Master Thesis

Egocentric Pedestrian Motion Prediction by Separately Modeling Body Pose and Position

（姿勢と位置の分離に基づく一人称視点映像中の人物の動き予測）

Donghao Wu

Department of Information and Communication Engineering
Graduate School of Information Science and Technology
The University of Tokyo

Advisor
Prof. Yoichi Sato

Abstract

Modelling human dynamics has long been a challenging problem in computer vision. Especially, it is crucial for a machine system to interact with human in a meaningful way with observed pose knowledge. Nevertheless, most of current research focus on modelling human motion from static vision or using fully supervised data, which is tough to be associated with daily scenarios and applied by machines practically. Meanwhile, the common problem on mean pose convergence of predicted future motions keeps bothering researchers.

In this thesis, we expect to achieve human dynamics forecasting upon practical scenarios — viewed from egocentric vision. We especially focused on pedestrian dynamics forecasting and take three granularities: the pose, scale and location of target person observed from egocentric videos into consideration. By separately modeling the pose and its corresponding scale and location temporally with two different deep neural networks whose outputs are later combined to make final prediction, our proposed method successfully forecast the position and body pose of the target person with ideal scale, relieving from the mean pose problem – a common problem usually occurs in pose forecasting works, that predicted poses act unnaturally and converge to a stable posture.

We evaluated the experiment based on First-Person Locomotion (FPL) dataset and demonstrated the effectiveness of our method.

Keywords: first-person video, egocentric vision, human dynamics, pose forecasting, neural network, deep learning

Contents

List of Figures **1**

List of Tables **3**

1. Introduction **5**
1.1. Overview 5
1.2. Contributions 7
1.3. Thesis Outlines 7

2. Related Work **9**
2.1. Human Motion Prediction based on Mocap Data 9
2.1.1. Motion Capture Data 9
2.1.2. Recurrent Neural Network based modelling structure 10
2.1.3. Convolutional Neural Network based modelling structure . . . 11
2.2. Human Motion Prediction based on Static Visions 12
2.3. Human Motion Prediction based on Egocentric Visions 13
2.3.1. Future human localisation in egocentric settings 13

3. Proposed Method **15**
3.1. Problem Formulation 15
3.2. Standardization of Pose Stream 16
3.3. Pose Displacement Stream 17
3.4. Model Architecture 18
3.4.1. Pose Forecasting Module 18
3.4.2. Scale & Location Forecasting Module 18

4. Experiments **19**
4.1. FPL Dataset 19
4.2. Implementation 19
4.2.1. Formulation 19
4.2.2. Network Choice 20
4.2.3. Loss Function 20
4.2.4. Optimization 21
4.3. Evaluation Protocols 21
4.3.1. Evaluation metric 21
4.3.2. Baseline methods 22

4.4. Results Analysis 24
4.4.1. Error analysis 24
4.4.2. Visualization analysis 24
4.4.3. Effect of scale-normalized error 25
4.4.4. Effect of using pose-displacement 25

5. Conclusion **31**
5.1. Summary 31
5.2. Limitation and Future Work 31

Acknowledgments **33**

A. Additional Results **35**

References **41**

List of Figures

1.1. Observing Human Motion from Egocentric Vision 6

2.1. Residual Recurrent Neural Network (RRNN) 10
2.2. Convolutional Sequence to Sequence Neural Network (CS2S) 11
2.3. Future human localisation in egocentric settings 13

3.1. Proposed Separate Modelling Approach 16
3.2. Pipeline of Pose Forecasting Module 17
3.3. Pipeline of Scale & Location Forecasting Module 18

4.1. Example images in FPL Dataset 20
4.2. Prediction results of **Ours 1**. 23
4.3. Visualization results of predicted poses. 28
4.4. Use scale-normalized error or not. 29
4.5. Error comparisons with using pose-displacement or pose. 30
4.6. Standardized pose visualization comparing with using pose-displacement or not. 30

A.1. Prediction results of **Ours2**. 36
A.2. Additional visualization of predicted pose sequence (standardized) . . 37
A.3. Additional visualization results comparing with ours and baseline method 38
A.4. Additional visualization results comparing with ours and baseline method (1) 39
A.5. Additional visualization results comparing with ours and baseline method (2) 39
A.6. Additional visualization results comparing with ours and baseline method (3) 40

List of Tables

4.1. Comparisons to baseline methods 23
4.2. Evaluation on Pose, Scale and Location granularity 26

1. Introduction

1.1. Overview

Human dynamics forecasting has been an increasingly important topic in the computer vision field, widely concerning with human-computer interaction. The problem is of great significance in real application. To enable machines to perceive and have interaction with moving people, knowledge of how the targets move is requested. Due to the complex physical dynamics of human activities and complicated surrounding contexts, it is crucial for the forecasting system to model the motion's temporal changes. Consistent with existed works [30, 24, 8, 2], since the position of skeleton joints can intuitively reflect the movement of human body, past skeletal sequence is generally applied to describe pose dynamics. However, most of previous works conducted research based on fully-supervised skeletons, such as Motion-Capture data [4], pixel motions [19, 42, 39] or studied based on static vision [37], which makes their proposed pose forecasting methods hardly to be practically applied in a wild natural environment.

To further extend human dynamics forecasting to daily scenario, in this work, we carried on our research upon egocentric visions [45]. Different from previous data settings, we observed a more complicated dynamics dependencies inherent in the articulated human body in our case. For former supervised pose skeletons in Mocap data (2D or 3D) or for the family of pose forecasting studying through single image, only the structural and temporal movements of joints in skeleton space require to be considered. While for the 2D pose sequence observed in first-person videos, the variance of skeleton joints in the pose stream is also accompanied by the changes of scale and location of target person. For example (see Figure 1.1), the size of the pose could turn larger when target person is moving toward to the observer. Meanwhile, the target's location appearing in the camera scene also varies in pixel units accordingly in the observed frame sequence.

Since we have empirically observed that current single DNN (deep neural network) based approaches have difficulty obtaining good performance on motion modeling, while striving to simultaneously learn visual spacial dependencies (the scale and location of the pose), we propose a separate modeling approach to tackle the problem. Specifically, we separately learn the pose and its corresponding scale and location with two forecasting modules based on different neural networks. For the pose forecasting module, we train the network modeling based on standardized input pose

Figure 1.1.: Observing Human Motion from Egocentric Vision. The temporal variance of skeleton joints in the target's pose stream is accompanied by the changes of corresponding scale and location of the body viewed from a moving camera.

sequences (with normalised scale and centralized position) aiming to only learn kinetic dependencies of pose ; for the scale&location forecasting module, we learn visual spacial dependencies unifying a convolution-deconvolution network proposed in [38] to capture the target's scale and location granularities. Different from [38] that has good prediction performance on location but weak on scale by accepting past scale-location sequences, we adopt pose-displacement cue that contains the temporal variance of pose with joints, describing how pose changes positions and scales in 2D space in the observed time sequence. With the help of this input cue, we achieved obtaining future pose sequence with ideal scale and location. After a combination process with outputs of the two modules, we could generate a complete future pose sequence that we expect.

We validated our trained model on FPL dataset and achieved considerable prediction performance with visualization.

1.2. Contributions

We list two main contributions in our thesis as follows:

- Firstly, we adopt a novel separate modelling approach consisting of pose forecasting module and scale-location forecasting module based on sequence to sequence architecture to model the complex granularities of human pose in egocentric scenes. The structure highly elevates the modelling ability, relieving mean pose convergence problem and generating natural-looking pose stream with promising scale and location appeared in the consecutive frames.
- Secondly, we propose to make use of pose-displacement Cue, defined as relative differences between the initial pose and the pose at the following certain time in a past pose stream, as input into our Scale&Location Forecasting Module. Experiment results indicate the effectiveness of pose-displacement Cue on scale and location prediction. And also, we notice that modeling pose-displacement can help the network to capture active joints movement, with which, we combine the outputs from the two modules and strengthen the joints flexibility of generated poses.

1.3. Thesis Outlines

The rest of this thesis is organized as follows. In Chapter 2, we review previous works on human motion prediction based on different observation perspectives. In Chapter 3, we introduce our model structure, and the utilization of the standardized pose stream and pose-displacement stream as inputs in our proposed modules. In Chapter 4, we introduce the implementation of the experiment and evaluate our method on the egocentric dataset showing with visualization results. In Chapter 5, we discuss the limitations of our work and possible future improvements as reference.

2. Related Work

In this chapter, we review previous works on human motion prediction.

2.1. Human Motion Prediction based on Mocap Data

A large family of human motion prediction works [16, 21, 1, 5, 6] are studied based on Motion capture data. Due to the success of deep learning technique in machine learning and computer vision, deep neural network based models have been introduced for human dynamics learning and proved to be much more effective than traditional approaches like [12, 27, 41, 9]. Recent works of this field proved that sequence to sequence models including Recurrent Neural Network (RNN) [29, 6, 18] and Convolutional Neural Network (CNN) [14, 17, 5] based structures showed good performance on both learning spatial and temporal dependencies of the body joints.

2.1.1. Motion Capture Data

Generally speaking, motion capture data refers a category of large human motion datasets collected through using series of accurate marker-based motion capture devices and actors dressed with kinetic detection sensors taking actions viewed by indoor backgrounds. Currently, the most widely used Mocap data in pose forecasting field is called Human3.6M [4]. There are over 3.6 million human poses, viewed from 4 different angles with an accurate motion capture system in this dataset. All the motions are acted by actors and covered with a diverse set of scenarios. It is noteworthy that for pose parametrisation considered in this motion data structure includes fully synchronised 3D joint positions and joint angles. It is a relative parametrisation with a special designated joint as root point, which is regarded as the center of the coordinates, while the other joints are estimated relative to it. One advantage of the data setting exists in the relative joint angles between limbs that keeps both body size and proportion invariant, making the model capture motion dependencies easier without considering the changes of pose scale and position. However, since all the pose data is fully supervised in a conditional indoor environment and acted under certain indications, it is hard to be adaptive to daily-life application scenarios.

Figure 2.1.: Residual Recurrent Neural Network (RRNN) The model accepts past poses at each time-step, and the step loss is calculated on the decoder network feeding its own predicted samples. The residual link is connected between the input and the output of the GRU cells in the decoder to forcibly model the pose velocity, which helps to keep continuity of future poses in the following time steps. Both encoder and decoder share weights each other.

2.1.2. Recurrent Neural Network based modelling structure

In past few years, a family of methods based on Recurrent Neural Network (RNN) have been successfully applied in modeling human pose sequence. RNN based structures including encoder-recurrent-decoder (ERD) [21], QuasiRNN [15], structural RNN (SRNN) [1], dropout auto-encoder LSTM [32], etc., are proved to be more effective than traditional probabilistic models [23, 31, 43]. Although these RNN models learn spatial and temporal dependencies from the pose data, they sometimes fail to generate natural-looking poses and a severe discontinuity is observed between the pose in the last input frame and the predicted pose in the first output frame.

A notable RNN based model (showed in Figure 2.1) proposed by Martinez et al.[18], highly elevated the prediction result comparing with previous works. The author addressed the motion prediction as the search for a function that conditions ground truth (the input sequence) to the forecasted sequence. To some extent, this process is similar to machine translation consisting of sequence to sequence structure. Different from general practice on encoder-decoder training thoughts, Martinez et al. forcibly share weights between the encoder and the decoder to facilitate convergence. Instead of relying heavily on hyper-parameter tuning (making the internal states of the RNN

Figure 2.2.: Convolutional Sequence to Sequence Neural Network (CS2S) The model encodes ground truth sequence as a whole. The decoder part also has a encoder-decoder to generate future pose sequence at each future time step. Sampling-based feeding is also adopted during the training process with a residual structure between the input and the output in the decoder part.

same in train and test time), the author used sampling-based training method, which means during the training process, they feed the decoder by always using its own producing samples. Thus, not only heavy parameter tuning is no longer required but also the length of output can be freely adjustable for research convenience.

Another novel point exists in the architecture is the designation of the residual connections [10] in the decoder part. The author noticed the significance of velocities (difference between the pose at a certain time step and the pose at the next adjacent time step), which is more efficient to describe property of human pose than pose itself . By adding a residual structure at the input and the output of GRU cells [36] in the decoder part, the pose velocity is forcibly modelled and directly output from the RNN cell. This setting strongly helps in motion consistency and thoroughly solved the discontinuity problem.

2.1.3. Convolutional Neural Network based modelling structure

Convolutional neural network was also adopted as a sequence learning structure [17, 20] for human pose modeling with the use of Mocap data. A relative work Li et al. [5] , aims to simultaneously capture the long-term and short-term information of actions with the use of convolutional long-term encoder and short-term encoder (showed in Figure 2.2). The long-term encoder works as the encoder of the whole model. The decoder also has an encoder-decoder structure, which consists of a short-term encoder and a spatial decoder. The short-term encoder has similar structure to the long-term encoder, including three convolutional layers and one fully connected layers, and the spatial decoder consists of two fully connected layers. Li et al. in their work prove that the hierarchical structure of CNN also makes it capable

of capturing both spatial and temporal correlations of pose joints effectively. In their experiment, the original input pose sequence is divided into long-term and short-term sequences that the Long Term Encoder captures pose correlation in the long-term while in the decoder part, a chained-RNN like form with sampling-based training method encodes the sub sequences clipped from the long-term sequence with a certain window size, and keep feeding its own output to replace the old pose in the former frame. With the use of this architecture, the author acclaimed that the convolutional network based sequence to sequence model could also achieve promising performance both in the short-term and the long-term pose prediction, even superior to the chain-structured RNN. Besides, simulating the work [18], there are also residual connections between the input and the output of the sub-encoder-decoder cells in the decoder part to model the velocity, which shows a crucial role for velocity helping to learn kinematic characteristics between poses in adjacent frames.

2.2. Human Motion Prediction based on Static Visions

The human skeleton data extracted by pose estimation algorithm through images is usually used in the case where it is inconvenient to collect highly supervised human body data due to the limitations of the experiment environment. Thus, much work in motion forecasting turns to relying on predicting future semantic action classes [11, 22, 25], or more generally semantic information [34, 40], utilising large amounts of unlabelled data with unconstrained settings. A typical work by Chao et al. [47] , proposed a PFNet as the first study on forecasting human dynamics from single RGB images. A 2D to 3D pose converting process was conducted in this work to show qualitative results. Jacob et al. [42] combined the strength of VAE to model human dynamics from static images and using conditional GAN [28] to generate future poses in pixel level. Wu et al. [44] adjusted the ResNet50 pretrained on Human3.6D as their model to evaluate 2D pose from RGB frames and conducted pose forecasting for Virtual Reality with stack-LSTM structure . However, these works achieved pose forecasting mainly based on fixed vision, or say third-person vision, without taking moving observation into account. We need to notice that practical human activities are extremely complicated and the unsteady observation perspective may add effect to the pose modelling performance. For helping machines to better forecast human motion in human-machine interaction scenarios, viewing human poses from egocentric scenes should also be considered.

Figure 2.3.: (a) Problem formulation of future location forecasting of target person in egocentric videos. (b) The Multi-Stream based Conv-Deconv Forecasting Model proposed by Yagi et al. [38].

2.3. Human Motion Prediction based on Egocentric Visions

Our work is interested in studying human motion prediction in egocentric videos. However, there are few works focusing on egocentric settings. A similar work [26] proposed by Mangalam et al., achieved trajectory forecasting of body joints. While they mainly focused on self-driving scenarios, in which the viewer perspective is more like static third-person visions. In [38], Yagi et al. designed a conv-deconv model to forecast the future location of target person in first-person videos, which highly inspired the ideas of our research. In this section, we will introduce the egocentric vision and have a brief review of work [38].

2.3.1. Future human localisation in egocentric settings

There were recent works focusing on studying problems in first-person scenes such as gaze anticipation [13], person identification [46] and location forecasting [38], etc.. Our research refers to the problem formulation of [38] and achieves extending the future location forecasting to the future pose forecasting in egocentric visions. In [38], Yagi et al. firstly formulated the problem of locations, or say trajectories forecasting of pedestrians in the egocentric scene. The goal was to foretell the future location of the target person based on the observation of the target's past movement. As illustrated in Figure 2.3 (a) , the author tackled the problem based on the target's scale and location, the target's pose, and the ego motion of the observer. A multi-stream based convolution-deconvolution neural network (Figure 2.3 (b)) was used to encode these three cues and then output the expected location streams. Mean

squared error (MSE) was adopted in the loss function and Adam [7] was used for model optimization.

As reference, the utilisation of features like human scales and locations viewed from egocentric visions strongly inspired our analysis on pose forecasting. In our work, we follow the data preprocessing procedure provided in [38] for the pose estimation and extraction. We also make use of the Conv-Deconv structure in our separate modelling approach to generate the future scale and location stream of poses.

3. Proposed Method

In this chapter, we introduce our proposed method of human pose forecasting in egocentric videos.

3.1. Problem Formulation

Firstly, we frame the task of human pose forecasting in egocentric videos as a sequence-to-sequence problem. The goal of our work is to predict future pose sequence of target person based on the observed pose sequences in the temporal domain. Suppose $\boldsymbol{p}_t$ denotes the human pose at time t, which contains N two-dimensional joints, *i.e.*, $\boldsymbol{p}_t = ((u^1, v^1), (u^2, v^2), ..., (u^N, v^N))^\top \in \mathbb{R}_+^{2N}$. We aim to forecast the target person's future poses $\boldsymbol{q}_t \in \mathbb{R}_+^{2N}$ in the subsequent T_f time steps with the observed poses $\boldsymbol{p}_t \in \mathbb{R}_+^{2N}$ in the past T_p time steps as input. This results in the input sequence and the output sequence as bellow:

$$\textit{input sequence}: \boldsymbol{P}_{in} = \begin{pmatrix} \boldsymbol{p}_{t-T_p+1} \\ \cdots \\ \boldsymbol{p}_{t-1} \\ \boldsymbol{p}_t \end{pmatrix}^\top \in \mathbb{R}_+^{2N \times T_p} \tag{3.1}$$

$$\textit{output sequence}: \boldsymbol{P}_{out} = \begin{pmatrix} \boldsymbol{q}_{t+1} \\ \cdots \\ \boldsymbol{q}_{t+T_f-1} \\ \boldsymbol{q}_{t+T_f} \end{pmatrix}^\top \in \mathbb{R}_+^{2N \times T_f} \tag{3.2}$$

As discussed in previous chapters, we observed that the target person viewed in egocentric visions not only has a time-variant pose, but also the corresponding scales and locations appeared in the scene varies temporally. Thus, to fully describe the human pose in moving scenes, we need to at least take three aspects into consideration, 1) what does the pose look like ? 2) how large the pose is ? 3) where does the pose show in the scene ? Based on these three questions, we analyse the human pose with three granularities (pose, scale and location). To obtain these features, we adopt OpenPose [3]– a famous pose estimation algorithm to capture 2D body skeleton of target person. For learning features effectively, we develop a separate modelling approach that separately modelling target's pose with a pose forecasting

Figure 3.1.: Proposed Separate Modelling Approach for forecasting future pose stream in egocentric visions. The model mainly consists of pose forecasting module and the scale&location forecasting module to respectively accept a pose stream and a pose-displacement stream for modelling motion dynamics and the variance of pose's scale and location. The observed pose stream is extracted with OpenPose [3] through series of egocentric video frames. The final predicted pose stream can be generated with the combination of the outputs from the two modules.

module and modelling corresponding scale and location using a scale&location forecasting module . By recombining the two output streams from the two modules, our model can generate future pose streams with the ideal posture as well as promising scale and location. Our proposed model (see Figure 3.1) can be trained end-to-end with back-propagation with a fast learning convergence.

3.2. Standardization of Pose Stream

While the scale of the target person differs by where the person appears, it is tough for a single neural network to simultaneously learning the temporal spatial dependencies of body joints and the variation of the body scale and the body position. To better model the motion dynamics from the input pose sequence, we use a standardised pose stream $\overline{\boldsymbol{P}}_{in} \in \mathbb{R}^{2N \times T_p}$ as input in the pose forecasting module. 'Standardisation' here means the scale normalisation and the location centralisation of observed pose sequences. During the construction of the standardized pose stream, we firstly subtract each location with every observed body joints to centralised the position $l_t \in \mathbb{R}^2_+$ of the pose at time step t. After that, we normalised the centralised pose in each time step with its corresponding scale. For mathematical operation convenience, we both stack l_t and s_t to $2N$ dimension same with pose $\boldsymbol{p}_t$. Here we define

Figure 3.2.: Pipeline of Pose Forecasting Module.

the $2D$ coordinates of neck joint as pose position and the $L2$ distance between neck joint and waist joint in the skeleton as the scale of the pose.

The structure of the variable shows as bellow:

$$input\ sequence\colon \overline{\boldsymbol{P}}_{in} = \begin{pmatrix} \overline{\boldsymbol{p}}_{t-T_p+1} \\ \cdots \\ \overline{\boldsymbol{p}}_{t-1} \\ \overline{\boldsymbol{p}}_{t} \end{pmatrix}^{\top} \in \mathbb{R}^{2N\times T_p} \tag{3.3}$$

$$output\ sequence\colon \overline{\boldsymbol{P}}_{out} = \begin{pmatrix} \overline{\boldsymbol{q}}_{t+1} \\ \cdots \\ \overline{\boldsymbol{q}}_{t+T_f-1} \\ \overline{\boldsymbol{q}}_{t+T_f} \end{pmatrix}^{\top} \in \mathbb{R}^{2N\times T_f} \tag{3.4}$$

$$standardization\colon \overline{\boldsymbol{p}}_i = \frac{\boldsymbol{p}_i - \boldsymbol{l}_i^{stack}}{\boldsymbol{s}_i^{stack}} \in \mathbb{R}^{2N} \tag{3.5}$$

3.3. Pose Displacement Stream

We define a novel cue called pose-displacement as input in the scale&location forecasting module. The mathematical definition of pose-displacement stream shows as below:

$$\boldsymbol{P}_{D_{in}} = \begin{pmatrix} \boldsymbol{p}_{t-T_p+2} - \boldsymbol{p}_{t-T_p+1} \\ \cdots \\ \boldsymbol{p}_{t-1} - \boldsymbol{p}_{t-T_p+1} \\ \boldsymbol{p}_{t} - \boldsymbol{p}_{t-T_p+1} \end{pmatrix}^{\top} \in \mathbb{R}^{2N\times(T_{previous}-1)} \tag{3.6}$$

As shown in Formula (3.6), the pose-displacement cue represents the displacements between the pose joints at the initial time step and the body joints at the subsequent time steps. It contains the temporal variance of pose with joints, describing

how pose changes positions and scales in 2D space. In the experiment, we let scale&location forecasting module to accept pose-displacement stream to generate future pose sequence with scale and location . Evaluation results show that modeling by pose-displacement can achieve a same-level forecasting performance of location and a much better scale prediction result than using scale-location cue proposed in [38].

3.4. Model Architecture

In this section, we will introduce the working pipeline of the two modules in our proposed approach.

3.4.1. Pose Forecasting Module

Figure 3.2 shows our pose forecasting pipeline. Notice that we conduct loss calculation here with the use of ground truth pose stream in standardized form. For modeling time-series data, recurrent neural networks are usually used. As chapter 2 introduced that recent works showed the convolutional model through training the entire sequence could also achieve promising prediction performance . In our experiment, we both tried RNN-based and CNN-based model for modelling human dynamics. We referred to the structure of two sate-of-the-art pose forecasting networks (**RRNN**, see Figure 2.1 and **CS2S**, see in Figure 2.2). Since the two works were designed for modelling motion-capture dataset, to fit our case, we did minor adjustment to the layer dimension for training human body with joints in 2D coordinates.

3.4.2. Scale & Location Forecasting Module

Figure 3.3 shows our scale&location forecasting pipeline. For predicting ideal scale and location of future pose sequences, we refer to **Conv-Deconv** network proposed by [38], who proves the effectiveness on learning complicated dynamics of human trajectories in egocentric videos. Unlike the original multi-input structure , in our work, we only take one channel to accept pose-displacement cue as input and achieve forecasting future pose stream with expected scale and location features.

Figure 3.3.: Pipeline of Scale & Location Forecasting Module.

4. Experiments

In this chapter, we report experimental results evaluating on FPL dataset and show the effectiveness of our approach.

4.1. FPL Dataset

To evaluate the performance of the proposed approach on pose forecasting in egocentric videos, we used First-Person Locomiton Dataset (FPL Dataset) that recorded realistic pedestrians activities in social environments with wearable observing devices. FPL Dataset consists of 4.5 hours video frames with more than 5,000 observed people. Figure 4.1 shows some samples of the dataset.

To evaluate our approach on the dataset, we followed the data preprocessing procedure of work [38], including video undistortion, median filtering and pose imputation etc. to relieve the erroneous or missing detection due to occlusion and motion blur. We estimate the pedestrian pose with OpenPose [3] to extract 25 2D body joints per person and remove 4 joints of the head part (two eyes joint, two ears joints) due to their bad detection. To further relieve the pose noise, we downsampled the raw video of 100fps to 25 fps rather than 10 fps in [38]. We confirm a more consecutive pose sequence can be better helpful for networks to learn the motion dynamics.

4.2. Implementation

4.2.1. Formulation

The given tuple $(\boldsymbol{p}_t, \overline{\boldsymbol{p}}_t, \boldsymbol{p}_{D_t}, \overline{\boldsymbol{q}}_t, \widetilde{\boldsymbol{q}}_t, \boldsymbol{q}_t)$ shows variables at time step t, where $\boldsymbol{p_t}$ is observed pose, $\overline{\boldsymbol{p}}_{\boldsymbol{t}}$ is standardized pose (centralized and normalized from observed pose), $\boldsymbol{p_{D_t}}$ is pose-displacement, $\overline{\boldsymbol{q}}_{\boldsymbol{t}}$ is output pose from pose forecasting module, $\widetilde{\boldsymbol{q}}_{\boldsymbol{t}}$ is output pose from scale & location forecasting Module, $\boldsymbol{q_t}$ is final predicted pose. In this experiment, we specify input and output sequence with 10, 15, 20, 25 frames (400ms, 600ms, 800ms, 1000ms long in a video with 25 fps). Notice that one time step indicates one frame.

Figure 4.1.: Example images in FPL Dataset. The dataset includes more than five thousands walking pedestrians in daily-life environments.

4.2.2. Network Choice

The modelling process is divided into two parts: one is for modelling motion dynamics of pure poses with standardized scale and position; the other one is mainly learning the scale and location information of observed pose sequence. Thus, there are two networks used in our approach. As introduced in chapter 3, we try both RNN-based [18] and CNN-based [5] architecture for pose forecasting and adopt convolution-deconvolution architecture [38] for scale and location prediction. For structure settings, we basically follow the specification of the networks introduced in their articles [18, 5]. For conv-deconv network, only one channel is used since only pose-displacement Cue is accepted in the network.

4.2.3. Loss Function

Both modules were training pose sequences with the L2 loss between the predicted pose and the input pose. In pose forecasting module, we calculate loss of the body joints based on standardized pose stream. In scale&location forecasting module, we

observed adding normalization with the scales of groundtruth during loss calculation can help to improve the prediction performance. Considering the pose we measured from the frames in 2D coordinates, it contains scale granularity which varies with the physical distance between the camera and the target person, if we want to model the location that the target person might appear in the frame, we need to reduce the visual error as much as possible that the scale of the pose might cause. Meanwhile, since both predicted pose and groundtruth pose are normailzed with the same scale from groundtruth sequence, this operation avoid accumulating redundant error during backpropogation and will not affect the process of learning the target scale. The loss functions in two modules are shown as below:

$$Pose\ Forecasting\ Module: \qquad Loss = \sum_{i=t+1}^{t+t_f} \left(\overline{\boldsymbol{q}}_{\boldsymbol{i}} - \overline{\boldsymbol{q}}_{\boldsymbol{GT_i}} \right) \tag{4.1}$$

$$Scale\&Location\ Forecasting\ Module: Loss = \sum_{i=t+1}^{t+t_f} \left(\widetilde{\boldsymbol{q}}_{\boldsymbol{i}} / \boldsymbol{s}^{\boldsymbol{stack}}_{\boldsymbol{q_{GT_i}}} - \boldsymbol{q}_{\boldsymbol{GT_i}} / \boldsymbol{s}^{\boldsymbol{stack}}_{\boldsymbol{q_{GT_i}}} \right) \tag{4.2}$$

where $\overline{\boldsymbol{q}}_{\boldsymbol{GT_i}}$ and $\boldsymbol{q}_{GT_i}$ respectively represents the standardized ground truth pose and the normal ground truth pose at time step i; $\boldsymbol{s}^{\boldsymbol{stack}}_{\boldsymbol{q_{GT_i}}}$ signifies the ground truth scale that has been stacked to the dimension same with pose for calculation convenience.

4.2.4. Optimization

To train the network, we set learning rate at 0.001 for learning motion dynamics in pose forecasting network and optimize it via stochastic gradient descent (SGD) [35]. For scale&location prediction, we set learning rate at 0.002 and use Adam to optimize the network. We trained 15000 iterations with mini-batches of 64 samples. All implementations were conducted based on Pytorch.

4.3. Evaluation Protocols

4.3.1. Evaluation metric

The straightforward measurement of modelling performance is to calculate the distance between body joints of predicted pose and the original pose. However, as shown in Figure 4.4, we find it is hard to be accurately evaluate simply by the $L2$ distance of keypoints of the pose. Since we calculate the pose error in pixel units, the average distance of body joints in pose pairs with large scale and small scale,

would take up different extents of pixels. To eliminate this visual error viewed from images, we propose to add scale-normalization to the error metric to evaluate the predicting performance of the model. The metric is shown as bellow:

$$Evaluation\ Metric\colon error = \sum_{i=t+1}^{t+t_f} \left(\boldsymbol{q}_i / \boldsymbol{s}^{\boldsymbol{stack}}_{\boldsymbol{q_{GT_i}}} - \boldsymbol{q_{GT_i}} / \boldsymbol{s}^{\boldsymbol{stack}}_{\boldsymbol{q_{GT_i}}} \right) \tag{4.3}$$

To further understand the details of the prediction results, we also propose to evaluate pose, scale and location separately with $L2$ distance. More details are shown in Table 4.2.

4.3.2. Baseline methods

There were few previous methods aiming to forecast human pose in egocentric visions, we have implemented the following baselines.

RRNN [18] This method as introduced in Chapter 2, is one of relevant methods to forecast human pose however based on Mocap data. We introduce this baseline to see how RNN-based network performance on modelling human pose with scale and location granularities. We adjust the network making it available to accept 2D body skeletons as input. The normalization of input pose sequence has been conducted to make it closed to zero means and unit standard deviation.

CS2S [5] Like **RRNN**, the network has also been introduced in Chapter 2. It is also one of state-of-the-art networks in pose prediction on Mocap data. We employ this model with several minor modifications in order to see if hierarchical CNN-based model can adapt to forecasting pose with scale and location information. In training process, we set short-term pose length to six to achieve best learning performance.

Conv-Deconv [38] This method firstly achieved human location forecasting in first-person videos. Since the model accepts pose cue as input for helping modeling human trajectory dynamics, we followed the method setting to input pose cue (normalized to zero means and unit standard deviation) to generate future pose sequence to evaluate the motion dynamics modeling ability of the method comparing with our proposed approach.

Millionseconds	400ms	600ms	800ms	1000ms
RRNN [18]	1.068	1.191	1.223	1.376
CS2S [5]	0.664	0.767	0.837	0.928
Conv-Deconv [38]	0.404	0.628	0.678	0.707
Ours 1	0.357	0.453	0.569	**0.628**
Ours 2	**0.343**	**0.450**	**0.567**	**0.628**

Table 4.1.: Comparisons to baseline methods. Each score describes the scale-normalized $L2$ distance between bodyjoints of predicted pose and ground truth. The error is in percentage with respect to the frame size of 960×1280 pixels based on the unit length of pose scale. For our proposed methods, **Ours 1** represents the approach with RRNN working in pose forecasting module and conv-deconv working as scale&location forecasting module; **Ours 2** represents CS2S working in pose forecasting module and conv-deconv working in scale&location forecasting module.

Figure 4.2.: Prediction results of **Ours 1**. Note that the figure shows the result with the input 15 frames and output 15 frames. Predicted pose sequence supports the effectiveness of our separate modelling approach that facilitates the model to learn both motion dynamics as well as scale and location dynamics temporally without severe mean pose convergence problem.

4.4. Results Analysis

4.4.1. Error analysis

Table 4.1 shows results of our proposed approach comparing to baseline methods on FPL dataset. All methods take observed pose sequences extracted from OpenPose as input and generate future pose streams. As shown in the chart, we confirm our separating modeling approach significantly helps modeling the complex pose features viewed from egocentric videos. We verified that only using a single neural network (CNN-based or RNN-based) has poor ability to learn motion dynamics and scale-location dynamics together.

Specifically speaking, unlike **CS2S** model with hierarchical structure, **RRNN** has worse performance on simultaneously modeling the three aspects granularities of target person. But speaking of the performance working in pose forecasting module, **RRNN** has same-level learning performance as **CS2S**. It seems without taking into account scale and location information, both RNN-based and CNN-based models have capability of learning temporal and spatial dependencies of the standardized pose sequence.

For **Conv-Deconv** method, we find out it achieves lower error than **CS2S**. This is partially because **Conv-Deconv** has a good learning ability of scale-location information inherent in pose sequence. But it is hard for people to tell only from the error value, which also reminds us of evaluating more specifically on pose, scale and location and, shows visualization results of the predicted poses.

4.4.2. Visualization analysis

We analysis the prediction performance of our methods and baseline methods intuitively viewed in egocentric video frames. Figure (4.2) gives several samples of visualization results based on **Ours 1** (Conv-Deconv + RRNN). Since **Ours 1** and **Ours 2** (Conv-Deconv + CS2S) have same-level performance, here in Figure (4.2), we only take visualization results of **Ours 1** as an example. It shows the results that given previous observed pose sequence (15 frames, 600 million seconds length in the video), and predicting future 600ms pose sequence. Colorful poses are the input pose sequence observed with the camera and red poses represent the ground truth of future poses. Blue poses are predicted results generated with our proposed approach. We can notice that our approach can obtain natural-looking poses with expected scales and locations. Also, comparing with baseline methods (see Figure 4.3), we confirm that with our approach, the predicted poses are basically much relieved from mean pose convergence, including pose convergence, scale convergence and location convergence. We can straightforwardly judge from the visualization results that performance of our proposed approach much surpass the baseline methods without

the separate modeling setting. Notice the blue pose sequence in $(a), (b) and (c)$, the predicted pose is gradually converging to a certain posture with time step increasing, while our approach keeps generating natural walking poses. We can also observe that baseline methods fail to generate pose with suitable scales and positions for future poses.

4.4.3. Effect of scale-normalized error

In our thesis, We firstly propose to use scale-normalized metric to evaluate the forecasting performance of human pose in egocentric videos. Although chapter 4.4.1 introduced the score results represented by the scale-normalized error, it is still not straightforward enough to show the necessity of using scale-normalized evaluation method.

Figure 4.4 shows the comparisons on using scale-normalized error or not, to evaluate the performance. As we can notice that, without normalization of the error (result **(a)**), it causes a confusing phenomenon that the bad prediction result (left) has the same-level score with the good prediction result (right). This is mainly because the scale of human varies according to the distance between the camera and the target person. When camera is closed to the person, the pose looks larger in the frame, thus distances between bodyjoints of pose would calculate in more pixels. This would cause that even though the predicted pose looks good from our raw eyes, the value of average distance of bodyjoints in large pose size would still huge. However, after conducting scale-normalization process, we basically depress this deviation. (result **(b)**)

4.4.4. Effect of using pose-displacement

To validate the effectiveness of using pose-displacement, we let scale&location forecasting module accept pose sequence or pose-displacement sequence as a comparison experiment. The comparison results are shown in Figure 4.5. We can tell from the line chart that using pose-displacement as input would much elevate the prediction performance.

Furthermore, instead of learning features from pose sequence, we also design an experiment to input scale-location cue in scale&location forecasting module to generate the future scale and location sequence later combined with the standardized pose stream output from pose forecasting module. We simulate the scale-location

Millionseconds=600ms	With Pose-displacement	With scale-location
Pose	**0.1787**	0.1841
Scale	**6.3407**	7.1237
Location	27.4471	**26.6623**

Table 4.2.: Evaluation on the pose, the scale and the location. The pose error is calculated from standardized predicted poses and standardized ground truth poses, where we fix the location of the pose stream and normalize pose with its own scale in each frame. The scale error is obtained by calculating the deviation between ground truth scale values and the scale values measured from the predicted poses. Similarly, we set the neck joint in 2D space as location feature so we obtain the location error by calculating the displacements between neck joints of future ground truth pose sequence and predicted pose sequence.

forecasting process provided in [38]. We concatenate scale and location vector (take the length of body spine as scale and the coordinates of the neck joint as location) extracted from observed pose stream to construct scale-location stream. Before training, we normalized the scale-location sequence to zero mean and unit variance. In Table 4.2, we show the evaluation details of three-aspect granularities that we considered. Note that using scale-location stream can achieve a better location prediction result, while on the contrary, have a weaker performance on scale and pose forecasting.

Figure 4.6 shows the comparison results of predicted pose sequence with the use of pose-displacement or not. Notice all the visualized poses have been processed with location-centralization and scale-normalization. In the figure, sequence **(a)** represents the ground truth poses and sequence **(b)** represents the predicted poses of **Ours 1**. Since we proposed two kinds of networks used in pose forecasting module (**Ours 1** and **Ours 2**) that have similar performance, here we only take result of **Ours 1** as an example. The pose sequence in **(c)** represents the predicted poses output from pose forecasting module. The figure tries to show a phenomenon that for some poses with active movements of body joints such as walking with a large pace, only by modeling by standardized pose sequence with the use of pose forecasting module output pose sequence converge to the posture with a small pace. As we mentioned in the previous chapter that we generate a pose stream with ideal scale and location with the use of the pose-displacement cue in scale&location forecasting module, we observed the poses in this output sequence suffered from severe convergence problem while they retain a little sensitivity to the joint parts with active movements. According to the observation above, in the combination process, we add the middle output from scale&location forecasting module to the predicted standardized pose sequence with suitable computation. Finally, we could obtain more natural poses showed in **(b)**. The result shows the effectiveness of pose-displacement cue to capture joints variance which could also be considered to

use for strengthening the flexibility of predicted poses.

Figure 4.3.: Visualization results of predicted poses. **(a)**,**(b)**,**(c)**,**(d)**,**(e)** respectively represents the visualization results of **RRNN**, **CS2S**, **Conv-Deconv**, **Ours 1**, and **Ours 2**. Red poses represent future ground truth pose sequence and blue poses are the predicted results.

Figure 4.4.: Use scale-normalized error or not. **(a)** and **(b)** shows the difference of whether using scale-normalized error or not.

Figure 4.5.: Error comparisons with using pose-displacement or pose.. We observe obvious improvement of using pose-displacement as input sequence in Scale&Location Forecasting Module.

Figure 4.6.: Standardized pose visualization comparing with using pose-displacement (b) or not (c). Red-black skeletons represent future ground truth pose sequence and green-blue skeletons represent predicted poses.

5. Conclusion

5.1. Summary

In this thesis, we propose a novel approach for pose forecasting in egocentric videos. We first analyze the problem by considering three-aspect factors: the pose, the scale and the location of the target person. Unlike previous motion forecasting works only dealing with modeling motion dynamics, our task also has to consider additional features like variance of positions that target person might appear in the frame or the changes of the scale of the pose according to the distance between the camera and the person. We propose to use two forecasting modules for modelling motion dynamics and scale-location dynamics respectively and recombine both outputs to obtain our expected final results. We redefine the evaluation metric with adding a scale-normalization calculation to eliminate the visual error that pose scale might bring to the model. We also propose to use pose-displacement vector to learn both scale-location information of observed pose sequence and also help to capture the flexible joints movements of the target person, recombined by which, we can obtain natural looking predicted pose sequence much relieved from so called mean pose convergence problem. Both experimental scores and visualization results show the effectiveness of our approach. All the features we used in the experiment are extracted from RGB frames captured from a single wearable camera.

5.2. Limitation and Future Work

In this section, we analysis some limitations that we realized during the experiments and the insights that we obtained from the results. Also, we list up several future directions of this work.

Noise depression

For noise depression process, besides the data preprocessing (including median filtering, pose imputation) that we followed in [38], we also remove 4 head joints (two eyes, two ears) which are poorly detected, and we downsample the videos to 25 fps rather than 10 fps to have more consecutive pose sequence for better learning motion dynamics. However, we still find ground truth data itself is noisy in some

cases. To depress the pose noise, we can consider to train a network working as a filter to further smoother the pose sequence as input to the next training step.

Including more general motion situations

In this thesis, we mainly consider the case where pedestrians are walking. However, a more realistic and complex motion scenarios should take various activities into consideration. It is crucial for the network to have generalization ability on multi-motions including sitting, standing, riding, etc.. Although we observed several samples with motions above in our dataset, our pose dorecasting module does not learn these motions as much well as walking behavior. It is tough for network to learn many activities without labels help. In our future work, maybe we need to turn our eyes on other dataset like PIE (Pedestrian Intention Estimation) Dataset [33] with over 300K labeled video frames and 1842 pedestrian samples.

Incorporating depth information

As only taken 2D image coordinates of body joints as inputs and outputs, it is tough for 2D information to fully describe the location and the scale features of the pose. We need to consider measuring deep information of target pose with depth camera and map the pose data into high-dimension position. The use of depth of information might improve the performance of modeling scale and location dependencies in temporal sequence.

Acknowledgments

Foremost, I would like to express my sincere gratitude to my advisor Prof. Yoichi Sato for my Master's study. I feel really appreciated to have a chance studying under the guidance of Sato sensei. I feel I cannot go this far without his patience, motivation and immense knowledge.

Besides my advisor, I would like to thank Doc.Takuma Yagi for advising his precious thoughts of my research and details instruction about the experiment.

My sincere thanks also goes to Matsui san, for offering me kind suggestions on my graduation project as well as helping me with my Rinko-text.

I thank all of my fellow labmates in Sato Lab for 2-year accompany.

Last but not the least, I would like to thank my family members who support me materially and spiritually during my postgraduate period abroad.

A. Additional Results

In the appendix, we show more visualization results in our experiments.

Figure (A.1) shows visualization results with the use of **ours2** with **CS2S** working in pose forecasting module and **Conv-Deconv** working in scale&location forecasting module.

Figure (A.2) shows additional visual examples of standardized pose comparing with using pose-displacement or not. **(a)** represents ground truth pose; **(b)** shows predicted pose with the utilization of pose-displacement; **(c)** shows predicted pose without using pose-displacement.

Figure (A.3) shows additional visualization examples comparing our approach to baseline methods.

Figure (A.4,A.5,A.6) shows other visualization results based on our approach comparing to baseline methods. The results support the effectiveness of our separate modelling setting that highly depresses mean pose convergence problem.

Figure A.1.: Prediction Results of **Ours2**, with **CS2S** working in pose forecasting module and **Conv-Deconv** working in scale&location forecasting module. Note that the figure shows the result with the input 15 frames and output 15 frames. Predicted pose sequence supports the effectiveness of our separate modelling approach that facilitates the model to learn both motion dynamics as well as scale and location dynamics temporally without mean pose convergence problem

Figure A.2.: Additional visualization of predicted pose sequence (standardized) comparing with using pose-displacement **(b)** or not **(c)**. **(a)** represents the ground truth future pose sequence. Note that the figure shows the result with the input 15 frames (red-black skeletons) and output 15 frames (green-blue skeletons).

Figure A.3.: **(a)**, **(b)**, **(c)**, **(d)**, **(e)** respectively shows the predicted results of using baseline methods **RRNN**, **CS2S**, **Conv-Deconv**, **Ours 1**, **Ours 2**. Red poses represent ground truth and blue poses are predicted results.

Figure A.4.: Visualization results comparing with ours and baseline method. The picture shows comparing results of the last output frame.

Figure A.5.: Visualization results comparing with ours and baseline method. The picture shows comparing results of the last output frame

Figure A.6.: Visualization results comparing with ours and baseline method. The picture shows comparing results of the last output frame.

References

[1] A.Jain, S. A.R.Zamir, and A.Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, page 5308–5317. IEEE, 2016.

[2] A.Tozeren. Human body dynamics: classical mechanics and human movement. *Springer Science Business Media*, 1999.

[3] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018.

[4] C.Ionescu, D.Papava, V.Olaru, and C.Sminchisescu. Human 3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TOG*, 21(3), 2014.

[5] C.Li, W. Z.Zhang, and G.H.Lee. Convolutional sequence to sequence model for human dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 5226–5234. IEEE, 2018.

[6] D.Pavllo, D.Grangier, and M.Auli. Quaternet: Aquaternion-based recurrent model for human motion. In *British Machine Vision Conference*, 2018.

[7] D.P.Kingma and J. Ba. Adam: A method for stochastic optimization, 2014.

[8] F.S.Grassia. Practical parameterization of rotations using the exponential map. In *Journal of graphics tools*, pages 29–48, 1998.

[9] G.W.Taylor, G.E.Hinton, and S.T.Roweis. Modeling human motion using binary latent variables. *Advances in neural information processing systems*, pages 1345–1352, 2007.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[11] M. Hoai and F. D. la Torre. Max-margin early event detectors. In *IJCV*, page 191–202, 2014.

[12] H.Sidenbladh, M.J.Black, and L.Sigal. Implicit probabilistic models of human motion for synthesis and tracking. In *European Conference on Computer Vision*, page 784–800, 2002.

[13] Y. Huang, Z. L. Minjie Cai, and Y. Sato. Predicting gaze in egocentric videos by learning task-dependent attention transition. In *European Conference on Computer Vision (ECCV)*, 2018.

[14] I.Sutskever, O.Vinyals, and Q.V.Le. Sequence to sequence learning with neural networks. In *Advances in neural infor-mation processing systems*, pages 3104–4112, 2014.

[15] J.Bradbury, C. S.Merity, and R.Socher. Quasi-recurrent neural networks. In *International Conference on Learning Representation (ICLR)*, 2016.

[16] J.Butepage, D. M.J.Black, and H.Kjellstrom. Deep representation learning for human motion prediction and classification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[17] J.Gehring, D. M.Auli, D.Yarats, and Y. N.Dauphin. Convolutional sequence to sequence learning. *Proceedings of the 34th International Conference on Machine Learning(PMLR)*, 70:1243–1252, 2017.

[18] J.Martinez, M.J.Black, and J.Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*. IEEE, 2017.

[19] J.Yuen and A.Torralba. A data-driven approach for event prediction. In *ECCV*, 2010.

[20] N. Kalchbrenner, L. Espeholt, K. Simonyan, A. van den Oord, A. Graves, and K. Kavukcuoglu. Neural machine translation in linear time. In *CoRR*, 2016.

[21] K.Fragkiadaki, P. S.Levine, and J.Malik. Recurrent network models for human dynamics. In *Proceedings of theIEEE International Conference on Computer Vision*, pages 4346–4345. IEEE, 2015.

[22] T. Lan, T.-C. Chen, and S. Savarese. A hierarchical representation for future action prediction. In *ECCV*, 2014.

[23] A. M. Lehrmann, P. V. Gehler, and S. Nowozin. A non-parametric bayesian network prior of human pose. In *International Conference on Computer Vision (ICCV)*, 2013.

[24] L.Kovar and M.Gleicher. Motion graphs. *TOG*, 21(3), 2002.

[25] S. Ma, L. Sigal, and S. Sclaroff. Learning activity progression in lstms for activity detection and early detection. In *CVPR*, 2016.

[26] K. Mangalam, E. Adeli, K.-H. Lee, A. Gaidon, and J. C. Niebles. Disentangling human dynamics for pedestrian locomotion forecasting with noisy supervision, 2019.

[27] M.Brand and A.Hertzmann. Style machines. *Conference on Computer Graphics and Interactive Techniques*, pages 183–192, 2000.

[28] M. Mirza and S. Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[29] N.Kalchbrenner and P.Blunsom. Recurrent continuous translation models. *Syntax*, 3:413, 2013.

[30] O.Arikan and D.A.Forsyth. Interactive motion generation from examples. *TOG*, pages 483–490, 2002.

[31] P. Pavlovic, J. M. Rehg, and J. MacCormick. Learning switching linear models of human motion. In *NIPS*, 2000.

[32] P.Ghosh, E. J.Song, and O.Hilliges. Learning human motion models for long-term predictions. In *International Conference on 3D Vision*, pages 458–466. IEEE, 2017.

[33] A. Rasouli, I. Kotseruba, T. Kunic, and J. K. Tsotsos. Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction. In *International Conference on Computer Vision (ICCV)*, 2019.

[34] N. Rhinehart and K. M. Kitani. Online semantic activity forecasting with darko. arXiv:1612.07796, 2016.

[35] H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics,*, 22, 1951.

[36] S.Hochreiter. Hochreiter.untersuchungen zu dynamischen neuronalen netzen. *Diploma thesis,Institut f urInforma, Lehrstuhl Prof. Braueri*, 1991.

[37] S.L.Pintea, J. Gemert, and A.W.Smeulders. Motion prediction in static images. In *ECCV*, 2014.

[38] Y. Takuma, M. Karttikeya, Y. Ryo, and S. Yoichi. Future person localization in first-person videos. In *Conf. Computer Vision and Pattern Recognition*, pages 7593–7602. IEEE, 2018.

[39] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu. Pixel recurrent neural networks. In *ICML*, 2016.

[40] C. Vondrick, H. Pirsiavash, and A. Torralba. Anticipating the future by watching unlabeled video. In *CVPR*, 2016.

[41] V.Pavlovic, J.M.Rehg, and J.MacCormick. Learning switching linear models of human motion. *Advances in neural information processing systems*, pages 981–987, 2001.

[42] J. Walker, K. Marino, A. Gupta, and M. Hebert. The pose knows: Video forecasting by generating pose futures. *CoRR*, abs/1705.00053, 2017.

[43] J. Wang, A. Hertzmann, and D. M. Blei. Gaussian processdynamical models. In *NIPS*, 2005.

[44] E. Wu and H. Koike. Futurepose -mixed reality martial arts training using real-time 3d human pose forecasting with a rgb camera. In *Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019.

[45] Y. Yao, M. Xu, C. Choi, D. J. Crandall, E. M. Atkins, and B. Dariush. Egocentric vision-based future vehicle localization for intelligent driving assistance systems. *CoRR*, abs/1809.07408, 2018.

[46] Y.Hoshen and S.Peleg. An egoccentric look at video photographer identity. In *Transactions on Pattern Analysis and Machine Intelligence*, pages 4284–4292, 2016.

[47] Y.W.Chao, B. J.Yang, S.Cohen, and J.Deng. Forecasting human dynamics from sstatic images. *CVPR*, 2017.