IDGuard: Robust, General, Identity-centric POI Proactive Defense Against Face Editing Abuse

Supplementary Material

6. Model Structure

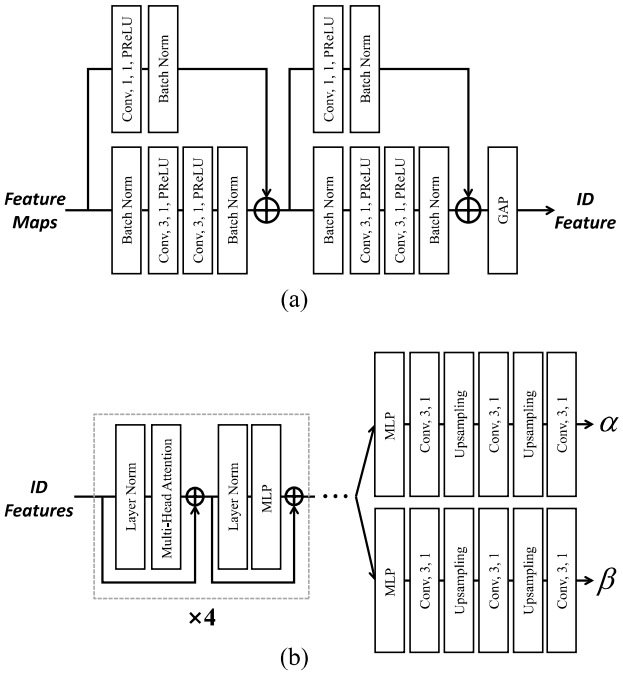

Figure 8. Architectures of (a) ID Extractor and (b) ID Encoder.

We present the details of the ID Extractor and the transformer-based ID Encoder in Fig 8. {Conv 3, 1, PReLU} denotes the configuration of the convolutional layer, e.g., the size, the stride, and the activation function. MLP and GAP stand for MultiLayer Perceptron and Global Average Pooling, respectively. We have omitted the channel numbers and feature dimensions for each layer, as they are specific and vary according to the model and position of the ID Normalization Layer.

7. Dataset Details

Table 8 illustrates the specific training/testing splits for CelebA and VGGFace2 used in this work. CelebA is entirely used for the training of face editing and the ID Extractor, while VGGFace2 serves as the POI. For VGGFace2, we utilized 1024 identities, each comprising 180 images for training. The test set is divided into two parts: one includes 20,480 images belonging to the same 1024 identities but not present in the training set, and the other consists of 200 identities not included in the training set.

Table 8. Details and training/testing splits of the datasets.

Dataset	Training		Evaluation	
	Images	ID	Images	ID
CelebA	202,599	10,177	202,599	10,177
VGGFace2	362,520	1024	20,480 4,000	1024 200

8. Training Setting

In this work, we mainly use training configurations following the official setting. For StarGAN and AGGAN, we train the model with batch size 16 for 500,000 steps using the Adam optimizer with a learning rate of 0.0001, beta 1 is 0.5, and beta 2 is 0.999. The discriminator is updated five times for each generator update. For AttGAN, we train the model with batch size 64 for 500,000 steps using the Adam optimizer with a learning rate of 0.0002, beta 1 is 0.5, and beta 2 is 0.999. The discriminator is updated two times for each generator update. For HiSD, we train the model with batch size 8 for 500,000 steps using the Adam optimizer with a learning rate of 0.0001, beta 1 is 0.0, and beta 2 is 0.99. The discriminator is updated two times for each generator update. For SimSwap, we train the model with batch size 32 for 500,000 steps using the Adam optimizer with a learning rate of 0.0004, beta 1 is 0.0, and beta 2 is 0.999. For FaceShifter, we train the model with batch size 16 for 500,000 steps using the Adam optimizer with a learning rate of 0.0004, beta 1 is 0.0, and beta 2 is 0.999. The learning rate for StarGAN and AGGAN are linearly decayed to zero from 250,000 steps. The weights of different loss terms in the total training loss λ_1, λ_2, and λ_3 are all set to 1.0.

9. Evaluation Metrics

SR_{mask} is calculated as following. First, a binary mask is first computed through Eq. 7:

$$\text{Mask}_{(i,j)} = \begin{cases} 1, & \text{if } \left\| G(x)_{(i,j)} - x_{(i,j)} \right\| > 0.5 \\ 0, & \text{else} \end{cases}, \quad (7)$$

where G is the original face editing model, x is the input face, and (i, j) is the coordinate of pixels. $\text{Mask}_{(i,j)}$ indicates whether the pixel difference between the edited image and the original image is greater than a given threshold.

Then the L^2_{mask} is calculated by:

$$L^2_{\text{mask}} = \frac{\sum_i \sum_j \text{Mask}_{(i,j)} \cdot \left\| G(x)_{(i,j)} - G_{IDG}(x)_{(i,j)} \right\|}{\sum_i \sum_j \text{Mask}_{(i,j)}}, \tag{8}$$

where G_{IDG} is the face editing model with IDGuard. L^2_{mask} represents the weighted L^2 differences between the outputs of G_{IDG} and G. Following [25], we use 0.05 as the threshold, which means we consider the protection to be successful when $L^2_{\text{mask}} > 0.05$. We denote the success rate of protecting face images as SR_{mask}. Moreover, we use $\log_{10}$ FID to measure the visual quality of forged faces of protected identities. Both high L^2_{mask} and $\log_{10}$ FID are preferable.

10. Visualization

Fig 9 shows both ID losses and POI losses of four face editing models and POI losses of two face swap models. Since both FaceShifter and SimSwap have built-in pre-trained face recognition modules for identity extraction, IDGuard does not train an additional ID extractor but rather uses the built-in extractor, thus ID Losses are not applicable.

We can observe that as the training progresses, both ID loss $\mathcal{L}_{E_{id}}$ and POI loss $\mathcal{L}_{\text{POI}}$ can successfully converge. Moreover, the convergence of the two losses exhibits synchronicity, indicating that once the ID Extractor outputs accurate ID features, the ID Encoder can effectively encode identity features into the parameters of the ID Normalization Layer and enable the model to reject editing attempts on POI.

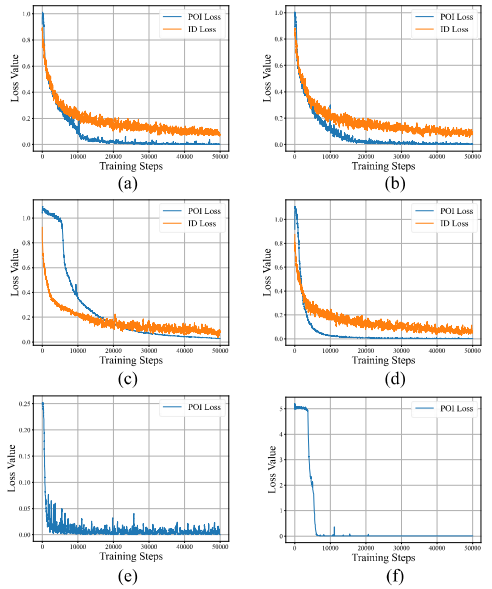

Figure 9. Losses of (a) StarGAN; (b)AGGAN; (c) AttGAN; (d) HiSD; (e)FaceShifter; (f) SimSwap.

11. Fidelity

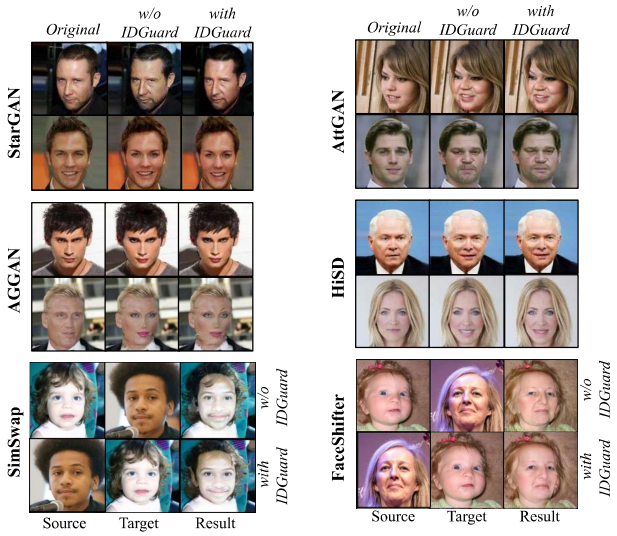

Figure 10. Original images and corresponding editing results of models with and without IDGuard.

In Fig. 10, we can observe that IDGuard has no visible negative impact on image quality in terms of visual and editing effects.

12. Number of Parameters.

Table 9 presents the number of parameters (in millions) of the original generators for different models, as well as those of their corresponding ID Extractors and ID Encoders. It can be observed that, in comparison to the original G, IDGuard only introduces a small number of additional parameters, which pose a minimal burden for both developers and users.

Table 9. Number of parameters (million) of the ID Extractor and the ID Encoder for different face editing models.

Models	G	ID Extractor	ID Encoder
StarGAN [12]	8.43	0.59	1.59
AGGAN [39]	8.43	0.59	0.79
AttGAN [23]	43.35	2.38	0.86
HiSD [32]	67.32	1.66	1.02
SimSwap [10]	59.77	-	0.45
FaceShifter [29]	377.88	-	0.87