MOTION REPRESENTATIONS FOR ARTICULATED ANIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose novel motion representations for animating articulated objects consisting of distinct parts. In a completely unsupervised manner, our method identifies meaningful object parts, tracks them in a driving video, and infers their motions by considering their principal axes. In contrast to the previous keypoint-based works, our method extracts meaningful and consistent *regions*, describing locations, shape, and pose. The regions correspond to semantically relevant and distinct object parts, that are more easily detected in frames of the driving video. To force decoupling of foreground from background, we model non-object related global motion with a homography. Our model[1] can animate a variety of objects, surpassing previous methods by a large margin on existing benchmarks. We present a challenging new benchmark with high-resolution videos and show that the improvement is particularly pronounced when articulated objects are considered.

1 INTRODUCTION

Animation—bringing static objects to life—has broad applications across education and entertainment. Animated characters and objects increase the creativity and appeal of content, improve the clarity of material through storytelling, and enhance user experiences. Imagine the Mona Lisa describing the manner in which she was painted, or Michelangelo's David detailing the method with which he was sculpted, or an influential historical figure shedding light on key events of the past (Fig. 1); how much more engaging this would be.

Figure 1: An animation produced by our method.

Until very recently, animation techniques necessary for achieving such results required a trained professional, specialized hardware, software, and a great deal of effort. Quality results generally still do, but vision and graphics communities have attempted to address some of these limitations by training data-driven methods (Wang et al., 2018a; Chan et al., 2019; Ren et al., 2020; Geng et al., 2019; Gafni et al., 2019) on object classes for which prior knowledge of object shape and pose can be learned. This, however, requires ground truth pose and shape data to be available during training.

Recent works have sought to avoid the need for ground truth data through *unsupervised* motion transfer (Wiles et al., 2018; Siarohin et al., 2019a;b). Significant progress on the several key challenges have been made, including training using image reconstruction as a loss, and disentangling motion from appearance. This has created the potential to animate a broader range of object categories, without any domain knowledge or labelled data, requiring only videos of objects in motion during training. However, two key problems remain open. The first is how to represent the parts of an articulated or non-rigid moving object, including their shapes and poses. The second is given the object parts, how to animate them using the sequence of motions in a driving video.

Initial attempts involved extracting unsupervised keypoints (Lorenz et al., 2019; Kim et al., 2019) in end-to-end frameworks (Wiles et al., 2018; Siarohin et al., 2019b;a), then warping a feature embedding of a source image to align its keypoints with those of a driving video. Follow on work (Siarohin et al., 2019a) additionally modelled the motion around each keypoint with local, affine transformations,

[1] We plan to publish the source code and trained models along with the paper.

and introduced a generation module that both composites warped source image regions and inpaints occluded regions, to render the final image. This enabled a variety of creative applications[2], for example needing only one source face image to generate a near photo-realistic animation, driven by a video of a different face.

However, the resulting unsupervised keypoints are detected on the boundary of the objects. While points on edges are easier to identify, tracking such keypoints between frames is problematic, as any point on the boundary is a valid candidate, making it hard to establish correspondences between frames. A further problem is that the unsupervised keypoints do not correspond to semantically meaningful object parts, and represent location and direction, but not shape. Due to this limitation, animating articulated objects, such as bodies, remains challenging. Furthermore, these methods assume static backgrounds, i.e., no camera motion, leading to leakage of background motion information into one or several of the detected keypoints. Despite significant breakthroughs, these remaining deficiencies limit the scope of the core innovation to more trivial object categories and motions, and lower quality outputs, especially when objects are articulated.

This work introduces two contributions critical to addressing these challenges. First, we redefine the underlying motion representation. Instead of using keypoints, we switch to *regions* that allow first-order motion to be *measured*, rather than regressed. This enables improved convergence, more stable, robust object and motion representations, and also empirically captures the shape of the underpinning object parts, leading to better motion segmentation. Fig. 3 contains several examples of region vs. keypoint-based motion representation.

Secondly, we explicitly model background or camera motion between training frames by predicting the parameters of a global homography explaining non-object related motions. This enables the model to focus solely on the foreground object, making the identified points more stable, and further improves convergence.

These contributions unlock significant gains in capability for unsupervised motion transfer methods, resulting in much improved animation of articulated objects in particular. Furthermore, the framework scales better in the number of unsupervised regions, resulting in more detailed motion. Our method outperforms previous unsupervised animation methods on a variety of datasets, including talking faces, taichi videos and animated pixel art. We additionally present a new dataset, TED talk speakers, to create a more challenging benchmark for the task of animating articulated objects.

2 Related work

Image animation methods can be separated into supervised, which require knowledge about the animated object during training, and unsupervised, which do not. Such knowledge typically includes landmarks (Cao et al., 2014; Zakharov et al., 2019; Qian et al., 2019; Ha et al., 2020), semantic segmentations (Nirkin et al., 2019), and parametric 3D models (Geng et al., 2019; Thies et al., 2016; Deng et al., 2020; Nagano et al., 2018; Liu et al., 2019). As a result, supervised methods are limited to a small number of object categories for which a lot of labelled data is available, such as faces and human bodies. Early face reenactment work (Thies et al., 2016) fitted a 3D morphable model to an image, animating and rendering it back using graphical techniques. Further works used neural networks to get higher quality rendering (Kim et al., 2018; Wang et al., 2018b), sometimes requiring multiple images per identity (Geng et al., 2019; Pumarola et al., 2018). A body of works treats animation as an image-to-image (Siarohin et al., 2018) or a video-to-video (Wang et al., 2018a; Chan et al., 2019; Ren et al., 2020) translation problem. Apart from some exceptions (Wang et al., 2019), these works further constrain the problem to animating a single instance of an object, such as a single face (Kim et al., 2018; Bansal et al., 2018) or a single human body (Chan et al., 2019; Ren et al., 2020; Wang et al., 2018a), requiring retraining (Bansal et al., 2018; Chan et al., 2019; Ren et al., 2020) or fine-tuning (Zakharov et al., 2019) for each new instance. Despite promising results, generalizing these methods beyond a limited range of object categories remains challenging. Additionally, they tend to transfer not only the motion but also the identity of the driving object, making the shape of the animated face or a body similar or identical to the driving face or body (Kim et al., 2018; Zakharov et al., 2019).

[2] E.g. a music video in which static face images are animated using prior work (Siarohin et al., 2019a).

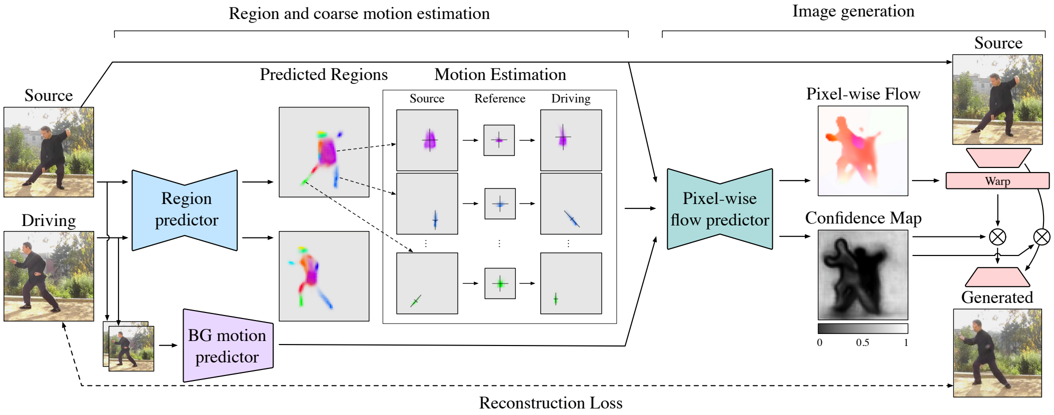

Figure 2: **Overview of our model.** The region predictor returns heatmaps for each part in the source and the driving images. We then compute principal axes of each heatmap, to transform each region from the source to the driving frame through a whitened reference frame. Region and background transformations are combined by the pixel-wise flow prediction network. The target image is generated by warping the source image in a feature space using the pixel-wise flow, and inpainting newly introduced regions, as indicated by the confidence map.

Unsupervised methods address some of these limitations. They do not require any labelled data regarding the shape or landmarks of the animated object. Video-generation-based animation methods predict future frames of a video, given the first frame and an animation class label, such as "make a happy face", "do jumping jack", or "play golf" (Tulyakov et al., 2018; Saito et al., 2017; Clark et al., 2019). A further group of works re-target animation from a driving video to a source frame. X2Face (Wiles et al., 2018) builds a canonical representation of an input face, and generates a warp field conditioned on the driving video. Monkey-Net (Siarohin et al., 2019b) learns a set of unsupervised keypoints to generate animations. Follow-up work substantially improves the quality of animation by considering a first order motion model (FOMM) (Siarohin et al., 2019a) for each keypoint, represented by regressing a local, affine transformation. Both of these works apply to a wider range of objects including faces, bodies, robots, and pixel art animations. Empirically, these methods extract keypoints on the boundary of the animated objects. Articulated objects such as human bodies are therefore challenging, as internal motion, for example, an arm moving across the body, is not well modeled, producing unconvincing animations.

This work presents an unsupervised method. We argue that the limitations of previous such methods in animating articulated objects is due to an inability of their internal representations to capture complete object parts, their shape and pose. X2Face (Wiles et al., 2018) assumes an object can be represented with a single RGB texture, while other methods find keypoints on edges (Siarohin et al., 2019b;a). Our new region and background motion representations address these shortcomings.

3 Method

Our unsupervised animation framework consists of a system design, and methods for training this system using two different frames, source $\mathbf{S}$, and driving $\mathbf{D}$, from the same video.

3.1 System design

FOMM (Siarohin et al., 2019a), the current state-of-the-art method in unsupervised animation learning, consists of two main parts: motion estimation and image generation. The contributions of our work lie in novel motion representations within the first part of this framework. Our system, outlined in Fig. 2, therefore follows the FOMM design as closely as possible, in order to demonstrate the impact due specifically to our contributions.

Table 1: Comparing our model with FOMM (Siarohin et al., 2019a) on TaiChiHD (256), for $K = 5$, 10 and 20. (Best result in bold.)

	5 regions			10 regions			20 regions		
	$\mathcal{L}_1$	(AKD, MKR)	AED	$\mathcal{L}_1$	(AKD, MKR)	AED	$\mathcal{L}_1$	(AKD, MKR)	AED
FOMM (Siarohin et al., 2019a)	0.062	(7.34, 0.036)	0.181	0.056	(6.53, 0.033)	0.172	0.062	(8.29, 0.049)	0.196
Ours w/o skip & bg	0.060	(6.44, 0.030)	0.169	0.058	(5.60, **0.026**)	0.162	0.057	(5.36, **0.026**)	0.156
Ours	**0.048**	(**6.09**, **0.029**)	**0.159**	**0.048**	(**5.45**, 0.028)	**0.152**	**0.046**	(**5.30**, **0.026**)	**0.142**

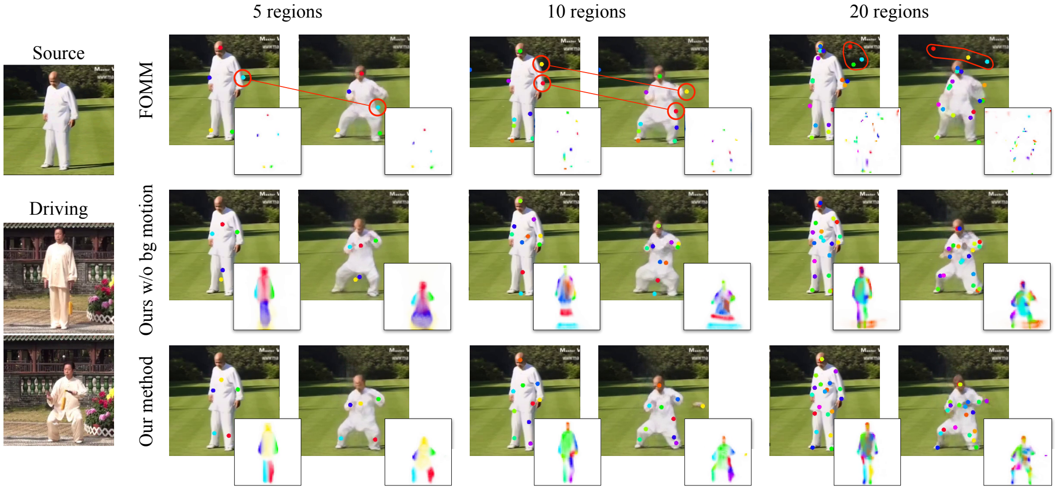

Figure 3: **Comparison of motion/part representations.** Regression-based keypoint representations do not provide consistent detection between frames (marked with red). Additionally, background motion leaks into one or several detected keypoints. Our PCA-based regions (with and without background motion) correctly identify meaningful parts, are consistent between frames, and use additional regions more effectively.

3.1.1 Regions and Coarse Motion

Regions FOMM learns to detect K distinct object regions, where K is a user-defined parameter. An encoder-decoder region predictor network takes an image as input, and outputs K heatmaps, $\mathbf{M}^1, .., \mathbf{M}^K$. The final network layer is a softmax operation, s.t. $\mathbf{M}^k \in [0,1]^{H \times W}$, where H and W are the height and width of the image respectively, and $\sum_{z \in \mathcal{Z}} m_z^k = 1$, where z is a pixel location (x, y coordinates) in the image, the set of all pixel locations being $\mathcal{Z}$, and m_z^k is the k-th heatmap weight at pixel z. We use the same region representation and encoder here. Nevertheless, the encoded regions differ significantly (see Fig. 3), ours mapping to meaningful object parts such as the limbs of an articulated body, due to our novel foreground motion representation, described below.

Estimating foreground region motion FOMM estimates a first-order transformation from an image $\mathbf{X}$ to a reference frame $\mathbf{R}$, for each region separately. The region heatmap encodes translation by its mean position, while other affine parameters are regressed per pixel and then pooled per region according to the heatmap weights. Here we change the way this transformation is represented: *all* motion is measured directly from the heatmap. Translation is given by the mean position, as before, while in-plane rotation and scaling in x- and y-directions are computed via a principal component analysis (PCA) of the heatmap. Shear is not captured, therefore our transform isn't fully affine, with only five degrees of freedom instead of six. Nevertheless, it captures sufficient motion, shear being a less significant component of the affine transform for this task. The transformation of a region from the reference frame to the image is computed as follows:

$$\mu^k = \sum_{z \in \mathcal{Z}} m_z^k z, \tag{1}$$

$$U^k S^k {V^k}^{\mathsf{T}} = \sum_{z \in \mathcal{Z}} m_z^k \left(z - \mu^k\right) \left(z - \mu^k\right)^{\mathsf{T}}, \quad \text{(via SVD)}, \tag{2}$$

$$A_{\mathbf{X} \leftarrow \mathbf{R}} = \left[U^k {S^k}^{\frac{1}{2}}, \mu^k \right]. \tag{3}$$

The singular value decomposition (SVD) approach to computing PCA (Wall et al., 2003) is used here. We refer to FOMM and our estimation approaches as *regression-based* and *PCA-based*, respectively. The reference frame in both is used only as an abstract, intermediate coordinate frame between the source and driving image coordinate frames. However, here (in contrast to FOMM) it is not in fact abstract, corresponding to the coordinate frame where the heatmap is whitened (i.e. has zero mean and identity covariance); see Fig. 2. Driving to source image motion is then

$$A^k_{\mathbf{S}\leftarrow\mathbf{D}} = A^k_{\mathbf{S}\leftarrow\mathbf{R}} \begin{bmatrix} A^k_{\mathbf{D}\leftarrow\mathbf{R}} \\ 0 \;\; 0 \;\; 1 \end{bmatrix}^{-1}. \tag{4}$$

Estimating background motion FOMM has no background motion model. We observe that with significant background motion between frames, e.g. due to camera motion, predicted regions can therefore include the moving background, reducing test-time accuracy. To resolve this, we additionally regress a background homography transformation, $\mathbf{H}$, using an encoder network that takes as input the source and driving images, concatenated along the channel dimension, and outputs eight real values, $h_1, .., h_8$, such that $\mathbf{H} = \left[[h_1, h_2, h_3]^\intercal \; [h_4, h_5, h_6]^\intercal \; [h_7, h_8, 1]^\intercal \right]$. With background motion well modeled, we show that the network is able to separate background and object motion in a completely unsupervised manner.

3.1.2 Image generation

Given these coarse motions, FOMM then renders the target image in two stages: a pixel-wise flow generator converts coarse motions to dense optical flow, then a composition network warps the source image according to the flow, and also inpaints missing regions. We follow this architecture, and summarize these two modules here, but refer the reader to Siarohin et al. (2019a) for the full details.

Pixel-wise flow generation Coarse motions are combined via a weighted sum, to compute a dense, per pixel motion, or flow. The per pixel weights, as well as a confidence map, are computed via an encoder-decoder network. The input is a $H \times W \times (4K + 3)$ tensor, with four channels per region, three for the source image warped according to the region's motion model, and one for a heatmap of the region, which is a gaussian approximation to $\mathbf{M}^k$, in order to avoid leakage of driving image appearance through the heatmap. Here we add a further three input channels (compared to FOMM) for the source image warped according to the background motion model.

Warping and inpainting The source image is passed through an encoder network. The resulting feature map is warped and masked according to the pixel-wise flow and confidence map from the previous module, respectively. A decoder then renders the final image, inpainting missing parts. In contrast to FOMM, but similar to Monkey-Net (Siarohin et al., 2019b), here skip connections are used between the encoder and decoder. The skip connection feature maps are also warped and masked.

3.2 Training

The proposed model is trained end-to-end using a reconstruction loss in the feature space of the pretrained VGG-19 network (Johnson et al., 2016; Wang et al., 2017). Following Siarohin et al. (2019a); Wang et al. (2003), we adopt a multi-resolution version of the reconstruction loss:

$$\mathcal{L}_{\text{rec}}(\hat{\mathbf{D}}, \mathbf{D}) = \sum_l \sum_i \left| \mathrm{V}_i(\mathrm{F}_l \odot \hat{\mathbf{D}}) - \mathrm{V}_i(\mathrm{F}_l \odot \mathbf{D}) \right|, \tag{5}$$

where $\hat{\mathbf{D}}$ is the generated image, V_i is the i^{th}-layer of the VGG-19 pretrained network, F_l is a downsampling operator. Similarly to Wang et al. (2017); Siarohin et al. (2019a) we used conv1_2, conv2_2, conv3_2, conv4_2, conv5_2 layers and downsampled the images to 1, 0.5, 0.25, 0.125 of the original edge size. In total we have 20 reconstruction terms. To improve detection of unsupervised regions we follow the unsupervised keypoint detection literature (Jakab et al., 2018; Zhang et al., 2018) and adopt the equivariance loss, denoted as $\mathcal{L}_{\text{eq}}$. We use a thin-plate spline implementation provided in FOMM (Siarohin et al., 2019a). The final loss is a sum of the two loss terms, $\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{eq}}$.

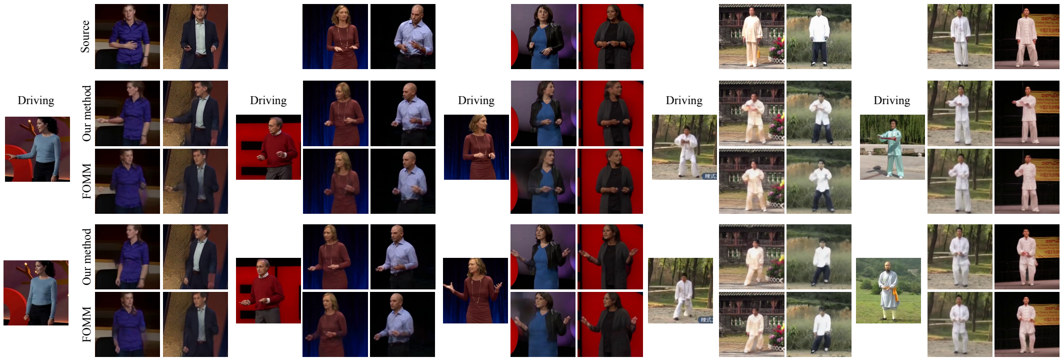

Figure 4: **Qualitative comparisons.** We show representative examples of articulated animation using our method and FOMM (Siarohin et al., 2019a), on two datasets of articulated objects: TED-talks (left) and TaiChiHD (right). Zoom in for greater detail.

4 Evaluation

We now discuss the datasets, metrics and experiments used to evaluate the proposed method. Later we compare with prior work, as well as ablate our contributions.

4.1 Toy Motion Representation Experiment

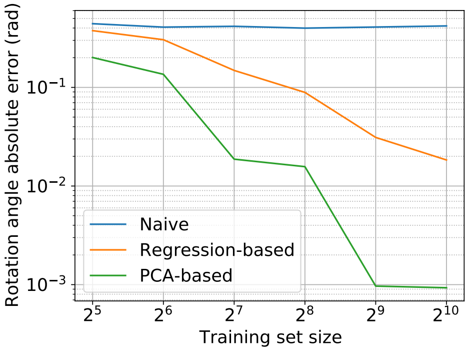

Figure 5: Mean test-time absolute rotation error, as a function of training set size.

To demonstrate the benefit of the proposed PCA-based motion representation, we devise an experiment on rotated rectangles (see Appendix E): the task is to predict the rotation angle of a rectangle in an image. To fully isolate our contribution, we consider a supervised task, where three different architectures learn to predict angles under the L_1 loss. The first, a Naive architecture, directly regresses the angle using an encoder-like architecture. The second is Regression-based, as in to FOMM (Siarohin et al., 2019a). The third uses our PCA-based approach (see Appendix E). Test results are presented in Fig. 5, against training set size. The Naive baseline struggles to produce meaningful results for any size of training set, while Regression-based performance improves with more data. However, the PCA-based significantly improves accuracy over the Regression-based one, being over an order of magnitude better with a large number of samples. This shows that it is significantly easier for the network to infer geometric parameters of the image, such as angle, using our proposed PCA-based representation.

4.2 Benchmarks

We evaluate our method on several benchmark datasets for animating human faces and bodies. Each dataset has separate training and test videos. The datasets are as follows:

- *VoxCeleb* (Nagrani et al., 2017) consists of interview videos of different celebrities. We extract square, face regions and downscale them to 256×256, following FOMM (Siarohin et al., 2019a). The number of frames per video ranging from 64 to 1024.
- *TaiChiHD* (Siarohin et al., 2019a) consists of cropped videos of full human bodies performing Tai Chi actions. We evaluate on two resolutions of the dataset: 256×256 (from FOMM (Siarohin et al., 2019a)), and a new, 512×512 subset, removing videos lacking sufficient resolution to support that size.
- *TED-talks* is a new dataset, collected for this paper in order to demonstrate the generalization properties of our model. We cropped the upper part of the human body from the videos, downscaling to 384×384. The number of frames per video ranges from 64 to 1024.

Table 2: Video reconstruction: comparison with the state of the art on four different datasets. For all methods we use $K = 10$ regions. (Best result in bold.)

	TaiChiHD (256)			TaiChiHD (512)			TED-talks			VoxCeleb		
	$\mathcal{L}_1$	(AKD, MKR)	AED	$\mathcal{L}_1$	(AKD, MKR)	AED	$\mathcal{L}_1$	(AKD, MKR)	AED	$\mathcal{L}_1$	AKD	AED
FOMM	0.056	(6.53, 0.033)	0.172	0.075	(17.12, 0.66)	0.203	0.033	(7.07, 0.014)	0.163	0.041	**1.27**	0.134
Ours	**0.048**	(**5.45**, **0.028**)	**0.152**	**0.064**	(**14.00**, **0.44**)	**0.171**	**0.026**	(**4.02**, **0.007**)	**0.119**	**0.040**	1.28	**0.133**

Further datasets are used in the supplementary material.

Since video animation is a relatively new problem, there are not currently many effective ways of evaluating it. For quantitative metrics, prior works (Siarohin et al., 2019b;a) use video reconstruction accuracy as a proxy for image animation quality. We adopt the same metrics here:

- $\mathcal{L}_1$ error measures the difference between reconstructed video and ground-truth video pixel values using the $\mathcal{L}_1$ metric.
- *Average keypoint distance* (AKD) and *missing keypoint rate* (MKR) evaluate the difference between poses of reconstructed and ground truth video. Landmarks are extracted from both videos using public, body (Cao et al., 2017) (for TaiChiHD and TED-talks) and face (Bulat & Tzimiropoulos, 2017) (for VoxCeleb) detectors. AKD is then the average distance between corresponding landmarks, while MKR is the proportion of landmarks present in the ground-truth that are missing in the reconstructed video.
- *Average Euclidean distance* (AED) evaluates how well identity is preserved in reconstructed video. Public re-identification networks for bodies (Hermans et al., 2017) (for TaiChiHD and TED-talks) and faces (Amos et al., 2016) extract identity from reconstructed and ground truth frame pairs, then we compute the average $\mathcal{L}_2$ norm of their difference across all pairs.

4.3 Comparison with the state of the art

We compare our method with the current state of the art for unsupervised animation, FOMM (Siarohin et al., 2019a), across all datasets, on both reconstruction (the training task) and animation (the test-time task). We used an extended training schedule compared to Siarohin et al. (2019a), with 50% more iterations. To compare fairly with FOMM (Siarohin et al., 2019a), we also re-trained it with the same training schedule.

Reconstruction quality Quantitative reconstruction results are reported in Tab. 2. We first show that our method reaches state-of-the-art results on a dataset with non-articulated objects such as faces. Indeed, when compared with FOMM (Siarohin et al., 2019a) on VoxCeleb our method shows on-par results. The situation changes, however, when articulated objects are considered, such as human bodies in TaiChiHD and TED-talks datasets, on which our improved motion representations boost all the metrics. The advantage over the state of the art holds at different resolutions, for TaiChiHD (256), TaiChiHD (512) and TED-talks, as well as for different numbers of selected regions (discussed later).

Animation quality Fig. 3 & 4 show selected and representative animations respectively, using our method and FOMM (Siarohin et al., 2019a), on articulated bodies, both using absolute motion. The results show clear improvements, in most cases, in animation quality, especially of limbs.

Animation quality was evaluated quantitatively through a user preference study similar to that of Siarohin et al. (2019a). AMT users were presented with the source image, driving video, and the output from our method and FOMM (Siarohin et al., 2019a), and asked which of the two videos they preferred. 50 such videos were evaluated, by 50 users each, for a total of 2500 preferences per study. The results, shown in Tab. 4, further support the reconstruction scores in Tab. 2. When the animated object is not articulated (VoxCeleb), the method delivers results comparable to the previous work. When bodies are animated (TaiChiHD & TED-talks), FOMM (Siarohin et al., 2019a) fails to correctly detect and animate the articulated body parts such as hands. Our method renders them in the driving pose even for extreme cases, leading to a high preference in favor of it.

Finally, we applied animation from a TED-talks video to a photograph of Winston Churchill, shown in Fig. 1, demonstrating animation of out of domain data.

Table 3: Ablation study on TaiChiHD (256) dataset with $K = 10$. (Best result in bold.)

	$\mathcal{L}_1$	(AKD, MKR)	AED
No pca or bg model	0.060	(6.14, 0.033)	0.163
No pca	0.049	(6.04, 0.034)	0.163
No bg model	0.059	(5.47, **0.027**)	0.164
Full method	**0.048**	(**5.45**, 0.028)	**0.152**

Table 4: User study: the proportion (%) of users that prefer our method over FOMM (Siarohin et al., 2019a).

Dataset	User preference (%)
VoxCeleb	52.2%
TaiChiHD (256)	83.0%
TED-talks	91.0%

4.4 Ablations

In order to understand how much benefit each of our contributions bring, we ran a number of ablation experiments, detailed in Tab. 3.

PCA-based vs. regression-based representations First we compare the PCA-based motion model with the previous, regression-based one (Siarohin et al., 2019a). From the qualitative, heatmap depictions in Fig. 3, we observe that the regression-based method localizes one edge of each corresponding part, while our method predicts regions that roughly correspond to the segmentation of the object into its constituent, articulated parts. This meaningful segmentation arises completely unsupervised.

From Tab. 1 we note that adding the PCA-based representation alone (second row) had marginal impact on the $\mathcal{L}_1$ score (dominated by the much larger background region), but it had a much larger impact on other metrics, which are more sensitive to object-part-related errors on articulated objects. This is corroborated by Tab. 3.

We intuit that PCA-based estimation both captures regions and improves performance because it is much easier for the convolutional network to assign pixels of an object part to the corresponding heatmap than to directly regress motion parameters to an abstract reference frame. This is borne out by our toy experiment (sec. 4.1). In order to estimate the heatmap it need only learn all appearances of the corresponding object part, whereas regression-based networks must learn the joint space of all appearances of a part in all possible geometric configurations (e.g. rotated, scaled etc.).

One of the most important hyper-parameters of our model is the number of regions, K. The qualitative and quantitative ablations of this parameter are shown in Fig. 3 and Tab. 1 respectively. We can observe that, while the regression-based representation fails when the number of keypoints grows to 20, our PCA-based representation scales well with the number of regions.

Modeling background motion Tab. 3 shows that methods with background motion modeling have much lower $\mathcal{L}_1$ error. Since background constitutes a large portion of the image, and $\mathcal{L}_1$ treats all pixels equally, this is to be expected. AED was also impacted, suggesting that the identity representation captures some background appearance. However, since AKD & MKR metrics evaluate object pose only, they are not improved by background modelling.

5 Conclusion

We have argued that previous unsupervised animation frameworks' poor results on articulated objects are due to their representations. We propose a new, PCA-based, region motion representation, which we believe both makes it easier for the network to learn region motion, and encourages it to learn semantically meaningful object parts. In addition, we propose a background motion estimation module to decouple foreground and background motion. Qualitative and quantitative results across a range of datasets and tasks demonstrate several key benefits: improved region distribution and stability, improved reconstruction accuracy and user perceived quality, and an ability to scale to more regions. We also introduce a new, more challenging dataset, TED-talks, for benchmarking future improvements on this task.

While we show some results on out of domain data (Fig. 1), generalization remains a significant challenge to making this method broadly practical in articulated animation of inanimate objects.

REFERENCES

Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition. 2016.

Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. Recycle-gan: Unsupervised video retargeting. In *Proceedings of the European Conference on Computer Vision*, 2018.

Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *ICCV*, 2017.

Chen Cao, Qiming Hou, and Kun Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on Graphics*, 2014.

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

A Clark, J Donahue, and K Simonyan. Adversarial video generation on complex datasets. *arXiv preprint arXiv:1907.06571*, 2019.

Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. Disentangled and controllable face image generation via 3d imitative-contrastive learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

Oran Gafni, Lior Wolf, and Yaniv Taigman. Vid2game: Controllable characters extracted from real-world videos. *arXiv preprint arXiv:1904.08379*, 2019.

Zhenglin Geng, Chen Cao, and Sergey Tulyakov. 3d guided fine-grained face manipulation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Sungjoo Ha, Martin Kersner, Beomsu Kim, Seokjun Seo, and Dongyoung Kim. Marionette: Few-shot face reenactment preserving identity of unseen targets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv:1703.07737*, 2017.

Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks through conditional image generation. In *Proceedings of the Neural Information Processing Systems Conference*, 2018.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision*, 2016.

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*, 2018.

Yunji Kim, Seonghyeon Nam, In Cho, and Seon Joo Kim. Unsupervised keypoint learning for guiding class-conditional video prediction. In *Proceedings of the Neural Information Processing Systems Conference*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014.

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 International Conference on Computer Vision*, 2019.

Dominik Lorenz, Leonard Bereska, Timo Milbich, and Bjorn Ommer. Unsupervised part-based disentangling of object shape and appearance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Koki Nagano, Jaewoo Seo, Jun Xing, Lingyu Wei, Zimo Li, Shunsuke Saito, Aviral Agarwal, Jens Fursund, and Hao Li. pagan: real-time avatars using dynamic textures. *ACM Transactions on Graphics*, 2018.

A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. In *INTERSPEECH*, 2017.

Yuval Nirkin, Yosi Keller, and Tal Hassner. Fsgan: Subject agnostic face swapping and reenactment. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

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*, 2018.

Shengju Qian, Kwan-Yee Lin, Wayne Wu, Yangxiaokang Liu, Quan Wang, Fumin Shen, Chen Qian, and Ran He. Make a face: Towards arbitrary high fidelity face manipulation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019.

Jian Ren, Menglei Chai, Sergey Tulyakov, Chen Fang, Xiaohui Shen, and Jianchao Yang. Human motion transfer from poses in the wild. *arXiv preprint arXiv:2004.03142*, 2020.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuilière, and Nicu Sebe. Deformable gans for pose-based human image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *Proceedings of the Neural Information Processing Systems Conference*, 2019a.

Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019b.

Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Michael E Wall, Andreas Rechtsteiner, and Luis M Rocha. Singular value decomposition and principal component analysis. In *A practical approach to microarray data analysis*, pp. 91–109. Springer, 2003.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

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 Neural Information Processing Systems Conference*, 2018a.

Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot video-to-video synthesis. In *Proceedings of the Neural Information Processing Systems Conference*, 2019.

Wei Wang, Xavier Alameda-Pineda, Dan Xu, Pascal Fua, Elisa Ricci, and Nicu Sebe. Every smile is unique: Landmark-guided diverse smile generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018b.

Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *Proceedings of the Thrity-Seventh Asilomar Conference on Signals, Systems & Computers*, 2003.

Olivia Wiles, A Sophia 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*, 2018.

Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

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*, 2018.

A TED-TALKS DATASET CREATION

To create the TED-talks dataset, we downloaded 3,035 YouTube videos, shared under the "CC BY – NC – ND 4.0 International" license[3], using the "TED talks" query. From these initial candidates, we selected the videos where the upper part of the person is visible for at least 64 frames, and the height of the person bounding box was at least 384 pixels. After that, we manually filtered out static videos and videos in which a person is doing something other than presenting. We ended up with 411 videos, and split these videos in 369 training and 42 testing videos. We then split each video into chunks from a consistent camera angle (i.e. with no cuts to another camera), and for which the presenter didn't move too far from their starting position in the chunk. We cropped the a square region around the presenter, such that they had a consistent scale, and downscaled this region to 384 × 384 pixels. Chunks that lacked sufficient resolution to be downscaled, or had a length shorter than 64 frames, were removed. Both the distance moved and the region cropping were achieved using a bounding box estimator for humans (Wu et al., 2019). Overall, we obtained 1,177 training video chunks and 145 test videos chunks.

B IMPLEMENTATION DETAILS

For a fair comparison, in order to highlight our contributions, we mostly follow the architecture design of FOMM (Siarohin et al., 2019a). Similar to FOMM, our region predictor, background motion predictor and pixel-wise flow predictor operate on a quarter of the original resolution, e.g. 64×64 for 256×256 images, 96×96 for 384×384 and 128×128 for 512×512. We use the U-Net (Ronneberger et al., 2015) architecture with five "convolution - batch norm - ReLU - pooling" blocks in the encoder and five "upsample - convolution - batch norm - ReLU" blocks in the decoder for both the region predictor and the pixel-wise flow predictor. For the background motion predictor, we use only the five block encoder part. Similarly to FOMM (Siarohin et al., 2019a), we use the Johnson architecture (Johnson et al., 2016) for image generation, with two down-sampling blocks, six residual-blocks, and two up-sampling blocks. However, we add skip connections that are warped and weighted by the confidence map. Our method is trained using Adam (Kingma & Ba, 2014) optimizer with learning rate $2e-4$ and batch size 48, 20, 12 for 256×256, 384×384 and 512×512 resolutions respectively. During the training process, the networks observe 3M source-driving pairs, each pair selected at random from a random video chunk, and we drop the learning rate by a factor of 10 after 1.8M and 2.7M pairs. We use 4 Nvidia P100 GPUs for training.

C MGIF DATASET

We run additional experiments on the MGif (Siarohin et al., 2019b) dataset to further demonstrate the superiority of PCA-based representations over regression-based ones. The dataset contains a set of animations of articulated, 2D, cartoon animals. The qualitative results are presented in the supplementary video. We can observe that our PCA-based representation successfully tracks all legs, while the regression-based representation often misses some of the legs, which leads to worse reconstruction quality. This observation is further confirmed by quantitative evaluation; the $\mathcal{L}_1$ error for FOMM (Siarohin et al., 2019a) is 0.0223, while for our method it is 0.0206.

D COMPARISON WITH OTHER METHODS

The main paper has focused on comparing our method to FOMM (Siarohin et al., 2019a), as it is both most similar to our work, and the current state-of-the-art. We show quantitative results using prior works (Wiles et al., 2018; Siarohin et al., 2019b) in Table 5. These are significantly inferior to both FOMM and our method.

[3] This license allows for non-commercial use.

Table 5: Video reconstruction comparison. (Best result in bold.)

	TaiChiHD (256)			VoxCeleb		
	$\mathcal{L}_1$	(AKD, MKR)	AED	$\mathcal{L}_1$	AKD	AED
X2Face	0.080	(17.65, 0.109)	0.27	0.078	7.69	0.405
Monkey-Net	0.077	(10.80, 0.059)	0.228	0.049	1.89	0.199
FOMM	0.056	(6.53, 0.033)	0.172	0.041	**1.27**	0.134
Ours	**0.048**	(**5.45**, **0.028**)	**0.152**	**0.040**	1.28	**0.133**

E Toy Experiment Details

The rotated rectangles dataset consists of images of rectangles randomly rotated from 0° to 90°, along with labels that indicate the angle of rotation. The rectangles have different, random colors. Visual samples are shown in Fig. 6.

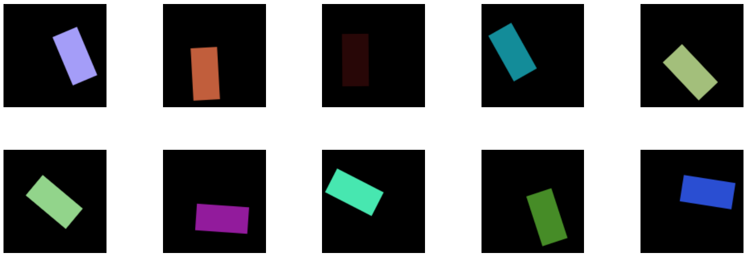

Figure 6: Examples of synthetic rectangle dataset.

We tested three different networks: Naive, Regression-based and PCA-based. The Naive network directly predicts an angle from an image using an encoder and a fully-connected layer. Regression-based is similar to FOMM (Siarohin et al., 2019a); the angle is regressed per pixel an using hourglass network, and pooled according to heatmap weights predicted using the same hourglass network. PCA-based is our method described in Sec. 3; we predict the heatmap using an hourglass network, PCA is performed according to eq. equation 2, and the angle is computed from matrix U as $\arctan(U_{10}/U_{00})$.

Each of the networks was trained, on subsets of the dataset of varying sizes, to minimize the $\mathcal{L}_1$ loss between predicted and ground truth rotation angle. All models were trained for 100 epochs, with batch size 8. We used the Adam optimizer, with a learning rate of 10^{-4}. We varied the size of the training set from 32 to 1024. Results, on a separate, fixed test set of size 128, were then computed, shown in Fig. 5.