Human Intention Prediction Using BIL-SCNN

Tianqi Ma, Lin Zhang, Xiumin Diao and Ou Ma

Abstract— **Prediction of human intention can largely boost a robot's performance to collaboratively work with humans in human-robot interaction tasks. However, human intentions are usually vague and implicit that it is even a challenging task for humans to predict others' intentions. Human actions embrace abundant behavioral information, thus is an ideal source to extract human intentions. In this paper, we propose a deep learning based approach to predict human intentions from outcomes of a ball tossing action. Based on experimentally collected sequences of RGB images of human actions and long-short term memory (LSTM), we utilized a deep neural network architecture named bidirectional LSTM sequential convolutional neural networks (BIL-SCNN), which is capable of extracting spatial and temporal features from an action sequence to predict the human action intentions. The experiment results implied 73.15% accuracy on human intention prediction with full action input and showed the predictive capability with incomplete input. The outcoming technology is potential to solve the inevitable time-delay problem for a robot to respond a human in a human-robot team by allowing the robot to act earlier based on the predicted human action.**

Index Terms— **Human-robot interaction, intention prediction, intention recognition, deep learning, LSTM**

I. Introduction

Human intention prediction is an active research area in human-robot interaction (HRI). In HRI, human actions are driven by various purposes or intentions. A robot can benefit from predicting these intentions to provide assistance ahead of time. Without the ability of predicting human intentions, however, leads to inefficiency and latency in HRI tasks. In other words, predicting human intentions in HRI tasks can solve the problem of time delaying [1][2][3]. It enables robots foreseeing possible outcomes given a sequence of human actions, even before the effective movement terminates. Intentions can be then inferred from the predicted outcomes. In this way, robots do not have to wait for the results of human actions to correctly respond. Therefore, HRI can be more efficient and less time-delaying with human intention prediction. Human intention prediction can be applied in many fields, such as intelligent traffic [4][5][6][7] and industry [8].

A challenging problem of human intention prediction is to extract intentions from humans' mind. Human actions are usually initiated with certain intentions. Through a lifetime development of motor skills, humans are trained very well to ensure the outcomes of the actions meet the initial intentions [9]. This phenomenon that how humans predict human actions inspires us that human actions are great sources to investigate human intentions. It is straightforward that different intentions lead to apparently varied actions (walk, dance, jump, etc..), however, subtly changed intentions could result in same action with little displaced outcomes. For example, how to infer a person's intention of grasping an object from the table? Obviously, we do not have to wait for the results occurring, because the action features are slightly different while doing these actions (e.g. your palm could be down if you take an orange on the table, but different when the intended object changes to a cup). To extract the subtle differences from actions driven by various intentions, researchers used high precision sensors such as wearable sensors [10] to record action details and devices to catch participants' gazes [11]. They even use biological signals [12][13] to precisely represent the actions. At the same time, multi-sensors [4] also improve the ability to extract features.

However, the high requirements for sensors limit the application and also influence the participants. A person cannot conduct action naturally with heavy sensors. Therefore, we explore the human intention prediction with only one camera and using raw RGB images as the input, which has not been widely explored before.

As for the methods of human intention prediction, researchers once modeled human intention through probabilistic graphical model (PGM) such as Markov process [8], conditional random fields [14], intention driven dynamical models (IDDM) [15] and Gaussian process dynamical models (GPDM)[16][17]. However, PGM methods are based on some hypothesis, which may bring some difficulties. For example, nonlinear actions are modeled by Gauss process, which assumes that current state is only relevant to the state of last one moment. Besides, prior probability and kernel functions need to be chosen carefully to fit the reality. Deep learning methods, on the other hand, do not rely on much assumptions and thus are more parametric. They are able to predict human action intention with an end-to-end model.

In this paper, we focus on predicting human intentions with images of action sequences using deep learning approaches. Although, our method looks similar to action recognition or action prediction [18][19][20], we train our intention predictor on dataset labeled with both human intentions and action

Tianqi Ma is with the Department of Automation, Tsinghua University, 100084 Beijing, China (e-mail: mtq19@mails.tsinghua.edu.cn).

Lin Zhang is with the Department of Aerospace and Engineering Mechanics, University of Cincinnati, Ohio 45221 USA (e-mail: zhang3l7@ucmail.uc.edu)

Xiumin Diao is with the School of Engineering Technology, Purdue University, West Lafayette, IN 47907 USA (e-mail: diaox@purdue.edu)

Ou Ma is with the Department of Aerospace and Engineering Mechanics, University of Cincinnati, Ohio 45221 USA (e-mail: maou@ucmail.uc.edu)

outcomes to strengthen the linkage between the actual intentions and the predicted results. We propose an improved sequential convolutional neural networks (SCNN) model [21] to distinguish the intentions between the similar observation in action sequences. We also evaluate our method in case of missing data. Experiments carried out on the open source dataset show that our method achieves 73.15% accuracy for full action input and over 50% accuracy with 15%-25% data missing. To conclude, the main contribution of this paper is to propose BIL-SCNN, an LSTM-based method to predict human intention from actions, and also an end-to-end method based on BIL-SCNN to achieve intention prediction with only a 15 frames per second (fps) action image sequences.

The rest of this paper is organized as follows. In Section II, we will report some related research about human action recognition, action prediction and human intention prediction. Section III introduces the details about our BIL-SCNN layer and network architecture. Then in Section IV, we will discuss the dataset and experiment implementation. The results are in Section V. Finally, summary of this paper and future work are included in Section VI.

II. Related Work

Technically, the research of human intention prediction is closely related to action recognition and action prediction because they all focus on analyzing action data collected from human. As Fig.1 illustrates, action recognition classifies the action by observing part or full of the action sequences (some researchers [22][23] suggests action recognition from part of the action, usually a few initial frames, as a new category named early detection or early recognition). Action prediction predicts future action trajectories by observing historical action sequences [14][24]. Action intention prediction predicts human intentions from the same class of action whose observed part is similar but ending is different [22][25][26].

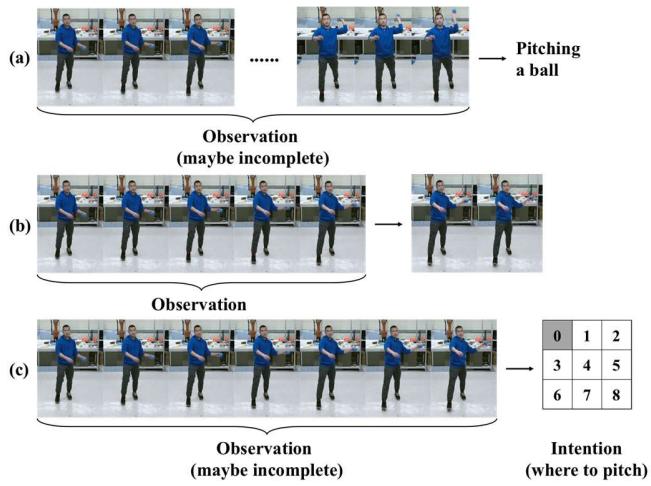


Fig.1. (a) Action recognition: recognize action labels from action image sequences (maybe incomplete). (b) Action prediction: predict future action frames from incomplete action image sequences. (c) Action intention prediction: predict the intention from action image sequences (maybe incomplete). For example, we want to know where the participant wants to pitch. Images are from dataset in [25].

Although methodologies may have common, the three categories of research are essentially different. Action recognition and prediction are to answer what the action (or future action) is, whereas intention prediction is to answer what the participant intends to do through the action.

If we regard extracting image features [27][28][29][30][31] for traditional machine learning methods as an encoding process, we can implement this process by deep learning methods. Deep learning methods are also more suitable for exploring the structure of high-dimensional data distribution. Therefore, research of human action recognition and prediction based on deep learning has also become a trend in recent years. Deep learning methods can be divided into the following four parts: 1) based on 3D convolutional neural networks (3D-CNN) [32][33], 2) based on LSTM [34][35], 3) based on two-stream method [18] and 4) based on generative adversarial networks (GAN) [36][37]. We will mainly focus on the second one.

The methods based on LSTM work for diverse data and have robustness. Researchers [38] extract features with space-time interest points (STIP) and histograms of optical flow (HOF), and achieve good recognition results even for low-quality videos. The methods based on LSTM can be roughly divided into two categories. The first one is to first extract image features, and then use LSTM for recognition / prediction. For example, Li et al. use LSTM for different types of human joint data [39]. Villegas et al. [40] divide the video into two parts: motion and content, and insert LSTM layer after the motion part. Donahue et al. [41] proposed long-term recurrent convolutional networks (LRCN) which directly encode each image in video and then recognize action with LSTM. Reference [23] and [38] respectively apply LRCN in early recognition and long video recognition. In addition, there are also improved methods under the frame of "feature extraction - LSTM". For example, [42] replaces LSTM in LRCN with a bidirectional LSTM (BI-LSTM) architecture. Reference [32] uses 3D recurrent neural networks (3D-RNN) and convolutional LSTM for hand gesture recognition. The second type of LSTM based methods is to blur the boundary between feature extraction and temporal feature processing. The SCNN model [21] can directly encode the spatial and temporal dimension of image together with convolutional neural networks (CNN).

In this paper, we will continue the research of human action intention prediction using deep learning methods with only RGB image sequences as input. We will propose an improved network architecture based on SCNN to achieve end-to-end human action intention prediction at a limited frame rate. In addition, we will show the prediction result of different lengths of the input, which is also a part to be discussed in previous research.

III. The Proposed BIL-SCNN Model

In this section, we will first introduce our proposed BIL-SCNN model to modify the SCNN model [21]. Then, we will show our end-to-end network architecture for action intention prediction based on AlexNet [43].

A. BIL-SCNN Model

SCNN [21] is a network layer model whose main idea is combining CNN and RNN in one layer, thereby maintaining the spatial structure of the input and more conducive to extracting spatial-temporal features in the video.

BI-LSTM [44] adds an extra LSTM structure in parallel on the basis of original LSTM, but in the opposite direction. Two LSTM are not affected by each other during the propagating, but the final output of each frame is jointly determined by the outputs of the two LSTM. With the addition of a backward LSTM, the temporal extracting characteristic of BI-LSTM also changes. This is because each output is jointly determined by forward LSTM and backward LSTM. LSTM is able to synthesize the current and historical input. Therefore, besides integrating historical information in forward LSTM, we can obtain "historical" information in backward LSTM, which in fact the future information. In other words, BI-LSTM is a combination of current, historical and future information.

Based on the ideas of SCNN and the good performance of BI-LSTM on LRCN model [41], we obtain a new model called BIL-SCNN (as shown in Fig.2) by using BI-LSTM as the RNN part in SCNN. The basic math principle of SCNN is as follows. Assuming that we have an action image sequences X with T frames $X = \{x_1, \dots x_T\}$, where $x_i \in \mathbb{R}^{N\times N}$, we have formulas of forward propagation for all $t = 1, \dots, T$:

$$i_t = \sigma(w_{ix} * x_t + w_{ih} * h_{t-1} + b_i) \quad (1)$$
$$f_t = \sigma(w_{fx} * x_t + w_{fh} * h_{t-1} + b_f) \quad (2)$$
$$o_t = \sigma(w_{ox} * x_t + w_{oh} * h_{t-1} + b_o) \quad (3)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(w_{cx} * x_t + w_{ch} * h_{t-1} + b_c) \quad (4)$$
$$h_t = o_t \odot tanh(c_t) \quad (5)$$

The symbols in the formula above mean as follows. First, $*$ and $\odot$ respectively represent convolution operation and Hadamard product. $tanh(\cdot)$ and $\sigma(\cdot)$ respectively represent hyperbolic tangent and activation function (Sigmoid function in general). As Fig.2 illustrates, $w_x \in \mathbb{R}^{d\times d}$, $w_h \in \mathbb{R}^{1\times 1}$ and $b \in \mathbb{R}^{n\times n}$ represent the corresponding weights and biases (the relationship between the dimensions before and after convolutional is shown in Table 1 and other operations in the formulas do not change dimensions). For each time index $t = 1, \dots, T$, $h_t \in \mathbb{R}^{n\times n}$ means hidden state, which can be regard as both output and historical information flowing in the network. $c_t \in \mathbb{R}^{n\times n}$ means the cell state. $i_t \in \mathbb{R}^{n\times n}$, $f_t \in \mathbb{R}^{n\times n}$ and $o_t \in \mathbb{R}^{n\times n}$ represent the output of memory gate, forgotten gate and output gate. At the same time, we assume $h_0 = \mathbf{0}$ and $c_0 = \mathbf{0}$ as initial condition. Similarly, we have backward propagation formulas (we use hat mark ^ to represent variables in backward LSTM to distinguish with forward LSTM):

$$\hat{i}_t = \sigma(\hat{w}_{ix} * \hat{x}_t + \hat{w}_{ih} * \hat{h}_{t+1} + \hat{b}_i) \quad (6)$$
$$\hat{f}_t = \sigma(\hat{w}_{fx} * \hat{x}_t + \hat{w}_{fh} * \hat{h}_{t+1} + \hat{b}_f) \quad (7)$$
$$\hat{o}_t = \sigma(\hat{w}_{ox} * \hat{x}_t + \hat{w}_{oh} * \hat{h}_{t+1} + \hat{b}_o) \quad (8)$$
$$\hat{c}_t = \hat{f} \odot \hat{c}_{t-1} + \hat{i}_t \odot tanh(\hat{w}_{cx} * \hat{x}_t + \hat{w}_{ch} * \hat{h}_{t+1} + \hat{b}_c) \quad (9)$$
$$\hat{h}_t = \hat{o}_t \odot tanh(\hat{c}_t) \quad (10)$$

The backward propagation starts from $t = T$ to $t = 1$ and we assume $h_{T+1} = \mathbf{0}$ and $c_{T+1} = \mathbf{0}$ as initial condition. Equation (1) to (10) together are BIL-SCNN propagation formulas. For each time index t, we will get the forward and backward LSTM outputs h_t and $\hat{h}_t$, then we have final output Out_t as

$$Out_t = (h_t + \hat{h}_t)/2 \quad (11)$$

In this way, we obtain the feature map sequence $\{Out_1, \dots, Out_T\}$ from the input sequence $X = \{x_1, \dots x_T\}$ by BIL-SCNN layer architecture, where each feature map contains current, historical and future information. Therefore, in our method, the latter part of the input sequence can affect the former part of the output, so that we have more information contained in former outputs with the same input. Through our method, we can reduce input length and improve prediction efficiency.

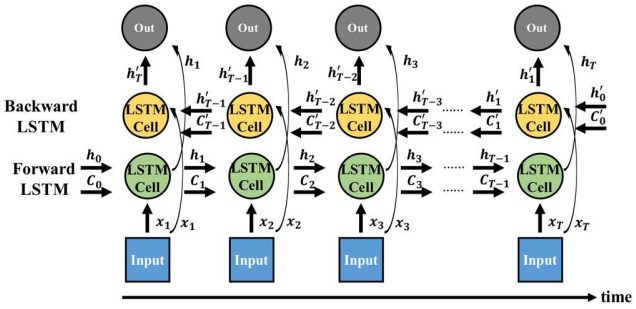


Fig.2. The general BIL-SCNN layer

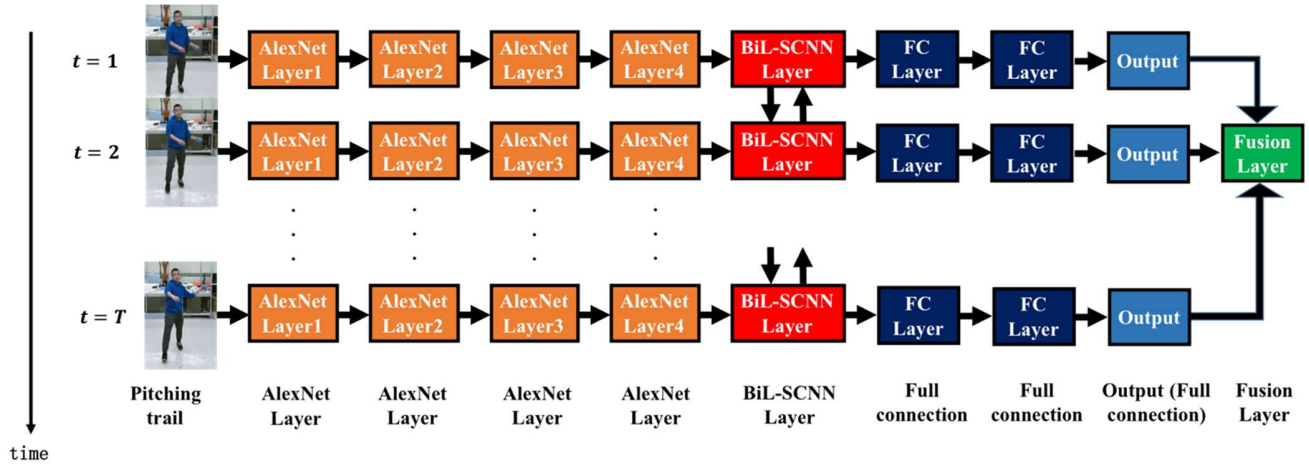


Fig.3. The architecture of modified AlexNet base on BIL-SCNN layer

B. *Modified AlexNet based on BIL-SCNN*

The original architecture of AlexNet consists of 5 convolution layers and 2 fully connection layers. It also applies ReLU, dropout and local response normalization (LRN) to improve the effect. We change the convolution layer 5 of a pre-trained AlexNet to a BIL-SCNN layer of the same size and use leaky ReLU [45] with the parameter $\alpha = 0.2$ to replace ReLU to avoid large-area inactivation of LSTM cells. Our proposed network architecture is shown in Fig.3 and Table 1. For the convenience of training, we will resize a training batch with S samples and L images for each sample to a set with $N = S \times L$ images in total. In the fusion layer, we will fuse the outputs of N images to S outputs for S samples. Average fusion is applied to calculate the final probability distribution of intentions.

TABLE I
DESCRIPTION OF BIL-SCNN BASED IMPROVED ALEXNET

Layer Number	Layer description	Output Dim
	Input	*N*×224×224×3
1	11×11 conv, 96 features, stride 4, VALID	*N*×54×54×96
	3×3 max pooling, stride 2, VALID+ LRN	*N*×26×26×96
2	5×5 conv, 256 features, stride 1, SAME	*N*×26×26×256
	3×3 max pooling, stride 2, VALID+ LRN	*N*×12×12×256
3	3×3 conv, 384 features, stride 1, SAME	*N*×12×12×256
4	3×3 conv, 384 features, stride 1, SAME	*N*×12×12×256
5	3×3 BIL-SCNN, 256 features, stride 1, SAME	*N*×12×12×256
	3×3 max pooling, stride 2, SAME	*N*×6×6×256
	flatten	*N*×9216
6	9216×4096 full connection	*N*×4096
7	4096×4096 full connection	*N*×4096
8	4096×9 full connection	*N*×9
	softmax	*N*×9
	fusion	*S*×9

If input is $N \times N$ and convolution kernel is $d \times d$, the dimension of the output $n \times n$ is dependent on the following two factors: 1) stride and 2) padding mode (SAME or VALID). Assuming that the stride is s for both x and y direction, then

$$n = \begin{cases} \text{ceil}\left(\frac{N}{s}\right), \text{if SAME} \\ \text{ceil}\left(\frac{N-d+1}{s}\right), \text{if VALID} \end{cases}$$

For more specific information, please refer to TensorFlow Document.

IV. DATASET AND EXPERIMENT

In this paper, we used the dataset proposed in [25], which contained sequential images of human participants performing ball pitching task as shown in Fig.4. The dataset collected data from 6 participants and the target was divided into 9 blocks as different sub-targets. Each participant pitched a ball to each target block 10 times. Therefore, we have 6×9×10=540 pitching trials in total. The researchers used Kinect V2 sensor (Microsoft Corporation, Redmond, WA, USA) to collect data and the frame rate was set to default (30 frames a second). Each trial lasted 3 seconds and thus had 90 frames. RGB images, depth images and anatomical joint positions were all collected.

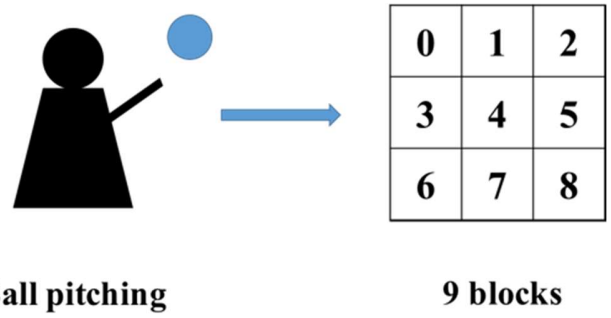


Fig.4. Experiment scenario in [25]

A. *Human Intention-emphasized Dataset*

Research of action recognition and action prediction usually employ the dataset with multiple types of discerned actions and labeled only with outcomes of the actions [46][47][48]. Since we aim at predicting human intentions not action outcomes, we have to make sure the outcome of a pitching trial exactly matching to the pitcher's initial intention. Therefore, dataset in [25] is a proper one because a double-validated labeling method is applied to only keep the data with matched intention and outcome. A warm-up session is also introduced for each pitcher to practice throwing the ball to the target on purpose. The pitcher has to secure at least one successful pitching for every target. In each recorded trial,

1) Computer picks target randomly and inform the target to the pitcher before the trial started.
2) The pitcher listens to the auditory cue and initiate throwing. The pitcher has to stand still before initiating the throwing. The initial pose is not mandatory, hence can be determined at pitcher's preference.
3) The pitcher throws the ball toward the pre-noticed target within 5 seconds. The pitcher is not allowed to perform any irrelevant action during and after throwing. We recommend the pitcher to return to the initial pose.
4) Unless the pitcher or computer operator was not satisfied with the trial, recorded data will be saved.
5) Repeat step 1 through 4 until the data collection is finished.

To perform the double-validated labeling, we discarded data in the trials that the pitcher throws the ball into the target that was not intended and kept data in the trials that the ball was thrown into the initially intended target. Due to the large amount of data is required, the pitchers are allowed to separate experiment into several sessions to avoid fatigue.

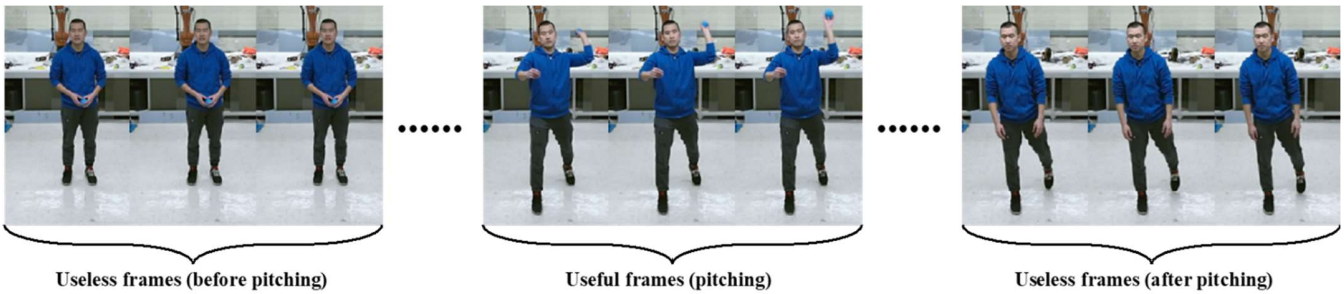


Fig.5. Useless frames before and after action

Before training, we need to first determine the effective action interval. This is because the original data in one trial contains useless frames before and after (as shown in Fig.5). Since multiple participants were involved and they had different motion patterns, the starting frame of each trial varied from person to person. It is necessary to calculate the starting frame for each trial instead of a fixed starting frame for all trials. A feasible way proposed in [25], which we call ***L2 distance method***, is based on joint positions. At time t, a pose θ_t is described as space coordinates of 25 body joints, i.e. $\theta_t = [\theta_{1t}, \dots, \theta_{25t}]$. Thus, for a trial from $t = 1$ to $t = T$, L2 distance method is as follows:

1) Calculate $\Delta\theta = [\Delta\theta_1, \dots, \Delta\theta_T]$, where $\Delta\theta_t = \|\theta_t - \theta_1\|_2, t = 1, \dots, T$.
2) Set a threshold $\delta = 0.16$. Find minimum i which satisfies a) $\Delta\theta_t \geq \delta, t = i, i+1, \dots i+19$ and b) $\Delta\theta_{t+1} \geq \Delta\theta_t, t = i, \dots i+18$.
3) Set frame i as the starting frame.
4) If no frame satisfies a) and b) in 2), set frame 45 as the starting frame.

We also use RGB images as another starting frame detection method, the ***Combination method***, to avoid local minimum in L2 distance method. We assume that the pitching does not start in the first 10 frames, which is reasonable on the dataset. For a trial with joint positions θ_t and normalized, central-cropped RGB images x_t from $t = 1$ to T, we have

1) Calculate $\Delta\theta = [\Delta\theta_{11}, \dots, \Delta\theta_T]$, where $\Delta\theta_t = \|\theta_t - \theta_{10}\|_2, t = 11, \dots, T$ and $\Delta x = [\Delta x_{11}, \dots, \Delta x_T]$, where $\Delta x_t = \|x_t - x_{10}\|_F, t = 11, \dots, T$.
2) Set thresholds $\delta_1 = 6$ and $\delta_2 = 0.16$. Find minimum i which satisfies a) $\Delta x_t \geq \delta_1, t = i, \dots, i+7$, b) $\Delta x_{t+1} \geq \Delta x_t, t = i, \dots, i+6$ and c) $\Delta\theta_i \geq \delta_2$.
3) Set frame i as the starting frame.

Experiment results show that all trials have a frame which satisfies a), b) and c) in 2).

We use a 40-frame interval from the starting frame for training and testing. In this paper, we will use both two methods to determine the starting frame for each trial.

B. Experiment Implementation

Network inputs. After selecting, each trial includes 40 frames, which is still too much for training, thus we need to sample frames from each trial. In our experiments, we choose frame 1, 3, 5, …39 and thus limit to 20 frames in a trial as input. Effectively, this equals to K-interval sampling [25] with K=20 and choose the first frame in each interval, but differently, we drop the other frames for both training and testing. For each image, we crop with a size 224×224 from the central part of the images. Finally, we regularize the pixels to [0,1].

Hyper-Parameter. We set the learning rate as 0.0001 and the batch-size as 27, which means an epoch has 16 batches. Dropout rate is 0.5 and other hyper-parameters are set the same as [43]. We choose Adam optimizer [49] to train our networks.

Test. We choose 8 out of 10 trials of each participant as the training set, and the rest 2 are serving as the testing set. Therefore, the size of training set and testing set is 432 and 108, respectively.

V. Results

A. Training results

We conducted experiments using TensorFlow v1.0 on a computer running Windows 10 with an Intel i9-9900kf CPU, 16 GB RAM, and an NVIDIA 2080Ti GPU card. We trained our networks with the two methods mentioned in Section IV-A and the results are shown in Table 2 and Fig.6.

TABLE II
TRAINING RESULTS

First frame detection method	Epoch	Training time	Average training time
L2 distance method	60	5784 s	96.4 s
Combination method	46	4461 s	97.0 s

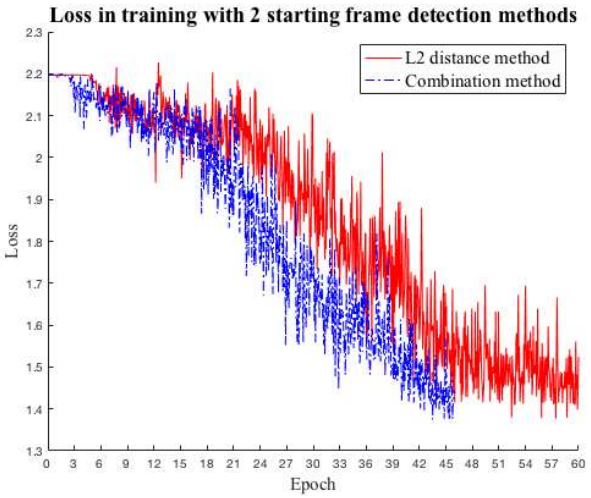


Fig.6. Loss in training with 2 starting frame detection methods

We find out that the average training time of an epoch with both methods is about 100 s, but we can shorten the total training time by 23.3% (from 60 epochs to 46 epochs) by the combination method, which can be explained that we obtain more qualified data with more obvious regularity, so it's more likely to train to convergence.

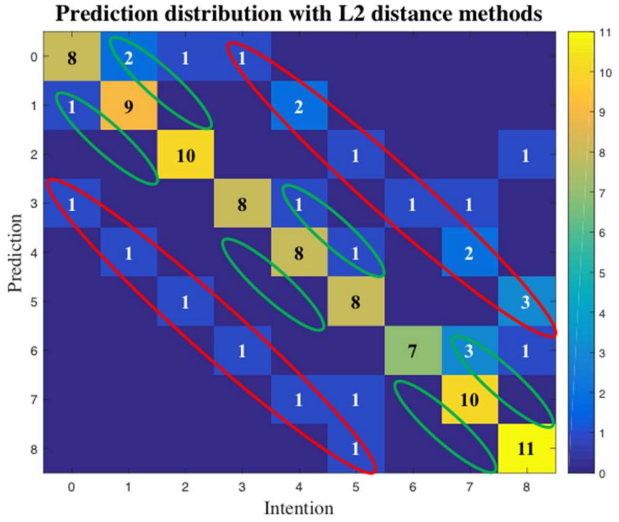
B. Accuracy on full actions

For full action intention prediction, we compare our method with those proposed in [25], as shown in Table 3. First, we can see that our intention prediction accuracy (73.15%) is significantly better than the Spatial only network (63.89%) and fusion networks (71.3%) in [25]. The Temporal networks (74.07%) were doing slightly better than our method. We are confident to claim that our proposed method dose effectively extract time feature in the pitching trials. Second, we achieved better results than the Spatial-Temporal fusion network in [25] but slightly worse than Temporal-only networks. We infer the reason is that for pitching actions, spatial information is almost the same in all trials. Therefore, temporal differences are more important in this task.

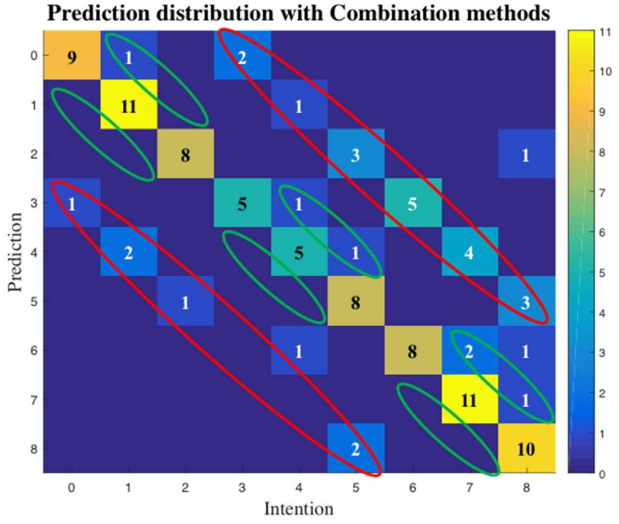
Although we can shorten training time by the combination detecting method, the intention prediction result is slightly worse than L2 distance method. We infer that the reason is that the data constructed by combination method leads to slight overfitting because the combination method shows the relationship between pitching trials and intentions.

Furthermore, the distributions of the intention prediction of our methods are shown in Fig.7. First, through the comparison of Fig.7a and 7b, we can see that the network with the combination method results in imbalance on intention prediction. For example, the prediction accuracy of intention 1 and 7 is over 2 times of the accuracy of intention 3 and 4. However, this situation does not occur if we use L2 distance method to detect first frame, which also illustrates the overfitting in the former method.

We can find out that for a certain intention, the majority of misprediction results is the neighborhood of the ground-truth intention area. We calculate from Fig.7 that the error rate of combination method and L2 distance method is 7.41% and 5.56%, respectively in the horizontal direction, as well as 14.81% and 22.22% in the vertical direction, respectively. This illustrates that the accuracy of our proposed network has higher accuracy in the horizontal direction than the vertical direction, which means that our network is more sensitive to changes in the horizontal pitching intention. The possible reason is related to the camera shooting angle. Because the camera is placed directly in front of the participant, changes in the horizontal direction are reflected obviously in the image as the difference between foreground and background, whereas the difference mainly reflected in the foreground for changes in the vertical direction, thus more difficult to predict.



(a)



(b)

Fig.7. Prediction distribution with different starting frame detection methods. (a) is with L2 distance method and (b) is with combination method. Black numbers on the diagonal are the numbers of correct prediction for each intention. White numbers in the green circles are the numbers of errors for horizontal neighborhood, and numbers in red circles for vertical neighborhood (grids with no number mean 0).

TABLE III
EVALUATION RESULTS

Network	Data Augmentation	Starting frame detection method	Prediction Accuracy
Best in [25] (Spatial only)	Yes	L2 distance method	63.89%
Best in [25] (Temporal only)	Yes	L2 distance method	74.07%
Average Fusion in Spatial and Temporal in [25]	Yes	L2 distance method	71.3%
BIL-SCNN based AlexNet	No	L2 distance method	73.15%
BIL-SCNN based AlexNet	No	Combination method	69.44%

C. Prediction ability on incomplete actions

We evaluate the prediction ability on incomplete pitching trial of our proposed networks. We use the networks trained in Section V-A and the intention prediction accuracy on incomplete actions with different length is shown in Fig.8.

From Fig 8 we can see that the intention prediction accuracy decreases with the increase of missing data of the pitching trial. Our proposed networks keep over 50% accuracy with about 15%-25% missing of the input. When the length is reduced to about 1/4 of the original action (5 frames), all pitching trials basically lose their characteristics, and the intentions cannot be effectively predicted by the networks. As for the comparison of the two first frame detection methods, the accuracy of the combination method is lower than the L2 distance method, and also decreases faster with the increasing of the missing data. This indicates that the action characteristics of dataset conducted by the combination method is lost more quickly than the L2 distance method with the increase of the missing data, which means a more concentrated action feature distribution.

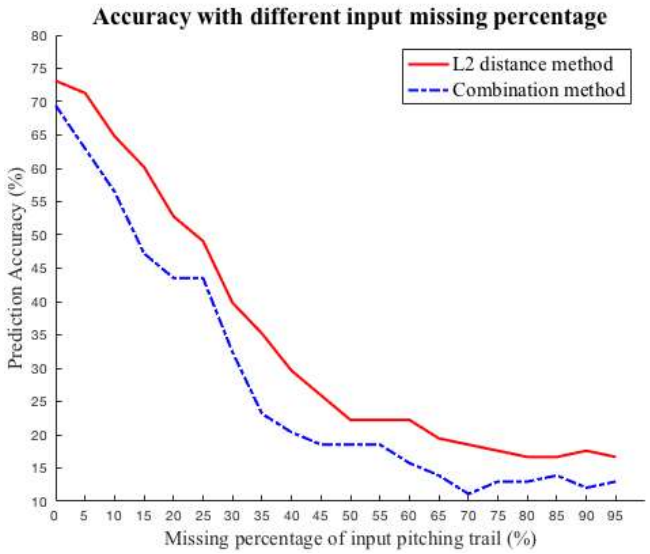


Fig.8. Changes of prediction accuracy for incomplete pitching trials

VI. Conclusion and Future Work

In this paper, we achieved human action intention prediction using a human intention-emphasized dataset and an LSTM-based deep neural network architecture, named BIL-SCNN. A modified end-to-end model based on AlexNet was integrated and tested. We propose a frame detection method combining RGB images and joint L2 distance and evaluated our method with two starting frame detection methods and on both full and incomplete action sequences. Experiment results showed that for full action intention prediction, BIL-SCNN achieved a maximum accuracy of 73.15%. We also found out that our networks had different sensitivity in different directions, resulting in higher action intention prediction accuracy in horizontal direction. Finally, we showed the intention prediction result for incomplete action inputs.

In future work, we will address on the following questions:

1) How to improve the accuracy of action intention prediction in the vertical direction? A possible way to achieve this is to change the shooting angle of the camera to balance the data, or we can add data from multi-view sensors.
2) Is there any better network architecture with BIL-SCNN? For example, where and how to better fuse the output from each input frame? Will more BIL-SCNN layer brings better performance? We may design and try some different network architectures. For example, using softmax of the last layer as in [42] is worth trying.
3) The dataset only records trials that satisfies the intentions of participants. However, intentions do not always lead to the corresponding action outcomes. We can enlarge the dataset and further analyze the relationship between intentions and action outcomes, which helps us to apply it to real robots in HRI tasks.

References

[1] E. Wang, "User's Delay Perception and Tolerance in Human-Computer Interaction," *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 46, no. 5, pp. 651–655, 2002, doi: 10.1177/154193120204600511.

[2] A. Steinfeld *et al.*, "Common metrics for human-robot interaction," *HRI 2006 Proc. 2006 ACM Conf. Human-Robot Interact.*, vol. 2006, no. March, pp. 33–40, 2006, doi: 10.1145/1121241.1121249.

[3] C. Von Hardenberg and F. Bérard, "Bare-hand human-computer interaction," *ACM Int. Conf. Proceeding Ser.*, vol. 15-16-Nove, p. 29, 2001, doi: 10.1145/971478.971513.

[4] A. Jain, H. S. Koppula, S. Soh, B. Raghavan, A. Singh, and A. Saxena, "Brain4Cars: Car That Knows Before You Do via Sensory-Fusion Deep Learning Architecture," Jan. 2016.

[5] Y. Zhang, Q. Lin, J. Wang, S. Verwer, and J. M. Dolan, "Lane-Change Intention Estimation for Car-Following Control in Autonomous Driving," *IEEE Trans. Intell. Veh.*, vol. 3, no. 3, pp. 276–286, Jun. 2018, doi: 10.1109/tiv.2018.2843178.

[6] S. Yoon, H. Jeon, and D. Kum, "Predictive Cruise Control Using Radial Basis Function Network-Based Vehicle Motion Prediction and Chance Constrained Model Predictive Control," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3832–3843, Oct. 2019, doi: 10.1109/TITS.2019.2928217.

[7] P. Gebert, A. Roitberg, M. Haurilet, and R. Stiefelhagen, "End-to-end prediction of driver intention using 3D convolutional neural networks," in *IEEE Intelligent Vehicles Symposium, Proceedings*, 2019, vol. 2019-June, pp. 969–974, doi: 10.1109/IVS.2019.8814249.

[8] T. Petković, D. Puljiz, I. Marković, and B. Hein, "Human intention estimation based on hidden Markov model motion validation for safe flexible robotized warehouses," *Robot. Comput. Integr. Manuf.*, vol. 57, pp. 182–196, Jun. 2019, doi: 10.1016/j.rcim.2018.11.004.

[9] V. G. Payne and L. D. (Larry D. Isaacs, *Human motor development : a lifespan approach*. McGraw-Hill, 2008.

[10] W. Wang, R. Li, Y. Chen, and Y. Jia, "Human Intention Prediction in Human-Robot Collaborative Tasks," in *ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 279–280, doi: 10.1145/3173386.3177025.

[11] B. Hu, X. Liu, W. Wang, R. Cai, F. Li, and S. Yuan, "Prediction of interaction intention based on eye movement gaze feature," in *Proceedings of 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference, ITAIC 2019*, 2019, pp. 378–383, doi: 10.1109/ITAIC.2019.8785737.

[12] L. Bi, A. Feleke, and C. Guan, "A review on EMG-based motor intention prediction of continuous human upper limb motion for human-robot collaboration," *Biomedical Signal Processing and Control*, vol. 51. Elsevier Ltd, pp. 113–127, 01-May-2019, doi: 10.1016/j.bspc.2019.02.011.

[13] H. Wang, L. Bi, and T. Teng, “EEG-based emergency braking intention prediction for brain-controlled driving considering one electrode falling-off,” in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2017, pp. 2494–2497, doi: 10.1109/EMBC.2017.8037363.

[14] J. Bütepage, M. J. Black, D. Kragic, and H. Kjellström, “Deep representation learning for human motion prediction and classification,” in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017, vol. 2017-January, pp. 1591–1599, doi: 10.1109/CVPR.2017.173.

[15] Z. Wang *et al.*, “Probabilistic movement modeling for intention inference in human-robot interaction,” in *International Journal of Robotics Research*, 2013, vol. 32, no. 7, pp. 841–858, doi: 10.1177/0278364913478447.

[16] J. M. Wang, D. J. Fleet, and A. Hertzmann, “Gaussian process dynamical models for human motion,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 283–298, Feb. 2008, doi: 10.1109/TPAMI.2007.1167.

[17] Z. Wang, A. Boularias, K. Mülling, B. Schölkopf, and J. Peters, “Anticipatory action selection for human–robot table tennis,” *Artif. Intell.*, vol. 247, pp. 399–414, Jun. 2017, doi: 10.1016/j.artint.2014.11.007.

[18] K. Simonyan and A. Zisserman, “Two-stream convolutional networks for action recognition in videos,” in *Advances in Neural Information Processing Systems*, 2014, vol. 1, no. January, pp. 568–576.

[19] T. Lan, T. C. Chen, and S. Savarese, “A hierarchical representation for future action prediction,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, vol. 8691 LNCS, no. PART 3, pp. 689–704, doi: 10.1007/978-3-319-10578-9_45.

[20] Y. Kong and Y. Fu, “Human Action Recognition and Prediction: A Survey,” *arXiv:1806.11230*, Jun. 2018.

[21] H. Yang, C. Yuan, J. Xing, and W. Hu, “SCNN: Sequential convolutional neural network for human action recognition in videos,” in *Proceedings - International Conference on Image Processing, ICIP*, 2018, vol. 2017-September, pp. 355–359, doi: 10.1109/ICIP.2017.8296302.

[22] A. Zunino, J. Cavazza, A. Koul, A. Cavallo, C. Becchio, and V. Murino, “Predicting human intentions from motion cues only: A 2D+3D fusion approach,” in *MM 2017 - Proceedings of the 2017 ACM Multimedia Conference*, 2017, pp. 591–599, doi: 10.1145/3123266.3123298.

[23] S. Ma, L. Sigal, and S. Sclaroff, “Learning Activity Progression in LSTMs for Activity Detection and Early Detection,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vol. 2016-December, pp. 1942–1950, doi: 10.1109/CVPR.2016.214.

[24] J. Martinez, M. J. Black, and J. Romero, “On human motion prediction using recurrent neural networks,” in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017, vol. 2017-January, pp. 4674–4683, doi: 10.1109/CVPR.2017.497.

[25] S. Li, L. Zhang, and X. Diao, “Deep-Learning-Based Human Intention Prediction Using RGB Images and Optical Flow,” *J. Intell. Robot. Syst. Theory Appl.*, vol. 97, no. 1, pp. 95–107, Jan. 2020, doi: 10.1007/s10846-019-01049-3.

[26] S. Li, L. Zhang, and X. Diao, “Improving Human Intention Prediction Using Data Augmentation,” in *RO-MAN 2018 - 27th IEEE International Symposium on Robot and Human Interactive Communication*, 2018, pp. 559–564, doi: 10.1109/ROMAN.2018.8525781.

[27] I. Laptev, “On Space-Time Interest Points,” *Int. J. Comput. Vis.*, vol. 64, no. 2–3, pp. 107–123, Sep. 2005, doi: 10.1007/s11263-005-1838-7.

[28] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004, doi: 10.1023/B:VISI.0000029664.99615.94.

[29] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, 2005, vol. I, pp. 886–893, doi: 10.1109/CVPR.2005.177.

[30] N. Dalal, B. Triggs, and C. Schmid, “Human Detection Using Oriented Histograms of Flow and Appearance,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3952 LNCS, 2006, pp. 428–441.

[31] H. Wang and C. Schmid, “Action recognition with improved trajectories,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3551–3558, doi: 10.1109/ICCV.2013.441.

[32] G. Zhu, L. Zhang, P. Shen, and J. Song, “Multimodal Gesture Recognition Using 3-D Convolution and Convolutional LSTM,” *IEEE Access*, vol. 5, pp. 4517–4524, 2017, doi: 10.1109/ACCESS.2017.2684186.

[33] S. Ji, W. Xu, M. Yang, and K. Yu, “3D Convolutional neural networks for human action recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, 2013, doi: 10.1109/TPAMI.2012.59.

[34] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[35] A. Grushin, D. D. Monner, J. A. Reggia, and A. Mishra, “Robust human action recognition via long short-term memory,” in *Proceedings of the International Joint Conference on Neural Networks*, 2013, doi: 10.1109/IJCNN.2013.6706797.

[36] L. Y. Gui, K. Zhang, Y. X. Wang, X. Liang, J. M. F. Moura, and M. Veloso, “Teaching Robots to Predict Human Motion,” in *IEEE International Conference on Intelligent Robots and Systems*, 2018, pp. 562–567, doi: 10.1109/IROS.2018.8594452.

[37] I. J. Goodfellow *et al.*, “Generative adversarial nets,” *Adv. Neural Inf. Process. Syst.*, vol. 3, no. January, pp. 2672–2680, 2014, doi: 10.3156/jsoft.29.5_177_2.

[38] J. Y. H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, “Beyond short snippets: Deep networks for video classification,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015, vol. 07-12-June-2015, pp. 4694–4702, doi: 10.1109/CVPR.2015.7299101.

[39] C. Li, P. Wang, S. Wang, Y. Hou, and W. Li, “Skeleton-based action recognition using LSTM and CNN,” in *2017 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2017*, 2017, pp. 585–590, doi: 10.1109/ICMEW.2017.8026287.

[40] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee, “Decomposing Motion and Content for Natural Video Sequence Prediction,” *5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc.*, Jun. 2017.

[41] J. Donahue *et al.*, “Long-Term Recurrent Convolutional Networks for Visual Recognition and Description,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 677–691, Apr. 2017, doi: 10.1109/TPAMI.2016.2599174.

[42] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, “Action Recognition in Video Sequences using Deep Bi-Directional LSTM with CNN Features,” *IEEE Access*, vol. 6, pp. 1155–1166, Nov. 2017, doi: 10.1109/ACCESS.2017.2778011.

[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, Jun. 2017, doi: 10.1145/3065386.

[44] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, 1997, doi: 10.1109/78.650093.

[45] B. Xu, N. Wang, T. Chen, and M. Li, “Empirical Evaluation of Rectified Activations in Convolutional Network,” May 2015.

[46] K. Soomro, A. R. Zamir, and M. Shah, “UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild,” *arXiv:1212.0402*, Dec. 2012.

[47] L. Xia, C. C. Chen, and J. K. Aggarwal, “View invariant human action recognition using histograms of 3D joints,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 20–27, doi: 10.1109/CVPRW.2012.6239233.

[48] W. Kay *et al.*, “The Kinetics Human Action Video Dataset,” *arXiv:1705.06950*, May 2017.

[49] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.