15th IAPR International Conference on Machine Vision Applications (MVA)
Nagoya University, Nagoya, Japan, May 8-12, 2017.

Continuous Action Recognition with Weakly Labelling Videos

Jun Lei, Guohui Li, Jun Zhang, Shuohao Li, Fenglei Wang
National University of Defense Technology
ChangSha, China
{leijun1987, guohli, zhangjun1975, lishuohao, wangfenglei}@nudt.edu.cn

Abstract

Continuous action recognition plays an important role for human behavior analysis. Most existing approaches require fully labelled action videos, which is labour and time consuming to get. In this paper, we propose a continuous action recognition approach for weakly labelled videos data, where only the orders of action labels are needed without its temporal locations. We bulid a deep network combing convolutional neural network (CNN) and latent-dynamic dynamic conditional random field (LDCRF) to learn action features and recognize actions in a unified procedure. A visual similarity extended connectionist temporal classification (CTC) layer is put on the top of the network to evaluate all possible of temporal locations of weakly labelled videos data. The whole network can be trained end-to-end under weakly supervision. Experimental results on dataset HumanEva show our approach is promising and pratical.

1 Introduction

Human action recognition has been researched extensively for decades. Traditional isolated action recognition [1, 2] is to classify the whole video to a single action, where the videos are usually manually pre-segmented. In contrast, continuous action recognition is a more realistic problem, where the videos contain a sequence of actions and the boundary locations between actions are unknown. Most existing works [3, 4] use supervised learning methods. For continuous actions, each frame of the video has to be annotated with an action label for training. However, with increasing amount of videos data, fully labelling of actions in videos at large scale is highly labour and time consuming. This greatly limits the application of these methods. Instead, we can label only the order of occurring actions without giving the temporal locations of actions. This weakly supervised labelling way is at low cost and more acceptable.

In this paper, we address the problem of continuous action recognition with weakly labelled videos. We aim at recognizing all of the actions and finding out where they occurs, even without temporal supervision in training. This problem is challenging. As no temporal locations are given during training, the number of possible alignments between action labels and video frames is numerous, and searching through all of these alignments is infeasible. Connectionist temporal classification (CTC) [5, 6] was specially designed for this problem where the alignment between the inputs and the target labels is unknown. It efficiently evaluates all of the possible alignments using dynamic programming. CTC was firstly applied for speech and handwriting recognition. When it comes to continuous action recognition in video, there is a much larger space of possible alignments as the duration of an action is much longer than that of a phoneme or a letter. Overmuch alignments cause the performance of CTC to deteriorate seriously. To solve this problem, Huang et al. [7] extended CTC, which introduces visual similarity, to decrease the possibilities of alignments. This is achieved by encouraging the alignment to be consistent with frame-to-frame visual similarities. We adopt CTC to solve the weakly supervised learning problem for continuous action recognition, and follow the same idea of the visual similarity mechanism.

Other challenges also exist in continuous action recognition. What kind of visual feature to represent actions is hard to determine as human actions have high variability of articulated motion, viewpoint and possible occlusions in natural environment. Besides, the modelling of continuous action is difficult. We have to recognize and segment these actions simultaneously. To these issues, we proposed a CNN and LDCRF model in our earlier work [8]. The CNN, one type of deep models, is used to automatically learn effective and robust action features from raw video data. The LDCRF, a probabilistic graphic model, can capture both internal sub-structures and extrinsic dynamics between actions. We integrate CNN and LDCRF seamless to a deep network, which incorporates the feature learning and action recognition in a unified framework. However, this hybrid CNN-LDCRF model is trained under supervision. In this paper, we extend the hybrid CNN-LDCRF model to be able to be trained under weakly supervision.

We propose a weakly supervised learning framework for continuous action recognition. Fig. 1 gives an overview of the proposed framework. The action video is segmented to small video clips. The CNN extracts features of these video clips and the LDCRF models the continuous action. At training phase, a CTC layer is connected to the output layer of the CNN-LDCRF. Visual similarity is embedded in CTC to prevent the alignments that lead to paths visual inconsistent. The errors computed from the CTC layer are back-propagated to LDCRF and CNN. Only giving incomplete action labels, the entire model can be efficiently trained in an end-to-end way.

2 Proposed Approach

In this section, we describe the key components of the proposed approach, including the hybrid CNN-LDCRF model and the visual similarity extended CTC, and their combination.

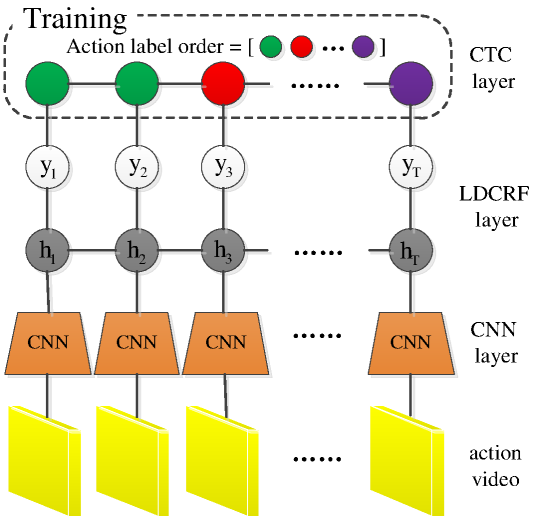

Figure 1. The framework of our model. It is the combination of CNN, LDCRF, and CTC network.

2.1 The Hybrid CNN-LDCRF Model

The action video is denoted by $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T\}$, where $\mathbf{x}_t$ are the raw pixels of video clip t. The video clips contain more information than isolated frames for CNN to exploit.

We design a 3D CNN to capture motion information from the video clip. Two groups of convolutional layers, pooling layers are stacked. The 3D kernel of the convolutional layer connects local neighborhood in spatial and continuous frames in temporal. We apply this network on three channels, including gray, gradient and optical flow. The outputs of the three channels are combined together to a vector, and fed to two fully connection layers. Local response normalized is applied after the first fully connection layer. The output of the second fully connection layer is the action feature. This process from input to output contains a series of linear and nonlinear transformations. We denote the total transformation by $\Psi(\mathbf{x}_i, \Theta)$.

After passing through CNN, we obtain the action features denoted by $\Psi(\mathbf{X}, \Theta) = \{\Psi(\mathbf{x}_1, \Theta), \Psi(\mathbf{x}_2, \Theta), ..., \Psi(\mathbf{x}_T, \Theta)\}$. These features are inputted to LDCRF. The LDCRF model predicts a sequence of action labels $\mathbf{Y} = \{y_1, y_2, ..., y_T\}$ by:

$$P(\mathbf{Y}|\mathbf{X}, \Phi, \Theta) = \frac{1}{Z(\psi(\mathbf{X},\Theta),\Phi)} \exp(\sum_t V_\Phi(t, h_t, \psi(\mathbf{X}, \Theta)) + \sum_t E_\Phi(t, h_{t-1}, h_t, \psi(\mathbf{X}, \Theta))) \quad (1)$$

where Θ and Φ are the parameters of LDCRF and CNN respectively. $\mathbf{h} = \{h_1, h_2, ..., h_T\}$ represents the substructure of the actions. $Z(\Psi(\mathbf{X}, \Theta), \Phi)$ is the partition function. V_Φ and E_Φ are the sum of feature functions on vertex t and edge $(t-1, t)$, which are defined as:

$$V_\Phi(t, h_t, \Psi(\mathbf{X}, \Theta)) = \sum_k \lambda_k s_k(t, h_t, \Psi(\mathbf{X}, \Theta)) \quad (2)$$

$$E_\Phi(t, h_{t-1}, h_j, \Psi(\mathbf{X}, \Theta)) = \sum_k \mu_k t_k(t, h_{t-1}, h_t, \Psi(\mathbf{X}, \Theta)) \quad (3)$$

where s_k are state functions, and t_k are transition functions.

State function s_k depends only on the action feature of current node. Our state function is defined as:

$$s_{hl}(t, h_t, \Psi(\mathbf{X}, \Theta)) = \delta(h, h_t)\Psi(\mathbf{x}_t, \Theta)(l) \quad (4)$$

where $\delta(h, h_t)$ is the indicator function. $\Psi(\mathbf{x}_t, \Theta)(l)$ is the l-th entry of action feature vector $\Psi(\mathbf{x}_t, \Theta)$.

Transition function t_k considers the relation between hidden variable h_{t-1} and h_t, which is defined as:

$$t_{hh'}(j, h_{j-1}, h_j, \psi(\mathbf{X}, \Theta)) = \delta(h, h_{j-1})\delta(h', h_j) \quad (5)$$

2.2 Visual Similarity extended CTC

The action labels of a sequence of video clips are $\mathbf{Y} = \{y_1, y_2, ..., y_T\} \in A^T$, where A be the set of all possible action labels. Define a many-to-one map $\mathcal{B}: A^T \mapsto A^{\leq T}$, where $\mathcal{B}$ removes all repeated labels. Suppose $\mathbf{l}$ be the labelling containing the ordering of actions without temporal localization of labels, the conditional probability of is the sum of the probabilities of all action labels paths corresponding to $\mathbf{l}$:

$$p(\mathbf{l}|\mathbf{X}) = \sum_{\{\mathbf{Y}|B(\mathbf{Y})=\mathbf{l}\}} p(\mathbf{Y}|\mathbf{X}) \quad (6)$$

where $p(\mathbf{Y}|\mathbf{X})$ is the conditional probability of action labels sequence $\mathbf{Y}$. By considering visual similarity, we have:

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{t=1}^{T} z_t^{y_t} \varphi_t^{t+1}, \varphi_t^{t+1} = \begin{cases} \max(\theta, s_t^{t+1}) & y_t = y_{t+1} \\ \theta & y_t \neq y_{t+1} \end{cases} \quad (7)$$

where the unary term $z_t^{y_t}$ is the probability of emitting action y_t. The binary term φ_t^{t+1} is the correlation between consecutive video clips. s_t^{t+1} is the visual similarity between video clips. θ is the minimum similarity threshold. This expression explicitly rewards the paths where visually similar video clips have the same action labels. This plays a key role in applying CTC in action recognition task.

The calculation of (6) can be solved with a dynamic programming algorithm similar to the forward-backward algorithm for HMMs [9]. The forward variable $\alpha(t, s)$ is defined as the summed probability of all t length paths that are mapped by $\mathcal{B}$ onto length s prefix of $\mathbf{l}$. It can be calculated recursively by:

$$\alpha(t, s) = \begin{cases} \hat{z}_t^{y_t} \alpha(t-1, s) & s = 1 \\ \hat{z}_t^{y_t} \alpha(t-1, s-1) + \tilde{z}_t^{y_t} \alpha(t-1, s) & s > 1 \end{cases} \quad (8)$$

where,

$$\hat{z}_t^{y_t} = \frac{\max(\theta, s_{t-1}^t) z_t^{y_t}}{\max(\theta, s_{t-1}^t) z_t^{y_t} + \theta(1 - z_t^{y_t})} \quad (9)$$

$$\hat{z}_t^{y_t} = \frac{\max(\theta, s_{t-1}^t) z_t^{y_t}}{\max(\theta, s_{t-1}^t) z_t^{y_t} + \theta(1 - z_t^{y_t})} \quad (10)$$

The backward variable $\beta(t, s)$ is defined as the summed probabilities of all paths starting at t that complete $\mathbf{l}$. It can be calculated similarly as forward variable. Then we have:

$$p(\mathbf{l}|\mathbf{X}) = \sum_{s=1}^{|\mathbf{l}|} \frac{\alpha(t, s)\beta(t, s)}{z_t^{l_s}} \quad (11)$$

2.3 Combination of CNN-LDCRF and CTC

We put a CTC layer on the bottom of CNN-LDCRF. LDCRF's marginal probabilities of emitting an action label are sent to CTC, that is:

$$z_t^k \triangleq P(y_t = k|\mathbf{X}, \Phi, \Theta) = \sum_{\mathbf{Y}:y_t=k} P(\mathbf{Y}|\mathbf{X}, \Phi, \Theta) \quad (12)$$

For training, our objective is to maximise the log probabilities of all the correct labelling in the training set S. This can be achieved by minimising the following objective function:

$$L(S) = -\sum_{(\mathbf{X},\mathbf{l})\in S} L(\mathbf{X}, \mathbf{l}) = -\sum_{(\mathbf{X},\mathbf{l})\in S} \ln p(\mathbf{l}|\mathbf{X}) \quad (13)$$

For convenience, we take one sample loss $L(\mathbf{X}, \mathbf{l})$ for illustration. The whole network can be trained using gradient descent. The error of CTC layer is the gradient with respect to z_t^k:

$$\delta_t^k = -\frac{\partial \ln p(\mathbf{l}|\mathbf{X})}{\partial z_t^k} = -\frac{\sum_{s:l_s=k} \alpha(t,s)\beta(t,s)}{(z_t^k)^2 p(\mathbf{l}|\mathbf{X})} \quad (14)$$

These errors propagate to the LDCRF layer. We compute the gradient respect to parameter Φ. For parameter λ_k associated with state function s_k, the gradient is computed as:

$$\frac{\partial L(\mathbf{X}, \mathbf{l})}{\partial \lambda_k} = \sum\nolimits_{t,t',a} \delta_t^k P(h_{t'} = a|y_t = k, \Psi(\mathbf{X}, \Theta), \Phi) s_k(t', a, \Psi(\mathbf{X}, \Theta)) - \sum\nolimits_{t,t',\mathbf{Y}',a} \delta_t^k P(h_{t'} = a, \mathbf{Y}'|\Psi(\mathbf{X}, \Theta), \Phi) s_k(t', a, \Psi(\mathbf{X}, \Theta)) + \frac{\lambda_k}{\sigma^2} \quad (15)$$

For parameter λ_k associated with transition function t_k, the gradient is computed as:

$$\frac{\partial L(\mathbf{X}, \mathbf{l})}{\partial \mu_k} = \sum\nolimits_{t,t',a,b} \delta_t^k P(h_{t'-1} = a, h_{t'} = b|y_t = k, \Psi(\mathbf{X}, \Theta), \Phi) t_k(t', a, b, \Psi(\mathbf{X}, \Theta)) - \sum\nolimits_{t,t',\mathbf{Y}',a,b} \delta_t^k P(h_{t'-1} = a, h_{t'} = b, \mathbf{Y}'|\Psi(\mathbf{X}, \Theta), \Phi) t_k(t', a, b, \Psi(\mathbf{X}, \Theta)) + \frac{\mu_k}{\sigma^2} \quad (16)$$

The marginal probabilities in (15) and (16) can be computed using belief propagation algorithm [10]. The last terms of (15) and (16) are regularization terms.

The gradient respect to CNN's parameter Θ can be easily computed by back propagation algorithm of neural network [11]. Denote the node of CNN's output as o_l, namely the l-th entry of action feature $\Psi(\mathbf{X}, \Theta)$. The error δ_l of node o_l is:

$$\delta_l = \frac{\partial G(\Omega)}{\partial o_l} = \sum\nolimits_{t,t',a} \delta_t^k P(h_{t'} = a|y_t = k, \Psi(\mathbf{X}, \Theta), \Phi) \lambda_{al} - \sum\nolimits_{t,t',\mathbf{Y}',a} \delta_t^k P(h_{t'} = a, \mathbf{Y}'|\Psi(\mathbf{X}, \Theta), \Phi) \lambda_{al} \quad (17)$$

The errors propagate backward to update the parameters of CNN.

After the parameters of the network are learnt, given a new sequence of video clips, the action label having maximum marginal probability output by LDCRF is assigned to each clip. All frames in a video clip have the same action label.

3 Experiments

In this section, we test our approach on a real human motion dataset: HumanEva. It consists of multiple persons performing a set of actions. Each video contains one action with several repetitions. The videos are captured by cameras of different viewpoints, therefore great variations of viewpoints exist. We choose four actions: walking, box, jog and gesture, performed by person S1, S2 and S3, captured by cameras C1, C2 and C3. We concatenate these four actions of each person under the same camera in an arbitrary order, totally generating 18 videos. We use 15 videos for training, and 3 videos for testing. The length of each video is about 1250 frames. The metric of recognition accuracy is used to evaluate our approach, which is defined as the ratio between the number of correct classified frames over the total number of frames.

3.1 Implementation Details

We use a detector to locate persons and track them. The person images are cropped and resized to size 90×50. The length of video clip is 5. For CNN, we use 16 and 8 3D kernels for the first and second covolutional layers, respectively. The node numbers of the two fully connection layer are 100 and 50, respectively. The hidden states of LDCRF for each action is 3. The batch size of gradient descent is set to be 1, that is the gradient descent is performed on one action video at each iteration. The gradient descent stops until the model converges.

To measure visual similarity, k-means clustering is applied to cluster video clips that have similar visual features and are temporally adjacent. If the video clip t and $t+1$ are in the same cluster, we set s_t^{t+1} to ∞, else, s_t^{t+1} is the cosine similarity of them. We use HOG3D descriptor [12] to extract visual feature for each video clip, and build a 100 words dictionary for k-means clustering.

Our model is named CNN-LDCRF-ECTC for short. We also test two other models on this dataset. 1) CNN-ECTC model: the CNN is directly connected to the extended CTC without LDCRF model. 2) CNN-LDCRF-CTC model: original CTC is used without adding visual similarity. Besides, we train the hybrid CNN-LDCRF model under supervision and show its result for comparision.

3.2 Results and Analysis

The segmentation and recognition results are presented by color bars in Fig. 2. Colors indicate action classes, and the horizontal axis is time in frames. Noted that the result of CNN-LDCRF-CTC model is not shown. This is because that this model does not converge at training, and fails to recognize actions. This proves our previous analysis that original CTC's performance deteriorates seriously when the length of sequence is very long. The predicted action labels of CNN-CTC model are fractional. This is because of the lack of dynamic constraint without the LDCRF. Our CNN-LDCRF-ECTC model achives relatively better result. Even without temporal locations of labels for training, many segmentation locations between actions are accurately predicted by our model. While the

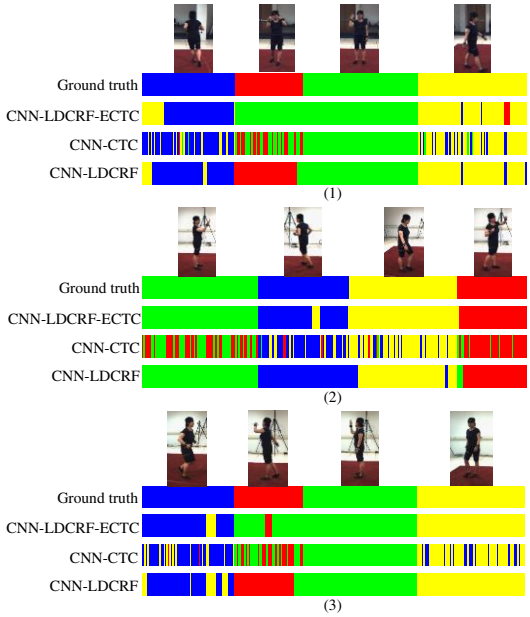

Figure 2. Comparison of different models on three test videos. Videos (1), (2) and (3) have 1300, 1300 and 1305 frames, respectively. Colors indicates different actions. Blue: jog, red: box; green: gesture, yellow: walking.

Table 1. Comparision of recognition accuracies.

Model	Recognition accuracy
CNN-LDCRF-ECTC	84.12%
CNN-CTC	74.39%
CNN-LDCRF-CNN	93.21%

recognition results of action box in video (1) and (3) are bad. Our model confuses action jog and walking sometimes. The same problem also happens to hybrid CNN-LDCRF model.

The recognition accuracy results are shown in Table. 1. Our model achieves the recognition accuray of 84.12%, which is 9.09% lower than the fully supervised hybrid CNN-LDCRF model.

4 Conclusion

In this paper, we propose an approach for continuous action recognition when action videos are weakly labelled. We introduce our hybrid CNN-LDCRF model for continuous action recognition. The hybrid CNN-LDCRF model is combined with visual similarity extented CTC, making it to be able trained under weakly supervision, where only the orders of actions are given even without the accurate temporal locations. Experiment results demonstrate the practicability and accuracy of our model. As future work, we should consider the situation where a part of videos is weakly labelled and the other part is fully labelled. We should simultaneously trained our model in a unified framework for weakly and fully labeled videos.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No: 6117019 and 71673293).

References

[1] I. Laptev, M. Marszalek, C. Schmid and C. Schmid. "Learning realistic human actions from movies". *in IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, June 2008, pp. 1-8.

[2] S. Ji, M. Yang, and K. Yu. "3d convolutional neural networks for human action recognition." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221-231, 2013.

[3] K. Kulkarni, G. Evangelidis, J.Cech, and R. Horaud. "Continuous action recognition based on sequence alignment." *International Journal of Computer Vision*, vol.1, vol 112, pp. 90-114, 2014.

[4] Z. Wang, J. Wang, J. Xiao, K. H. Lin, and T. Huang, "Substructure and boundary modeling for continuous action recognition." *in IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Providence, USA, June 2014, pp. 1330-1337.

[5] A. Graves, S. Fernández, F. Gomez and J. Schmidhuber. "Connectionist Temporal Classication: Labelling Unsegmented Sequence Data with Recurrent Neural Networks." *International Conference on Machine Learning*, Pittsburgh, USA, June 2016, pp. 369-376.

[6] A. Graves. "Supervised Sequence Labelling with Recurrent Neural Networks." *Springer Berlin Heidelberg*, 2012.

[7] D. A. Huang, F. F. Li and J. C. Niebles. "Connectionist Temporal Modeling for Weakly Supervised Action Labeling." *European Conference on Computer Vision*, Amsterdam, Netherlands, October 2016, pp. 137-153.

[8] J. Lei, G. H. Li, S. H. Li, D. Tu and Q. Guo. "Continuous Action Recognition Based on Hybrid CNN-LDCRF Model." *International Conference on Image, Vision and Computing*, Portsmouth, UK, August 2016, pp. 68-74.

[9] L. R. Rabiner. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition." *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, 1998.

[10] J. Lafferty, A. McCallum, and Fernando C. N. Pereira. "Conditional random elds: Probabilistic models for segmenting and labeling sequence data." *in Proceedings of the 18th International Conference on Machine Learning*, Williamstown, USA, June 2001, pp. 282-289.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.

[12] A. Kläser, M.Marszalek, and C. Schmid. "A spatio-temporal descriptor based on 3d-gradients." *British Machine Vision Conference*, Leeds, UK, September 2008, pp. 275:1-10.