MT-VAE: Learning Motion Transformations to Generate Multimodal Human Dynamics

Xinchen Yan[1*]   Akash Rastogi[1]   Ruben Villegas[1]   Kalyan Sunkavalli[2]
Eli Shechtman[2]   Sunil Hadap[2]   Ersin Yumer[3]   Honglak Lee[1,4]

[1] University of Michigan, Ann Arbor
[2] Adobe Research
[3] Argo AI
[4] Google Brain

Abstract. Long-term human motion can be represented as a series of *motion modes*—motion sequences that capture short-term temporal dynamics—with transitions between them. We leverage this structure and present a novel *Motion Transformation Variational Auto-Encoders (MT-VAE)* for learning motion sequence generation. Our model jointly learns a feature embedding for motion modes (that the motion sequence can be reconstructed from) and a feature transformation that represents the transition of one motion mode to the next motion mode. Our model is able to generate multiple diverse and plausible motion sequences in the future from the same input. We apply our approach to both facial and full body motion, and demonstrate applications like analogy-based motion transfer and video synthesis.

1   Introduction

Modeling the dynamics of human motion — both facial and full body motion — is a fundamental problem in computer vision, graphics, and machine intelligence, with applications ranging from virtual characters [1,2], video-based animation and editing [3,4,5], and human-robot interfaces [6]. Human motion is known to be highly structured and can be modeled as a sequence of atomic units that we refer to as *motion modes*. A motion mode captures the *short-term* temporal dynamics of a human action (e.g., smiling or walking), including its related stylistic attributes (e.g., how wide is the smile, how fast is the walk). Over the *long-term*, a human action sequence can be segmented into a series of motion modes with transitions between them (e.g., a transition from a neutral expression to smiling to laughing). This structure is well known (referred to as basis motions [7] or walk cycles) and widely used in computer animation.

This paper leverages this structure to learn to generate human motion sequences, i.e., given a short human action sequence (present motion mode), we want to synthesize the action going forward (future motion mode). We hypothesize that (1) each motion mode can be represented as a low-dimensional feature vector, and (2) transitions between motion modes can be modeled as *transformations* of these features. As shown
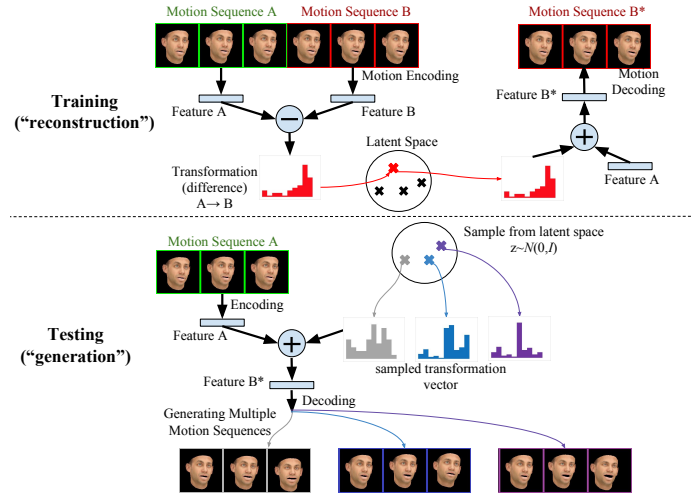

Fig. 1: Top: Learning motion sequence generation using *Motion Transformation VAE*. Bottom: Generating multiple future motion sequences from the transformation space.

in Figure 1, we present a novel model termed *Motion Transformation Variational Auto-Encoders (MT-VAE)* for learning motion sequence generation. Our MT-VAE is implemented using an LSTM encoder-decoder that embeds each short sub-sequence into a feature vector that can be decoded to reconstruct the motion. We further assume that the transition between current and future modes can be captured by a certain transformation. In the paper, we demonstrate that the proposed MT-VAE learns a motion feature representation in an unsupervised way.

A challenge with human motion is that it is inherently multimodal, i.e., the same initial motion mode could transition into different motion modes (e.g., a smile could transition to a frown, or a smile while looking left, or a wider smile, etc.). A deterministic model would not be able to learn these variations and may collapse to a single-mode distribution. Our MT-VAE supports a stochastic sampling of the feature transformations to generate multiple plausible output motion modes from a single input. This allows us to model transitions that may be rare (or potentially absent) in the the training set.

We demonstrate our approach on both facial and full human body motions. In both domains, we conduct extensive ablation studies and comparisons with previous work showing that our generation results are more plausible (i.e., better preserve the structure of human dynamics) and diverse (i.e., explore multiple motion modes). We further demonstrate applications like 1) analogy-based motion transfer (e.g., transferring the act of smiling from one pose to another pose) and 2) future video synthesis (i.e., generating multiple possible future videos given input frames with human motions). Our key contributions are summarized as follows:

- We propose a generative motion model that consists of a sequence-level motion feature embedding and feature transformations, and show that it can be trained in an unsupervised manner.

- We show that stochastically sampling the transformation space is able to generate future motion dynamics that are diverse and plausible.
- We demonstrate applications of the learned model to challenging tasks like motion transfer and future video synthesis for both facial and human body motions.

2   Related Work

Understanding and modeling human motion dynamics has been a long-standing problem for decades [8,9,10]. Due to the high dimensionality of video data, early work mainly focused on learning hierarchical spatio-temporal representations for video event and action recognition [11,12,13]. In recent years, predicting and synthesizing motion dynamics using deep neural networks has become a popular research topic. Walker et al. [14], Fischer et al. [15] learn to synthesize dense flow in the future from a single image. Walker et al. [16] extended the deterministic prediction framework by modeling the flow uncertainty using variational auto-encoders. Chao et al. [17] proposed a recurrent neural network to generate movement of 3D human joints from a single observation with a 3D in-network projection layer. Taking one step further, Villegas et al. [18], Walker et al. [19] explored hierarchical structure (e.g., 2D human joints) for motion prediction in the future using recurrent neural networks. Li et al. [20] proposed an auto-conditional recurrent framework to generate long-term human motion dynamics through time. Besides human motion, face synthesis and editing is another interesting topic in vision and graphics. Methods for reenacting and interpolating face sequences in video have been developed [3,21,22,23] based on a 3D morphable face representation [24]. Very recently, Suwajanakorn et al. [5] introduced a speech-driven face synthesis system that learns to generate lip motions with a recurrent neural network.

Besides the flow representation, motion synthesis has been explored in a broader context, namely, video generation. For example, synthesizing video sequence in the future from a single or multiple video frames as initialization. Early works employed patch-based method for short-term video generation using mean squared mean squared loss [25] or perceptual loss [26]. Given an atomic action as additional condition, previous works extended with action-conditioned (i.e., rotation, location, etc) architectures that enable better semantic control in video generation [27,28,29,30]. Due to the difficulty in holistic video frame prediction, the idea of disentangling video factors into motion and content is explored in [31,32,33,34,35,36]. Video generation has also been approached with architectures that output multinomial distribution vectors over the possible pixel values for each pixel in the generated frame [37].

The notion of feature transformations has also been exploited for other tasks. Mikolov et al. [38] showcased the composition additive property of word vectors learned in an unsupervised way from language data; Kulkarni et al. [39], Reed et al. [40] suggested that additive transformation can be achieved via reconstruction or prediction task by learning from parallel paired image data. In the video domain, Wang et al. [41] studied a transformation-aware representation for semantic human action classification; Zhou et al. [42] investigated time-lapse video generation given additional class labels.

Multimodal conditional generation has recently been explored for images [43,44], sketch drawings [45], natural language [46,47], and video prediction [48,49]. As noted

in previous work, learning to generate diverse and plausible visual data is very challenging for the following reasons: first, mode collapse may occur without one-to-many pairs. Collecting sequence data where one-to-many pairs exist is non-trivial. Second, posterior collapse could happen when the generation model is based on a recurrent neural network.

3   Problem Formulation and Methods

We start by giving an overview of our problem. We are given a sequence of T observations $S_A = [x_1, x_2, \cdots, x_T]$, where $x_t \in \mathbb{R}^D$ is a D dimensional vector representing the observation at time t. These observations encode the structure of the moving object and can be represented in different ways, for e.g., as keypoint locations or shape and pose parameters. *Changes* in these observations encode the motion that we are interested in modeling. We refer to the entire sequence as a *motion mode*. Given a motion mode, $S_A \in \mathbb{R}^{T \times D}$, we aim to build a model that is capable of predicting a future motion mode, $S_B = [y_1, y_2, \cdots, y_T]$, where $y_t \in \mathbb{R}^D$ represents the predicted t-th step in the *future*, i.e., $y_1 = x_{T+1}$. We first start with a discussion of two potential baseline models that could be used for this task (Section 3.1), and then present our method (Section 3.2).

3.1   Preliminaries

Prediction LSTM for Sequence Generation. Figure 2(a) shows a simple encoder-decoder LSTM [50,25] as a baseline for the motion prediction task. At time t, the encoder LSTM takes the motion x_t as input and updates its internal representation. After going through the entire motion mode S_A, it outputs a fixed-length feature $e_A \in \mathbb{R}^{N_e}$ as an intermediate representation. We initialize the internal representation of decoder LSTM using the feature e_A computed. At time t of the decoding stage, the decoder LSTM predicts the motion y_t. This way, the decoder LSTM gradually predicts the entire motion mode $S_B^* = [y_1, y_2, \cdots, y_T]$ in the future within T steps. We denote the encoder LSTM as function $f : \mathbb{R}^{T \times D} \to \mathbb{R}^{N_e}$ and the decoder LSTM as function $g : \mathbb{R}^{N_e} \to \mathbb{R}^{T \times D}$. As a design choice, we initialize the decoder LSTM with additional input x_T for smoother prediction.

Vanilla VAE for Sequence Generation. As the deterministic LSTM model fails to reflect the multimodal nature of human motion, we consider a statistical model, $p_\theta(S_B|S_A)$, parameterized by θ. Given the observed sequence S_A, the model estimates a probability for the possible future sequence S_B instead of a single outcome. To model the multimodality (i.e., S_A can transition to different S_B's), a latent variable z (sampled from prior distribution) is introduced to capture the inherent uncertainty. The future sequence S_B is generated as follows:

1. Sample latent variable $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
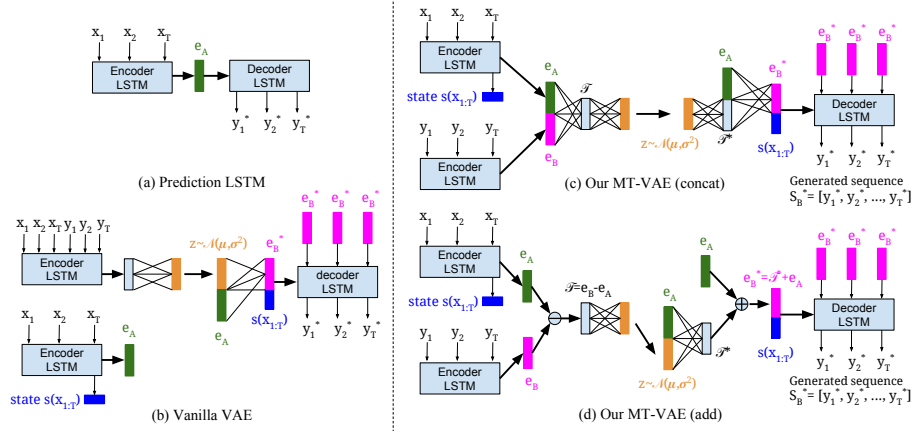2. Given S_A and z, generate a sequence of length T: $S_B \sim p_\theta(S_B|z, S_A)$;

Fig. 2: Illustrations of different models for motion sequence generation. $s(x_{1:T})$ indicates the hidden state of the Encoder LSTM at time T.

Following previous work on VAEs [51,43,52,53,16,33,19], the objective is to maximize the variational lower-bound of the conditional log-probability $\log p_\theta(S_B|S_A)$:

$$\mathcal{L}_{\text{VAE}} = -KL(q_\phi(z|S_B, S_A)||p_\theta(z)) + \mathbb{E}_{q_\phi(z|S_B,S_A)}\big[\log p_\theta(S_B|S_A, z)\big] \qquad (1)$$

In Eq. 1, $q_\phi(z|S_B, S_A)$ is referred as an auxiliary posterior that approximates the true posterior $p_\theta(z|S_B, S_A)$. Specifically, the prior $p_\theta(z)$ is assumed to be $\mathcal{N}(\mathbf{0}, \mathbf{I})$. The posterior $q_\phi(z|S_B, S_A)$ is a multivariate Gaussian distribution with mean and variance μ_ϕ and σ_ϕ^2, respectively. Intuitively, the first term in Eq. 1 regularizes the auxiliary posterior $q_\phi(z|S_B, S_A)$ with prior $p_\theta(z)$. The second term $\log p_\theta(S_B|S_A, z)$ can be considered as an auto-encoding loss, where we refer to $q_\phi(z|S_B, S_A)$ as an encoder or recognition model, and $p_\theta(S_B|z, S_A)$ as a decoder or generation model.

As shown in Figure 2(b), the vanilla VAE model adopts similar LSTM encoder and decoder for sequence processing. In contrast to Prediction LSTM model, the vanilla VAE decoder takes both motion feature e_A and latent variable z into account. Ideally, this allows to generate diverse motion sequences by drawing different samples from the latent space. However, the semantic role of the latent variable z in this vanilla VAE model is not straight-forward and may not effectively represent long-term trends (e.g., dynamics in a specific motion mode or during change of modes).

3.2 Motion-to-Motion Transformations in Latent Space

To further improve motion sequence generation beyond vanilla VAE, we propose to explicitly enforce the structure of motion modes in the latent space. We assume that (1) each motion mode can be represented as low-dimensional feature vector, and (2) transitions between motion modes can be modeled as *transformations* of these features. Our design is also supported by early studies on hierarchical motion modeling and prediction [8,54,55].

We present a *Motion Transformation VAE (or MT-VAE)* (Fig. 2(c)) with four components:

1. An LSTM encoder $f : \mathbb{R}^{T \times D} \to \mathbb{R}^{N_e}$ maps the input sequences into motion features through $e_A = f(S_A)$ and $e_B = f(S_B)$, respectively.
2. A latent encoder $h_{e \to z} : \mathbb{R}^{2 \times N_e} \to \mathbb{R}^{N_z}$ computes the transformation in the latent space $z = h_{e \to z}([e_A, e_B])$ by concatenating motion features e_A and e_B. Here, N_z indicates the latent space dimension.
3. A latent decoder $h_{z \to e} : \mathbb{R}^{N_z + N_e} \to \mathbb{R}^{N_e}$ synthesizes the motion feature in the future from latent transformation z and current motion feature e_A via $e_B^* = h_{z \to e}([z, e_A])$.
4. An LSTM decoder $g : \mathbb{R}^{N_e} \to \mathbb{R}^{T \times D}$ synthesizes the future sequence given motion feature: $S_B^* = g(e_B^*)$.

Similar to the Prediction LSTM, we use an LSTM encoder/decoder to map motion modes into feature space. The MT-VAE further maps these features into *latent transformations* and stochastically samples these transformations. As we demonstrate, this change makes the model more expressive and leads to more plausible results. Finally, in the sequence decoding stage of MT-VAE, we feed the synthesized motion feature e_B^* as input to the decoder LSTM, with internal state initialized using the same motion feature e_B^* with an additional input x_t.

3.3   Additive Transformations in Latent Space

Although MT-VAE explicitly models motion transformations in latent space, this space might be unconstrained because the transformations are computed from vector concatenation of motion features e_A and e_B in our latent encoder $h_{e \to z}$. To better regularize the transformation space, we present an additive variant of MT-VAE, that is depicted in Figure 2(d). To distinguish between the two variants, we call the previous model *MT-VAE (concat)* and this model *MT-VAE (add)*, respectively. Our model is inspired by recent success of *deep analogy-making* methods [40,31] where a relation (or transformation) between two examples can be represented as a difference in the embedding space. In this model, we strictly constrain the latent encoding and decoding steps as follows:

1. Our latent encoder $h_{\mathcal{T} \to z} : \mathbb{R}^{N_e} \to \mathbb{R}^{N_z}$ computes the difference between two motion features e_A and e_B via $\mathcal{T} = e_B - e_A$; then it maps the difference feature $\mathcal{T}$ into a transformation in the latent space via $z = h_{\mathcal{T} \to z}(\mathcal{T})$.
2. Our latent decoder $h_{z \to \mathcal{T}} : \mathbb{R}^{N_z + N_e} \to \mathbb{R}^{N_e}$ reconstructs the difference feature $\mathcal{T}^*$ from latent variable z and current motion feature e_A via $\mathcal{T}^* = h_{z \to \mathcal{T}}(z, e_A)$.
3. Finally, we apply a simple additive interaction to reconstruct the motion feature via $e_B^* = e_A + \mathcal{T}^*$;

In step one, we infer the latent variable using $h_{\mathcal{T} \to z}$ from the difference of e_A and e_B (instead of a applying a linear layer on concatenated vectors). Intuitively, the latent code is expected to capture the mode transition from the current motion to the future motion rather than a concatenation of two modes. In step two, we reconstruct
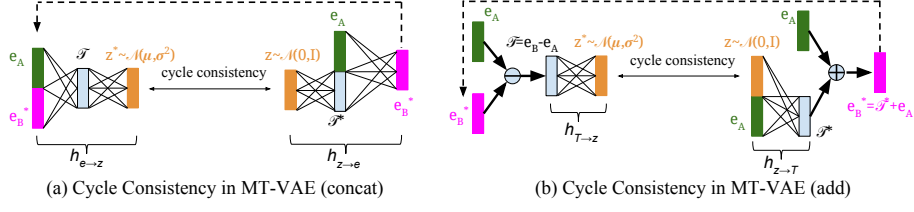
(a) Cycle Consistency in MT-VAE (concat)   (b) Cycle Consistency in MT-VAE (add)

Fig. 3: Illustrations of cycle consistency in MT-VAE variations.

the transformation from the latent variable via $h_{z \to \mathcal{T}}(z, e_A)$ where z is obtained from recognition model. In this design, the feature difference is dependent on both latent transformation z and current motion feature e_A. Alternatively, we can make our latent decoder $h_{z \to \mathcal{T}}$ context-free by removing input from motion feature e_A. This way, the latent decoder is supposed to hallucinate the motion difference solely from the latent space. We provide this ablation study in Section 4.1.

Besides the architecture-wise regularization, we introduce two additional objectives while training our model.

Cycle Consistency. As mentioned previously, our training objective $\mathcal{L}_{\text{VAE}}$ in Eq. 1 is composed of a KL term and a reconstruction term at each frame. The KL term regularizes the latent space, while the reconstruction term ensures that the data can be explained by our generative model. However, we do not have direct regularization in the feature space. We therefore introduce a cycle-consistency loss in Eq. 2 (for MT-VAE (concat)) and Eq. 3 (for MT-VAE (add)). Figure 3 illustrates the cycle consistency in details.

$$\mathcal{L}_{\text{cycle}}^{\text{concat}} = ||z^* - z||, \text{ where } z^* = h_{e \to z}([e_A, h_{z \to e}(z, e_A)]) \text{ and } z \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (2)$$

$$\mathcal{L}_{\text{cycle}}^{\text{add}} = ||z^* - z||, \text{ where } z^* = h_{\mathcal{T} \to z}(h_{z \to \mathcal{T}}(z, e_A)) \text{ and } z \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (3)$$

In our preliminary experiments, we also investigated a consistency loss with a bigger cycle (involving the actual motion sequences) during training but we found it ineffective as a regularization term in our setting. We hypothesize that vanishing or exploding gradients make the cycle-consistency objective less effective, which is a known issue when training recurrent neural networks.

Motion Coherence. Specific to our motion generation task, we introduce a motion coherence loss in Eq. 4 that encourages a smooth transition in velocity in the first K steps of prediction. We define the velocity $v_1 = y_1 - x_T$ and $v_k = y_k - y_{k-1}$ when $k \geq 2$. Intuitively, such loss prevents the generated sequence from deviating too far from the future sequence sampled from the prior.

$$\mathcal{L}_{\text{motion}} = \frac{1}{K} \sum_{t=1}^{K} ||v_t^* - v_t||, \text{ where } g(e_B^z) = [y_1^*, \cdots, y_T^*] \text{ and } z \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (4)$$

Finally, we summarize our overall loss in Eq. 5, where λ_{cycle} and λ_{motion} are two balancing hyper-parameters for cycle consistency and motion coherence, respectively.

$$\mathcal{L}_{\text{MT-VAE}} = \mathcal{L}_{\text{VAE}} + \lambda_{\text{cycle}}\mathcal{L}_{\text{cycle}} + \lambda_{\text{motion}}\mathcal{L}_{\text{motion}} \tag{5}$$

4   Experiments

Datasets. The evaluation is conducted on the datasets involving two representative human motion modeling tasks: Affect-in-the-wild (Aff-Wild) [56] for facial motions and Human3.6M [57] for full body motions. To better focus on face motion modeling (e.g., expressions and head movements), we leveraged the 3D morphable face model [58,24] (e.g., face identity, face expression, and pose) in our experiments. We fitted 198-dim identity coefficients, 29-dim expression coefficients, and 6-dim pose parameters to each frame with a pre-trained 3DMM-CNN [59] model, followed by a face fitting algorithm [60] based on optimization. Human3.6M is a large-scale database containing more than 800 human motion sequences captured by 11 professional actors (3.6 million frames in total) in an indoor environment. For experiments on Human3.6M, we used the raw 2D trajectories of 32 keypoints and further normalized the data into coordinates within the range $[-1, 1]$.

Architecture Design. Our MT-VAE model consists of four components: sequence encoder network, sequence decoder network, latent encoder network, and latent decoder network. We build our sequence encoder and decoder using Long Short-term Memory units (LSTMs) [50]. We used 1-layer LSTM with 1,024 hidden units for both networks. Given past and future motion features extracted from our sequence encoder network, we build three fully-connected layers with skip connections within our latent encoding network. We adopted a similar architecture (three fully-connected layers with skip connections) for our latent decoder network.

Due to the page limit, please see the implementation details in the supplementary document. Also, please see the website for more visualizations: https://goo.gl/2Q69Ym.

4.1   Multimodal Motion Generation

We evaluate our model's capacity to generate diverse and plausible future motion patterns for a given sequence on the Aff-Wild and Human3.6M test sets. Given sequence S_A as initialization, we generated multiple motion trajectories in the future using our proposed sampling and generation process. For the Prediction LSTM model, we only sample one motion trajectory in the future since the predicted future is deterministic.

Quantitative Evaluations. We evaluate our model and baselines quantitatively using the minimum squared error metric and conditional log-likelihood metric, which have been used in evaluating conditional generative models [43,16,53,48]. As defined in Eq. 6, *Reconstruction* minimum squared error (or R-MSE) measures the squared error of the closest reconstruction to ground-truth when sampling latent variables from the recognition model. This is a measure of the quality of reconstruction given both current and

Table 1: Quantitative evaluations for multimodal motion generation. We compare against two simple data-driven baselines for quantitative comparison: *Last-step Motion* that recursively applies the motion (velocity only) from the last step observed; *Sequence Motion* that recursively adds the average sequence velocity from the observed frames.

(a) Results on Aff-Wild with facial expression coefficients.

Method / Metric	R-MSE $\downarrow (\times 10^{-1})$		S-MSE $\downarrow (\times 10^{-1})$		Test CLL $\uparrow (\times 10^3)$
	`train`	`test`	`train`	`test`	
Last-step Motion	—	—	63.8 ± 1.31	74.7 ± 5.59	0.719 ± 0.077
Sequence Motion	—	—	18.4 ± 0.25	19.1 ± 1.02	1.335 ± 0.057
Prediction LSTM [18]	—	—	1.53 ± 0.01	3.03 ± 0.06	2.232 ± 0.003
Vanilla VAE [48]	0.32 ± 0.00	1.28 ± 0.02	0.79 ± 0.00	1.79 ± 0.03	2.749 ± 0.012
Our MT-VAE (concat)	0.22 ± 0.00	0.73 ± 0.01	1.04 ± 0.00	1.76 ± 0.03	2.817 ± 0.023
Our MT-VAE (add)	0.20 ± 0.00	**0.47 ± 0.01**	1.02 ± 0.00	**1.54 ± 0.04**	**3.147 ± 0.018**

(b) Results on Human3.6M with 2D joints.

Method / Metric	R-MSE $\downarrow$		S-MSE $\downarrow$		Test CLL $\uparrow (\times 10^4)$
	`train`	`test`	`train`	`test`	
Last-step Motion	—	—	35.2 ± 0.49	32.1 ± 0.80	0.390 ± 0.004
Sequence Motion	—	—	37.8 ± 0.49	35.2 ± 0.73	0.406 ± 0.003
Prediction LSTM [18]	—	—	1.69 ± 0.02	11.2 ± 0.17	0.602 ± 0.002
Vanilla VAE [48]	0.36 ± 0.00	1.05 ± 0.02	3.18 ± 0.02	3.88 ± 0.05	0.993 ± 0.011
Our MT-VAE (concat)	0.36 ± 0.00	0.97 ± 0.02	2.26 ± 0.03	**2.84 ± 0.05**	1.033 ± 0.010
Our MT-VAE (add)	0.25 ± 0.00	**0.75 ± 0.01**	2.37 ± 0.02	**2.87 ± 0.05**	**1.141 ± 0.009**

future sequences. As defined in Eq. 7, *Sampling* minimum squared error (or S-MSE) measures the squared error of the closest sample to ground-truth when sampling latent variables from prior. This is a measure of how close our samples are to the reference future sequences.

$$\text{R-MSE} = \min_{1 \leq k \leq K} \|S_B - S_B^*(z^{(k)})\|^2, \text{ where } z^{(k)} \sim q_\phi(z|S_A, S_B). \quad (6)$$

$$\text{S-MSE} = \min_{1 \leq k \leq K} \|S_B - S_B^*(z^{(k)})\|^2, \text{ where } z^{(k)} \sim p_\theta(z). \quad (7)$$

In terms of generation diversity and quality, a good generative model is expected to achieve low R-MSE and S-MSE values, given sufficient number of samples. Note that *posterior collapse* issue is usually featured by low S-MSE but high R-MSE, as latent z sampled from the recognition model is being ignored to some extent. In addition, we measure the test conditional log-likelihood of the ground-truth sequences under our model via Parzen window estimation (with a bandwidth determined based on the validation set). We believe that Parzen window estimation is a reasonable approach for our setting as the dimensionality of data (sequence of keypoints) is not too high (unlike in the case of high-resolution videos). For each example, we used 50 samples to compute R-MSE metric, and 500 samples to compute S-MSE and conditional log-likelihood metrics. On Aff-Wild, we evaluate the models on 32-step expression coefficients pre-

diction ($29 \times 32 = 928$ dimensions in total). On Human3.6M, we evaluate the models on 64-step 2D joints prediction ($64 \times 64 = 4096$ dimensions in total). Please note that such measurements are approximate, as we do not evaluate the model performance for every sub-sequence (e.g., essentially, every frame can serve as a starting point). Instead, we repeat the evaluations every 16 frames on Aff-Wild dataset and every 100 frames on Human3.6M dataset.

As we see in Table 1, data-driven approaches that simply repeat the motion computed from last-step velocity or averaged over the observed sequence performed poorly on both datasets. In contrast, the Prediction LSTM [18] baseline greatly reduces the S-MSE metric compared to simple data-driven approaches, due to the deep sequence encoder and decoder architecture in modeling more complex motion dynamics through time. Among all three models using latent variables, our MT-VAE (add) model achieve the best quantitative performance. Compared to MT-VAE (concat) that adopts vector concatenation, our additive version achieves lower reconstruction error with similar sampling eror. This suggests that the MT-VAE (add) model is able to regularize the learning of motion transformation further.

Qualitative Results. We provide qualitative side-by-side comparisons across different models in Figure 4. For Aff-Wild, we render 3D face models using the generated expression-pose parameters along with the original identity parameters. For Human3.6M, we directly visualize the generated 2D keypoints. As shown in the generated sequences, our MT-VAE model is able to generate multiple diverse and plausible sequences in the future. In comparison, the sequences generated by Vanilla VAE are less realistic. For example, given a sitting down motion (lower-left part in Fig. 4) as initialization, the vanilla model fails to predict the motion trend (sitting down), while creating some artifacts (e.g., scale change) in the future prediction. Also note that MT-VAE produces more natural transitions from the last observed frame to the first generated one (see mouth shapes in the face motion examples and distances between two legs in full-body examples). This demonstrates that MT-VAE learns a more robust and structure-preserving representation of motion sequences compared to other baselines.

Crowd-sourced Human Evaluations. We conducted crowd-sourced human evaluations via Amazon Mechanical Turk (AMT) on 50 videos (10 Turkers per video) from Human3.6M dataset. This evaluation presents the past action, and 5 generated future actions for each method to a human evaluator and asks the person to select the most (1) realistic and (2) diverse results. In this evaluation, we also added comparisons to a recently published work [49] on stochastic video prediction, which we refer to as SVG. Table 2 presents the percentage of users who selected each method for each task. The Prediction LSTM produces the most realistic but the least diverse result; Babaeizadeh et al. [48] produces the most diverse but the least realistic result; Our MT-VAE model (we use the additive variant here) achieves a good balance between realism and diversity.

Ablation Study. We analyze variations of our MT-VAE (add) models on Human3.6M. As we see in Table 3, removing the cycle consistency or motion coherence results in a drop in reconstruction performance. This shows that cycle consistency and motion coherence encourage the motion feature to preserve motion structure and hence be more

Fig. 4: Multimodal Sequence Generation. Given an input sequence (green boundary), we generate future sequences (red boundary). We predict 32 frames given 8 frames for face motion, and 64 frames given 16 frames for human body motion. Given the initial frames as condition, we demonstrate (top to bottom) the ground truth sequence, Prediction LSTM, Vanilla VAE, and our MT-VAE model. Overall, our model produces (1) diverse and structured motion patterns and (2) more natural transitions from the last frame observed to the first frame generated (See the subtle mouth shape and scale change from the last observed frame to the first generated one).

Table 2: Crowd-sourced Human Evaluations on Human3.6M. *We did not include Prediction LSTM for the diversity evaluation, as it makes deterministic prediction.

Metric	Vanilla VAE [48]	SVG [49]	Our MT-VAE (add)	Pred LSTM [18]
Realism (%)	19.2	23.8	26.4	30.6
Diversity (%)	51.6	22.3	26.1	0.0*

Table 3: Ablation Study on Different variants of MT-VAE (add) model: We evaluate models trained without motion coherence objective, without cycle consistency objective, and the model with context-free latent decoder.

Method / Metric	R-MSE (test) ↓	S-MSE (test) ↓	Test CLL ↑ ($\times 10^4$)
MT-VAE (add)	0.75 ± 0.01	2.87 ± 0.05	1.141 ± 0.009
MT-VAE (add) w/o Motion Coherence	1.01 ± 0.02	2.93 ± 0.04	1.012 ± 0.014
MT-VAE (add) w/o Cycle Consistency	1.18 ± 0.03	2.71 ± 0.05	0.927 ± 0.019
MT-VAE (add) Context-free Decoder	0.31 ± 0.05	4.05 ± 0.05	1.299 ± 0.007

discriminative in nature. We also evaluate a *context-free* version of the MT-VAE (add) model, where the the transformation vector $\mathcal{T}^*$ is not conditioned on input feature e_A. This version produces poor S-MSE value since it is challenging for the additive latent decoder to hallucinate transformation vector $\mathcal{T}^*$ solely from latent variable z.

4.2 Analogy-based Motion Transfer

We evaluate our model on an additional task of *transfer by analogy*. In this analogy-making experiment, we are given three motion sequences A, B (which is the subsequent motion of A), and C (which is a different motion sequence). The objective is to recognize the *transition* from A to B and transfer it to C. This experiment can demonstrate whether our learned latent space models the mode transition across motion sequences. Moreover, this task has numerous graphics applications like transferring expressions and their styles, video dubbing, gait style transfer, and video-driven animation [22].

In this experiment, we compare Prediction LSTM, Vanilla VAE, and our MT-VAE variants. For the stochastic models, we compute the latent variable z from motion sequence A and B via the latent encoder, i.e., $z = h_{\mathcal{T} \to z}(e_B - e_A)$, and then decode using motion sequence C as $e_D^* = h_{z \to \mathcal{T}}(z, e_C)$. For Prediction LSTM model, we directly performed the analogy-making in the feature space $e_D^* = e_B - e_A + e_C$ since there is no notion of a latent space in that model. As shown in Figure 5, our MT-VAE model is able to combine the transformation learned from A to B transitions with the structure in sequence C. The other baselines failed at either adapting the mode transition from A to B or preserving the structure in C. The analogy-based motion transfer task is significantly more challenging than motion generation, since the combination of three reference motion sequences A, B, and C may never appear in the training data. Yet, our model is able to synthesize realistic motions. Please note that motion modes may not explicitly correspond to semantic motions, as we learn the motion transformation in an unsupervised manner.
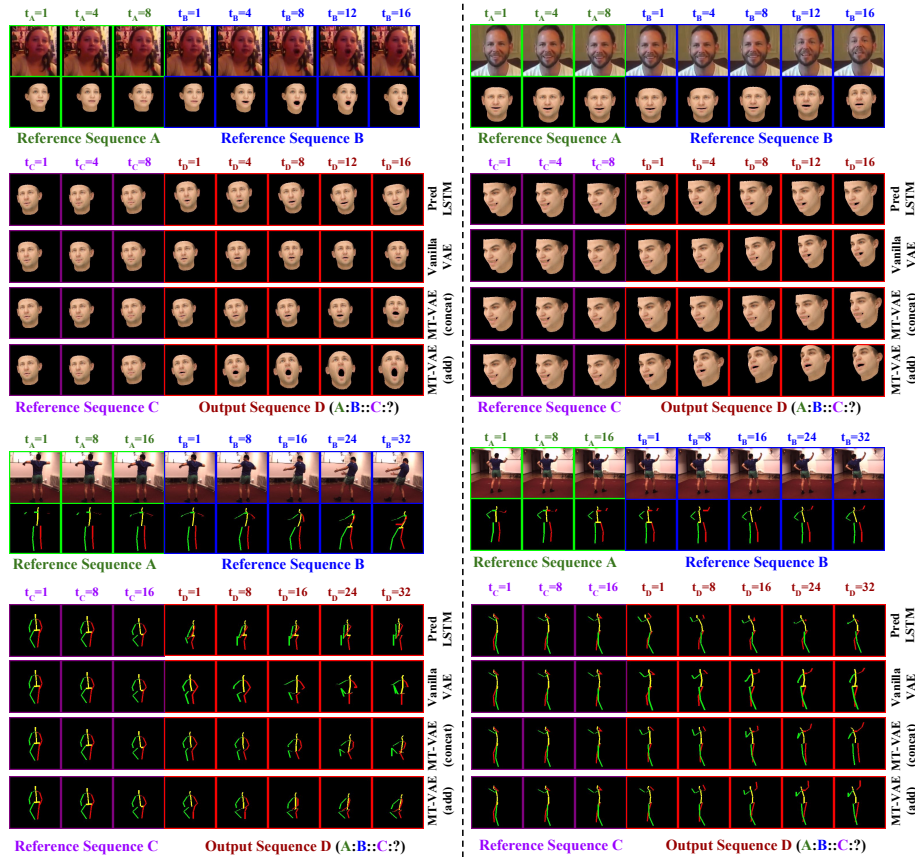
Fig. 5: Analogy-based motion transfer. Given three motion sequences A, B, and C from test set, the objective is to extract the motion mode transition from A to B and then apply it to animate the future starting from sequence C. For fair comparison, we set the encoder Gaussian distribution parameter σ to zero during evaluation.

4.3   Towards Multimodal Hierarchical Video Generation

As an application, we showcase that our multimodal motion generation framework can be directly used for generating diverse and realistic pixel-level video frames in the future. We trained the keypoint-conditioned image generation model [18] that takes both previous image frame A and predicted motion structure B (e.g., rendered face or human joints) as input and hallucinates image C by combining the image content adapted from A but with motion adapted from B. In Figure 6, we show a comparison of video generated in a deterministic way by Prediction LSTM (i.e., single future), and in a stochastic way driven by the predicted motion sequence (i.e., multiple futures) from our MT-VAE (add) model. We use our generated motion sequences for performing video generation experiments on the Aff-Wild (with 8 input frames observed) and Human3.6M (with 16 input frames observed).

Fig. 6: Multimodal Hierarchical video generation. Top rows: Face video generation results from 8 observed frames. Bottom rows: Human video generation results from 16 observed frames.

5   Conclusions

Our goal in this work is to learn a conditional generative model for human motions. This is an extremely challenging problem in the general case and can require significant amount of training data to generate realistic results. Our work demonstrates that this can be accomplished with minimal supervision by enforcing a strong structure on the problem. In particular, we model long-term human dynamics as a set of motion modes with transitions between them, and construct a novel network architecture that strongly regularizes this space and allows for stochastic sampling. We have demonstrated that this same idea can be used to model both facial and full body motion, independent of the representation used (i.e., shape parameters, keypoints).

References

1. de Aguiar, E., Stoll, C., Theobalt, C., Ahmed, N., Seidel, H.P., Thrun, S.: Performance capture from sparse multi-view video. ACM Trans. Graph. **27**(3) (August 2008) 98:1–98:10

2. Beeler, T., Hahn, F., Bradley, D., Bickel, B., Beardsley, P., Gotsman, C., Sumner, R.W., Gross, M.: High-quality passive facial performance capture using anchor frames. ACM Trans. Graph. **30**(4) (July 2011) 75:1–75:10

3. Yang, F., Wang, J., Shechtman, E., Bourdev, L., Metaxas, D.: Expression flow for 3d-aware face component transfer. In: ACM Transactions on Graphics (TOG). Volume 30., ACM (2011) 60

4. Suwajanakorn, S., Seitz, S.M., Kemelmacher-Shlizerman, I.: What makes tom hanks look like tom hanks. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 3952–3960

5. Suwajanakorn, S., Seitz, S.M., Kemelmacher-Shlizerman, I.: Synthesizing obama: learning lip sync from audio. ACM Transactions on Graphics (TOG) **36**(4) (2017) 95

6. Sermanet, P., Lynch, C., Hsu, J., Levine, S.: Time-contrastive networks: Self-supervised learning from multi-view observation. arXiv preprint arXiv:1704.06888 (2017)

7. Rose, C., Guenter, B., Bodenheimer, B., Cohen, M.F.: Efficient generation of motion transitions using spacetime constraints. In: SIGGRAPH. (1996)

8. Bregler, C.: Learning and recognizing human dynamics in video sequences. In: Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on, IEEE (1997) 568–574

9. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: null, IEEE (2003) 726

10. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. IEEE transactions on pattern analysis and machine intelligence **29**(12) (2007) 2247–2253

11. Laptev, I.: On space-time interest points. International journal of computer vision **64**(2-3) (2005) 107–123

12. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE (2011) 3169–3176

13. Wang, J., Liu, Z., Wu, Y., Yuan, J.: Mining actionlet ensemble for action recognition with depth cameras. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 1290–1297

14. Walker, J., Gupta, A., Hebert, M.: Dense optical flow prediction from a static image. In: Computer Vision (ICCV), 2015 IEEE International Conference on, IEEE (2015) 2443–2451

15. Fischer, P., Dosovitskiy, A., Ilg, E., Häusser, P., Hazırbaş, C., Golkov, V., Van der Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. arXiv preprint arXiv:1504.06852 (2015)

16. Walker, J., Doersch, C., Gupta, A., Hebert, M.: An uncertain future: Forecasting from static images using variational autoencoders. In: ECCV. (2016)

17. Chao, Y.W., Yang, J., Price, B., Cohen, S., Deng, J.: Forecasting human dynamics from static images. In: CVPR. (2017)

18. Villegas, R., Yang, J., Zou, Y., Sohn, S., Lin, X., Lee, H.: Learning to generate long-term future via hierarchical prediction. In: ICML. (2017)

19. Walker, J., Marino, K., Gupta, A., Hebert, M.: The pose knows: Video forecasting by generating pose futures. In: 2017 IEEE International Conference on Computer Vision (ICCV), IEEE (2017) 3352–3361

20. Li, Z., Zhou, Y., Xiao, S., He, C., Huang, Z., Li, H.: Auto-conditioned recurrent networks for extended complex human motion synthesis. In: ICLR. (2018)

21. Yang, F., Bourdev, L., Shechtman, E., Wang, J., Metaxas, D.: Facial expression editing in video using a temporally-smooth factorization. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 861–868

22. Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., Nießner, M.: Face2face: Real-time face capture and reenactment of rgb videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 2387–2395

23. Averbuch-Elor, H., Cohen-Or, D., Kopf, J., Cohen, M.F.: Bringing portraits to life. ACM Transactions on Graphics (Proceeding of SIGGRAPH Asia 2017) **36**(6) (2017) 196

24. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: Proceedings of the 26th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co. (1999) 187–194

25. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using lstms. In: International conference on machine learning. (2015) 843–852

26. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. In: ICLR. (2016)

27. Hinton, G.E., Krizhevsky, A., Wang, S.D.: Transforming auto-encoders. In: International Conference on Artificial Neural Networks, Springer (2011) 44–51

28. Oh, J., Guo, X., Lee, H., Lewis, R.L., Singh, S.: Action-conditional video prediction using deep networks in atari games. In: NIPS. (2015)

29. Finn, C., Goodfellow, I.J., Levine, S.: Unsupervised learning for physical interaction through video prediction. In: NIPS. (2016)

30. Yang, J., Reed, S.E., Yang, M.H., Lee, H.: Weakly-supervised disentangling with recurrent transformations for 3d view synthesis. In: Advances in Neural Information Processing Systems. (2015) 1099–1107

31. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction. ICLR **1**(2) (2017) 7

32. Denton, E.L., Birodkar, V.: Unsupervised learning of disentangled representations from video. In: Advances in Neural Information Processing Systems. (2017) 4417–4426

33. Xue, T., Wu, J., Bouman, K., Freeman, B.: Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks. In: NIPS. (2016) 91–99

34. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: NIPS. (2016) 613–621

35. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: Mocogan: Decomposing motion and content for video generation. arXiv preprint arXiv:1707.04993 (2017)

36. Wichers, N., Villegas, R., Erhan, D., Lee, H.: Hierarchical long-term video prediction without supervision. In: ICML

37. Kalchbrenner, N., Oord, A.v.d., Simonyan, K., Danihelka, I., Vinyals, O., Graves, A., Kavukcuoglu, K.: Video pixel networks. arXiv preprint arXiv:1610.00527 (2016)

38. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. (2013) 3111–3119

39. Kulkarni, T.D., Whitney, W.F., Kohli, P., Tenenbaum, J.: Deep convolutional inverse graphics network. In: Advances in Neural Information Processing Systems. (2015) 2539–2547

40. Reed, S.E., Zhang, Y., Zhang, Y., Lee, H.: Deep visual analogy-making. In: Advances in neural information processing systems. (2015) 1252–1260

41. Wang, X., Farhadi, A., Gupta, A.: Actions˜ transformations. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. (2016) 2658–2667

42. Zhou, Y., Berg, T.L.: Learning temporal transformations from time-lapse videos. In: European Conference on Computer Vision, Springer (2016) 262–277

43. Sohn, K., Yan, X., Lee, H.: Learning structured output representation using deep conditional generative models. In: Advances in Neural Information Processing Systems. (2015) 3483–3491

44. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: Advances in Neural Information Processing Systems. (2017) 465–476

45. Ha, D., Eck, D.: A neural representation of sketch drawings. In: ICLR. (2018)

46. Bowman, S.R., Vilnis, L., Vinyals, O., Dai, A.M., Jozefowicz, R., Bengio, S.: Generating sentences from a continuous space. arXiv preprint arXiv:1511.06349 (2015)

47. Hu, Z., Yang, Z., Liang, X., Salakhutdinov, R., Xing, E.P.: Controllable text generation. arXiv preprint arXiv:1703.00955 (2017)

48. Babaeizadeh, M., Finn, C., Erhan, D., Campbell, R.H., Levine, S.: Stochastic variational video prediction. In: ICLR. (2018)

49. Denton, E., Fergus, R.: Stochastic video generation with a learned prior. In: ICML. (2018)

50. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8) (1997) 1735–1780

51. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)

52. Gregor, K., Danihelka, I., Graves, A., Rezende, D., Wierstra, D.: Draw: A recurrent neural network for image generation. In: International Conference on Machine Learning. (2015) 1462–1471

53. Yan, X., Yang, J., Sohn, K., Lee, H.: Attribute2image: Conditional image generation from visual attributes. In: European Conference on Computer Vision, Springer (2016) 776–791

54. Smith, K.A., Vul, E.: Sources of uncertainty in intuitive physics. Topics in cognitive science **5**(1) (2013) 185–199

55. Lan, T., Chen, T.C., Savarese, S.: A hierarchical representation for future action prediction. In: European Conference on Computer Vision, Springer (2014) 689–704

56. Zafeiriou, S., Kollias, D., Nicolaou, M.A., Papaioannou, A., Zhao, G., Kotsia, I.: Aff-wild: Valence and arousal in-the-wild challenge

57. 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) (2014) 1325–1339

58. Paysan, P., Knothe, R., Amberg, B., Romdhani, S., Vetter, T.: A 3d face model for pose and illumination invariant face recognition, Genova, Italy, IEEE (2009)

59. Tran, A.T., Hassner, T., Masi, I., Medioni, G.: Regressing robust and discriminative 3d morphable models with a very deep neural network. In: Computer Vision and Pattern Recognition (CVPR). (2017)

60. Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S.Z.: Face alignment across large poses: A 3d solution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 146–155