Deep Video Generation, Prediction and Completion of Human Action Sequences

Haoye Cai[1,3][0000−0001−7041−563X]⋆, Chunyan Bai[1,4][0000−0001−5431−795X]⋆,
Yu-Wing Tai[2][0000−0002−3148−0380], and Chi-Keung Tang[1][0000−0001−6495−3685]

[1] Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
{hcaiaa,cbai}@connect.ust.hk cktang@cs.ust.hk
[2] Tencent Youtu, Shenzhen, China
yuwingtai@tencent.com
[3] Stanford University, Stanford, CA 94305, USA
[4] Carnegie Mellon University, Pittsburgh, PA 15213, USA
Project page: https://iamacewhite.github.io/supp

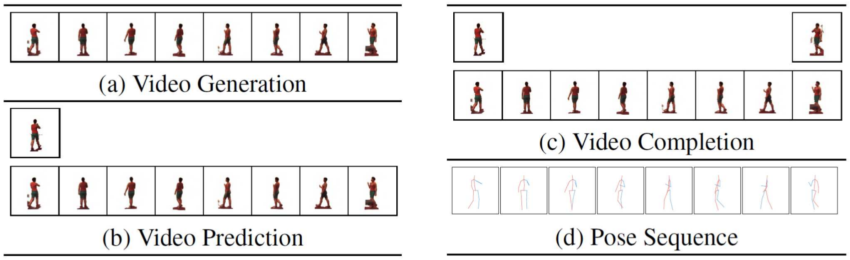
Abstract. Current video generation/prediction/completion results are limited, due to the severe ill-posedness inherent in these three problems. In this paper, we focus on human action videos, and propose a general, two-stage deep framework to generate human action videos with no constraints or arbitrary number of constraints, which uniformly addresses the three problems: video generation given no input frames, video prediction given the first few frames, and video completion given the first and last frames. To solve video generation from scratch, we build a two-stage framework where we first train a deep generative model that generates human pose sequences from random noise, and then train a skeleton-to-image network to synthesize human action videos given the human pose sequences generated. To solve video prediction and completion, we exploit our trained model and conduct optimization over the latent space to generate videos that best suit the given input frame constraints. With our novel method, we sidestep the original ill-posed problems and produce for the first time high-quality video generation/prediction/completion results of much longer duration. We present quantitative and qualitative evaluations to show that our approach outperforms state-of-the-art methods in all three tasks.

Keywords: Video Generation · Generative Models

1 Introduction

In this paper we propose a general, two-stage deep framework for human video generation (i.e. generating video clips directly from latent vectors), prediction (i.e. predicting future frames of a short clip or single frame), and completion (i.e. completing the intermediate content given the beginning and the ending),

⋆ Equal Contribution.



(a) Video Generation

(b) Video Prediction

(c) Video Completion

(d) Pose Sequence

Fig. 1. (a) Video generation, (b) prediction and (c) completion of human action videos using our general two-stage deep framework. (d) In all cases, a complete human pose skeleton sequence is generated in the first stage

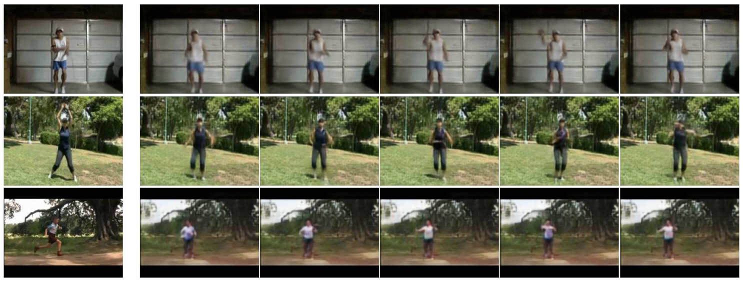


Fig. 2. Real-world examples. We use reference images in the first column (arbitrary unrelated actions) to generate *Direction/Greeting* actions. 1st and 2nd row: UCF-101 results. 3rd row: Forrest Gump results. See full videos in supplemental material

where each problem was previously addressed as separate problems (Fig. 1). Previous video generation capitalizing state-of-the-art deep convolutional neural network (CNN), such as [35], has demonstrated the significant difficulty of the problem, where their first results were still far from photorealism. Current future prediction [19] in the form of video prediction [37, 34] generates a short video from a given frame to predict future actions in a very limited scope with blurry results. Lastly, while there exist deep learning works on image completion [45], there is no known representative deep learning work on video completion.

To better address the general video synthesis problem, we need to understand how pixels change to generate a full temporal object action. With a higher level of uncertainty in the exact movement between frames of the moving object on pixel level, as observed in [37, 34], the problem is more tractable by modeling the uncertainty with underlying structure of the moving objects. Hence, we utilize this idea and conduct our experiments on human action videos, which is a well-studied and useful class of videos in various computer vision applications, and in this case the natural choice of underlying structure is human poses (or skeletons). Thus we divide the video generation task into human pose sequence generation (pose space) followed by image generation (pixel space) from the generated human pose sequences. Then, for the prediction and completion problems, we can solve them using the same model by regarding them as constrained generation.

Specifically, our general deep framework for video generation has two stages: first, a new conditional generative adversarial network (GAN) to generate plausible pose sequences that perform a given category of actions; second, a supervised reconstruction network with feature matching loss to transfer pose sequences to the pixel space. Our general video generation framework can be specialized to video prediction/completion (i.e. constrained generation) by optimizing in the latent space to generate video results closest to the given input constraints. Hence our approach can either generate videos from scratch, or complete/predict a video with arbitrary number of input frames available given the action class. We provide extensive qualitative and quantitative experimental results to demonstrate that our model is able to generate and complete natural human motion videos. We also test our model on real-world videos (Fig. 2).

2 Related Work

We review here recent representative state-of-the-art works related to this paper.
Video Prediction/Generation In video prediction, research has been done to model uncertain human motion in pose space [37, 34]. Attempts have also been made to learn the deep feature representation [19, 36, 16, 43]. For video generation, work has been done to generate videos directly in pixel space [35, 18] or generate from caption [18]. While these works shed light on how to model the uncertain temporal information in videos, the results are suboptimal. Our proposed method achieves higher quality, and more importantly, aims at a higher goal: video completion, prediction and generation in the same framework.
Image/Video Completion Much work has been focusing on image completion with Generative Models [45], but video completion with deep learning has remain unexplored despite its importance [13, 42]. If the temporal distance to be completed is small, e.g., [23] then video frame interpolation can be performed to fill in the in-between frames. However, we are dealing with a different problem where input frames are far apart from each other. The modeling of such uncertainty increases the difficulty of this task. In our paper, we aim to perform video completion by optimizing the latent space under the constraint of input frames.
Human Pose Estimation Various research efforts [4, 33, 41, 6, 22] have been made to produce state-of-the-art human pose estimation results, providing us with reliable human pose extractor. In our paper, we leverage the reliable human pose estimation results by [22, 4] as input to our completion pipeline.
Generative Models Our work is based on Generative Adversarial Networks (GAN). Goodfellow et al [10] first proposed GAN that can implicitly generate any probabilistic distribution. Then conditional GAN [20] was proposed to enable generation under constraint. Subsequent works include usage of convolution neural networks [26], improvement of training stability [28] followed by WGAN [1] and Improved WGAN [11] which further made GAN reliable. In our

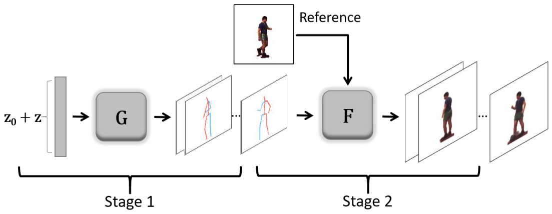


Fig. 3. Overview of our two-stage video generation. In the first stage we generate skeleton motion sequences by G from random noise, while in the second stage we use our skeleton-to-image transformer F to transform skeleton sequence to image space

paper, we first train a conditional WGAN to generate single frame human pose, then we train a conditional sequence GAN to generate latent vector sequences for the single frame model to output human action sequences.

Optimization over Input Data To specialize to video prediction and completion, we model them as constrained video generation and update input latent vector to find the motion sequence that best matches the input frames. Recently, back-propagation on input data is performed on image inpainting [45] to generate the best match of corrupted image. Zhu et al [48] utilized such method to enable generative visual manipulation. Google DeepDream [21] also used optimization on latent manifold to generate dream-like images. Earlier, similar method has been employed to perform texture synthesis and style transfer [8, 9, 15].

Skeleton to Image Transformation Our two-stage model involves a second stage that transforms human poses to pixel level images, which has been attempted by various deep learning methods. Recent works [44, 17, 37, 34, 46] utilize GAN or multi-stage method to complete this task. We propose a simple yet effective supervised learning framework comparable to state-of-the-arts.

3 Methodology

We present a general generative model that uniformly addresses video generation, prediction and completion problems for human motions. The model itself is originally designed for video generation, i.e., generating human action videos from random noise. We split the generation process into two stages: first, we generate human skeleton sequences from random noise, and then we transform from the skeleton images to the real pixel-level images (Fig. 3). In Section 3.1 we will elaborate the model and methods we use to generate human skeleton motion sequences, and in Section 3.2 we will present our novel method for solving the skeleton-to-image transformation problem. Lastly, in Section 3.3, we will show that we can specialize this model without modification to accomplish video prediction and completion by regarding them as constrained video generation.

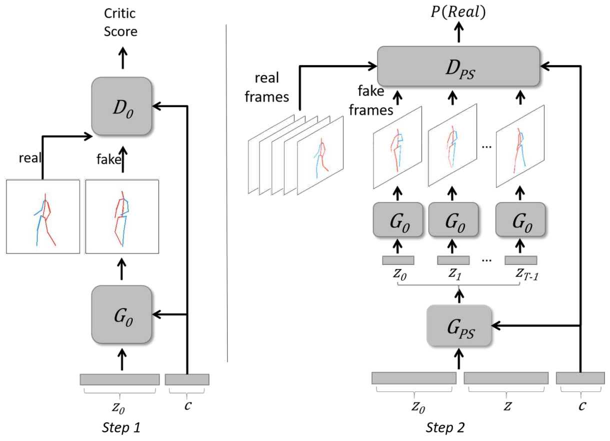


Fig. 4. Illustration of our two-step generation pipeline. In step one (left) G_0 takes a random noise vector and outputs the generated pose vector. The D_0 then differentiate between real and fake pose vectors. Both inputs to G_0 and D_0 are concatenated with conditional class vector. In step two (right), G_{PS} takes the random noise z conditioned on the latent vector of the first frame and the class vector, and generates a sequence of latent vectors which can be transformed to pose vectors via G_0. Then D_{PS} takes as input real/fake frames to determine $P(Real)$

3.1 General Generative Model

We propose a two-step generation model that generates human skeleton motion sequences from random noise.

Let J be the number of joints of human skeleton, and we represent each joint by its (x,y) location in image space. We formulate a skeleton motion sequence V as a collection of human skeletons across T consecutive frames in total, i.e., $V \in \mathbb{R}^{T\times 2J}$, where each skeleton frame $V_t \in \mathbb{R}^{2J}, t \in \{1 \cdots T\}$ is a vector containing all (x, y) joint locations. Our goal is to learn a function $G : \mathbb{R}^n \rightarrow \mathbb{R}^{T\times 2J}$ which maps an n-dimensional noise vector to a joint location vector sequence.

To find this mapping, our experiments showed that human pose constraints are too complicated to be captured by an end-to-end model trained from direct GAN method [10]. Therefore, we switch to our novel two-step strategy, where we first train a *Single Pose Generator* $G_0 : \mathbb{R}^m \rightarrow \mathbb{R}^{2J}$ which maps a m-dimensional latent vector to a single-frame pose vector, and then train a *Pose Sequence Generator* $G_{PS} : \mathbb{R}^n \rightarrow \mathbb{R}^{T\times m}$ which maps the input random noise to the latent vector sequences, the latter of which can be transformed into human pose vector sequences through our *Single Pose Generator* in a frame-by-frame manner.

Fig. 4 shows the overall pipeline and the results for each step. The advantage of adopting this two-step method is that by training the single-frame generator, we enforce human pose constraints on each frame, which alleviate the difficulty compared to end-to-end training and thus enable the model to generate longer

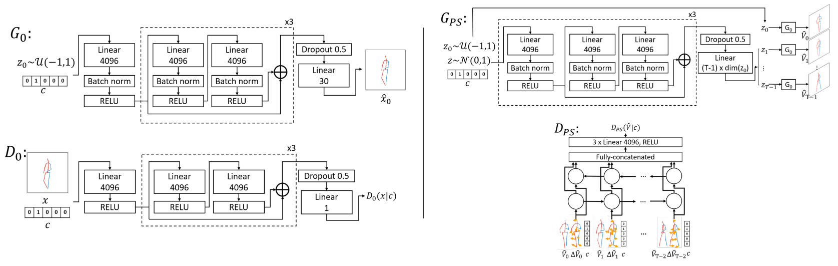


Fig. 5. Two-step generation architecture. Detailed architecture configuration of step one (on the left) and step two (on the right) are shown respectively. Here $\bigoplus$ stands for element wise addition and $\bigcirc$ stands for an LSTM cell

sequences. Additionally, in order to generate different types of motions, we employ the Conditional GAN [20] method and concatenate an one-hot class vector indicating the class of motion to the input of our generators.

Single Pose Generator In the first step, we employ the improved WGAN [11] method with gradient penalty for our adversarial training. We build a multilayer perceptron for both our generator and critic with similar structures and add condition to the input of both of them according to Conditional GAN [20]. Our generator G_0 takes as input an m-dimensional latent vector z_0 concatenated with a one-hot class vector c and outputs a pose vector $G_0(z_0|c)$. Our critic D_0 takes as input a real pose vector x_0 or a generated one, concatenated with c, yielding a critic score. The detailed architecture configurations are shown in Fig. 5, and are detailed in supplementary materials. Thus the WGAN objective is:

$$\min_{G_0} \max_{D_0 \in \mathcal{D}} \mathbb{E}_{c \sim p_c} [\mathbb{E}_{x_0 \sim p_{pose}} [D_0(x_0|c)] - \mathbb{E}_{z_0 \sim p_{z_0}} [D_0(G_0(z_0|c)|c)]] \tag{1}$$

where $\mathcal{D}$ is the set of 1-Lipschitz functions, p_c is the distribution of different classes, p_{pose} is the distribution of the real pose data, and p_{z_0} is the uniform noise distribution.

Pose Sequence Generator In the second step, we use the normal GAN [10] method instead for training our *Pose Sequence Generator*, since in our experiments normal GAN performs better than WGAN for this specific task. The generator G_{PS} generates a sequence of latent vectors, which are then fed into the *Single Pose Generator* resulting in a sequence of pose vectors $\hat{V}$, from a random noise vector z conditioned on z_0 and c. Note that z_0 is a random noise vector describing the initial condition of the generated pose.

In our implementation we generate latent vector sequences by generating the shifts between two consecutive frames, namely, the output of the network is $s_0, s_1, ..., s_{T-2}$ where $z_{t+1} = s_t + z_t$ for all $t \in \{0...T-2\}$ and z_t is the latent vector for the t-th frame (z_0 is given from the noise distribution).

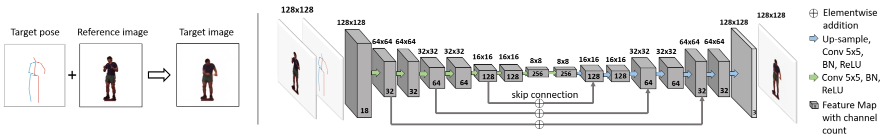


Fig. 6. Left: Transferring target pose to a real image. Right: Skeleton-to-Image Network. Image sizes and feature dimensions are shown in the figure. Note that the input has 18 channels, which consist of 3 RGB channels of reference image and 15 joint heat maps

For the discriminator, we employ a bi-directional LSTM structure, whose input of each time step t is the shift of consecutive frames $\Delta\hat{V}_t = \hat{V}_{t+1} - \hat{V}_t$ conditioned on $\hat{V}_t$ and c. The structural details are shown in Fig. 5. The objective function for the training in this step is:

$$\begin{aligned} \min_{G_{PS}} \max_{D_{PS}} \mathbb{E}_{c \sim p_c} [\mathbb{E}_{V \sim p_{video}} [\log D_{PS}(V|c)] + \\ \mathbb{E}_{z_0 \sim p_{z_0}, z \sim p_z} [\log(1 - D_{PS}(G_{PS}(z_0|z,c)|c))]] \end{aligned} \tag{2}$$

where $\mathcal{P}_c$ is the distribution of different classes, p_{video} is the distribution of the real video sequence data, p_{z_0} is the uniform noise distribution and p_z is the Gaussian noise distribution. We also add an L2 regularization term on the generated latent vector shifts for temporal smoothness.

3.2 Skeleton to Image Transformation

In this stage, we train a skeleton-to-image transformation to convert pose space to image space. Formally, given an input pose vector $x \in \mathbb{R}^{2J}$ and a reference image $y_0 \in \mathbb{R}^{w \times h \times 3}$ where h and w are the width and height of images, we need to transform x to a pixel-level image $y \in \mathbb{R}^{w \times h \times 3}$. In order to make the dimensions of inputs well-aligned, we first convert the pose vector x to a set of heat maps $S = (S_1, S_2, ..., S_J)$, where each heat map $S_j \in \mathbb{R}^{w \times h}, j \in \{1...J\}$ is a 2D representation of the probability that a particular joint occurs at each pixel location. Specifically, let $\mathbf{l_j} \in \mathbb{R}^2, (\mathbf{l_j} = (x_{2j}, x_{2j+1}))$ be the 2D position for joint j. The value at location $\mathbf{p} \in \mathbb{R}^2$ in the heat map S_j is then defined as,

$$S_j(\mathbf{p}) = \exp(-\frac{\|\mathbf{p} - \mathbf{l_j}\|_2^2}{\sigma^2}) \tag{3}$$

where σ controls the variance. Then our goal is to learn a function $F : \mathbb{R}^{w \times h \times J} \rightarrow \mathbb{R}^{w \times h \times 3}$ that transforms joint heat maps into pixel-level human images, conditioned on the input reference image. We train a supervised network here.

Skeleton-to-Image Network To learn our function F, we employ a U-Net like network [27, 17] (i.e., convolutional autoencoder with skip connections as shown in Fig. 6) that takes, as input, a set of joint heat maps S and a reference image y_0 and produces, as output, a human image $\hat{y}$. For the encoder part, we employ a convolutional network which is adequately deep so that the final receptive field covers the entire image. For the decoder part, we use symmetric structure to gradually generate the image. To avoid inherit checkerboard artifact in transposed convolution layers, there has been several papers proposing solutions including sub-pixel convolution, resize and convolution etc [24, 29, 7]. In our case we apply nearest neighbor up-sampling followed by convolution layer in decoder.

Loss Function To train our skeleton-to-image network, we compare the output image with the corresponding ground truth image by binary cross entropy loss. We calculate the binary cross entropy loss for intensity values at each pixel, i.e.

$$\mathcal{L}_{bce} = -\frac{1}{k} \sum (1-y) \log(1 - F(\mathbf{x}|y_0)) + y \log(F(\mathbf{x}|y_0)) \tag{4}$$

where y is the ground truth image, x is pixel and k is the number of pixels. Our experiments show that only using binary cross entropy loss tends to produce blurry results. Hence, in order to enforce details in the produced images, we further employ a feature-matching loss (in some papers also referred as perceptual loss), as suggested in [5, 14]. We match the activations in a pre-trained visual perception network that is applied to the ground truth image and the generated image respectively. Different layers in the network represent different levels of abstraction, providing comprehensive guidance towards more realistic images.

Specifically, let Φ be the visual perception network (VGG-19 [30]), and Φ_l be the activations in the l-th layer. Our feature-matching loss is defined as,

$$\mathcal{L}_2 = \sum_l \lambda_l \|\Phi_l(F(\mathbf{x}|y_0)) - \Phi_l(y)\|_1 \tag{5}$$

where λ_l is the weight for the l-th layer, which are manually set to balance the contribution of each term. For layers Φ_l, we use 'conv1_2', 'conv2_2', 'conv3_2', 'conv4_2' and 'conv5_2' in VGG-19 [30].

The overall loss for our skeleton-to-image network is therefore defined as

$$\mathcal{L} = \mathcal{L}_1 + \lambda \mathcal{L}_2 \tag{6}$$

where λ denotes the regularization factor of feature matching loss.

3.3 Prediction and Completion

To uniformly address video completion and video prediction, we model them as constrained video generation, which is ready to be defined by the general generative model. We optimize on the latent space to achieve our goal. For simplicity, the optimization is conducted on generated pose sequence, and we can transform to complete video by our skeleton-to-image transformer using the completed pose sequence. We utilize state-of-the-art human pose estimator like [22] to obtain pose sequences.

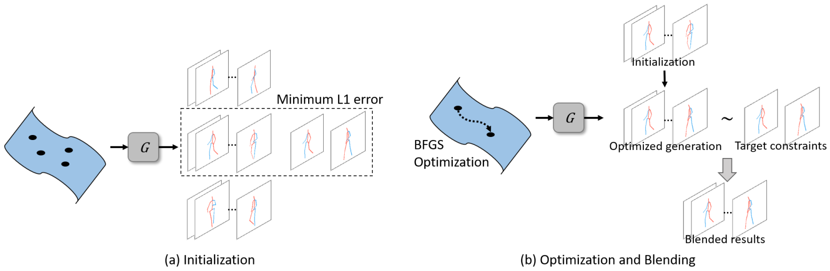


Fig. 7. Our completion/prediction pipeline. (a) Initialization: we randomly sample from the latent space and compare L1 error with the constraint frames. Dashed box shows the best initialization chosen. (b) We run BFGS optimization algorithms starting at our initialization, then finally blend the constraints and the generated results

Video Completion To fill in missing frames of a video, our method utilizes the generator G trained with full-length human pose sequence. The learned latent space $\mathbf{z}$ is effective in representing p_{data}. We perform video completion by finding the latent vector $\hat{\mathbf{z}}$ on the manifold that best fits the input frames constraint. As illustrated in Fig. 7, we can generate the missing content using the trained generative model G. The constraints can be arbitrary number of frames.

Objective Function: We regard this problem as an optimization problem. Let $\mathbf{I} \in \mathbb{R}^{t\times 2J}$ be the input frame constraints and $\mathbf{z}$ denote the learned latent space of G. We define the optimal completion encoding $\hat{\mathbf{z}}$ by:

$$\hat{\mathbf{z}} = \arg\min_{\mathbf{z}} \{\mathcal{L}_c(\mathbf{z}|\mathbf{I}) + \alpha \times \mathcal{L}_p(\mathbf{z})\}, \tag{7}$$

where $\mathcal{L}_c$ denotes the contextual L1 loss between the constrained frames and corresponding generated frames and $\mathcal{L}_p$ denotes the perceptual loss of generated frames, i.e. "realness" of the pose sequence. α denotes a regularization factor of the perceptual loss. $\mathcal{L}_c$ and $\mathcal{L}_p$ are defined as follows:

$$\mathcal{L}_c(\mathbf{z}|\mathbf{I}) = \sum_{i\in I} |G(\mathbf{z})_i - \mathbf{I}_i| \tag{8}$$

$$\mathcal{L}_p(\mathbf{z}) = -\log(D(G(\mathbf{z}))) \tag{9}$$

where $\mathbf{I}$ is the set of constrained frames, z is latent vector, i denotes the index of frames in $\mathbf{I}$; i can be arbitrary numbers subject to the given constraints. By optimizing Eq. (7), we obtain a full generated sequence $G(\hat{\mathbf{z}}) \in \mathbb{R}^{T\times 2J}$ which is the "closest" match to the input frames.

Two-Step Optimization: In order to optimize Eq. (7), we employ a two-step method illustrated in Fig.7. To address the optimization of such highly non-convex latent space, we first randomly sample from the latent space and compare the loss of Eq. (7) to find the best initialization, namely $\mathbf{z}_0$.

As proposed in [48], taking the initialization $\mathbf{z}_0$ as the starting point, we apply Limited Broyden-Fletcher-Goldfarb-Shanno optimization (L-BFGS-B) [3] on the $(n+m)$-dimension latent space to find the optimal completion result $\hat{\mathbf{z}}$.

Video Blending: Ideally, $G(\hat{\mathbf{z}})$ should be the result. However, slight shift and distortion from input constraints are observed as our method does not guarantee perfect alignment with the input. To address this, we use Poisson blending [25] to make our final pose sequence consistent with the input constraints. The key idea is to maintain the gradients on the temporal direction of $G(\hat{\mathbf{z}})$ to preserve motion smoothness while shifting the generated frames to match the input constraint. Our final solution, $\hat{\mathbf{x}}$, can be obtained by

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\nabla_t \mathbf{x} - \nabla_t G(\hat{\mathbf{z}})\|_2^2, \text{ s.t. } \mathbf{x}_i = \mathbf{I}_i \text{ for } i \in \mathbb{R}^{t\times 2J} \tag{10}$$

where ∇_t is the gradient operator on the temporal dimension. This blending preserves the naturalness of the videos while better aligning with the input frame constraints.

Video Prediction Video prediction can be solved under the same general framework (same as in Fig. 7) as it can be essentially interpreted as video generation with first few frames as constraints.

Formally, let $I \in \mathbb{R}^{t\times 2J}$ be consecutive frames at time step 0 to t as input, we generate future frames $G_t, G_{t+1}, \cdots G_T$ so that $I_0, I_1, \cdots, I_t, G_{t+1}, \cdots G_T$ form a natural and semantically meaningful video. To achieve such goal, we model video prediction as video generation with first few frames as constraint. In other words, we perform the same steps as in the previous section with the input described above to obtain a completed video sequence.

4 Experiments

4.1 Dataset

We evaluate our model primarily on Human3.6m dataset [12]. The dataset provides ground truth 2D human poses. In our experiments, in order to reduce redundant frames and encourage larger motion variations, we subsample the video frames to 16 fps. The action classes we select are ‘Direction’, ‘Greeting’, ‘Sitting’, ‘Sitting Down’, ‘Walking’, all of which contain large human motions.

For our skeleton sequence generation task, we randomly select 5 subjects as training set and reserve 2 subjects as test set. For our skeleton-to-image transformation task, we treat the unchosen action classes as training set, and our chosen 5 action classes as test set.

Since our major concern is human motion, we thus subtract all the backgrounds and generate the foreground human figure only for this dataset. To test our method under real-world setting with background, we further train our networks on UCF-101 [32] training set, and test the model using UCF-101 [32] test set as well as real-world movie footages from Forrest Gump (1995).

4.2 Evaluation

Evaluating the quality of synthesized videos is a difficult problem for video generation due to no corresponding ground truth. For video prediction and completion, one can measure the difference from the ground truth frames by Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [40], but we argue that, since videos tend to have multiple possible futures, it is not advisable to compare predicted results against one ground truth. Furthermore, they do not measure the temporal smoothness and human-likeness.

In order to evaluate the visual quality of our results, we measure whether our generated videos are adequately realistic such that a pre-trained recognition network can recognize the object and action in the generated video. This method is inherently similar to Inception Score in [37, 28], object detection evaluation in [39] and Semantic Interpretability in [47]. Though Inception Score has its limitations [2], it remains the best systematic metric for video generation. Current state-of-the-art video action recognition model is the two-stream network proposed by Yan et al. [31] and improved by Wang et al [38]. We employ [38] and fine-tune it on our dataset, and evaluate the following two scores measuring the visual quality of generated image frames and video sequences respectively:

Inception Score for frames One criterion for video evaluation score is that they should reflect if the video contains natural images along the sequence. Thus we calculate the inception score [28] based on the output classification result of the RGB stream [38] for each frame generated as the evaluation metric. The average score across the whole video should reflect the overall image quality. Additionally, we also show the Inception Score obtained at each time step, which gives us a detailed snapshot of how the quality of video vary over time.

Inception Score for videos As proposed in [37], we evaluate the inception score [28] based on the fused classification results from our two-stream action classifier. By taking in to consideration the motion flow across the whole video, the output classes serve as an accurate indicator of the actions perceived in the video. Thus such score can give an overall quality of the full video sequence.

4.3 Baselines

We present several baseline methods to provide comparisons of our results with results from previous methods.

For Video Generation, our baseline is *Video-GAN* (VGAN) [35]. This approach trains a GAN that generates videos in pixel space. It is first successful attempt on video generation with deep learning methods.

For Video Prediction, the first baseline is *PredNet* [16], one of the latest results in video prediction. The second baseline is *Multi-Scale GAN* (MS-GAN) proposed by Mathieu et al. [19]. This approach has been successful in various video prediction tasks including human action videos. The third baseline is *PoseVAE*, a sequential model proposed in [37], which utilized pose representation and have produced state-of-the-art results.

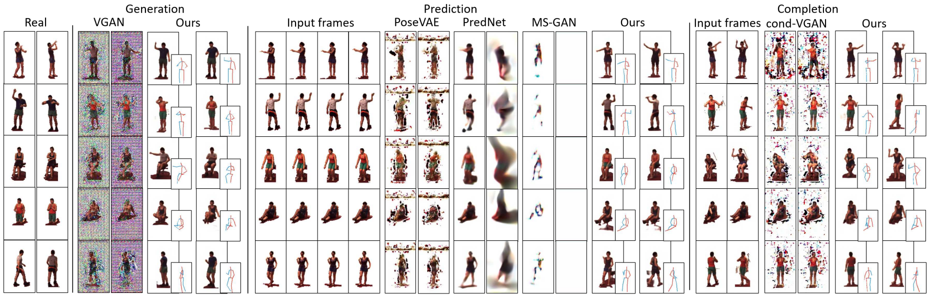


Fig. 8. Qualitative comparisons. Each image-pair column corresponds to a generation method (the first column is real data), and columns are grouped together in the order of generation, prediction and completion, respectively. Each row corresponds to an action class, from top to bottom: Direction, Greeting, Sitting, Sitting Down, Walking. For each method we show the 10th and the 40th frames. For our method we also show the generated pose results

For Video Completion, our baseline is *Conditional Video-GAN* (cond-VGAN) [35]. The model can predict next frames given input as in the paper, therefore we adapt it to video completion by changing its input to the first and last frame.

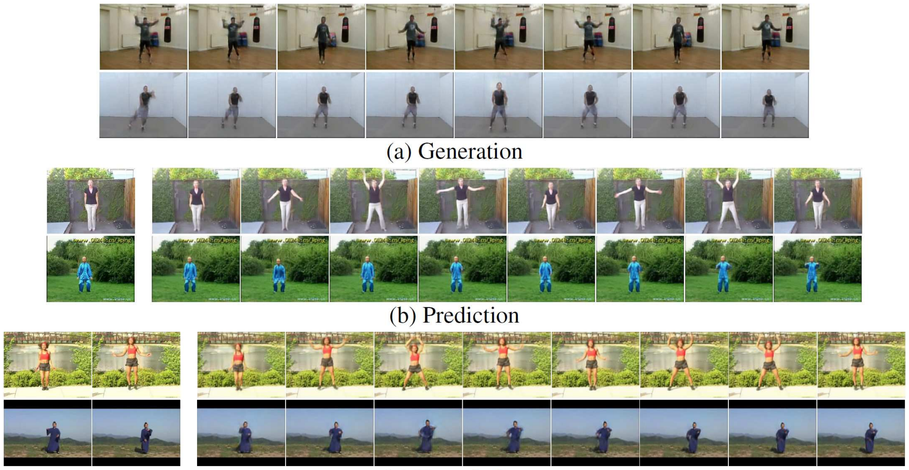
5 Results

For video generation, we generate videos from random noise vectors with dimensions consistent with the proposed models. For video prediction, we feed the first 4 frames as inputs, i.e. the baselines make prediction based on the input 4 frames, and our model generates videos with the first 4 frames as constraints. For video completion, we fix the the first and the last frames as constraints. In order to calculate the proposed metrics, we randomly generate 320 50-frame video samples for each method (except for the Video-GAN method [35] which is fixed by architecture to generate only 32 frames).

5.1 Qualitative Results

In Fig. 8 we show the qualitative results of our model on Human3.6m dataset [12], in comparison with other state-of-the-art methods. Since the results are videos, we strongly suggest readers to check our supplementary materials. The baseline methods are all fine-tuned/re-trained on our Human3.6m dataset [12]. We show generated results for each of our selected classes. Due to the page limit, we only show the beginning and the middle frames in the result videos.

By examining the results, we find that our model is capable of generating plausible human motion videos with high visual quality. In terms of image quality, we find that our model generates the most compelling human images, while other models tend to generate noisy (particularly Video-GAN) and blurry results



(a) Generation

(b) Prediction

(c) Completion

Fig. 9. Real-world results on UCF-101. For each task, we display 8 frames of our generated videos for the *JumpingJack* (1st row) and *TaiChi* (2nd row) actions. (a) is generated from random noise, (b) is generated given the first four frames (we only show the first frame in the first column), and (c) is generated given the first and last frames (shown in the first two columns). See full videos in supplemental material

due to their structural limitations. By examining the video sequences (provided in supplementary materials), we find that our model can generate natural and interpretable human motions. A key distinction is that we are able to produce large-scale and detailed motion. Another important observation is that, our results maintain high quality over the entire time interval, while the others' quality (especially prediction models) tend to degrade quickly over time.

In Fig. 9 we show the qualitative results for all three tasks on real-world video scenes from UCF-101 [32] to demonstrate our model's capability under real-world environments with background. As shown in the results, we successfully generate videos with high visual quality and interpretability. Additionally, we also test our model on real-world movie footages from the famous Forrest Gump scenes, as shown in Fig. 2. We generate a *Directing* action conducted by the little boy using the running scene as a reference.

5.2 Quantitative Results

Table 1 tabulates our quantitative evaluation results, "frame-IS" stands for Inception Score for frames, and "video-IS" stands for Inception Score for videos. While the ground truth (real) videos have the largest Inception Scores of both types, which matches our intuition, our generated videos have the highest scores among all the competing methods. This suggests that our model generates videos that possess meaningful visual features closer to real videos in both image and

Table 1. Frame and Video Inception Score (IS)

Method	Real	VGAN [35]	Ours	PoseVAE [37]	PredNet [16]	MS-GAN [19]	Ours	cond-VGAN	Ours
frame-IS	**4.53 ± 0.01**	1.53 ± 0.04	**3.99 ± 0.02**	1.91 ± 0.01	2.60 ± 0.04	1.48 ± 0.01	**3.87 ± 0.02**	2.35 ± 0.02	**3.91 ± 0.02**
video-IS	**4.63 ± 0.09**	1.40 ± 0.16	**3.99 ± 0.18**	2.17 ± 0.11	2.94 ± 0.15	1.88 ± 0.10	**4.09 ± 0.15**	2.00 ± 0.06	**4.10 ± 0.07**

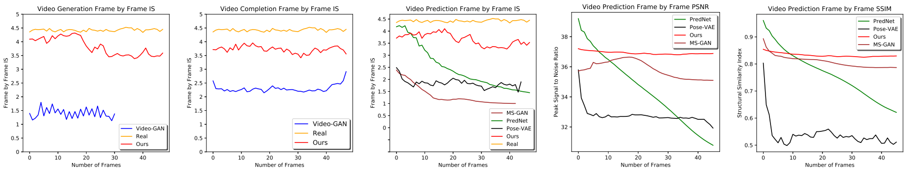


Fig. 10. Left three figures: Frame-by-Frame Inception Score for generation, completion and prediction, respectively. Right two figures: Frame-by-Frame PSNR and SSIM for prediction

video (temporal) domains, thus further indicating that our videos are more realistic. We also observe that other methods have much lower scores than ours, and VGAN [35] and MS-GAN [19] are even worse than PredNet [16]. All the statistics are consistent with our qualitative results.

Fig. 10 (left three figures) shows a comparison of frame-by-frame Inception Score. We find that the ground truth videos maintain the highest scores at all time steps, and our results have considerably high scores closest to the ground truth quality. A more important observation is that, for the compared prediction models, PredNet [16] and MS-GAN [19], the scores tend to fall across time, indicating that the image quality is deteriorating over time. Although PoseVAE [37] does not decline, its overall image quality is much lower than ours. This observation is consistent with our qualitative evaluation. We also show in Fig. 10 (right two figures) the frame-by-frame PSNR and SSIM (though these are not encouraged). Our methods still outperform others by a large margin in longer timespan. This further illustrates our improvement over current state-of-the-arts.

6 Conclusion

We present a general generative model that addresses the problem of video generation, prediction and completion uniformly. By utilizing human pose as intermediate step with our novel generation strategy, we are able to generate large-scale human motion videos with longer duration. We are then able to solve the later two problems by constrained generation using our model. We find that our model can generate plausible human action videos both from scratch and under constraint, which surpasses current methods both quantitatively and visually.

Acknowledgement This research is supported in part by Tencent Youtu.

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN. ArXiv e-prints (Jan 2017)
2. Barratt, S., Sharma, R.: A note on the inception score. arXiv preprint arXiv:1801.01973 (2018)
3. Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing **16**(5), 1190–1208 (1995)
4. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: CVPR (2017)
5. Chen, Q., Koltun, V.: Photographic image synthesis with cascaded refinement networks. arXiv preprint arXiv:1707.09405 (2017)
6. Chen, Y., Shen, C., Wei, X., Liu, L., Yang, J.: Adversarial posenet: A structure-aware convolutional network for human pose estimation. CoRR **abs/1705.00389** (2017), http://arxiv.org/abs/1705.00389
7. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence **38**(2), 295–307 (2016)
8. Gatys, L., Ecker, A.S., Bethge, M.: Texture synthesis using convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 262–270 (2015)
9. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2414–2423 (2016)
10. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS, pp. 2672–2680 (2014), http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf
11. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. CoRR **abs/1704.00028** (2017), http://arxiv.org/abs/1704.00028
12. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(7), 1325–1339 (jul 2014)
13. Jia, J., Tai, Y.W., Wu, T.P., Tang, C.K.: Video repairing under variable illumination using cyclic motions. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(5), 832–839 (2006)
14. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision. pp. 694–711. Springer (2016)
15. Li, C., Wand, M.: Combining markov random fields and convolutional neural networks for image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2479–2486 (2016)
16. Lotter, W., Kreiman, G., Cox, D.: Deep predictive coding networks for video prediction and unsupervised learning. arXiv preprint arXiv:1605.08104 (2016)
17. Ma, L., Jia, X., Sun, Q., Schiele, B., Tuytelaars, T., Van Gool, L.: Pose guided person image generation. arXiv preprint arXiv:1705.09368 (2017)
18. Marwah, T., Mittal, G., Balasubramanian, V.N.: Attentive semantic video generation using captions. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 1435–1443. IEEE (2017)

19. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. CoRR **abs/1511.05440** (2015), http://arxiv.org/abs/1511.05440
20. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)
21. Mordvintsev, A., Olah, C., Tyka, M.: Inceptionism: Going deeper into neural networks. Google Research Blog. Retrieved June **20**, 14 (2015)
22. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision. pp. 483–499. Springer (2016)
23. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive separable convolution. arXiv preprint arXiv:1708.01692 (2017)
24. Odena, A., Dumoulin, V., Olah, C.: Deconvolution and checkerboard artifacts. Distill **1**(10), e3 (2016)
25. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. In: ACM Transactions on graphics (TOG). vol. 22, pp. 313–318. ACM (2003)
26. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
27. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. CoRR **abs/1505.04597** (2015), http://arxiv.org/abs/1505.04597
28. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X., Chen, X.: Improved techniques for training gans. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 2234–2242. Curran Associates, Inc. (2016), http://papers.nips.cc/paper/6125-improved-techniques-for-training-gans.pdf
29. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1874–1883 (2016)
30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014)
31. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems. pp. 568–576 (2014)
32. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
33. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1653–1660. CVPR '14, IEEE Computer Society, Washington, DC, USA (2014). https://doi.org/10.1109/CVPR.2014.214, http://dx.doi.org/10.1109/CVPR.2014.214
34. Villegas, R., Yang, J., Zou, Y., Sohn, S., Lin, X., Lee, H.: Learning to generate long-term future via hierarchical prediction. In: International Conference on Machine Learning. pp. 3560–3569 (2017)
35. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29, pp. 613–621. Curran Associates, Inc. (2016), http://papers.nips.cc/paper/6194-generating-videos-with-scene-dynamics.pdf

36. Walker, J., Doersch, C., Gupta, A., Hebert, M.: An uncertain future: Forecasting from static images using variational autoencoders. In: European Conference on Computer Vision. pp. 835–851. Springer (2016)
37. Walker, J., Marino, K., Gupta, A., Hebert, M.: The pose knows: Video forecasting by generating pose futures. CoRR **abs/1705.00053** (2017), http://arxiv.org/abs/1705.00053
38. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: European Conference on Computer Vision. pp. 20–36. Springer (2016)
39. Wang, X., Gupta, A.: Generative image modeling using style and structure adversarial networks. In: European Conference on Computer Vision. pp. 318–335. Springer (2016)
40. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
41. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. In: CVPR (2016)
42. Wexler, Y., Shechtman, E., Irani, M.: Space-time completion of video. IEEE Trans. Pattern Anal. Mach. Intell. **29**(3), 463–476 (Mar 2007). https://doi.org/10.1109/TPAMI.2007.60, http://dx.doi.org/10.1109/TPAMI.2007.60
43. Xue, T., Wu, J., Bouman, K., Freeman, B.: Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks. In: Advances in Neural Information Processing Systems. pp. 91–99 (2016)
44. Yan, Y., Xu, J., Ni, B., Yang, X.: Skeleton-aided articulated motion generation. arXiv preprint arXiv:1707.01058 (2017)
45. Yeh, R.A., Chen, C., Lim, T.Y., Schwing, A.G., Hasegawa-Johnson, M., Do, M.N.: Semantic image inpainting with deep generative models. In: CVPR (2017)
46. Zanfir, M., Popa, A.I., Zanfir, A., Sminchisescu, C.: Human appearance transfer
47. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: European Conference on Computer Vision. pp. 649–666. Springer (2016)
48. Zhu, J.Y., Krähenbühl, P., Shechtman, E., Efros, A.A.: Generative visual manipulation on the natural image manifold. In: European Conference on Computer Vision. pp. 597–613. Springer (2016)