Adaptive Affine Transformation: A Simple and Effective Operation for Spatial Misaligned Image Generation

Zhimeng Zhang  
zhangzhimeng@corp.netease.com  
Virtual Human Group  
Netease Fuxi AI Lab

Yu Ding∗  
dingyu01@corp.netease.com  
Virtual Human Group  
Netease Fuxi AI Lab

Figure 1: Adaptive affine transformation can be used for misaligned image generation, including talking face generation, face reenactment, pose transfer, person image generation and so on.

ABSTRACT

One challenging problem, named spatial misaligned image generation, describing a translation between two face/pose images with large spatial deformation, is widely faced in tasks of face/pose reenactment. Advanced researchers use the dense flow to solve this problem. However, under a complex spatial deformation, even using carefully designed networks, intrinsical complexities make it difficult to compute an accurate dense flow, leading to distorted results. Different from those dense flow based methods, we propose one simple but effective operator named AdaAT (Adaptive Affine Transformation) to realize misaligned image generation. AdaAT simulates spatial deformation by computing hundreds of affine transformations, resulting in less distortions. Without computing any dense flow, AdaAT directly carries out affine transformations in feature channel spaces. Furthermore, we package several AdaAT operators to one universal AdaAT module that is used for different face/pose generation tasks. To validate the effectiveness of our AdaAT, we conduct qualitative and quantitative experiments on four common datasets in the tasks of talking face generation, face reenactment, pose transfer and person image generation. We achieve state-of-the-art results on three of them.1

CCS CONCEPTS

• Computing methodologies ➔ Reconstruction.

KEYWORDS

misaligned image generation, talking head generation, face reenactment, pose transfer, person image generation

ACM Reference Format:


∗Yu Ding is the corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM ’22, October 10–14, 2022, Lisboa, Portugal © 2022 Association for Computing Machinery. ACM ISBN 978-1-4503-9203-7/22/10 ... $15.00 https://doi.org/10.1145/3503161.3548330

1The project is in https://github.com/MRzzm/AdaAT
1 INTRODUCTION

Rapid development of deep learning promotes areas of media production, including talking face generation [2, 3, 7, 11, 24, 26, 41, 48, 50, 56–58], face reenactment [29, 33, 34, 44, 45, 51], pose transfer [15, 28, 36], person image generation [30, 31, 35, 38, 39] and so on. These tasks attract increasing researchers due to broad and interesting applications.

One common challenging problem on these tasks, named spatial misaligned image generation, is first proposed in [35]. As shown in Fig. 2 (b), this problem describes the translation between two images (e.g. facial images, human images, etc.) with a large spatial deformation. Compared with the spatial aligned image generation (shown in Fig. 2 (a)), misaligned condition is more difficult and can not be coped with well by traditional CNN based networks [6, 35], such as U-net [32].

Researchers make great efforts on solving this challenging problem. Recent advanced works [29, 39, 45, 47, 51, 52] propose several dense flow based frameworks to realize misaligned image generation. Specifically, they first utilize carefully designed networks to compute a dense flow. Then, they warp image feature maps with the dense flow in all the feature spaces (as shown in Fig 3 (a)). Finally, they synthesize images with aligned features. However, under a complex spatial deformation, intrinsical complexities make it difficult for networks to compute an accurate dense flow, leading to distorted results (see the synthetic results of X2face [47] and PRender [29] in Fig 7).

In this paper, we first propose one operator named AdaAT (Adaptive Affine Transformation) to replace the dense flow in realizing misaligned image generation. Due to two advanced designs, AdaAT is effective but simple. In the first design, AdaAT computes hundreds of affine transformations to simulate the sophisticated spatial deformation, like adding a regularization term on deformations, thus avoids synthesizing distorted results. Some other works [10, 34, 44] also utilize affine transformations, but the number of transformations is restricted between 10 and 20. By contrast, our AdaAT has a stronger capacity for simulating complex spatial deformation by computing affine transformations at least 100 times more than them. In the second design, without designing complex networks to compute spatial dense flow, AdaAT directly carries out affine transformations in feature channel spaces, leading to a very simple structure. As shown in Figure 3 (b), to align the image features, AdaAT first computes the parameters of the affine transformation for each feature channel, and then perform different affine transformations in different feature channels.

We further package several AdaAT operators to one AdaAT module that can be used for different face/pose generation tasks. The details of AdaAT Module are shown in Fig 4. We conduct experiments with the AdaAT module on the tasks of talking face generation, face reenactment, pose transfer and person image generation. We conduct qualitative and quantitative experiments on four common datasets and achieve the state-of-the-art results on three of them. As shown in Fig 1, our AdaAT module is effective in dealing with the problem of spatial misaligned image generation.

We summarize our contributions as follows:

- We propose one simple but effective operator named AdaAT to solve the problem of spatial misaligned image generation.

- We design one AdaAT module that can be used for different tasks of face/pose generation.

- To validate the effectiveness of our AdaAT, we conduct experiments on four common datasets in the tasks of talking face generation, face reenactment, pose transfer and person image generation, and achieve the state-of-the-art results on three of them.

2 RELATED WORK

In this section, we first briefly introduce the concept of spatial aligned and misaligned image generation. Then, we review works in the tasks of talking face generation, face reenactment, pose transfer and person image generation.

Alignment v.s. misalignment. Fig. 2 illustrates the differences between two conditions in image generation: spatial alignment and spatial misalignment. The first condition (shown in Fig. 2 (a)) requires the source image and the synthetic image have aligned spatial semantics. Many works [4, 9, 25, 27, 42, 43] focus on solving problems under this condition. [9] first proposes one pix2pix framework to realize image-to-image translation. To further improve visual quality of synthetic images, [4, 27, 43] utilize cascade structure and coarse-to-fine training strategy. [25] proposes one SPADE operation to improve visual fidelity and [42] utilizes a dense flow to improve smoothness in video synthesis. The second condition (shown in Fig 2 (b)), spatial misalignment, however, only requires the source image and the synthetic image from the same identity. This condition is more challenging [6, 35] and is widely faced in the tasks of talking face generation, face reenactment, pose transfer and so on.

Advanced researchers [15, 28, 29, 31, 33–36, 38, 39, 41, 44, 45, 51, 56] propose varied dense flow based frameworks to solve this condition. We will introduce these works, respectively, according to different tasks.

Talking face generation. In one-shot talking face generation, recent works leverage intermediate facial representations, including facial landmarks [2, 3, 5, 58], facial key points [40, 41] and 3DMM [48, 56], to split a pipeline into two cascade modules. The first module produces facial animation parameters (lip, eyebrow and head) from driving audio. The second module converts facial animation parameters into talking face videos. In the second module, driving head motion in the reference face encounters the problem of
misaligned image generation. Some works [3, 5, 58] ignore this problem, leading to synthesized talking head videos with a static head pose. Other works [40, 41, 56] propose dense flow based networks to synthesize talking videos with head movements. However, computing accurate dense flow is difficult, e.g., [56] uses two masks and three networks to compute dense flow with flaws.

**Face reenactment** In face reenactment, early works [49, 54] disentangle facial appearance and facial structure with landmarks to avoid the problem of spatial misalignment. However, due to imperfect disentanglement, their methods have poor generalization. Recent advanced works [29, 45, 47, 51] propose dense flow based frameworks to realize face reenactment. However, they synthesize distorted facial image under extreme head movements due to difficulties in computing an accurate dense flow. Other works [34, 44] utilize affine transformations to simulate spatial deformation to avoid synthesizing distorted results. However, the number of transformations is limited in their methods. Our AdaAT computes affine transformations at least 100 times more than them.

**Pose transfer & Person image generation** In pose transfer and person image generation, the direct exposed problem is misaligned image generation, so researchers [6, 14, 15, 28, 31, 35, 36, 39] design diverse dense flow based frameworks to synthesize person image. The dense flows is computed from unsupervised body parts [36], parametric statistical human body model [14, 15] or key joint points [6, 14, 28, 31, 35, 39]. Computing accurate dense flow is difficult, so local/global region fusion [31, 39] and multi resolution [28] need to be considered to improve the quality of dense flow.

![Feature map](Feature channel (a) Feature channel (b))

Figure 3: Comparison between the dense flow based methods and our AdaAT in the feature map spaces. (a) Dense flow based methods. Pseudo color represents dense flow. (b) AdaAT. Different colors represent different affine transformations.

### 3 METHOD

#### 3.1 Adaptive Affine Transformation Operator

AdaAT is proposed to deal with the problem of misaligned image generation. To facilitate understanding, we first briefly introduce the basic knowledge of dense flow [29, 39, 45, 47, 52]. Then, we introduce the details of AdaAT. Figure 3 (a) illustrates how the dense flow works. The pseudo color maps represent the dense flow and describe the spatial motion direction (the white arrow) of each pixel between two frames. With warping operations at the same position across all channels, the feature maps realize spatial alignment.

Different from the dense flow based methods, our AdaAT realizes feature spatial alignment through different spatial affine transformations in different feature channels. The details of AdaAT is illustrated in Figure 3 (b). After the AdaAT operation, the following convolutional layers merge all affine transformations into one sophisticated spatial deformation. We compute different affine transformations in different feature channels (the yellow, green and blue parts in the figure). Assume one image feature map $F \in \mathbb{R}^{C \times H \times W}$, where $C$, $H$, $W$ represent the channel size, height and width respectively. AdaAT computes a set of affine transformation matrix $A = \{A^c \in \mathbb{R}^{2 \times 2}\}_{c=1}^C$ according to the number of feature channels. For the $c_{th}$ channel in feature maps, the affine transformation is written as

$$
\begin{bmatrix}
x_{c} \\
y_{c}
\end{bmatrix} = A^c
\begin{bmatrix}
x_c \\
y_c
\end{bmatrix},
$$

where $x_c/\hat{x}_c$ and $y_c/\hat{y}_c$ are coordinates before/after affine transformation.

Traditional affine transformation has 6 parameters, controlling the transformation of scale, rotation, shear and translation. In our experiments, to facilitate the convergence of networks, we discard shear transformation and only compute 4 parameters of scale $s \in (0, 2)$, rotation $\theta \in (-\pi, \pi)$ and translation $t_x/t_y \in (-W/H, W/H)$ in each channel. The affine transformation matrix is denoted as

$$
A^c = \begin{bmatrix}
\cos(\theta) & -\sin(\theta)
\sin(\theta) & \cos(\theta)
\end{bmatrix}
\begin{bmatrix}
x_c \\
y_c
\end{bmatrix}.
$$

#### 3.2 Adaptive Affine Transformation Module

We package several AdaAT operators to one AdaAT module for misaligned face/pose image generation in the generalized applications. The structural details of AdaAT module is illustrated in Figure 4. Due to widely used key points (facial landmarks and pose joints) in different face/pose generation tasks, AdaAT module takes one source image $I_s$, one source heatmap image $I_s^{hm}$ and one driving heatmap image $I_d^{hm}$ as input. Then, $I_s$, $I_s^{hm}$ and $I_d^{hm}$ are concatenated and input into one appearance encoder to extract the appearance feature map $F^{opp}$. Then, $F^{opp}$ is input into one transformation encoder to compute the affine transformation parameters of scale $s$, rotation $\theta$ and translation $t_x/t_y$. Then, with equations 1 and 2, affine transformations are employed on $F^{opp}$ to generate the aligned feature map $F^{opp}_{align}$. To better simulate a sophisticated spatial deformation, two AdaAT operators and three convolutional layers are used alternately in feature alignment. We utilize one AdaIN [8] operation on $F^{opp}_{align}$ to add textural details. Finally, the aligned feature maps $F^{opp}_{align}$ are input into one appearance decoder to synthesize the output image. The details of AdaAT module are in supplementary materials.

#### 3.3 Loss Function

When training the AdaAT module, we use the LSGAN loss [21] $l_{GAN}$ and the perceptual loss [12] $l_{perc}$. The LSGAN loss is the patch GAN loss as same as in [34, 44, 56]. The perceptual loss is a two-scale loss as same as in [34, 44]. The final loss $L$ is written as
We conduct experiments on the tasks of talking face generation, face reenactment, pose transfer and person image generation to validate the effectiveness of AdaAT operation and AdaAT module. In this section, we first introduce the dataset and implementation details in our experiments. Next, we show the synthetic results and carry out quantitative and qualitative comparisons with other state-of-the-art works under different face/pose generation tasks. Finally, we conduct an online user study to validate our method and do an ablation study to evaluate the AdaAT module.

4 EXPERIMENTS

4.1 Dataset

In our experiments, we use four common datasets in face/pose generation.

**HDTF dataset**[56]. HDTF dataset is built for talking face generation with 512 × 512 resolution. It contains about 16 hours of videos with 300 subjects. In our experiment, we synthesize videos with 512 × 512 resolution and randomly select 5% of the HDTF dataset for testing.

**Voxceleb dataset** [23]. Voxceleb dataset contains about 352 hours videos with 1251 subjects. We use data processing strategy as similar as in [34, 45] to crop and resize all videos into 256 × 256 resolution. There are about 22496 training videos and 525 testing videos.

**iPER dataset** [16]. The iPER dataset consists of 206 videos with 30 subjects. To realize cross-identity pose transfer, we leverage SMPL [18] model to disentangle the body shape and pose. As same as in [15, 28], we select 185 videos for training and 21 videos for testing. The resolution of all videos is 256 × 256.

**DeepFashion dataset** [17]. DeepFashion dataset is one popular dataset in person image generation. We follow the data preprocessing strategy in [38, 59]. There are 101966 training pairs and 8570 testing pairs. All images are cropped into 256 × 176.

4.2 Implementation Details

In the task of talking face generation, our method does not focus on animation generation while relying on the animation generation module of [13]. In the task of face reenactment, the cross-identity face reenactment relies swapping 3DMM parameters of facial expression and head pose between source face and driving face, inspired from [29, 52]. In the stage of facial image generation, we project 68 3D facial key points to 2D image and then transform them to a heatmap image. In the task of pose transfer, similar to face reenactment, we use SMPL model to realize cross-identity pose transfer. In the stage of human image generation, we project 24 3D body joints to 2D image and transform them to a heatmap image. In the task of person image generation, we directly transform 2D key points to a heatmap image. More implementation details are in supplementary materials.

5 SYNTHETIC RESULTS

Figure 5 shows the synthetic results of talking face generation, face reenactment, pose transfer and person image generation. Rows 1 – 3 display frames of 3 different identities driven by the same audio. Our method has the ability to synthesize the 512×512 talking head videos. Rows 4 – 9 display the reenacted face/pose frames of four face/pose images. Due to effective face/pose statistical models (3DMM and SMPL), our method has the ability to realize cross-identity face/pose reenactment. Rows 10 – 11 display synthetic person images with large spatial deformation. It validates the effectiveness of our method in misaligned image generation.

We further visualize image feature maps before and after the 1st AdaAT operation in Figure 6. Figure 6 draws the feature maps and corresponding source/synthetic images. In AdaAT, different channels tend to encode different semantic or spatial aware features instead of similar spatial layouts, and the feature maps make rich affine transformations. We propose one view to explain this
phenomenon. To realize misaligned image generation, different spatial regions need to conduct different spatial deformation. In each channel of AdaAT, full spatial region does the same affine transformation. To achieve misaligned image generation, different channels need to encode the features of different spatial regions.

5.1 Comparison with State-of-the-art Works

We also compare our method with state-of-the-art works in the tasks of taking face generation, face reenactment, pose transfer and person image generation.
More feature maps are in supplementary materials.

5.1.1 Talking Face Generation. In the task of talking face generation, we compare our method with AVTG [3], RhyHead [2], wav2lip [26], MakeItTalk [58], PC-AVS [57] and FGNet [56]. Figure 7 shows the qualitative results. AVTG [3] and RhyHead [2] are limited in synthesizing 128 × 128 videos. Our method synthesizes 512 × 512 videos, wav2lip [26] only focuses on repairing the mouth region. Our method synthesizes talking videos with expression/head movements. MakeItTalk [58] neglects the problem of misaligned image generation, so their framework synthesizes blurry videos in large head motions. Our method utilizes AdaAT operators to deal with large head movements. PC-AVS [57] requires one extra reference head sequence as head movements which may be mismatched with the simultaneous speech. Our method synthesizes head movements from speech and are able to reflect the prosody of speech. FGNet [56] generates distorted face in large head motions due to inaccurate dense flow. Our method simulates a spatial deformation with hundreds of affine transformations to avoid synthesizing distorted results.

We also carry out quantitative comparisons with state-of-the-art works to validate our method. We reproduce MakeItTalk [58], MonkeyNet [33] and FGNet [56] on HDTF dataset. Table 1 illustrates the quantitative results, relying on metrics of SSIM [46] and LPIPS [55]. As observed, our method gets the best visual quality.

5.1.2 Face Reenactment. In the task of face reenactment, we compare our method with X2Face [47], Bi-layer [53], FOMM [34] and PIRender [29]. Figure 7 shows the qualitative comparisons. In Row 1, under the condition of small head movements, X2face and Bi-layer synthesize poor visual quality. In Rows 2 – 6, under the condition of extreme head pose, our method gets better results than FOMM and PIRender. Compared with PIRender, our method synthesizes facial images without facial distortion. The main reason is that it is difficult for networks to compute an accurate dense flow under a complex spatial deformation. Our method utilizes affine transformations to regularize the spatial deformation, thus avoids the distorted results. The results of methods with affine transformations (FOMM and adaAT) have fewer distortion than the dense flow based methods (X2face and PIRender), which also verify the effectiveness of affine transformation. Compared with FOMM, our method synthesizes higher visual quality. The main reason is that the number of transformations in FOMM is limited to 10, while our AdaAT computes affine transformations at least 100 times more than FOMM.

We conduct an online user study to validate our proposed method. In the talking face generation, we randomly select six pairs of reference images and driving audio from the internet. In the face reenactment, pose transfer and person image generation, we randomly select five pairs of source face/pose and driving face/pose from test data. 18 volunteers are invited to rate each frame or video.

Table 1: Quantitative comparisons with state-of-the-art works.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>SSIM ↑</th>
<th>LPIPS ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDTF dataset</td>
<td>MakeItTalk [58]</td>
<td>0.8273</td>
<td>0.2135</td>
</tr>
<tr>
<td></td>
<td>MonkeyNet [33]</td>
<td>0.8297</td>
<td>0.1184</td>
</tr>
<tr>
<td></td>
<td>FGNet [56]</td>
<td>0.8205</td>
<td>0.1402</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td><strong>0.8421</strong></td>
<td><strong>0.1120</strong></td>
</tr>
<tr>
<td>Voxceleb dataset</td>
<td>X2Face [47]</td>
<td>0.7190</td>
<td>0.2400</td>
</tr>
<tr>
<td></td>
<td>FOMM [34]</td>
<td>0.7230</td>
<td>0.1220</td>
</tr>
<tr>
<td></td>
<td>Bi-layer [53]</td>
<td>0.3190</td>
<td>0.2527</td>
</tr>
<tr>
<td></td>
<td>PIRender [29]</td>
<td>0.7325</td>
<td>0.1285</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td><strong>0.7508</strong></td>
<td><strong>0.1254</strong></td>
</tr>
<tr>
<td>iPER dataset</td>
<td>PG2 [20]</td>
<td>0.8540</td>
<td>0.1350</td>
</tr>
<tr>
<td></td>
<td>SHUP [1]</td>
<td>0.8320</td>
<td>0.0990</td>
</tr>
<tr>
<td></td>
<td>LiquidGAN [15]</td>
<td>0.8400</td>
<td>0.0870</td>
</tr>
<tr>
<td></td>
<td>TBN [28]</td>
<td><strong>0.8680</strong></td>
<td><strong>0.0860</strong></td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>0.8649</td>
<td>0.0893</td>
</tr>
<tr>
<td>DeepFashion dataset</td>
<td>PATN [59]</td>
<td>0.7730</td>
<td>0.2533</td>
</tr>
<tr>
<td></td>
<td>Intr-flow [14]</td>
<td>0.7780</td>
<td>0.2131</td>
</tr>
<tr>
<td></td>
<td>GFLA [31]</td>
<td>0.7900</td>
<td>0.2341</td>
</tr>
<tr>
<td></td>
<td>ADGAN [22]</td>
<td>0.7720</td>
<td>0.2256</td>
</tr>
<tr>
<td></td>
<td>XingGAN [38]</td>
<td>0.7780</td>
<td>0.2927</td>
</tr>
<tr>
<td></td>
<td>BiGraphGAN [37]</td>
<td>0.7780</td>
<td>0.2444</td>
</tr>
<tr>
<td></td>
<td>SPGNet [19]</td>
<td>0.7820</td>
<td>0.2105</td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td><strong>0.7952</strong></td>
<td><strong>0.1989</strong></td>
</tr>
</tbody>
</table>

Table 1 shows the quantitative comparisons on voxceleb dataset. As observed, our method gets the best SSIM.

5.1.3 Pose Transfer. In the task of pose transfer, we compare our method with PG2 [20], SHUP [1], LiquidGAN [15] and TBN [28]. Table 1 shows the quantitative results on iPER dataset. Our method gets competitive results when compared with previous works. We further analyze the factors that decrease the visual quality of our method. The main reason is that iPER dataset has too small data scale (only 30 identities and 185 training videos), leading to over fitting of cloth texture on some identities. We show the over fitted identity in supplementary materials. It indicates that large scale datasets benefit the training of AdaAT operator.

5.1.4 Person Image Generation. In the task of person image generation, we compare our method with PATN [59], Intr-flow [14], GFLA [31], ADGAN [22], XingGAN [38], BiGraphGAN [37] and SPGNet [19]. Table 1 shows the quantitative results on DeepFashion dataset. Our method gets the best results. Figure 8 shows the qualitative results. Compared with previous works, our method synthesizes person images with more reasonable details (please see the red rectangle). This may be due to the powerful capabilities of AdaAT in simulating sophisticated spatial deformation.

5.2 User Study

We conduct an online user study to validate our proposed method. In the talking face generation, we randomly select six pairs of reference images and driving audio from the internet. In the face reenactment, pose transfer and person image generation, we randomly select five pairs of source face/pose and driving face/pose from test data. 18 volunteers are invited to rate each frame or video.
Figure 7: Qualitative comparisons with state-of-the-art works in face generation.

from 1 (pretty fake) to 5 (pretty real). Higher scores represent more realistic videos. In the talking face generation, the rating results are AVTG (3.12), RhyHead (2.71), wav2lip (3.14), MakeItTalk (3.75), PC-AVS (4.02), FGNet (3.82) and ours (4.07). In the face reenactment, the rating results are X2Face (1.23), Bi-layer (2.02), FOMM (3.89), PIRender (3.78) and ours (3.92). In the pose transfer, the rating results are LiquidGAN (3.91) and ours (3.74). In the person image generation, the rating results are PATN (1.67), Intr-flow (2.10), GFLA (1.97), ADGAN (2.13), XingGAN (1.43), BiGraphGAN (2.22), SPGNet (3.89).
5.3 Ablation Study

We conduct ablation experiments to validate our method. Specifically, we set 5 conditions in the AdaAT module: (1) ours w/o adaAT & adaIN: removing the AdaAT and the AdaIN operation; (2) ours w/o adaAT: removing the AdaAT operation. (3) ours w/o adaIN: removing the AdaIN operation. (4) ours + dense flow: replacing the AdaAT with dense flow. (5) ours: complete AdaAT module. Figure 9 shows the qualitative results in face and pose generation. In condition 1 & 2, the synthetic images are more blurry than ours, the main reason is that the vanilla network and AdaIN can not handle the large spatial deformation well. In condition 3, the synthetic images have poorer textural details, e.g., the hair region, than ours. One possible reason is that AdaAT is capable of aligning feature maps by affine transformations, but is lack of adding extra detailed information on the feature maps. In condition 4, the synthetic images are more blurry and have more distortion than ours. The main reason is that it is difficult for networks to compute accurate dense flow under a complex spatial deformation. Table 2 shows the qualitative results on four datasets. Our complete method gets the best results.

<table>
<thead>
<tr>
<th>Method</th>
<th>HDTF</th>
<th>Voxceleb</th>
<th>iPER</th>
<th>DeepFashion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SSIM</td>
<td>LPIPS</td>
<td>SSIM</td>
<td>LPIPS</td>
</tr>
<tr>
<td>Ours w/o AdaAT &amp; AdaIN</td>
<td>0.812</td>
<td>0.202</td>
<td>0.738</td>
<td>0.147</td>
</tr>
<tr>
<td>Ours w/o AdaAT</td>
<td>0.828</td>
<td>0.127</td>
<td>0.739</td>
<td>0.141</td>
</tr>
<tr>
<td>Ours w/o AdaIN</td>
<td>0.837</td>
<td>0.113</td>
<td>0.742</td>
<td>0.121</td>
</tr>
<tr>
<td>Ours + dense flow</td>
<td>0.783</td>
<td>0.255</td>
<td>0.617</td>
<td>0.353</td>
</tr>
<tr>
<td>Ours</td>
<td>0.842</td>
<td>0.112</td>
<td>0.751</td>
<td>0.125</td>
</tr>
</tbody>
</table>

7 CONCLUSION

In this paper, we propose one novel AdaAT operator to solve the problem of misaligned image generation. We package several AdaAT operators to one AdaAT module that can be used in the tasks of head and pose generation. In the dataset of HDTF, voxceleb and DeepFashion, our method outperforms other works in objective and subjective comparisons. In the future, we will make great efforts to solve the above limitations.


