FAST ONLINE ACTION RECOGNITION WITH EFFICIENT STRUCTURED BOOSTING

Masamichi Shimosaka, Yu Nejigane, Taketoshi Mori and Tomomasa Sato

Department of Mechano-Informatics, the University of Tokyo, Japan

ABSTRACT

In this paper, we propose a novel robust action recognition framework with the following capabilities: 1) *online* encoding motions to multi-label sequence where the output in each frame is a tuple of labels rather than a single label, 2) providing efficient automatic relevant motion selection framework, 3) learning systems so as to be optimal for online multi-label sequence classification. As for multi-label classification, our approach incorporates contextual information about action not only temporal information but hierarchical information of actions. Inference tends to be complex so as to achieve such complex recognition scheme, however, we propose an efficient Viterbi-like decoding algorithm which integrates forward algorithm and loopy message passing algorithm. As for the learning process, the algorithm optimizes the parameters so as to maximize log likelihood of the model. Boosting, ensemble approach of machine learning, is leveraged to provide efficient feature selection framework in the training process. The experimental results show that the proposed method successfully exploits the impact of contextual information then significantly outperforms the traditional approaches in dynamic gait motion classification.

Index Terms— Action recognition, Boosting

1. INTRODUCTION

Recognizing human action in natural environments is of increasing interest in applications in intelligent human computer interaction and human activity surveillance. Action recognition algorithm [1] also becomes a good application of recent machine learning algorithms. This is because issues around human action recognition contain hot topics of recent machine learning algorithms.

One issue is about temporal information. It must be considered in human action classification. To exploit temporal information, many researchers utilize a sort of probabilistic models, such as hidden Markov models (HMMs), conditional random fields (CRFs) [2]. In addition to the temporal information, it sometimes needs relational contexts of action labels. Some activities have relational constraints. Though two actions are exclusive in most situations, they are sometimes compositional or concurrent. They also have hierarchy. *Walking while waving hands* is a good example of concurrency. *Running never occurs when walking, but standing must occur when walking* corresponds to exclusiveness and hierarchy. Note that, the standard HMMs and CRFs cannot encode hierarchical and concurrent context information of actions. Even though there exist factorial graphical models to tackle relational contexts [3, 4], online inference process cannot run in them.

Another issue is feature selection problem. HMMs and CRFs are suitable for classifying complex observation sequences, however, we must design motion features in a cautious manner. The key to success is to incorporate discriminative (or relevant) features. It seems easy to leverage as many features as we can have for models, but the training process with large number features is challenging. Large number features sometimes lead over-fitting problem when the size of training data is relatively small. Hence we wish to make an algorithm that automatically induces relevant (or discriminative) observation features. Boosting based approaches [5] can select the relevant motion features from the complex observations, however, the standard boosting algorithm cannot exploit contextual information of actions.

Thus, our goal in this paper is to propose a novel contextual probabilistic model where feature selection process can be automatically executed. Similar motivation rises state of the art techniques on boosting with graphical models [6–9], however, all the models are not designed for action recognition. One drawback is that they are designed with *offline* use. The graphical models in them are not original ones but kinds of CRFs. Hence, the performance is optimized for offline contextual inference process. Optimal performance is not available in online classification. Furthermore, the previous research works need huge amount of computational cost among huge parameter space of contextual information because they utilize boosting algorithm to optimize both discriminative feature discovery and contextual parameters.

2. ONLINE ACTION CLASSIFICATION WITH PROBABILISTIC CONTEXTUAL MODELS

2.1. Problem statement

Fig. 1 illustrates an example of online action recognition framework treated in this paper. At time t, the framework out-

puts a collection of the binary labels $\boldsymbol{y}_t = [y_t^{(1)}, y_t^{(2)}, \cdots, y_t^{(M)}]^\mathsf{T}$ from sequence of motion features $\boldsymbol{x}_{1:t} = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_t$, where M is a number of actions to be recognized. In the collection of labels, let $y_t^{(i)} \in \{\pm 1\}$ be an indicator of occurrence of i-th action class at time t. Collection of binary indicators provides us flexibility of incorporating relational constraints between action categories, such as exclusionary, multi-categorical assumption of actions. Even if there is multi-inclusion relation between actions, the binary approach can handle the situation.

2.2. Probabilistic formulation

We build a probabilistic graphical model of actions that leverages relational constraints and temporal assumption of actions (see Fig. 2. Nodes of motion feature $\boldsymbol{x}_{1:T}$ are omitted for simplicity). Directed edge represents temporal assumption of actions and undirected edge represents relational (hierarchical, exclusionary and multi-categorical) assumption among actions. This graphical model is a combination of directed and undirected graphical model. For our convenience, let $\mathcal{N}_i$ be a collection of actions that has relation with i-th action category. In Fig. 2, action b, c corresponds to $\mathcal{N}_a$: $\mathcal{N}_a = \{b, c\}$. Also let $\boldsymbol{y}_t^{(\mathcal{N}_i)}$ be binary indicators of $\mathcal{N}_i$ at time t.

By virtue of the fact that the model leverages directed edges, we can factor the joint distribution of label sequence $\boldsymbol{y}_{1:T}$ as the product of potential functions.

$$p(\boldsymbol{y}_{1:T}|\boldsymbol{x}_{1:T}) = \prod_t p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{y}_{t-1}). \tag{1}$$

This factorization makes algorithm usable in online inference process. We also factorize approximately the density function $p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{y}_{t-1})$ by using the idea of the pseudo conditional independence approach as

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{y}_{t-1}) \approx \prod_i p(y_t^{(i)}|\boldsymbol{x}_t, \boldsymbol{y}_t^{(\mathcal{N}_i)}, y_{t-1}^{(i)}). \tag{2}$$

For online action recognition, we wish to obtain MAP estimation of $\boldsymbol{y}_t$. From the structure of the graphical model, we design a Viterbi-like forward inference process as in HMMs. Even though we simplify the decoding process, the algorithm is not so simple as the decoding process of HMMs. This comes from the loopy structure of graphical model at time t. Hence we leverage iterative process to tackle this problem. This is similar to the iterated conditional updating algorithm [10]. Specifically, the algorithm iterates the following updates as

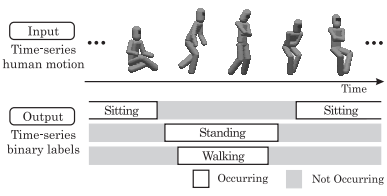

Fig. 1. Input and output in proposed method

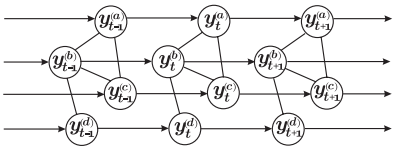

Fig. 2. Graphical model for action recognition in proposed method

$$\hat{y}_t^{(i)} \leftarrow \underset{y_t^{(i)}}{\operatorname{argmax}}\, \delta_t^{(i)}(y_t^{(i)}) \tag{3}$$

$$\delta_t^{(i)}(y_t^{(i)}) \leftarrow \max_{y_{t-1}^{(i)}} \delta_{t-1}^{(i)}(y_{t-1}^{(i)}) p(y_t^{(i)}|\boldsymbol{x}_t, \hat{\boldsymbol{y}}_t^{(\mathcal{N}_i)}, y_{t-1}^{(i)}) \tag{4}$$

The update formula in (4) iterates over action categories i until $\hat{\boldsymbol{y}}_t$ converges. Note that this updating formula runs through only current frame. Thanks to the forward decoding process, our model can be used in online action recognition. Note that loopy belief propagation is often used in relational contextual labeling algorithm [3], however, they cannot be used in online usage because the messages run over the sequence. Next, we define probability distribution $p(y_t^{(i)}|\boldsymbol{x}_t, y_{t-1}^{(i)}, \boldsymbol{y}_t^{(\mathcal{N}_i)})$ as logistic regression function form:

$$p(y_t^{(i)} = 1|\boldsymbol{x}_t, y_{t-1}^{(i)}, \boldsymbol{y}_t^{(\mathcal{N}_i)}) = \frac{\sigma_{i,t}}{1 + \sigma_{i,t}}, \tag{5}$$

where $\ln \sigma_{i,t} = F_i(\boldsymbol{x}_t) + G_i\left(y_{t-1}^{(i)}, \boldsymbol{y}_t^{(\mathcal{N}_i)}\right)$. Specifically, function F_i represents the tendency of $y_t^{(i)} = +1$ calculated from the observation $\boldsymbol{x}_t$. Function G_i indicates tendency of $y_t^{(i)} = +1$ among the tuple of $\boldsymbol{y}_t^{(\mathcal{N}_i)}$ and $y_{t-1}^{(i)}$. The function F_i and G_i is built from the training process. We design the local potential F_i as additive model. The pairwise potential G_i is defined as parametric representation of neighbor action states. The detailed explanation of F_i and G_i and their training process proceeds in the next section.

3. LEARNING THE CONTEXTUAL MODELS

3.1. Overview

As the learning process of the contextual models, we wish to minimize log loss function of action categories calculated from the above filtering algorithm,

$$J = \prod_t \prod_i -\ln p(y_t^{(i)}|\boldsymbol{x}_t, \boldsymbol{y}_t^{(\mathcal{N}_i)}, y_{t-1}^{(i)}). \tag{6}$$

To keep the algorithm so simple, we factor the training process into two phases. One is for local classification optimization. Another is for interdependent potential optimization. The process iterates local potential optimization and pairwise potential optimization until the log loss function converges.

The algorithm takes the idea of EM-like coordinate ascent algorithm. In the training process (a), the local potential F_i is updated via additive gradient tree boosting process with the fixed G_i. As described below, the algorithm automatically selects discriminative motion features and updates F_i. In the training process (b), the parameters of the pairwise potential G_i is optimized by the fixed F_i. This separation accelerates the efficiency of the training process of structured classification process. Compared to the standard structured boosting or standard gradient based CRF training algorithm, our algorithm drastically converges at optimal point.

3.2. Learning the local potentials via boosting

The algorithm updates stage-wise local potential F_i. We update F_i as additive model by optimizing the second order Taylor expansion of (6). This idea is similar to the process of logitBoost [5]. The algorithm starts with the stage at $F_i \equiv 0$, then adds simple weak learner to F_i at every stage. Let $f_i^{(k)}$ be a binary indicator function for i-th target action at k-th training stage. We also assume the function $f^{(k)}$ computes the output from the single value of observation, then this can be defined as $f_i^{(k)}(\boldsymbol{x}_t) = \mathrm{sgn}\left(\phi_i^{(k)}(\boldsymbol{x}_t) - \alpha_i^{(k)}\right)$, where $\phi_i^{(k)}$ represents a feature extractor from motion feature $\boldsymbol{x}_t$ to some scalar value and $\alpha_i^{(k)}$ represents a threshold value. This type of function is called a decision stump. Computation of feature extraction and threshold processing is so simple. Collection of extractors $\phi_i^{(k)}$ is given priori. When $\phi_i^{(k)}$ represents a single feature extractor, then the stage-wise optimization of F_i serves as forward feature selection framework. The training process (a) is done by solving the following weighted least squares problem.

$$f_i^{(k)} \quad = \quad \underset{f_i}{\mathrm{argmin}} \sum_t w_t^{(i)} \left(f_i(\boldsymbol{x}_t) - z_t^{(i)}\right)^2 \qquad (7)$$

where $w_t^{(i)} = p^*(y_t^{(i)} = +1)(1 - p^*(y_t^{(i)} = +1))$, $z_t^{(i)} = \left({y_t^{(i)}}^* - p^*(y_t^{(i)} = +1)\right)/w_t^{(i)}$ and ${y_t^{(i)}}^* = (y_t^{(i)} + 1)/2$. The optimization process to determine the optimal threshold $\alpha_i^{(k)}$ is efficiently optimized via bound limited scalar optimization algorithm. After the optimal $f_i^{(k)}$ is obtained from the above optimization, F_i is updated as $F_i \leftarrow F_i + \nu f_i^{(k)}$. Coefficient ν is usually set as 1, however, we set $\nu = 0.3$ to avoid numerical instability of the building process of F_i.

3.3. Learning the pairwise potentials

We define pairwise potential function G_i as $G_i(y_{t-1}^{(i)}, \boldsymbol{y}_t^{(\mathcal{N}_i)}) = \beta_i y_{t-1}^{(i)} + \sum_{j \in \mathcal{N}_i} \beta_{i,j} g_{i,j}(y_t^{(j)})$, where the first term represents temporal assumption of $y_{t-1:t}^{(i)}$ and the second term corresponds relational constraints between labels, $g_{i,j}(\cdot)$ outputs consistency indicator related to i-th action. The output $g_{i,j} = -1$ represents inconsistent pair of $y_t^{(i)} = +1$ and $y_t^{(j)}$. If the pair of actions is consistent, the function returns $g_{i,j} = 0$. We set the parameters satisfy $\beta_{i,j} = \beta_{j,i}$ for symmetry. As the training process of G_i, the parameters $\beta_i, \beta_{i,j}$ are optimized by gradient based optimization algorithm such as BFGS and conjugate gradient method. To avoid overfitting problem, we leverage Gaussian prior of $\beta_i, \beta_{i,j}$ to add the criterion J as $\tilde{J}$,

$$\tilde{J}(\beta_i, \beta_{i,j}) = -\sum_{t,i} \ln p^*(y_t^{(i)}) + \frac{C}{2}\left(\sum_i \beta_i^2 + \sum_{i<j, j \in \mathcal{N}_i} \beta_{i,j}^2\right).$$

This means we estimate MAP (maximum a posteriori) of $\beta_i, \beta_{i,j}$. Compared to the previous approach of structured boosting method, our algorithm can explicitly leverage regularization factor in the learning process.

4. EXPERIMENTAL RESULTS

This section describes evaluation experiments of our contextual models. We demonstrate effectiveness of the algorithm by comparing cognitive performance with other boosting based algorithms.

4.1. Target actions and data

Target actions to be classified is walking, running and moving forward. These actions are good to clarify the effectiveness of relational and temporal assumption of actions. Walking and running action are exclusive. Forward motion includes both the gait dynamic actions. Even though there are several actions without movements such as lying and sitting, we target only actions with movements in the experiments. This is because actions without movements have innate poses and actions with movements vary according to time and then have not innate poses, and then we expect that actions without movement is easy to be classified compared to classification of actions with movements.

The measured motion data for the experiments are sequential human motion data fetched by wearable magnetic motion sensors at 30 Hz. Performer in this experiment wears 11 sensors. It contains 36 degrees of freedom of position and posture information of joints. The actions included in the data are walking, running and standing still. The data also contains transition from an action to another. This means one motion clip contains several action categories. We annotate motion data per frame per action. The motion data are divided into 4 sets and we evaluate the proposed algorithm by cross validation. Each set contains about 2000 frames (about 1 minute) motion data. The number of the performers is one. Even though the performance will be better when we utilize motions multiple performers, the goal of this experiment is to clarify the impact of contextual information.

4.2. Candidates of discriminative motion features

In the experiments, we first eliminate some redundant motion sensors with wrapper feature selection method. Then we select only 4 sensors attached to both thighs and second thighs and give the following scalar motion features of each joint of sensor information: 1) vertical components of the joint's posture matrix in world coordinate system (3 elements per joint), 2) every component of relative posture matrix between two joints (9 elements per pair), 3) temporal subtraction of each component listed above. The candidates mentioned above are about 400. As you can see, we utilize only posture features, though we can compute location information of each sensor from kinematics computation. This is because we have planned to make practical recognition application with gyro or magnetic sensors.

4.3. Evaluation Criteria

For cognitive performance measure, we use F-measure per action category. F-measure is a harmonic average of recall rate and precision rate. To validate the impact of contextual information of the algorithm, we also compare other algorithms. One is *boosting only* algorithm. This algorithm does not consider any contextual assumptions of actions ($G_i \equiv 0$), hence, the algorithm is equivalent to original logitBoost algorithm. The second one is called *boosting + temporal optimization*. This does not consider relational constraint among the actions ($\beta_{i,j} = 0$) but consider temporal assumption ($\beta_i \neq 0$). We also compare the performance of another algorithm called *boosting + smoothing*. This algorithm smoothes the output of boosting only method of focused window. This approach has similar effect that we set β_i some constant value. Because the performance of this algorithm depends on window size, we adjust several size of window and evaluate the performance.

4.4. Experimental results

As the experimental result, we show the performance among the algorithms in Table 1. M-F represents *moving forward*. The performance is calculated when the number of functions $f^k(\cdot)$ is 100 in every algorithms. The proposed algorithm significantly outperforms the other algorithms in both F-measure and the number of consistent recognition result frames. We increment the number of conflicts if the result is inconsistent, for example, *walking and non-moving-forward and non-running*. Note that the number of the conflict frames drastically reduces compared to the other algorithm. In contrast, boosting + smoothing algorithm fails to strengthen the performance of the boosting-only algorithm even if we change the window size. We also show the F-measure with respect to the number of weak classifiers in F_i in each method. The F-measure is calculated from the test data. Fig. 3 shows the performance of walking recognition.

Table 1. Experimental results for target actions

	Walking	**Running**	**M-F**	**conflict**
Proposed	**97.6**	**96.4**	97.4	**62**
Boosting + temporal optim.	96.0	96.2	**97.5**	95
Boosting only	93.2	94.7	96.8	172
Boosting + smoothing(5)	95.1	94.9	96.6	99
Boosting + smoothing(10)	93.9	93.1	94.4	93

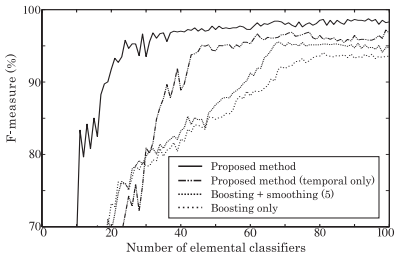

Fig. 3. Number of weak classifiers vs. f-measure for test data in walking classification

5. CONCLUSION

We proposed a novel online robust action classification algorithm with sequence labeling algorithm of contextual models. The model can handle not only temporal but also exclusionary, hierarchical and multi-categorical assumption of action labels. In order to make algorithm online usable, we propose a Viterbi-like decoding algorithm. The learning algorithm automatically selects relevant motion features via boosting process. The algorithm is inspired by logitBoost, however, our model can obtain higher recognition performance by using contextual information. The experimental results show that the proposed method significantly outperforms traditional approaches in dynamic gait motion classification.

6. REFERENCES

[1] D. Cao et al., "Online motion classification using support vector machine," in *Proc. of ICRA 2004*.

[2] J. Lafferty et al., "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proc. of ICML 2001*.

[3] C. Sutton et al., "Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data," in *Proc. of ICML 2004*.

[4] M. Shimosaka et al., "Robust action recognition and segmentation with multi-task conditional random fields," in *Proc. of ICRA 2007*.

[5] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," Tech. Rep., Stanford University, 1998.

[6] A. Torralba et al., "Contextual models for object detection using boosted random fields," in *Advances in NIPS 17*.

[7] T. Truyen et al., "AdaBoost.MRF: Boosted Markov random forests and application to multi-level activity recognition," in *Proc. of CVPR 2006*.

[8] L. Liao et al., "Training conditional random fields using virtual evidence boosting," in *Proc. of IJCAI 2007*.

[9] Y. Altun et al., "Discriminative learning for label sequences via boosting," in *Advances in NIPS 15*.

[10] J. Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society B*, vol. 48, no. 259–302, 1986.