Stabilizing the Conditional Adversarial Network by Decoupled Learning

Zhifei Zhang [1] **Yang Song** [1] **Hairong Qi** [1]

Abstract

Incorporates encoding-decoding nets with adversarial nets has been widely applied in image generation tasks. During training, the gradient from reconstruction and adversarial losses are both imposed on the generator/decoder, which may causes instability except carefully selecting an appropriate weighting factor between the two losses like existing works. In this paper, we propose a novel structure *decoupled learning*, where the reconstruction and adversarial losses are backpropagated to separate networks (i.e., the encoding-decoding net and the adversarial net, respectively), thus effectively tackles the instability problem caused by their interaction. The essential benefit is that there is no need to introduce the weighting factor between the two losses, alleviating from manual parameter adjustment while largely improving the generalization capacity of the designed model to different applications. We design a new evaluation metric, *normalized relative discriminative score* (NRDS), that assesses the relative quality of the generated images. Experimental results demonstrate that the proposed decoupled learning effectively enhances the stability and achieves competitive performance in multiple image generation tasks without the need of weight adjustment.

1. Introduction

In most recent practices, the encoding-decoding networks (ED), e.g., VAE (Kingma & Welling, 2013), AAE (Makhzani et al., 2015), autoencoder, etc., have been the popular structure to be incorporated with GANs for image-conditional modeling, where the encoder extracts features, which are then fed to the decoder/generator to generate the target images. The encoding-decoding network tends to yield blurry images. Incorporating a discriminator, as empirically demonstrated in many works (Larsen et al., 2015; Isola et al., 2017; Ledig et al., 2016; Zhang et al., 2017; Liu et al., 2017; Zhu et al., 2017), effectively increases the fidelity/resolution of generated images from the encoding-decoding networks.

In existing works that incorporate the encoding-decoding networks (ED) to GANs, the reconstruction loss (from the ED) and the adversarial loss (from the discriminator) are both imposed on a single generator/decoder. Although the ED is known to be stable in training, and many GANs works, e.g., DCGAN (Radford et al., 2015), WGAN (Arjovsky et al., 2017), LSGAN (Mao et al., 2016), etc., have stabilized the training of GANs, coupling the reconstruction loss and the adversarial loss by making them interact/compete with each other may yield unstable results or introduce artifacts as shown in Fig.1. Existing works introduce a weighting factor to balance the effect of the two losses. However, to find an appropriate weight is often time-consuming due to the exhaustive searching mechanism adopted in these works. How to set an appropriate weight automatically or completely remove the necessity of the weighting factor is a problem unexplored.

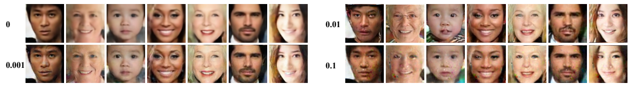

Figure 1. Artifacts introduced by the adversarial network. The top-left row shows the generated images from ED only. The rest rows show the images from the coupled structure (ED plus GAN) with different weights (marked in the left of each row) applied on the adversarial loss.

Fig. 1 illustrates the effect of adding the adversarial loss (with different weights from 0.001 to 0.1) to the reconstruction loss. We observe the increased fidelity of generated images as compared to the image generated from ED only (the top-left row in Fig. 1). However, we also observe the obvious artifacts introduced by adding the adversarial loss (e.g., the 1st, 2nd faces with weights of 0.01 and 0.1). Generally, the trade-off between the two losses needs to be carefully tuned, otherwise, the generated images may present significant artifacts, e.g., stripe, overexposure, spots, or anything visually unrealistic.

We denote this coupled structure between ED and GAN as

[1] University of Tennessee, Knoxville, TN USA. Correspondence to: Zhifei Zhang <zzhang61@vols.utk.edu>.

ICML 2017 Workshop on Implicit Models. Copyright 2017 by the author(s).

ED+GAN[1], where a higher weight on the adversarial loss preserves richer details in generated images but suffering higher risk of introducing significant artifacts or even causing instability, while a lower weight on the adversarial loss would not effectively boost the image fidelity. In this paper, we propose a novel *decoupled learning* structure, aiming to solve the instability issue induced by the coupled structures. We denote the decoupled structure as ED//GAN[2].

2. Decoupled Learning

Compared to ED+GAN, the uniqueness of the proposed ED//GAN lies in the two decoupled backpropagation paths where the reconstruction and adversarial losses are backpropagated to separate networks, instead of imposing both losses to generator/decoder (Dec) as done in ED+GAN. Fig. 2 illustrates the major difference between the coupled versus decoupled designs.

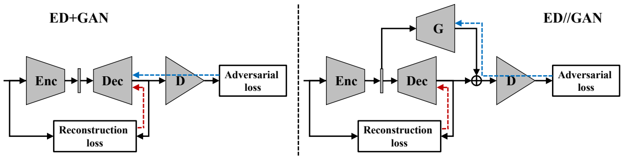

Figure 2. Comparison between ED+GAN and ED//GAN. Left: the existing ED+GAN. Right: the proposed ED//GAN, i.e., decoupled learning. Enc and Dec are the encoder and decoder networks, and G and D are the generator and discriminator, respectively. Solid black arrows denote feedforward path, and dashed arrows in red and blue indicate backpropagation of the reconstruction loss and the adversarial loss, respectively.

In ED+GAN, both reconstruction and adversarial losses are backpropagated to Dec, and the general objective could be written as

$$\min_{Enc,Dec,D} \mathcal{L}_{const}+\lambda\mathcal{L}_{adv} \quad \text{or} \quad \min_{Enc,Dec,D} \lambda\mathcal{L}_{const}+\mathcal{L}_{adv},$$

where $\mathcal{L}_{const}$ and $\mathcal{L}_{adv}$ denote the reconstruction and adversarial losses, respectively. The parameter λ is the weight to balance the two losses. In ED//GAN, we could relax the weight λ, and the general objective for ED//GAN becomes

$$\min_{Enc,Dec,G,D} \mathcal{L}_{const} + \mathcal{L}_{adv}.$$

The proposed decoupled learning (ED//GAN) is detailed in Fig. 3. The reconstructed image from ED is I_{ED}, which is a blurred version of the input image I. The generator G, together with the discriminator D, learns I_G which contains the residual information of the generated image that effectively increases the resolution and photo-realism of the image. This structure generates an output image, $\hat{I}$. Since $I \approx I_{ED} + I_G = \hat{I}$, the generator G learns the residual map I_G that reflects the details learned from the adversarial network. In addition, the residual map directly illustrates how the adversarial learning boosts the performance of ED.

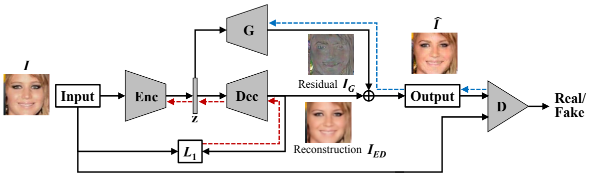

Figure 3. The flow of proposed decoupled learning, i.e., ED//GAN. L_1 indicates the pixel-level ℓ_1-norm. Solid black arrows denote the feedforward path, and dashed arrows in red and blue indicate the backpropagation from reconstruction loss (L_1) and adversarial loss (from D), respectively.

Unlike existing works that couple the learning of G and Dec (or together with Enc), we learn them separately. In the following, we elaborate on the reconstruction learning of Enc and Dec and the adversarial learning of G and D. Enc and Dec (i.e., ED) are trained separately from G and D (i.e., GAN), updated through the ℓ_1-norm in pixel level as shown by the red dashed arrow in Fig. 3. G and D are updated by the adversarial loss as indicated by the blue dashed arrow. The final output image is obtained by pixel-wise summation of the outputs from G and Dec, as denoted by $\bigoplus$. In the proposed ED//GAN framework, the gradient derived from reconstruction and adversarial losses are directed in separated flows without any interaction, avoiding the competition between the ED net and GAN which may cause instability albeit widely used in existing works as discussed in Sec. 1.

Reconstruction Learning The encoding-decoding network (ED) aims to minimize the pixel-level error between the input image I and the reconstructed image I_{ED}. The ED could be any structure specifically designed for any applications, e.g., the U-Net (Isola et al., 2017) or the conditional network (Zhang et al., 2017) with/without batch normalization. The reconstruction loss from ED can be expressed as

$$\mathcal{L}_{const}(Enc, Dec) = \|I - Dec(Enc(I))\|_1,$$

where Enc and Dec indicate the encoder and decoder.

Adversarial Learning In the proposed ED//GAN, GAN works differently from the vanilla GAN in two aspects: 1) The inputs of G are features of the input image (sharing the latent variable z with Dec) rather than the random noise. 2) The faked samples fed to D are not directly generated by G. Instead, they are conditioned on the output from Dec.

[1] The coupled structures used in existing works are denoted as ED+GAN because they add the effects of ED and GAN together during training.

[2] The proposed decoupled learning is denoted as ED//GAN, indicating that the effect from ED and GAN are learned/propagated separately through the two networks.

Therefore, the losses of GAN can be expressed as

$$\begin{aligned}\mathcal{L}_{adv}(D) =& \mathbb{E}\left[\log\left(1 - D(I)\right)\right] + \mathbb{E}\left[\log D\left(I_{ED} + I_G\right)\right],\\ \mathcal{L}_{adv}(G) =& \mathbb{E}\left[\log\left(1 - D\left(I_{ED} + G(z)\right)\right)\right],\end{aligned}$$

where $I_{ED} = Dec(z)$ and $z = Enc(I)$. Finally, we obtain the objective of the proposed decoupled learning (ED//GAN),

$$\min_{Enc,Dec} \mathcal{L}_{const}(Enc, Dec) + \min_{G}\mathcal{L}_{adv}(G) + \min_{D}\mathcal{L}_{adv}(D).$$

Note that there are no weighting parameters between the losses in the objective function, which relaxes the manual tuning that may require an expert with strong domain knowledge and rich experience. During training, each component could be updated alternatively and separately because the three components do not overlap in backpropagation, i.e., the backpropagation paths are not intertwined.

Visualizing the Boost from Adversarial Learning The ED//GAN helps to investigate how the discriminator independently boosts the quality of generated images. In ED+GAN, however, the effect of discriminator is difficult to directly identify because it is coupled with the effect of ED. The learned residual in ED//GAN is considered the boosting factor from the adversarial learning (discriminator). Generally, the images from ED tend to be blurry, while the residual from GAN carries the details or important texture information for photo-realistic image generation. Imposing the residual onto the reconstructed images is supposed to yield higher-fidelity images as compared to the reconstructed images. In Fig. 4, we can observe that the adversarial learning mainly enhances the edges at eyebrow, eyes, mouth, teeth, etc. (see Fig. 4, middle of each triple) Adding the residual to the blurry images from ED (Fig. 4 left), the output images present finer details.

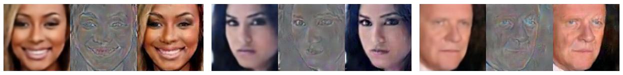

Figure 4. Visualization of the boost from adversarial learning. From left to right in each triple: reconstruction, residual, and output images from ED//GAN.

3. Experimental Evaluation

We evaluate the proposed decoupled learning mainly from its ability in relaxing the weight setting and stabilizing the training process. We compare the proposed ED//GAN to the traditional ED+GAN based on two datasets, i.e., UTKFace (Zhang et al., 2017) and the CMP Facade Database (Radim Tyleček, 2013). Details regarding the network structures and datasets are given in the supplementary materials.

3.1. Evaluation Metric

Evaluation metrics for generative models or generated images are still limited and human evaluation has been applied in many works (Isola et al., 2017; Zhang et al., 2017). Inception score (Salimans et al., 2016) and related methods (Odena et al., 2016) were proposed in recent years, aiming to provide a metric to evaluate the quality of generated images or the generative model. However, inception score highly depends on the availability of good classifiers which require labeled datasets, and is inappropriate for our evaluations. We propose the *normalized relative discriminative score* (NRDS) to assess the images generated from different models. Instead of providing an absolute score for each individual model, the proposed NRDS aims to compare two or multiple models, giving a relative score to illustrate which model is better than the others. NRDS trains a discriminator/classifier on real images and generates images from two or multiple models that need to be compared. During training, the output from the discriminator will tend to 1 for real images and approach 0 for generated images, but the approaching speed will differ for different models. Intuitively, a model that generates relatively more photo-realistic images will approach 0 slower, and vice versa. A toy example is shown in supplementary materials. Therefore, the speed of approaching 0 implies the image quality generated from a model. There are three steps to compute NRDS: 1) Obtain the curve $\mathcal{C}_i$ $(i = 1, 2, \cdots, n)$ of discriminator output vs. epoch (or mini batch) for each model (assuming n models) during training; 2) Compute the area under each curve $A(\mathcal{C}_i)$; and 3) Compute NRDS of the ith model by $NRDS_i = \frac{A(\mathcal{C}_i)}{\sum_{j=1}^{n} A(\mathcal{C}_j)}$. Generally, a larger NRDS indicates relatively better image quality, and $\sum_i NRDS = 1$.

3.2. Stabilizing the Training Process

To evaluate the stability of the proposed decoupled learning regardless of the weight setting and batch normalization, we compare the proposed decoupled learning (ED//GAN) with the traditional method (ED+GAN). We increase the weight of adversarial loss to compare the quality of generated images from the two structures. We fix the weight of reconstruction at 1 and increase the weight of the adversarial loss from 0.001 to 1 with the step of 10x. After 200 epochs with the batch size of 25, Fig. 5 compares the output images without/with batch normalization. Compared to ED+GAN, the proposed ED//GAN can yield more stable outputs that are insensitive to weight changes. The NRDS results of images illustrated in Fig. 5 is listed in Table 2. Testing results from all methods (with different weight settings) are collected to train a single discriminator. We observe that ED//GAN generally yield higher NRDS, indicating better image quality. In addition, the NRDS values for

Table 1. NRDS on the results partially illustrated in Fig. 6 (left) and Fig. 7 (right).

Method	ED+GAN			ED//GAN	ED+GAN			ED//GAN
	1:1	100:1	1000:1		$1:10^{-4}$	$1:10^{-2}$	1:1	
NRDS	.2190	**.2641**	.2572	.2597	.2527	.2496	.2430	**.2547**

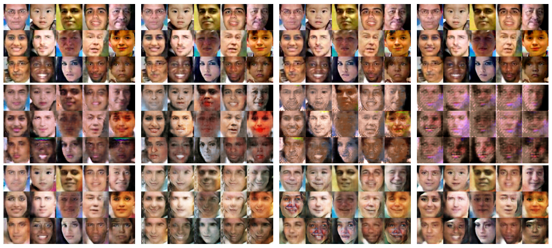

Figure 5. Comparison of ED//GAN (top) and ED+GAN without (middle) and with (bottom) batch normalization on ED using the UTKFace dataset. From left to right, the weights on the adversarial loss are 0.001, 0.01, 0.1, and 1, respectively.

ED//GAN vary much less than those of ED+GAN, as observed from the lower standard deviation (std), indicating robustness against different weights.

Table 2. NRDS with different weight settings and their std.

	0.001	0.01	0.1	1	std
ED+GAN	.1172	.1143	.1163	.0731	.0215
ED+GAN2	.1066	.1143	.1268	.1267	.0099
ED//GAN	.1432	.1434	.1458	.1466	.0017

ED+GAN2 denote the structure with batch normalization

3.3. From Coupled Learning to Decoupled Learning

To illustrate the effectiveness of ED//GAN, we adapt several existing works that use the ED+GAN structure to ED//GAN and compare the generated images. We modify two works: 1) Pix2Pix (Isola et al., 2017) for image transformation and 2) CAAE (Zhang et al., 2017) for image manipulation (face aging). We modify them by parallelizing an extra generator to the original one to learn the residual. More details are shown in the supplementary. The weight of reconstruction and adversarial losses is set to be 100 and 1 (i.e., 100:1) in Pix2Pix (Isola et al., 2017). In Fig. 6, we use the weights of 1:1, 100:1, and 1000:1 for the original structure (ED+GAN) and compare with the modified version using the decoupled structure (ED//GAN).

We observe that the generated images with the weight of 1:1 introduce significant artifacts (please zoom in for better view). With higher weight on the reconstruction loss, 100:1 and 10:1 yield more realistic images, whose quality is similar to that from the modified decoupled structure that does not need weight setting.

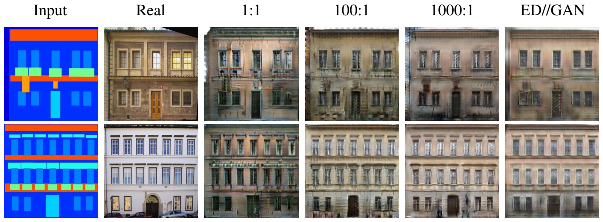

Figure 6. Comparison between Pix2Pix (Isola et al., 2017) and the modified version using the proposed ED//GAN.

We modify CAAE to the decoupled ED//GAN structure. Fig. 7 shows some random examples to compare the original and modified structures. The weights of the reconstruction and adversarial losses are 1 and 10^{-4} (i.e., $1:10^{-4}$) in the original work. We use a couple of different weight settings, $1:10^{-4}$, $1:10^{-3}$, $1:10^{-2}$, and 1:1, for the original structure and compare the results with the modified decoupled structure. Table 1 lists the NRDS.

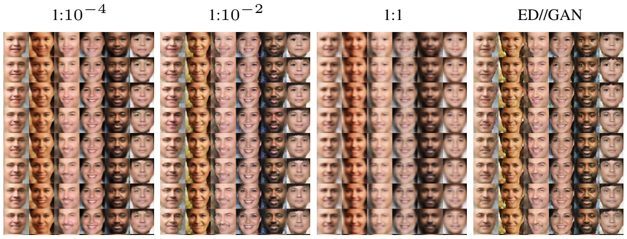

Figure 7. Comparison between CAAE (Zhang et al., 2017) and the modified version using the proposed ED//GAN.

4. Conclusion

This paper proposed the novel decoupled learning structure (ED//GAN) for image generation tasks with image-conditional models. Different from existing works where the reconstruction loss (from ED) and the adversarial loss (from GAN) are backpropagated to a single decoder, referred to as the coupled structure (ED+GAN), in ED//GAN, the two losses are backpropagated through separate networks, thus avoiding the interaction/competition between each other. The essential benefit of the decoupled structure is such that the weighting factor that has to be fine-tuned in ED+GAN is no longer needed in the decoupled structure, thus improving stability without looking for the best weight setting. This would largely facilitate the wider realization of more specific image generation tasks. The experimental results demonstrated the stability of the decoupled learning.

References

Arjovsky, Martin, Chintala, Soumith, and Bottou, Léon. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

Isola, Phillip, Zhu, Jun-Yan, Zhou, Tinghui, and Efros, Alexei A. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

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

Larsen, Anders Boesen Lindbo, Sønderby, Søren Kaae, Larochelle, Hugo, and Winther, Ole. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015.

Ledig, Christian, Theis, Lucas, Huszár, Ferenc, Caballero, Jose, Cunningham, Andrew, Acosta, Alejandro, Aitken, Andrew, Tejani, Alykhan, Totz, Johannes, Wang, Zehan, et al. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint arXiv:1609.04802*, 2016.

Liu, Ming-Yu, Breuel, Thomas, and Kautz, Jan. Unsupervised image-to-image translation networks. *arXiv preprint arXiv:1703.00848*, 2017.

Makhzani, Alireza, Shlens, Jonathon, Jaitly, Navdeep, Goodfellow, Ian, and Frey, Brendan. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

Mao, Xudong, Li, Qing, Xie, Haoran, Lau, Raymond YK, and Wang, Zhen. Least squares generative adversarial networks. *arXiv preprint ArXiv:1611.04076*, 2016.

Nilsback, M-E. and Zisserman, A. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008.

Odena, Augustus, Olah, Christopher, and Shlens, Jonathon. Conditional image synthesis with auxiliary classifier gans. *arXiv preprint arXiv:1610.09585*, 2016.

Radford, Alec, Metz, Luke, and Chintala, Soumith. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Radim Tyleček, Radim Šára. Spatial pattern templates for recognition of objects with regular structure. In *Proc. GCPR*, Saarbrucken, Germany, 2013.

Salimans, Tim, Goodfellow, Ian, Zaremba, Wojciech, Cheung, Vicki, Radford, Alec, and Chen, Xi. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pp. 2226–2234, 2016.

Wah, Catherine, Branson, Steve, Welinder, Peter, Perona, Pietro, and Belongie, Serge. The caltech-ucsd birds-200-2011 dataset. 2011.

Zhang, Zhifei, Song, Yang, and Qi, Hairong. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Zhu, Jun-Yan, Park, Taesung, Isola, Phillip, and Efros, Alexei A. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*, 2017.

supplementary

A. Stabilizing the Training Process

To evaluate the stability of the proposed decoupled learning regardless of the weight setting and batch normalization, we compare the proposed decoupled learning (ED//GAN) with the traditional method (ED+GAN). Two factors are considered here: 1) the weight of adversarial loss and 2) batch normalization which is a common way to stabilize the training process. We increase the weight of adversarial loss to compare the quality of generated images from the two structures. We fix the weight of reconstruction at 1 and increase the weight of the adversarial loss from 0.001 to 1 with the step of 10x. After 200 epochs with the batch size of 25, Figs. 8 and 9 compare the output images with and without batch normalization, respectively.

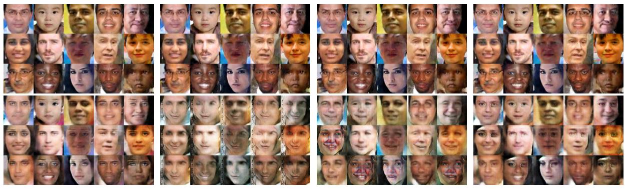

Figure 8. Comparison of ED//GAN (top) and ED+GAN (bottom) with batch normalization on ED using the UTKFace dataset. From left to right, the weights on the adversarial loss are 0.001, 0.01, 0.1, and 1, respectively. Please zoom in for better view.

The output images from ED//GAN generates relatively higher-fidelity images regardless of the weight change. However, the outputs of ED+GAN are significantly affected by the weight value. It obtains relatively better results when the weight of the adversarial loss is 0.001. As the weight increases, the outputs become unstable, i.e., the images start presenting significant artifacts and fall into fewer modes.

In Fig. 9, the batch normalization in ED is removed. The proposed ED//GAN can still yield stable outputs, while the ED+GAN generates images with stripe, spots, noise, etc.

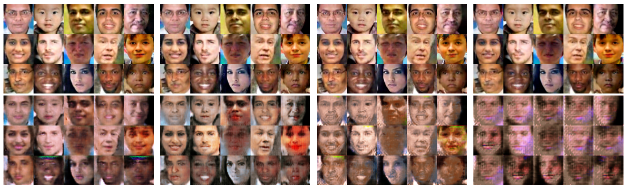

Figure 9. Comparison of ED//GAN (top) and ED+GAN (bottom) without batch normalization on ED using the UTKFace dataset. From left to right, the weights on the adversarial loss are 0.001, 0.01, 0.1, and 1, respectively.

From the two experiments, ED//GAN vs. ED+GAN with/without batch normalization on ED, we can observe that the proposed decoupled learning presents stable performance and is insensitive to weight changes, thus relaxing the weight setting in ED//GAN. The NRDS results of images illustrated in Figs. 8 and 9 are listed in Tables 3 and 4, respectively. In each table, testing results from all methods (with different weight settings) are collected to train a single discriminator. From Tables 3 and 4, we observe that ED//GAN generally yield higher NRDS, indicating better image quality. In addition, the NRDS values for ED//GAN vary much less than those of ED+GAN, as observed from the lower standard deviation (std), indicating robustness against different weights.

Table 3. NRDS and their standard derivation (std) on the results partially illustrated in Fig. 8.

	0.001	0.01	0.1	1	std
ED+GAN	.1066	.1143	.1268	.1267	.0099
ED//GAN	.1320	.1300	.1300	.1336	.0017

Table 4. NRDS and their standard derivation (std) on the results partially illustrated in Fig. 9.

	0.001	0.01	0.1	1	std
ED+GAN	.1172	.1143	.1163	.0731	.0215
ED//GAN	.1432	.1434	.1458	.1466	.0017

We also apply the proposed ED//GAN structure on the CUB-200 and Oxford Flower datasets without any weight parameters to further demonstrate the generality and stability of the decoupled learning structure. Fig. 10 displays the results after 200 epochs. The outputs gain more details compared to the reconstructed images. The residual illustrates that both details and colors are enhanced by the adversarial network.

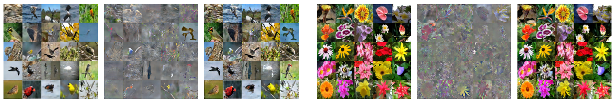

Figure 10. Output of ED//GAN trained on the CUB-200 and Oxford Flower datasets. From left to right in each triple: reconstruction, residual, and output images.

B. From Coupled Learning to Decoupled Learning

In both Enc and Dec, the kernel size is 5×5, and the stride is 2. The activation function is ReLU or Leaky ReLU (LReLU) for each hidden layer. The output layer adopts the hyperbolic tangent (tanh) function. Batch normalization is optionally applied before activation functions. ADAM optimizer is adopted with the learning rate of 0.0002 and $\beta_1 = 0.5$.

B.1 Network Structure for Section 3.2

This network is neither specifically designed for any applications nor delicately fine-tuned to achieve the best result. The goal is to demonstrate the stability of the propose method (ED//GAN) as compared to the traditional method (ED+GAN). Therefore, we only need to ensure that both methods are compared fairly on the same structure and using the same hyperparameters.

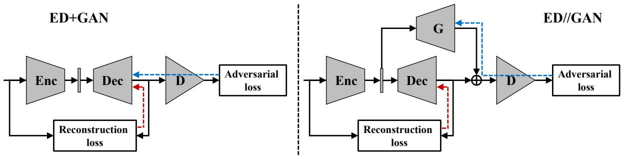

Figure 11. Left: the existing ED+GAN. Right: the adaption based on the proposed ED//GAN, i.e., decoupled learning. Solid black arrows denote the feedforward path, and dashed arrows in red and blue indicate backpropagation from the reconstruction loss and the adversarial loss, respectively. The network details are listed in Table 5.

Table 5. Structure of the Enc and Dec networks as shown in Fig. 11. The value 50 indicates a vector with the length of 50. Batch normalization is optional as indicated by (BN). The size of each layer is denoted by $h \times w \times c$, corresponding to width, and number of channels, respectively.

Enc	Size
Input	$128 \times 128 \times 3$
Conv, (BN), ReLU	$64 \times 64 \times 64$
Conv, (BN), ReLU	$32 \times 32 \times 128$
Conv, (BN), ReLU	$16 \times 16 \times 256$
Conv, (BN), ReLU	$8 \times 8 \times 512$
Conv, (BN), ReLU	$4 \times 4 \times 1024$
Reshape, FC, tanh	50
Dec	**Size**
Input	50
FC, ReLU, (BN), Reshape	$4 \times 4 \times 1024$
Deconv, (BN), ReLU	$8 \times 8 \times 512$
Deconv, (BN), ReLU	$16 \times 16 \times 256$
Deconv, (BN), ReLU	$32 \times 32 \times 128$
Deconv, (BN), ReLU	$64 \times 64 \times 64$
Deconv, tanh	$128 \times 128 \times 3$

The generator G uses the same structure as Dec, and they share the latent variable z. The discriminator D adopts the similar structure as Enc, but the length of output is 1 (indicating real and fake) instead of 50 (the latent variable z). In G and D networks, batch normalization is applied to ensure stable training of GAN as suggested in DCGAN (Radford et al., 2015).

B.2 Network Structure for Image Transformation in Section 3.3

We adapt the network in Pix2Pix (Isola et al., 2017), which is ED+GAN structure, to the proposed ED//GAN structure as shown in Fig. 12.

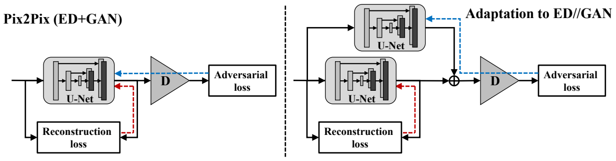

Figure 12. Left: the structure of Pix2Pix (ED+GAN). Right: the adaption to the proposed ED//GAN, i.e., decoupled learning. Solid black arrows denote the feedforward path, and dashed arrows in red and blue indicate backpropagation from the reconstruction loss and the adversarial loss, respectively.

In Pix2Pix, the ED is implemented by the U-Net, which directly passes feature maps from encoder to decoder, preserving more details. For simplicity and not breaking the structure of U-Net, we apply another U-Net as the generator G in correspondingly adaption to ED//GAN. The discriminator adopts the same network structure as in Pix2Pix. The network details are listed in Table 6.

Table 6. Structure of U-Net network as shown in Fig. 12. Enc and Dec denote the encoding and decoding part in U-Net, respectively. The number of channels of the hidden layers in Dec are twice of those in Enc because of the direct passing.

Enc	Size
Input	$256 \times 256 \times 3$
Conv, BN, LReLU	$128 \times 128 \times 64$
Conv, BN, LReLU	$64 \times 64 \times 128$
Conv, BN, LReLU	$32 \times 32 \times 256$
Conv, BN, LReLU	$16 \times 16 \times 512$
Conv, BN, LReLU	$8 \times 8 \times 512$
Conv, BN, LReLU	$4 \times 4 \times 512$
Conv, BN, LReLU	$2 \times 2 \times 512$
Conv, BN, LReLU	$1 \times 1 \times 512$
Dec	**Size**
Input	$1 \times 1 \times 512$
Deconv, BN, ReLU, Dropout	$2 \times 2 \times 1024$
Deconv, BN, ReLU, Dropout	$4 \times 4 \times 1024$
Deconv, BN, ReLU, Dropout	$8 \times 8 \times 1024$
Deconv, BN, ReLU	$16 \times 16 \times 1024$
Deconv, BN, ReLU	$32 \times 32 \times 512$
Deconv, BN, ReLU	$64 \times 64 \times 256$
Deconv, BN, ReLU	$128 \times 128 \times 128$
Deconv, Tanh	$256 \times 256 \times 3$

B.3 Network Structure for Image Manipulation in Section 3.3

We adapt the face aging work (Zhang et al., 2017) (CAAE), which proposed a conditional ED+GAN structure, to the proposed ED//GAN structure as shown in Fig. 13. CAAE generated aged face by manipulating the label concatenated to the latent variable z from Enc.

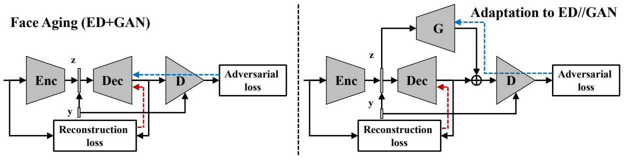

Figure 13. Left: the ED+GAN structure used in CAAE (Zhang et al., 2017). Right: the adaption to the proposed ED//GAN, i.e., decoupled learning. Solid black arrows denote the feedforward path, and dashed arrows in red and blue indicate backpropagation from the reconstruction loss and the adversarial loss, respectively. The age label y is concatenated to z to control the age of generated faces.

The original network used in CAAE has an extra discriminator on z to fore z uniformly distributed. We do not show this discriminator in Fig. 13 because it does not affect the adaptation. The network details are listed in Table 7.

Table 7. Structure of the Enc, Dec, G, and D networks as shown in Fig. 7. The value 50 indicates a vector with the length of 50.The concatenated age vector y is 50. Structure of the "G" and "D" networks as shown in Fig. 13. The value 50 indicates a vector with the length of 50. The concatenated age vector y is 50. If use WGAN or LSGAN, the BN is optional in G.

Enc	Size
Input	$128 \times 128 \times 3$
Conv, ReLU	$64 \times 64 \times 64$
Conv, ReLU	$32 \times 32 \times 128$
Conv, ReLU	$16 \times 16 \times 256$
Conv, ReLU	$8 \times 8 \times 512$
Conv, ReLU	$4 \times 4 \times 1024$
Reshape, FC, tanh	50

Dec	Size
Input	$50 + 50$ (length of the label)
FC, ReLU, Reshape	$8 \times 8 \times 1024$
Deconv, ReLU	$16 \times 16 \times 512$
Deconv, ReLU	$32 \times 32 \times 256$
Deconv, ReLU	$64 \times 64 \times 128$
Deconv, ReLU	$128 \times 128 \times 64$
Deconv, tanh	$128 \times 128 \times 3$

G	Size
Input	$50 + 50$ (length of the label)
FC, BN, ReLU, Reshape	$8 \times 8 \times 1024$
Deconv, BN, ReLU	$16 \times 16 \times 512$
Deconv, BN, ReLU	$32 \times 32 \times 256$
Deconv, BN, ReLU	$64 \times 64 \times 128$
Deconv, BN, ReLU	$128 \times 128 \times 64$
Deconv, tanh	$128 \times 128 \times 3$

D	Size
Input	$128 \times 128 \times 3$
Conv, BN, ReLU	$64 \times 64 \times (16 + 50)$
Conv, BN, ReLU	$32 \times 32 \times 32$
Conv, BN, ReLU	$16 \times 16 \times 64$
Conv, BN, ReLU	$8 \times 8 \times 128$
Reshape, FC, ReLU	1024
FC, sigmoid	1

C. Normalized Relative Discriminative Score (NRDS)

To illustrate the computation of NRDS, Fig. 14 shows a toy example. Assume the samples of fake-close and fake-far are generated from two different models, aiming to simulate the real samples. We train a discriminator on the real and fake (i.e., fake-close and fake-far) samples. The structure of discriminator is a neural network with two hidden layers, both of which has 32 nodes, and the ReLU is adopted as the activation function. After each epoch, the discriminator is tested on the samples from real, fake-close, and fake-far, respectively. Specifically, all the real samples are fed to the discriminator, and then we compute the mean of the outputs from the discriminator. Repeating this process, we get the averaged outputs from the samples in fake-close and fake-far, respectively. Finally, we achieve the curves as shown in Fig. 14 (right). Intuitively, the curve of fake-close approaches zero slower than that of fake-far because the samples in fake-close are more close (similar) to the real samples.

NRDS is computed based on these curves — area under the curve. The area under the curves of fake-close ($\mathcal{C}_1$) and fake-far ($\mathcal{C}_2$) are $A(\mathcal{C}_1) = 145.4955$ and $A(\mathcal{C}_2) = 71.1057$, respectively. Then, the NRDS for fake-close ($NRDS_1$) and fake-far ($NRDS_2$) can be computed by

$$NRDS_1 = \frac{A(\mathcal{C}_1)}{\sum_{i=1}^{2} A(\mathcal{C}_i)} = 0.6717, \tag{1}$$

$$NRDS_2 = \frac{A(\mathcal{C}_2)}{\sum_{i=1}^{2} A(\mathcal{C}_i)} = 0.3283, \tag{2}$$

where NRDS of fake-close is higher than that of fake-far, indicating better generated samples, i.e., closer to real samples. Therefore, we can evaluate the model of generating fake-close as relatively better.

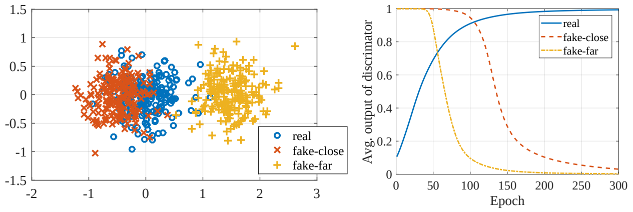

Figure 14. A toy example of computing NRDS. **Left:** the real and fake samples randomly sampled from 2-D normal distributions with different means but with the same (identity) covariance. The real samples (blue circle) is with zero mean. The red "x" and yellow "+" denote fake samples with the mean of $[-.5, 0]$ and $[1.5, 0]$, respectively. The notation fake-close/far indicates that the mean of correspondingly fake samples is close to or far from that of the real samples. **Right:** the curves of epoch vs. averaged output of discriminator on corresponding sets (colors) of samples.

D. Datasets

Four datasets are used to evaluate the effectiveness and stability of the proposed decoupled learning: 1) UTKFace (Zhang et al., 2017), 2) Caltech-UCSD Birds 200 (CUB-200) (Wah et al., 2011), 3) Oxford Flower (Nilsback & Zisserman, 2008) and 4) CMP Facade Database (Radim Tyleček, 2013). The UTKFace dataset consists of about 20,000 aligned and cropped faces with large diversity in age and race. The decoupled learning applied on the UTKFace dataset aims to demonstrate the performance on image manipulation tasks. The CUB-200 dataset has 6,033 images of 200 birds species with large and diverse background, including ocean, trees, flowers, etc. The Oxford Flower dataset are images of flowers with diverse species and colors. The decoupled learning is applied on these two datasets to demonstrate the stability and generalization on image generation tasks. The CMP Facade dataset is utilized to illustrate the performance of the decoupled learning on image transformation tasks. These four datasets are in different domains (because of different objects and background). The experimental results validate the robustness and stability of the decoupled learning in parameter relaxation, i.e., guaranteeing stable training without parameter tuning on any datasets.