Motion Transformer for Unsupervised Image Animation

Jiale Tao[1*], Biao Wang[2], Tiezheng Ge[2], Yuning Jiang[2], Wen Li[1†], and Lixin Duan[1]

School of Computer Science and Engineering & Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China
{jialetao.std, liwenbnu, lxduan}@gmail.com
Alibaba Group
{eric.wb, tiezheng.gtz, mengzhu.jyn}@alibaba-inc.com

Abstract. Image animation aims to animate a source image by using motion learned from a driving video. Current state-of-the-art methods typically use convolutional neural networks (CNNs) to predict motion information, such as motion keypoints and corresponding local transformations. However, these CNN based methods do not explicitly model the interactions between motions; as a result, the important underlying motion relationship may be neglected, which can potentially lead to noticeable artifacts being produced in the generated animation video. To this end, we propose a new method, the motion transformer, which is the first attempt to build a motion estimator based on a vision transformer. More specifically, we introduce two types of tokens in our proposed method: i) image tokens formed from patch features and corresponding position encoding; and ii) motion tokens encoded with motion information. Both types of tokens are sent into vision transformers to promote underlying interactions between them through multi-head self attention blocks. By adopting this process, the motion information can be better learned to boost the model performance. The final embedded motion tokens are then used to predict the corresponding motion keypoints and local transformations. Extensive experiments on benchmark datasets show that our proposed method achieves promising results to the state-of-the-art baselines. Our source code will be public available.

1 Introduction

Image animation (also known as motion transfer) is a technique that aims to animate a source image based on the motion information extracted from a given driving video, such that the generated video can mimic the motion in the driving video while simultaneously retaining the appearance of the target object in the source image. This approach enables people to quickly create innovative content without the need to start from scratch, which can save large amounts

∗ Work done during an intership at Alibaba Group
† Corresponding author

time. Motion transfer has gained significant attention from the computer vision community in recent years [21,5,11,27,34,35,37], owing to its wide range of practical applications across entertainment and education such as virtual try-on [2,7], video conferencing [44], e-commerce advertising [49], and so on.

Existing image animation works can be roughly divided into two categories, namely supervised methods and unsupervised methods. In more detail, supervised methods typically focus on the animation of a specific object type (*e.g.*, human body, human face, *etc.*) and utilize a third-party model to extract structural representations, which might take the forms of 2D keypoints [23,24], 3D meshes [57], 3D optical flow [21] and so on. This type of method has advantages in modeling accurate object structures, but is limited by the object-specific approach to image animation. On the other hand, unsupervised methods [34,35,37] aim to avoid the requirement for object-specific predefined structure representations. These approaches usually learn intermediate motion representations (*e.g.*, keypoints and affine matrices) between two images by warping one image to reconstruct another. Currently proposed unsupervised methods generally comprise of two modules: a motion estimator and an image generator. In these methods, the image generators tend to be quite similar, while the motion estimators are always the research focus and are proven to be quite crucial for animation performance. For example, Siarohin et al. [34] utilize an unsupervised keypoint detector to estimate sparse motions. These authors later boost the performance of their method by adding a head in order to better predict the affine matrix [35]. Moreover, the method proposed in [37] further improves the motion learning process, by entangling a keypoint and the corresponding affine matrix into a single heat-map estimation.

In order to animate arbitrary objects, we follow the unsupervised setting and focus primarily on motion estimation in this work. It is worth noting that all CNN-based methods discussed above fail to consider the interactions between motions, which may prevent these methods from learning robust motion estimators. We believe the robustness of motion estimators can be boosted by the global information of motions. Accordingly, in this work, we make the first attempt to model the global motion information by employing vision transformers in unsupervised image animation. More specifically, we explicitly model the motion (*i.e.*, the keypoint and corresponding affine matrix) as a query token in the transformer, which we refer to as motion tokens and treat as learnable parameters. We further introduce image tokens, which are obtained by projecting the flattened image patch features to the same dimension as the motion tokens. These motion tokens, conditioned on image tokens, are then decoded to final keypoints and affine matrices through several transformer layers. Intuitively, the motion transformer compensates for the lack of prior structural representations by naturally introducing global motion information to assist with part motion learning; this procedure is efficiently implemented through the self-attention mechanism. We can summarize the advantages of the motion transformer in two aspects with reference to different objects: i) For objects with relatively non-rigid motions (such as human body), it learns the set of local motions in a more stable fashion; ii) for

objects with relatively rigid motions (such as faces), it exhibits a strong ability to learn global motion patterns simultaneously for all motion tokens.

We conduct extensive experiments on four benchmark datasets, which contain various kinds of objects such as talking heads, human bodies, animals, etc. The superior performance of our proposed method relative to existing baselines clearly demonstrates that global motion information can help to improve the robustness of motion estimators, as well as showing the success of our proposed motion transformer in capturing the global motion information.

2 Related Work

Image animation: Supervised methods [23,24,11,27,29,55,60,19,5,30,56,36,26] focus on the animation of a specific object type. Among these, the human body [22,29,13,54,31,6,1,33,52,14] and human face [11,12,28,51,46,48,3,41,47,16] are the most popular animation objects. Methods of this kind rely on object-specific landmark detectors, 3D models or other forms of supervision, which are usually pre-trained on a large amount of labeled data. On one hand, the advantage of these methods is that based on the pre-obtained structure representations, it is easier to further learn the warping flow between two images. On the other hand, these methods are also hampered by an obvious limitation, as they are only suitable for a specific object type.

Unsupervised methods [34,35,37,40] have been recently proposed to address the above limitation. These approaches typically leverage a large amount of easy-to-obtain unlabeled web videos and design image reconstruction losses to learn intermediate motion representations (*e.g.*, keypoints and affine matrices). Benefiting from the unsupervised scenario, methods of this kind can be applied to animate a wide range of objects, including human bodies, human faces, animals, etc. It is worth noting that no predefined structural representations of objects are available for training in those methods. Specifically, Monkey-Net [34] proposes to learn intermediate part keypoints as sparse motions by means of the downstream image reconstruction task. Subsequently, FOMM [35] improves on this approach by simultaneously regressing the local affine matrices along with the keypoints of object parts. Moreover, MRAA [37] further improves FOMM by combining the learning of part keypoints and local affine matrices into a single heat-map estimation process. Among these approaches, however, Monkey-Net is limited by the coarsely defined motion model, FOMM suffers from regressing stable affine matrices, while MRAA struggles in modeling relatively rigid motions (*e.g.*, human face) and fails to consider cooperative part motions.

It is worth noting that all above unsupervised methods focus on the motion estimation process in image animation. Our method also lies in this research scope, with a newly proposed motion transformer as the motion estimator. Similar to FOMM, our method also regresses the affine matrices, while it avoids the instability problem by adopting a global-assisted approach; moreover, benefiting from this approach, our method is also better able to handle rigid motions and cooperative local motions when compared with MRAA.

Vision transformer: Transformers [42] have achieved great success in natural language processing (NLP) community. Recently, they have also achieved promising results in computer vision tasks. Among them, DETR [4] and ViT [9] are the pioneering methods, and have been followed by a series of other vision transformer methods [59,58,20,18,53,50,43]. DETR [4] was the first to introduce transformers to the object detection. It follows the encoder-decoder architecture of traditional transformers in NLP, where object queries are introduced as learnable parameters. ViT [9] proposes a transformer encoder architecture for image classification, which directly splits the image into patches and introduces a learnable classification token to aid in performing the task. Recently, several methods have proposed variant forms of transformers for landmark detection [45,18,50]. Our method is motivated by these recent works, in that we regress the object keypoints as well as affine matrices for image animation.

3 Methodology

In the context of unsupervised image animation, we are given a source image S and a driving video $D = \{Z_i\}$, where Z_i is the i-th video frame. Unless otherwise noted, in the remainder of this work we will use Z to represent a video frame for the sake of simplicity.

3.1 The General Framework for Image Animation

Existing unsupervised image animation methods [34,35,37] generally perform image animation in a frame-by-frame manner. Given the source image and each frame of the driving video, the image animation model outputs a synthesized image that mimics the pose of the object in the driving frame while also preserving the appearance of the object in the source image.

Unsupervised models typically comprise two stages: motion estimation and image generation. The motion estimation stage produces the relative motion (often in the form of optical flow) between each driving video frame and the source image, while the image generation stage warps the source image based on the relative motion in order to generate the synthesized image.

To obtain the relative motion, the motion information of the source image or a driving video frame is first separately predicted and then ensembled to calculate the dense motion flow. More specifically, the motion information of a single image is disentangled as a set of transformations of object parts, each of which is represented by a keypoint and its affine transformation from an latent reference image. A motion estimator is designed to predict the keypoints and the corresponding affine transformations for the input image.

Motion estimation: The first stage of the general image animation framework involves estimating relative motions between the source image and driving frame. This stage plays a critical role in the process, as the estimation accuracy largely determines the overall quality of the generated video. Existing works [34,35,37] generally follow CNN-style models, where the transformation of each object part

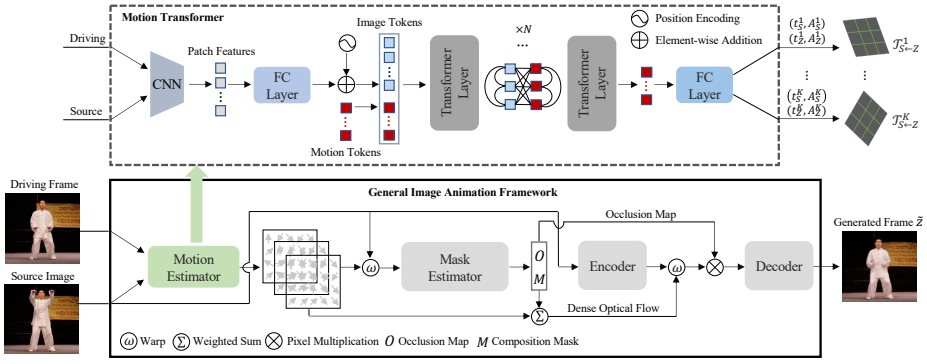

Fig. 1. Overview of the general image animation framework and our proposed motion transformer. Unlike the existing CNN based works [35,37], our motion transformer introduces image tokens and motion tokens, which encode visual and motion information respectively. And those tokens are further sent into multiple transformer layers to mine the underlying interactions between them, the self attention and cross attention are denoted by the straight and curved lines. By using a linear head, the output motion tokens are finally regressed to keypoints and their corresponding affine matrices.

is derived by learning a mapping from the learned feature maps. We contend that current CNN-based methods may not adequately capture the global motion information, as they do not consider interactions between part motions.

Assume there are K parts in an object for either the source image or each driving frame. In the motion estimation stage, the goal is to learn a motion transformation (t^k, A^k) for the k-th part, where $t^k \in \mathbb{R}^{2\times 1}$ denotes a keypoint (*i.e.*, the centroid of the transformation), $A^k \in \mathbb{R}^{2\times 2}$ represents the corresponding affine transformation matrix, and $k = 1, ..., K$.

Moreover, since the affine transformation A^k should be applied only to a certain neighboring area (also known as a mask) of the object part rather than the entire image, we constrain the effect of A^k by further learning the corresponding mask M^k. In the literature [35,37], the mask estimator is usually designed as a CNN-based encoder-decoder architecture. Specifically, it takes the warped source image as input and generates the masks $\{M^k|_{k=1}^K\}$'s of K object parts. Furthermore, an additional occlusion map can also be learned to guide the image generator in inpainting the occluded regions. We refer the readers to [35,37] for further details.

Motion representation: After learning (t_S^k, A_S^k) and (t_Z^k, A_Z^k), we can obtain the following motion flow from the driving frame Z to the source image S for the k-th object part based on the first-order motion model [35,37], as follows:

$$\mathcal{T}_{S\leftarrow Z}^k(c) = t_S^k + A_S^k(A_Z^k)^{-1}(c - t_Z^k), \tag{1}$$

where c denotes any image coordinate in the driving frame.

We next obtain the dense motion flow $\mathcal{T}_{S\leftarrow Z}(c)$ by combining $\mathcal{T}^k_{S\leftarrow Z}(c)$ with masks $M^k(c)$ as linear weights:

$$\mathcal{T}_{S\leftarrow Z}(c) = \sum_{k=1}^{K} M^k(c) \cdot \mathcal{T}^k_{S\leftarrow Z}(c), \tag{2}$$

where $M^k(c)$ is the mask of the k-th object part at the coordinate c and $\sum_{k=1}^{K} M^k(c) = 1$. Moreover, the dense motion flow $\mathcal{T}_{S\leftarrow Z}(c)$ represents that the pixel value at coordinate c of the generated image $\tilde{Z}$ is warped and obtained based on the pixel value at coordinate $\mathcal{T}_{S\leftarrow Z}(c)$ of the source image.

Image generation: Given the source image S and the dense motion flow $\mathcal{T}_{S\leftarrow Z}(c)$, a Unet-based encoder-decoder generator is introduced to generate the synthesized image $\tilde{Z}$. Specifically, the source image S is passed through the encoder to obtain feature maps, after which it is then warped according to the dense motion flow $\mathcal{T}_{S\leftarrow Z}(c)$. Finally, the decoder learns the synthesized image $\tilde{Z}$ based on the warped feature map.

3.2 Motion Transformer

Since existing CNN-based methods do not explicitly model the interactions between motions, the underlying motion relationship is not fully exploited and cannot be properly captured. We argue that this underlying relationship is critical to the process and helps reduce artifacts in generated animation videos. For instance, when people smile, the movement of their mouths and eyes occurs simultaneously, meaning that they are highly correlated. Considering this limitation of existing CNN-based models, we aim to seek out a better way of modeling the motion interactions.

To address the above issue, we propose to take advantage of the recently proposed vision transformer. We accordingly name our method the *motion transformer*. In a vision transformer layer, raw data are processed to form tokens, which act as the layer input. The underlying relationship among those tokens can be effectively mined through the attention mechanism. As a result of adopting this approach, meaningful embeddings can be learned for those tokens. Our motion transformer employs multiple vision transformer layers.

In our proposed motion transformer for image animation, we explicitly model the motions of object parts as input query tokens (*motion tokens*) to the transformer. We further obtain *image tokens* by projecting image patch features through a fully connected layer and subsequently embedding them with position encoding. By feeding those two types of tokens together into the transformer, the motion tokens are able to utilize the global context information of the entire image through attention with image tokens, which aids in better capturing the interaction between object part motions. Moreover, a linear head is designed in the last transformer layer to directly regress the keypoints and affine matrices of the motions. The entire process is illustrated in Fig 1.

Tokens: Two types of tokens are introduced in our motion transformer. We first introduce a set of motion tokens, inspired by the recent vision transformers [4].

Each motion token is expected to encode the motion information of an object part (*i.e.*, a keypoint and its affine transformation). These motion tokens are considered as learnable embeddings in our method; we denote them as $\{P_0^k|_{k=1}^K\}$, where $P_0^k \in \mathbb{R}^d$ represents the k-th object part and d is the embedding dimension.

The second type of tokens is the image token. Rather than directly using raw images, we first extract low-level image features by utilizing a CNN model. Subsequently we flatten the patch image features, with each patch projected to dimension d. To maintain the position information of the patch image features, we add the projected features by the absolute position encoding. We refer to the result features as image tokens, denoted as $I_0^n \in \mathbb{R}^d, n = 1, ..., N$.

Multiple vision transformers: When an object moves, different object parts are not completely independent. Rather, they often correlate with each other, which, however, was not discussed in existing motion transfer methods [35,37]. To model the relation among tokens, we utilize the natural advantages of vision transformer for building attention. In particular, a motion token is updated via all motion tokens and image tokens, and correspondingly we build two types of attention for the motion tokens. i) self attention for mining the underlying relationship between motion tokens; ii) cross attention for decoding motion tokens to the final keypoints and affine matrices.

Formally, let us denote by P_{l-1}^i a motion token that to be input to the l-th transformer layer, in which P_{l-1}^i is linearly projected to the query, key and value features $Q_{P_{l-1}^i}, K_{P_{l-1}^i}, V_{P_{l-1}^i}$; and similarly for image tokens we have $Q_{I_{l-1}^i}, K_{I_{l-1}^i}, V_{I_{l-1}^i}$. For ease of illustration, we temporally drop the subscript and define the multi-head self attention (MSA) as follow:

$$head^j = \text{softmax}\left(\frac{QW_Q^j\left(KW_K^j\right)^T}{\sqrt{d}}\right)VW_V^j, \tag{3}$$

$$\text{MSA}\left(Q, K, V\right) = [head^1, ..., head^h]W_O, \tag{4}$$

W_O, W_Q^j, W_K^j, W_V^j are learnable parameters, and h represents the total number of heads in each transformer layer. In practice, Q is the query from a token, while K, V are from another token. With this definition, the motion token is updated by self attention and cross attention as follows:

$$P_l^i = \sum_j \text{MSA}(Q_{P_{l-1}^i}, K_{P_{l-1}^j}, V_{P_{l-1}^j}) + \sum_j \text{MSA}(Q_{P_{l-1}^i}, K_{I_{l-1}^j}, V_{I_{l-1}^j}), \tag{5}$$

$$P_l^i = \text{FFN}\left(\text{LN}\left(P_l^i\right) + P_l^i\right), \tag{6}$$

where FFN and LN denote the feed forward network and layer normalization respectively. On one hand, the left term of Eqn. (5) (*i.e.*, the self attention) indicates that each motion token tends to query all other motion tokens, in this way the underlying relationship between motion tokens could be effectively captured; on the other hand, the motion token can be gradually embedded with

motion information through querying the image tokens, as formulated in the right term of Eqn. (5) (*i.e.*, the cross attention).

The two types of attention process above enable the efficient interactions between tokens. In implementation, different from recent works [4,20] in which the two types of attention process are separately conducted, we found it more efficient to unify the self attention and cross attention in a single transformer architecture. To do this, we directly concatenate the image tokens and motion tokens as the initial input tokens $F_0 = [P_0^1; ...; P_0^K; I_0^1; ...; I_0^N] \in \mathbb{R}^{(N+K)\times d}$, and use the single self attention process to update the concatenated tokens:

$$F_l^i = \sum_j \text{MSA}(Q_{F_{l-1}^i}, K_{F_{l-1}^j}, V_{F_{l-1}^j}), \tag{7}$$

$$F_l^i = \text{FFN}\left(\text{LN}\left(F_l^i\right) + F_l^i\right). \tag{8}$$

Note that in this procedure, image tokens are also updated with attention to all tokens, while to our observation, this dose not affect the performance, and the unified self attention is more efficient in end-to-end training.

At the final transformer layer, we take out the motion tokens P_L^k from the output token feature F_L, and employ a linear head to directly regress the affine matrix and translation vector. Let $W_h \in \mathbb{R}^{d\times 6}$ denote the parameters of the linear head; the decoded part affine matrix and translation vector of each motion token are then computed as $[A^k, t^k] = P_L^k W_h$.

3.3 Training

We consider the following losses to formulate the objective function of our method. The whole training process is conducted in an end-to-end fashion.

Perceptual loss: Following FOMM and MRAA, we adopt the multi-resolution perceptual loss [15] defined with a pre-trained VGG-19 [38] network. Given the driving frame Z with resolution index i, the generated image $\tilde{Z}$, and the feature extractor ϕ with layer index l, the perceptual loss can be written as follows:

$$\mathcal{L}_{per} = \sum_i \sum_l \left\| \phi_l\left(Z_i\right) - \phi_l(\tilde{Z}_i) \right\|_1. \tag{9}$$

Equivariance loss: Following FOMM and MRAA, the equivariance loss is adopted here. Given a random geometric transformation $\mathbf{T}$ and a driving image Z, this loss can be written as follows:

$$\mathcal{L}_{equi} = \sum_k \left\| \mathbf{T}(t_Z^k) - t_{\mathbf{T}(Z)}^k \right\|_1. \tag{10}$$

Background losses: To handle those situations in which the background is not static, we follow the MRAA in utilizing a background predictor network to predict the background motion flow. To facilitate the separation of the background and foreground motion learning, inspired by [10], we adopt the following losses:

$$\mathcal{L}_{mask} = \left\| M^0 - 1 \right\|_1 + \sum_{k\neq 0} \left\| M^k - 0 \right\|_1, \tag{11}$$

and

$$\mathcal{L}_{con} = \sum_{k \neq 0} \sum_{c} M^k(c) \cdot \left(c - u^k\right)^2 / \operatorname{sum}\left(M^k\right), \tag{12}$$

where $u^k = \sum_c M^k(c) \cdot c / \operatorname{sum}\left(M^k\right)$. Intuitively, the mask loss $\mathcal{L}_{mask}$ is used to constrain the portion of foreground and background area in an image, and the foreground motion mask is considered to be more focused under the constraint of the concentration loss $\mathcal{L}_{con}$.

Overall loss: We combine all of the above losses to formulate the overall object function of our proposed vision transformer, as follows:

$$\mathcal{L} = \mathcal{L}_{per} + \mathcal{L}_{equi} + \lambda(\mathcal{L}_{mask} + \mathcal{L}_{con}). \tag{13}$$

It should be noted here we adopt only the above background losses $\mathcal{L}_{mask}$ and $\mathcal{L}_{con}$ for the TaichiHD dataset with $\lambda = 0.1$ to handle the dynamic background change in TaichiHD videos in the experiments. As videos from other datasets typically have a static background, we omit $\mathcal{L}_{mask}$ and $\mathcal{L}_{con}$ from the overall loss for those datasets.

4 Experiments

Datasets: The following benchmark datasets are used in our experiments:

- VoxCeleb [25]: A talking head dataset consisting of 20047 videos. All videos are cropped and resized to 256×256.
- TaiChiHD [35]: This dataset contains 3120 videos. All videos are cropped and resized to 256×256.
- TED-talks [37]: This is a talking show dataset containing 1255 videos. All videos are cropped and resized to 384×384.
- MGIF [34]: This dataset contains 1000 cartoon animal videos, all of which are resized to 256×256, following [35].

Evaluation metrics: We follow [35,37] in evaluating the video reconstruction quality, where videos are reconstructed with appearance representations by using their first frame and motion representations w.r.t. all frames. Four commonly used evaluation metrics are listed below.

- L1 distance: The average L1 distance between the generated and ground-truth video frames.
- Average keypoint distance (AKD): The average distance of detected keypoints between the generated and ground-truth video frames. This metric is designed for evaluating the pose quality of generated videos.
- Missing keypoint rate (MKR): The percentage of keypoints that are not detected in the generated video frames but do exist in the ground-truth.
- Average Euclidean distance (AED): The average Euclidean distance between generated and ground-truth video frames, as in the feature space. This metric evaluates the identity information of the generated video frames.

Implementation details: For image generation, we adopt Unet [32] to construct the mask predictor and generative encoder-decoder. Skip connections are added in the encoder-decoder architecture similar to [34,37]. For motion estimation, we adopt the first three stages of the HRNet-W32 encoder [39] pretrained on ImageNet [8] as the CNN backbone in our model. After the CNN encoder has been applied, image features are down-sampled by a scale factor of 4. We utilize a 12-layer standard transformer encoder architecture. Moreover, the sine function [42] is used for position encoding. The image patch size is set to 4×4 in all experiments, with 256 image tokens used for the input resolution of 256×256 and 576 for 384×384. The token dimension d is set to 192. The number of motion tokens is set to 10, as in [35,37]. The Adam optimizer [17] is adopted, where the initial learning rate is set as 2×10^{-4} and dropped by a factor of 10 at the end of $60th$ and $90th$ epoch. We train the entire networks on eight NVIDIA V100 GPU cards for 100 epochs.

4.1 Comparison with State-of-the-Art

Model capacity: Under the general image animation framework, we analyze the difference between motion estimators from the model capacity view. As listed in Table 1, the proposed motion estimator has slightly less parameters than FOMM and MRAA, while the FLOPs of it are much heavier, which is caused by the high-resolution (4 times lower than the input image resolution) computation in the CNN encoder and in the global attention process in the vision transformer layers. It should be noted here, compared to the image generator that is always the same between our method and existing methods, the motion estimator tends to take a small computation cost in the whole image animation process. While being considerably lighter than the image generator, the motion estimator is proved to be efficient and effective for improving image animation performance, as in our method and recent works [35,37].

Quantitative comparison: The video reconstruction results are presented in Table 2. As can be seen from the table, our method generally performs the best across all evaluation metrics, as well as across all benchmark datasets with object types including human body, human face and animal etc., reflecting the superiority of the motion transformer to perform general image animation. More specifically, a lower L1 distance straightforwardly indicates better video reconstruction quality achieved by our method. It is also worth noting that our method achieves considerable improvements in terms of AKD on the three datasests, which strongly suggests that our method achieves better transferred motion. This can be further validated in the qualitative results of Fig. 2. Moreover, our method also achieves the best performance on the AED metric, indicating that the identity information can be better preserved using our method for conducting image animation. The superiority on the AED metric is even more obvious on the VoxCeleb dataset, we draw reason that the identity information is especially important for a human face, while our method generally learns the global motion pattern for the human face, which enables it to better capture the global face structure.

Table 1. Parameters comparison of the proposed motion estimator with that of FOMM and MRAA. For clearness, we also list the parameters of the image generator (the encoder-decoder generator together with the mask predictor). The model parameters and FLOPs are computed with input image resolution 256×256.

	Parameters	FLOPs
ImageGenerator	45.57M	53.64G
MotionEstimator-FOMM	14.21M	1.28G
MotionEstimator-MRAA	14.20M	1.26G
MotionEstimator-Ours	12.23M	7.54G

Table 2. Quantitative comparisons with FOMM [35] and MRAA [37] on the video reconstruction task. We present results on four benchmarks, our method generally achieves the best performance on all datasets across all metrics.

	TaiChiHD			TEDTalks			VoxCeleb			MGIF
	L1	(AKD, MKR)	AED	L1	(AKD, MKR)	AED	L1	AKD	AED	L1
FOMM	0.057	(6.649, 0.036)	0.172	0.029	(4.382, 0.008)	0.127	0.041	1.29	0.133	0.0224
MRAA	0.048	(5.246, 0.024)	0.150	0.027	(3.955, **0.007**)	0.118	0.040	1.28	0.133	0.0274
Ours	**0.045**	(**4.670, 0.021**)	**0.148**	**0.026**	(**3.456, 0.007**)	**0.113**	**0.038**	**1.18**	**0.116**	**0.0200**

User preference: To evaluate the cross-identity image animation, we conduct a user study with fifty participants. In more detail, we first prepare fifty comparison videos, each of which is a concatenation of a source image, a driving video featuring a different-identity, and videos generated by the three methods. Note that the spatial locations of the generated videos are randomly placed. Participants are required to evaluate these three videos according to the transferred motion and identity preservation. The results in Table 3 show that our method is clearly awarded more user preferences than other existing methods.

Qualitative comparison: In Fig. 2, we present representative animation examples on the TaichiHD, Voxceleb1 and TEDTalks dataset. As the figure shows, our method is generally better at handling both global and local motions. In more detail, for the human face, despite the fact that both FOMM and MRAA can capture the head rotation, our method can synthesize the most realistic and detailed expression information.

Our analysis suggests that it is often the case that a rigid human face turns from left to right, and occlusion occurs in this kind of rigid or global motion. Our method can effectively learn global motion patterns for a human face; accordingly, this makes it easier to detect the occlusion caused by the head rotation and then guide the image generator to inpaint this occluded face structure. In FOMM and MRAA, the motion is learned in a relatively local manner, which makes it more difficult to capture the global face structure. For the human body, it can also be observed that our method synthesizes the most motion-stable results, while FOMM and MRAA often fail to capture the driving motions. We believe this occurs because, lacking awareness of the global motion information, FOMM and MRAA are easier to be affected by large motions and intervention of

Fig. 2. Qualitative comparisons on the cross-identity image animation task. We show results on three datasets (from left to right: VoxCeleb, TaichiHD and TEDTalks), each with three paired examples.

Table 3. User preferences of our method against FOMM and MRAA on the TaichiHD, TEDTalks, and Voxceleb dataset.

	TaiChiHD	TEDTalks	VoxCeleb
FOMM	96.5%	66.4%	60.8%
MRAA	68.5%	57.1%	69.8%

Table 4. Performance comparison on the TaichiHD dataset with and without position encoding, denoted as w PE and w/o PE.

	L1	(AKD, MKR)	AED
w/o PE	0.047	(5.482, 0.028)	0.158
w PE	**0.045**	(**4.670, 0.021**)	**0.148**

background features. By contrast, our motion transformer learn the part motions in a global-assisted fashion, enabling it to learn the more stable part motions.

4.2 Ablation Study and Parameter Analysis

In this section we study the influence of different components of our motion transformer. More specifically, we conduct video reconstruction experiments on the TaichiHD dataset for the purpose of quantitative analysis.

Position encoding: As can be seen Table 4, it is crucial to add position encoding to the image tokens in our experiments. We can explain the importance of position encoding from two angles. On one hand, the motion (*a.k.a.*, keypoint and its corresponding affine matrix) estimation is a highly position-sensitive task, in which image tokens equipped with position encoding ease the learning process. On the other hand, in order to learn geometry-consistent keypoint and affine matrix representations in an unsupervised manner, the equivariance loss (*i.e.*, Eqn. (10)) is considered to be more effective with position encoding, since the order of the image patches are shuffled by a geometric transformation; however, the position encoding is invariant to the shuffling process, thus the consistency loss can enforce the network to better capture useful image patch features.

CNN encoder: To explore the influence of the image feature representation, we implement the motion transformer with different CNN backbones. As can be seen from Table 5, compared to our basic setting (*i.e.*, HR-w32), a light-weight CNN (*i.e.*, Stem net [39], a widely used CNN for quickly down-sampling the image by a scale factor of 4) yields worse performance, while a heavy CNN (*i.e.*,

Table 5. Performance comparison on the TaichiHD dataset with different CNN backbones of the motion transformer.

CNN	Param.	L1	(AKD, MKR)	AED
Stem	5.56M	0.048	(6.056, 0.030)	0.161
HR-W32	12.23M	**0.045**	(**4.670**, 0.021)	**0.148**
HR-W48	21.30M	**0.045**	(4.829, **0.020**)	0.149

Table 6. Performance comparison on the TaichiHD dataset with respect to different numbers of transformer layers.

Layers	L1	(AKD, MKR)	AED
4	0.046	(5.320, 0.027)	0.155
8	0.046	(5.226, 0.025)	0.154
12	**0.045**	(**4.670, 0.021**)	**0.148**

HR-w48 [39]) brings no significant improvement. We accordingly conclude that in the absence of a good image feature representation, the motion transformer can't work well; at the same time, the promotion of the CNN backbone to the final performance is limited, which we attribute to the lack of supervision in the unsupervised image animation.

Vision transformer layers: We further conduct experiments to explore the influence of different numbers of vision transformer layers used in our motion transformer. As can be seen from Table 6, with smaller numbers of layers, the performance declines considerably (especially on AKD and MKR, which evaluate the motion quality). This reflects the fact that the motion transformer with relatively deeper transformer layers facilitates to learn better motion embeddings for the regression of the motion information.

4.3 Visualization

In this section, we visualize intermediate results to analyze how the motion transformer learns global motions with different object types and what motion patterns have been learned. Samples are randomly chosen; while our observations suggest that the model behaves similarly on the entire dataset.

Visual attention: We visualize the attention maps between motion tokens and image tokens, to reveal how the motion transformer learns the global and local motion. The results on the VoxCeleb and TEDTalks dataset are presented in Fig. 3. For human faces, it can be seen that the whole face region tends to be attended by all different motion tokens; this implies that each motion part is learned with awareness of the global motion, which is in line with our motivation. For human bodies, we first observe that the global motion is effectively captured; as can be seen in the third row, the motion token learned with a global pattern attends almost the whole regions of the object in the image. Moreover, we find that in the initial transformer layers, the local motions of the woman's hands are well captured. As the depth increases, motion tokens can also find some other meaningful relationship. For example, as illustrated in the attention results of the TEDTalks sample in Fig. 3, the motion token representing the woman's right hand (*i.e.*, the first row) actually shows that it is related to her upper body and head, as in the attention map of the sixth transformer layer. This is reasonable because, when a woman presents something in the talk, her body may move together with the hand gestures to communicate with the whole body language.

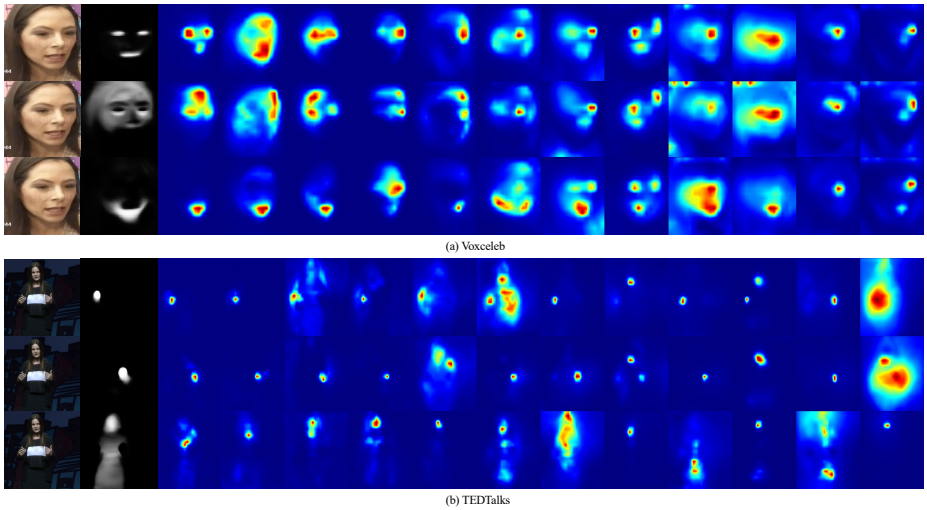

Fig. 3. Visualizations of visual attention maps between motion tokens and image tokens on the VoxCeleb and TEDTalks datasets. From left to right of in each row, the presented content respectively represents the driving image, the corresponding motion mask and the visual attention maps of each transformer layer. We present visual attention maps of three representative motion tokens for each dataset. Note that we reshape and resize the sequence attention values to the original image sizes.

5 Conclusion

We propose a new method, called the motion transformer, under the general image animation framework for unsupervised image animation. The motion transformer introduces both image tokens and learnable motion tokens. To encourage the interactions between image and motion tokens, our motion transformer network employs multiple transformer layers, which take those tokens as input in order to learn the underlying motion relationship and obtain better motion embeddings. We further conduct extensive experiments on four benchmark datasets. Our experimental results validate the effectiveness of capturing the global motion information in our motion transformer.

Acknowledgement: This work is supported by the Major Project for New Generation of AI under Grant No. 2018AAA0100400, the National Natural Science Foundation of China (Grant No. 62176047), Sichuan Science and Technology Program (No. 2021YFS0374, 2022YFS0600), Beijing Natural Science Foundation (Z190023), and Alibaba Group through Alibaba Innovation Research Program. This work is also partially supported by the Science and Technology on Electronic Information Control Laboratory.

References

1. Balakrishnan, G., Zhao, A., Dalca, A.V., Durand, F., Guttag, J.: Synthesizing images of humans in unseen poses. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8340–8348 (2018) 3
2. Bhatnagar, B.L., Tiwari, G., Theobalt, C., Pons-Moll, G.: Multi-garment net: Learning to dress 3d people from images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5420–5430 (2019) 2
3. Burkov, E., Pasechnik, I., Grigorev, A., Lempitsky, V.: Neural head reenactment with latent pose descriptors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13786–13795 (2020) 3
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision. pp. 213–229. Springer (2020) 4, 6, 8
5. Chan, C., Ginosar, S., Zhou, T., Efros, A.A.: Everybody dance now. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5933–5942 (2019) 2, 3
6. Chen, X., Song, J., Hilliges, O.: Unpaired pose guided human image generation. In: Conference on Computer Vision and Pattern Recognition (CVPR 2019). Computer Vision Foundation (CVF) (2019) 3
7. Chopra, A., Jain, R., Hemani, M., Krishnamurthy, B.: Zflow: Gated appearance flow-based virtual try-on with 3d priors. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5433–5442 (2021) 2
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009) 10
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021) 4
10. Gao, Q., Wang, B., Liu, L., Chen, B.: Unsupervised co-part segmentation through assembly. In: International Conference on Machine Learning (2021) 8
11. Geng, Z., Cao, C., Tulyakov, S.: 3d guided fine-grained face manipulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9821–9830 (2019) 2, 3
12. Ha, S., Kersner, M., Kim, B., Seo, S., Kim, D.: Marionette: Few-shot face reenactment preserving identity of unseen targets. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 10893–10900 (2020) 3
13. Huang, Z., Han, X., Xu, J., Zhang, T.: Few-shot human motion transfer by personalized geometry and texture modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2297–2306 (2021) 3
14. Jiang, Y., Yang, S., Qiu, H., Wu, W., Loy, C.C., Liu, Z.: Text2human: Text-driven controllable human image generation. ACM Transactions on Graphics (TOG) **41**(4), 1–11 (2022). https://doi.org/10.1145/3528223.3530104 3
15. 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) 8
16. Kim, H., Garrido, P., Tewari, A., Xu, W., Thies, J., Nießner, M., Pérez, P., Richardt, C., Zollöfer, M., Theobalt, C.: Deep video portraits. ACM Transactions on Graphics (TOG) **37**(4), 163 (2018) 3

17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015) 10
18. Li, Y., Zhang, S., Wang, Z., Yang, S., Yang, W., Xia, S.T., Zhou, E.: Tokenpose: Learning keypoint tokens for human pose estimation. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2021) 4
19. Li, Y., Huang, C., Loy, C.C.: Dense intrinsic appearance flow for human pose transfer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3693–3702 (2019) 3
20. Li, Y., He, J., Zhang, T., Liu, X., Zhang, Y., Wu, F.: Diverse part discovery: Occluded person re-identification with part-aware transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2898–2907 (2021) 4, 8
21. Liu, W., Piao, Z., Min, J., Luo, W., Ma, L., Gao, S.: Liquid warping gan: A unified framework for human motion imitation, appearance transfer and novel view synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5904–5913 (2019) 2
22. Lorenz, D., Bereska, L., Milbich, T., Ommer, B.: Unsupervised part-based disentangling of object shape and appearance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10955–10964 (2019) 3
23. Ma, L., Jia, X., Sun, Q., Schiele, B., Tuytelaars, T., Van Gool, L.: Pose guided person image generation. Advances in Neural Information Processing Systems **30**, 406–416 (2017) 2, 3
24. Ma, L., Sun, Q., Georgoulis, S., Van Gool, L., Schiele, B., Fritz, M.: Disentangled person image generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 99–108 (2018) 2, 3
25. Nagrani, A., Chung, J.S., Zisserman, A.: Voxceleb: a large-scale speaker identification dataset. arXiv preprint arXiv:1706.08612 (2017) 9
26. Neverova, N., Guler, R.A., Kokkinos, I.: Dense pose transfer. In: Proceedings of the European conference on computer vision (ECCV). pp. 123–138 (2018) 3
27. Nirkin, Y., Keller, Y., Hassner, T.: Fsgan: Subject agnostic face swapping and reenactment. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7184–7193 (2019) 2, 3
28. 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) 3
29. Ren, J., Chai, M., Tulyakov, S., Fang, C., Shen, X., Yang, J.: Human motion transfer from poses in the wild. In: European Conference on Computer Vision. pp. 262–279. Springer (2020) 3
30. Ren, J., Chai, M., Woodford, O.J., Olszewski, K., Tulyakov, S.: Flow guided transformable bottleneck networks for motion retargeting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10795–10805 (2021) 3
31. Ren, Y., Yu, X., Chen, J., Li, T.H., Li, G.: Deep image spatial transformation for person image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7690–7699 (2020) 3
32. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015) 10

33. Sarkar, K., Mehta, D., Xu, W., Golyanik, V., Theobalt, C.: Neural re-rendering of humans from a single image. In: Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI. p. 596–613. Springer-Verlag, Berlin, Heidelberg (2020). https://doi.org/10.1007/978-3-030-58621-8_35, https://doi.org/10.1007/978-3-030-58621-8_35 3
34. Siarohin, A., Lathuilière, S., Tulyakov, S., Ricci, E., Sebe, N.: Animating arbitrary objects via deep motion transfer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2377–2386 (2019) 2, 3, 4, 9, 10
35. Siarohin, A., Lathuilière, S., Tulyakov, S., Ricci, E., Sebe, N.: First order motion model for image animation. In: Advances in Neural Information Processing Systems (2019) 2, 3, 4, 5, 7, 9, 10, 11
36. Siarohin, A., Sangineto, E., Lathuiliere, S., Sebe, N.: Deformable gans for pose-based human image generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3408–3416 (2018) 3
37. Siarohin, A., Woodford, O., Ren, J., Chai, M., Tulyakov, S.: Motion representations for articulated animation. In: CVPR (2021) 2, 3, 4, 5, 7, 9, 10, 11
38. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015) 8
39. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5693–5703 (2019) 10, 12, 13
40. Tao, J., Wang, B., Xu, B., Ge, T., Jiang, Y., Li, W., Duan, L.: Structure-aware motion transfer with deformable anchor model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3637–3646 (2022) 3
41. Tripathy, S., Kannala, J., Rahtu, E.: Facegan: Facial attribute controllable reenactment gan. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1329–1338 (2021) 3
42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017) 4, 10
43. Wan, Z., Zhang, J., Chen, D., Liao, J.: High-fidelity pluralistic image completion with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4692–4701 (October 2021) 4
44. Wang, T.C., Mallya, A., Liu, M.Y.: One-shot free-view neural talking-head synthesis for video conferencing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10039–10049 (2021) 2
45. Watchareeruetai, U., Sommanna, B., Jain, S., Noinongyao, P., Ganguly, A., Samacoits, A., Earp, S.W., Sritrakool, N.: Lotr: Face landmark localization using localization transformer. arXiv preprint arXiv:2109.10057 (2021) 4
46. Wei, D., Xu, X., Shen, H., Huang, K.: C2f-fwn: Coarse-to-fine flow warping network for spatial-temporal consistent motion transfer. Proceedings of the AAAI Conference on Artificial Intelligence **35**(4), 2852–2860 (May 2021), https://ojs.aaai.org/index.php/AAAI/article/view/16391 3
47. Wei, Y., Liu, M., Wang, H., Zhu, R., Hu, G., Zuo, W.: Learning flow-based feature warping for face frontalization with illumination inconsistent supervision. In: European Conference on Computer Vision. pp. 558–574. Springer (2020) 3
48. Wiles, O., Koepke, A., Zisserman, A.: X2face: A network for controlling face generation using images, audio, and pose codes. In: Proceedings of the European conference on computer vision (ECCV). pp. 670–686 (2018) 3

49. Xu, B., Wang, B., Tao, J., Ge, T., Jiang, Y., Li, W., Duan, L.: Move as you like: Image animation in e-commerce scenario. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 2759–2761 (2021) 2
50. Yang, S., Quan, Z., Nie, M., Yang, W.: Transpose: Keypoint localization via transformer. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2021) 4
51. Yao, G., Yuan, Y., Shao, T., Li, S., Liu, S., Liu, Y., Wang, M., Zhou, K.: One-shot face reenactment using appearance adaptive normalization. Proceedings of the AAAI Conference on Artificial Intelligence **35**, 3172–3180 (May 2021) 3
52. Yoon, J.S., Liu, L., Golyanik, V., Sarkar, K., Park, H.S., Theobalt, C.: Pose-guided human animation from a single image in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 15039–15048 (June 2021) 3
53. Yu, X., Rao, Y., Wang, Z., Liu, Z., Lu, J., Zhou, J.: Pointr: Diverse point cloud completion with geometry-aware transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12498–12507 (2021) 4
54. Zablotskaia, P., Siarohin, A., Zhao, B., Sigal, L.: Dwnet: Dense warp-based network for pose-guided human video generation. In: BMVC. p. 51 (2019) 3
55. Zakharov, E., Shysheya, A., Burkov, E., Lempitsky, V.: Few-shot adversarial learning of realistic neural talking head models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9459–9468 (2019) 3
56. Zhang, J., Li, K., Lai, Y.K., Yang, J.: Pise: Person image synthesis and editing with decoupled gan. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7982–7990 (2021) 3
57. Zhao, L., Peng, X., Tian, Y., Kapadia, M., Metaxas, D.: Learning to forecast and refine residual motion for image-to-video generation. In: Proceedings of the European conference on computer vision (ECCV). pp. 387–403 (2018) 2
58. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6881–6890 (2021) 4
59. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable {detr}: Deformable transformers for end-to-end object detection. In: International Conference on Learning Representations (2021) 4
60. Zhu, Z., Huang, T., Shi, B., Yu, M., Wang, B., Bai, X.: Progressive pose attention transfer for person image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2347–2356 (2019) 3