TuiGAN: Learning Versatile Image-to-Image Translation with Two Unpaired Images

Jianxin Lin[1⋆], Yingxue Pang[1⋆], Yingce Xia[2], Zhibo Chen[1], Jiebo Luo[3]

[1]University of Science and Technology of China
[2]Microsoft Research Asia [3]University of Rochester
{linjx, pangyx}@mail.ustc.edu.cn yingce.xia@microsoft.com
chenzhibo@ustc.edu.cn jluo@cs.rochester.edu

Abstract. An unsupervised image-to-image translation (UI2I) task deals with learning a mapping between two domains without paired images. While existing UI2I methods usually require numerous unpaired images from different domains for training, there are many scenarios where training data is quite limited. In this paper, we argue that even if each domain contains a single image, UI2I can still be achieved. To this end, we propose TuiGAN, a generative model that is trained on only two unpaired images and amounts to one-shot unsupervised learning. With TuiGAN, an image is translated in a coarse-to-fine manner where the generated image is gradually refined from global structures to local details. We conduct extensive experiments to verify that our versatile method can outperform strong baselines on a wide variety of UI2I tasks. Moreover, TuiGAN is capable of achieving comparable performance with the state-of-the-art UI2I models trained with sufficient data. Our code is available at https://github.com/linjx-ustc1106/TuiGAN-PyTorch.

Keywords: Image-to-Image Translation. Generative Adversarial Network. One-Shot Unsupervised Learning.

1 Introduction

Unsupervised image-to-image translation (UI2I) tasks aim to map images from a source domain to a target domain with the main source content preserved and the target style transferred, while no paired data is available to train the models. Recent UI2I methods have achieved remarkable successes [26,22,38,25,3]. Among them, conditional UI2I gets much attention, where two images are given: an image from the source domain used to provide the main content, and the other one from the target domain used to specify which style the main content should be converted to. To achieve UI2I, typically one needs to collect numerous unpaired images from both the source and target domains.

However, we often come across cases for which there might not be enough unpaired data to train the image translator. An extreme case resembles one-shot unsupervised learning, where only one image in the source domain and one

⋆ The first two authors contributed equally to this work

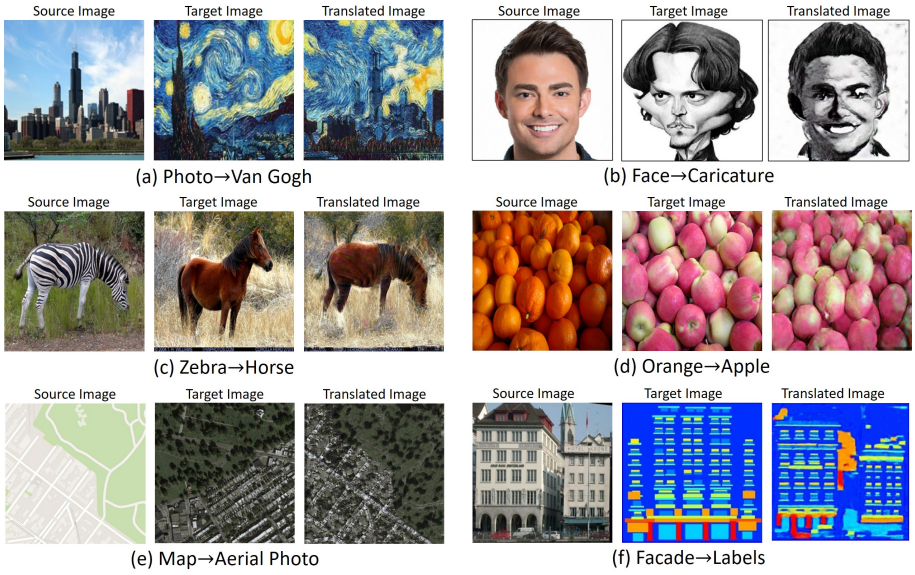

Fig. 1. Several results of our proposed method on various tasks ranging from image style transfer (Figures (a), (b)) to object transformation (Figures (c), (d)) and appearance transformation (Figures (e), (f)). In each sub-figure, the three pictures from left to right refer to the source image (providing the main content), target image (providing the style and high-level semantic information), and translated image.

image in the target domain are given but unpaired. Such a scenario has a wide range of real-world applications, e.g., taking a photo and then converting it to a specific style of a given picture, or replacing objects in an image with target objects for image manipulation. In this paper, we take the first step towards this direction and study UI2I given only two unpaired images.

Note that the above problem subsumes the conventional image style transfer task. Both problems require one source image and one target image, which serve as the content and style images, respectively. In image style transfer, the features used to describe the styles (such as the Gram matrix of pre-trained deep features [7]) of the translated image and the style image should match (e.g., Fig. 1(a)). In our generalized problem, not only the style but the higher-level semantic information should also match. As shown in Fig. 1(c), on the zebra-to-horse translation, not only the background style (e.g., prairie) is transferred, but the high-level semantics (i.e., the profile of the zebra) is also changed.

Achieving UI2I requires the models to effectively capture the variations of domain distributions between two domains, which is the biggest challenge for our problem since there are only two images available. To realize such one-shot translation, we propose a new conditional generative adversarial network, TuiGAN, which is able to transfer the domain distribution of input image to the target domain by progressively translating image from coarse to fine. The

progressive translation enables the model to extract the underlying relationship between two images by continuously varying the receptive fields at different scales. Specifically, we use two pyramids of generators and discriminators to refines the generated result progressively from global structures to local details. For each pair of generators at the same scale, they are responsible for producing images that look like the target domain ones. For each pair of discriminators at the same scale, they are responsible for capturing the domain distributions of the two domains at the current scale. The "one-shot" term in our paper is different from the ones in [1,4], which use a single image from the source domain and a set of images from the target domain for UI2I. In contrast, we only use two unpaired images from two domains in our work.

We conduct extensive experimental validation with comparisons to various baseline approaches using various UI2I tasks, including horse $\leftrightarrow$ zebra, facade $\leftrightarrow$ labels, aerial maps $\leftrightarrow$ maps, apple $\leftrightarrow$ orange, and so on. The experimental results show that the versatile approach effectively addresses the problem of one-shot image translation. We show that our model can not only outperform existing UI2I models in the one-shot scenario, but more remarkably, also achieve comparable performance with UI2I models trained with sufficient data.

Our contributions can be summarized as follows:

- We propose a TuiGAN to realize image-to-image translation with only two unpaired images.
- We leverage two pyramids of conditional GANs to progressively translate image from coarse to fine.
- We demonstrate that the a wide range of UI2I tasks can be tackled using our versatile model.

2 Related Works

2.1 Image-to-Image Translation

The earliest concept of image-to-image translation (I2I) may be raised in [11] which supports a wide variety of "image filter" effects. Rosales et al. [31] propose to infer correspondences between a source image and another target image using Bayesian framework. With the development of deep neural networks, the advent of Generative Adversarial Networks (GAN) [8] really inspires many works in I2I. Isola et al. [15] propose a conditional GAN called "pix2pix" model for a wide range of supervised I2I tasks. However, paired data may be difficult or even impossible to obtain in many cases. DiscoGAN [20], CycleGAN [38] and DualGAN [35] are proposed to tackle the unsupervised image-to-image translation (UI2I) problem by constraining two cross-domain translation models to maintain cycle-consistency. Liu et al. [27] propose a FUNIT model for few-shot UI2I. However, FUNIT requires not only a large amount of training data and computation resources to infer unseen domains, but also the training data and unseen domains to share similar attributes. Our work does not require any pre-training and specific form of data. Related to our work, Benaim et al. [1] and Cohen et al. [4]

propose to solve the one-shot cross-domain translation problem, which aims to learn an unidirectional mapping function given a single image from the source domain and a set of images from the target domain. Moreover, their methods cannot translate images in the opposite direction as they claim that one seen sample in the target domain is difficult for capturing domain distribution. However, in this work, we focus on solving UI2I given only two unpaired image from two domains and realizing I2I in both directions.

2.2 Image Style Transfer

Image style transfer can be traced back to Hertzmann et al.'s work [10]. More recent approaches use neural networks to learn the style statistics. Gatys et al. [7] first model image style transfer by minimizing the Gram matrix of pre-trained deep features. Luan et al. [28] further propose to realize photorealistic style transfer which can preserve the photorealism of the content image. To avoid inconsistent stylizations in semantically uniform regions, Li et al. [24] introduce a two-step framework in which both steps have a closed-form solution. However, it is difficult for these models to transfer higher-level semantic structures, such as object transformation. We demonstrate that our model can outperform Li et al. [24] in various UI2I tasks.

2.3 Single Image Generative Models

Single image generative models aim to capture the internal distribution of an image. Conditional GAN based models have been proposed for texture expansion [37] and image retargeting [33]. InGAN [33] is trained with a single natural input and learns its internal patch-distribution by an image-specific GAN. Unconditional GAN based models also have been proposed for texture synthesis [2,23,16] and image manipulation [32]. In particular, SinGAN [32] employs an unconditional pyramidal generative model to learn the patch distribution based on images of different scales. However, these single image generative models usually take one image into consideration and do not capture the relationship between two images. In contrast, our model aims to capture the distribution variations between two unpaired images. In this way, our model can transfer an image from a source distribution to a target distribution while maintaining its internal content consistency.

3 Method

Given two images $I_A \in A$ and $I_B \in B$, where A and B are two image domains, our goal is to convert I_A to $I_{AB} \in B$ and I_B to $I_{BA} \in A$ without any other data accessible. Since we have only two unpaired images, the translated result (e.g., I_{AB}) should inherit the domain-invariant features of the source image (e.g., I_A) and replace the domain-specific features with the ones of the target image (e.g.,

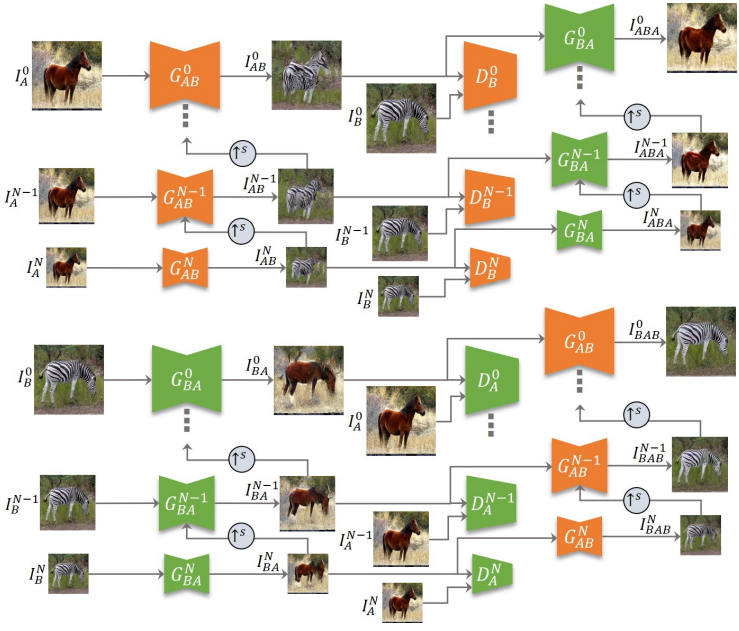

Fig. 2. TuiGAN network architecture: TuiGAN consists of two symmetric pyramids of generators (G^n_{AB}s and G^n_{BA}s) and discriminators (D^n_Bs and D^n_As), $0 \leq n \leq N$. At each scale, the generators take the downsampled source image and previously translated image to generate the new translated image. The discriminators learn the domain distribution by progressively narrowing the receptive fields. The whole framework is learned in a scale-to-scale fashion and the final result is obtained at the finest scale.

I_B) [38,22,13]. To realize such image translation, we need to obtain a pair of mapping functions $G_{AB} : A \mapsto B$ and $G_{BA} : B \mapsto A$, such that

$$I_{AB} = G_{AB}(I_A), \quad I_{BA} = G_{BA}(I_B). \tag{1}$$

Our formulation aims to learn the internal domain distribution variation between I_A and I_B. Considering that the training data is quite limited, G_{AB} and G_{BA} are implemented as two multi-scale conditional GANs that progressively translate images from coarse to fine. In this way, the training data can be fully leveraged at different resolution scales. We downsample I_A and I_B to N different scales, and then obtain $\mathcal{I}_A = \{I^n_A | n = 0, 1, \cdots, N\}$ and $\mathcal{I}_B = \{I^n_B | n = 0, 1, \cdots, N\}$, where I^n_A and I^n_B are downsampled from I_A and I_B, respectively, by a scale factor $(1/s)^n$ ($s \in \mathbb{R}$).

In previous literature, multi-scale architectures have been explored for unconditional image generation with multiple training images [18,19,5,12], conditional image generation with multiple paired training images [34] and image generation with a single training image [32]. In this paper, we leverage the benefit of multi-scale architecture for one-shot unsupervised learning, in which only two unpaired images are used to learn UI2I.

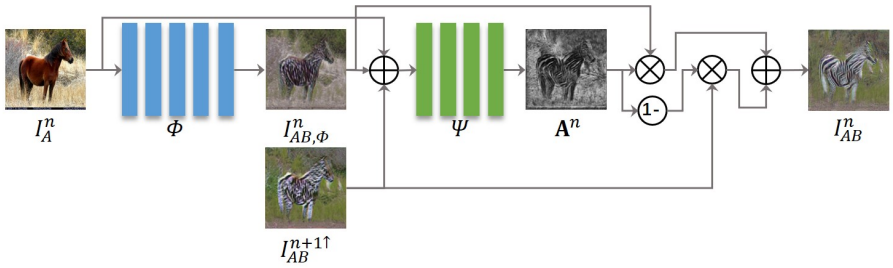

Fig. 3. Architecture of the generator G_{AB}^n, which achieves the $I_A^n \rightarrow I_{AB}^n$ translation. There are two modules, Φ and Ψ. The input I_A^n is first transformerd via Φ to obtain $I_{AB,\Phi}^n$. Then, the transformed $I_{AB,\Phi}^n$, original input I_A^n and the output of previous scale $I_{AB}^{n+1\uparrow}$ are fused by model Ψ to generated a mask $\mathbf{A}^n$. Finally, I_A^n and $I_{AB}^{n+1\uparrow}$ are linearly combined through $\mathbf{A}^n$ to obtain the final output.

3.1 Network Architecture

The network architecture of the proposed TuiGAN is shown in Fig. 2. The entire framework consists of two symmetric translation models: G_{AB} for $I_A \rightarrow I_{AB}$ (the top part in Fig.2) and G_{BA} for $I_B \rightarrow I_{BA}$ (the bottom part in Fig. 2). G_{AB} and G_{BA} are made up of a series of generators, $\{G_{AB}^n\}_{n=0}^N$ and $\{G_{BA}^n\}_{n=0}^N$, which can achieve image translation at the corresponding scales. At each image scale, we also need discriminators D_A^n and D_B^n ($n \in \{0, 1, \cdots, N\}$), which is used to verify whether the input image is a natural one in the corresponding domain.

Progressive Translation The translation starts from images with the lowest resolution and gradually moves to the higher resolutions. G_{AB}^N and G_{BA}^N first map I_A^N and I_B^N to the corresponding target domains:

$$I_{AB}^N = G_{AB}^N(I_A^N);\ I_{BA}^N = G_{BA}^N(I_B^N). \tag{2}$$

For images with scales $n < N$, the generator G_{AB}^n has two inputs, I_A^n and the previously generated I_{AB}^{n+1}. Similarly, G_{BA}^n takes I_B^n and I_{BA}^{n+1} as inputs. Mathematically,

$$I_{AB}^n = G_{AB}^n(I_A^n, I_{AB}^{n+1\uparrow}),\ I_{BA}^n = G_{BA}^n(I_B^n, I_{BA}^{n+1\uparrow}), \tag{3}$$

where $\uparrow$ means to use bicubic upsampling to resize image by a scale factor s. Leveraging I_{AB}^{n+1}, G_{AB}^n could refine the previous output with more details, and I_{AB}^{n+1} also provides the global structure of the target image for current resolution. Eqn.(3) is iteratively applied until the eventual output I_{AB}^0 and I_{BA}^0 are obtained.

Scale-aware Generator The network architecture of G_{AB}^n is shown in Fig. 3. Note that G_{AB}^n and G_{BA}^n shares the same architecture but have different weights. G_{AB}^n consists of two fully convolutional networks. Mathematically, G_{AB}^n works as follows:

$$\begin{aligned} &I_{AB,\Phi}^n = \Phi(I_A^n),\ \mathbf{A}^n = \Psi(I_{AB,\Phi}^n, I_A^n, I_{AB}^{n+1\uparrow}), \\ &I_{AB}^n = \mathbf{A}^n \otimes I_{AB,\Phi}^n + (1 - \mathbf{A}^n) \otimes I_{AB}^{n+1\uparrow}, \end{aligned} \tag{4}$$

where $\otimes$ represents pixel-wise multiplication. As shown in Eqn.(4), we first use Φ to preprocess I_A^n into $I_{AB,\Phi}^n$ as the initial translation. Then, we use an attention model Ψ to generate a mask $\mathbf{A}^n$, which models long term and multi-scale dependencies across image regions [36,30]. Ψ takes $I_{AB,\Phi}^n$, $I_{AB}^{n+1\uparrow}$ and I_A^n as inputs and outputs $\mathbf{A}^n$ considering to balance two scales' results. Finally, $I_{AB,\Phi}^n$ and $I_{AB}^{n+1\uparrow}$ are linearly combined through the generated $\mathbf{A}^n$ to get the output I_{AB}^n.

Similarly, the translation $I_B \rightarrow I_{BA}$ at n-th scale is implemented as follows:

$$\begin{aligned} &I_{BA,\Phi}^n = \Phi(I_B^n);\ \mathbf{A}^n = \Psi(I_{BA,\Phi}^n, I_B^n, I_{BA}^{n+1\uparrow}), \\ &I_{BA}^n = \mathbf{A}^n \otimes I_{BA,\Phi}^n + (1-\mathbf{A}^n) \otimes I_{BA}^{n+1\uparrow}. \end{aligned} \tag{5}$$

In this way, the generator focuses on regions of the image that are responsible of synthesizing details in current scale and keeps the previously learned global structure untouched in the previous scale. As shown in Fig. 3, the previous generator has generated global structure of a zebra in $I_{AB}^{n+1\uparrow}$, but still fails to generate stripe details. In the n-th scale, the current generator generates an attention map to add stripe details on the zebra and produces better result I_{AB}^n.

3.2 Loss Functions

Our model is progressively trained from low resolution to high resolution. Each scale keeps fixed after training. For any $n \in \{0, 1, \cdots, N\}$, the overall loss function of the n-th scale is defined as follows:

$$\mathcal{L}_{\text{ALL}}^n = \mathcal{L}_{\text{ADV}}^n + \lambda_{\text{CYC}}\mathcal{L}_{\text{CYC}}^n + \lambda_{\text{IDT}}\mathcal{L}_{\text{IDT}}^n + \lambda_{\text{TV}}\mathcal{L}_{\text{TV}}^n, \tag{6}$$

where $\mathcal{L}_{\text{ADV}}^n$, $\mathcal{L}_{\text{CYC}}^n$, $\mathcal{L}_{\text{IDT}}^n$, $\mathcal{L}_{\text{TV}}^n$ refer to adversarial loss, cycle-consistency loss, identity loss and total variation loss respectively, and λ_{CYC}, λ_{IDT}, λ_{TV} are hyper-parameters to balance the tradeoff among each loss term. At each scale, the generators aim to minimize $\mathcal{L}_{\text{ALL}}^n$ while the discriminators is trained to maximize $\mathcal{L}_{\text{ALL}}^n$. We will introduce details of these loss functions.

Adversarial Loss The adversarial loss builds upon that fact that the discriminator tries to distinguish real images from synthetic images and generator tries to fool the discriminator by generating realistic images. At each scale n, there are two discriminators D_A^n and D_B^n, which take an image as input and output the probability that the input is a natural image in the corresponding domain. We choose WGAN-GP [9] as adversarial loss which can effectively improve the stability of adversarial training by weight clipping and gradient penalty:

$$\begin{aligned} \mathcal{L}_{\text{ADV}}^n = &D_B^n(I_B^n) - D_B^n(G_{AB}^n(I_A^n)) + D_A^n(I_A^n) - D_A^n(G_{BA}^n(I_B^n)) \\ &- \lambda_{\text{PEN}}(\|\nabla_{\hat{I}_B^n} D_B^n(\hat{I}_B^n)\|_2 - 1)^2 - \lambda_{\text{PEN}}(\|\nabla_{\hat{I}_A^n} D_A^n(\hat{I}_A^n)\|_2 - 1)^2, \end{aligned} \tag{7}$$

where $\hat{I}_B^n = \alpha I_B^n + (1-\alpha) I_{AB}^n$, $\hat{I}_A^n = \alpha I_A^n + (1-\alpha) I_{BA}^n$ with $\alpha \sim U(0,1)$, λ_{PEN} is the penalty coefficient.

Cycle-Consistency Loss One of the training problems of conditional GAN is mode collapse, i.e., a generator produces an especially plausible output whatever

the input is. We utilize cycle-consistency loss [38] to constrain the model to retain the inherent properties of input image after translation: $\forall n \in \{0, 1, \cdots, N\}$,

$$\begin{aligned}
&\mathcal{L}_{\text{CYC}}^{n} = \|I_A^n - I_{ABA}^n\|_1 + \|I_B^n - I_{BAB}^n\|_1, \quad \text{where} \\
&I_{ABA}^n = G_{BA}^n(I_{AB}^n, I_{ABA}^{n+1\uparrow}),\ I_{BAB}^n = G_{AB}^n(I_{BA}^n, I_{BAB}^{n+1\uparrow}), \quad \text{if } n < N; \\
&I_{ABA}^N = G_{BA}^N(I_{AB}^N),\ I_{BAB}^N = G_{AB}^N(I_{BA}^N), \quad \text{if } n = N.
\end{aligned} \tag{8}$$

Identity Loss We noticed that relying on the two losses mentioned above for one-shot image translation could easily lead to color [38] and texture misaligned results. To tackle the problem, we introduce the identity loss at each scale, which is denoted as L_{IDT}^n. Mathematically,

$$\begin{aligned}
&\mathcal{L}_{\text{IDT}}^{n} = \|I_A^n - I_{AA}^n\|_1 + \|I_B^n - I_{BB}^n\|_1, \quad \text{where} \\
&I_{AA}^n = G_{BA}^n(I_A^n, I_{AA}^{n+1\uparrow}),\ I_{BB}^n = G_{AB}^n(I_B^n, I_{BB}^{n+1\uparrow}), \quad \text{if } n < N; \\
&I_{AA}^N = G_{BA}^N(I_A^N), \quad I_{BB}^N = G_{AB}^N(I_B^N), \quad \text{if } n = N.
\end{aligned} \tag{9}$$

We found that identity loss can effectively preserve the consistency of color and texture tone between the input and the output images as shown in Section 4.4.

Total Variation Loss To avoid noisy and overly pixelated, following [29], we introduce total variation (TV) loss to help in removing rough texture of the generated image and get more spatial continuous and smoother result. It encourages images to consist of several patches by calculating the differences of neighboring pixel values in the image. Let $x[i, j]$ denote the pixel located in the i-th row and j-th column of image x. The TV loss at the n-th scale is defined as follows:

$$\begin{aligned}
&\mathcal{L}_{\text{TV}}^{n} = L_{tv}(I_{AB}^n) + L_{tv}(I_{BA}^n), \\
&L_{tv}(x) = \sum_{i,j} \sqrt{(x[i, j+1] - x[i, j])^2 + (x[i+1, j] - x[i, j])^2},\ x \in \{I_{AB}^n, I_{BA}^n\}.
\end{aligned} \tag{10}$$

3.3 Implementation Details

Network Architecture As mentioned before, all generators share the same architecture and they are all fully convolutional networks. In detail, Φ is constructed by 5 blocks of the form 3x3 Conv-BatchNorm-LeakyReLU [14] with stride 1. Ψ is constructed by 4 blocks of the form 3x3 Conv-BatchNorm-LeakyReLU. For each discriminator, we use the Markovian discriminator (PatchGANs) [15] which has the same 11x11 patch-size as Φ to keep the same receptive field as generator.

Training Settings We train our networks using Adam [21] with initial learning rate 0.0005, and we decay the learning rate after every 1600 iterations. We set our scale factor $s = 4/3$ and train 4000 iterations for each scale. The number of scale N is set to 4. For all experiments, we set weight parameters $\lambda_{\text{CYC}} = 1$, $\lambda_{\text{IDT}} = 1$, $\lambda_{\text{TV}} = 0.1$ and $\lambda_{\text{PEN}} = 0.1$.

4 Experiments

We conduct experiments on several tasks of unsupervised image-to-image translation, including the general UI2I tasks[1], image style transfer, animal face translation and paint-to-image translation, to verify our versatile TuiGAN. To construct datasets of one-shot image translation, given a specific task (like horse↔zebra translation [38]), we randomly sample an image from the source domain and the other one from the target domain, respectively, and train models on the selected data.

4.1 Baselines

We compare TuiGAN with two types of baselines. The first type leverages the full training data without subsampling. We choose CycleGAN [38] and DRIT [22] algorithms for image synthesis. The second type leverages partial data, even one or two images only. We choose the following baselines:
(1) OST [1], where one image from the source domain and a set of images in the target domain are given;
(2) SinGAN [32], which is a pyramidal unconditional generative model trained on only one image from the target domain, and injects an image from the source domain to the trained model for image translation.
(3) PhotoWCT [24], which can be considered as a special kind of image-to-image translation model, where a content photo is transferred to the reference photo's style while remaining photorealistic.
(4) FUNIT [27], which targets few-shot UI2I and requires lots of data for pre-training. We test the one-shot translation of FUNIT.
(5) ArtStyle [6], which is a classical art style transfer model.

For all the above baselines, we use their official released code to produce the results.

4.2 Evaluation Metrics

(1) **Single Image Fréchet Inception Distance (SIFID) [32]**: SIFID captures the difference of internal distributions between two images, which is implemented by computing the Fréchet Inception Distance (FID) between deep features of two images. A lower SIFID score indicates that the style of two images is more similar. We compute SIFID between translated image and corresponding target image.
(2) **Perceptual Distance (PD) [17]**: PD computes the perceptual distance between images. A lower PD score indicates that the content of two images is more similar. We compute PD between translated image and corresponding source image.
(3) **User Preference (UP)**: We conduct user preference studies for performance evaluation since the qualitative assessment is highly subjective.

[1] In this paper, we refer to general UI2I as tasks where there are multiple images in the source and target domains, i.e., the translation tasks studied in [38].

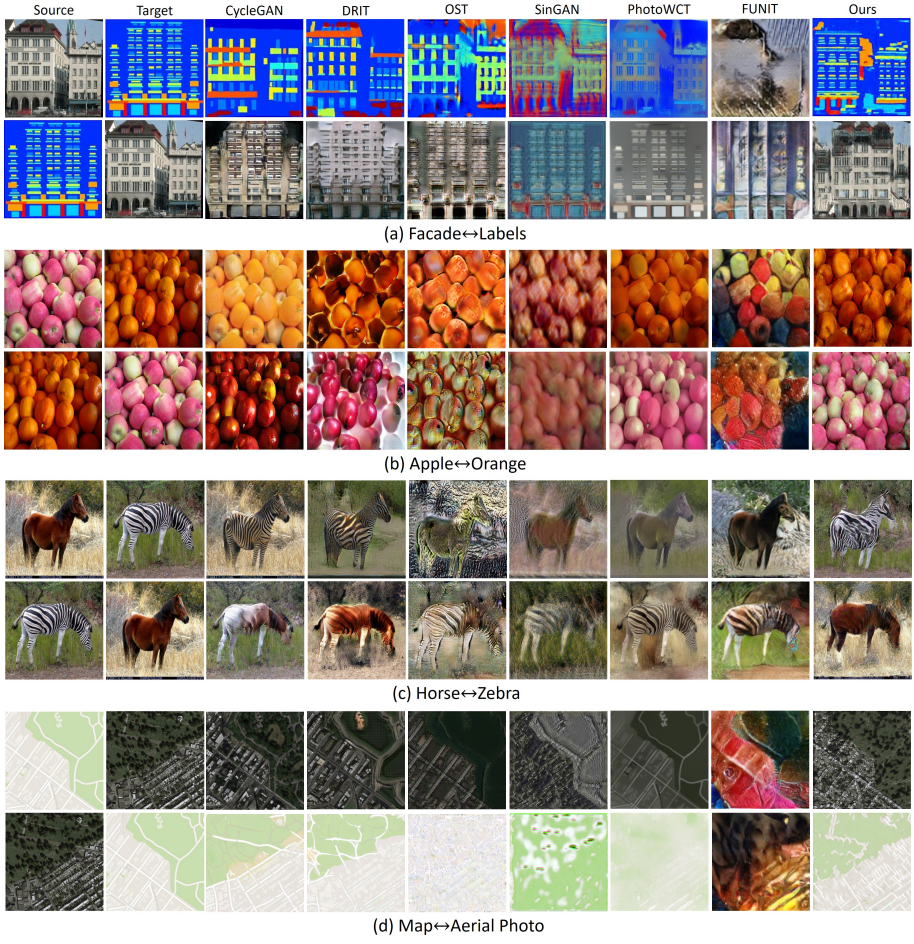

Fig. 4. Results of general UI2I tasks using CycleGAN (trained with full training dataset), DRIT (trained with the full training dataset), OST (trained with 1 sample in the source domain and full data in the target domain), SinGAN (trained with one target image), PhotoWCT (trained with two unpaired images), FUNIT (pre-trained) and our TuiGAN (trained with two unpaired images).

4.3 Results

General UI2I Tasks Following [38], we first conduct general experiments on Facade↔Labels, Apple↔Orange, Horse↔Zebra and Map↔Aerial Photo translation tasks to verify the effectiveness of our algorithm. The visual results of our proposed TuiGAN and the baselines are shown in Fig. 4.

Overall, the images generated by TuiGAN exhibit better translation quality than OST, SinGAN, PhotoWCT and FUNIT. While both SinGAN and PhotoWCT change global colors of the source image, they fail to transfer the high-

Table 1. Average SIFID, PD and UP across different general UI2I tasks.

Metrics	CycleGAN	DRIT	OST	SinGAN	PhotoWCT	FUNIT	Ours
SIFID ($\times 10^{-2}$)	0.091	0.142	0.123	0.384	717.622	1510.494	0.080
PD	5.56	8.24	10.26	7.55	3.27	7.55	7.28
UP	61.45%	52.08%	26.04%	6.25%	25.00%	2.08%	-

level semantic structures as our model (e.g., in Facade↔Labels and Horse↔Zebra). Although OST is trained with the full training set of the target domain and transfers high-level semantic structures in some cases, the generated results contain many noticeable artifacts, e.g., the irregular noises on apples and oranges. Compared with CycleGAN and DRIT trained on full datasets, TuiGAN achieves comparable results to them. There are some cases that TuiGAN produces better results than these two models in Labels→Facade, Zebra→Horse tasks, which further verifies that our model can actually capture domain distributions with only two unpaired images.

The results of average SIFID, PD and UP are reported in Table 1. For user preference study, we randomly select 8 unpaired images, and generate 8 translated images for each general UI2I task. In total, we collect 32 translated images for each subject to evaluate. We display the source image, target image and two translated images from our model and another baseline method respectively on a webpage in random order. We ask each subject to select the better translated image at each page. We finally collect the feedback from 18 subjects of total 576 votes and 96 votes for each comparison. We compute the percentage from a method is selected as the User Preference (UP) score.

We can see that TuiGAN obtains the best SIFID score among all the baselines, which shows that our model successfully captures the distributions of images in the target domain. In addition, our model achieves the third place in PD score after CycleGAN and PhotoWCT. From the visual results, we can see that PhotoWCT can only change global colors of the source image, which is the reason why it achieves the best PD score. As for user study, we can see that most of the users prefer the translation results generated by TuiGAN than OST, SinGAN, PhotoWCT and FUNIT. Compared with CycleGAN and DRIT trained on full data, our model also achieves similar votes from subjects.

Image Style Transfer We demonstrate the effectiveness of our TuiGAN on image style transfer: art style transfer, which is to convert image to the target artistic style with specific strokes or textures, and photorealistic style transfer, which is to obtain stylized photo that remains photorealistic. Results are shown in Fig. 5. As can be seen in the first row of Fig.5, TuiGAN retains the architectural contour and generates stylized result with vivid strokes, which just looks like Van Goghs painting. Instead, SinGAN fails to generate clear stylized image, and PhotoWCT [24] only changes the colors of real photo without capturing the salient painting patterns. In the second row, we transfer the night image to

Fig. 5. Results of image style transfer. The first row represents the results of art style transfer, and the second row is the results of photorealistic style transfer. We amplify the green boxes in photorealistic style transfer results at the third row to show more details.

Fig. 6. Results of animal face translation. Our model can accurately transfer the fur colors, while FUNIT, a model pre-trained on animal face dataset, does not work as well as our model.

photorealistic day image with the key semantic information retained. Although SinGAN and ArtStyle produce realistic style, they fail to the maintain detailed edges and structures. The result of PhotoWCT is also not as clean as ours. Overall, our model achieves competitive performance on both types of image style transfer, while other methods usually can only target on a specific task but fail in another one.

Animal Face Translation To compare with the few-shot model FUNIT, which is pretained on animal face dataset, we conduct the animal face translation experiments as shown in Fig.6. We also include SinGAN and PhotoWCT for comparison. As we can see, our model can better transfer the fur colors from image in the target domain to the that of the source domain than other baselines: SinGAN [32] generates results with faint artifacts and blurred dog shape; PhotoWCT [24] can not transfer high-level style feature (e.g. spots) from the target image although it preserves the content well; and FUNIT generates results that are not consistent with the target dog's appearance.

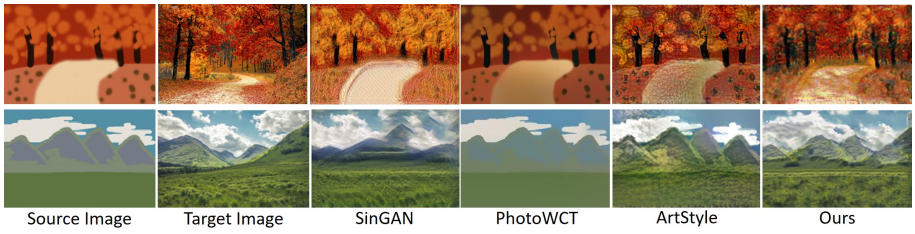

Fig. 7. Results of painting-to-image translation. TuiGAN can translate more specific style patterns of the target image (e.g., leaves on the road in the first row) and maintain more accurate content of the source images (e.g., mountains and clouds in the second row).

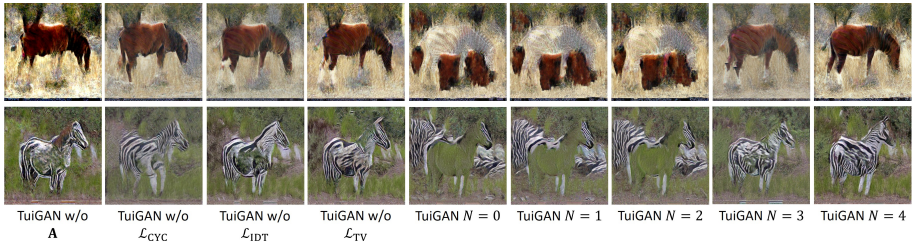

Fig. 8. Visual results of ablation study.

Painting-to-Image Translation This task focuses to generate photo-realistic image with more details based on a roughly related clipart as described in SinGAN [32]. We use the two samples provided by SinGAN for comparison. The results are shown in Fig.7. Although two testing images share similar elements (e.g., trees and road), their styles are extremely different. Therefore, PhotoWCT and ArtStyle fail to transfer the target style in two translation cases. SinGAN also fails to generate specific details, such as leaves on the road in the first row of Fig.7, and maintain accurate content, such as mountains and clouds in the second row of Fig.7. Instead, our method preserves the crucial components of input and generates rich local details in two cases.

4.4 Ablation Study

To investigate the influences of different training losses, generator architecture and multi-scale structure, we conduct several ablation studies based on Horse↔Zebra task. Specifically,
(1) Fixing $N = 4$, we remove the cycle-consistent loss (TuiGAN w/o L_{CYC}), identity loss (TuiGAN w/o L_{IDT}), total variation loss (TuiGAN w/o L_{TV}) and compare the differences;
(2) We range N from 0 to 4 to see the effect of different scales. When $N = 0$, our model can be roughly viewed as the CycleGAN [38] that is trained with two unpaired images.

Table 2. Quantitative comparisons between different variants of TuiGAN in terms of SIFID and PD scores. The best scores are in bold.

Metrics	w/o **A**	w/o $\mathcal{L}_{\text{CYC}}$	w/o $\mathcal{L}_{\text{IDT}}$	TuiGAN w/o $\mathcal{L}_{\text{TV}}$	$N=0$	$N=1$	$N=2$	$N=3$	$N=4$
SIFID ($\times 10^{-4}$) Horse→Zebra	1.08	3.29	2.43	2.41	2.26	2.32	2.31	2.38	**1.03**
SIFID ($\times 10^{-4}$) Zebra→Horse	2.09	5.61	5.54	10.85	3.75	3.86	3.77	6.30	**1.79**
PD Horse→Zebra	8.00	6.98	8.24	6.90	6.40	6.82	6.76	6.25	**6.16**
PD Zebra→Horse	10.77	7.92	8.00	6.48	7.77	7.92	8.68	6.87	**5.91**

(3) We remove the attention model Ψ in the generators, and combine $I^n_{AB,\Phi}$ and $I^{n+1\uparrow}_{AB}$ by simply addition (briefly denoted as TuiGAN w/o **A**).

The qualitative results are shown in Fig.8. Without L_{IDT}, the generated results suffers from inaccurate color and texture (e.g., green color on the transferred zebra). Without attention mechanism or L_{CYC}, our model can not guarantee the completeness of the object shape (e.g., missed legs in the transferred horse). Without L_{TV}, our model produces images with artifacts (e.g., colour spots around the horse). The results from $N = 0$ to $N = 3$ either have poor global content information contained (e.g. the horse layout) or have obvious artifacts (e.g. the zebra stripes). Our full model (TuiGAN $N = 4$) could capture the salient content of the source image and transfer remarkable style patterns of the target image.

We compute the quantitative ablations by assessing SIFID and PD scores of different variants of TuiGAN. As shown in Table 2, our full model still obtains the lowest SIFID score and PD score, which indicates that our TuiGAN could generate more realistic and stylized outputs while preserving the content unchanged.

5 Conclusion

In this paper, we propose TuiGAN, a versatile conditional generative model that is trained on only two unpaired image, for image-to-image translation. Our model is designed in a coarse-to-fine manner, in which two pyramids of conditional GANs refine the result progressively from global structures to local details. In addition, a scale-aware generator is introduced to better combine two scales' results. We validate the capability of TuiGAN on a wide variety of unsupervised image-to-image translation tasks by comparing with several strong baselines. Ablation studies also demonstrate that the losses and network scales are reasonably designed. Our work represents a further step toward the possibility of unsupervised learning with extremely limited data.

References

1. Benaim, S., Wolf, L.: One-shot unsupervised cross domain translation. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. pp. 2108–2118. Curran Associates Inc. (2018)
2. Bergmann, U., Jetchev, N., Vollgraf, R.: Learning texture manifolds with the periodic spatial gan. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 469–477. JMLR. org (2017)
3. Choi, Y., Uh, Y., Yoo, J., Ha, J.W.: Stargan v2: Diverse image synthesis for multiple domains. arXiv preprint arXiv:1912.01865 (2019)
4. Cohen, T., Wolf, L.: Bidirectional one-shot unsupervised domain mapping. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1784–1792 (2019)
5. Denton, E.L., Chintala, S., Fergus, R., et al.: Deep generative image models using a laplacian pyramid of adversarial networks. In: Advances in neural information processing systems. pp. 1486–1494 (2015)
6. Gatys, L.A., Ecker, A.S., Bethge, M.: A neural algorithm of artistic style. Nature Communications (2015)
7. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2414–2423 (2016)
8. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
9. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: Advances in Neural Information Processing Systems. pp. 5767–5777 (2017)
10. Hertzmann, A.: Painterly rendering with curved brush strokes of multiple sizes. In: Proceedings of the 25th annual conference on Computer graphics and interactive techniques. pp. 453–460 (1998)
11. Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. In: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. pp. 327–340 (2001)
12. Huang, X., Li, Y., Poursaeed, O., Hopcroft, J., Belongie, S.: Stacked generative adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5077–5086 (2017)
13. Huang, X., Liu, M.Y., Belongie, S., Kautz, J.: Multimodal unsupervised image-to-image translation. In: ECCV (2018)
14. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning. pp. 448–456 (2015)
15. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1125–1134 (2017)
16. Jetchev, N., Bergmann, U., Vollgraf, R.: Texture synthesis with spatial generative adversarial networks. arXiv preprint arXiv:1611.08207 (2016)
17. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)

18. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)
19. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4401–4410 (2019)
20. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning. pp. 1857–1865 (2017)
21. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
22. Lee, H.Y., Tseng, H.Y., Huang, J.B., Singh, M.K., Yang, M.H.: Diverse image-to-image translation via disentangled representations. In: European Conference on Computer Vision (2018)
23. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: European conference on computer vision. pp. 702–716. Springer (2016)
24. Li, Y., Liu, M.Y., Li, X., Yang, M.H., Kautz, J.: A closed-form solution to photorealistic image stylization. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 453–468 (2018)
25. Lin, J., Xia, Y., Qin, T., Chen, Z., Liu, T.Y.: Conditional image-to-image translation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)(July 2018). pp. 5524–5532 (2018)
26. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Advances in Neural Information Processing Systems. pp. 700–708 (2017)
27. Liu, M.Y., Huang, X., Mallya, A., Karras, T., Aila, T., Lehtinen, J., Kautz, J.: Few-shot unsupervised image-to-image translation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 10551–10560 (2019)
28. Luan, F., Paris, S., Shechtman, E., Bala, K.: Deep photo style transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4990–4998 (2017)
29. Mahendran, A., Vedaldi, A.: Understanding deep image representations by inverting them. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5188–5196 (2015)
30. Pumarola, A., Agudo, A., Martinez, A.M., Sanfeliu, A., Moreno-Noguer, F.: Ganimation: Anatomically-aware facial animation from a single image. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 818–833 (2018)
31. Rosales, R., Achan, K., Frey, B.J.: Unsupervised image translation. In: iccv. pp. 472–478 (2003)
32. Shaham, T.R., Dekel, T., Michaeli, T.: Singan: Learning a generative model from a single natural image. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4570–4580 (2019)
33. Shocher, A., Bagon, S., Isola, P., Irani, M.: Ingan: Capturing and remapping the" dna" of a natural image. arXiv preprint arXiv:1812.00231 (2018)
34. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8798–8807 (2018)
35. Yi, Z., Zhang, H., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. In: Proceedings of the IEEE international conference on computer vision. pp. 2849–2857 (2017)

36. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. In: International Conference on Machine Learning. pp. 7354–7363 (2019)
37. Zhou, Y., Zhu, Z., Bai, X., Lischinski, D., Cohen-Or, D., Huang, H.: Non-stationary texture synthesis by adversarial expansion. ACM Transactions on Graphics (TOG) **37**(4), 1–13 (2018)
38. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)

6 Additional Results

In this section, we provide additional results of four general unpaired image-to-image translation tasks: Apple↔Orange in Fig. 9, Horse↔Zebra in Fig. 10, Facade↔Labels in Fig. 11, and Map↔Aerial Photo in Fig. 12. From the additional results provided, we can further verify that most of the translation results generated by TuiGAN are better than OST, SinGAN, PhotoWCT and FUNIT. Compared with CycleGAN and DRIT trained on full data, our model can also achieve comparable performance in many cases.

Fig. 9. The top three rows are Apple→Orange results and the bottom three rows are Orange→Apple results.

Fig. 10. The top three rows are Horse→Zebra results and the bottom three rows are Zebra→Horse results.

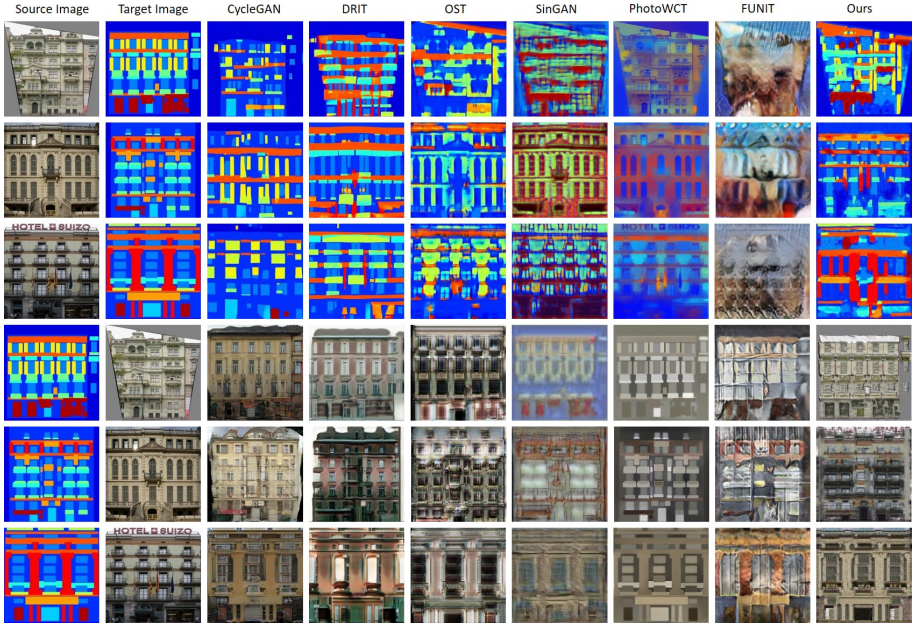

Fig. 11. The top three rows are Facade→Labels results and the bottom three rows are Labels→Facade results.

Fig. 12. The top three rows are Map→Aerial Photo results and the bottom three rows are Aerial Photo→Map results.