Deep Gesture Video Generation with Learning on Regions of Interest

Runpeng Cui, Zhong Cao, Weishen Pan, Changshui Zhang, *Fellow, IEEE*, and Jianqiang Wang

***Abstract*—Generating videos with semantic meaning, such as gestures in sign language, is a challenging problem. The model should not only learn to generate videos with realistic appearance, but also take notice of crucial details in frames to convey precise information. In this paper, we focus on the problem of generating long-term gesture videos containing precise and complete semantic meanings. We develop a novel architecture to learn the temporal and spatial transforms in regions of interest, i.e., gesticulating hands or face in our case. We adopt a hierarchical approach for generating gesture videos, by first making predictions on future pose configurations, and then using the encoder-decoder architecture to synthesize future frames based on the predicted pose structures. We develop the scheme of action progress in our architecture to represent how far the action has been performed during its expected execution, and to instruct our model to synthesize actions with various paces. Our approach is evaluated on two challenging datasets for the task of gesture video generation. Experimental results show that our method can produce gesture videos with more realistic appearance and precise meaning than the state-of-the-art video generation approaches.**

***Index Terms*—video generation, action progress, regions of interest.**

I. Introduction

UNDERSTANDING object motion and human action is one of the core issues in computer vision. Both in video recognition tasks [1], [2], [3], [4], [5] and future video prediction tasks [6], [7], [8], it is crucial to learn how the object and scene transform in videos.

Sign language gesture is regarded as one of the most grammatically structured categories of semantics in videos. This nature of sign language gesture makes it an ideal test bed to develop methods for analysis on video data. In this work, we are interested in the problem of predicting future frames of continuous gestural utterance under conditions of the corresponding gesture labels and an initial frame of the target signer. The presented approach may provide insights into many related fields such as video prediction, motion analysis and human-computer interaction. Further developments in this field could help to facilitate communication between normal and hearing-impaired people.

Early studies on gesture and sign language synthesis [9], [10], [11], [12], [13] use signing avatars to produce signing animation. These synthetic approaches require laborious efforts in motion capture and editing with expert knowledge in sign languages, and the comprehensibility in synthesized animations, however, still suffers from artificial appearance and movement [14]. Recently, deep neural networks have achieved tremendous progress in problems on videos, and researchers have made their efforts on future frame generation with human activity [8], [15], [16], [17], [18]. A widely-used approach in these studies is to learn the general pose-to-image transformation conditioned on the prior scene and action label. For example, Villegas *et al*. [15] develops an analogy-based network, enforcing the representations for image and pose structure to have addition/subtraction relationships in feature space. However, synthesizing video with only coarse skeleton but lack of crucial details is not sufficient for gesture generation, since gesticulating with similar hand motion trajectory but different hand-shapes or gesture evolution can represent totally different meanings. Distinct from previous approaches, our model can learn to synthesize realistic videos with special notice of the regions of interest in frames. Our main contribution is to propose a localized transform module to capture the evolution of regions of interest in our architecture. By focusing on semantic key-points in frames, our approach can generate videos with more precise gestural meanings.

Unlike most periodic actions of human such as walking or eating, the execution of words in sign language is highly structured and always contains a core segment for conveying semantic information. Sign gesture analysis requires observation of multiple components, *e.g*., manual signing, hand movements, body postures, and their precise synchronization [19]. To disambiguate the multimodal synchronization in gestures, we introduce the concept of action progress [20] to explicitly represent the temporal state of gesture for each time step, which enforces our model to capture precise temporal evolution during actions. With the instruction of action progress encoding, our model can readily synthesize actions of various gesticulating paces.

Synthesizing full frames with gesture requires model to figure out how pixels change over time. As observed in [8], [15], pixel evolution is closely related to the movements and deformation of active objects. It would be more tractable to predict the frames with understanding of the active object's

This work was supported by the National Natural Science Foundation of China (Grant No. 61876095, and No. 61751308), Beijing Academy of Artificial Intelligence (BAAI), and the National Science Fund for Distinguished Young Scholars (Grant No. 51625503).

R. Cui and J. Wang are with the State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 100084, China. R. Cui were with the State Key Laboratory of Intelligent Technology and Systems, Department of Automation, Tsinghua University, Beijing 100084, China. (e-mail: {cuirunpeng, wjqlws}@mail.tsinghua.edu.cn)

Z. Cao, W. Pan and C. Zhang are with the Institute for Artificial Intelligence, Tsinghua University (THUAI), Beijing National Research Center for Information Science and Technology (BNRist), State Key Laboratory of Intelligent Technology and Systems, Department of Automation, Tsinghua University, Beijing 100084, China. (e-mail: {caozhong14@mails, pws15@mails, zcs@mail}.tsinghua.edu.cn)

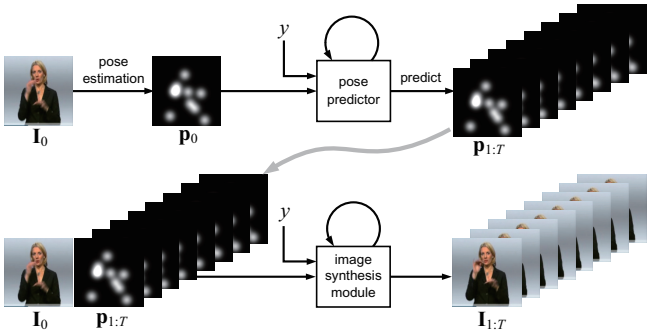

Fig. 1. Overview of our approach to gesture video generation. Given a single reference frame $\mathbf{I}_0$ and the gesture label y as inputs, our algorithm is to synthesize future video $\mathbf{I}_{1:T}$ containing complete gestural meaning. We first generate sequence $\mathbf{p}_{1:T}$ of future locations for the pose landmarks using the pose predictor. Then with the input of preceding frame and target pose configuration, the image synthesis module sequentially generates the target frames. Here pose heatmaps are used for illustration purpose.

structure, which is human skeletal configuration in our case. Therefore, we first develop a pose predictor to generate future human pose configuration. Based on the estimation of high-level pose configuration, we present an encoder-decoder architecture for frame synthesis, and we adopt recurrent neural networks to learn the latent evolution of visual appearance in video sequences.

In this work, we propose a deep neural architecture aiming at synthesizing sign language gesture videos. The main contributions of our work can be summarized as follows:

1) We propose a novel localized transform architecture to represent both spatial transformation and temporal evolution in regions of interest for video generation.
2) We introduce the concept of action progress for video generation to explicitly represent the gesticulating temporal state, and enable our model to synthesize actions of various paces.
3) To the best of our knowledge, we are the first to develop neural model for synthesizing gesture videos with special notice of regions of interest.

We evaluate our method on RWTH-PHOENIX-Weather 2014 [21] and NATOPS datasets [22]. Our approach outperforms the state-of-the-art methods of video generation, yielding synthesized videos with realistic appearance and superiority in conveying semantic information.

The remainder of this paper is organized as follows. Section II reviews related work on video prediction. Section III introduces the formulation of our architecture for gesture video synthesis. Section IV provides implementation details on our model. Section V presents the experimental results of the proposed method and Section VI concludes the paper.

II. Related Work

Future prediction is one of the core problems in computer vision and machine intelligence. Many studies on future prediction are interested in predicting certain representation of future, *e.g.*, object trajectories [23], optical flow [24], motion fields [25], [26], and human body poses [27]. Video prediction can be seen as a specific research field in future prediction, as its main focus is to make predictions on full frames in contrast to specific features of future.

Recent approaches in future prediction have moved from predicting representation of future to generating full frames [6], [7], [8], [28], [29], [30], [31], given the reference frame or video as inputs. Oh *et al*. [28] develop an encoder-decoder network to predict action-conditional frames in Atari games. Mathieu *et al*. [6] propose a multi-scale convolutional network to predict future image from a video sequence, and they adopt adversarial training to preserve the sharpness of the predicted frames. Vondrick *et al*. [7] develop a generative adversarial network (GAN) which untangles the foreground from the background to generate videos from static images. Instead of directly learning the evolution of pixels in videos, some recent work [29], [30] attempts to incorporate motion in frame prediction. Villegas *et al*. [29] build a deep neural network modeling content and motion separately, to better handle the evolution of pixels in videos. Liang *et al*. [30] develop a model that explicitly generates future frames with the predicted optical flows, they adopt adversarial training to make predictions for frames and optical flows more realistic.

Since directly learning the evolution of pixel values is difficult, there is a recent research trend to predict future frames under conditions of object structures, and video prediction with human activity becomes an ideal test bed for these approaches by incorporating landmarks in human pose [8], [15], [16], [17], [18], [32]. Walker *et al*. [8] adopt a variational autoencoder for future pose generation and use both frame and pose sequence as input to predict future scene with human. Villegas *et al*. [15] propose an analogy-based image generator, which learns a joint embedding of image and pose to make future frames in accordance with predictions on human pose. Jang *et al*. [17] propose an adversarial training approach to generate videos based on appearance and motion conditions. These previously mentioned approaches can generate future frames on general constraints such as pose configuration or an action label, but they do not pay special attention to evolution of key regions in detail, which, however, can be crucial to video contents in conveying precise meanings. Our work is distinct from these works in that we propose a localized transform module, in order to capture the dynamics in local regions and to provide instructions on what and where to draw in future frames. As we show in the experimental results, our proposed method can synthesize crucial regions in future frames with more precise appearance and evolution, thus better conveying semantic meanings. Some recent studies have been able to preserve important details such as facial features in synthesized videos. Chan *et al*. [32] employ a specialized generator to add residual to original face region prediction. This specialized model only accounts for increasing visual realism in regions. In contrast, our model not only pays attention to visual realism, but also produces regions of interest with correct shape and evolution according to sign meanings.

View synthesis with human pose [33], [34], [35] is also a research line related to our work. Given a reference image and target pose, their aim is to synthesize static image of the same person according to the target pose. These methods, similar to ours, can generate full frames with the control of pose

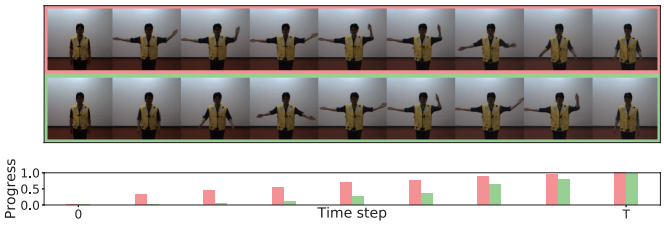

Fig. 2. Synthesized samples given different action progress. These two gesture sequences are generated by our approach with the same reference frame and target label but different action progress. The sequences of action progress to control the generation are plotted below.

configuration, but they usually place more emphasis on visual realism rather than the semantic contents in videos. In contrast, our model not only generates visual details for controlled keypoints in future frames, but also produces samples with more precise semantic meaning.

Some recent great progress has been been made in sequence-to-sequence synthesis. Wang *et al*. [36] study the problem of video-to-video synthesis. They propose a generative adversarial learning framework to capture the mapping function from a source video, *e.g*. semantic segmentation mask sequence, to an output realistic video. Shlizerman *et al*. [37] use long short-term memory (LSTM) networks to transform music signal to body skeleton playing musical instruments. Suwajanakorn *et al*. [38] develop a specific framework to synthesize video of a talking face from audio. Ginosar *et al*. [39] apply GANs to speech-to-gesture translation, their model is able to produce appropriate gesture predictions from audio of speech. These methods solve different applied tasks from ours, and they rely on continuous sequence as inputs to produce translation across different modalities. Our model, in contrast, focuses on the problem of gesture video generation, with no extra continuous inputs but only single reference frame and gesture label incorporated in our approach.

III. Method

Our work tackles the problem of gesture video synthesis. Given a single reference frame $\mathbf{I}_0$ as the input, our model is asked to generate full videos $\mathbf{I}_{1:T}$ containing a complete sign language word utterance. Fig. 1 describes an overview of our approach. The architecture is developed with two components, the future pose predictor and the image synthesis module. Given the initial frame $\mathbf{I}_0$ from video, we obtain the pose configuration $\mathbf{p}_0$ by the pose estimation approach proposed in [40]. Under conditions of the gesture class label y and initial pose configuration $\mathbf{p}_0$, we first generate sequence of future locations $\mathbf{p}_{1:T}$ for the pose landmarks using the pose predictor. Then with the input of previous frame $\mathbf{I}_t$ $(0 \leq t < T)$ and target pose configuration $\mathbf{p}_{t+1}$, the image synthesis module sequentially synthesizes $\mathbf{I}_{t+1}$ at each time step.

A. Encoding for gesticulating states

Different from periodic human actions such as walking and eating, the utterance of sign language words always contains a core segment for conveying semantic information. In our approach, we use the concept of action progress [20] to explicitly encode the temporal ordering. Action progress is interpreted as the fraction of the action that has already passed, with the value ranging from 0 to 1, *e.g*., the progress for the m-th frame from a continuous action segment of length M is $\frac{m}{M}$. Therefore, the whole process of gesture movement can be explicitly represented by a sequence of increasing values of action progress from 0 to 1.

Here we develop a state embedding $\mathbf{s}_t$ to represent the condition of generating gesture y from time step t to $t+1$. Let action progress for time steps t and $t+1$ be a_t and a_{t+1} respectively with $a_{t+1} > a_t$. We first quantize the progress values, and then transform them into progress embedding vectors $\mathbf{a}_t$ and $\mathbf{a}_{t+1}$ using an embedding layer. As for the gesture category ID y, we first convert it to a one-hot vector, and then adopt an embedding layer to produce the representation vector $\mathbf{y}$. We therefore obtain the state embedding $\mathbf{s}_t$ by

$$\mathbf{s}_t = f(\mathbf{a}_t, \mathbf{a}_{t+1}, \mathbf{y}), \tag{1}$$

where f denotes the transform of fully-connected layer, taking $\mathbf{a}_t$, $\mathbf{a}_{t+1}$ and $\mathbf{y}$ as the inputs. The state embedding $\mathbf{s}_t$ supplements essential action and progress information to facilitate learning on the temporal evolution.

At the training stage, we randomly sample T frames from the T'-frame original gesture segments for training. Let the indices of sampled frames be $n_1, \cdots, n_T$ $(1 \leq n_1 < \cdots < n_t \leq T')$. We then have the progress $a_t = \frac{n_t}{T'}$ for time step t. At the inference stage, we can assign an arbitrary sequence of monotonically increasing values in range $[0, 1]$ as action progress values. Fig. 2 presents two instances synthesized by our model with different sequences of provided progress values. The result indicates that our model has successfully capture the relationship between temporal advancements of action and progress values. With action progress explicitly representing temporal evolutions, our model can produce generations for target gestures with various gesticulating paces and video lengths.

We note that recent study on video generation in [31] introduces a variable named "time counter" to provide temporal hints for generation, which is similar to "action progress" proposed in this manuscript. Wang *et al*. [31] combine time counter and encoding of targeted end-frame to enforce model to be aware of the temporal progress to the end. The proposed time counter is applied to incorporate with single modality, *i.e*. the encodings of frame only. In our problem, since the evolution of gesture video is highly related to the gestural meaning to be conveyed, we use action progress variable together with gesture label, instead of frame encodings, to build up the state embedding for temporal hints. We demonstrate that our state embedding can provide shared hints for multiple modalities including pose, RoIs and other regions, and facilitate the synchronization among them.

B. Future pose prediction

This section describes the future pose predictor developed in our work. Given the initial pose configuration $\mathbf{p}_0$, our pose predictor generates the sequence of future poses $\mathbf{p}_{1:T}$

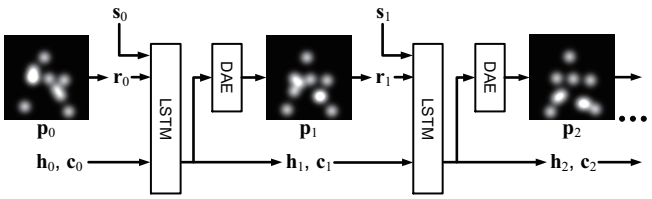

Fig. 3. Overview of our pose prediction pipeline. The LSTM module takes the observed pose $\mathbf{p}_t$ and state embedding $\mathbf{s}_t$ as inputs to make a coarse prediction on pose. We then use DAE to further refine the pose prediction $\mathbf{p}_{t+1}$. In this figure, we use the heatmaps of pose only for illustration purpose, while in the implementation of our method, both the input and predicted pose configurations of the pose predictor are represented by 2D coordinates.

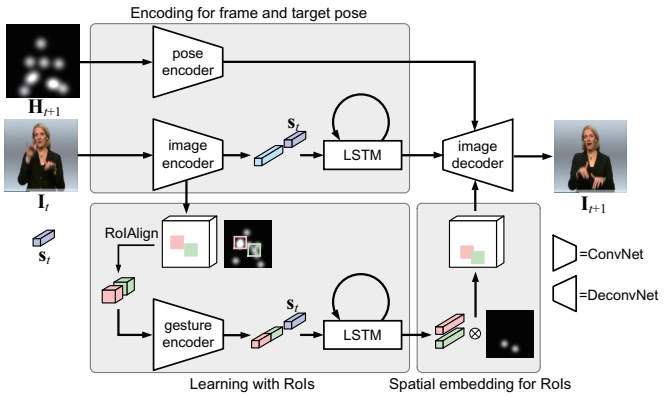

Fig. 4. Illustration of our image synthesis module. The module takes the current frame $\mathbf{I}_t$ and target pose heatmap $\mathbf{H}_{t+1}$ as inputs to generate subsequent frame $\mathbf{I}_{t+1}$. This generation process is conditioned on the state embedding $\mathbf{s}_t$ which represents the gesture label and action progress.

under conditions of the gesture class label y. Fig. 3 describes the outline of our pose predictor, which we build based on architecture developed in [41]. To predict the locations of joints $\mathbf{p}_{t+1}$ at time $t+1$, our model takes the locations $\mathbf{p}_t$ of J joints at time t as the network inputs. Here $\mathbf{p}_t \in \mathbb{R}^{2\times J}$ is composed of 2D coordinates. We first transfer pose coordinates into representation $\mathbf{r}_t$ with a fully-connected layer, and obtain the state embedding $\mathbf{s}_t$ representing gesture label and current action progress. We feed the concatenation of both representations at time t into LSTM module [42] by

$$\mathbf{h}_{t+1}, \mathbf{c}_{t+1} = \texttt{LSTM}(\text{concat}(\mathbf{r}_t, \mathbf{s}_t), \mathbf{h}_t, \mathbf{c}_t), \tag{2}$$

where $\text{concat}(\cdot)$ denotes the concatenation operation. Then we predict the locations of joints at time $t+1$ by $\hat{\mathbf{p}}_{t+1} = \texttt{FC}(\mathbf{h}_{t+1})$, where $\texttt{FC}(\cdot)$ represents the nonlinear transformations of a fully-connected layer. To further refine the pose configuration in consistency with spatial dependencies, we import a dropout autoencoder (DAE) as in [41] to improve the predicted joint locations by $\mathbf{p}_{t+1} = f_{\text{DAE}}(\hat{\mathbf{p}}_{t+1})$. Our pose predictor only takes $\mathbf{p}_0$ from initial frame as the reference pose, and predicts the sequence of future poses $\mathbf{p}_{1:T}$ recursively.

In order to represent joint coordinates as spatial features for subsequent synthesis process, we encode pose $\mathbf{p}_{1:T}$ into heatmaps $\mathbf{H}_{1:T}$. Given the 2D coordinates $\mathbf{p}_t^{(j)}$ for the j-th joint at time t, the value at location $\mathbf{l}$ in the j-th channel of $\mathbf{H}_t$ is defined as

$$\mathbf{H}_t^{(j)}(\mathbf{l}) = \exp\left(-\frac{\|\mathbf{l} - \mathbf{p}_t^{(j)}\|_2^2}{\sigma^2}\right), \ j = 1, 2, \cdots, J, \tag{3}$$

where $\mathbf{l} \in \{(x, y) \mid 0 \le x < W, \ 0 \le y < H\}$, and σ controls the variance in heatmaps. Here each channel of $\mathbf{H}_t$ is $W \times H$ pixels, the same size as the video frames.

C. Image synthesis with regions of interest

Our image synthesis module can be generally regarded as an encoder-decoder structure, where our encoder transforms current full frame $\mathbf{I}_t$ and predicted future pose heatmap $\mathbf{H}_{t+1}$ into feature spaces, and decoder learns to synthesize future frame $\mathbf{I}_{t+1}$ in pixel-level with the encoded representations.

Fig. 4 illustrates our image synthesis module. Our model uses an image encoder f_{img} and a pose encoder f_{pose} to process the input frame $\mathbf{I}_t \in \mathbb{R}^{W\times H\times 3}$ and pose heatmaps $\mathbf{H}_{t+1} \in \mathbb{R}^{W\times H\times J}$ respectively. We develop a gesture encoder module f_{gest} which attends to feature maps produced by f_{img}, in order to capture local representation for gesticulating hands or face. Subsequently, LSTMs are trained with representations for full image and regions of interest (RoIs) respectively, in order to learn the appearance evolution conditioned on action state over time. We propose the spatial embedding approach to explicitly encode RoI representations to target locations, telling the decoder where to "draw" for RoIs. Finally, given the representations for target pose and transformed features for both full image and RoIs, we develop the image decoder f_{dec} to learn the feature-to-image transformation and synthesize future frame $\mathbf{I}_{t+1}$. The whole process can run recurrently to generate full videos, conditioned on the gesture label y and action progress represented by the state embedding.

1) Encoding for frame and target pose: Video synthesis with provided pose configurations is not only about learning the skeleton-to-image transformation, temporal evolution should also be considered. Here we employ recurrent neural networks (RNNs) to capture the dependencies over time.

Given the input frame $\mathbf{I}_t$, we have $\mathbf{f}_t = f_{\text{img}}(\mathbf{I}_t)$, where $\mathbf{f}_t$ denotes the output representation produced by frame encoder at time step t. We subsequently use LSTM to transform the frame encoding by

$$\mathbf{h}_{t+1}^{\text{img}}, \mathbf{c}_{t+1}^{\text{img}} = \texttt{LSTM}(\text{concat}(\mathbf{f}_t, \mathbf{s}_t), \mathbf{h}_t^{\text{img}}, \mathbf{c}_t^{\text{img}}). \tag{4}$$

We then obtain the transformed frame encoding by $\hat{\mathbf{f}}_{t+1} = \texttt{FC}(\mathbf{h}_{t+1}^{\text{img}})$. The transformed representation $\hat{\mathbf{f}}_{t+1}$ is further fed into the decoder f_{dec} for image synthesis.

As for the target pose, we feed its heatmap $\mathbf{H}_{t+1}$ to the convolutional pose encoder f_{pose} to obtain its encoding, which is further fed into the decoder to enforce the spatial constraints of target pose.

2) Learning with regions of interest: Evolution of gesticulating hands usually plays an important role in conveying information in gestures. Utterances with similar hand motion trajectory but different hand-shapes or gesture transforms can represent different meanings. However, most of previous works in synthesizing human activity in videos are unable to generate future frames with details in controlled regions, and

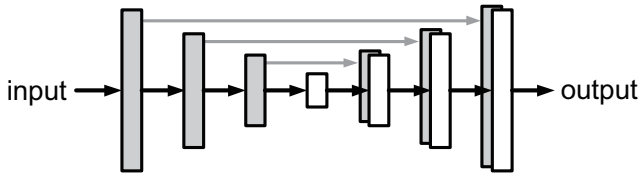

Fig. 5. Illustration of an example for U-Net architecture, where skip connections are built by concatenation in channels from layers in encoder (colored in gray) to corresponding ones in decoder.

thus failing to generate decent gesture videos with sufficient details in gesticulating parts.

Different from previous works in video prediction, we propose a localized module to capture the transform in regions of interest, *i.e*., the gesticulating hands or face. The RoI centers are all included in the J predicted joints. We denote the N RoI centers as $\{\mathbf{q}_t^{(n)}\}_{n=1}^N$. We have $\{\mathbf{q}_t^{(n)}\}_{n=1}^N \subseteq \{\mathbf{p}_t^{(j)}\}_{j=1}^J$ and $N = 2$ for both hands, or $N = 3$ for hands and face in our case.

Given the locations of RoIs, we use RoIAlign [43] to extract local representations of fixed spatial size for RoIs from intermediate feature map of f_{img}. Afterwards, we apply a convolutional gesture encoder f_{gest} to transform the local feature maps respectively and concatenate them to get gestural representation $\mathbf{g}_t$. To capture the temporal transform for gestural representation, we adopt LSTM module as

$$\mathbf{h}_{t+1}^{\text{gest}}, \mathbf{c}_{t+1}^{\text{gest}} = \texttt{LSTM}(\text{concat}(\mathbf{g}_t, \mathbf{s}_t), \mathbf{h}_t^{\text{gest}}, \mathbf{c}_t^{\text{gest}}). \tag{5}$$

We then have $\hat{\mathbf{g}}_{t+1} = \texttt{FC}(\mathbf{h}_{t+1}^{\text{gest}})$ as the transformed gestural representation, which is partitioned to get the separated appearance encodings $\{\hat{\mathbf{g}}_{t+1}^{(n)}\}_{n=1}^N$ for each RoI to generate.

3) Spatial embedding for RoIs: We propose the spatial embedding method to build spatial feature maps for RoI synthesis, with transformed representations for RoIs explicitly encoded to the predicted locations. The embedded spatial representations tell the decoder what and where to "draw" for RoIs.

We first encode the predicted RoI centers $\{\mathbf{q}_{t+1}^{(n)}\}_{n=1}^N$ into N spatial masks $\{\mathbf{M}_{t+1}^{(n)}\}_{n=1}^N$. Each mask is a $w \times h \times d$ tensor, which is obtained by resizing the corresponding heatmap $\mathbf{H}_{t+1}^{(j)}$ to the size of $w \times h$ and then being replicated on depth axis by d times. Here d is the same dimension as $\hat{\mathbf{g}}_{t+1}^{(n)}$. For each RoI, we multiply the transformed feature $\hat{\mathbf{g}}_{t+1}^{(n)}$ pixel-wise with the spatial mask $\mathbf{M}_{t+1}^{(n)}$ to obtain a masked feature map $\mathbf{G}_{t+1}^{(n)}$, which is a $w \times h \times d$ tensor with RoI representation embedded at the predicted location. We then use element-wise addition on these masked feature maps $\{\mathbf{G}_{t+1}^{(n)}\}_{n=1}^N$ to obtain the fused feature map $\mathbf{G}_{t+1}$, which is subsequently fed to the image decoder f_{dec}.

4) Encoder-decoder architecture: We adopt U-Net-style model [44], [45] as the general architecture of the encoder-decoder module. U-Net architecture (see Fig. 5) generally adds skip connections from layers of encoder to those of decoder by concatenation in channels. It allows low-level information to have shortcut across the architecture, and has shown great success in image synthesis [35], [44].

In our architecture, the decoder incorporates not only the feature maps extracted from full frame by f_{img}, but also the representations for evolving general pose configuration by f_{pose}, and transformed representations for RoIs. These input representations provide the decoder with information of frame appearance, target pose and RoIs. We implement this fusion by using concatenation in channels.

D. Training and inference

1) Objective function: Pixel-wise loss such as mean square error or ℓ_1-loss is one of the most commonly adopted objectives [7], [8], [16] for model optimization in video synthesis. However, as observed in [35], [46], using pixel-wise loss between synthesized and target image usually gives rise to blurry results with little high frequency content. To alleviate this problem, we develop feature loss as

$$\mathcal{L}_{\text{feat}} = \frac{1}{T}\sum_{t=1}^{T}\sum_{\phi\in\mathbf{\Phi}} \|\phi(\mathbf{I}_t) - \phi(\mathbf{I}_t^*)\|_1, \tag{6}$$

where $\mathbf{I}_t^*$ is the target frame at time t, and each element of set $\mathbf{\Phi}$ represents a mapping defined by stacked layers from pre-trained neural networks, transforming input image into feature map. In our experiments, we develop $\mathcal{L}_{\text{feat}}$ by calculating error in all feature maps produced by the first 12 convolutional layers in VGG19 network [47], from layer "conv1_1" to "conv4_4" respectively. The adopted VGG19 net pre-trained on large-scale image classification task [48] can capture wide range of visual patterns, and minimization on feature loss will facilitate generation of images with more precise visual contents.

We also adopt a weighted feature objective $\mathcal{L}_{\text{RoI}}$ based on $\mathcal{L}_{\text{feat}}$ for the architecture to better learn the contents in RoIs. Let $(W/s)\times(H/s)\times D$ be the size of error map $\phi(\mathbf{I}_t)-\phi(\mathbf{I}_t^*)$, with s as the scale and D as the number of channels. Error at location $\mathbf{l}$ is weighted by

$$w_t^{\phi}(\mathbf{l}) = \sum_{n=1}^{N} \exp\left(-\frac{\|\mathbf{l} - \frac{1}{s}\mathbf{q}_t^{(n)}\|_2^2}{(\hat{\sigma}/s)^2}\right), \tag{7}$$

where $\{\mathbf{q}_t^{(n)}\}_{n=1}^N$ are the centers of RoIs as defined before, and $\hat{\sigma}$ controls the variance. Given the weight tensor $\mathbf{W}_t^{\phi}$ with each element defined in (7), we denote $\mathcal{L}_{\text{RoI}}$ as

$$\mathcal{L}_{\text{RoI}} = \frac{1}{T}\sum_{t=1}^{T}\sum_{\phi\in\mathbf{\Phi}} \|\mathbf{W}_t^{\phi} \odot (\phi(\mathbf{I}_t) - \phi(\mathbf{I}_t^*))\|_1, \tag{8}$$

where $\odot$ represents element-wise multiplication. We set error near the RoIs with larger weight value for optimization.

The objective function for training is finally defined as

$$\mathcal{L} = \mathcal{L}_{\text{feat}} + \lambda\mathcal{L}_{\text{RoI}}, \tag{9}$$

where λ controls the weight for optimizing the content in RoIs. We adopt $\mathcal{L}_{\text{feat}}$ to enforce the model to capture various patterns in full frames, and we add $\mathcal{L}_{\text{RoI}}$ to encourage our model to pay more attention to the RoIs. Fig. 6 shows the example of our model using different objectives for optimization. We find that our training objective can encourage the synthesis module to generate richer visual details, especially for the gesticulating hands.

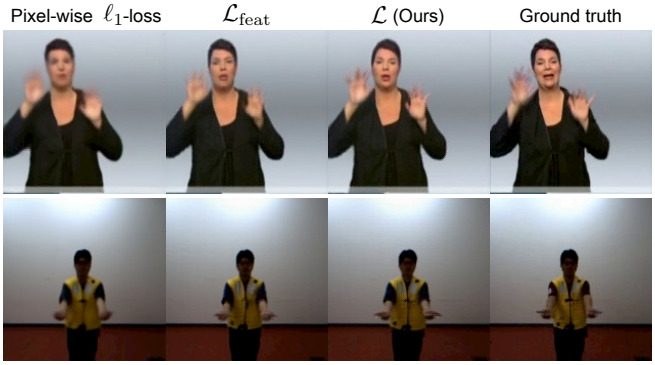

Fig. 6. Output frame examples synthesized by models minimizing different loss functions. Given the same pose configuration and target label, these frames are generated by model trained with pixel-wise ℓ_1-loss, $\mathcal{L}_{\text{feat}}$ and the loss adopted by our model $\mathcal{L}$, respectively. Our model trained with designed loss $\mathcal{L}$ can produce frames with the most realistic appearance, including additional detail to regions of hands.

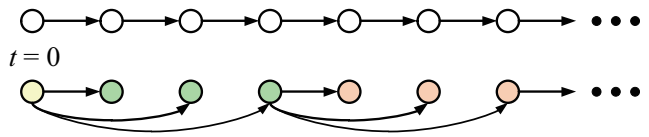

Fig. 7. Illustration of the approach to predicting a long sequence. In contrast to generating from last frame at each time step (top), we partition the generation process into segments, and our model takes the last frame of previous segment as the reference image (bottom). Segments are denoted with different colors in this figure.

2) Inference: To make long-term predictions on more time steps during inference, we first assign a sequence of monotonically increasing progress values to control the process of generation. We use the pose predictor to sequentially generate the pose configurations for each time step.

Since our image synthesis module generates frames recurrently by observing preceding predicted frame, inference with over-length recurrent steps can give rise to the problem of error accumulation [15]. To tackle this problem, we develop an approach to generate the target videos segment-by-segment. Fig. 7 illustrates the approach that we apply for the generation task at the inference stage. We linearly partition the gesture execution into segments, and our model sequentially generates each segment to build up a complete action. For each video segment to generate, our model takes the last frame of previous segment as the observed image, and then makes one-step inferences for all frames in the current segment, with the same cells and hidden states for LSTMs. In contrast to making predictions frame-by-frame, we synthesize full videos with fewer recurrent steps, with each step observing input frame of less quality decline. This generation process can notably reduce the quality deterioration over long-term generation, and give rise to frames with realistic continuous evolution in appearance, since our model has been taught to deal with various temporal states during training.

IV. Model Implementation

In this section, we provide more implementation details of our approach.

A. Model Design

Our model for gesture video synthesis is developed with two components, the future pose predictor and the image synthesis module. Let `Ck` denote a convolutional layer with k filters of size 3×3 and stride 1, `CTk` be a transposed convolutional layer with k filters, `Pool` denote a max-pooling layer with pooling stride 2, `FCk` represent a fully-connected layer with output dimension k, and `LSTMk` denote the LSTM architecture with k hidden units. For both components we use $\mathrm{ReLU}(\cdot)$ for the nonlinearity in the architecture, except for the last layer where $\tanh(\cdot)$ is used, scaling the output into range $[-1, 1]$.

1) Pose predictor: The pose predictor is composed of recurrent model and a dropout autoencoder, as instructed in [41]. The recurrent model is developed as `FC256-LSTM256-LSTM256-FC20` to predict the coarse pose configuration. The dropout autoencoder is built as `FC256-FC256-FC256-FC20` to refine the prediction [41].

2) Image synthesis module: The image synthesis module consists of encoders, decoder, and RNNs for full frame and RoIs respectively.

The image encoder f_{img} is taking the VGG16 net [47] up to layer "pool4" as the base convolutional layers, which we denote as `ConvNet`. We then build the image encoder as `ConvNet-C512-Pool-FC4096-FC1024`, which transforms input image of 160×160 pixels to 1024-dimensional features. The pose encoder f_{pose} employs the VGG16 net up to layer "pool4", transforming input heatmap to 512-dimensional feature maps of size 10×10. Note that the input channel of inputs for pose encoder is 10, with each channel corresponding to the heatmap for one pose landmark. The RoI encoder is developed as `C256- C512-RoIAlign-FC512`, where we apply RoIAlign [43] approach to extract feature maps of a fixed size 5×5 from the locations of RoIs. Our RoI encoder takes the output feature map from "pool3" layer of image encoder as the inputs.

The recurrent modules for full frame and RoIs are built as `LSTM512-FCk`, where $k = 512$ for image representations and $k = 2 \times 512$ for RoIs.

The decoder f_{dec} is built as `CT512-CT512-CT512-CT256-CT128-CT64-C3`, where the first transposed convolutional layer is with 5×5 kernels and stride 1, and the others with 4×4 kernels and stride 2. We adopt U-Net architecture [45] as the general encoder-decoder structure, where we add skip connections from the output feature maps of i-th max-pooling layer in f_{img} to the input tensor of $(7-i)$-th layer in f_{dec} $(i = 1, 2, 3, 4)$. The output of f_{pose} and spatial representations of RoIs are also concatenated to the input of the third transposed convolutional layer in f_{dec}.

B. Implementation Details

The pose predictor and image synthesis module in our framework are trained separately. For training the pose predictor, we use pose sequences around 15 in length as training examples. We select 10 key-points (nose, neck, L/R-shoulders, L/R-elbows, L/R-wrists, L/R-hands) estimated by approach in [40] to represent the pose configuration, from which we take

two gesticulating hands as RoIs for both datasets. Locations for all joints are rescaled into range $[-1, 1]$.

For training our image synthesis module, we randomly sample 5 frames from original action clip for each example. Note that we use state embedding to represent the temporal information for actions at each time step, which enforces our model to capture diverse temporal dependencies. To increase the variability of inputs, we add intensity noises to the input frames, and randomly jitter the height and width by $\pm 10\%$. All the input images are transformed to the size of 160×160 pixels and RGB values are rescaled into range $[-1, 1]$. The weights in image and pose encoders in our architecture are initialized from VGG16 model [47] pre-trained on ISLVRC-2014 dataset [48].

We use Adam optimizer [49] for the training of our model with a learning rate of 5×10^{-5}. We set $\sigma = 4$ for the pose heatmap encoding, $\hat{\sigma} = 15$ for weighting on RoIs and $\lambda = 5.0$ in experiments on both datasets.

V. Experiments

This section reports experiments performed on two public datasets for gesture and sign language to validate our approach. In Section V-A and V-B, we introduce the datasets and the experimental protocol that we follow in the experiments. In the remainder of this section, we present and analyze our experimental results on gesture datasets, and we finally compare the performance of our generative approach with the state-of-the-arts.

A. Datasets

We evaluate our approach on RWTH-PHOENIX-Weather 2014 dataset [21] and NATOPS human action dataset [22].

RWTH-PHOENIX-Weather 2014 (Phoenix-2014) dataset is collected from TV broadcast of weather forecasts, containing 5,672 videos of German sign language sentences performed by 9 signers. Each sentence in Phoenix-2014 is annotated with a sequence of sign captions in correct order. To generate the mapping between video segments and gesture tokens, we adopt the alignment approach developed in [50]. For each sign language sentence, Cui *et al*. [50] use the pre-trained model for continuous sign language recognition to calculate the probabilities of all possible alignments that match the ground truth label sequence, and then allocate each gesture label to video segment, according to the alignment with maximal probability. This procedure produces the most probable label-to-segment mapping inferred by the pre-trained recognition model.

To ensure enough training samples for each gesture category, we only select those categories with over 280 utterances for our synthetic task. The number of selected categories is 56 in total. The training and validation sets are partitioned the same way as the original database.

The NATOPS database contains 9,600 video segments performing 24 categories of aircraft handling gestures. These videos are performed by 20 signers with each action repeated for 20 times. We use videos from 18 signers for training and two unseen signers for test.

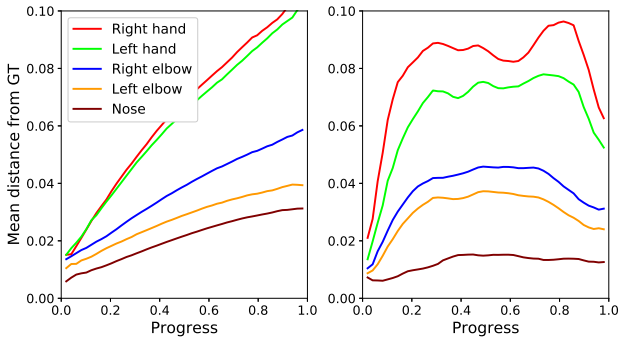

Fig. 8. Mean Euclidean distance from ground truth for pose landmarks over action progress on Phoenix-2014 (left) and NATOPS (right). The distances are represented as the ratios with respect to the image size.

B. Evaluation

To evaluate the generation results quantitatively, we employ inception score [51] and Fréchet video distance [52] as criteria. Inception score (IS) is a widely-used criterion for evaluating results for video generation [8], [16]. Let $\mathbf{z}$ be the video sample and y denote the action label of the generated video, inception score of a generative module G is calculated by

$$\mathrm{IS}(G) = \exp\left(\mathbb{E}_{\mathbf{z}\sim p_G} D_{\mathrm{KL}}(p(y|\mathbf{z}) \parallel p(y))\right), \tag{10}$$

where $\mathbf{z} \sim p_G$ indicates that $\mathbf{z}$ is a video sampled from p_G, $p(y|\mathbf{z})$ is the conditional label distribution, $p(y) = \int p(y|\mathbf{z}) p_G(\mathbf{z}) \mathrm{d}\mathbf{z}$ is the marginal label distribution, and $D_{\mathrm{KL}}(p\|q)$ is the Kullback-Leibler divergence between the distributions p and q. High inception score on one hand encourages the model to generate samples with clear semantic information, yielding low entropy for $p(y|\mathbf{z})$. On the other hand, high inception score indicates that the generative model is generating samples of high diversity, resulting in $p(y)$ of high entropy.

Fréchet video distance (FVD) [52] is developed to calculate the distance between the real world data distribution p_R and the distribution p_G defined by the generative model. It is given as

$$|\mu_R - \mu_G|^2 + \mathrm{Tr}\left(\mathbf{\Sigma}_R + \mathbf{\Sigma}_G - 2(\mathbf{\Sigma}_R \mathbf{\Sigma}_G)^{1/2}\right), \tag{11}$$

where μ_G and μ_R are the means and $\mathbf{\Sigma}_G$ and $\mathbf{\Sigma}_R$ are the co-variance matrices of distribution p_G and p_R, respectively.

In our experiments, we train the gesture classifier separately based on the spatio-temporal feature learning model in [50] to calculate inception score on full frames. We use I3D networks pre-trained on Kinetics-400 dataset [53] to calculate Fréchet video distance. To evaluate whether the synthesized videos can represent the given gesture labels correctly, we also apply the pre-trained classifier to the task of classification on these synthesized samples, and we use the classification accuracy as one criterion to evaluate these generative approaches.

C. Pose prediction results

Most pose-conditional generative models are trained to learn the pose-to-image transformation, and they rely on the skeletal configuration for video generation task. Therefore, the

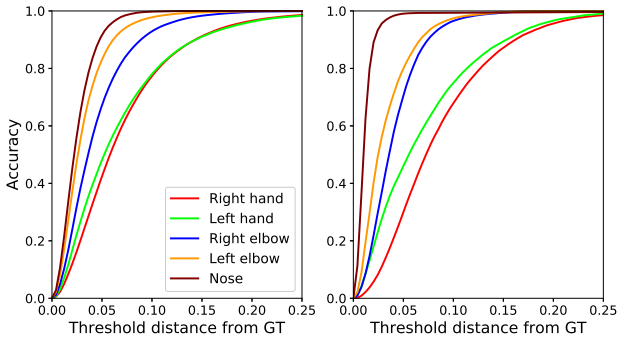

Fig. 9. Pose prediction accuracy as the threshold distance from ground truth is increased on Phoenix-2014 (left) and NATOPS (right). The threshold distances are represented as the ratios with respect to the image size.

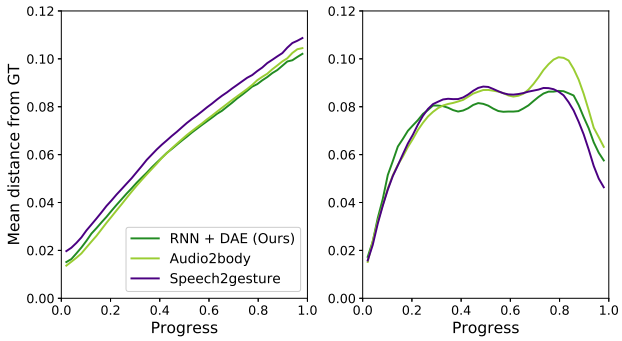

Fig. 10. Mean Euclidean distance from ground truth for predicted locations of hands over action progress on Phoenix-2014 (left) and NATOPS (right). The distances are represented as the ratios with respect to the image size.

prediction of future pose is crucial to the final synthesized result. Here we evaluate the performance of the pose predictor on Phoenix-2014 and NATOPS databases. Note that there can be various possible evolutions for pose even with the same target action and initial pose configuration.

Fig. 8 presents the quantitative evaluation of mean distance from the ground truth over time for the predicted pose landmarks. Notice that the error of predicted location on two databases evolves differently over action progress. This can be explained by the fact that gestures from Phoenix-2014 database are cropped from continuous sign language sentences and the ending postures can vary due to the succeeding gestures, while all gesticulating executions collected in NATOPS dataset end with a fixed pose. Fig. 9 shows the accuracy of the prediction for key-points as the threshold deviation from ground truth is increased. We can observe that our pose predictor can provide fine estimates on future pose landmarks. For example, the predicted locations for L/R-hands have the accuracy of 96.3% and 96.7% on Phoenix-2014 database, and 96.9% and 95.9% on NATOPS database, given the deviation threshold of $0.20H$ from the ground truth. The average accuracy over all joints under threshold of $0.20H$ is 98.9% and 98.8% on Phoenix-2014 and NATOPS respectively. Synthesized gesture video with predicted pose also shows similar performance compared to those with ground truth pose (*e.g.*, "GT pose" vs. "predicted pose" in Table I and Table II), indicating that our pose predictor can generate skeletal actions with interpretable meanings.

To further illustrate the effectiveness of our pose predictor, we compare it with state-of-the-art pose generative models, Audio2body [37] and Speech2gesture [39], on our challenging corpus. In order to adapt these approaches to our synthesis task without audio inputs, we substitute the audio features in their work with our temporal state embedding at each time-step. Fig. 10 shows the mean distance from the ground truth over time for the predicted locations of hands on Phoenix-2014 and NATOPS. We find that our pose predictor shows competitive performance on both databases.

D. Ablation study

To evaluate the effect of each individual component in the proposed approach, we carry on ablation study to look into the ingredients of our architecture. We take the fully developed architecture as the baseline approach, and alter a single component for each experiment to evaluate its impact. Table I and II present the experimental results of the ablation study, where "w/o seq." represents removing LSTMs for the latent transform from the baseline approach, "w/o progress" represents method removing action progress embedding, "w/o label" represents removing target gesture label for generation, "w/o $\mathcal{L}_{\text{RoI}}$" represents removing $\mathcal{L}_{\text{RoI}}$ from the training objective in (9), "Pixel-wise ℓ_1-loss" represents replacing feature loss to pixel-wise ℓ_1-loss in RGB space, and "w/o RoI mod." represents removing the localized transform module and spatial embedding of RoIs.

With regard to temporal modeling, both action progress encoding and sequential learning module contribute to the final

TABLE I
ABLATION STUDY RESULTS ON PHOENIX-2014 DATASET

model	with ground truth pose			with predicted pose		
	IS	FVD	acc.	IS	FVD	acc.
w/o seq.	27.0	131.0	0.495	25.2	301.7	0.435
w/o progress	34.5	83.0	0.626	31.7	184.0	0.561
w/o label	30.6	84.3	0.490	26.9	175.2	0.395
w/o $\mathcal{L}_{\text{RoI}}$	26.6	99.5	0.484	22.5	176.9	0.444
Pixel-wise ℓ_1-loss	18.4	236.1	0.385	15.0	312.9	0.328
w/o RoI mod.	30.4	183.6	0.490	26.0	280.6	0.354
Ours	**37.4**	**81.0**	**0.713**	**36.1**	**164.2**	**0.699**

TABLE II
ABLATION STUDY RESULTS ON NATOPS DATASET

model	with ground truth pose			with predicted pose		
	IS	FVD	acc.	IS	FVD	acc.
w/o seq.	23.2	147.2	0.982	23.3	316.8	0.978
w/o progress	23.1	148.9	0.979	23.7	301.7	0.984
w/o label	21.4	136.2	0.897	20.3	296.7	0.821
w/o $\mathcal{L}_{\text{RoI}}$	23.2	142.6	0.976	22.9	317.8	0.944
Pixel-wise ℓ_1-loss	21.8	300.1	0.933	22.7	416.5	0.965
w/o RoI mod.	23.5	164.1	0.987	23.6	302.2	0.974
Ours	**23.6**	**133.0**	**0.991**	**23.7**	**299.8**	**0.985**

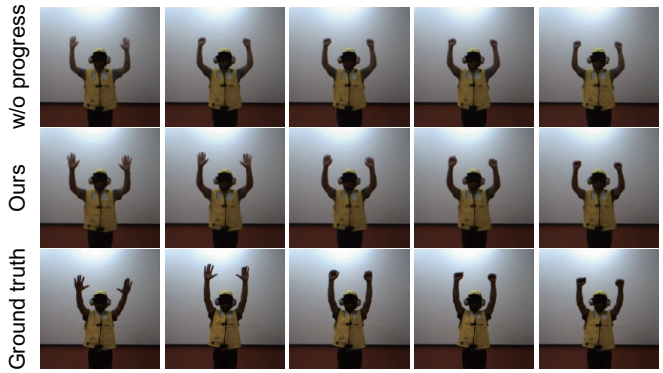

Fig. 11. Output frame examples by approaches with and without action progress encoding, along with the ground truth frames below them. Synthesized frames are taken from the gesture "BRAKES_ON" with action progress from 0.3 to 0.7. The method "w/o progress" denotes the image synthesis model with action progress encoding removed, and "Ours" represents our full synthetic model. Note that there is natural evolution in hand-shape upon the addition of action progress encoding.

TABLE III
EXPERIMENT RESULTS OF ROI TRANSFORM MODULE ON PHOENIX-2014 DATASET

model	with ground truth poses			with predicted poses		
	IS	FVD	acc.	IS	FVD	acc.
w/o RoI mod.	30.4	183.6	0.490	26.0	280.6	0.354
Ours (hands)	37.4	81.0	0.713	36.1	164.2	0.699
Ours (hands + face)	**40.9**	**66.1**	**0.827**	**39.2**	**135.4**	**0.821**

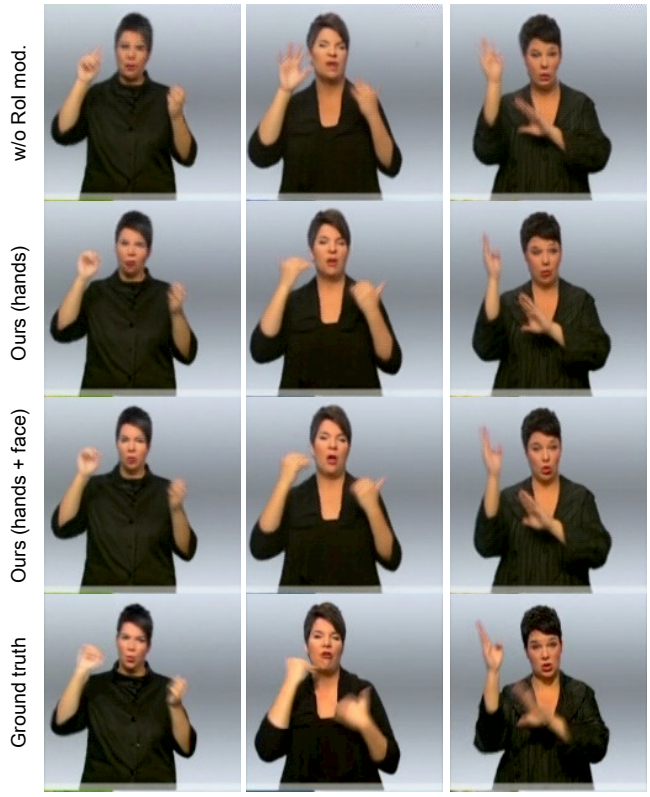

Fig. 12. Output frame examples by approaches with different operations on RoI transform module. The method "w/o RoI mod." denotes the image synthesis model with RoI transform module removed, "Ours (hands)" is our full synthetic model with left and right hands as RoIs, and "Ours (hands + face)" is our full synthetic model with both hands and face region incorporated. The RoI transform module corrects false appearance, adds details and decreases distortion in selected RoIs.

synthesis results in Table I and II ("Ours" vs. "w/o progress" and "Ours" vs. "w/o seq.") . Our proposed scheme of action progress not only allows to generate actions with different gesticulating paces (see Fig. 2), but also helps to disambiguate from temporal steps with the same pose configuration but different action advancements. Fig. 11 shows the comparison of samples generated by "w/o progress" and our proposed approach. In this segment, there is substantial advancement in gesture but little change in posture. Our approach can synthesize the target action with natural evolution in hand-shape. In contrast, model without progress embedding with only skeletal configuration fails to capture the hand-shape evolution in action.

When comparing different training objectives, we find that our approach has a notable improvement compared with method "w/o $\mathcal{L}_{\mathrm{RoI}}$" and "Pixel-wise ℓ_1-loss". For example, our approach achieves a classification accuracy of 69.9% on Phoenix-2014 database, while removing $\mathcal{L}_{\mathrm{RoI}}$ reduces the accuracy to 44.4%, and replacing the objective to pixel-wise loss only gets 32.8%. Examples for comparison on different objectives have been shown in Fig. 6. From Table I and II we can also find that the input of target action label makes consistent contributions to the generation with clear and accurate meanings ("Ours" vs. "w/o label"). We observe notable improvements in these metrics on both datasets.

TABLE IV
EXPERIMENT RESULTS OF ROI TRANSFORM MODULE ON NATOPS DATASET

model	with ground truth poses			with predicted poses		
	IS	FVD	acc.	IS	FVD	acc.
w/o RoI mod.	23.5	164.1	0.987	23.6	302.2	0.974
Ours (hands)	**23.6**	**133.0**	**0.991**	**23.7**	**299.8**	**0.985**
Ours (hands + face)	23.5	138.5	0.990	23.6	316.2	0.981

E. Control experiments on RoI transform module

Note that in our image synthesis model, the RoIs that go through the localized transform module can be readily added or removed. In this section, we make changes to the RoI transform module to evaluate its effect.

Table III and IV show the results of the control experiments on RoI transform module, on Phoenix-2014 and NATOPS datasets, respectively. Model "w/o RoI mod." indicates that we remove the RoI transform module in our image synthesis model, and "Ours (hands)" is our full synthetic model taking left and right hands as RoIs. Since facial expression is an important component of non-manual signals in sign language gestures, we also take the area of face into account in RoI transform module (see "Ours (hands + face)"). The coordinates

TABLE V
COMPARISON OF SYNTHESIZED RESULTS ON PHOENIX-2014 DATASET

model	with ground truth poses			with predicted poses		
	IS	FVD	acc.	IS	FVD	acc.
Video Pred. [16]	29.8	100.0	0.543	24.9	208.7	0.482
Hierarch. Pred. [15]	22.6	128.2	0.405	20.1	246.0	0.354
PoseWarp [35]	32.5	85.4	0.618	26.8	196.5	0.522
Ours	**40.9**	**66.1**	**0.827**	**39.2**	**135.4**	**0.821**
Ground truth	42.3	0.00	0.839		—	

TABLE VI
COMPARISON OF SYNTHESIZED RESULTS ON NATOPS DATASET

model	with ground truth poses			with predicted poses		
	IS	FVD	acc.	IS	FVD	acc.
Video Pred. [16]	19.2	245.0	0.864	18.5	376.8	0.863
Hierarch. Pred. [15]	20.2	217.6	0.897	19.6	375.6	0.926
Posewarp [35]	21.8	135.4	0.939	21.8	331.2	0.949
Ours	**23.6**	**133.0**	**0.991**	**23.7**	**299.8**	**0.985**
Ground truth	24.0	0.00	1.000		—	

of nose are used to locate the region of face. We observe that the development of localized transform module ("Ours (hands)" vs. "w/o RoI mod.") including gesticulating hands brings consistent improvement to the generation performance on both databases. When adding facial region to RoI transform module, we find that our model achieves better performance on Phoenix-2014 database, however, there is no notable improvement on NATOPS database. We anticipate that facial details and expressions help to convey meanings and improve realism in sign language gestures, but show little relevance to aircraft handling gestures in NATOPS.

Fig. 12 compares the samples synthesized by different approaches. We find that our proposed RoI transform module helps to synthesize target region with correct semantic contents and clearer appearance for RoIs. We can observe notable improvements in visual details and decrease in distortions upon the addition of face region in RoI transform module.

F. Quantitative results

We compare our method with recent approaches in video prediction and image synthesis with human activity, including Video Pred. [16], Hierarch. Pred. [15], and PoseWarp [35]. For fair comparison on synthesis performance, all the approaches use the same prediction of future pose configurations provided by our pose predictor, and synthesize frames conditioned on the target action label.

The comparison results for performance on conveying semantic information is presented in Table V and VI. Our approach outperforms the state-of-the-arts with notable improvements in inception score, Fréchet video distance and accuracy. For example, our model achieves an accuracy of 98.5% on NATOPS database, which is a very close performance to the ground truth in conveying the semantic information, showing notable improvements over other baselines with Video Pred. (12.2%), Hierarch. Pred. (5.9%) and PoseWarp (3.6%). The high inception scores of our approach also indicate that our approach generates samples with clear semantic information. We notice that in Table V the accuracy of the pre-trained gesture classifier on ground truth test samples only reaches 83.9%, which indicates the complexity and variety of sign langauge gestures on Phoenix-2014 database.

TABLE VII
MANUAL CLASSIFICATION ACCURACIES ON PHOENIX-2014 AND NATOPS DATASETS

model	Phoenix-2014	NATOPS
Video Pred. [16]	47.7%	79.6%
Hierarch. Pred. [15]	53.5%	75.2%
PoseWarp [35]	52.2%	82.8%
Ours	**62.7%**	**95.3%**
Ground truth	78.9%	98.3%

To evaluate whether the synthesized videos can convey gestural meanings correctly to audience, we ask people to classify the gesture videos manually. The participants are given the correct gestures, including all categories from NATOPS dataset and 10 gestures from Phoenix-2014 dataset, and they are asked to tell the meanings of 150 gesticulating videos. Each video could be synthetic or real sample. We recruit 10 participants to join in this experiment. Table VII shows the results of this experiment. The samples synthesized by our method show the highest accuracies on both datasets, which are closest to those of ground truth among all the approaches. This result demonstrates that our method can generate gesture videos with more precise meanings than these state-of-the-art approaches. Note that the accuracy of ground truth only reaches 78.9% on Phoenix-2014 dataset. One possible reason might be that the recruited people are not sign language experts. The high diversity and motion blur of real sign language gestures without context could also add difficulty to the manual annotation task.

G. Qualitative results

To evaluate the quality of the generated videos in terms of appearance, we follow the evaluation metric similar to [7], [15]. We present synthesized videos from two methods to people and ask them to judge which has a more realistic appearance. The methods for comparison include approaches in [15], [16], [35] and the ground truth. We randomly choose 100 videos for comparison and thus produce 400 pairs for each dataset. We recruit 10 participants to rate the video pairs. The synthesized examples are around 1-2 seconds, including 20-40 frames.

The results are presented in Table VIII where participants rate the appearance. We find that the participants prefer ours to other baselines on both datasets. The preference ratios are 69.6% on Phoenix-2014 and 83.4% on NATOPS over PoseWarp [35], which is the best baseline for comparison in terms of appearance. When compared to the ground truth

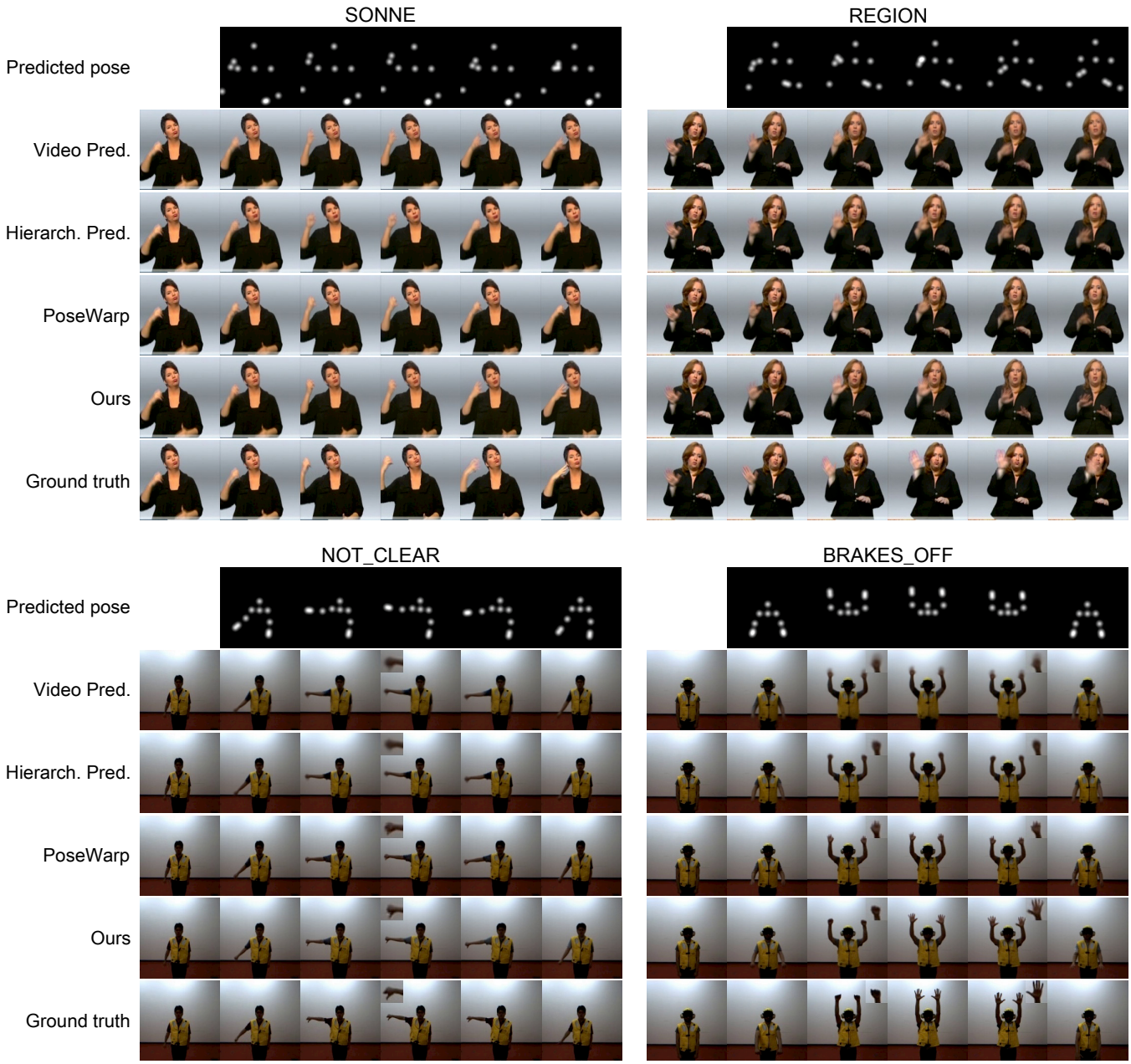

Fig. 13. Synthesized results with different approaches. The left-most column of each action shows the reference frame as the input for these approaches, and the remaining columns present synthesized frames with action progress 0.1 / 0.3 / 0.5 / 0.7 / 0.9 from left to right respectively. Our approach outperforms other baselines with improvements in hand-shape evolution and details, presenting precise gesticulating close to the ground truth action.

sequence, the preference ratios for our approach are 13.9% on Phoenix-2014 and 21.5% on NATOPS.

Fig. 13 presents some synthesized examples by different approaches. German sign language words "SONNE" and "REGION" are from Phoenix-2014 dataset, and gestures "NOT_CLEAR" and "BRAKES_OFF" are from NATOPS dataset. All the compared approach share the same reference frame and the predicted pose sequence. We observe that our method is capable of generating gesture videos with more precise visual contents. Our approach outperforms other baselines with notable improvements in hand-shape evolution and details, presenting realistic and precise gesticulating close to the ground truth action. For example, in gesture "BRAKES_OFF" the sequence of frames generated by our approach clearly shows the process of changing the closed fists to open palms. In contrast, other approaches only learns a coarse pose-to-image transformation, and they fail to reflect this evolution in hand-shape.

VI. Conclusion

We develop a generative model that addresses the problem of gesture video synthesis. The key novelty lies in the proposed localized transform architecture, which captures the temporal and spatial transforms in regions of interest, and provides instructions on what and where to draw for these key regions in future frames. Our idea can be generally applied to other

TABLE VIII
HUMAN PREFERENCE RESULTS ON PHOENIX-2014 AND NATOPS DATASETS

preference	Phoenix-2014	NATOPS
Prefer ours over Video Pred. [16]	83.5%	98.4%
Prefer ours over Hierarch. Pred. [15]	80.4%	97.1%
Prefer ours over PoseWarp [35]	69.6%	83.4%
Prefer ours over ground truth	13.9%	21.5%

video generation tasks with regions of interest for special attention. We also propose the encoding of action progress to explicitly build its relationship with the temporal evolution. We evaluate the proposed method on two datasets for the task of gesture video synthesis. Experimental results present that our approach can produce gesture videos with more realistic appearance and precise meaning than the state-of-the-art video prediction approaches. In the future, to generate videos containing complete sign language sentences may be a promising direction and also a highly challenging task. Predicting full sign language sentences requires the model to handle the transitions between gestures and to keep results in high quality during long-term inference. Besides, the model should learn to synthesize variation of sign language words according to the context. We leave this as future work.

REFERENCES

[1] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F.-F. Li, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014.

[2] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[3] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 2625–2634.

[4] X. Wang, L. Gao, P. Wang, X. Sun, and X. Liu, "Two-stream 3-D convNet fusion for action recognition in videos with arbitrary size and length," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 634–644, 2018.

[5] J. Hou, X. Wu, Y. Sun, and Y. Jia, "Content-attention representation by factorized action-scene network for action recognition," *IEEE Trans. Multimedia*, vol. 20, no. 6, pp. 1537–1547, 2018.

[6] M. Mathieu, C. Couprie, and Y. Lecun, "Deep multi-scale video prediction beyond mean square error," in *Proc. Int. Conf. Learning Representations*, 2016.

[7] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 613–621.

[8] J. Walker, K. Marino, A. Gupta, and M. Hebert, "The pose knows: Video forecasting by generating pose futures," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3352–3361.

[9] R. Kennaway, "Synthetic animation of deaf signing gestures," in *Int. Gesture Workshop*, 2001, pp. 146–157.

[10] A. B. Grieve-Smith, "SignSynth: A sign language synthesis application using Web3D and Perl," in *Int. Gesture Workshop*, 2001, pp. 134–145.

[11] Y. Chen, W. Gao, G. Fang, C. Yang, and Z. Wang, "CSLDS: Chinese sign language dialog system," in *Int. Workshop on Analysis and Modeling of Faces and Gestures*, 2003, pp. 236–237.

[12] L. Havasi and H. M. Szabó, "A motion capture system for sign language synthesis: overview and related issues," in *IEEE EUROCON, Int. Conf. Comput. as a Tool*, 2005, pp. 445–448.

[13] A. Irving and R. Foulds, "A parametric approach to sign language synthesis," in *Proc. Int. ACM SIGACCESS Conf. on Comput. and Accessibility*, 2005, pp. 212–213.

[14] M. Kipp, A. Heloir, and Q. Nguyen, "Sign language avatars: Animation and comprehensibility," in *Int. Workshop on Intelligent Virtual Agents*, 2011, pp. 113–126.

[15] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee, "Learning to generate long-term future via hierarchical prediction," in *Proc. Int. Conf. Mach. Learning*, 2017, pp. 3560–3569.

[16] H. Cai, C. Bai, Y.-W. Tai, and C.-K. Tang, "Deep video generation, prediction and completion of human action sequences," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 374–390.

[17] Y. Jang, G. Kim, and Y. Song, "Video prediction with appearance and motion conditions," in *Proc. Int. Conf. Mach. Learning*, 2018, pp. 2230–2239.

[18] L. Zhao, X. Peng, Y. Tian, M. Kapadia, and D. Metaxas, "Learning to forecast and refine residual motion for image-to-video generation," in *Proc. Eur. Conf. Comput. Vis.*, 2018.

[19] S. C. Ong and S. Ranganath, "Automatic sign language analysis: A survey and the future beyond lexical meaning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 6, pp. 873–891, 2005.

[20] F. Becattini, T. Uricchio, L. Ballan, L. Seidenari, and A. Del Bimbo, "Am I done? Predicting action progress in videos," *arXiv preprint arXiv:1705.01781*, 2017.

[21] J. Forster, C. Schmidt, O. Koller, M. Bellgardt, and H. Ney, "Extensions of the sign language recognition and translation corpus RWTH-PHOENIX-Weather," in *Int. Conf. Language Resources and Evaluation*, 2014, pp. 1911–1916.

[22] Y. Song, D. Demirdjian, and R. Davis, "Tracking body and hands for gesture recognition: NATOPS aircraft handling signals database," in *IEEE Int. Conf. Autom. Face Gesture Recog.*, 2011, pp. 500–506.

[23] J. Walker, A. Gupta, and M. Hebert, "Patch to the future: Unsupervised visual prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 3302–3309.

[24] ——, "Dense optical flow prediction from a static image," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2443–2451.

[25] S. L. Pintea, J. C. van Gemert, and A. W. Smeulders, "Dèjá vu: motion prediction in static images," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 172–187.

[26] J. Walker, C. Doersch, A. Gupta, and M. Hebert, "An uncertain future: Forecasting from static images using variational autoencoders," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 835–851.

[27] Y.-W. Chao, J. Yang, B. Price, S. Cohen, and J. Deng, "Forecasting human dynamics from static images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 3643–3651.

[28] J. Oh, X. Guo, H. Lee, R. L. Lewis, and S. Singh, "Action-conditional video prediction using deep networks in Atari games," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2863–2871.

[29] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee, "Decomposing motion and content for natural video sequence prediction," *arXiv preprint arXiv:1706.08033*, 2017.

[30] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion GAN for future-flow embedded video prediction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1762–1770.

[31] T.-H. Wang, Y.-C. Cheng, C. H. Lin, H.-T. Chen, and M. Sun, "Point-to-point video generation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019.

[32] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros, "Everybody dance now," *arXiv: Graphics*, 2018.

[33] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, "Pose guided person image generation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 406–416.

[34] A. Siarohin, E. Sangineto, S. Lathuiliere, and N. Sebe, "Deformable GANs for pose-based human image generation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[35] G. Balakrishnan, A. Zhao, A. V. Dalca, F. Durand, and J. Guttag, "Synthesizing images of humans in unseen poses," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[36] T. Wang, M. Liu, J. Zhu, G. Guilin, A. Tao, J. Kautz, and B. Catanzaro, "Video-to-video synthesis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1144–1156.

[37] E. Shlizerman, L. M. Dery, H. Schoen, and I. Kemelmacher-Shlizerman, "Audio to body dynamics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 7574–7583.

[38] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing Obama: learning lip sync from audio," *ACM Transactions on Graphics*, vol. 36, no. 4, p. 95, 2017.

[39] S. Ginosar, A. Bar, G. Kohavi, C. Chan, A. Owens, and J. Malik, "Learning individual styles of conversational gesture," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2019, pp. 3497–3506.

[40] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 1302–1310.
[41] P. Ghosh, J. Song, E. Aksan, and O. Hilliges, "Learning human motion models for long-term predictions," in *IEEE Int. Conf. 3D Vis.*, 2017, pp. 458–466.
[42] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[43] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 2980–2988, 2017.
[44] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 5967–5976.
[45] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Int. Conf. Medical Image Computing and Comput. Assisted Intervention*, 2015, pp. 234–241.
[46] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 105–114.
[47] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[48] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, and M. Bernstein, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
[49] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[50] R. Cui, H. Liu, and C. Zhang, "A deep neural framework for continuous sign language recognition by iterative training," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1880–1891, 2019.
[51] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.
[52] T. Unterthiner, S. Van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly, "Towards accurate generative models of video: A new metric & challenges," *arXiv: Computer Vision and Pattern Recognition*, 2018.
[53] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the Kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 4724–4733.

Runpeng Cui received the B.E. degree and Ph.D. degree from Tsinghua University, Beijing, China, in 2013 and 2019, respectively. He is currently a research assistant with the School of Vehicle and Mobility, Tsinghua University, Beijing, China. His research interests include pattern recognition and computer vision.

Zhong Cao received his B.E. degree from Tsinghua University, Beijing, China, in 2014. He is currently a Ph.D. student at the State Key Laboratory of Intelligent Technology and Systems, Department of Automation, Tsinghua University, Beijing, China. His research interests include pattern recognition and machine learning.

Weishen Pan received his B.E. degree from Tsinghua University, Beijing, China, in 2015. He is currently a Ph.D. student at the State Key Laboratory of Intelligent Technology and Systems, Department of Automation, Tsinghua University, Beijing, China. His research interests include pattern recognition and causal inference.

Changshui Zhang (M'02-SM'15-F'18) received the B.S. degree in mathematics from Peking University, Beijing, China, in 1986, and the M.S. and Ph.D. degrees in control science and engineering from Tsinghua University, Beijing, in 1989 and 1992, respectively.

In 1992, he joined the Department of Automation, Tsinghua University, where he is currently a professor. He has authored more than 200 articles. His current research interests include pattern recognition and machine learning.

Jianqiang Wang received the B.Tech. and M.S. degrees from the Jilin University of Technology, Changchun, China, in 1994 and 1997, respectively, and the Ph.D. degree from Jilin University, Changchun, China, in 2002. He is currently a professor with the School of Vehicle and Mobility, Tsinghua University, Beijing, China.

Prof. Wang is the author or co-author of over 40 journal papers, and the co-holder of 30 patent applications. His active research interests include intelligent vehicles, driving-assistance systems, and driver behavior.