Contrastive Pseudo Learning for Open-World DeepFake Attribution Supplementary Material

Zhimin Sun[1,2,3*] Shen Chen[2*] Taiping Yao[2] Bangjie Yin[2]
Ran Yi[1†] Shouhong Ding[2†] Lizhuang Ma[1]
[1]Shanghai Jiao Tong University [2]Tencent YouTu Lab
[3]Shanghai Key Laboratory of Computer Software Testing & Evaluating

A. Comparison of OW-DFA with other Tasks

We summarize similarities and differences between OW-DFA and related tasks in Table 1.

Comparison with GAN attribution. GAN attribution [34, 35, 33, 12] is a multi-classification task focusing on attributing GAN models. A common strategy is to use the fingerprints of different GAN models to attribute those generated images. However, they only consider the close-world scenario where the training and test sets have the same category distribution. Such an assumption is not applicable in OW-DFA, since novel forgeries emerge greatly under open-world scenarios.

Comparison with OW-GAN attribution. OW-GAN attribution is a multi-classification task that focuses on attributing GAN models and discovering unseen GANs in an open-world scenario, which is proposed by Open-world GAN [11]. Although some progress has been made in open-world scenarios, the fingerprint assumption it relied on may not hold in the fake faces generated by non-GAN methods. Besides GAN methods, OW-DFA also covers other forgery types, including *identity swap* and *expression transfer*, making the task more realistic and challenging.

Comparison with Deepfake Detection. Deepfake detection focuses on real/fake detection, and many related works [25, 38] have been proposed in recent years. However, the generalization performance on novel attacks is still limited. As fake faces become visually realistic and need to be interpreted in legal proceedings, OW-DFA extends the binary detection task to a multi-classification task for enhancing the interpretability of deepfake detection. At the same time, the additional provision of unlabeled novel attack data also provides a higher possibility for further improvement of generalizability.

*Equal contribution. This work was done when Zhimin Sun was a research intern at Tencent YouTu Lab.

†Corresponding authors.

B. Pre-processing Details of Datasets

We present the five datasets that are used in our OW-DFA benchmark and describe the detail of data processing for each dataset.

- **FaceForensics++** [27] is the most widely used dataset for deepfake detection tasks, consisting of 1,000 original video sequences that have been manipulated with 4 face manipulation methods, including Deepfakes, Face2Face, FaceSwap, and NeuralTextures. As part of the data for OW-DFA, we include both real and fake images from FF++. We sample 20 frames for each manipulated video and 200 frames for each original video. After that, we use dlib to crop out the faces from those frames and save them as new images.
- **CelebDF** [23] is a challenging dataset for deepfake detection. It consists of 590 celebrity videos (Celeb-real) and 300 additional videos (YouTube-real) downloaded from YouTube, as well as 5,639 high-quality synthesized videos. The inclusion of real celebrity videos in CelebDF makes it suitable for evaluating the OW-DFA benchmark under Protocol-2, which requires distinguishing between real and fake images from different sources. We sample 100 frames for each Celeb-real video and use dlib to crop the faces at the same time.
- **ForgeryNet** [15] is the largest publicly available multi-purpose deep face forgery analysis benchmark dataset. It contains 2.9 million images and 15 forgery methods. Due to its large scale and diverse range of attack types, ForgeryNet is the most suitable dataset for deepfake attribution tasks. A significant portion of the data in the OW-DFA benchmark is obtained from ForgeryNet. For each forgery method in Protocol-1, we extract 20,000 frames and apply dlib to ensure data consistency.
- **DFFD** [10] is a diverse deepfake face dataset that contains 600,000 face images. Of these images, 500,000

Table 1. Relationship between our novel OW-DFA and related tasks.

Task	Task Goal	Data Type	Known Classes	Novel Classes
Deepfake Detection	Classification of real/fake faces	Deepfake	✓	-
GAN Attribution	Classification of GAN images	GAN-generated	✓	-
Open-world GAN Attribution	Classification of GAN images	GAN-generated	✓	✓
Open-world Semi-Supervised Learning	Classification of object	Various object images	✓	✓
Open-world DeepFake Attribution	Classification of deepfake faces	Deepfake	✓	✓

Table 2. List of forgery methods and corresponding train/test splits used in **Protocol-1** and **Protocol-2**. Note that some train images are unlabeled.

Face Type	Source Dataset	Method	# of Train	# of Test
Identity Swap	FaceForensics++	FaceSwap	1200	300
		Deepfakes	1600	400
	ForgeryNet	FaceShifter	1200	300
		DeepFaceLab	1600	400
		FSGAN	1200	300
Expression Transfer	FaceForensics++	Face2Face	1600	400
		NeuralTextures	1200	300
	ForgeryNet	Talking-Head-Video	1200	300
		ATVG-Net	1200	300
		FOMM	1600	400
Attribute Manipulation	ForgeryNet	MaskGAN	1600	400
		StarGAN2	1200	300
		SC-FEGAN	1200	300
	DFFD	FaceAPP	1600	400
		StarGAN	1200	300
Entire Face Syncthesis	ForgeryNet	StyleGAN2	1200	300
	DFFD	PGGAN	1200	300
		StyleGAN	1600	400
	ForgeryNIR	CycleGAN	1600	400
		StyleGAN2	1200	300
Real Face	FaceForensics++	Youtube-Real	16000	4000
	CelebDFv2	Celeb-Real	4000	1000

are synthetic or manipulated and 100,000 are real. The images originate from various publicly accessible datasets and are synthesized or manipulated using publicly accessible methods. Owing to its diversity of attack types and inclusion of data on attribute manipulation and entire face synthesis, DFFD is incorporated into the OW-DFA benchmark. For FaceAPP and GAN generation attacks, we randomly select 20,000 images for each method.

- **ForgeryNIR** [32] is a near-infrared face forgery and detection dataset that contains over 50,000 real and fake identities. It also includes various perturbations to simulate real-world scenarios. Since the fake images in ForgeryNIR are generated using multiple GAN techniques, we randomly select 20,000 images for both CycleGAN and StyleGAN2 and include them in OW-DFA.

Train and test splits. We download all datasets from the official links. We select images according to **Protocol-1** (20 manipulation methods) and **Protocol-2** (20 manipulation methods and 2 real face types). Then, we randomly sample images according to the corresponding number of each forgery attack method. Train and test sets are split based on the ratio of $4:1$. Table 2 summarizes the class-wise train and test splits used in Protocol-1 and Protocol-2. Note that some train images are unlabeled. Protocol-1 covers 20 forgery methods and includes a total of 272,000 training images and 68,000 test images. Protocol-2 covers both 2 real face and 20 forgery methods and includes a total of 472,000 training images and 118,000 test images.

C. Implementation for Multi-stage Paradigm

To further improve the performance of the OW-DFA task, we extend CPL to a multi-stage paradigm with a pre-training technique and iterative learning. Here we also provide the specific implementation details of different stages.

- **Stage-1** aims to pre-train on the labeled dataset to improve the performance on known attacks. Specifically, we conduct supervised training based on the labeled data in OW-DFA using Eq. 11 in the main text as the loss function with a learning rate of $2e^{-4}$ for 20 epochs. After completing Stage-1 training, we obtain a weight that performs well on known attacks and can be used as the pretrained weight for Stage-2.

- **Stage-2** aims to leverage the unlabeled data to enhance the robustness and generalization of the model. We initialize the model with the pretrained weights from Stage-1 and apply CPL on both labeled and unlabeled sets in a semi-supervised manner. We use Eq. 13 in the main text as the loss function with a learning rate of $2e^{-4}$ for 50 epochs. We also ensure a strict half-sampling of labeled and unlabeled data in a batch, maintaining a balanced ratio of the two types of data during training.

- **Stage-3** aims to exploit the clustering structure of the feature space and assign more accurate labels to the unlabeled data. We leverage the Semi-Supervised k-means algorithm [30] and iterative learning to further refine the pseudo-labels and fine-tune the model. We first extract features of all training samples, both labeled and unlabeled, with the feature extractor in Stage-2. Next, we set up initial clustering centers with 10 samples by K-Means++. Then, Semi-Supervised k-means (refer to Figure 4 in [30]) will be repeated

for at most 100 iteration times until the k-means algorithm converges with a tolerance of $1e^{-4}$. After obtaining pseudo-labels with assigned clusters, we further fine-tune our models using Eq. 11 in the main text as the loss function with a learning rate of $2e^{-4}$ for 20 epochs.

D. Implementation for Comparison Methods

Our baseline comparison includes a total of 8 methods, comprising GAN attribution and OW-SSL methods. To ensure a fair comparison between methods, we use the actual number of categories as the output head number for the classifier. We exclude strong and weak augmentation strategies due to their inapplicability to the OW-DFA task. All methods use ResNet50 pre-trained on ImageNet as the feature extractor. It is trained with a learning rate of $2e^{-4}$ for 50 epochs and a batch size of 128.

- **Lower bound** is established using supervised learning on labeled data. Since this experiment applies to a closed-world setting, we obtain the output result based on its original classifier and evaluate its performance directly.

- **Upper bound** is established using supervised learning on all data, including both labeled and unlabeled data. Since this experiment is trained with all types of samples exposed, its performance must be optimal.

- **DNA-Det** [33] is a closed-set approach that attributes GAN-generated images based on GAN fingerprints. We include classification loss, contrastive loss, and automatic weighted loss with default configurations.

- **Open-World GAN** [11] is an approach that discovers and attributes GAN-generated images based on an open-world setting. We config the class lists of both protocols and repeat iteration for 4 times according to the default configuration. We extend evaluation to an additional test set and report results on this extra set.

- **RankStats** [14] is a novel class discovery method that can be extended to solve OW-SSL tasks by exploring Top-K ranked dimensions of features. Sample pairs can be pulled or pushed based on their similarity. We use the default setting of $K = 5$ as the number of ranked dimensions.

- **ORCA** [6] is the first approach to propose the task of OW-SSL and uses cosine distance as a similarity matrix to bring pairs with high similarity closer together. We reproduce ORCA with both a fixed negative margin and a dynamic margin and report the best result with a fixed negative margin of $m = -0.2$.

- **OpenLDN** [26] uses a bi-level optimization rule to enhance feature representation and applies close-world iterative training to improve performance. However, we only evaluate its performance using its semi-supervised feature learning component. We change the backbone to ResNet50 while keeping the configuration of simnet unchanged, and use 0.5 as the default threshold for pseudo-label assignment.

- **NACH** [13] is a recently introduced approach that improves ORCA's performance by filtering out erroneous samples and synchronizing the learning pace between seen and unseen classes. We use the default setting of $K = 2$ as the index for the labeled sample when filtering pairs.

E. Implementation for Experiments

Due to space limitations in the main text, we have omitted some experimental details. In this section, we provide additional explanations for the specific implementation of those experiments.

Implementation Details of the GLVM ablation study. To fairly compare the strengths and weaknesses of different methods in similarity learning, we compare the accuracy of similarity pair matching at various training stages. We use the ground truth of unlabeled samples to distinguish between known and novel classes. To visually represent this selection process, we calculate the accuracy of sample pairing and present it as a line chart. Further validation results for each forgery method can be found in Figure 2.

Implementation Details of the PPLM ablation study. To ensure an equitable comparison between methods, we exclude strong and weak augmentation strategies due to their inapplicability to the OW-DFA task. Since the pseudo-label strategy relies on prior similarity learning, we use the GLV loss constraint as a baseline to ensure that the feature extractor and classifier have some ability to classify novel classes. In our comparison of all methods, we uniformly use a weight of 0.5 for the pseudo-label cross-entropy loss. Directly assigning labels refers to the strategy of choosing the prediction with the highest output value as the label. For the fixed-threshold approach, we use a threshold of 0.95 for both known and novel classes and only assign labels to predictions that exceed this threshold. For dynamic-threshold approaches [36, 31], we reproduce them using their open-source code and default configuration. For ST Gumbel Softmax, we directly use the output of Gumbel Softmax as the label with a default temperature of $\tau = 1$.

Implementation Details of Real/Fake Detection. To verify the importance of deepfake attribution for deepfake detection, we compare the performance of the deepfake detection task based on Protocol-2. We compare the results

Table 3. List of methods and corresponding datasets utilized in OW-DFA with $5\times$ scale.

Face Type	Labeled Sets	Unlabeled Sets	Source Dataset	Method	Tag	Labeled #	Unlabeled #
Identity Swap	Deepfakes [2] DeepFaceLab [1]	Deepfakes DeepFaceLab FaceSwap [4] FaceShifter [22] FSGAN [24]	FaceForensics++ [27]	Deepfakes FaceSwap	Known Novel	7500 -	2500 7500
			ForgeryNet [15]	DeepFaceLab FaceShifter FSGAN	Known Novel Novel	7500 - -	2500 7500 7500
Expression Transfer	Face2Face [29] FOMM [28]	Face2Face FOMM NeuralTextures [5] Talking-Head-Video [37] ATVG-Net [7]	FaceForensics++	Face2Face NeuralTextures	Known Novel	7500 -	2500 7500
			ForgeryNet	FOMM ATVG-Net Talking-Head-Video	Known Novel Novel	7500 - -	2500 7500 7500
Attribute Manipulation	MaskGAN [21] FaceAPP [3]	MaskGAN FaceAPP StarGAN2 [9] SC-FEGAN [16] StarGAN [8]	ForgeryNet	MaskGAN StarGAN2 SC-FEGAN	Known Novel Novel	7500 - -	2500 7500 7500
			DFFD [10]	FaceAPP StarGAN	Known Novel	7500 -	2500 7500
Entire Face Synthesis	StyleGAN [18] CycleGAN [39]	StyleGAN CycleGAN PGGAN [17] StyleGAN2 [19]	ForgeryNet	StyleGAN2	Novel	-	7500
			DFFD	StyleGAN PGGAN	Known Novel	7500 -	2500 7500
			ForgeryNIR [32]	CycleGAN StyleGAN2	Known Novel	7500 -	2500 7500
Real Face	Youtube-Real [27]	Celeb-Real [23]	FaceForensics++	Youtube-Real	Known	75000	25000
			CelebDFv2 [23]	Celeb-Real	Novel	-	25000

Table 4. Benchmark Evaluation on **Protocol-1** and **Protocol-2** with dataset of $5\times$ scale.

Method	Protocol-1: Fake							Protocol-2: Real & Fake						
	Known	Novel			All			Known	Novel			All		
	ACC	ACC	NMI	ARI	ACC	NMI	ARI	ACC	ACC	NMI	ARI	ACC	NMI	ARI
Lower Bound	**99.68**	40.86	47.55	26.33	46.91	63.43	37.33	**99.84**	34.57	42.98	19.37	61.46	66.05	62.16
Upper Bound	98.93	96.99	94.18	94.94	97.91	95.87	95.91	99.27	97.12	94.89	96.78	98.43	96.48	98.27
RankStats [14]	99.17	62.05	64.60	52.87	79.52	78.87	72.90	98.86	51.19	57.56	37.56	78.25	77.37	88.07
ORCA [6]	98.30	73.61	70.20	63.50	85.23	83.99	80.86	97.09	62.10	64.96	49.15	83.44	82.68	88.64
OpenLDN [26]	98.78	54.12	57.54	45.43	72.90	77.22	70.03	97.03	48.26	52.77	33.72	73.97	75.13	84.37
NACH [13]	98.34	73.43	71.61	65.33	85.16	84.90	82.31	97.28	69.39	70.03	54.28	86.47	84.76	90.09
CPL	98.68	**75.21**	**73.19**	**65.71**	**86.25**	**85.58**	**82.35**	97.45	**69.57**	**70.67**	**54.67**	**86.51**	**85.44**	**90.30**

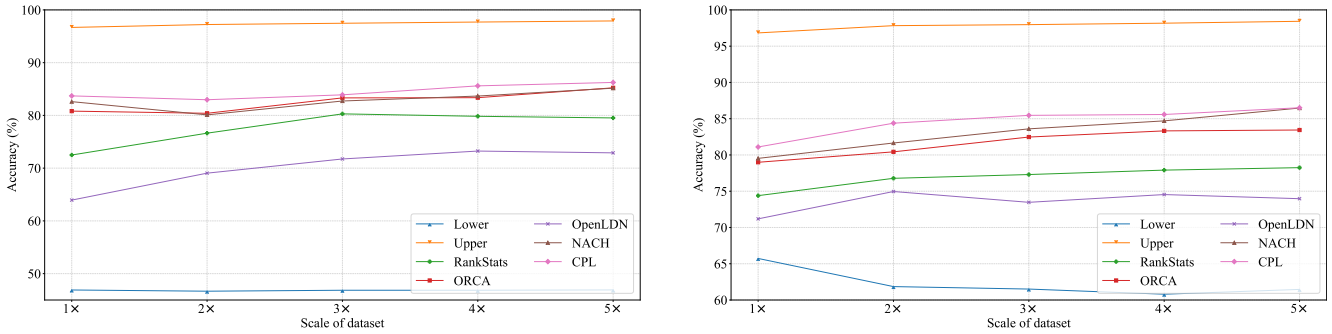
Table 5. Ablation study of patch division on **Protocol-1**.

Patch	Known	Novel			All		
	ACC	ACC	NMI	ARI	ACC	NMI	ARI
3×3	**97.50**	**71.89**	**68.20**	**59.37**	**83.70**	**82.31**	**77.64**
5×5	96.80	69.66	66.35	55.25	82.41	81.20	75.56
7×7	96.68	67.13	64.70	52.88	81.12	80.93	75.15

of three approaches: a) Deepfake binary classification, b) Deepfake multi-classification, and c) CPL framework. **a) Deepfake binary classification** is trained on the labeled set and outputs 0/1 to represent fake/real. When testing, we directly evaluate the performance based on the AUC result. **b) Deepfake multi-classification** is trained on the labeled set with 9 classifier outputs representing 1 real face and 8 forgery methods. Since there is only one real face type, we directly evaluate the AUC results using predicted output when testing. **c) The CPL framework** is trained on both labeled and unlabeled sets using semi-supervised learning with 22 classifier outputs representing 2 real faces and 20 forgery methods. Since multiple real face types appear, we first acquire the mapping relationship between prediction results and ground truth labels using the Hungarian algorithm [20]. Then during testing, we sum up all prediction results for real faces to evaluate AUC results.

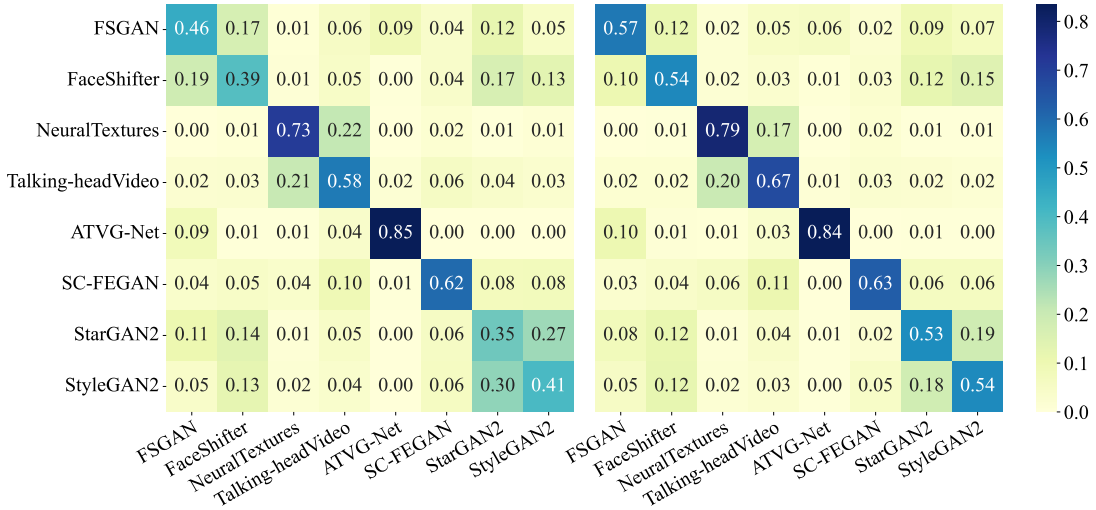
F. Additional Experiments

Ablation Study on Scale of Dataset. To assess the scalability of each method further, we conduct an additional ex-



(a) Result on **Protocol-1** (b) Result on **Protocol-2**

Figure 1. Study of the relation between the performance of different methods and the scale of the dataset.



(a) Confusion Matrix with GR loss (b) Confusion Matrix with GLV loss

Figure 2. This confusion matrix displays the correct ratio of sample pairing using (a) GR loss and (b) GLV loss. The X-axis represents the actual forgery method, while the Y-axis represents the predicted forgery method.

periment to evaluate the performance of different methods on datasets of varying sizes. Due to the limited performance of DNA-Det [33] and Openworld-GAN[11] in the OW-DFA task, we exclude these two methods from this experiment. Specifically, based on our original dataset, we scale up both Train and Test set to $2\times \sim 5\times$ their original size. The results of the ablation study are shown in Figure 1. As expected, the performance of each method improves to some extent as the size of the dataset increases. However, our proposed method CPL consistently achieves the best results across all dataset sizes. We provide the specific settings for the $5\times$ scale of dataset in Table 3, and the corresponding evaluation results are presented in Table 4.

Confusion Matrix for Different Forgery Methods. We record the predicted result and actual label of samples during similarity learning to analyze factors that contribute to ineffective classification. We present this information using a confusion matrix in Figure 2. To focus on categories with high confusion, we filter out all categories with prediction accuracy $>90\%$ and only include methods with low classification results. The method with GLR loss constraint can reduce confusion between similar categories while obtaining more accurate predictions. It has an accuracy of $>50\%$ on all categories. However, some samples are still confused with each other, especially when a) their data source is the same, such as StarGAN2, FaceShifter, and StyleGAN2, or when b) they belong to the same forgery type, including the confusion of NeuralTextures and Talking-Head-Video, and that of FaceShifter and FSGAN.

Ablation Study on Patch Division. To compare the performance of different patch sizes and evaluate their impact on overall performance, we conduct an ablation study on patch division. Table 5 presents the results of the ablation study, which show that the optimal performance is achieved with a smaller number of patch splits of 3×3. Specifically, we observe that using a smaller grid for local region partitioning can alleviate the problem of the same forged region being sliced into different local patches.

References

[1] Deepfacelab. https://github.com/iperov/DeepFaceLab. Accessed: 2023-2-28. 4

[2] Deepfakes. https://github.com/deepfakes/faceswap. Accessed: 2023-2-28. 4

[3] Faceapp. https://faceapp.com/app. Accessed: 2023-2-28. 4

[4] Faceswap. https://github.com/MarekKowalski/FaceSwap/. Accessed: 2023-2-28. 4

[5] Neuraltextures. https://github.com/SSRSGJYD/NeuralTexture. Accessed: 2023-2-28. 4

[6] Kaidi Cao, Maria Brbic, and Jure Leskovec. Open-world semi-supervised learning. In *International Conference on Learning Representations*, 2022. 3, 4

[7] Lele Chen, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7832–7841, 2019. 4

[8] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 4

[9] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 4

[10] 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. 1, 4

[11] Sharath Girish, Saksham Suri, Sai Saketh Rambhatla, and Abhinav Shrivastava. Towards discovery and attribution of open-world gan generated images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14094–14103, 2021. 1, 3, 5

[12] Luca Guarnera, Oliver Giudice, Matthias Nießner, and Sebastiano Battiato. On the exploitation of deepfake model recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 61–70, 2022. 1

[13] Lan-Zhe Guo, Yi-Ge Zhang, Zhi-Fan Wu, Jie-Jing Shao, and Yu-Feng Li. Robust semi-supervised learning when not all classes have labels. In *Advances in Neural Information Processing Systems*, 2022. 3, 4

[14] Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Automatically discovering and learning new visual categories with ranking statistics. In *International Conference on Learning Representations (ICLR)*, 2020. 3, 4

[15] Yinan He, Bei Gan, Siyu Chen, Yichun Zhou, Guojun Yin, Luchuan Song, Lu Sheng, Jing Shao, and Ziwei Liu. Forgerynet: A versatile benchmark for comprehensive forgery analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4360–4369, 2021. 1, 4

[16] Youngjoo Jo and Jongyoul Park. Sc-fegan: Face editing generative adversarial network with user's sketch and color. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 4

[17] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 4

[18] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 4

[19] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *Proc. CVPR*, 2020. 4

[20] HW Kuhn et al. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955. 4

[21] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4

[22] Lingzhi Li, Jianmin Bao, Hao Yang, Dong Chen, and Fang Wen. Faceshifter: Towards high fidelity and occlusion aware face swapping. *arXiv preprint arXiv:1912.13457*, 2019. 4

[23] 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. 1, 4

[24] Yuval Nirkin, Yosi Keller, and Tal Hassner. FSGAN: Subject agnostic face swapping and reenactment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7184–7193, 2019. 4

[25] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII*, pages 86–103. Springer, 2020. 1

[26] Mamshad Nayeem Rizve, Navid Kardan, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Openldn: Learning to discover novel classes for open-world semi-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXI*, pages 382–401. Springer, 2022. 3, 4

[27] 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. 1, 4

[28] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019. 4

[29] 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 conference on computer vision and pattern recognition*, pages 2387–2395, 2016. 4

[30] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7492–7501, 2022. 2

[31] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Zhen Wu, and Jindong Wang. Freematch: Self-adaptive thresholding for semi-supervised learning. In *International Conference on Learning Representations*, 2023. 3

[32] Yukai Wang, Chunlei Peng, Decheng Liu, Nannan Wang, and Xinbo Gao. Forgerynir: deep face forgery and detection in near-infrared scenario. *IEEE Transactions on Information Forensics and Security*, 17:500–515, 2022. 2, 4

[33] Tianyun Yang, Ziyao Huang, Juan Cao, Lei Li, and Xirong Li. Deepfake network architecture attribution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 4662–4670, 2022. 1, 3, 5

[34] Ning Yu, Larry S Davis, and Mario Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7556–7566, 2019. 1

[35] Ning Yu, Vladislav Skripniuk, Sahar Abdelnabi, and Mario Fritz. Artificial fingerprinting for generative models: Rooting deepfake attribution in training data. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 14448–14457, 2021. 1

[36] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419, 2021. 3

[37] Sibo Zhang, Jiahong Yuan, Miao Liao, and Liangjun Zhang. Text2video: Text-driven talking-head video synthesis with phonetic dictionary. *arXiv preprint arXiv:2104.14631*, 2021. 4

[38] Hanqing Zhao, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, and Nenghai Yu. Multi-attentional deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2185–2194, 2021. 1

[39] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017. 4