Unsupervised and Generalizable Deepfake Detection using Singular Value Decomposition

Syamantak Sarkar[†], Revoti P. Bora[‡], Sudhish N George[†], Kiran Raja[‡]
[†]National Institute of Technology Calicut, India
[‡]NTNU Gjøvik, Norway

***Abstract*—Generalizing deepfake detection remains a challenge as generative models evolve. Existing methods struggle to generalize to unseen deepfake generation schemes, limiting their real-world applicability. This work proposes a novel anomaly detection-based approach that uses reconstruction loss from low-rank Singular Value Decomposition (SVD) representations of images. The low-rank representation of the images suppresses manipulations, which are manifested in the high-rank components. Hence, our proposed approach trains a model to reconstruct the real images from those images' low-rank representation while minimizing the reconstruction loss. Since the model is trained to minimize the reconstruction loss for real images, they exhibit significantly higher reconstruction loss for deepfakes. By thresholding the reconstruction loss, we effectively detect deepfake images. Our method demonstrates high generalization compared to existing approaches in different unseen data sets, achieving an average ROC-AUC improvement of 6% to 29% compared to SOTA approaches. Further, we show that our approach is robust against perturbations (e.g., blur, compression) without performance degradation in cross-manipulation scenarios. In addition, we use heatmaps to explain the difference in reconstruction loss between real and deepfake images. Our code and additional results are made available at GitHub Link.**

***Index Terms*—Deepfake Detection, Low-Rank Representation, Singular Value Decomposition, Reconstruction-Based Approach.**

I. Introduction

Deepfake technology enables the creation of highly realistic synthetic images, raising concerns about the spread of misinformation, identity theft, and privacy violations [19]. Despite various deepfake detection methods based on Convolutional Neural Networks (CNNs) and transformers, generalization across datasets and manipulation with different manipulation techniques remain unsolved [3], [11]. Although artifact localization, detection-based approaches, and frequency-based techniques offer improvements, these approaches remain sensitive to compression and novel forgeries, which limits real-world effectiveness [17], [28]. These challenges highlight the need for a more generalizable deepfake detection approaches.

In addition to many supervised approaches that train CNNs or transformers [3] on labeled datasets, techniques based on anomaly detection and forensic analysis-driven solutions are proposed to detect deepfakes. Anomaly detection methods identify inconsistencies such as unnatural facial movements or physiological signals [26], but rely on handcrafted features that may not generalize well. Forensic analysis detects image artifacts such as compression inconsistencies [13] or frequency anomalies [26], improving robustness against known attacks, but does not address generalization across unseen attacks. perform

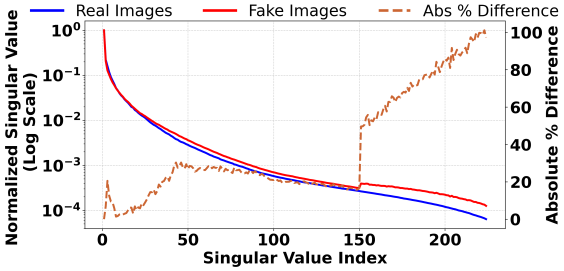

Fig. 1: Plot showing the mean SVD values and percentage change in SVD values (i.e., $|\text{SVD}_{\text{deepfake}} - \text{SVD}_{\text{real}}|$) for each index across 100 images from the FF++ dataset. The plot shows that SVD values of deepfake images exhibit significantly, particularly in the higher indices.

We focus on a generalizable solution for detecting deepfake images in unseen data by learning to reconstruct real/pristine images from the low-rank SVD [18] approximation images. The singular values in lower indices, i.e., low-rank components, retain the image structures while the higher indices retain subtle information about the images [6]. In Figure 1, we show the mean and percentage differences in the SVD values between real and deepfake images (100 images from FF++ [16]). The SVD values are normalized between [0,1] and then log transformed for plotting. The percentage differences of the SVD values are higher for the high-rank components, indicating that the deepfake manipulations mostly affect the high-rank SVD components.

Therefore, we utilize the low-rank approximation of an image[1], approximated from its lower SVD indices, as input to our approach. A model is trained to reconstruct real images from the low-rank representations, thereby learning the distribution of real images. Since the training explicitly focuses on real images, the model struggles to accurately reconstruct deepfake images from their low-rank images, resulting in a higher reconstruction error. This disparity in reconstruction errors allows us to effectively distinguish between real and deepfake images. Recently, some works have used SVDs for

[1] From hereon we will use the term low-rank image to denote low-rank approximation of an image

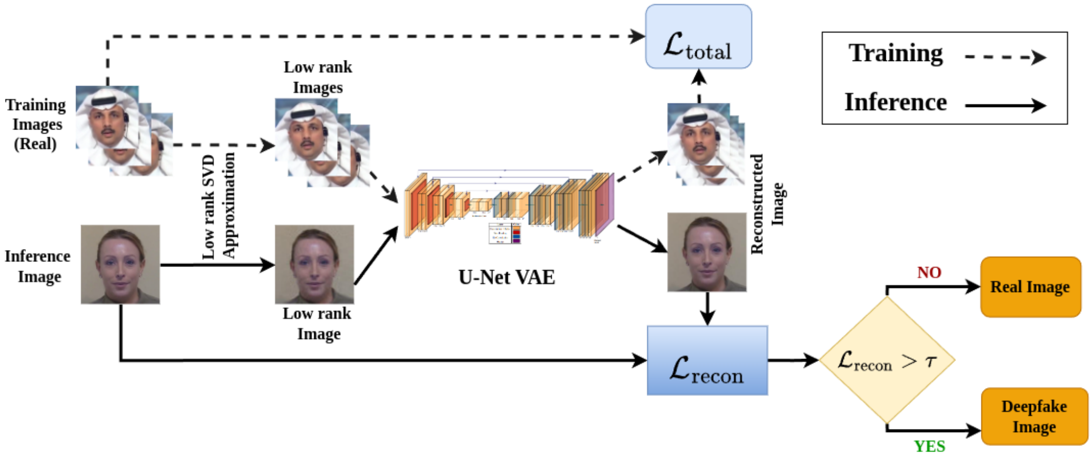

Fig. 2: Overview of the proposed deepfake detection approach. In the training phase (indicated using dotted lines), the U-Net VAE model is trained only using real images. The trained model is used during inference (denoted using solid lines) to compute $\mathcal{L}_{\text{recon}}$. The optimal threshold τ^*, calculated on the validation set (consisting of real and deepfake images), is then applied on $\mathcal{L}_{\text{recon}}$ to classify an image as real vs. deepfake.

analyzing deepfakes. Abdali et al. [1] used SVDs for deepfake detection by extracting eigenfaces, performing tensor decomposition, and leveraging multilinear projections to distinguish real and deepfake images. Further, Yan et al. [25] used SVD to decompose pre-trained weights, preserving knowledge of the real image while learning deepfake-specific features through a residual component. However, none of these approaches reconstruct low-rank images to the original image for detecting deepfakes.

Our contributions in this work are

- We propose a novel deepfake detection approach using reconstruction loss from a low-rank image, making it generalizable.
- We perform extensive experiments on six publicly available deepfake datasets consisting of different generation schemes to show our approach's generalization and cross-manipulation capabilities.
- Further, we demonstrate the robustness of our approach against the commonly seen perturbation of Gaussian blur and different JPEG compression levels.

II. Proposed Method

For a given image $\mathbf{I} \in \mathbb{R}^{m \times n}$, the singular value decomposition can be provided as:

$$\mathbf{I} = \mathbf{U}\mathbf{S}\mathbf{V}^T, \tag{1}$$

Where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices and $\mathbf{S}$ is a diagonal matrix that contains singular values. The low-rank image ($\mathbf{I}_{\text{low}}$), which retains only the top-k singular values that preserve $\approx 90\%$ of the total energy, is given by:

$$\mathbf{I}_{\text{low}} = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \quad \text{s.t.} \quad \frac{\sum_{i=1}^{k} \sigma_i^2}{\sum_{i=1}^{r} \sigma_i^2} \approx 0.90. \tag{2}$$

Where, r is the full rank of the image matrix and σ_i denotes the i^{th} singular value. The vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ represent the left and right singular vectors corresponding to σ_i, respectively.

The low-rank image ($\mathbf{I}_{\text{low}}$ from Equation (2)) is used to reconstruct the original image $\mathbf{I}$ using a U-Net Variational Autoencoder (U-Net VAE). U-Net VAE preserves spatial details while learning efficient latent representations [15] making it an apt choice for the proposed approach. The U-Net VAE is trained to minimize the reconstruction loss of real images as shown in Figure 2 (dotted line), thus learning the distribution of real images. The total loss ($\mathcal{L}_{\text{total}}$) includes three aspects, i.e., Mean Squared Error (MSE), Kullback Leibler (KL) divergence, and L_1 Loss, is given by:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{MSE}} + \beta\mathcal{L}_{\text{KL}} + \lambda\mathcal{L}_{\text{L1}}. \tag{3}$$

Where, $\mathcal{L}_{\text{MSE}}$ is the Mean Squared Error, $\mathcal{L}_{\text{KL}}$ is the Kullback-Leibler (KL) Divergence, and $\mathcal{L}_{\text{L1}}$ is the L_1 loss. The $\mathcal{L}_{\text{MSE}}$ minimizes the difference between the original image and its reconstructed counterpart [2] while $\mathcal{L}_{\text{KL}}$ and $\mathcal{L}_{\text{L1}}$ enhances the generalization capability of the trained model [12], [22]. The hyperparameters β and λ control the contribution of the KL divergence and L_1 loss, respectively. Higher values of β and λ enforce stronger regularization on the latent space, reducing overfitting. These parameters are tuned empirically to balance reconstruction quality and generalization.

Since training is conducted exclusively on real images, the total loss function ($\mathcal{L}_{\text{total}}$) cannot distinguish between real images and deepfakes. To determine an appropriate reconstruction loss threshold, a separate validation set containing both real and fake images is utilized. Specifically, the optimal threshold (τ^*) must be identified. The trained U-Net VAE model computes reconstruction losses for all images in the validation set, generating two distinct distributions (one for real images and another for deepfakes) to determine (τ^*).

$$\tau^* = \arg\max_{\tau} J(\tau) \tag{4}$$

where $J(\tau)$, i.e., Youden's Index [27] can be expressed as:

$$J(\tau) = \int_{\tau}^{\infty} p_f(x)\,dx + \int_{-\infty}^{\tau} p_r(x)\,dx - 1 \tag{5}$$

$p_r(x)$ is the probability density function (PDF) of the real class, $p_f(x)$ be the PDF of the deepfake class.

As illustrated in Figure 2 (solid lines), a low-rank image $\mathbf{I}_{\text{low}}$ is derived from the input image $\mathbf{I}^{\text{input}}$. This low-rank image is then fed into the trained U-Net VAE model, which generates a reconstructed image $\hat{\mathbf{I}}$. The reconstruction loss $\mathcal{L}_{\text{recon}}$ is computed between $\mathbf{I}^{\text{input}}$ and $\hat{\mathbf{I}}$ using the following equation:

$$\mathcal{L}_{\text{recon}} = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\mathbf{I}_{i,j}^{\text{input}} - \hat{\mathbf{I}}_{i,j}\right)^2 . \tag{6}$$

Where $\mathbf{I}_{i,j}^{\text{input}}$ and $\hat{\mathbf{I}}_{i,j}$ are the pixel values at i^{th} row and j^{th} column index for images $\mathbf{I}^{\text{input}}$ and $\hat{\mathbf{I}}$ respectively. Based on $\mathcal{L}_{\text{recon}}$ and the optimal threshold τ^*, images are classified as real or deepfake. Specifically, if $\mathcal{L}_{\text{recon}}$ exceeds τ^*, the image is classified as a deepfake; otherwise, it is considered real.

III. Experiments and Results

A. Experimental Setup

Datasets We evaluate our approach on multiple deepfake datasets: FaceForensics++ (FF++) [16], DeepfakeDetection (DFD) [4], Deepfake Detection Challenge (DFDC) [7], preview version of DFDC (DFDCP) [8], and CelebDF (CDF) [14]. FF++ consists of more than 1.8 million forged images generated using Deepfakes (DF) [5], Face2Face (F2F) [20], FaceSwap (FS) [9], and NeuralTexture (NT) [21]. We adopt the c23 (lightly compressed) version for comparison. The dataset split follows the standard protocol established by DeepfakeBench [26]. Frame-level Area Under Curve of the Receiver Operating Characteristic (ROC-AUC) is used as the evaluation metric.

Training Details: We employ a 6-layer U-Net VAE trained for 15 epochs with a batch size of 128, using an Adam optimizer (weight decay = 0.5) and a learning rate of 0.001. The hyper parameters β and λ of the $\mathcal{L}_{\text{total}}$ (in Equation 3) are set to 0.6.

B. Results

Generalization Evaluation: Following a common generalization protocol, our model is trained using a dataset (e.g., FF++ (c23) [16]) and tested on other unseen datasets such as CDF [14] and DFDC [7], etc. Unlike previous works that vary in preprocessing and evaluation settings, we standardize our experiments using DeepfakeBench [26] to ensure fair comparisons. Here, we group prior SOTA methods into three categories: *naive* (direct classifiers without explicit interpretability), *spatial* (methods using visual/semantic features), and *frequency* (methods leveraging spectral artifacts). As we see from Table I, our approach consistently achieves superior ROC-AUC scores compared to the SOTA approaches. This indicates better generalization capabilities, with an average increase of 6% to 29% compared to the previous methods.

Impact of Threshold on ROC-AUC: The choice of threshold significantly affects the performance of deepfake detection. To analyze this, we vary the threshold and compute the corresponding ROC-AUC scores on the test set of different datasets. The optimal threshold (τ^*) used for the evaluation is determined from the validation set, which is held exclusively from the training set to ensure unbiased selection.

In Figure 3 we show the performance variations of the model trained on FF++ and tested on other datasets across different thresholds. The optimal threshold (τ^*) was calculated on the FF++ validation set. It can be seen that, while varying the threshold within 5% of τ^*, the ROC-AUC on the other datasets remains stable, demonstrating the stability of our approach.

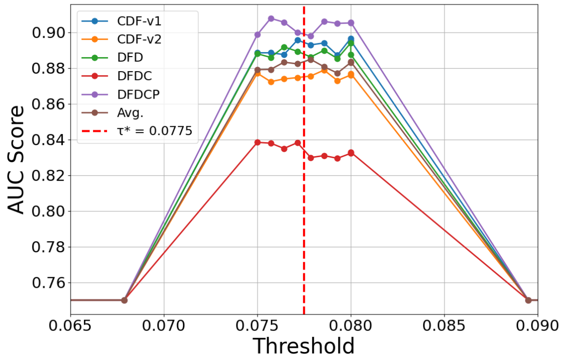

Fig. 3: Effect of threshold variation on ROC-AUC. The optimal threshold (τ^*) is selected based on ROC analysis using a validation set held out from FF++.

Cross-Manipulation Performance: We analyze the robustness of our model by training it on one type of manipulation (e.g., DF [5]) and testing it on other types of manipulations like F2F [20], FS [9] and NT [21] manipulations. Table II presents the results, showing that our method performs comparable to previous augmentation-based approaches such as Face X-ray + BI [26] , PCL + I2G [26], and EFNB4 + SBI [26]. Despite lacking explicit augmentation strategies, our model generalizes well across different deepfake techniques, demonstrating its effectiveness in detecting deepfakes for different datasets.

Robustness Evaluation: Figure 4 presents the performance of our model under different image degradations, specifically Gaussian blur and JPEG compression, evaluated at five different perturbation levels as defined by Jiang et al. [10]. Our method performs better EFNB4 + SBI [26] in the case of Gaussian blur and is comparable in the case of JPEG compression. Our method performs significantly better than Face X-ray [26] and FWA [26] in both scenarios. This resilience demonstrates that variations in input image quality have minimal

[2] Accuracies are not compatable due to different testing protocol.

Method	Detector	Backbone	CDF-v1 [14]	CDF-v2 [14]	DFD [4]	DFDC [7]	DFDCP [8]	Avg.
Naive	Meso4 [26]	MesoNet	0.736	0.609	0.548	0.556	0.599	0.610
Naive	MesoIncep [26]	MesoNet	0.737	0.697	0.623	0.576	0.684	0.663
Naive	CNN-Aug [26]	ResNet	0.742	0.703	0.646	0.636	0.617	0.669
Naive	Xception [26]	Xception	0.791	0.739	0.816	0.680	0.737	0.753
Naive	EfficientB4 [26]	EfficientNet	0.791	0.749	0.815	0.696	0.728	0.756
Naive	Detect and Locate^2 [23]	Xception	0.706	-	0.762	0.633	-	0.630
Spatial	CapsuleNet [26]	Capsule	0.791	0.747	0.684	0.647	0.657	0.705
Spatial	FWA [26]	Xception	0.719	0.710	0.667	0.638	0.690	0.685
Spatial	Face X-ray [26]	HRNet	0.709	0.679	0.766	0.633	0.694	0.696
Spatial	FFD [26]	Xception	0.780	0.748	0.780	0.734	0.753	0.759
Spatial	CORE [26]	Xception	0.780	0.743	0.802	0.743	0.753	0.754
Spatial	Recce [26]	Custom	0.768	-	0.812	0.713	0.734	0.752
Spatial	UCF [26]	Xception	0.779	-	0.810	0.759	0.763	0.778
Frequency	F3Net [26]	Xception	0.777	0.735	0.798	0.702	0.735	0.749
Frequency	SPSL [26]	Xception	0.815	0.726	0.804	0.741	0.761	0.769
Frequency	SRM [26]	Xception	0.793	0.755	0.812	0.704	0.741	0.760
Frequency	EFNB4 + LSDA [24]	EfficientNet	0.867	0.830	0.880	0.736	0.815	0.826
SVD (Ours)	U-Net VAE	U-Net VAE	**0.892** (+0.025)	**0.876** (+0.046)	**0.890** (+0.010)	**0.834** (+0.098)	**0.903** (+0.088)	**0.881** (+0.055)

TABLE I: Cross-dataset evaluations using the **frame-level ROC-AUC** metric on the deepfake benchmark [26]. All detectors are trained on FF++-c23 [16] and evaluated on other datasets. The best results are highlighted in **bold** and the second best are underlined. The increment of ROC-AUC value is given in blue.

Method	DF [5]	F2F [20]	FS [9]	NT [21]
Face X-ray + BI [26]	0.9917	0.9857	0.9821	0.9813
PCL + I2G [26]	1.0	0.9897	0.9986	0.9765
EFNB4 + SBIs [26]	0.9999	0.9988	0.9991	0.9879
Ours	0.9975	0.9798	0.9934	0.9852

TABLE II: Cross-Manipulation Evaluation on FF++. This table presents the cross-manipulation performance of various methods when trained on DF and tested on F2F, FS, and NT manipulations.

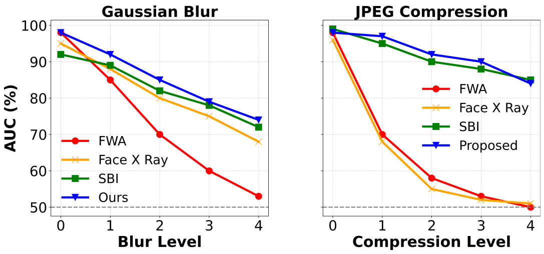

Fig. 4: Robustness to unseen Perturbations—Frame-level ROC-AUC (%) across different degradation levels. The left plot represents Gaussian Blur, while the right plot corresponds to JPEG compression. Our method (blue) outperforms other approaches (red, yellow, green).

impact on the effectiveness of our model, further strengthening its robustness against real-world distortions. Here, the images are taken from the FF++ [16] dataset.

Separability of Real vs Deepfake using τ^*: As illustrated in Figure 5, the reconstruction loss $\mathcal{L}_{\text{recon}}$ for real and deepfake images is plotted across different test datasets. The threshold (τ^*) is determined from the FF++ [16] validation dataset and is consistently applied to other test datasets, including CelebDF [14], DFDC [7], and DFD [4]. The loss values form distinct clusters for real and deepfake images across all datasets, with minimal misclassification. This demonstrates the strong generalizability of our approach across different datasets.

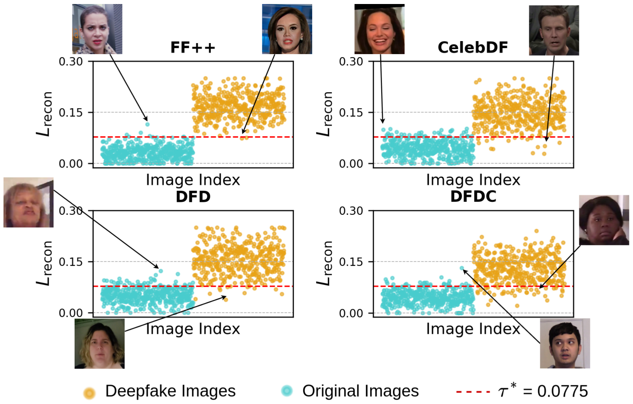

Fig. 5: Visualization of Reconstruction Loss ($\mathcal{L}_{\text{recon}}$) of real and deepfake images for different test datasets like FF++, CelebDF, DFD, DFDC. The model is trained on FF++ data and optimal threshold (τ^*) is calculated on FF++ validation dataset. The clear separation of real vs deepfake samples using τ^* demonstrates the model's generalization capability.

Visual Explanation of Reconstruction Loss: Figure 6 provides a visual comparison of Reconstruction Loss ($\mathcal{L}_{\text{recon}}$) for real and deepfake images. The upper row presents the input images, while the second row displays the corresponding normalized $\mathcal{L}_{\text{recon}}$ in the form of a heatmap. This heatmap

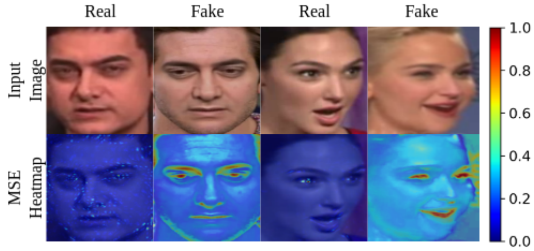

Fig. 6: Comparison of $\mathcal{L}_{\text{recon}}$ between real and deepfake images. The top row presents the input images, while the second row illustrates the Reconstruction MSE heatmap. The corresponding heatmap scale is provided alongside the diagram for reference.

illustrates the pixel-wise MSE between the generated and original images. In particular, for deepfake images, the model struggles to reconstruct critical facial features such as the eyes and lips, as evidenced by the high MSE in these areas. In contrast, real images exhibit minimal reconstruction loss. The key observation is that $\mathcal{L}_{\text{recon}}$ for deepfake images is substantially higher than that for real images supporting our idea of using it for detecting deepfakes.

IV. Conclusion

The proposed approach demonstrates robust deepfake detection performance, generalization across multiple datasets and manipulation techniques. Extensive experiments on FaceForensics++ [16], CelebDF [14], DFDC [7], and other benchmark datasets has shown consistent and superior detection performance, outperforming SOTA models in cross-dataset evaluations. Additionally, cross-manipulation analysis confirms the adaptability of our approach across different deepfake techniques without requiring explicit augmentation strategies. Further, our robustness evaluation highlights the stability of our approach against real-world distortions such as Gaussian blur and JPEG compression. These results reinforce the reliability and effectiveness of our approach in detecting deepfakes under diverse conditions. Future works can extend our approach to detect deepfakes in video data considering temporal information.

References

[1] Sara Abdali, M. Alex Vasilescu, and Evangelos E. Papalexakis. Deepfake representation with multilinear regression. *MIS2-KDD 2021 : The Second International MIS2 Workshop: Misinformation and Misbehavior Mining on the Web-2021*.

[2] Christopher M. Bishop. Pattern recognition and machine learning. 2006.

[3] Hao Dang, Feng Liu, Joel Stehouwer, Xiaoming Liu, and Anil K. Jain. On the detection of digital face manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern recognition*, pages 5781–5790, 2020.

[4] Deepfakedetection. https://ai.googleblog. com / 2019 / 09 / contributing - data - to - deepfakedetection . html. 2021.

[5] DeepFakes. www.github.com/deepfakes/ faceswap. 2020.

[6] Chris Ding and Jieping Ye. Two-dimensional singular value decomposition (2dsvd) for 2d maps and images. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, pages 32–43, 2005.

[7] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset. arXiv preprint, 2020.

[8] Brian Dolhansky, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) preview dataset. arXiv preprint, 2019.

[9] FaceSwap. www . github . com / marekkowalski / faceswap. 2021.

[10] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In Proceedings of the IEEE/CV, editor, *Mesonet*, pages 1–7, IEEE, 2018. In 2018 IEEE international workshop on information forensics and security (WIFS).

[11] Ali Khodabakhsh, Raghavendra Ramachandra, Kiran Raja, Pankaj Wasnik, and Christoph Busch. Fake face detection methods: Can they be generalized? In *2018 international conference of the biometrics special interest group (BIOSIG), IEEE*, pages 1–6, 2018.

[12] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[13] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5001–5010, 2020.

[14] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3207–3216, 2020.

[15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[16] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. Faceforensics++: Learning to detect manipulated facial images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1–11. IEEE, October 2019.

[17] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1–11, 2019.

[18] Rowayda A. Sadek. Svd based image processing applications: state of the art, contributions and research challenges. arXiv preprint, 2012.

[19] Jingxiang Sun, Xuan Wang, Yong Zhang, Xiaoyu Li, Qi Zhang, Yebin Liu, and Jue Wang. Fenerf: Face editing in neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7672–7682, 2022.

[20] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016*, 5, 2016. 2.

[21] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *Acm Transactions on Graphics (TOG)*, 38(4):1–12, 2019.

[22] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[23] Okan Kopuklu Tursun, Oğuzhan Akar, and Gerhard Rigoll. Detect and locate: Exposing face manipulation by semantic- and noise-level telltales. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2025–2035, 2022.

[24] Zhiyuan Yan, Yuhao Luo, Siwei Lyu, Qingshan Liu, and Baoyuan Wu. Transcending forgery specificity with latent space augmentation for generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8984–8994, 2024.

[25] Zhiyuan Yan, Jiangming Wang, Peng Jin, Ke-Yue Zhang, Chengchun Liu, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Orthogonal subspace decomposition for generalizable ai-generated image detection, 2025.

[26] Zhiyuan Yan, Yong Zhang, Xinhang Yuan, Siwei Lyu, and Baoyuan Wu. Deepfakebench: A comprehensive benchmark of deepfake detection. arXiv preprint, 2023.

[27] W.J. Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.

[28] Tianchen Zhao, Xiang Xu, Mingze Xu, Hui Ding, Yuanjun Xiong, and Wei Xia. Learning self-consistency for deepfake detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15023–15033, 2021.