Attribute-Driven Spontaneous Motion in Unpaired Image Translation

Ruizheng Wu\textsuperscript{1} Xin Tao\textsuperscript{2} Xiaodong Gu\textsuperscript{3} Xiaoyong Shen\textsuperscript{2} Jiaya Jia\textsuperscript{1,2}
\textsuperscript{1}The Chinese University of Hong Kong \textsuperscript{2}Tencent YouTu Lab \textsuperscript{3}Harbin Institute of Technology, Shenzhen
\{rzwu, leojia\}@cse.cuhk.edu.hk \{xintao, dylanshen\}@tencent.com guxiaodong@stu.hit.edu.cn

Abstract

Current image translation methods, albeit effective to produce high-quality results in various applications, still do not consider much geometric transform. We in this paper propose the spontaneous motion estimation module, along with a refinement part, to learn attribute-driven deformation between source and target domains. Extensive experiments and visualization demonstrate effectiveness of these modules. We achieve promising results in unpaired-image translation tasks, and enable interesting applications based on spontaneous motion.

1. Introduction

High-quality image generation is a fascinating task and has gained much attention in computer vision community. There has been great progress using generative adversarial networks (GAN)\textsuperscript{[9, 32]}. Image translation, which produces modified images in target domain based on a given input from source domain, has been widely used in applications of style transfer\textsuperscript{[22]}, sketch/photo conversion\textsuperscript{[3, 15]}, label-based image synthesis\textsuperscript{[31]}, face editing\textsuperscript{[39]} etc. Recent research trends continuously towards high practicality, e.g., images in high resolutions\textsuperscript{[36]}, and unpaired image translation\textsuperscript{[42]}, or using better latent space for more effective control\textsuperscript{[13, 21]}

Image translation mostly imposes the requirement of aligned or similar domains for texture or appearance transform. For example, in style transfer, the output image generally shares the same content with input. The building blocks of these networks, such as convolution/deconvolution layers and activation functions, are spatially corresponding. As shown in Fig.\textsuperscript{[1]}(a)-(d), visual artifacts, such as ghosting, could appear when non-smile and smile faces are not geometrically aligned in image space.

In this paper, we take advantage of geometric correspondence in appearance transform. Taking the smiling face as an example in Fig.\textsuperscript{[1]}(e)-(f), decent results can be produced by applying geometric transform only, and they are better when a small refinement network follows. This architecture can greatly reduce visual artifacts mentioned above. This paper tackles the following three issues.

Single Image Deformation Although several previous methods\textsuperscript{[37, 40, 35]} employ various flow/warping/deformation by estimating motion, they are different from what we need in two aspects. First, traditional motion field estimation requires a pair of images to construct dense pixel correspondence, while in our task only one image is available. Second, motion fields for deformation are conditioned by examples, where one input image may need various motion constrained by target examples.

Different from all these settings, our goal is more like estimation of natural tendency of input images. We term it spontaneous motion (SPM) to distinguish from ordinary optical flow. This new tool adds a new dimension to image translation by introducing unpaired geometric transform. It also enables new ways of visualization, and finds interesting applications (described in Sec.\textsuperscript{4.1}). For example, in our framework, SPM for different target domains can be viewed as motion basis (Fig.\textsuperscript{[5]}), and linearly combining SPM basis...
enables convenient geometric edit (Fig. 7).

**High Ill-posedness** Our framework is trained with neither paired data nor ground-truth motion field across domains. Cycle reconstruction loss and learning common latent space were considered to deal with unpaired data. [32, 33]. Our geometric transform estimation across domains is even more ill-posed, since this set-to-set motion is more ambiguous compared to image-to-image correspondence given no ground-truth motion. Our spontaneous motion module applies two domain classifiers for translation result generation and motion estimation.

**Inevitable Errors** Estimated motion fields are inevitably with errors due to large prediction freedom, missing-motion area to be filled (teeth of smiling faces are area missing in non-smile faces), and fine texture requirement for high-quality results. Our system has a refinement module to fix remaining visual artifacts with an attention mask to filter out unnecessary changes on original images.

Our contributions are as follows. 1) We propose an end-to-end unpaired image translation system considering geometric deformation. 2) A conditional spontaneous motion estimation module, along with domain classifiers and a refinement step, to boost performance. 3) Our new framework achieves promising results in image translation, especially for unaligned scenarios.

## 2. Related Work

### Unpaired Image Translation

Several unsupervised image translation methods were proposed. By introducing cycle-consistency loss for reconstruction, methods of [32, 31, 35] train the translation network across two domains without paired data. They train two separate networks for bidirectional image generation between source and target domains. Methods of [4, 30] extend the framework by introducing additional conditions to generate images in multiple domains. Another stream of research [23, 24, 34, 38, 28] is based on the assumption that images in source and target domains share the same latent space and in [13, 21], style and content are disentangled to control generated image style. They successfully translate images across domains. Because geometric relationship between domains is not considered, data that is not aligned or structurally very different cannot be well dealt with.

### Geometry-Aware Image Translation

There exists work to build geometric relationship during image translation/generation. In [8, 21], geometric inconsistency between domains is mitigated with designed discriminator or losses. We note the generators are still composed of convolution-based blocks, which limit the generation power. Methods of [6, 37, 7, 40] estimate correspondence between two images. Dong et al. [6] relied on human body parsing, while Geng et al. [7] generated dense correspondence based on face landmarks. Methods of [37, 40] directly learn dense correspondence between two images. For these methods, paired reference images are needed to train or test, which does not fit semantic-level set-to-set transformation. Cao et al. [1] added another network for landmark learning; it cannot be trained in an end-to-end manner.

Our method is different. We do not need reference images and our framework is designed in an end-to-end way. Besides, the estimated spontaneous motion is conditioned on source domain content and target domain attributes, which can achieve semantic-level geometric transformation and generation.

## 3. Proposed Method

Given an image in source domain $I_s \in \mathbb{R}^{H \times W \times 3}$ and target domain indicator $c_t \in \{0, 1\}^N$ ($N$ is the total number of attributes), e.g. smiling, angry, and surprising. Our goal is to generate a high-quality image $I_t$ with attribute $c_t$ while keeping the identity of $I_s$. Our framework resembles previous generative models by iteratively training generator/discriminator networks. However, in order to better handle geometric transform, we incorporate two new modules in generator $G$ as spontaneous motion module $SPM$ (Sec. 3.1) and refinement module $R$ (Sec. 3.2). Two types of classifiers are proposed as new losses to facilitate training. We extend our framework to high-resolution image generation ($512 \times 512$) with special designs (Sec. 3.3). Our overall framework is depicted in Fig. 2. We elaborate on each module in the following.

### 3.1. Spontaneous Motion Module

According to the analysis in Sec. 1, our spontaneous motion module aims to predict motion field $w$ based on input image $I_s$ and target indicator $c_t$. We use an encoder-decoder network structure for its powerful fitting ability. For the design of this module, we consider the following facts.

#### Motion Field Decoding

In conventional image regression problems [14, 35, 8], activation functions in the final layer is usually not applied, in order to leave the output unbounded. This is because ground-truth pixel values are always in range $[0, 1]^{H \times W \times 3}$, which supervise and prevent network output from divergence. In our motion estimation task, contrarily the output motion values can be largely varied, and the network can only be trained under indirect supervision, making convergence an issue, as verified in our experiments.

In order to mitigate this problem, we utilize $\tanh(\cdot)$ as the last activation function to limit output in $[-1, 1]$ rather than $[-\infty, +\infty]$. Moreover, we introduce an empirical multiplier $\lambda_w$ to get the final $w$. This seemingly tricky coefficient is actually quite reasonable in many tasks, such as face edit-
and 6 residual blocks to extract layers (each followed by instance normalization and ReLU).

For deforming, we construct the encoder with 3 stride-2 convolution layers to a 2-channel motion field with the map $\tilde{f}$ where the deformation step with bilinear interpolation as $V$. Coordinates as $i^s(i)$, we formulate the deformation step with bilinear interpolation as

$$I_d(i) = \begin{cases} \tilde{I}_s(i + w(i)), & \text{if } i + w(i) \in V, \\ 0, & \text{otherwise.} \end{cases}$$

where $\tilde{I}_s(i)$ is bilinear interpolation operator.

As for the network structure of spontaneous motion module, we construct the encoder with 3 stride-2 convolution layers (each followed by instance normalization and ReLU) and 6 residual blocks to extract $8 \times \downarrow$ down-sampled feature map $f$. A decoder then processes and up-samples $f$ by 3 deconvolution layers to a 2-channel motion field with the same size as the input image. In addition, high-level feature $f$ is utilized for attention mask learning, which will be described in Sec. 3.2.

To generate different dense motion fields for target indicators, we design two classifiers as constraints for both generation results and motion field estimation.

**Domain Classifiers** We design image and motion domain classifiers in training. For image classifier $D_c$, like that of [4], we add $D_c$ on top of discriminator $D$ as a constraint to classify the generated images into target domain $c$. During training on $D$, real image $I_s$ and its attribute $c_s$ are utilized to train $D_c$ with loss $L^c_{cr}$. At the stage of training generator $G$, $D_c$ is fixed and the classification loss $L^g_{cr}$ of generated images is utilized to optimize $G$. The losses $L^d_{cr}$ and $L^g_{cr}$ are defined as

$$L^d_{cr} = \mathbb{E}_{I_s,c_s}[-\log D_c(c_s | I_s)]$$

$$L^g_{cr} = \mathbb{E}_{I_s,c_s}[-\log D_c(c_t | G(I_s, c_t))]$$

Although the classifier for generation results can guide the prediction of motion, we note a constraint on motion helps it more directly and better – motion fields for one domain (such as a face expression exemplified later in Fig. 5) have common features. It does not vary much even with different input images. With this observation, we design a classifier $D_w$ for motion fields, which classifies different motion fields into categories according to the target condition. It makes motion under different conditions share similar patterns and thus reduces bias or noise in generation steps. The classification loss for this classifier is formulated as

$$L_{cw} = \mathbb{E}_{I_s,c_t}[-\log D_w(c_t | SPM(I_s, c_t), I_s)].$$

**3.2. Refinement Module**

The deformed image $I_d$ is further refined to reduce artifacts and enhance textural details. Specifically, two components are used.

Figure 2. Our overall framework. Our generator $G$ contains spontaneous motion module $SPM$ and refinement module $R$. Two domain classifiers $D_w$ and $D_c$ are utilized to drive generation of final results and motion fields under different conditions, while $D_d$ is utilized to distinguish real images from fake ones.
Refinement with Residual Learning. We employ a refinement sub-network after deforming image $I_d$. Instead of directly learning in images space, we learn the residual $r$ between deformed image and the unknown target, i.e. $I_t = I_d + r$, since to learn residual for a well deformed image is easier and more reliable. As for the network structure, $n$ residual blocks [10] are sequentially concatenated without any downsample operation. The residual blocks are used for finer structure update. Thus we do not shrink images spatially, and instead take multiple stacked residual blocks to ensure final effect. In our experiments, we set $n=12$ to balance performance and efficiency.

Attention Mask. In image translation, generally only essential regions need to be updated (e.g. only mouth and its surrounding are changed when transforming neutral faces to smiling ones). We propose learning an attention mask $m$, which marks important regions. The results are denoted as $I_t = I_d + r \cdot m$.

Specifically, as mentioned in Sec. 3.1, we obtain down-sampled feature map $f$ in module $SPM$. $f$ catches high-level semantic information. We utilize it for attention mask learning. We build the attention mask module $M$ with 3 de-convolutional layers to up-sample $f$ into a 1-channel mask $m$, the same size as the input. $Sigmoid$ layer is used as the final activation layer to range the output mask in $[0, 1]$.

Directly learning an attention mask without any additional constraints is difficult, due to possible trivial solution of a mask with all-region selected. To avoid this problem, we introduce a regularization term $L_m$ to enforce sparsity of masks in $L_1$-norm:

$$L_m = \frac{1}{CHW} \sum_{i=1}^{C} \sum_{j=1}^{H} \sum_{k=1}^{W} |m_{ijk}|,$$

where $C$, $H$, and $W$ are channel number, height and width of the mask respectively. The loss forces the attention mask to focus on the most important region.

3.3. Higher Resolution

To generate high resolution (HR) image by a single generator is difficult. Previous work [16] [23] [17] adopted coarse-to-fine or multi-stage training strategies. By incorporating motion estimation and refinement modules, we extend previous coarse-to-fine strategies to a pipeline with extra priors.

Priors and Adaptation. Previous coarse-to-fine strategies are usually applied to final output, i.e. using generated low-res (LR) images to guide HR generation. In our framework, we have more useful clues from LR, i.e. motion field $w^l$ for deformation, residual $r^l$ for refinement, mask $m^l$ for attention in low-resolution form. We utilize them to facilitate HR result generation.

We first train our initial framework with LR images $I^l_s$ until convergence. For higher resolution results, we feed in HR image $I^h_s$ and start from the well-trained LR framework while the weight of LR model is updated simultaneously in this stage. After obtaining motion field $w^l$, residual $r^l$, attention mask $m^l$ from LR generator with down-sampled $I^l_s$ from $I^h_s$, we up-sample them to produce coarse results, i.e. $U(w^l)$, $U(r^l)$ and $U(m^l)$ with the same size as HR images $I^h_s$. We further incorporate three light-weighted enhancement networks ($T_u$, $T_r$ and $T_m$) respectively, each only contains two convolutional layers and a residual block. Finally, we estimate motion field $w^h$ as

$$w^h = U(w^l) + T_u(U(w^l)).$$

Bilinear upsampling is used with the same process to obtain residual $r^h$ and attention mask $m^h$. With these intermediate results, we deform $I^h_s$ by $w^h$ to get $I^h_s$ and then refine $I^h_s$ to yield final output $I^h = I^h_s + r^h \cdot m^h$.

Resolution Adaptive Discriminator. During training, the discriminators are designed as follows. In the LR-image training stage, we only train the LR image discriminator $D^l$. We set real image $I^l_s$ as the positive sample while the generated $I^l_s$ is the negative one. In the HR image training stage, for $D^h$, we have down-sampled $I^l_s$ as the positive sample and generated LR image $I^l_s$ as the negative one. Besides, we down-sample generated HR image $I^h_s$ to LR and feed them to $D^l$ as another type of negative samples. As for $D^h$, $I^h_s$ and final generation result $I^h$ are positive and negative samples respectively. $D^h$ share similar network structure as $D^l$, and yet with more convolution layers.

3.4. Other Loss Functions

Adversarial Loss. Ordinary generative adversarial loss is set for $G$ and $D_d$ formulated as

$$L_{adv} = \mathbb{E}_{I_s}[\log D_d(I_s)] + \mathbb{E}_{I_s,c_s}[\log (1 - D_d(G(I_s,c_s)))]$$

Reconstruction Loss. Similar to [4] [42], we reconstruct images in cycle flow. With source image $I_s$, generated image $I_t$, and source image attribute $c_s$, we formulate the reconstruction loss $L_{rec}$ as

$$L_{rec} = \mathbb{E}_{I_s,c_s}||I_s - G(I_s, c_s)||_1.$$

Total Loss. The final loss function for generator $G$ is

$$L_g = \lambda_{cr} \cdot L_{cr} + \lambda_{cw} \cdot L_{cw} + \lambda_{m} \cdot L_m + \lambda_{adv} \cdot L_{adv} + \lambda_{rec} \cdot L_{rec}$$

The loss function for $D$ is

$$L_d = \lambda_{cr} \cdot L_{cr} - \lambda_{adv} \cdot L_{adv}.$$
4. Experiments

We conduct experiments on both CelebA [26] and RaFD [20]. CelebA contains 200K celebrity images and 40 attributes for each image with resolution 218 × 178. We utilize CelebA-HQ [16] in resolution 1024 × 1024 for high-res image usage. To demonstrate the effectiveness of our framework, we select attributes with geometric deformation, i.e. ‘Smiling’, ‘Arched_eyebrow’, ‘Big_Nose’, and ‘Pointy_nose’ as condition to train our framework. RaFD is a smaller dataset with 67 identities, each displaying 8 emotional expressions, 3 gaze directions and 5 camera angles. We only train on frontal faces for robustness.

We implement the system on PyTorch [29] and run it on a TITAN Xp card. During our two-stage training, we first train on LR framework with 128 × 128 images and batch size 16 for 1 × 10^5 iterations. Then we train our extended network on higher resolutions 256 × 256 or 512 × 512 with batch size 8 for another 2 × 10^5 iterations. We use Adam [19] with learning rate 1e-4 to optimize our framework.

4.1. Analysis

Effectiveness of SPM Module  We first visualize learned SPM. We experiment with an extreme case to learn image translation between a set of squares and circles. The position, color and size are random. The results in Fig. 5 demonstrate that our SPM module produces reasonable shapes. Remaining visual artifacts are further reduced by the refinement module.

Roles of Different Modules  For the spontaneous motion module, we aim to generate reasonable geometric movement, e.g. lips stretched for smiling faces. For the refinement module, it further suppresses noise and adds more texture on deformation results to make images look more realistic. A few intermediate and final ‘smiling’ results produced from these modules under different resolutions (128 × 128 to 256 × 256) are shown in Fig. 6. Effects from these two stages are clearly and respectively demonstrated.

Besides, to further study the roles of different modules, we train our framework with no spontaneous motion module (No_M) and no refinement module (No_R) respectively to see how results are altered. We show results in Fig. 4 and the quantitative comparison in Tab. 1. Without motion estimation, the geometric shape of images are wrong. The effect is like pasting patterns from the target domain to specific regions. Without the final refinement, results may contain distortions (right face in the 1st example) and artifacts (nose in the 2nd sample). Images also lack details to be a smiling face.

These experiments manifest the usefulness of both modules and our framework leverages their advantages.

SPM Field in Different Conditions  Motion patterns for the same face expression are generally similar even with different input images. For example, non-smiling to smiling faces need to ‘stretch’ pixels of lips. Taking the RaFD dataset as an example, motion fields for different emotions are visualized in Fig. 5. They tell different parts of faces required to be updated to achieve ideal facial expression.

Spontaneous Motion Field Basis Combination  Transformation varies when applying different motions to the same image. Since motion fields are independent, we can combine motion fields with simple addition operations. By adding differently learned motion fields, we achieve rough expression combination without re-training the network, under the condition that the combined transformation does not conflict with each other. We demonstrate the effect of combination in Fig. 7. The difference in generated micro-expression is very useful for fine face attribute creation.

4.2. Comparisons

4.2.1 Visual Comparison

We conduct experiments on the two datasets for comparative evaluation. On the CelebA dataset, we treat each attribute $x$ transformation as a two-domain translation from non-$x$ to $x$. Fig. 8 shows that CycleGAN, MUNIT, and Ganimorph cannot capture domain information when the attribute transformation is subtle, like ‘Big_Nose’ and ‘Pointy_Nose’. They tend to reconstruct the input image instead. Both StarGAN and our method handle such subtle domain translation thanks to the domain classifiers.

Our method better tackles geometry variation and image misalignment. For other attributes like ‘smiling’, though all previous methods transform source images to target domain, various types of geometric deformation lead to quality difference on results, causing noticeable ghosting or artifacts. Our method, contrarily, alleviates this issue.

For the RaFD dataset (Fig. 9), similarly, StarGAN handles domain transformation and yet are with room to improve details and geometric shapes, especially for the ‘happy’ expression. Our framework satisfies target conditions better thanks to our explicit spontaneous motion module and our two domain classifiers for training.

4.2.2 Quantitative Comparison

Distribution Discrepancy To evaluate generated faces quantitatively, we extract features with a deep face feature extractor VGGFace2 [2] and use FID [12] to measure feature distribution discrepancy between real and generated faces. For each attribute, we first extract feature $F_r$ from
Figure 7. Motion field basis combination. First row: motion fields under different conditions. Second row: deformation results by applying corresponding motion fields. \( M_s \): ‘smiling’ transform, \( M_e \): ‘arched_eyebrow’ transform, \( M_n \): ‘pointy_nose’ transform. \( M_s + M_e \), \( M_s + M_n \), \( M_s + M_e + M_n \) are with two or three corresponding motion field combination.

Figure 8. Visual quality comparison on the CelebA dataset.

We calculate FID between \( F_r \) and \( F_{gi} \) for each method. Results in Tab.1 demonstrate that our framework achieves the lowest FID score among all methods, which indicates that the feature distribution of our generated images is closest to that of real images.

Classification Accuracy  Following [4], we compute the classification accuracy of facial expression on generated images. We first train a facial expression classifier with real faces with such an attribute in test set, and then extract features \( F_{gi} \) from translated images (to this attribute) by each method to be compared.
Figure 9. Visual quality comparison on different expressions on the RaFD dataset.

<table>
<thead>
<tr>
<th>Methods / (×1e3)</th>
<th>S (%)</th>
<th>BN (%)</th>
<th>PN (%)</th>
<th>AE (%)</th>
<th>Acc.(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>StarGAN</td>
<td>3.676</td>
<td>7.875</td>
<td>6.933</td>
<td>3.751</td>
<td>95.67</td>
</tr>
<tr>
<td>CycleGAN</td>
<td>4.011</td>
<td>5.262</td>
<td>4.886</td>
<td>4.171</td>
<td>91.23</td>
</tr>
<tr>
<td>Ganimorph</td>
<td>4.689</td>
<td>5.645</td>
<td>5.129</td>
<td>5.570</td>
<td>86.25</td>
</tr>
<tr>
<td>MUNIT</td>
<td>5.189</td>
<td>5.551</td>
<td>4.761</td>
<td>5.271</td>
<td>81.43</td>
</tr>
<tr>
<td>Ours (No M)</td>
<td>3.224</td>
<td>6.051</td>
<td>5.682</td>
<td>4.667</td>
<td>88.29</td>
</tr>
<tr>
<td>Ours (No R)</td>
<td>3.022</td>
<td>6.505</td>
<td>5.894</td>
<td>3.911</td>
<td>90.96</td>
</tr>
<tr>
<td>Ours (Full)</td>
<td>2.907</td>
<td>5.137</td>
<td>4.704</td>
<td>3.678</td>
<td>97.85</td>
</tr>
<tr>
<td>Real</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>98.75</td>
</tr>
</tbody>
</table>

Table 1. Quantitative comparison in terms of distribution discrepancy and classification accuracy. For each facial attribute, we compare FID scores among methods. “S”, “BN”, “PN” and “AE” indicate Smiling, Big Nose, Pointy Nose and Arched Eyebrow respectively, while “Acc.” refers to classification accuracy.

Table 2. User study for different attribute translation among methods. The value refers to the ratio of selecting as best item.

<table>
<thead>
<tr>
<th>Methods</th>
<th>S (%)</th>
<th>BN (%)</th>
<th>PN (%)</th>
<th>AE (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>StarGAN</td>
<td>10.17</td>
<td>25.66</td>
<td>16.83</td>
<td>24.24</td>
</tr>
<tr>
<td>CycleGAN</td>
<td>14.41</td>
<td>8.85</td>
<td>14.85</td>
<td>28.79</td>
</tr>
<tr>
<td>Ganimorph</td>
<td>5.93</td>
<td>5.31</td>
<td>5.93</td>
<td>1.52</td>
</tr>
<tr>
<td>MUNIT</td>
<td>11.44</td>
<td>7.96</td>
<td>11.88</td>
<td>3.03</td>
</tr>
<tr>
<td>Ours</td>
<td>58.05</td>
<td>52.21</td>
<td>49.50</td>
<td>42.42</td>
</tr>
</tbody>
</table>

4.2.3 User Study

We also conduct user study for method comparison among 101 subjects, with 21 groups of generated samples. Given an input image, subjects are instructed to choose the best item based on quality of attribute transfer, perceptual realism, and preservation of identity. The results in Table 2 demonstrate that our method performs best among different facial attribute transformation methods, while StarGAN [4] performs well for subtle facial attribute (e.g. Big Nose) transformation and CycleGAN [42] yields decent output on obvious attributes (e.g. Arched Eyebrow, Smiling).

5. Conclusion

In this paper, we have introduced geometric deformation into image translation frameworks. We proposed spontaneous motion estimation module followed by refinement to fix remaining artifacts in deformation results. Extensive experiments manifest the effectiveness of our proposed framework. It achieves promising results for image translation and enables new visualization and applications. Our method may also shed lights on geometric-aware image translation.
References


