Structure-Aware Motion Transfer with Deformable Anchor Model

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

[1]School of Computer Science and Engineering & Shenzhen Institute for Advanced Study,
University of Electronic Science and Technology of China

[2]Alibaba Group

{jialetao.std, liwenbnu, lxduan}@gmail.com, xbr_2017@std.uestc.edu.cn

{eric.wb, tiezheng.gtz, mengzhu.jyn}@alibaba-inc.com

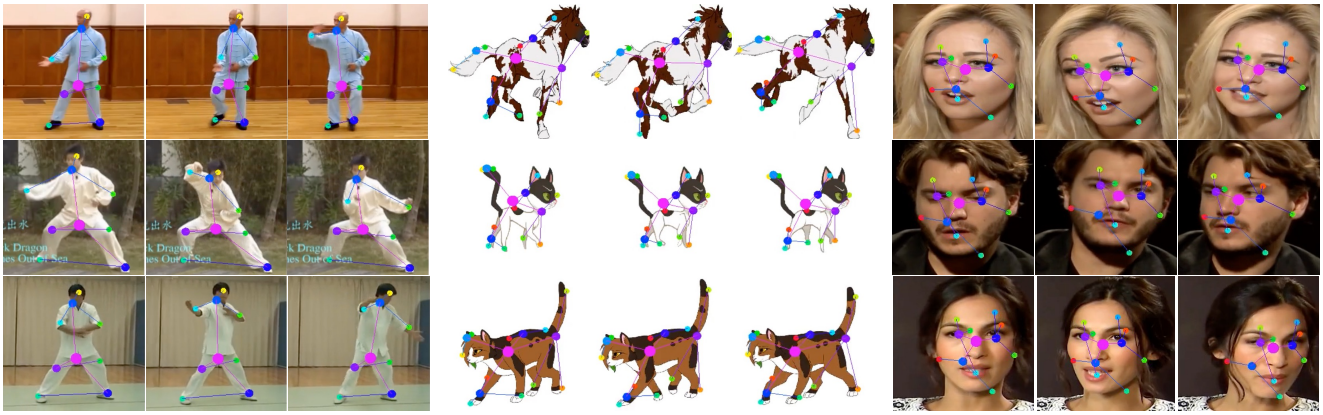

Figure 1. Examples of structures learned by our proposed approach for the motion transfer task. We present results on three different datasets: TaiChiHD, MGIF and VoxCeleb1. It is worth noting that no prior structural information (*e.g.*, skeleton) is used in our approach.

Abstract

Given a source image and a driving video depicting the same object type, the motion transfer task aims to generate a video by learning the motion from the driving video while preserving the appearance from the source image. In this paper, we propose a novel structure-aware motion modeling approach, the deformable anchor model (DAM), which can automatically discover the motion structure of arbitrary objects without leveraging their prior structure information. Specifically, inspired by the known deformable part model (DPM), our DAM introduces two types of anchors or keypoints: i) a number of motion anchors that capture both appearance and motion information from the source image and driving video; ii) a latent root anchor, which is linked to the motion anchors to facilitate better learning of the representations of the object structure information. Moreover, DAM can be further extended to a hierarchical version through the introduction of additional latent anchors to model more complicated structures. By regularizing motion anchors with latent anchor(s), DAM enforces the correspondences between them to ensure the structural information is well captured and preserved. Moreover, DAM can be learned effectively in an unsupervised manner. We validate our proposed DAM for motion transfer on different benchmark datasets. Extensive experiments clearly demonstrate that DAM achieves superior performance relative to existing state-of-the-art methods.

1. Introduction

Recently, motion transfer has gained increasing attention from computer vision researchers, due to its numerous potential applications in the fields of video re-enactment [4], fashion design [7], face swapping [24], and so on. Given a source image and a driving video of the same object type, the goal of motion transfer is to generate a video that depicts the motion pattern contained in the driving video while preserving the appearance from the source image.

*Work done during an internship at Alibaba Group

†The corresponding author

‡Codes will be available at https://github.com/JialeTao/DAM.git

Finding the correspondence between a source image and a driving video is the key to successful motion transfer. Existing motion transfer methods address this issue in two ways. On one hand, model-based methods [9, 14] utilize a pre-trained third-party model to extract the structural information of an object (*e.g.*, human bodies, human faces, etc.). However, specific predefined structure priors are required for different objects. On the other hand, model-free methods [23, 24, 33] treat motion keypoints as unknown variables, then design models to predict them by optimizing the image reconstruction loss. While these approaches do not require a predefined object structure, they often suffer from false correspondences, leading to considerable artifacts emerging in the generated videos (see Fig. 2 for examples).

To address these issues, in this paper, we propose a novel structure-aware motion transfer approach referred to as the deformable anchor model (DAM). In DAM, we take advantage of both the model-free and model-based methods. On one hand, similar to the model-free methods, we represent motion keypoints (a.k.a., "anchors") as unknown variables, which enables our model to perform motion transfer on an arbitrary object without knowing its prior structural information. On the other hand, to prevent the false correspondences, we also encode the structural information to constrain those motion anchors. Unlike model-based methods, our approach does not employ any pre-trained third-party model. Instead, as inspired by the well-known deformable part model (DPM) [8], DAM introduces a latent root anchor to regularize the motion anchors and model the object structure, enabling the correspondence between the source image and driving video to be enforced and thus further improving the performance. Furthermore, by introducing additional latent anchors, DAM can be easily extended to a hierarchical version that can more effectively model complicated object structures. Note that all latent anchors in our DAM are unknown variables, and that DAM can be learned in an end-to-end manner, similarly to previous model-free methods.

We conduct experiments on four benchmark datasets (*i.e.*, TaiChiHD, FashionVideo, VoxCeleb1 and MGIF) for performance evaluation. The experimental results show that our method not only achieves the best quantitative performance, but also exhibits a strong capacity to capture the motion structure of different objects, such as human bodies, faces, animals, and so on.

2. Related Work

Video-to-video synthesis: Motion transfer has been studied in the area of video-to-video synthesis to some extent. Video2video [29] proposed to synthesize photo-realistic videos via input video semantic maps. Chan *et al*. [4, 34] further extended the generation scheme to synthesize human dance videos conditioned on input video pose sequences and a source identity. These methods are good at utilizing the input source appearance information and can generate realistic videos. However, they are also identity-specific methods, meaning that they require a large amount of source images with diverse views and ranges of motion and moreover take a long time to train.

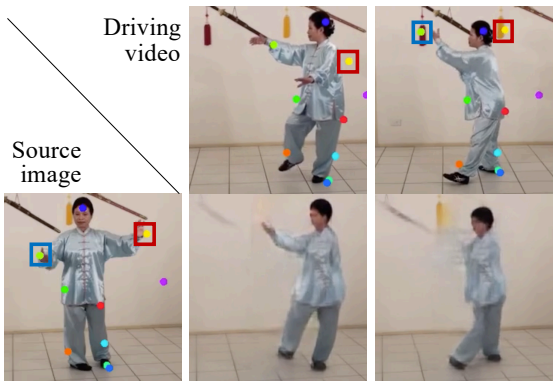

Figure 2. Failure cases from the FOMM method [24]. Inaccurate correspondences between motion points cause parts of the human body to be missed in the generated videos.

Motion transfer: Early methods [1, 14, 15, 25, 31] mainly focus on pose-guided human image generation. These works use off-the-shelf pose estimators or keypoint detectors to pre-extract pose information, which is then adopted for conditioning the image generation process. A series of works [5, 12, 13, 17, 19–22, 36, 39, 42] have adopted this approach. In addition, many works have proposed facial animation methods [2, 6, 9, 11, 28, 30, 32, 35] which can be seen as a kind of facial motion transfer. Similarly, these methods also employ an off-the-shelf facial landmark detector for expression modeling.

Despite their ability to transfer the pose of a human body or the expression on a human face, these methods heavily rely on third-party models and are object-specific. Inspired by Jakab et al. [10], which proved that object landmarks can be learned in an unsupervised way via image reconstruction, Monkey-Net [23] was the first to propose a model-free motion transfer method for arbitrary objects, which was achieved by building backward motion flow from aligned keypoints to warp the source image feature to driving pose. This warping-based method can achieve superior motion modeling and transferring performance, but this performance begins to suffer when the motions in question are large and complex. FOMM [24] enhances the motion model by introducing local affine transformations to motion keypoints. Since no structural information is provided, however, this approach often suffers from unstable correspondence between the source and driving image. RegionMM [26] further extends the FOMM by defining regions that can be used to model parts of an object, although

it does not consider the dependent structure between the different regions.

Other related work: Most of the above motion transfer methods rely on keypoint detection for encoding pose information. Generally speaking, model-based methods tend to adopt supervised keypoint detection or pose estimation methods [3, 18, 37, 41], while model-free methods tend to be unsupervised keypoint detection methods [10, 40]. For supervised cases, keypoints are learned on additional and richly annotated datasets. For unsupervised cases, keypoints are usually learned via an auxiliary image reconstruction task. Specifically, detected keypoints are considered to represent the structural information of an image object; the image should be reconstructed via combining the structural and appearance information.

Our work is partially inspired by DPM [8], which is a traditional human object detection approach. It breaks down the task into individual part detection task across human body and defines the score of positive detection of a root location by considering the spatial distance prior between root location and part locations. Intuitively, if the current relative distance from a part (*e.g.* the left leg of a human) to the root (*e.g.* the head of a human) is much larger than the prior relative distance, then these root-part pair locations tend to be assigned a lower positive detection score in DPM. In a similar spirit, we consider the motion prior of a root anchor which is formulated in a similar way to the spatial distance constraint in DPM.

3. Structure-Aware Motion Transfer

In this section, we present our structure-aware motion transfer approach which we name the deformable anchor model (DAM). We develop our approach based on the recent first-order motion model (FOMM [24]), with the addition of two novel deformable anchor models to encode the motion structure information. Below, we first present a brief review of the FOMM method in Section 3.1, and then introduce the basic deformable anchor model in Section 3.2. A more effective hierarchical deformable anchor model is presented in Section 3.3, followed by a summary of the entire model in Section 3.4.

3.1. Motion Flow Modeling

Given a source image and a driving video, FOMM [24] generates the motion transfer video by warping the source image to mimic the driving video in a frame-by-frame manner. For this purpose, they firstly estimate the dense motion flow between these two images. They then warp the source image in the feature space, and synthesize the driving frame with an image generator based on the warped source image feature. The entire process is illustrated in the bottom part of Figure 4.

Formally, given a source image S and a driving frame D, the motion between two images is modeled by the motion flow $\mathcal{T}_{S\leftarrow D}(z)$, where z denotes the coordinates of any pixel in the image. Estimating the dense motion flow is non-trivial. To ease this process, FOMM employs a set of motion anchors; these anchors are intended to represent identical keypoints of the object in the source image and driving frame (for example, the corresponding physical parts of the human body). With the aid of aligned motion anchors the dense motion flow can be derived through affine transformations.

In more detail, let z_k^s and z_k^d denote the k-th pair of corresponding anchors in S and D respectively; here, $k = 1, \ldots, K$, where K is the number of motion anchors. This yields the following:

$$z_k^s = \mathcal{T}_{S\leftarrow D}(z_k^d) \tag{1}$$

Given a motion anchor, the motion flow for pixels at the local region around the anchor can be approximately modeled with an affine transformation. For convenience, let $\mathcal{T}_k$ denote the dense motion flow derived by the k-th motion anchor. The affine transformation can thus be described as follows:

$$\mathcal{T}_k(z) = \mathcal{T}_k(z_k^d) + \theta_k(z - z_k^d) \tag{2}$$

where θ_k is the parameter of local affine transformation for the k-th anchor.

Intuitively, the dense motion flow of a pixel z can be derived from any nearby motion anchor. Thus a weight parameter $M_k(z)$ is introduced to automatically combine the $\mathcal{T}_k(z)$ from different anchors. The dense motion flow of any pixel z can thus be represented as follows:

$$\mathcal{T}_{S\leftarrow D}(z) = \sum_{k=1}^{K} M_k(z) \cdot \mathcal{T}_k(z), \tag{3}$$

where $\sum_{k=0}^{K} M_k(z) = 1, \forall z$, in which $M_0(z)$ is an additional mask for modeling background similar to the approach [26].

With z_k^d,z_k^s,θ_k, and M_k, it is possible to obtain a dense motion flow between the source image S and driving frame D, after which the source image can be warped to mimic the driving frame with an image generator. By enforcing an image reconstruction loss on the image generator, a motion estimator can be trained to automatically predict these unknown variables (*i.e.*, z_k^d,z_k^s,θ_k, and M_k) (see Fig. 4).

As FOMM has shown, the motion anchors tend to have coarse physical meanings (*e.g.*, a motion anchor may always locate at the head region of a human). However, false correspondences may occur if large motion or background variance are present, which will lead to considerable artifacts in the generated videos, as shown in Fig.2. We will discuss how to address these issues by encoding latent object structure information in the following subsections.

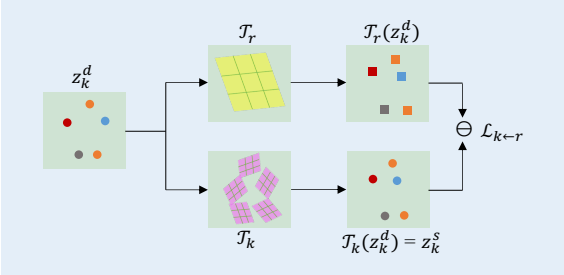

Figure 3. Illustration of Eqn. (5). The colored squares denote prior flow derived from the root anchor, as described in Eqn. (4), while the colored dots denote motion anchors. Euclidean distance is minimized between the pairs.

3.2. Deformable Anchor Model (DAM)

As discussed above, artifacts can be observed in the generated videos by FOMM. This is largely due to the motion anchors in FOMM are not properly regularized. Although different anchors are summed by the $M_k(z)$'s through Eqn (3), we observe that $M_k(z)$ tends to focus on only a local region around the anchor z_k due to the assumption of affine transformation. As a result, the z_k^s and z_k^d predicted by the motion estimator may not be accurately corresponded, leading to errors in the dense motion flow and artifacts in the generated video.

To address this issue, we propose a new deformable anchor model (DAM) to discover the motion structure information of the object, then employ this information to regularize the motion anchors. In more detail, our model is inspired by DPM [8]. We introduce an additional *latent root anchor* to establish communications among motion anchors. In a similar spirit to DPM, by connecting motion anchors with the root anchor, we expect the model to become aware of the motion structure of an object, even if its appearance varies in source images and driving videos.

Intuitively, given a source image and a driving frame, the root anchor represents the global motion between the two objects, which means that the flow of motion anchors should be related to that of the root anchor. Let z_r^d denote the latent root anchor of the driving frame. We then model the relation between the motion and root anchors with an affine transformation, as follows:

$$\mathcal{T}_r\left(z_k^d\right) = \mathcal{T}_r\left(z_r^d\right) + \theta_r\left(z_k^d - z_r^d\right) \tag{4}$$

where $\mathcal{T}_r\left(z_k^d\right)$ is the derived flow based on the latent root anchor using the affine transformation model. We then regularize the motion flow of z_k^d to be similar to the derived flow using the following loss:

$$\mathcal{L}_{k\leftarrow r} = \left\|\mathcal{T}_k(z_k^d) - \mathcal{T}_r\left(z_k^d\right)\right\|_2 \tag{5}$$

A further explanation of Eqn. (4) and (5) is provided in Fig. 3. In a departure from the original FOMM, where the motion anchors are almost independent, we encode a latent object structure to regularize the motion anchors. Eqn. (4) implies that we assume an affine transformation relation between the flow of the root anchor and motion anchors. While this may be stricter than required, divergence from ideal cases is permitted, and we use the derived flow as a prior to regularize the motion anchors with Eqn. (5).

On the other hand, through the use of Eqn. (4) and (5), the motion anchors also guide us to learn a meaningful root anchor. As shown in Fig. 6, the root anchor is always located at the object centroid to capture the global movement of the object from one image to another.

It should further be noted that, at the training stage, the latent root anchor z_r^d and the affine transformation parameters θ_r can be obtained by the motion estimator in a similar way as the motion anchors. At the testing stage, the root anchor is discarded, and we only need to use the predicted motion anchors to generate dense motion flow in the same way as FOMM. The overall architecture of our method is illustrated in Fig. 4.

3.3. Hierarchical DAM

As discussed above, using an affine transformation to model the structure prior might be too restrictive, especially for objects with complicated motion. Taking the human body as an example, a movable part (*e.g.*, left leg) might contain multiple joints, meaning that a single affine transformation can scarcely be expected to describe such a complex structure prior.

This motivates us to construct a hierarchical deformable anchor model to facilitate the modeling of more complicated object structures. In more detail, we additionally introduce a set of latent intermediate anchors into the basic deformable anchor model. Rather than directly regularizing the motion anchors with the latent root anchor, we instead use latent intermediate anchors to regularize motion anchors and the latent root anchor to regularize latent intermediate anchors. Similarly, the affine transformation prior is applied between different types of anchors. Let z_i^d denote an intermediate anchor; accordingly, we have:

$$\mathcal{T}_r\left(z_i^d\right) = \mathcal{T}_r\left(z_r^d\right) + \theta_r\left(z_i^d - z_r^d\right) \tag{6}$$

$$\mathcal{T}_i\left(z_k^d\right) = \mathcal{T}_i\left(z_i^d\right) + \theta_i\left(z_k^d - z_i^d\right) \tag{7}$$

where θ_r and θ_i are affine transformation parameters of the root anchor z_r^d and intermediate anchor z_i^d, while $\mathcal{T}_r$ and $\mathcal{T}_i$ are the respective derived flows.

Accordingly, the loss for regularizing the motion flow of motion anchors can be written as follows:

$$\mathcal{L}_{k\leftarrow i} = \left\|\mathcal{T}_k(z_k^d) - \mathcal{T}_i\left(z_k^d\right)\right\|_2 \tag{8}$$

$$\mathcal{L}_{i\leftarrow r} = \left\|\mathcal{T}_i(z_i^d) - \mathcal{T}_r\left(z_i^d\right)\right\|_2 \tag{9}$$

Note that although the loss $\mathcal{L}_{k\leftarrow i}$ is defined for every pair of z_i^d and z_k^d, we expect it takes effect on several z_k^d's nearby

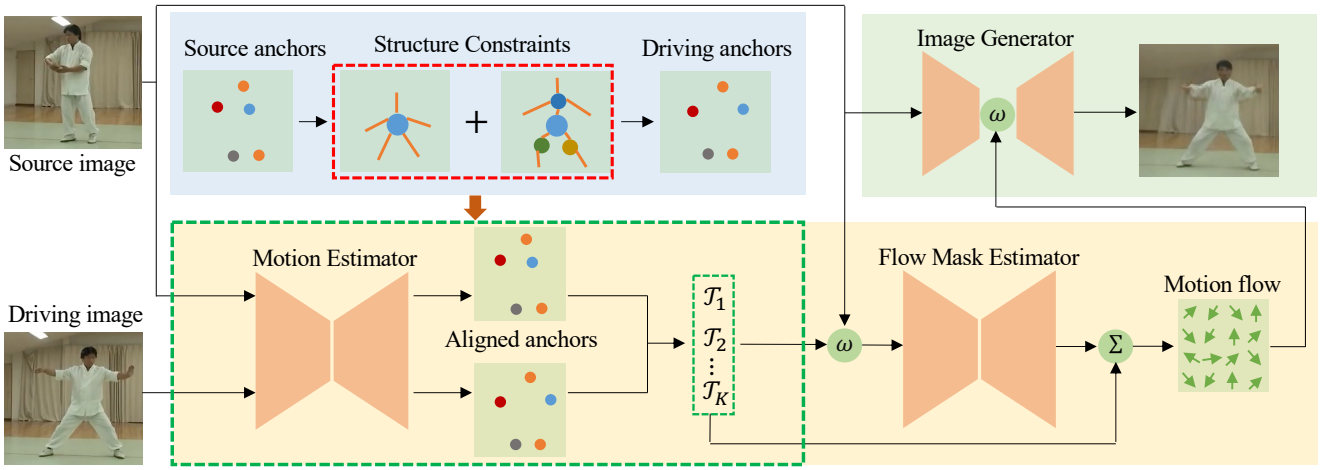

Figure 4. Overview of the proposed method. Anchors of the source image and driving image are respectively predicted through the motion estimator (we draw five motion anchors for clarity). The generated anchors are then are fed into a flow mask estimator together with the source image. The motion anchors and the flow masks are subsequently combined to obtain the dense warping flow for image generation. Note that motion anchors are constrained by the root anchor (*e.g.* the largest dot) and intermediate root anchors (*e.g.* medium-sized dots).

z_i^d only. Therefore, in implementation, we assign attention weights to all z_k^d's for each z_i^d, and allow the model to adjust these weights automatically.

With latent intermediate anchors, we can model a three-level hierarchical structure for object motion. To this end the procedure illustrated in Fig. 3 can be further extended, where image pixels are involved, above which are the motion anchors, intermediate anchors and the root anchor respectively. By applying the affine transformation prior between the adjacent levels, we are able to model more complex object structures.

3.4. Training DAM and HDAM

In both the basic deformable anchor model and hierarchical deformable anchor model, the newly introduced latent root anchors and the latent intermediate anchors can be predicted by the motion estimator network, which can be trained similarly to FOMM, *i.e.* in an end-to-end fashion by optimizing the image reconstruction loss.

More specifically, following FOMM [24], we utilize the perceptual loss as our main driving loss, which is usually defined with a pre-trained VGG-19 networks [27]. Given a driving image D, the perceptual loss can be expressed as follows:

$$\mathcal{L}_{per} = \frac{1}{C \cdot H \cdot W} \sum_{l} \left\| \phi_l(D) - \phi_l(\tilde{D}) \right\| \tag{10}$$

where $\tilde{D}$ is the generated driving image, ϕ_l denotes the feature extractor using the l-th layer of the VGG-19 network, and C, H, W denote the number of channels, feature map height and width respectively.

Additionally, similar to recent works [24, 40], an equivariance loss is adopted to ensure the geometric consistency of the learned anchors. For a known geometric transformation $\mathbf{T}$ and a given image I, the loss is defined as follows:

$$\mathcal{L}_{equi} = \sum_{k} \left\| z_k^I - \mathbf{T}^{-1}(z_k^{\mathbf{T}(I)}) \right\| \tag{11}$$

Training DAM: For the basic deformable anchor model, we write the loss for regularizing motion anchors as follows:

$$\mathcal{L}_{dam} = \sum_{k=1}^{K} \mathcal{L}_{k \leftarrow r} \tag{12}$$

where $\mathcal{L}_{k \leftarrow r}$ is defined in Eqn. (5).

The total training loss of our DAM model can be defined as:

$$\mathcal{L} = \mathcal{L}_{per} + \mathcal{L}_{equi} + \mathcal{L}_{dam} \tag{13}$$

where we apply equal weights for all losses.

Training HDAM: For the hierarchical deformable anchor model, assuming a total of I intermediate anchors are used, the loss can be written as follows:

$$\mathcal{L}_{hdam} = \sum_{i}^{I} \left(\sum_{k=1}^{K} \omega_{ik} \mathcal{L}_{k \leftarrow i} + \mathcal{L}_{i \leftarrow r} \right) \tag{14}$$

where $\mathcal{L}_{k \leftarrow i}$ and $\mathcal{L}_{i \leftarrow r}$ are respectively defined in Eqn. (8) and Eqn. (9); moreover, ω_{ik} denotes the attention weight between motion anchor k and intermediate anchor i, which is computed through a fully connected layer. More detailed information about the attention process is presented in the supplementary material.

The total training loss of our HDAM model can be defined as:

$$\mathcal{L} = \mathcal{L}_{per} + \mathcal{L}_{equi} + \mathcal{L}_{hdam} \tag{15}$$

where we also apply equal weights for all losses. In practice, when training HDAM, we use a pretrained DAM model as the initial model, then optimizing the loss in Eqn. (15).

4. Experiments

In this section, we evaluate our method on the benchmark datasets, and further provide insightful analysis by means of an ablation study and qualitative results.

4.1. Experimental Setup

Datasets: We follow FOMM [24] and RegionMM [26] in evaluating our method on four benchmark datasets containing different types of object:

- TaiChiHD [24] contains 2867 training videos and 253 test videos. This dataset contains Tai-chi performers with different identities and various backgrounds, and is thought to be the most challenging dataset in this area due to its large motion. Two resolution variants of this dataset are evaluated: 1) all raw videos are cropped and resized to the basic 256×256 resolution, as with FOMM; 2) the 512×512 resolution, is a subset that removes any raw videos that fail to satisfy the resolution request for cropping, which contains 962 training videos and 112 testing videos.
- FashionVideo [38] contains 500 training videos and 100 test videos. Videos in this dataset depict a single posing model with diverse clothing and textures. All videos are resized to a 256×256 resolution.
- MGIF collected in [23], is a cartoon animal dataset containing 900 training videos and 100 test videos. All videos are resized to a 256×256 resolution.
- VoxCeleb1 [16] is a talking head dataset, containing 19522 training 525 test videos. All videos are resized to a 256×256 resolution.

Evaluation protocols: Since ground-truth videos are not available for use in evaluating generated videos for the motion transfer task, we follow the FOMM [24] evaluation protocol and take self-reconstruction as a proxy task to quantitatively evaluate the proposed method. More specifically, an input video is reconstructed from the appearance representation of its first frame and the motion flow of the entire videos according to Eqn. (3). The same four different metrics as in [24] are used for evaluation.

- $\mathcal{L}_1$. The average $\mathcal{L}_1$ distance between the pixel values of generated and ground-truth video frames.
- Average Keypoint Distance (AKD). This metric computes the average keypoint distance between generated and ground-truth video frames. It is designed to evaluate the pose quality of the generated video frames.
- Missing Keypoint Rate (MKR). For human body datasets, we further report MKR, which represents the percentage of keypoints that are not detected in generated video frames but are localized in the ground truth video frames.
- Average Euclidean Distance (AED). This metric is designed to assess the identity quality of generated video frames based on specific feature representations; in the feature space, the average Euclidean distance between generated and ground-truth video frames is computed.

Implementation details: Seen in the supplementary.

4.2. Comparison with Existing Methods

We compare our method with two recent model-free motion transfer methods: FOMM [24] and RegionMM [26].

Quantitative results: The comparisons are summarized in Table 1. We can observe that our proposed HDAM approach generally achieves the best performance on all evaluation metrics. In particular, the fact that our $\mathcal{L}_1$ score is the lowest reflects the good quality of the videos generated by our method. Moreover, the improvement to AKD and MKR indicates that our method achieves good motion transfer, while the improved AED also reflects the appearance quality of the videos generated using our method.

In more detail, compared to the FOMM method, we achieve a notable improvement on the TaiChiHD, FashionVideo and MGIF dataset, while also gaining better results on the VoxCeleb dataset. This clearly proves the effectiveness of using deformable anchor models to regularize motion anchors. Moreover, the fact that our work outperforms the most recent related work, RegionMM, further proves the advantages of modeling object structure; notably, this superiority also holds in the case of higher-resolution inputs. We further note that the improvements on the VoxCeleb dataset are not as significant as those on the TaiChiHD and MGIF datasets. This is possibly because the structures of the human face are relatively simple, while the human body consists of multiple joints and movable parts, meaning that its motion are usually quite complicated. These results reflect that the deformable anchor model helps to transfer motion on various objects, especially those with complicated structures, which also validates the motivation of this work.

User study: We conduct a user study for cross-identity motion transfer. More specifically, we prepare 50 concatenated results consisting of a source frame, driving videos and videos generated by FOMM, RegionMM and our method; the synthesized videos are placed in random order in each of the concatenated videos. Fifty participants are asked to rank the three videos based on the appearance preservation and transferred motion. As Table 2 shows, participants clearly identified our videos as being of higher quality than the synthesized videos produced by existing methods.

Figure 5. Qualitative comparisons on cross-identity motion transfer. We present four source identities driven by two videos from the TaiChiHD dataset. It can be seen that our method generally synthesizes the most structure-stable results.

	TaiChiHD 256 × 256			TaiChiHD 512 × 512			Fashion			VoxCeleb1			MGIF
	L1	(AKD, MKR)	AED	L1	(AKD, MKR)	AED	L1	(AKD, MKR)	AED	L1	AKD	AED	L1
Monkey-Net	0.077	(10.798, 0.059)	0.228	-	-	-	-	-	-	0.049	1.89	0.199	-
FOMM	0.057	(6.649, 0.036)	0.172	0.065	(15.08, 0.061)	0.202	0.013	(1.142,0.005)	0.059	0.041	1.28	0.133	0.0224
RegionMM	0.048	(5.246, 0.024)	0.150	0.057	(11.97, 0.028)	0.166	**0.011**	(1.187,0.005)	0.056	0.040	1.28	0.133	0.0206
Ours DAM	0.045	(5.102, 0.024)	0.150	0.054	(10.83, 0.032)	0.158	**0.011**	(1.116, 0.005)	0.055	0.040	1.26	0.130	0.0207
Ours HDAM	**0.044**	(**4.790, 0.021**)	**0.146**	**0.053**	(**10.19, 0.027**)	**0.156**	**0.011**	(**1.041, 0.004**)	**0.054**	**0.039**	**1.24**	**0.124**	**0.0201**

Table 1. Quantitative comparisons on the self-reconstruction task. We present results on four benchmarks; here, a lower score is preferred for all metrics. For fair comparison, motion anchors are set to 10 for all methods.

	TaiChiHD	Fashion	Voxceleb1
Ours vs FOMM	91.6%	76.0%	54.2%
Ours vs RegionMM	59.1%	66.6%	66.0%

Table 2. User preferences favoring our approach.

Qualitative results: We additionally present examples of the videos generated by the three methods in Fig. 5. Generally speaking, FOMM often synthesizes an abnormal body shape or an incorrect motion from the driving video. Moreover, while RegionMM is able to roughly depict the motion contained in the driving video, it may also fail to capture more detailed structural information, leading to obvious artifacts (*e.g.*, the lost or weirdly warped human arms). By contrast, our method is generally able to capture the motion details well and produces more stable results. More qualitative results are provided in the supplementary material.

4.3. Ablation Study

We next conduct an ablation study to analyze the impact of our proposed components. Specifically, we study two variants of our proposed approach: 1) the basic deformable anchor model in Section 3.2 (referred to as "Ours (DAM)"), and 2) the hierarchical deformable anchor model in Section 3.3 (referred to as "Ours (HDAM)"). We further employ FOMM in which no deformable anchor model is used, as a baseline for comparison.

As seen in Table 1, we conduct experiments on the TaiChiHD dataset and analyze the results. We observe that *Ours (DAM)* achieves considerable improvements relative to the baseline FOMM, confirming the validity of exploiting object structures with a deformable anchor model in order to improve motion transfer. Moreover, by introducing the hierarchical deformable anchor model, *Ours (HDAM)*) achieves further improvements.

In Fig. 6, we present qualitative examples of our ablation study to reveal how our method works. We draw predicted anchors on generated frames to facilitate detailed analysis. As can be seen from the figure, FOMM generally fails to capture the local structure of the human body (such as hands and legs) due to the incorrectly aligned motion anchors; by contrast, *Ours(DAM)* can synthesize a relatively complete object structure, reflecting the effectiveness of DAM in constraining the object structure. Furthermore, *Ours(HDAM)* generally learns the meaningful structure and synthesizes high-quality results while capturing stable and complete structure information, which further verifies the superiority of modeling the hierarchical object structure.

4.4. Parameter Analysis

To validate the robustness of the proposed method, we study the influence of different hyper-parameter settings in this section. Specifically, we examine two important hyper-parameters in our model, namely the number of motion an-

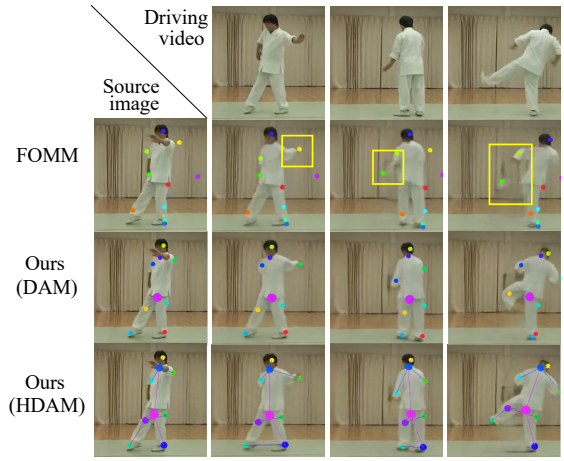

Figure 6. Qualitative ablation study. We visualize the latent root anchor as the largest dot and denote the intermediate anchors using medium-sized dots; correspondingly, the smallest dots represent motion anchors. Adjacent anchors are connected through straight lines according to the max attention weight, which reflects the constraint relation. Note that all learned intermediate anchors are overlapped with a motion anchor in this dataset; further analysis of this is provided in the supplementary material.

chors and the number of intermediate anchors. We conduct experiments on the TaiChiHD and MGIF datasets to perform this analysis. The quantitative results are obtained from the *Ours (HDAM)* model. As seen in Table 3, our method continues to improve as more motion anchors are involved, with a large improvement visible when moving from 5 to 10 motion anchors and a relatively small one from 10 to 20 motion anchors. Moreover, our method generally works well with $2 \sim 6$ intermediate anchors, as can be seen from Table 4. Our observations further suggest that, when more intermediate anchors are involved, the HDAM model tends to learn that some of them are meaningless or overlapping with other intermediate anchors; we explain that the object structure is overfitted in these situations. Overall, our method generalizes well to these hyper-parameters.

4.5. Structure Visualizations

To understand the proposed methods in more depth, we visualize the predicted hierarchical anchors of video frames on different datasets in Fig. 1 and Fig. 6; more qualitative results are provided in the the supplementary material. As is evident from the results, the learned root anchor is always located at the object centroid regardless of its identity or background; moreover, intermediate root anchors are often located at different local regions of an object, which enables them to capture more detailed motions of the parts in question. Note that in our hierarchical model, as seen in the fourth row of Fig. 6, when different motions occur, motion anchors can be regularized by different intermediate anchors according to the attention weights in Eqn. (14).

	MGIF	TaiChiHD		
	L1	L1	(AKD, MKR)	AED
5	0.0235	0.048	(5.730, 0.028)	0.159
10	0.0201	0.044	(4.790, 0.021)	0.146
20	**0.0185**	**0.043**	**(4.615, 0.018)**	**0.138**

Table 3. Quantitative performance with different number of motion anchors. We assess performance at 5, 10, 20 motion anchors respectively. The number of intermediate anchors is fixed at 3.

	MGIF	TaiChiHD		
	L1	L1	(AKD, MKR)	AED
2	0.0202	0.045	(**4.763, 0.021**)	**0.146**
3	0.0201	**0.044**	(4.790, **0.021**)	**0.146**
4	0.0201	**0.044**	(4.836, 0.022)	**0.146**
5	**0.0199**	0.045	(4.926, 0.023)	**0.146**
6	0.0200	**0.044**	(4.792, 0.022)	**0.146**

Table 4. Quantitative performance with different numbers of intermediate anchors. We tune $2 \sim 6$ intermediate anchors respectively. The number of motion anchors is fixed at 10.

This reflects the ability of our HDAM model to flexibly constrain the motion structures according to the varying motions in the dataset. In summary, this star-like motion structure learned by our deformable anchor model exhibits a strong ability to model stable motions between images.

5. Conclusion

This paper proposes a novel structure-aware motion transfer approach with deformable anchor model. In DAM, the latent root anchor is designed to constrain the motion anchors. We then explore the intermediate latent root anchors to build hierarchical DAM, leading to structure-stable motion transfer and yielding the best performance (both qualitatively and quantitatively) relative to existing benchmarks. We further interpret our method through insightful an ablation study and validate the robustness of our method to different hyper-parameter settings.

Societal impact and limitations: Motion transfer techniques could be misused for generating fake videos, which might bring negative societal impact. People should be cautious and get authorized when manipulating videos using these techniques. Moreover, while we demonstrate state-of-the-art performance, the results are not perfect. Some artifacts can still be observed when there exists occlusion, large motion, complex background, *etc*. We will study these issues in the future.

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), Beijing Natural Science Foundation (Z190023), and Alibaba Group through Alibaba Innovation Research Program.

References

[1] Guha Balakrishnan, Amy Zhao, Adrian V Dalca, Fredo Durand, and John Guttag. Synthesizing images of humans in unseen poses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8340–8348, 2018. 2

[2] Egor Burkov, Igor Pasechnik, Artur Grigorev, and Victor Lempitsky. Neural head reenactment with latent pose descriptors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13786–13795, 2020. 2

[3] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017. 3

[4] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5933–5942, 2019. 1, 2

[5] Xu Chen, Jie Song, and Otmar Hilliges. Unpaired pose guided human image generation. In *Conference on Computer Vision and Pattern Recognition (CVPR 2019)*. Computer Vision Foundation (CVF), 2019. 2

[6] Zhuo Chen, Chaoyue Wang, Bo Yuan, and Dacheng Tao. Puppeteergan: Arbitrary portrait animation with semantic-aware appearance transformation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13518–13527, 2020. 2

[7] Haoye Dong, Xiaodan Liang, Xiaohui Shen, Bowen Wu, Bing-Cheng Chen, and Jian Yin. Fw-gan: Flow-navigated warping gan for video virtual try-on. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1161–1170, 2019. 1

[8] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2009. 2, 3, 4

[9] Kuangxiao Gu, Yuqian Zhou, and Thomas Huang. Flnet: Landmark driven fetching and learning network for faithful talking facial animation synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10861–10868, 2020. 2

[10] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks through conditional image generation. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 4020–4031, 2018. 2, 3

[11] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. Deep video portraits. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018. 2

[12] Yining Li, Chen Huang, and Chen Change Loy. Dense intrinsic appearance flow for human pose transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3693–3702, 2019. 2

[13] Wen Liu, Zhixin Piao, Jie Min, Wenhan Luo, Lin Ma, and Shenghua Gao. 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*, pages 5904–5913, 2019. 2

[14] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. *Advances in Neural Information Processing Systems*, 30:406–416, 2017. 2

[15] Liqian Ma, Qianru Sun, Stamatios Georgoulis, Luc Van Gool, Bernt Schiele, and Mario Fritz. Disentangled person image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 99–108, 2018. 2

[16] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*, 2017. 6

[17] Natalia Neverova, Riza Alp Guler, and Iasonas Kokkinos. Dense pose transfer. In *Proceedings of the European conference on computer vision (ECCV)*, pages 123–138, 2018. 2

[18] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016. 3

[19] Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. Ganimation: Anatomically-aware facial animation from a single image. In *Proceedings of the European conference on computer vision (ECCV)*, pages 818–833, 2018. 2

[20] Jian Ren, Menglei Chai, Oliver J Woodford, Kyle Olszewski, and Sergey Tulyakov. Flow guided transformable bottleneck networks for motion retargeting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10795–10805, 2021. 2

[21] Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H Li, and Ge Li. Deep image spatial transformation for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7690–7699, 2020. 2

[22] Kripasindhu Sarkar, Dushyant Mehta, Weipeng Xu, Vladislav Golyanik, and Christian Theobalt. 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*, page 596–613, Berlin, Heidelberg, 2020. Springer-Verlag. 2

[23] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2377–2386, 2019. 2, 6

[24] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *Advances in Neural Information Processing Systems*, 2019. 1, 2, 3, 5, 6

[25] Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuiliere, and Nicu Sebe. Deformable GANs for pose-based human

image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3408–3416, 2018. 2

[26] Aliaksandr Siarohin, Oliver Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. In *CVPR*, 2021. 2, 3, 6

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[28] Soumya Tripathy, Juho Kannala, and Esa Rahtu. Facegan: Facial attribute controllable reenactment gan. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1329–1338, 2021. 2

[29] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 1152–1164, 2018. 2

[30] Ting-Chun Wang, Arun Mallya, and Ming-Yu Liu. One-shot free-view neural talking-head synthesis for video conferencing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10039–10049, 2021. 2

[31] Dongxu Wei, Xiaowei Xu, Haibin Shen, and Kejie Huang. 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. 2

[32] Yuxiang Wei, Ming Liu, Haolin Wang, Ruifeng Zhu, Guosheng Hu, and Wangmeng Zuo. Learning flow-based feature warping for face frontalization with illumination inconsistent supervision. In *European Conference on Computer Vision*, pages 558–574. Springer, 2020. 2

[33] Olivia Wiles, A Koepke, and Andrew Zisserman. X2face: A network for controlling face generation using images, audio, and pose codes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–686, 2018. 2

[34] Zhuoqian Yang, Wentao Zhu, Wayne Wu, Chen Qian, Qiang Zhou, Bolei Zhou, and Chen Change Loy. Transmomo: Invariance-driven unsupervised video motion retargeting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5306–5315, 2020. 2

[35] Guangming Yao, Yi Yuan, Tianjia Shao, Shuang Li, Shanqi Liu, Yong Liu, Mengmeng Wang, and Kun Zhou. One-shot face reenactment using appearance adaptive normalization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35:3172–3180, May 2021. 2

[36] Jae Shin Yoon, Lingjie Liu, Vladislav Golyanik, Kripasindhu Sarkar, Hyun Soo Park, and Christian Theobalt. 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)*, pages 15039–15048, June 2021. 2

[37] Xiang Yu, Feng Zhou, and Manmohan Chandraker. Deep deformation network for object landmark localization. In *European Conference on Computer Vision*, pages 52–70. Springer, 2016. 3

[38] Polina Zablotskaia, Aliaksandr Siarohin, Bo Zhao, and Leonid Sigal. Dwnet: Dense warp-based network for pose-guided human video generation. In *BMVC*, page 51, 2019. 6

[39] Jinsong Zhang, Kun Li, Yu-Kun Lai, and Jingyu Yang. Pise: Person image synthesis and editing with decoupled gan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7982–7990, 2021. 2

[40] Yuting Zhang, Yijie Guo, Yixin Jin, Yijun Luo, Zhiyuan He, and Honglak Lee. Unsupervised discovery of object landmarks as structural representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2694–2703, 2018. 3, 5

[41] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Learning deep representation for face alignment with auxiliary attributes. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):918–930, 2015. 3

[42] Zhen Zhu, Tengteng Huang, Baoguang Shi, Miao Yu, Bofei Wang, and Xiang Bai. Progressive pose attention transfer for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2347–2356, 2019. 2