MaLP: Manipulation Localization Using a Proactive Scheme
– Supplementary material –

Vishal Asnani[1], Xi Yin[2], Tal Hassner[2], Xiaoming Liu[1]
[1]Michigan State University, [2]Meta AI
[1]{asnanivi, liuxm}@msu.edu, [2]{yinxi, thassner}@meta.com

1. Implementation Details

experimental Setup and Hyperparameters We train MaLP for $150,000$ iterations with a batch size of 4. For all of the networks, we use Adam optimizer except for the transformer which uses AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay $0.5e^{-5}$ and eps $1e^{-8}$. The learning rate is $1e^{-5}$ for all networks. The constraint weights are set as: $\lambda_1 = 100, \lambda_2 = 5, \lambda_3 = 4, \lambda_4 = 25, \lambda_5 = 25, \lambda_6 = 25, \lambda_7 = 50, \lambda_8 = 15, \lambda_9 = 20, \lambda_{10} = 50$. We use a template set size of 1 and template strength as 30% unless mentioned. All experiments are conducted on one NVIDIA K80 GPU.

Network Architecture. We show the network architecture of various components of MaLP in Fig. 1. The shared network consists of 1 stem convolutional layer and 4 convolution blocks. Each convolution block consists of convolutional and batch normalization layers followed by ReLU activation. The output of the shared network is given to $\mathcal{E}_E$ and $\mathcal{E}_C$, both having the same architecture with 3 convolution blocks and 1 stem convolutional layer. We use the transformer $\mathcal{E}_T$ in the second branch of the framework where the ViT [5] architecture is adopted. The transformer consists of 6 encoder blocks, and a dropout of 0.1 is used. The features of the transformer are reshaped to the shape of the fakeness map *i.e.* $1 \times 128 \times 128$. Finally, we use a classifier $\mathcal{C}$ on the predicted fakeness maps to perform real *vs.* fake binary classification. The classifier has 8 convolution blocks, 1 stem convolutional layer, and 3 fully connected layers. We apply the ReLU activation between the layers.

GMs and dataset license information. We use a variety of face and generic GMs to show the effectiveness of MaLP. The information for all the GMs along with their training datasets, is shown in Tab. 1. We also show more visualization samples of the predicted fakeness maps by MaLP in Fig. 2- 5. All the fakeness maps are shown in ”pink” cmap for better representation. We also indicate the cosine similarity between the predicted and ground truth fakeness maps. We observe that the fakeness maps for encrypted images have minimal bright regions. However, for fake images, MaLP is able to localize the modified regions well, considering the modified attributes/GMs are unseen in training.

Table 1. List of GMs along with their training datasets

Dataset	GMs
CelebA [14]	STGAN [12], AttGAN [6], StarGAN [3], GANimation [20], CouncilGAN [16], ESRGAN [25], GDWCT [2]
CelebA-HQ [10]	SEAN [30], StarGAN-v2 [4], ALAE [19], DRGAN [22], ColorGAN [15],
Facades [23]	CycleGAN [28], BicycleGAN [29], Pix2Pix [9]
COCO [1]	GauGAN [17]
Horse2Zebra [28]	AutoGAN [27]
Summer2Winter [28]	DRIT [11]
GTA2CITY [21]	UNIT [13]
Edges2Shoes [9]	MUNIT [7]
Paris Street-view [16]	Cont_Enc [18]
Sketch-Photo [24]	DualGAN [26]

The face datasets include CelebA [14] and CelebA-HQ [10], both of which don’t have any associated Institutional Review Board (IRB) approval. The authors for both datasets mention the availability of the dataset for non-commercial research purposes, which we strictly adhere to. For generic images datasets, we use Facades [23], COCO [1], Horse2Zebra [28], Summer2Winter [28], GTA2CITY [21], Edges2Shoes [9], Paris street-view [18] and Sketch-Photo [24] datasets. All the mentioned generic image datasets can be used for non-commercial research purposes, as mentioned by the authors, and we use the datasets for the same purposes.

Image Editing Degradations. We apply several image editing degradations to the test set to verify the robustness of MaLP. The details of these operations are listed below:

1. JPEG compression: We compress the image with the compression quality of 50%.
2. Blur: We apply the Gaussian blur with a filter size of 7×7.
3. Noise: We apply a Gaussian noise with zero mean and

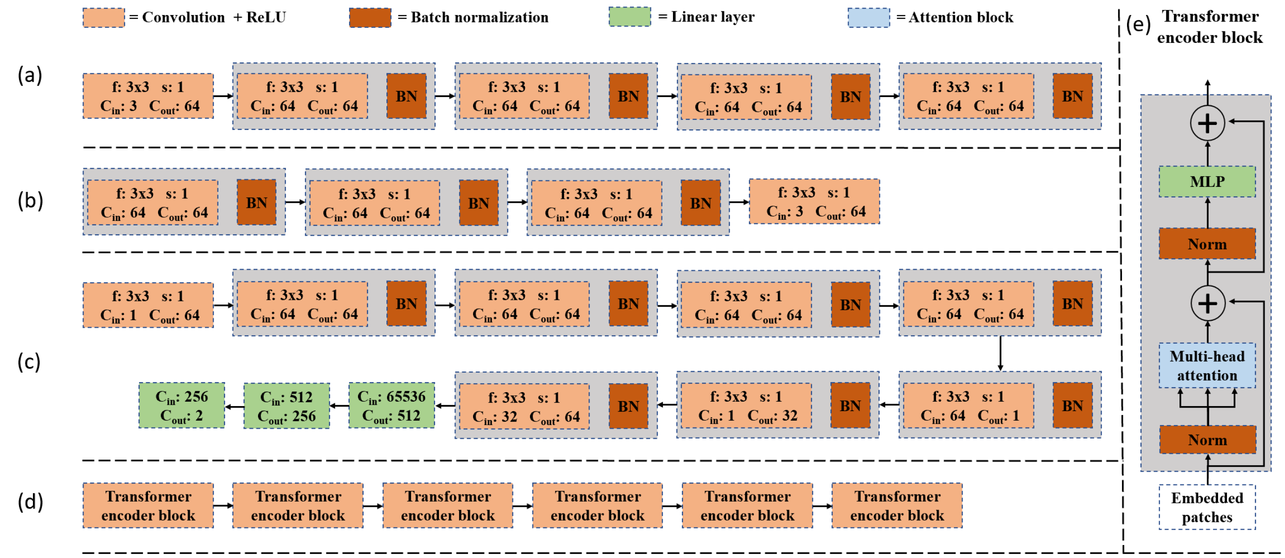

Figure 1. Network architecture for different components of MaLP. (a) Shared network, (b) Encoder $\mathcal{E}_E$ and CNN network $\mathcal{E}_C$, (c) Classifier $\mathcal{C}$, (d) Transformer $\mathcal{E}_T$, and (e) Transformer encoder block.

Table 2. Ablation for localization loss.

Loss	CS ↑	PSNR ↑	SSIM ↑
CS	0.9356	22.16	0.7114
CS + L_2	0.9230	18.98	0.6614
CS + SSIM + L_2	0.9211	19.12	0.6816
CS + SSIM + L_1	0..8777	14.01	0.3712
CS + SSIM	**0.9394**	**23.020**	**0.7312**

unit variance.

4. Low-resolution: We resize the image to half the original resolution and restore it back to the original resolution using linear interpolation.

Potential Societal Impact The problem of manipulation localization is crucial from the perspective of media forensics. Localizing the fake regions not only helps in the detection of these fake media but, in the future, can also help recover the original image that the GM has manipulated. We also show that MaLP can be used as a discriminator to improve the quality of GMs. While this is an interesting application of MaLP, it can be a possibility that the GMs become more robust to our framework, decreasing the localization performance if the training of the GM is done from scratch.

2. Additional Experiments

Localization Loss. We show the importance of manipulation loss (defined in Eq. 8) in Sec. 4.6. We perform an ablation to formulate the loss of fakeness maps for manipulated images. As shown in Tab. 2, we try experimenting with various loss functions *i.e.* cosine similarity (CS), L_1, L_2 and structural similarity index measure (SSIM). Using

Table 3. Comparison with [8] using multiple GMs in training. MaLP is able to outperform [8] by training images manipulated by only STGAN.

Method	Training GMs	Cosine similarity ↑		
		AttGAN	StarGAN	StyleGAN
Hunag *et al.* [8]	STGAN + ICGAN + PGGAN + StyleGAN + StyleGAN2 + StarGAN + AttGAN	0.6940	0.8494	0.7479
MaLP	STGAN	**0.8557**	**0.8718**	**0.8255**

Table 4. Performance of MaLP across different attribute modifications seen in training.

Method	Cosine similarity ↑					
	Bald	Bangs	Black Hair	Eyeglasses	Mustache	Smile
[8]	0.9014	0.8850	0.8817	0.9093	0.9152	0.8634
MaLP	**0.9478**	**0.9329**	**0.9367**	**0.9549**	**0.9470**	**0.9489**

just the CS loss results in better performance compared to combining it with L_1 or L_2 loss. We observe a huge deterioration in performance when using L_1 loss. This can be explained as PSNR and SSIM are directly related to mean squared error which is optimized by either an L_2 or SSIM loss. Finally, adopting an SSIM loss with CS loss results in a better performance as both of them are more related to the metrics, making it easier for MaLP to converge.

Comparison with Baseline. Due to the limited GPU memory, we conduct proactive training with one GM only because the GM needs to be loaded to the memory and used on the fly. On the other hand, passive methods can be trained on multiple GMs because the image generation processes are conducted offline. As shown in Tab. 3, [8] trains on images manipulated by 7 different GMs, unlike MaLP, which is trained on images manipulated by only 1 GM. We show the performance on three GMs, which are

Table 5. Ablation study for transformer architecture.

Optimizer	Depth	Dropout	Cosine similarity↑	Accuracy↑
Adam	6	0.1	0.8839	0.9514
AdamW	1	0.0	0.8825	0.9647
AdamW	1	0.0	0.8826	0.9680
AdamW	3	0.0	0.8830	0.9705
AdamW	6	0.1	**0.8848**	**0.9856**

seen for [8], but unseen for MaLP. MaLP performs better even though these GMs' images are not seen in training. Therefore, even though the training of MaLP is limited by 1 GM, it can achieve better generalization to other GMs proving the effectiveness of proactive schemes.

Multiple Attribute Modifications. Instead of training on bald attribute modification by STGAN, we train and test MaLP on multiple attribute modifications. These include bald, bangs, black hair, eyeglasses, mustache, and smile manipulation. We show the results in Tab. 4. MaLP performs better for all the attribute modifications compared to the passive method [8]. We also observe an increase in cosine similarity compared to when MaLP is trained on only bald attribute modification. This is expected, as the more types of modifications MaLP sees in training, the better it learns to localize.

Transformer Architecture Ablation. We ablate various parameters of the transformer to select the best architecture for manipulation localization. We experiment with parameters that include optimizer, depth *i.e*. number of blocks, and dropout. We only use the transformer branch and switch off the CNN branch during training. The results are shown in Tab. 5. We observe that the localization performance is almost the same when using the transformer to predict fakeness maps. However, the detection accuracy has a significant impact. Having dropout does increase the performance for detection and localization. Further, using the weighted Adam optimizer is more beneficial than using the vanilla Adam optimizer. Therefore, we adopt the architecture of the transformer with 6 blocks and optimize it with a weighted Adam optimizer. Finally, we also include the dropout to achieve the best performance for localization and detection.

References

[1] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *CVPR*, 2018. 1

[2] Wonwoong Cho, Sungha Choi, David Keetae Park, Inkyu Shin, and Jaegul Choo. Image-to-image translation via group-wise deep whitening-and-coloring transformation. In *CVPR*, 2019. 1

[3] 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 *CVPR*, 2018. 1

[4] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. StarGAN v2: Diverse image synthesis for multiple domains. In *CVPR*, 2020. 1

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 1

[6] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *IEEE transactions on image processing*, 28:5464–5478, 2019. 1

[7] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018. 1

[8] Yihao Huang, Felix Juefei-Xu, Qing Guo, Yang Liu, and Geguang Pu. FakeLocator: Robust localization of gan-based face manipulations. *IEEE Transactions on Information Forensics and Security*, 17:2657–2672, 2022. 2, 3

[9] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 1

[10] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*, 2018. 1

[11] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *ECCV*, 2018. 1

[12] Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen. STGAN: A unified selective transfer network for arbitrary image attribute editing. In *CVPR*, 2019. 1

[13] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *NeurIPS*, 2017. 1

[14] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 1

[15] Kamyar Nazeri, Eric Ng, and Mehran Ebrahimi. Image colorization using generative adversarial networks. In *AMDO*, 2018. 1

[16] Ori Nizan and Ayellet Tal. Breaking the cycle - colleagues are all you need. In *CVPR*, 2020. 1

[17] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. GauGAN: semantic image synthesis with spatially adaptive normalization. In *ACM*, 2019. 1

[18] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 1

[19] Stanislav Pidhorskyi, Donald A Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *CVPR*, 2020. 1

[20] Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. Ganimation: Anatomically-aware facial animation from a single image. In *ECCV*, 2018. 1

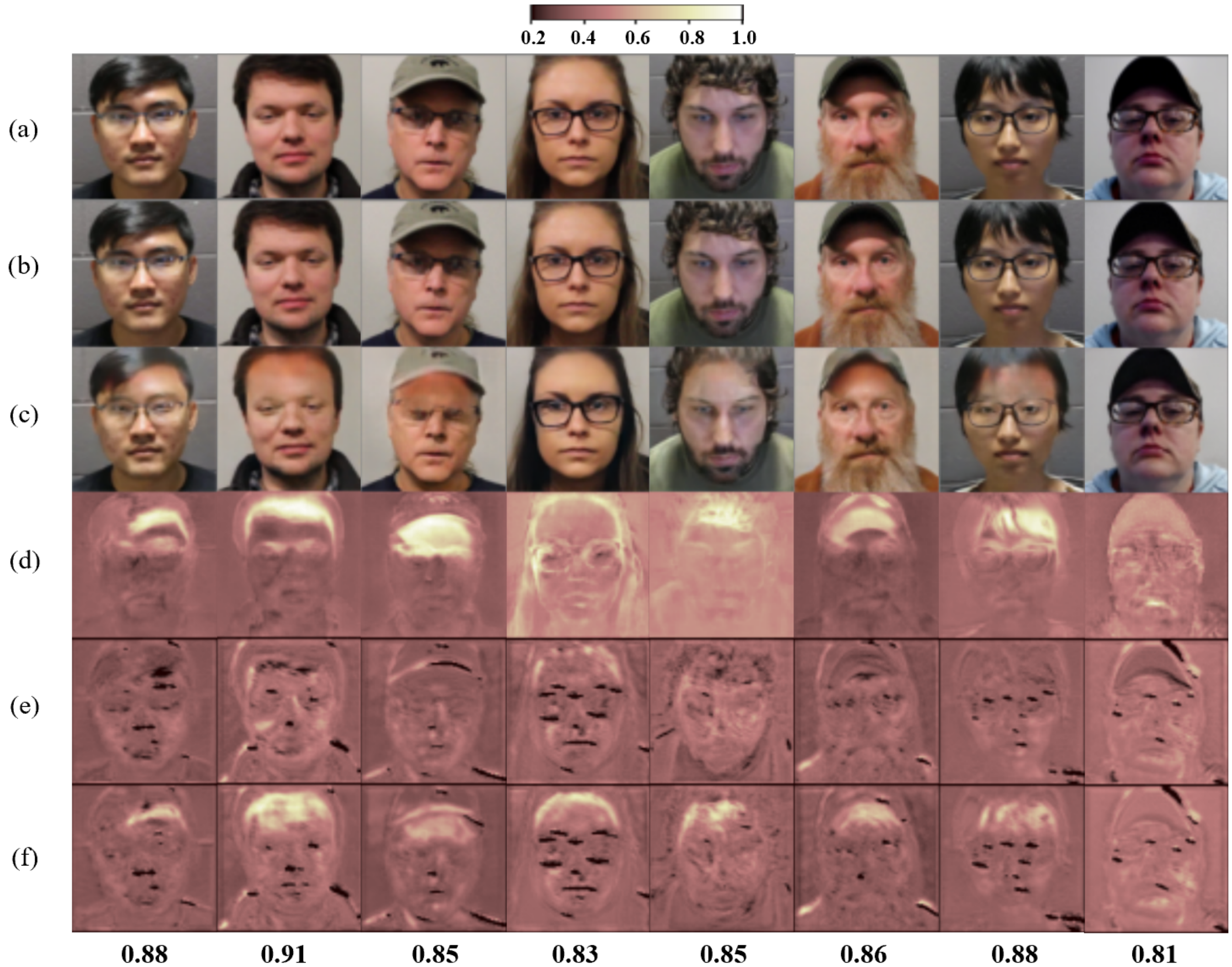

Figure 2. Visualization of fakeness maps for different attribute modifications by STGAN. (a) Real image, (b) encrypted image, (c) manipulated image, (d) ground-truth M_{GT}, (e) predicted fakeness map for encrypted images, and (f) predicted fakeness map for manipulated images. We also show the cosine similarity between the predicted and ground-truth fakeness map below (f).

[21] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016. 1

[22] Luan Tran, Xi Yin, and Xiaoming Liu. Disentangled representation learning gan for pose-invariant face recognition. In *CVPR*, 2017. 1

[23] Radim Tyleček and Radim Šára. Spatial pattern templates for recognition of objects with regular structure. In *GCPR*, 2013. 1

[24] Xiaogang Wang and Xiaoou Tang. Face photo-sketch synthesis and recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31:1955–1967, 2008. 1

[25] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In *CVPR*, 2021. 1

[26] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dual-GAN: Unsupervised dual learning for image-to-image translation. In *CVPR*, 2017. 1

[27] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in GAN fake images. In *WIFS*, 2019. 1

[28] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 1

[29] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *NeurIPS*, 2017. 1

[30] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. SEAN: Image synthesis with semantic region-adaptive normalization. In *CVPR*, 2020. 1

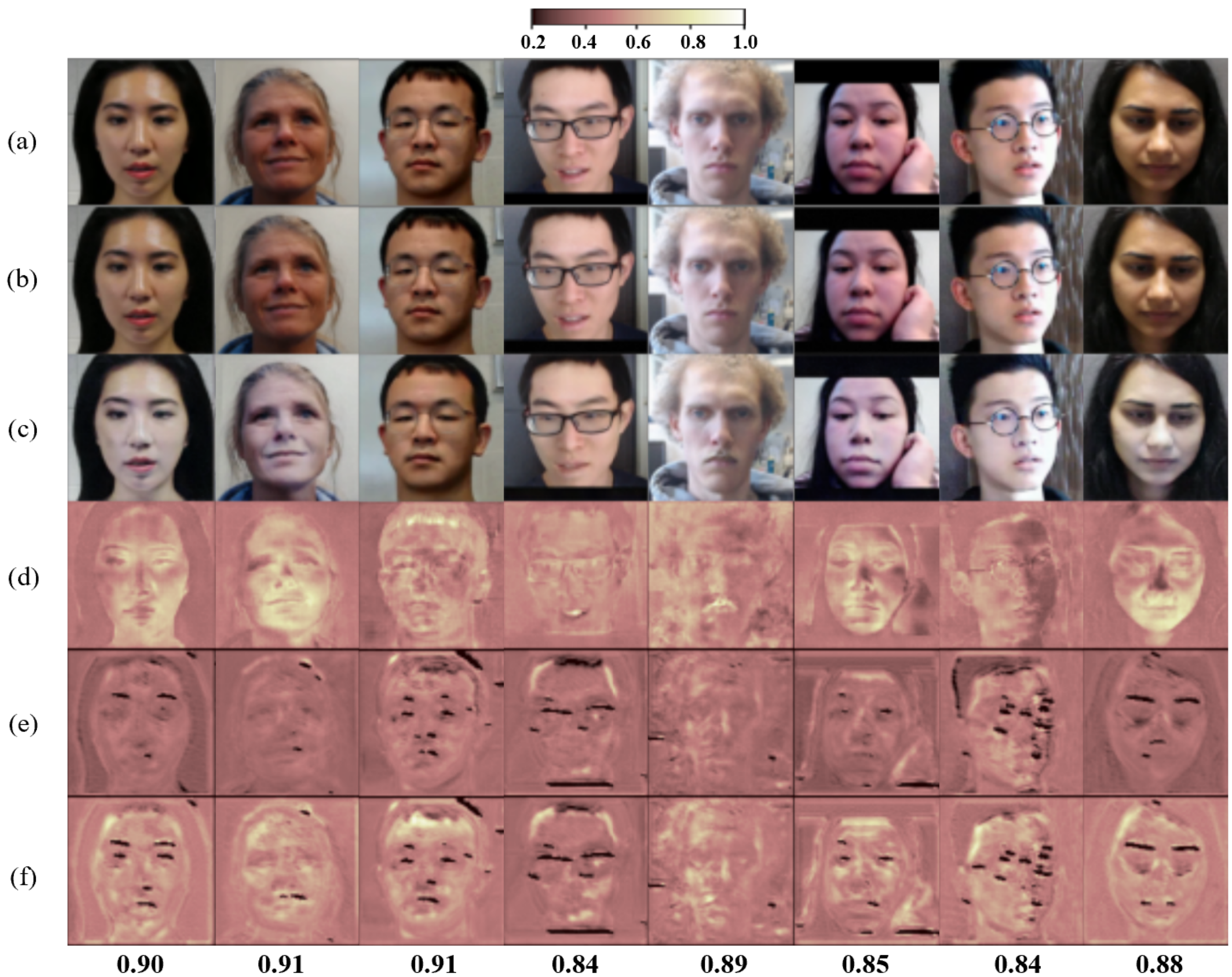

Figure 3. Visualization of fakeness maps for different attribute modifications by STGAN. (a) Real image, (b) encrypted image, (c) manipulated image, (d) ground-truth M_{GT}, (e) predicted fakeness map for encrypted images, and (f) predicted fakeness map for manipulated images. We also show the cosine similarity between the predicted and ground-truth fakeness map below (f).

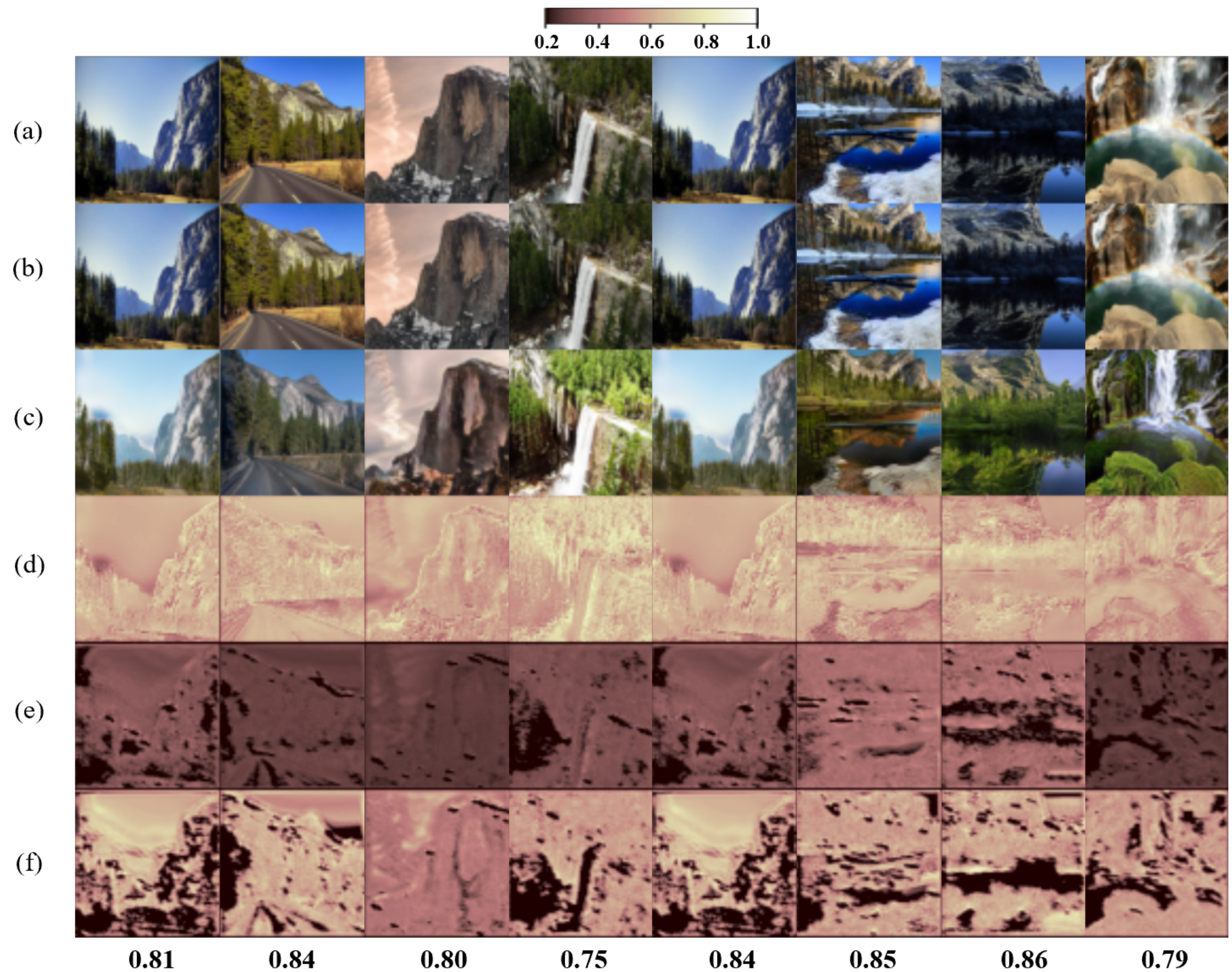

Figure 4. Visualization of fakeness maps for manipulation by DRIT. (a) Real image, (b) encrypted image, (c) manipulated image, (d) ground-truth M_{GT}, (e) predicted fakeness map for encrypted images, and (f) predicted fakeness map for manipulated images. We also show the cosine similarity between the predicted and ground-truth fakeness map below (f).

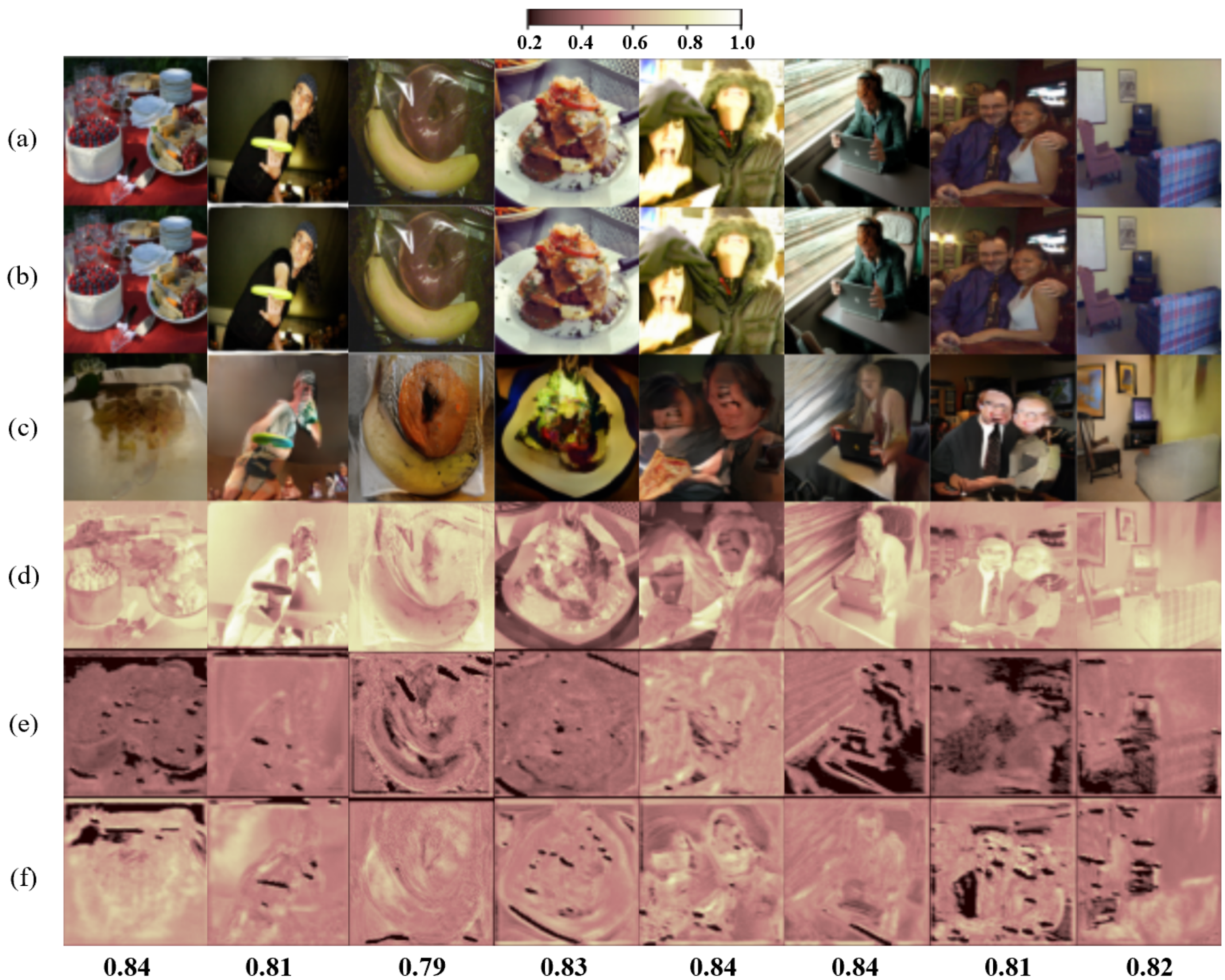

Figure 5. Visualization of fakeness maps for manipulation by GauGAN. (a) Real image, (b) encrypted image, (c) manipulated image, (d) ground-truth M_{GT}, (e) predicted fakeness map for encrypted images, and (f) predicted fakeness map for manipulated images. We also show the cosine similarity between the predicted and ground-truth fakeness map below (f).