different in these two models. In low quality model, a face was generated using a frame from a target video as an input, and a facial mask is detected via a learning-based face segmentation method [34]. The detected mask is used to blend the generated face with the face in the target video. In high quality model, a facial landmark detection method MTCNN [35] is used between generated face and the original face in the target video to blend the generated face with the face in the target video. Besides, a histogram normalization is applied to adjust for the different lighting conditions. Finally 320 videos are generated corresponding to each version, resulting in 640 total videos with faces swapped.

IV. Experiments

In our experiment, we extracted four types of image features, including Fast Fourier Transform (FFT), Image Quality Measure (IQM), Image Distortion Analysis (IDA), and deep learning based feature, on three manipulation face datasets, for binary classification (real or fake) using Support Vector Machine (SVM) with linear and RBF kernal.

A. *Comparison on different methods*

This subsection compare the performance of FFT based method, image quality based method, image distortion based method and deep learning method.

For binary classification, we use both original face images and manipulated faces. The experiment on 100k-generated fake faces, both 100k-generated (100k images) and FFHQ (70k images) faces are used. All images are split into training (85k) and test sets(85k). FFT features are test on all test dataset. Since IQM, IDA, Xception are time consuming, we extract a

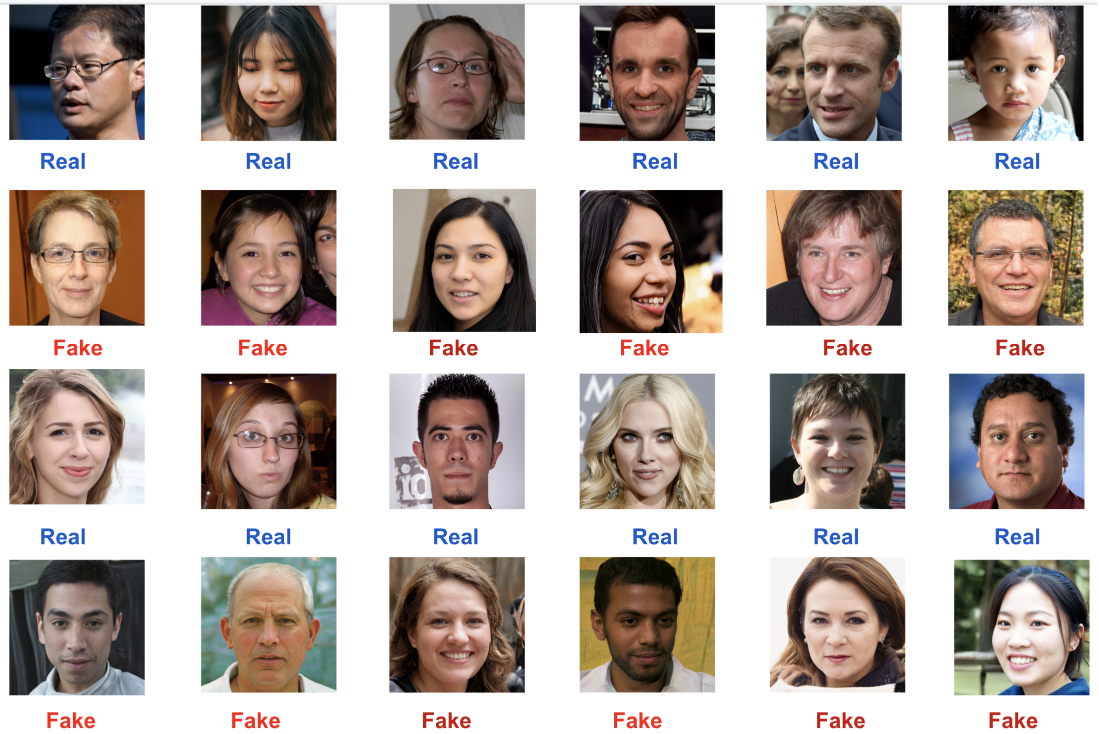

Fig. 7. Samples of 100k-generated faces and Flickr-Face-HQ faces. The real images are from FFHQ dataset, and the fake images are from 100k-generated dataset.

TABLE I
THE ACCURACY (%) OF FAKE DETECTION ON 100K-GENERATED DATASET BY THE FFT METHOD AND OTHER THREE METHODS.

Method	SVM	Accuracy(%)
FFT	Linear	99.99
	RBF	95.20
IQM-18	Linear	94.20
	RBF	95.35
IDA-121	Linear	70.40
	RBF	60.90
IQM+IDA-139	Linear	93.95
	RBF	90.75
Steganalysis	Linear	100
	RBF	93.12
NSS-sseq	Linear	88.05
	RBF	85.75
NSS-brisque	Linear	76.20
	RBF	77.00
NSS-bliinds-II	Linear	92.40
	RBF	90.10
NSS-diivine	Linear	93.80
	RBF	74.75
NSS-curvelet	Linear	71.70
	RBF	72.30
NSS-niqe	Linear	61.55
	RBF	61.55
NSS-tmiqa	Linear	60.85
	RBF	60.85
Xception	-	41.78

small dataset with 4000 training and 2000 test. Table I gives the accuracies of each methods on 100k-generated dataset.

The experiment on FaceForensics++, we establish a small dataset by extracting 4 frames from each original and manipulated videos. Finally, a total of 20k images (4000 original frames with 4000 manipulation frames for each manipulation method) are generated. Similarly, half real and half fake images are selected for training, and the remaining for test. Table II gives the accuracies of each methods on FaceForensics++ dataset.

For experiment on DeepfakeTIMIT, VidTIMIT [10] is adopted as original videos, in which, 32 subjects (320 videos) are chosen with same subjects in DeepfakeTIMIT. DeepfakeTIMIT [5] is chosen as manipulated version, containing 32 subjects of 640 videos (320 high quality, 320 low quality). A total of 102,047 images are generated finally. And half real and half fake images are selected for training, and the remaining for test too. Table III gives the accuracies of each methods on DeepfakeTIMIT dataset.

B. *Influence to variant sizes of images*

This subsection analyzes the influence of the four methods to various sizes of images on 100k-generated dataset. Table IV shows the accuracy (%) of fake detection on 100k-generated dataset by FFT method and other methods on various sizes of images. One can see that though FFT based method achieves the best performance on the images with size of 1024 × 1024, its performance significantly drops with the decrease of image size. We will analyze the sentivity of methods to interpolation based methods in the next section.

C. *Sensitivity to interpolation method*

Taking into account the negative influence bring by low resolution of images, we propose a method which combines

TABLE II
THE ACCURACY (%) OF FAKE DETECTION ON FACEFORENSICS++ DATASET BY FFT METHOD AND OTHER METHODS. DF REPRESENTS DEEPFAKES METHOD, $F2F$ REPRESENTS FACE2FACE METHOD, FS REPRESENTS FACESWAP METHOD, AND NT REPRESENTS NEURAL TEXTURES METHOD.

Method	SVM	Generating methods of fake images			
		DF	$F2F$	FS	NT
FFT	Linear RBF	67.77 68.48	87.18 87.77	95.98 96.03	76.51 77.90
IQM-18	Linear RBF	73.88 74.06	91.13 88.95	95.97 95.79	77.91 81.20
IDA-121	Linear RBF	70.81 67.44	56.09 56.31	71.65 68.46	64.10 62.93
IQM+IDA-139	Linear RBF	76.26 72.56	89.81 87.58	96.07 94.58	77.22 76.92
Steganalysis	Linear RBF				
NSS-sseq	Linear RBF	61.07 61.12	55.21 53.15	56.98 56.41	63.09 60.92
NSS-brisque	Linear RBF	55.56 62.70	52.83 66.37	56.48 60.79	56.48 60.79
NSS-bliinds-II	Linear RBF	67.12 64.98	54.28 55.76	68.57 64.09	67.35 62.79
NSS-diivine	Linear RBF	70.13 59.03	68.24 55.98	81.88 56.13	70.23 56.99
NSS-curvelet	Linear RBF	71.84 72.02	63.59 71.17	69.92 81.51	67.98 70.23
NSS-niqe	Linear RBF	65.69 50.11	62.01 50.08	61.51 50	56.79 50.08
NSS-tmiqa	Linear RBF	65.06 50.31	58.16 50.03	56.41 50.05	63.12 50.10
Xception	-	91.35	98.75	98.78	53.38

TABLE III
THE ACCURACY (%) OF FAKE DETECTION ON DEEPFAKETIMIT DATASET BY FFT METHOD AND OTHER METHODS.

Method	SVM	Subset		
		Low quality	High quality	ALL
FFT	Linear RBF	81.89 81.79	79.04 77.50	79.60 79.40
IQM-18	Linear RBF	98.21 95.96	90.61 89.24	93.76 93.04
IDA-121	Linear RBF	86.94 81.63	78.22 69.82	83.06 78.04
IQM+IDA-139	Linear RBF	95.19 90.47	87.94 83.39	91.25 87.53
Steganalysis	Linear RBF			
NSS-sseq	Linear RBF	100 100	100 100	100 100
NSS-brisque	Linear RBF	92.77 91.85	79.01 73.82	86.07 75.27
NSS-bliinds-II	Linear RBF	98.06 96.20	94.92 91.72	96.52 94.23
NSS-diivine	Linear RBF	98.89 97.75	98.70 96.43	98.31 97.22
NSS-curvelet	Linear RBF	99.36 99.09	97.88 98.43	98.21 98.67
NSS-niqe	Linear RBF	88.05 81.13	71.35 64.42	78.96 74.25
NSS-tmiqa	Linear RBF	91.07 83.18	77.98 64.40	84.69 66.72
Xception	-	49.98	50.01	33.33

TABLE IV
THE ACCURACY (%) OF FAKE DETECTION ON 100K-GENERATED DATASET BY FFT METHOD AND OTHER METHODS ON VARIOUS SIZES OF INPUT IMAGES.

Method	SVM	Image size				
		1024	800	600	400	299
FFT	Linear RBF	99.99 95.20	99.25 99.65	92.75 88.25	75.60 69.05	65.30 64.05
IQM-18	Linear RBF	94.20 95.35	93.15 93.02	92.89 92.11	91.45 92.01	91.22 91.89
IDA-121	Linear RBF	70.40 60.90	68.54 60.15	67.46 60.01	65.98 60.00	65.27 59.07
IQM+IDA-139	Linear RBF	93.95 90.75	93.15 90.55	92.89 90.01	91.65 89.52	90.15 88.07
Steganalysis	Linear RBF	100 93.12	99.10 91.15	94.35 84.15	91.40 79.15	91.05 60.85
NSS-sseq	Linear RBF	88.05 85.75	81.45 79.10	74.55 73.25	71.60 67.35	68.20 65.20
NSS-brisque	Linear RBF	76.20 77.00	60.85 60.85	60.85 63.90	60.85 60.85	60.85 60.85
NSS-bliinds-II	Linear RBF	92.40 90.10	84.60 77.75	74.80 64.55	65.85 63.65	65.95 63.55
NSS-diivine	Linear RBF	93.80 74.75	82.90 62.75	72.45 61.80	70.60 61.35	70.10 61.30
NSS-curvelet	Linear RBF	71.70 72.30	70.75 70.05	70.05 71.45	67.10 64.60	64.90 62.90
NSS-niqe	Linear RBF	61.55 61.55	61.50 62	61.65 61.90	61.55 61.45	61.40 61.60
NSS-tmiqa	Linear RBF	60.85 60.85	60.85 60.85	60.85 60.85	60.85 60.85	60.85 60.85
Xception	-	41.78	40.65	40.70	40.80	40.65

an adaptive interpolation method with FFT/IQM-18 for fake detection. More specifically, first, this method utilizes an adaptive interpolation method (described in Section II-B) to generate images with high resolution. Then the FFT based method and IQM-18 method are applied to these generated images, respectively. Table V shows the results on 100k-generated database. Here ”299 to 600“ represents the set of images which are resized from 299 × 299 to 600 × 600 by the adaptive interpolation method. Comparing with the results given in Table IV, one can see that Adaptive+FFT method achieves higher accuracy than FFT based method on each set, while Adaptive+IQM18 method can not outperform the IQM18. This demonstrates that combining the adaptive interpolation with FFT leads to better performance on fake detection.

REFERENCES

[1] F. github, “https://github.com/marekkowalski/faceswap,” *Accessed:2018-10-29*, 2018.

[2] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, “Face2face: Real-time face capture and reenactment of rgb videos,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2387–2395.

[3] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, “Faceforensics++: Learning to detect manipulated facial images,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1–11.

[4] D. github, “https://github.com/deepfakes/faceswap,” *Accessed:2018-10-29*, 2018.

[5] P. Korshunov and S. Marcel, “Deepfakes: a new threat to face recognition? assessment and detection,” *arXiv preprint arXiv:1812.08685*, 2018.

[6] faceswap GAN github, “https://github.com/shaoanlu/faceswap-gan,” *Accessed:2018*, 2018.

TABLE V
THE ACCURACY (%) OF FAKE DETECTION BY THE ADAPTIVE INTERPOLATION BASED METHOD ON 100K-GENERATED DATASET.

Method	SVM	Size of interpolation			
		299 to 600	400 to 800	600 to 1000	800 to 1000
FFT + Adaptive	Linear RBF	71.34 70.55	85.12 81.15	93.24 88.05	99.31 88.25
IQM18 + Adaptive	Linear RBF	90.75 91.05	92.01 92.11	92.95 92.55	94.10 93.95

[7] 100k-generated images github, "https://github.com/nvlabs/stylegan," *Accessed:2020-01-29*, 2020.

[8] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[9] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–12, 2019.

[10] C. Sanderson and B. C. Lovell, "Multi-region probabilistic histograms for robust and scalable identity inference," in *International conference on biometrics*. Springer, 2009, pp. 199–208.

[11] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer, "The deepfake detection challenge (dfdc) preview dataset," *arXiv preprint arXiv:1910.08854*, 2019.

[12] P. K. A. V. V. T. L. J. C. Nicholas Dufour, Andrew Gully and C. Bregler, "Deepfakes detection dataset (dfd)," 2019.

[13] Y. Li, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics," in *IEEE Conference on Computer Vision and Patten Recognition (CVPR)*, Seattle, WA, United States, 2020.

[14] L. Jiang, R. Li, W. Wu, C. Qian, and C. C. Loy, "Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection," in *CVPR*, 2020.

[15] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *2007 IEEE Conference on computer vision and pattern recognition*. Ieee, 2007, pp. 1–8.

[16] A. Srivastava, A. B. Lee, E. P. Simoncelli, and S.-C. Zhu, "On advances in statistical modeling of natural images," *Journal of mathematical imaging and vision*, vol. 18, no. 1, pp. 17–33, 2003.

[17] M. Jiang, B.-Y. Cui, N. Schmid, M. McLaughlin, and Z.-C. Cao, "Wavelet denoising of radio observations of rotating radio transients (rrats): Improved timing parameters for eight rrats," *The Astrophysical Journal*, vol. 847, no. 1, p. 75, 2017.

[18] S. P. Jaiswal, V. Jakhetiya, A. Kumar, and A. K. Tiwari, "A low complex context adaptive image interpolation algorithm for real-time applications," in *2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings*. IEEE, 2012, pp. 969–972.

[19] J. Galbally and S. Marcel, "Face anti-spoofing based on general image quality assessment," in *2014 22nd International Conference on Pattern Recognition*. IEEE, 2014, pp. 1173–1178.

[20] J. Galbally, S. Marcel, and J. Fierrez, "Image quality assessment for fake biometric detection: Application to iris, fingerprint, and face recognition," *IEEE transactions on image processing*, vol. 23, no. 2, pp. 710–724, 2013.

[21] D. Wen, H. Han, and A. K. Jain, "Face spoof detection with image distortion analysis," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 746–761, 2015.

[22] S. A. Shafer, "Using color to separate reflection components," *Color Research & Application*, vol. 10, no. 4, pp. 210–218, 1985.

[23] A. Anjos, L. El-Shafey, R. Wallace, M. Günther, C. McCool, and S. Marcel, "Bob: a free signal processing and machine learning toolbox for researchers," in *Proceedings of the 20th ACM international conference on Multimedia*, 2012, pp. 1449–1452.

[24] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[25] L. Liu, B. Liu, H. Huang, and A. C. Bovik, "No-reference image quality assessment based on spatial and spectral entropies," *Signal Processing: Image Communication*, vol. 29, no. 8, pp. 856–863, 2014.

[26] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on image processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[27] M. A. Saad, A. C. Bovik, and C. Charrier, "Blind image quality assessment: A natural scene statistics approach in the dct domain," *IEEE transactions on Image Processing*, vol. 21, no. 8, pp. 3339–3352, 2012.

[28] A. K. Moorthy and A. C. Bovik, "Blind image quality assessment: From natural scene statistics to perceptual quality," *IEEE transactions on Image Processing*, vol. 20, no. 12, pp. 3350–3364, 2011.

[29] L. Liu, H. Dong, H. Huang, and A. C. Bovik, "No-reference image quality assessment in curvelet domain," *Signal Processing: Image Communication*, vol. 29, no. 4, pp. 494–505, 2014.

[30] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a "completely blind" image quality analyzer," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 209–212, 2012.

[31] A. Mittal, G. S. Muralidhar, J. Ghosh, and A. C. Bovik, "Blind image quality assessment without human training using latent quality factors," *IEEE Signal Processing Letters*, vol. 19, no. 2, pp. 75–78, 2011.

[32] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[33] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[34] Y. Nirkin, I. Masi, A. T. Tuan, T. Hassner, and G. Medioni, "On face segmentation, face swapping, and face perception," in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*. IEEE, 2018, pp. 98–105.

[35] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.