ONLINE ACTION DETECTION AND FORECAST VIA MULTITASK DEEP RECURRENT NEURAL NETWORKS

*Chunhui Liu, Yanghao Li, Yueyu Hu, Jiaying Liu**

Institiude of Computer Science and Technology, Peking University, Beijing, China

ABSTRACT

Online human action detection and forecast on untrimmed 3D skeleton sequences is a novel task based on traditional action recognition and has not been fully studied. Its aim is to localize and recognize one action in a long sequence while doing forecasting task at the same time. In this paper, we propose an online detection algorithm featuring Multi-Task Recurrent Neural Network to solve this problem. First, a deep Long Short Term Memory (LSTM) network is designed for feature extraction and temporal dynamic modeling. Then we utilize a classification subnetwork to classify one action, and predict the status of it at the same time. To forecast the occurrence of actions and estimate the accurate time of occurrence, we incorporate a regression subnetwork to our model. Then we split the action classes to three stages and train the model by optimizing a joint classification regression objective function. Experimental results show that the proposed model achieves satisfactory results on online action detection and forecast.

***Index Terms*—** Online Action Detection, Online Action Forecast, Recurrent Neural Network

1. INTRODUCTION

Human action detection remains a challenging problem in the field of computer vision, facilitating a wide range of practical applications like video surveillance, video understanding and human-computer interaction. For these applications, the ability to detect or even predict human's actions can greatly improve the functionality, especially in designing intelligent robot system. Thus, online human action detection and forecast is developed, taking both the accuracy and the latency into consideration.

Action recognition and detection problem has been studied through the past decade and several methods have been proposed. For action recognition, by tracking feature points in consecutive frames and encoding the extracted trajectories with bag-of-feature (BOF) technique [1], motion of objects in thecitedeo can be captured and well recognized. This method was lately improved by densely tracking points in the optical flow field while adding more features and encoding with Fisher Vector [2, 3]. For action detection, besides accurate classification, the temporal information is also needed. Existing works utilize a variety of methods like sliding-window scheme [4, 5, 6, 7] or action proposal approaches [8, 9].

Recently, the developments of Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) show great potentials in the field of action recognition [10, 11]. The combination of traditional hand-craft features and CNN extracted features [12, 13] shows improvement in classification performance. A Long-term Recurrent Convolutional Network (LRCN) is proposed for activity recognition [14] and early activity detection [15].

In order to capture more information from the objects, there is a trend of adding depth camera to the sensors. One important feature for the additional depth camera is the ability to extract the skeleton from human body. Skeleton, as a kind of high level representation of human body, can provide valuable information for recognizing actions [16]. For skeleton based action recognition, hierarchical RNN [17] and fully connected Long Short Term Memory (LSTM) [18] are investigated to model the temporal dynamics. In [19], a skeleton-data based Joint Classification Regression RNN was designed to solve online action detection and forecasting problem with a Gaussian-like curve to describe the confidence. However, it only achieved a weak prediction to locate the time points in an interval, but not the exact time point.

In this paper, we propose a Multi-Task Recurrent Neural Network to localize the start and end positions of the action and forecast the accurate action occurrence from the streaming skeleton data. The architecture is shown in Fig.1. To be specific, LSTM layers are used to extract video features and perform long-range temporal dynamics modeling automatically. Here, our network is end-to-end trainable by optimizing a joint objective function of classification task for detection and regression task for forecast. On one hand, the classification subnetwork is designed to classify the frame-wise action so that the action interval can be localized online. On the other hand, the regression subnetwork aims to forecast the specific start and end point of actions by learning the countdown of the action. Additionally, to evaluate our method, we formulate some new evaluation protocols and define two standard

∗Corresponding author

This work was supported by National Natural Science Foundation of China under contract No. 61472011 and Microsoft Research Asia (project ID FY17-RES-THEME-013).

curves for the comparison.

The rest of this paper is organized as follows. Sec.2 formulates the problem of online action detection and forecast. In Sec.3, we introduce the proposed Multi-Task Recurrent Neural Network. Experimental results are shown in Sec.4 and concluding remarks are given in Sec.5.

2. OVERVIEWS ON ONLINE ACTION DETECTION AND FORECAST

In this section, we overview the formulation of online action detection and action forecast which is first illustrated in [19]. Since this new problem is far from solving, to clarify the difference from other problems is of great significance.

2.1. Online Action Detection

We define M kinds of actions in all the videos. Given a video observation $V = \{v_0, ..., v_{N-1}\}$ composed of N frames, offline action detection which uses full video segment to detect the human action is formulated as

$$A = \bigcup_i \{(s_i, e_i, k_i) \,|s_i < e_i < N\}, \tag{1}$$

where A is the action set, s_i, e_i and k_i is the start point, the end point, and the class of action i occurred in the video.

Different from offline action detection, online action detection aims to locate the start point and the end point synchronously according to the previous video segment and determine the action class underway. One similar work called *early detection* has been study in [15] which aims to classify the action as soon as possible. For online action detection, the main idea is to perform classification and segment detection at one time. Thus, the online detection method must estimate the starting point, the end point and status of the current action automatically. The problem is formulated as below,

$$A_t = \bigcup_i \{(s_i, e_i, k_i) \,|e_i \leqslant t\} \cup \{(s, t, k)\}, \tag{2}$$

which is to calculate A_t for every $t < N$, where s is the start point of action k happened in time t.

To achieve the goal of online processing, the interval (s_i, e_i, k_i) is added into set A_t one by one. Consequently, this problem can be transformed into a classification task to calculate $\mathbf{y}_t^*$ as,

$$\mathbf{y}_t^* = \max_t P(\mathbf{y}_t \,|v_0, ...v_t), \tag{3}$$

where $\mathbf{y}_t \in R^{1\times(M+1)}$ is the possible action label vector for frame v_t. For *Background* which cannot be classified as a member of M classes, we set up one additional class to label it.

2.2. Online Action Forecast

Besides online action detection, one forecasting process to decide how soon the objective action will take place can also be running at the same time. Online action forecast aims to predict the start and the end point of the action. A weak forecast of probability that one action is about to happen in time T is formulated as

$$(\mathbf{y}_t^*, \mathbf{s}_t^*, \mathbf{e}_t^*) = \max_{\mathbf{y}_t, \mathbf{s}_t, \mathbf{e}_t} P(\mathbf{y}_t, \mathbf{s}_t, \mathbf{e}_t \,|v_0, ...v_t), \tag{4}$$

where $\mathbf{y}_t, \mathbf{s}_t, \mathbf{e}_t \in R^{1\times(M+1)}$ are three vectors deciding which action is in progress, starting or ending. For example, $\mathbf{s}_{t,j} = 1$ suggests that

$$P_t(t \leqslant s_j \leqslant t + T \,|v_0, ..., v_t) \geqslant \theta_s, \tag{5}$$

where s_j is the start point of action j and θ_s is an constant threshold value.

To further predict the time point accurately is a more challenging task defined as to determine

$$(\mathbf{F}_t^*, \mathbf{L}^*) = \max_{\mathbf{F}_t, \mathbf{L}} P(\mathbf{F}_t, \mathbf{L} \,|v_0, ...v_t), \tag{6}$$

where

$$\mathbf{F}_t = \{\mathbf{y}_t, \mathbf{s}_t, \mathbf{e}_t\},$$

and $\mathbf{L}$ describe the exact number of frames to the start or end points.

3. PROPOSED MULTITASK RNN FOR ONLINE DETECTION AND FORECAST

Leveraging the insights from previous works, we propose a new end-to-end Multi-Task RNN framework based on Joint Classification-Regression RNN, whose architecture is shown in Fig.1. It contains three major components: three stacked LSTM layers, one classification subnetwork and a regression subnetwork. LSTM is an advanced RNN network which is widely used for extracting frame level action features and learning the long-range dependencies of a temporal sequence. To be specific, the network is constructed by three LSTM layers and three non-linear fully-connected (FC) layers. Those FC layers act as a role of feed-forward layers for robust feature learning. A regression loss together with a classification loss is added into the learning objective, in order to perform action interval locating and explicit action forecasting:

$$\mathcal{L}(V) = \mathcal{L}_c(V) + \lambda \mathcal{L}_r(V), \tag{7}$$

where $\mathcal{L}_c(V)$ and $\mathcal{L}_r(V)$ are the classification loss and the regression loss respectively. λ is constant value.

In the following, we will briefly introduce the multi-task subnetworks for online action detection and forecast.

3.1. Classification Task

The classification task subnetwork is designed for recognizing action interval by labeling each frame. We split an action class into three parts: k_b, k_p and k_e for *action about to begin*, *in progress* and *about to end* respectively. For example, class

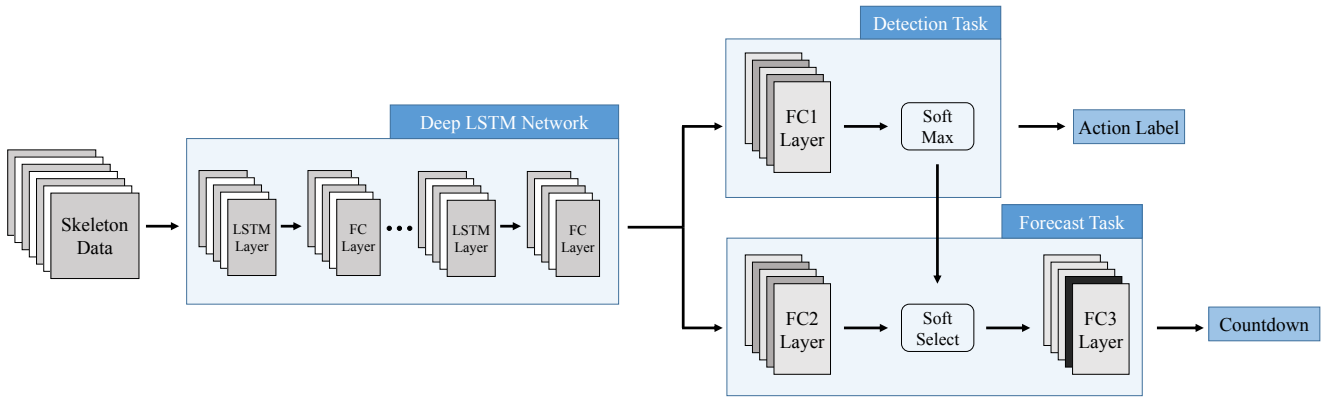

Fig. 1. Architecture of the proposed multi-task RNN framework for online detection and forecast.

k_b indicates an action is to occur soon and the end of it indicates the start of an action. We consider it necessary to train the model in a stronger supervision for the forecasting problem. Thus, we carefully refine data with tripling the number of labels and the loss function of classification task is defined as

$$\mathcal{L}_c(V) = -\frac{1}{N}\sum_{t=0}^{N-1}\sum_{k=0}^{3\times M} z_{t,k}\ln P(y_{t,k}|v_0,...v_t), \quad (8)$$

where M is the number of all action classes, and $z_{t,k}$ corresponds to the score of frame v_t for class k. When k is the ground truth label, $z_{t,k}$ sets to 1. For the other action classes, the score is zero.

The structure of this subnetwork is shown on the upper part of Fig.1. Deep LSTM network is designed to model the spatial structure and to extract the temporal dynamics features of current frame. A fully-connected layer FC1 with a SoftMax activation is added for the classification task.

3.2. Regression Task

The regression task subnetwork is designed for forecasting the exact positions of action. To perform the forecast task, an alternative is using Gaussian-like curve to describe the confidence:

$$c_t^s = exp\left(\frac{-(t-s_j)^2}{2\sigma^2}\right), \quad (9)$$

where s_j is the start point of the nearest action j to the frame v_t, and σ is a user-defined value. However, a constant value for σ is unappropriate because the properties of different actions may differ. For example, some actions like hitting sandbags or short-distance dash usually occur abruptly, while others may occur gently. Thus, it is essential to change σ dynamically.

Consequently, We define L_t as the ground truth distance between frame t and next action point:

$$L_t = min\{(s_j - t), (e_j - t)|s_j, e_j > t\}, \quad (10)$$

where s_i and e_j is time point after frame t.

This definition has two advantages. One is that the distance can be easily calculated without variable σ. For the other, the forecasting problem is transformed into a linear regression in which specific distance can be calculated more easily than a regression of Gaussian-like model. Thus, the regression problem can be seen as the optimization problem of the following loss function

$$\mathcal{L}_r(V) = -\frac{1}{N}\sum_{t=0}^{N-1}\|L_t - p_t\|^2, \quad (11)$$

where p_t is the prediction distance to next action point.

As shown in Fig.1, we achieve the specific forecasting work by constructing the regression work subnetwork with a non-linear fully-connected layer FC2, a Soft Selector layer, and a non-linear fully-connected layer FC3. The Soft Selector layer is designed to indicate the output of SoftMax layer in classification network to select the information of regression by mapping the value of paraments in FC2. We train the network using stochastic gradient descent with momentum and Back Propagation Through Time (BPTT) . One dropout regularization is utilized on LSTM fully-connect layers to prevent over-fitting.

4. EXPERIMENTAL EVALUATION AND RESULT

4.1. Dataset and Setting

We evaluate the performance of our approach for online action detection and forecasting with refined **Online Action Detection Dataset (OAD)**. **OAD** is a new dataset featuring Kinect v2 sensors which collects color images, depth images and human skeleton joints synchronously. It contains 59 videos (216 minutes in total) of 10 common action like *drinking water* and *washing hands*.

As shown in Fig.1, the proposed network contains a deep LSTM network, a classification task subnetwork and a regression task subnetwork. The number of neurons in deep LSTM network is set to [100,100,110,110,100,100] for three LSTM layers and three FC layers alternatively, and the dropout ratio is set as [0.2,0.3,0.5]. As for multi-task network, there are the $3M+1$ neurons in the FC1 layer corresponding to the number of action classes M, and the number of neurons of FC2 and FC3 is set to $10 \times (M+1)$ and 1 respectively . The wieght λ in loss function 7 is set to 0.1.

4.2. Evaluation and Results

4.2.1. *Online action detection*

We firstly test the performance of action detection in unrefined **OAD** with same protocols in [19].

F1-Score: This score is similar to the protocol used in object detection from images [20] to evaluate both presicion and recall values.

SL-Score, **EL-Score**: These scores are designed to evaluate the accuracy of the localization of the start point and the end point for an action respectively.

We implemented several baselines for comparison.

SVM-SW: A SVM detector with sliding window design.

RNN-SW: A state-of-the-art action recognition method using deep LSTM model [18] with a sliding window.

JCR-RNN: Joint Classification Regression RNN [19].

We summarize the results in terms of the average score in Table 1.

Table 1. Comparison of results between baseline and proposed model.

Scores	SVM-SW	RNN-SW	JCR-RNN	Proposed
F1	0.540	0.600	**0.653**	0.628
SL	0.316	0.366	0.418	**0.566**
EL	0.325	0.376	0.443	**0.560**

Our model improves the accuracy of localization of start and end points. This is because the label of segment before action gives more information of action localization. Thus, the noise near the action interval can be recognized more easily. Our linear regression curve is weaker than Gaussian-like curve (JCR-RNN), leading to accuracy loss of F1-score. However, the linear curve can bring out a more accurate forecast of action than Gaussian-like curve.

4.2.2. *Online action forecast*

Our method shows a promising performance on online forecast. Fig.2 shows an example of forecast results.

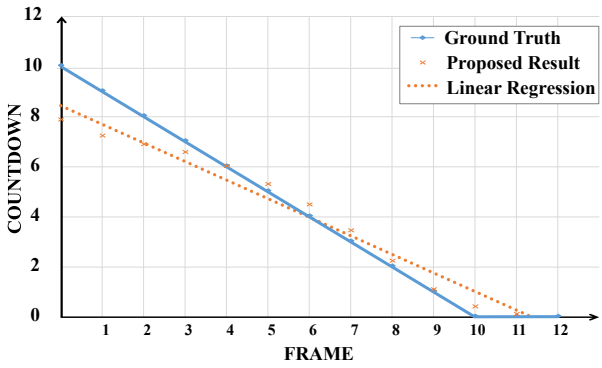

Fig. 2. An example countdown forecast.

Since there is no previous method and evaluation formulation proposed for the action forecast problem, we define a formulation to evaluate the performance of our method.

Forecast-T Score: Considering that the performance near the important action point may be more important, we illustrate a T-base evaluation formulation:

$$FS_T = \frac{\sum_i^K (\sum_{t=s_i-T}^{s_i+T} + \sum_{t=e_i-T}^{e_i+T}) \|L_t - p_t\|}{\sum_i^K 4 \times T}. \quad (12)$$

where K is the sum of actions occurred in a video and L_t , p_t is ground truth and proposed countdown respectively.

Besides, we define two standard forecast curves to evaluate our method: *K-delayed* and *K-advanced*. *K-delayed* is a standard forecast curve delayed by K frames, which means it will forecast K frames later than ground truth. Similarly, *K-advanced* will forecast the action K frames earlier. Fig.3 compares the standard curves with proposed model.

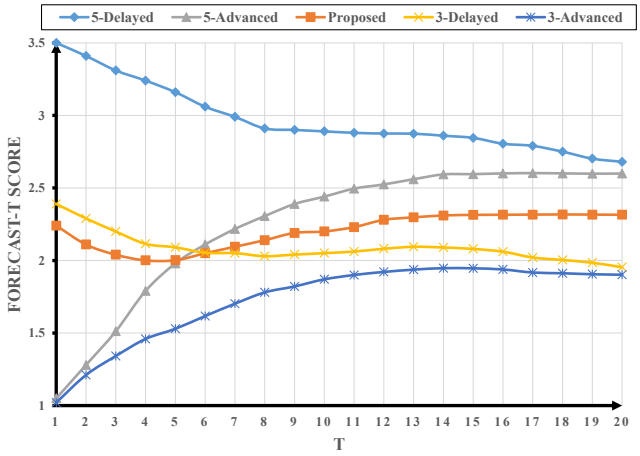

Fig. 3. Comparison with proposed model and four standard curves: 5-delayed, 5-advanced, 3-delayed and 3-advanced.

From Fig.3 we can notice there is an 3-5 frames deviation approximately of our method. Note that when $T \leqslant 5$, the *K-advanced* curve have a more outstanding result than *K-delayed* which is reasonable that we usually prefer a advanced forecast rather than a delayed one. Our method performs better than *3-delayed*. When T goes up, K is dominant element that determine how the curve performs.

5. CONCLUSION

In this work, we propose a new Multi-Task Recurrent Neural Network for human action detection and forecast. Firstly, we split action class into three to detect and forecast human action. Then, we use a deep LSTM network to model temporal dynamics and perform the classification task. Then we design a regression task subnetwork to achieve accurate action forecast. Experimental results show that the proposed method greatly improves the accuracy of detection and obtains an effective result of action forecast.

6. REFERENCES

[1] Jianbo Shi and Carlo Tomasi, "Good features to track," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*. IEEE, 1994, pp. 593–600.

[2] Florent Perronnin, Jorge Sánchez, and Thomas Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. European Conference on Computer Vision*. Springer, 2010, pp. 143–156.

[3] Heng Wang and Cordelia Schmid, "Action recognition with improved trajectories," in *Proc. Int'l Conf. Computer Vision*, 2013, pp. 3551–3558.

[4] Parthipan Siva and Tao Xiang, "Weakly supervised action detection.," in *BMVC*, 2011, vol. 2, p. 6.

[5] Minh Hoai and Fernando De la Torre, "Max-margin early event detectors," *Int'l Journal of Computer Vision*, vol. 107, no. 2, pp. 191–202, 2014.

[6] Amr Sharaf, Marwan Torki, Mohamed E Hussein, and Motaz El-Saban, "Real-time multi-scale action detection from 3d skeleton data," in *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2015, pp. 998–1005.

[7] Limin Wang, Yu Qiao, and Xiaoou Tang, "Action recognition and detection by combining motion and appearance features," *THUMOS14 Action Recognition Challenge*, vol. 1, pp. 2, 2014.

[8] Limin Wang, Zhe Wang, Yuanjun Xiong, and Yu Qiao, "Cuhk&siat submission for thumos15 action recognition challenge," *THUMOS15 Action Recognition Challenge*, vol. 3, no. 4, 2015.

[9] Mihir Jain, Jan Van Gemert, Hervé Jégou, Patrick Bouthemy, and Cees GM Snoek, "Action localization with tubelets from motion," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2014, pp. 740–747.

[10] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid, "P-cnn: Pose-based cnn features for action recognition," in *Proc. Int'l Conf. Computer Vision*, 2015, pp. 3218–3226.

[11] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici, "Beyond short snippets: Deep networks for video classification," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2015, pp. 4694–4702.

[12] Karen Simonyan and Andrew Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[13] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2016.

[14] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2015, pp. 2625–2634.

[15] Shugao Ma, Leonid Sigal, and Stan Sclaroff, "Learning activity progression in lstms for activity detection and early detection," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2016, pp. 1942–1950.

[16] Gunnar Johansson, "Visual perception of biological motion and a model for its analysis," *Perception & psychophysics*, vol. 14, no. 2, pp. 201–211, 1973.

[17] Yong Du, Wei Wang, and Liang Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2015, pp. 1110–1118.

[18] Wentao Zhu, Cuiling Lan, Junliang Xing, Wenjun Zeng, Yanghao Li, Li Shen, and Xiaohui Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," *Proc. AAAI Conf. on Articial Intelligence*, 2016.

[19] Yanghao Li, Cuiling Lan, Junliang Xing, Wenjun Zeng, Chunfeng Yuan, and Jiaying Liu, "Online human action detection using joint classification-regression recurrent neural networks," in *Proc. European Conference on Computer Vision*, 2016.

[20] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman, "The pascal visual object classes (voc) challenge," *Int'l Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.