DwNet: Dense warp-based network for pose-guided human video generation

Polina Zablotskaia[12]
pzablots@cs.ubc.ca
Aliaksandr Siarohin[3]
aliaksandr.siarohin@unitn.it
Bo Zhao[12]
bzhao03@cs.ubc.ca
Leonid Sigal[12]
lsigal@cs.ubc.ca

[1] Department of Computer Science
University of British Columbia
Vancouver, BC, Canada

[2] Vector Institute for Artificial Intelligence
Toronto, ON, Canada

[3] DISI
University of Trento
Trento, Italy

Abstract

Generation of realistic high-resolution videos of human subjects is a challenging and important task in computer vision. In this paper, we focus on human motion transfer – generation of a video depicting a particular subject, observed in a single image, performing a series of motions exemplified by an auxiliary (driving) video. Our GAN-based architecture, DwNet, leverages dense intermediate pose-guided representation and refinement process to warp the required subject appearance, in the form of the texture, from a source image into a desired pose. Temporal consistency is maintained by further conditioning the decoding process within a GAN on the previously generated frame. In this way a video is generated in an iterative and recurrent fashion. We illustrate the efficacy of our approach by showing state-of-the-art quantitative and qualitative performance on two benchmark datasets: TaiChi and Fashion Modeling. The latter is collected by us and will be made publicly available to the community.

1 Introduction

Generative models, both conditional and un-conditional, have been at the core of computer vision field from its inception. In recent years, approaches such as GANs [9] and VAEs [17] have achieved impressive results in a variety of image-based generative tasks. The progress on the video side, on the other hand, has been much more timid. Of particular challenge is generation of videos containing high-resolution moving human subjects. In addition to the need to ensure that each frame is realistic and video is overall temporally coherent, additional challenge is contending with coherent appearance and motion realism of a human subject itself. Notably, visual artifacts exhibited on human subjects tend to be most glaring for observers (an effect partially termed "uncanny valley" in computer graphics).

In this paper, we address a problem of human motion transfer. Mainly, given a single image depicting a (source) human subject, we propose a method to generate a high-resolution video of this subject, conditioned on the (driving) motion expressed in an auxiliary video.

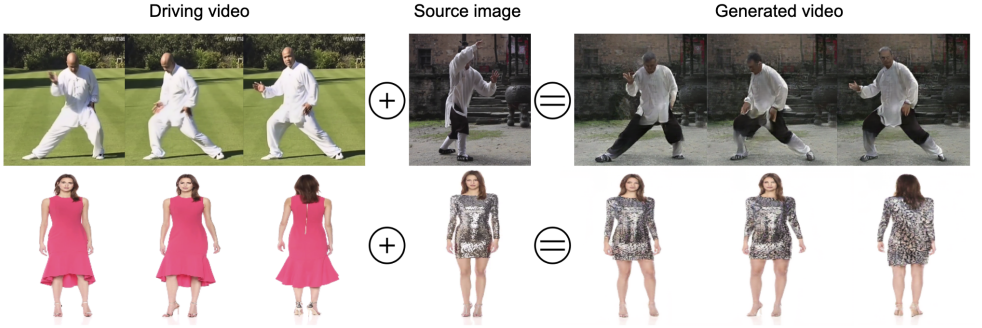


Figure 1: **Human Motion Transfer.** Our pose-guided approach allows generation of a video for the source subject, depicted in an image, that follows the motion in the driving video.

The task is illustrated in Figure 1. Similar to recent methods that focus on pose-guided image generation [2, 6, 7, 10, 19, 20, 22, 27, 28, 38], we leverage an intermediate pose-centric representation of the subject. However, unlike those methods that tend to focus on sparse keypoint [2, 6, 19, 20, 27] or skeleton [7] representations, or intermediate dense optical flow obtained from those impoverished sources [28], we utilize a more detailed dense intermediate representation [10] and texture transfer approach to define a fine-grained warping from the (source) human subject image to the target poses. This texture warping allows us to more explicitly preserve the appearance of the subject. Further, we focus on temporal consistency which ensures that the transfer is not done independently for each generated frame, but is rather sequentially conditioned on previously generated frames. We also note that unlike [5, 35], we rely only on a *single* (source) image of the subject and not a video, making the problem that much more challenging.

Contributions: Our contributions are multiple fold. First, we propose a dense warp-based architecture designed to account for, and correct, dense pose errors produced by our intermediate human representation. Second, we formulate our generative framework as a conditional model, where each frame is generated conditioned not only on the source image and the target pose, but also on previous frame generated by the model. This enables our framework to produce a much more temporally coherent output. Third, we illustrate the efficacy of our approach by showing improved performance with respect to recent state-of-the-art methods. Finally, we collect and make available a new high-resolution dataset of fashion videos.

2 Related work

Image Generation. Image generation has become an increasingly popular task in the recent years. The goal is to generate realistic images, mimicking samples from true visual data distribution. Variational Autoencoders (VAEs) [17] and Generative Adversarial Networks (GANs) [9] are powerful tools for image generation that have shown promising results. In the case of the unconstrained image generation, resulting images are synthesized from random noise vectors. However, this paradigm can be extended to the *conditional* image generation [14, 36], where apart from the noise vector the network input includes conditional information, which can be, for example, a style image [14, 16], a descriptive sentence [13] or an object layout [39] designating aspects of desired image output. Multi-view synthe-

sis [24, 31, 41] is one of the largest topics in the conditional generation and it is the the one that mostly related to our proposal. The task of multi-view synthesis is to generate unseen view given one or more known views.

Pose guided image generation. In the pioneering work of Ma *et al*. [19] pose guided image generation using GANs has been proposed. Ma *et al*. [19] suggest to model human poses as a set of keypoints and use standard image-to-image translation networks (*e.g*., UNET [25]). Later, it has been found [2, 27] that for UNET-based architectures it is difficult to process inputs that are not spatially aligned. In case of pose-guided models, keypoints of target are not spatially aligned with the source image. As a consequence, [2, 27] propose new generator architectures that try to first spatially align these two inputs and then generate target images using image-to-image translation paradigms. Neverova *et al*. [22] suggest to exploit SMPL [18] representation of a person, which they estimate using DensePose [11], in order to improve pose-guided generation. Compared to keypoint representations, DensePose [11] results provide much more information about the human pose, thus using it as a condition allows much better generation results. Grigorev *et al*. [10] propose coordinate based inpainting to recover missing parts in the DensePose [11] estimation. Coordinate based inpainting explicitly predicts from where to copy the missing texture, while regular inpainting predicts the RGB values of the missing texture itself. Contrary to Grigorev *et al*. [10], our work can perform both standard inpainting and coordinate based inpainting.

Video Generation. The field of video generation is much less explored, compared to the image generation. Initial works adopt 3D convolutions and recurrent architectures for video generation [26, 32, 34]. Later works start to focus on conditional video generation. The most well studied task, in conditional video generation, is future frame prediction [1, 8, 23, 30]. Recent works exploit intermediate representation, in the form of learned keypoints, for future frame prediction [33, 37, 40]. However, the most realistic video results are obtained by conditioning video generation on another video. This task is often called video-to-video translation. Two recent works [5, 35] suggest pose-guided video generation, which is a special case of video-to-video translation. The main drawback of these models is that they need to train a separate network for each person. In contrast, we suggest to generate a video based only on a *single* image of a person. Recently, this task was addressed by Siarohin *et al*. [28], but they try to learn a representation of a subject in an unsupervised manner which leads to sub-optimal results. Conversely, in our work, we exploit and refine the richer structure and representation from DensePose [11] as an intermediate guide for video generation.

3 Method

The objective of this work is to generate a video, consisting of N frames $[\mathbf{t}_1, \dots, \mathbf{t}_N]$, containing a person from a *source* image $\mathbf{s}$, conditioned on a sequence of driving video frames of another person $[\mathbf{d}_0, \dots, \mathbf{d}_N]$, such that the person from the source image replicates motions of the person from a driving video. Our model is based on standard image-to-image [14] translation framework. However, the standard convolutional networks are not well suited for the task where the condition and the result are not well aligned. Hence, as advocated in the recent works [2, 27], the key to a precise human subject video generation lies in leveraging of motion from the estimated poses. Moreover, perceptual quality of the video is highly dependent on the temporal consistency between nearby frames. We design our model having these goals and intuitions in mind (see Figure 2(a)).

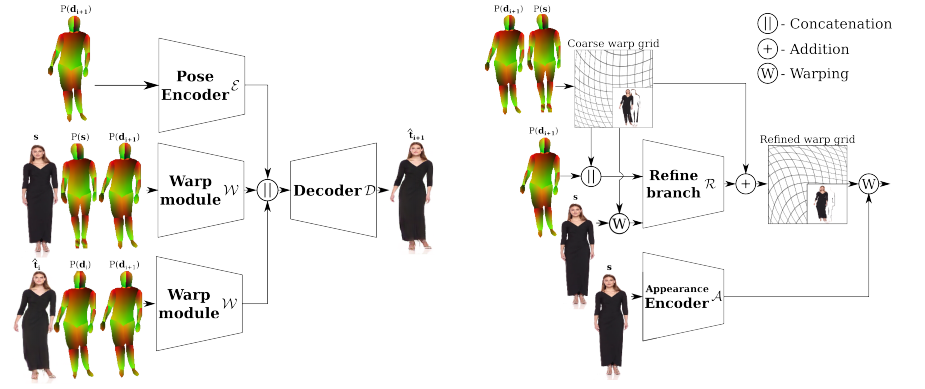


Figure 2: **Proposed Architecture.** Overview of the entire model (a). The model consists of three main parts: Pose Encoder ($\mathcal{E}$), Warp Module ($\mathcal{W}$) and Decoder ($\mathcal{D}$). The encoder $\mathcal{E}$ takes as input the next driving video pose $P(\mathbf{d}_{i+1})$ and produces its feature representation. On the other hand source image $\mathbf{s}$, source image pose $P(\mathbf{s})$ and the next frame's pose $P(\mathbf{d}_{i+1})$ are used by $\mathcal{W}$ to produce deformed feature representation of the source image $\mathbf{s}$. Similarly, $\mathcal{W}$ produces a deformed feature representation of previously generated frame $\hat{\mathbf{t}}_i$. Decoder $\mathcal{D}$ combines these representations and generates a new frame $\hat{\mathbf{t}}_{i+1}$. Detailed overview of the warp module $\mathcal{W}$ is illustrated in (b). In the beginning, we compute a coarse warp grid from $\mathcal{D}_{i+1}$ and $P(\mathbf{s})$. The coarse warp grid is used to deform the source image $\mathbf{s}$, this deformed image along with driving pose $P(\mathbf{d}_{i+1})$ and coarse warp coordinates is used by Refine branch $\mathcal{R}$ to predict correction to coarse warp grid. We employ this refined estimate to deform feature representation of the source image $\mathbf{s}$. Note, deformation of $\hat{\mathbf{t}}_i$ takes similar form.

First, differently from standard pose-guided image generation frameworks [2, 19, 27], in order to produce temporary consistent videos we add a markovian assumptions to our model. If we generate each frame of the video independently, the result can be temporary inconsistent and have a lot of flickering artifacts. To this end, we condition generation of each frame $\mathbf{t}_i$ on a previously generated frame $\mathbf{t}_{i-1}$, where $i \in [2,\dots,N]$.

Second, we use a DensePose [11] architecture to estimate correspondences between pixels and parts of the human body, in 3D. We apply DensePose [11] to the initial image $P(\mathbf{s})$ and to every frame of a driving video $[P(\mathbf{d}_0),\dots,P(\mathbf{d}_N)]$, where function $P(\cdot)$ denotes the output of the DensePose. Using this information we obtain a partial correspondence between pixels of any two human images. This correspondence allows us to analytically compute the coarse warp grid estimate $W\left[P(\mathbf{s}) \rightarrow P(\mathbf{d}_i)\right]$ and $W\left[P(\mathbf{d}_i) \rightarrow P(\mathbf{d}_{i+1})\right]$, where $i \in [1,\dots,N-1]$. The coarse warp grid, in turn, allows us to perform texture transfer and estimate motion flow. Even though DensePose produces high quality estimates, it is not perfect and sometimes suffers from artifacts, such as false human detections and missing body parts. Another important drawback of using pose estimators is lack of information regarding clothing. This information is very important to us, since we are trying to map a given person onto a video, while preserving their body shape, facial features, hair and clothing. This motivate us to compute refined warp grid estimates $\overline{W}\left[P(\mathbf{s}) \rightarrow P(\mathbf{d}_i)\right]$ and $\overline{W}\left[P(\mathbf{d}_i) \rightarrow P(\mathbf{d}_{i+1})\right]$, where $i \in [0,\dots,N-1]$ (see Figure 2(b)). We train this component end-to-end using standard image generation losses.

To sum up, our generator $\mathcal{G}$ consists of three blocks: pose encoder $\mathcal{E}$, warp module $\mathcal{W}$ and the decoder $\mathcal{D}$ (see Figure 2(a)). $\mathcal{G}$ generates video frames iteratively one-by-one. First, the encoder $\mathcal{E}$ produces a representation of the driving pose $\mathcal{E}(P(\mathbf{d}_{i+1}))$. Then given the source image $\mathbf{s}$, the source pose $P(\mathbf{s})$ and the driving pose $P(\mathbf{d}_i)$, warp module $\mathcal{W}$ estimates a grid $\overline{W}[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})]$, encodes the source image $\mathbf{s}$ and warps this encoded image according to the grid. This gives us a representation $\mathcal{W}(\mathbf{s}, P(\mathbf{s}), P(\mathbf{d}_{i+1}))$. Previous frame is processed in the same way, *i.e.*, after transformation we obtain a representation $\mathcal{W}(\mathbf{d}_i, P(\mathbf{d}_i), P(\mathbf{d}_{i+1}))$. These two representations together with $\mathcal{E}(P(\mathbf{d}_{i+1}))$ are concatenated and later processed by the decoder $\mathcal{D}$ to produce an output frame $\hat{\mathbf{t}}_{i+1}$.

3.1 Warp module

Coarse warp grid estimate. As described in Section 3 we use DensePose [11] for coarse estimation of warp grids. For simplicity, we describe only how to obtain warp grid estimates $W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})]$; the procedure for $W[P(\mathbf{d}_i) \to P(\mathbf{d}_{i+1})]$ is similar. For each body part of SMPL model [18] DensePose [11] estimates UV coordinates. However, the UV pixel coordinates in the source image $\mathbf{s}$ may not exactly match with the UV pixel coordinates in the driving frame $\mathbf{d}_{i+1}$, so in order to obtain a correspondence we use nearest neighbour interpolation. In more detail, for each pixel in the driving frame we find nearest neighbour in the UV space of source image that belongs to the same body part. In order to perform efficient nearest neighbour search we make use of the KD-Trees [3].

Refined warp grid estimate. While the coarse warp grid estimation preserves general human body movement, it contains a lot of errors because of self occlusions and imprecise DensePose [11] estimates. Moreover the movement of the person outfit is not modeled. This motivates us to add an additional correction branch $\mathcal{R}$. The goal of this branch is, given source image $\mathbf{s}$, coarse warp grid $W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})]$ and target pose $P(\mathbf{d}_{i+1})$, to predict refined warp grid $\overline{W}[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})]$. This refinement branch is trained end-to-end with the entire framework. Differentiable warping is implemented using bilinear kernel [15]. The main problem of the bilinear kernel is limited gradient flow, *e.g.*, from each spatial location gradients are propagated only to 4 nearby spatial locations. This makes the module highly vulnerable to the local minimums. One way to address the local minimum problem is good initialization. Having this in mind, we adopt residual architecture for our module, *i.e.*, the refinement branch predicts only the correction $\mathcal{R}(\mathbf{s}, W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})], P(\mathbf{d}_{i+1}))$ which is latter added to the coarse warp grid:

$$\overline{W}[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})] = W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})] + \mathcal{R}(\mathbf{s}, W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})], P(\mathbf{d}_{i+1})). \quad (1)$$

Note that since we transform intermediate representations of source image $\mathbf{s}$, the spatial size of the warp grid should be the equal to the spatial size of the representation. In our case the spatial size of the representation is 64×64. Because of this, and to save computational resources, $\mathcal{R}$ predicts corrections of size 64×64; moreover, coarse warp grid $W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})]$ is downsampled to the size of 64×64. Also, since convolutional layers are translation equivariant they can not efficiently process absolute coordinates of coarse grid warp $W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})]$. In order to alleviate this issue we input to the network relative shifts, *i.e.*, $W[P(\mathbf{s}) \to P(\mathbf{d}_{i+1})] - I$, where I is an identity warp grid.

3.2 Training

Our training procedure is similar to [2], however, specifically adopted to take markovian assumption into account. At the training time we sample four frames from the same training video (three of which are consecutive), the indices of these frames are $i, j, j+1, j+2$, where $i \in [1,\dots,N]$, $j \in [1,\dots,N-2]$ and N is the total number of frames in the video. Experimentally we observe that using four frames is the best in terms of temporal consistency and computational efficiency. We treat frame i as the source image $\mathbf{s}$, while the rest are treated as both the driving $\mathbf{d}_i, \mathbf{d}_{i+1}, \mathbf{d}_{i+2}$ and the ground truth target $\mathbf{t}_i, \mathbf{t}_{i+1}, \mathbf{t}_{i+2}$ frames. We generate the three frames as follows:

$$\hat{\mathbf{t}}_i = \mathcal{G}\left(\mathbf{s}, P(\mathbf{s}), \mathbf{s}, P(\mathbf{s}), P(\mathbf{d}_i)\right), \tag{2}$$

for the first frame, where the source frame is treated as the "previous" frame, and for the rest:

$$\begin{aligned} \hat{\mathbf{t}}_{i+1} &= \mathcal{G}\left(\mathbf{s}, P(\mathbf{s}), \hat{\mathbf{t}}_i, P(\mathbf{d}_i), P(\mathbf{d}_{i+1})\right), \\ \hat{\mathbf{t}}_{i+2} &= \mathcal{G}\left(\mathbf{s}, P(\mathbf{s}), \hat{\mathbf{t}}_{i+1}, P(\mathbf{d}_{i+1}), P(\mathbf{d}_{i+2})\right). \end{aligned} \tag{3}$$

This formulation has low memory consumption, but at the same time allows standard pose-guided image generation which is needed to produce the first target output frame. Note that if in Eq. 3 we use real previous frame $\mathbf{d}_i$ the generator will ignore the source image $\mathbf{s}$, since $\mathbf{d}_i$ is much more similar to $\mathbf{d}_{i+1}$ than $\mathbf{s}$.

Losses. We use a combination of losses from pix2pixHD [36] model. We employ the least-square GAN [21] for the adversarial loss:

$$\mathcal{L}^{\mathcal{D}}_{\text{gan}}(\mathbf{t}_i, \hat{\mathbf{t}}_i) = (\mathcal{C}(\mathbf{t}_i) - 1)^2 + \mathcal{C}(\hat{\mathbf{t}}_i)^2, \tag{4}$$

$$\mathcal{L}^{\mathcal{G}}_{\text{gan}}(\hat{\mathbf{t}}_i) = (\mathcal{C}(\hat{\mathbf{t}}_i) - 1)^2, \tag{5}$$

where $\mathcal{C}$ is the patch bases critique [14, 36],

To drive image reconstruction we also employ a feature matching [36] and perceptual [16] losses:

$$\mathcal{L}_{\text{rec}}(\mathbf{t}_i, \hat{\mathbf{t}}_i) = \|\mathcal{N}_k(\hat{\mathbf{t}}_i) - \mathcal{N}_k(\mathbf{t}_i)\|_1, \tag{6}$$

where $\mathcal{N}_k$ is feature representation from k-th layer of the network, for perceptual loss this is VGG-19 [29] and for the feature matching it is the critique $\mathcal{C}$. The total loss is given by:

$$\mathcal{L} = \sum_i \mathcal{L}^{\mathcal{G}}_{\text{gan}}(\hat{\mathbf{t}}_i) + \lambda \mathcal{L}_{\text{rec}}(\mathbf{t}_i, \hat{\mathbf{t}}_i), \tag{7}$$

following [36] $\lambda = 10$.

4 Experiments

We have conducted an extensive set of experiments to evaluate the proposed DwNet. We first describe our newly collected dataset, then we compare our method with previous state-of-the-art models for pose-guided human video synthesis and for pose-guided image generation. We show superiority of our model in aspects of realism and temporal coherency. Finally, we evaluate the contributions of each architecture choice we made and show that each part of the model positively contributes to the results.

4.1 The Fashion Dataset

We introduce a new Fashion dataset containing 500 training and 100 test videos, each containing roughly 350 frames. Videos from our dataset are of a single human subject and characterized by the high resolution and static camera. Most importantly, clothing and textures are diverse and cover large space of possible appearances. The dataset is publicly released at: https://vision.cs.ubc.ca/datasets/fashion/.

4.2 Setup

4.2.1 Datasets

We conduct our experiments on the proposed Fashion and Tai-Chi [32] datasets. The latter is composed of more than 3000 tai-chi video clips downloaded from YouTube. In all previous works [28, 32], the smaller 64×64 pixel resolution version of this dataset has been used; however, for our work we use 256×256 pixel resolution. The length varies from 128 to 1024 frames per video. Number of videos per train and test sets are 3049 and 285 respectively.

4.2.2 Evaluation metrics

There is no consensus in the community on a single criterion for measuring quality of the generated videos from the perspective of realism, texture similarity and temporal coherence. Therefore we choose a set widely accepted evaluation metrics to measure performance.

Perceptual loss. The texture similarity is measured using a perceptual loss. Similar to our training procedure, we use VGG-19 [29] network as a feature extractor and then compute L_1 loss between the extracted features from the real and generated frames.

FID. We use Frecht Inception Distance [17] (FID) to measure realism of the individual frames. FID is known to be a widely used metric for comparison of the GAN-based methods.

AKD. We evaluate if the motion is correctly transferred by the means of Average Keypoint Distance (AKD) metric [28]. Similarly to [28] we employ human pose estimator [4] and compare average distance between ground truth and detected keypoints. Intuitively this metric measures if the person moves in the same way as in the driving video. We do not report Missing keypoint rate (MKR) because it similar and close to zero for all methods.

User study. We conduct a user study to measure the overall temporal coherency and quality of the synthesised videos. For the user study we exploit Amazon Mechanical Turk (AMT). On AMT we show users two videos (one produced by DwNet and another by a competing method) in random order and ask users to choose one, which has higher realism and consistency. To conduct this study we follow the protocol introduced in Siarohin *et al.* [28].

4.3 Implementation details

All of our models are trained for two epochs. In our case epoch denotes a full pass through the whole set of video frames, where each sample from the dataset is a set of four frames, as explained in the Section 3.2. We train our model starting with the learning rate 0.0002 and bring it down to zero during the training procedure. Generally, our model is similar to Johnson *et al.* [16]. Novelties of the architecture such as pose encoder $\mathcal{E}$ and the appearance

Figure 3: **Qualitative Results on Tai-Chi Dataset.** First row illustrates the driving videos; second row are results of our method; third row are results obtained with Monkey-Net [28].

encoder $\mathcal{A}$ both contain 2 downsampling Conv layers. Warp module's refine branch $\mathcal{R}$ is also based on 2 Conv layers and additional 2 ResNet blocks. Our decoder $\mathcal{D}$ architecture is made out of 9 ResNet blocks and 2 upsampling Conv layers. We perform all our training procedures on a single GPU (Nvidia GeForce GTX 1080). Our code will be released.

4.4 Comparison with the state-of-the-art

We compare our framework with the current state-of-the-art method for motion transfer MonkeyNet [28], which solves a similar problem for the human synthesis. The first main advantage of our method, compared to MonkeyNet, is ability to generate frames with a higher resolution. Originally, MonkeyNet was trained on 64×64 size frames. However, to conduct fair experiments we re-train MonkeyNet from scratch to produce the same size images with our method. Our task is quite novel and there is limited number of baselines. To this end, we also compare with Coordinate Inpainting framework [10] which is state-of-the-art for image (not video) synthesis, *i.e.*, synthesise of a new image of a person based on a single image. Even though this framework solves a slightly different problem, we still choose to compare with it, since it is similarly leverages DensePose [11]. This approach doesn't have any explicit background handling mechanisms therefore there is no experimental results on a Tai-Chi dataset. Note that since authors of the paper haven't released the code for the method we were only able to run our experiments on a pre-trained network.

The quantitative comparison is reported in Table 1. Our approach outperforms MonkeyNet and Coordinate Inpainting on both datasets and according to all metrics. With respect to MonkeyNet, this can be explained by its inability to grasp complex human poses, hence it completely fails on the Tai-Chi dataset which contains large variety of non-trivial motions. This can be further observed in Figure 3. MonkeyNet simply remaps person from the source image without modifying the pose. In Figure 4 we can still observe a large difference in terms of the human motion consistency and realism. Unlike our model, MonkeyNet produces images with missing body parts. For Coordinate Inpainting, poor performance could be explained by the lack of temporal consistency, since (unlike our method) it generates each frame independently and hence lacks consistency in clothing texture and appearance. Coordinate Inpainting is heavily based on the output of the DensePose and doesn't correct resulting artifacts, like is done in our model using refined warp grid estimate. As one can see from Figure 4 the resulting frames are blurry and inconsistent in small details. This can also explains why such a small percentage of users prefer results of Coordinate Inpainting. The user study comparison is reported in Table 2 where we can observe that videos produced by

	Fashion			Tai-Chi		
	Perceptual (↓)	FID (↓)	AKD (↓)	Perceptual (↓)	FID (↓)	AKD (↓)
Monkey-Net	0.3726	19.74	2.47	0.6432	94.97	10.4
Coordinate Inpainting	0.6434	66.50	4.20	-	-	-
Ours	**0.2811**	**13.09**	**1.36**	**0.5960**	**75.44**	**3.77**

Table 1: **Quantitative Comparison with the State-of-the-Art.** Performance on Fashion and Tai-chi datasets is reported in terms of the Perceptual Loss, Frechet Inception Distance (FID) and Average Keypoint Distance (AKD)

our model were significantly more often preferred by users in comparison to the videos from the competitors models.

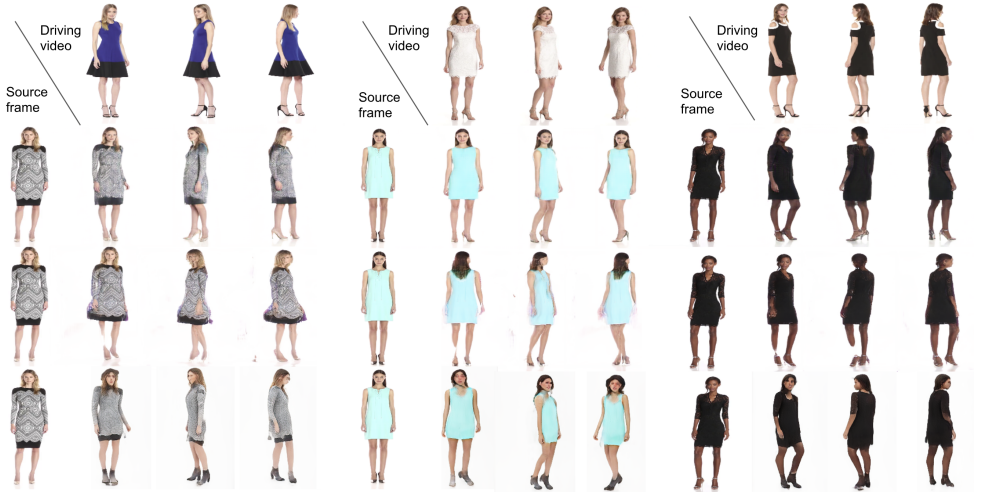


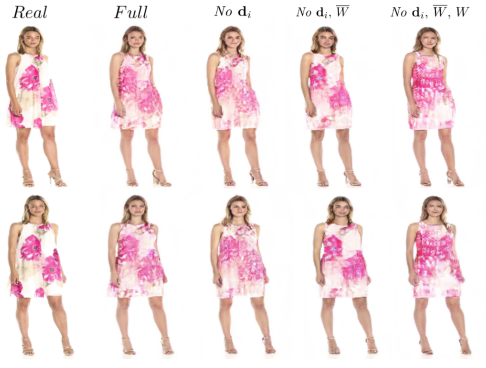
Figure 4: **Qualitative Results on Fashion Dataset.** First row illustrates the driving videos; second row are results of our method; third row are results obtained with Monkey-Net [28]; forth row are results of Coordinate Inpainting [10]. Please zoom in for detail.

4.5 Ablation

Table in Figure 5 (right) shows the contribution of each of the major architecture choices, *i.e.*, markovian assumption (*No* $\mathbf{d}_i$), refined warp grid estimate (*No* $\mathbf{d}_i$, $\overline{W}$) and coarse warp grid estimate (*No* $\mathbf{d}_i$, $\overline{W}$, W). For these experiments we remove mentioned parts from DwNet and train the resulting model architectures. As expected, removing markovian assumption, *i.e.*, not conditioning on the previous frame, leads to a worse realism and lower similarity with the features of a real image. Mainly it is because this leads to a loss of temporal coherence. Further removal of both warping grid estimators, in the generation pipeline, results in worse performance in the FID score. Perceptual loss is not affected by this change which can be explained by the fact that warp mostly results in removal of the artifacts and naturalness of the texture on a person. In Figure 5 (left) we see the qualitative reflection of our quantitative results. Full model produces the best results, third column shows misalignment between

	Fashion User-Preference (↑)	Tai-Chi User-Preference (↑)
Monkey-Net	60.40%	85.20%
Coordinate Inpainting	99.60%	-

Table 2: **User study.** Percentage of the time our method is preferred over one of the competing approaches.



	Fashion Perceptual (↓)	Fashion FID (↓)
No $\mathbf{d}_i, \overline{W}, W$	0.29	17.18
No $\mathbf{d}_i, \overline{W}$	0.29	15.37
No $\mathbf{d}_i$	0.29	15.05
Full	**0.28**	**13.09**

Figure 5: **Ablations on Fashion Dataset.** On the (left) are qualitative result from the ablated methods; on the (right) are corresponding quantitative evaluations. See text for details.

textures of two frames. The architecture without a refined warp produces less realistic results, with a distorted face. Lastly, an architecture without any warp produces blurry, unrealistic results with an inconsistent texture.

5 Conclusion

In this paper we present DwNet a generative architecture for pose-guided video generation. Our model can produce high quality videos, based on the source image depicting human appearance and the driving video with another person moving. We propose novel markovian modeling to address temporal inconsistency, that typically arises in video generation frameworks. Moreover we suggest novel warp module that is able to correct warping errors. We validate our method on two video datasets, and we show superiority of our method over the baselines. Some possible future directions may include multiple source generation and exploiting our warp correction for improving DensePose [11] estimation.

References

[1] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H. Campbell, and Sergey Levine. Stochastic variational video prediction. In *ICLR*, 2017.

[2] Guha Balakrishnan, Amy Zhao, Adrian V. Dalca, Fredo Durand, and John Guttag. Synthesizing images of humans in unseen poses. In *CVPR*, 2018.

[3] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

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

[5] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *ECCV*, 2018.

[6] Haoye Dong, Xiaodan Liang, Ke Gong, Hanjiang Lai, Jia Zhu, and Jian Yin. Soft-gated warping-gan for pose-guided person image synthesis. In *NIPS*, 2018.

[7] Patrick Esser, Ekaterina Sutter, and Björn Ommer. A variational u-net for conditional appearance and shape generation. In *CVPR*, 2018.

[8] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *NIPS*, 2016.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[10] Artur Grigorev, Artem Sevastopolsky, Alexander Vakhitov, and Victor Lempitsky. Coordinate-based texture inpainting for pose-guided image generation. In *CVPR*, 2019.

[11] Riza Alp Guler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018.

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017.

[13] Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. Inferring semantic layout for hierarchical text-to-image synthesis. In *CVPR*, 2018.

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *NIPS*, 2015.

[16] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[17] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[18] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

[19] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. In *NIPS*, 2017.

[20] Liqian Ma, Qianru Sun, Stamatios Georgoulis, Luc Van Gool, Bernt Schiele, and Mario Fritz. Disentangled person image generation. In *CVPR*, 2018.

[21] Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *ICCV*, 2017.

[22] Natalia Neverova, Riza Alp Guler, and Iasonas Kokkinos. Dense pose transfer. In *ECCV*, 2018.

[23] Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. In *NIPS*, 2015.

[24] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C Berg. Transformation-grounded image generation network for novel 3d view synthesis. In *CVPR*, 2017.

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

[26] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017.

[27] Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuilière, and Nicu Sebe. Deformable gans for pose-based human image generation. In *CVPR*, 2018.

[28] Aliaksandr Siarohin, Stéphane Lathuilière, Segey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *CVPR*, 2019.

[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[30] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *ICML*, 2015.

[31] Shao-Hua Sun, Minyoung Huh, Yuan-Hong Liao, Ning Zhang, and Joseph J Lim. Multi-view to novel view: Synthesizing novel views with self-learned confidence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[32] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *CVPR*, 2018.

[33] Ruben Villegas, Jimei Yang, Yuliang Zou, Sungryull Sohn, Xunyu Lin, and Honglak Lee. Learning to generate long-term future via hierarchical prediction. In *ICML*, 2017.

[34] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NIPS*, 2016.

[35] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *NIPS*, 2018.

[36] 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 *CVPR*, 2018.

[37] Wei Wang, Xavier Alameda-Pineda, Dan Xu, Pascal Fua, Elisa Ricci, and Nicu Sebe. Every smile is unique: Landmark-guided diverse smile generation. In *CVPR*, 2018.

[38] Mihai Zanfir, Alin-Ionut Popa, Andrei Zanfir, and Cristian Sminchisescu. Human appearance transfer. In *CVPR*, 2018.

[39] Bo Zhao, Lili Meng, Weidong Yin, and Leonid Sigal. Image generation from layout. In *CVPR*, 2019.

[40] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris Metaxas. Learning to forecast and refine residual motion for image-to-video generation. In *ECCV*, 2018.

[41] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *ECCV*, 2016.