Continuous pattern detection and recognition in stream - a benchmark for online gesture recognition

Nehla Ghouaiel, Pierre-François Marteau, Marc Dupont

Continuous pattern detection and recognition in stream -A benchmark for on-line gesture recognition-

Nehla Ghouaiel*, Pierre-François Marteau*, Marc Dupont*†
*IRISA, Université de Bretagne Sud, Campus de Tohannic, Vannes, France
†Thales Optronique, 2 Avenue Gay Lussac, Elancourt, France

Abstract—Very few benchmark exists for assessing pattern detection and recognition in streams in general and for gesture processing in particular. We propose a dedicated benchmark based on the construction of isolated gestures and gesture sequences datasets. This benchmark is associated to a general assessment methodology for streaming processing which first consists in labelling the stream according to some heuristics (that can be optimized on training data) and then aligning the ground truth labelling with the predicted one. 6 pattern recognition models (including DTW, KDTW, HMM, HCRF and SVM) have been accordingly evaluated using this benchmark. It turns out that the regularized kernelized version of DTW measure (KDTW) associated to a SVM is quite efficient, comparatively to the other models, for detecting and recognizing continuous gestures in streams.

1. Introduction

This paper aims at presenting a benchmark dedicated to compare methods for gesture spotting and recognition in streams using low-cost sensors, such as the Microsoft Kinect or data gloves. Online human gesture recognition has a wide range of applications including video surveillance, human computer/robot interaction, sports video analysis, video retrieval and many other sensor-acquisition related application. Accurate spotting and recognition of human gestures in streams is still a quite challenging task, despite the research efforts in the past decade and many encouraging advances. We present in this paper a benchmark procedure that includes two datasets, an assessment methodology and a case study that involves beanchmarking pattern detection and recognition models applicable to stream processing.

The remainder of this paper is organized as follows; Sec. 2 is dedicated to the state of the art of previous work in on-line gesture recognition. Sec. 3 describes the data set considered in the experiments. The technical approach is presented in Sec. 4. We detail carried out experiments and discuss obtained results in Sec. 5 before providing some conclusions and directions for future research.

2. Previous work in on-line gesture recognition

Stream processing requires (explicitly or implicitly) temporal segmentation of sequences of temporal data. For human motion it refers to the task of temporally cutting sequences into segments with different semantic meanings (gesture/action) which is an important step in gesture/action analysis and recognition. The existing approaches proposed to deal with this issue can be classified into two categories. The first category is about gesture recognition from color videos and hence is based on visual features. The second one rely instead on the features of motion data that describe the human-body-part movements.

On the one hand, for the video-based methods, it is a common practice to locate spatio-temporal interest points like STIP (1), and use the distributions of the local features like HOF (2) or HOG (3) to represent local spatio-temporal pattern. Li et al. (4) proposed the Bag of Visual Words (BoVW) model, which is used by many researchers for action recognition from color videos. BoVW model uses histograms as the features for gestures recognition. In order to count histograms, frames must be segmented first which makes BoVW model not suitable for online gesture recognition. Moreover, Song et al. (5) proposed a system for gesture recognition based on the combination of body and hand pose information. In their system, the body pose is estimated by using a multi-hypothesis Bayesian inference framework with a particle filter (6). A multi-class SVM classifier (7) is trained off-line based on HOG descriptors (3) extracted from manually-segmented images of hands, and is used to classify hand poses. Nevertheless, this system is not allowing non-segmented continuous time-series input.

On the other hand, several researchers claim that the movement of the human skeleton can be used for distinguishing different human gestures. In this context, the proposed approaches for gesture recognition are based on the features of motion data describing a specific human-body-part motion. Müller et al.(8) present an approach to label motion capture (MOCAP) data according to a given set of motion categories or classes, each specified by a set of motions. The presented method employs Motion Template (MT) to segment and label the motion data. A template is a generic gesture instance used to match with the data stream for a class of gestures. In their method, it is represented by a matrix averaging the training motions expressed by relational features. The gesture-level motion template approaches have a major weakness on dealing with intra-class variations which depend on the person performing the gesture. Consequently, this method is inefficient for dealing with real-time data

streams. Additionally, Wang et al.(9) proposed to learn one subset of human body joints for each action class. The subset joints are representative of one action compared to others. However, their approach is only applicable to the recognition of pre-segmented instances and cannot be used in online recognition of unsegmented data streams. Furthermore, Gong et al.(10) proposed an alignment algorithm for action segmentation. The proposed method called Kernelized Temporal Cut (KTC), is a temporal extension of Hilbert space embedding of distributions (11) with kernelized two-sample test (12) and was applied to sequentially estimate temporal cut points (change points) in human motion sequences. Since their approach is based on structure similarity between frames, it is only suitable for segmenting cyclic actions. Among others, Zhao et al. (13)(14) proposed a feature which they call Structured Streaming Skeletons (SSS) in order to represent inherent human motion characteristics. Each SSS feature vector consists of a number of attributes represented as distance values. Each value is a minimum DTW distance between all the scanned sub-sequences (ending at the current frame) and a template in the dictionary for the given pair of joints. Here a template is defined as a one-dimensional time series representing distance values of two joints of human body during the time of a gesture instance. The Jointly Sparse Coding (15) classifier was used to learn a gesture model from the extracted SSS feature vectors. Prediction is performed by a linear regression method that assigns each feature vector with a gesture label based on the learned gesture model.

In this paper, we tackle gesture recognition through the tracking of skeleton joint positions. Following (13)(14), to avoid segmenting streams beforehand, we develop an approach that simply requires the definition of the gesture vocabulary in the form of isolated patterns. These patterns are then used to detect and recognize gestures in streams. Stream segmentation is a by-product of the detection and recognition process.

3. Benchmark dataset

The datasets we have designed are complementary to the one developed by Microsoft Research (MSR Action Recognition Datasets) (16): they allow to explore the hand-shape and the upper body movement using 3D positions of skeletal joints.

To develop our benchmark data dedicated to upper body movement recognition, we have selected an excerpt of the Naval Air Training and Operating Procedures Standardization (NATOPS) dataset presented in (5) which includes six of the twenty-four body-hand gestures used when handling aircraft on the deck of an aircraft carrier. The dataset includes automatically tracked 3D body postures and hand shapes using a Vicon high definition sensor. The body feature includes 3D joint velocities for left/right elbows and wrists, and is represented as a 12D input feature vector. The hand feature includes probability estimates of five predefined hand shapes: opened/closed-palm,thumb-up/down, and "no hand".

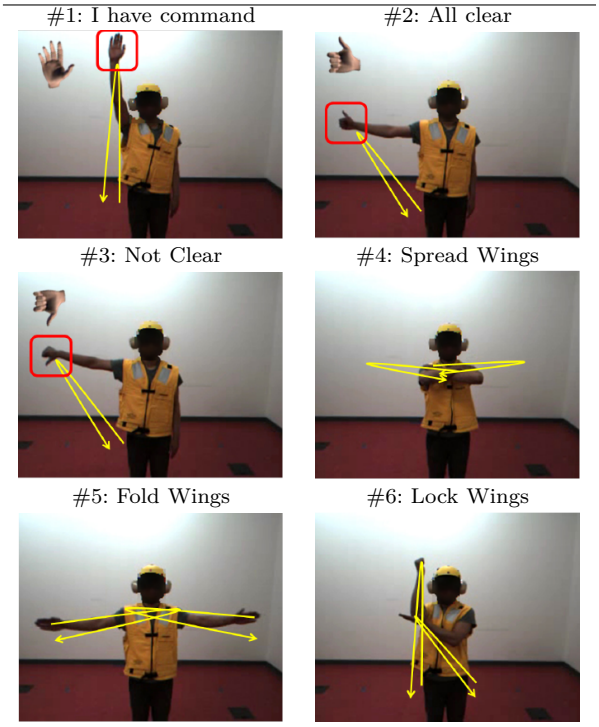

TABLE 1. The 6 selected gestures from the NATOPS data base; figures have been extracted from (5)

The simplified version of the NATOPS database we have produced includes the gesture pair (i.e., #1 − #2, #3−#4, and #5−#6). As highlighted in (17), the gestures (#4, #5) and (#2, #3) are very similar and represent the intricacy of the entire set.

A good simplification of the complexity of the human body is to estimate only the position of joints since they apparently provide a sufficient representation of the human posture (18). In our database, each gesture is represented by 24-dimensional feature vector, composed with the 3D-coordinates of hands (HandTips, thumbs and wrists) and arms (elbows). Figure 1 presents the location on the skeleton shape of the 3D-coordinates as blue (dark) dots.

The motion capture has been performed using a Kinect 2 sensor that produces a stream of 30 frames per second. Each frame consists in a 24 dimensional feature vector.

Isolated gesture dataset[1]. 20 subjects have been selected (15 male and 5 female) to perform in front of the sensor (at a three meters distance) the six selected NATOPS gestures. Each subject repeated each gesture three times. Hence the isolated gesture dataset is composed of 360 gesture utterances that have been manually segmented to a fixed length of 51 frames (1.7 sec. duration).

Gesture sequence dataset[1]. The same 20 subjects also performed 3 times the previous mentioned gestures in a continual mode. The obtained dataset consists of 60 samples of motion data stream.

1. This dataset will be made available for the community at the earliest feasible opportunity

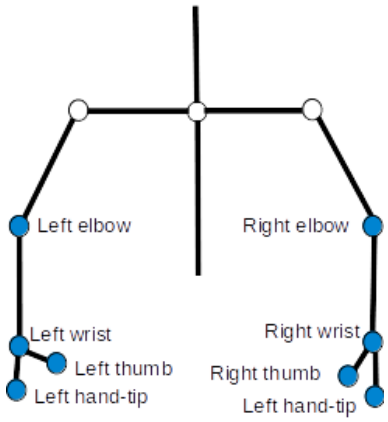

Figure 1. Partial skeleton reconstructed from motion data captured from the Kinect 2 sensor. Blue dots (dark) represent the 3D joint positions tracked during the capture of the gestures.

4. Methodology

4.1. Models

We hereinafter introduce the general classification models that we have used to learn and recognize isolated gestures and gestures in stream, namely first near neighbor with Dynamic time Warping (DTW) measure, first near neighbor with Kernelized DTW (KDTW), Hidden Markov Model (HMM), Hidden Conditional Random Fields (HCRF), Support Vector Machine (SVM) associated to DTW and KDTW.

4.1.1. HMM. An HMM (19) models a sequence of observations $X = \{x_t\}_{t=1}^{T}$ by assuming that there is an underlying sequence of states $Y = \{y_t\}_{t=1}^{T}$ drawn from a finite state set S. To model the joint distribution $p(y, x)$ tractably, an HMM makes two independence assumptions. First, it assumes that each state depends only on its immediate predecessor. Second, an HMM assumes that each observation variable x_t depends only on the current state y_t. With these assumptions, HMM can be specified using three probability distributions: first, the distribution $p(y_1)$ over initial states; second, the transition distribution $p(y_t|y_{t-1})$; and finally, the observation distribution $p(x_t|y_t)$. Therefore, the joint probability of a state sequence y and an observation sequence x factorizes as :

$$p(y, x) = \prod_{t=1}^{T} p(y_t|y_{t-1})p(x_t|y_t), \tag{1}$$

Each gesture is characterized by an HMM model and for each HMM, two procedures are required. In the training phase, the aim is to adjust the model parameters $\lambda = (A, B, \pi)$ in order to maximize $P(O|\lambda)$, where A is the state transition probability distribution, B is the observation symbol probability distribution and π is the initial state distribution. No tractable algorithm is known for solving this problem exactly, but a local maximum likelihood can be derived efficiently using the Baum-Welch algorithm (20). The Baum–Welch algorithm finds a local maximum for $\theta^* = \arg\max_\theta P(Y|\theta)$ (i.e. the HMM parameters θ that maximise the probability of the observation).

4.1.2. HCRF. The task performed by HCRF (21) is to predict the class y from the data x, where y is an element of the set Y of possible gesture labels and x is the set of vectors of temporal observations $x = \{x_1, x_2, ..., x_m\}$. Each local observation x_j is represented by a feature vector $\Phi(x_j) \in Rd$ where d is the dimensionality of the representation. The training set contains a set of labeled samples (x_i, y_i), for $i = 1...n$ where $y_i \in Y$ and $x_i = \{x_{i,1}, x_{i,2}, ..., x_{i,m}\}$. For any x_i a vector of latent variables $h = \{h_1, h_2, ..., h_m\}$ is assumed, providing the state sequence of the data. Each possible value for h_j is member of a finite set H of possible hidden states. HCRF is defined by a conditional probabilistic model :

$$p(y, h|x, \theta) = \frac{\exp^{\psi}(y, h, x; \theta)}{\sum_{y',h} \exp^{\psi}(y', h, x; \theta)} \tag{2}$$

Here θ are the parameters of the model, and $\Psi(y, h, x; \theta) \in R$ is a potential function parameterized by θ. The function $P(y|x, \theta)$ is defined by a summation over the h variables applied to the precedent equation.

$$p(y|x, \theta) = \sum_{y} p(y, h|x, \theta) = \frac{\exp^{\psi}(y, h, x; \theta)}{\sum_{y',h} \exp^{\psi}(y', h, x; \theta)} \tag{3}$$

Given a new test example x and parameter values θ^* induced from the training set, the label for the example is taken to be $\arg\max_{y \in Y} P(y|x, \theta^*)$. In the training phase, the following objective function is used for the estimation of parameter values θ :

$$L(\theta) = \sum_{i} \log P(y_i|x_i, \theta) - \frac{1}{2\sigma^2}||\theta||^2 \tag{4}$$

where the first term in (4) is the log-likelihood of the data. The second term is the log of a Gaussian prior with variance σ^2. Under this criterion gradient ascent can be used to search for the optimal parameter values $\theta^* = \arg\max_\theta L(\theta)$.

4.1.3. Support Vector Machine and time elastic kernels. The Support Vector Machine (SVM) is a state-of-the-art classification method introduced in 1992 by Boser, Guyon, and Vapnik (22). An SVM classifies data by finding the best hyperplane, meaning the largest margin, that separates all data points belonging to the first class from those belonging to the other class. Hence, a Support Vector Machine seeks a decision function f which is defined in the space spanned by the kernel basis functions $K(x, x_i)$ of the support vectors x_i:

$$y = f(x) = \sum_{i=1}^{n} w_i * K(x, x_i) + b. \tag{5}$$

In 1995, to cope with non-separable cases, (23) suggested a modified maximum margin idea that allows for mislabeled examples. If there exists no hyperplane that can split the "positive" and "negative" examples, the "Soft Margin"

method will choose a hyperplane that splits the examples as properly as possible, while still maximizing the distance to the nearest proper split examples. By solving for the Lagrangian dual of the optimization equivalent problem, one obtains the simplified problem

$$\max\ L(\alpha_1 \dots \alpha_n) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i \alpha_i K(x_i, x_j) y_j \alpha_j$$

$$\text{subject to } \sum_{i=1}^{n} \alpha_i y_i = 0, \text{ and } 0 \leq \alpha_i \leq C \text{ for all } i.$$

Where C is a penalty cost associated to the so-called slack variables. Since the dual minimization problem is a quadratic function of the α_i subject to linear constraints, it is efficiently solvable by quadratic programming algorithms. In this paper, the kernel function $K(.,.)$ refers to the time elastic kernels defined below (K_{dtw} and K_{rdtw}).

Dynamic Time Warping (DTW), (24), (25), by far the most used elastic measure, is defined as

$$\begin{aligned} d_{dtw}(X_p, Y_q) &= d_E^2(x(p), y(q)) \\ &+ \text{Min} \begin{cases} d_{dtw}(X_{p-1}, Y_q) & sup \\ d_{dtw}(X_{p-1}, Y_{q-1}) & sub \\ d_{dtw}(X_p, Y_{q-1}) & ins \end{cases} \end{aligned} \quad (6)$$

where $d_E(x(p), y(q)$ is the Euclidean distance (or its square) defined on $\mathbb{R}^k$ between the two samples in sequences X and Y taken at times p and q respectively. Besides the fact that this measure does not respect the triangle inequality, it does not directly define a positive definite kernel.

Regularized DTW (KDTW): (26), (27) have successfully proposed guidelines to regularize kernels constructed from time elastic measures such as DTW. KDTW is an instance of such regularized kernel derived from (27), and having proved to be quite efficient for isolated gesture recognition (18), takes the following form, which relies on two recursive terms :

$$\mathcal{K}_{rdtw}(X_p, Y_q) = K^{xy}_{rdtw}(X_p, Y_q) + K^{xx}_{rdtw}(X_p, Y_q) \quad (7)$$

$$K^{xy}_{rdtw}(X_p, Y_q) = \frac{1}{3} e^{-\nu d_E^2(x(p), y(q))} \sum \begin{cases} h(p-1, q) K^{xy}_{rdtw}(X_{p-1}, Y_q) \\ h(p-1, q-1) K^{xy}_{rdtw}(X_{p-1}, Y_{q-1}) \\ h(p, q-1) K^{xy}_{rdtw}(X_p, Y_{q-1}) \end{cases}$$

$$K^{xx}_{rdtw}(X_p, Y_q) = \frac{1}{3} \sum \begin{cases} h(p-1, q) K^{xx}_{rdtw}(X_{p-1}, Y_q) e^{-\nu d_E^2(x(p), y(p))} \\ \Delta_{p,q} h(p, q) K^{xx}_{rdtw}(X_{p-1}, Y_{q-1}) e^{-\nu d_E^2(x(p), y(q))} \\ h(p, q-1) K^{xx}_{rdtw}(X_p, Y_{q-1}) e^{-\nu d_E^2(x(q), y(q))} \end{cases} \quad (8)$$

where $\Delta_{p,q}$ is the Kronecker's symbol, $\nu \in \mathbb{R}^+$ is a stiffness parameter which weights the local contributions, i.e. the distances between locally aligned positions, and $d_E(.,.)$ is a distance defined on $\mathbb{R}^k$. $h(.,.) \in \{0, 1\}$ is used to specify a symmetric corridor around the main diagonal of the alignment matrix, which allows for computational speed-up. The initialization is simply

$$K^{xy}_{rdtw}(X_0, Y_0) = K^{xx}_{rdtw}(X_0, Y_0) = 1 \quad .$$

The main idea behind this line of regularization is to replace the operators min or max (which prevent symmetrization) by a summation operator ($\sum$). This leads to consider, not only the best possible alignment, but also all the best (or nearly the best) paths by summing up their overall cost. The parameter ν is used to tune the local matches, thus penalizing more or less alignments moving away from the optimal ones. This parameter can be easily optimized through a cross-validation.

Time elastic kernels: we consider in this paper only the exponential kernel (Gaussian or RBF-type) constructed from the two previous time elastic measures d_{dtw} and K_{rdtw}, i.e. $K_{dtw}(.,.) = e^{-d_{dtw}(.,.)/\sigma}$. For the regularized DTW kernel, a data dependent normalization heuristic is required and the final kernel takes the form $K_{rdtw}(.,.) = e^{\beta \mathcal{K}^{\alpha}_{rdtw}(.,.)/\sigma}$, with

- $\alpha = 1/log(max(\mathcal{K}_{rdtw}(.,.))/min(\mathcal{K}_{rdtw}(.,.)))$ and
- $\beta = exp(-\alpha \cdot log(min(\mathcal{K}_{rdtw}(.,.))))$,

where min, max are taken over all the training data pairs.

5. Experiment

5.1. Assessing pattern recognition in stream

All recognition models considered in this paper provide as output a scoring file that contains for each processed time t a vector of scores (either a distance, a probability or any similarity or dissimilarity value) $< S_i(t) >_{i=1..nC}$, where nC is the number of categories and $S_i(t)$ is the score for the i^{th} category at time t. Two meta parameters are defined to labelled the stream from this vector or scores: S_0, the similarity threshold value and the minimal duration value T_0. Given these two thresholds, and supposing the score value has to be maximized, the labelling algorithm (LA) consists in outputting label C_i if and only if $S_i(t) > S_j(t) \forall j \neq i$ and for $t \in [t_n, t_{n+1}]$ such that $t_{n+1} - t_n > T_0$, as depicted in Figure 2 for a binary classification problem.

The alignment of the ground truth labels with the predicted labelled stream $< S_i(t) >_{i=1..nC}$ provides confusion matrix and conventional assessment measures, namely the average error rate (ERR), the macro-average precision (PRE), the macro-average recall (REC) and the F_1 measure ($F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$). In addition, the average latency estimation (LAT) and the average match duration (DUR) are given for the gesture spotting and recognition in stream task. When a match is detected, the latency and the duration of the match measures are defined accordingly to Figure 3. The latency is defined as the difference of the matched mid-segment time locations.

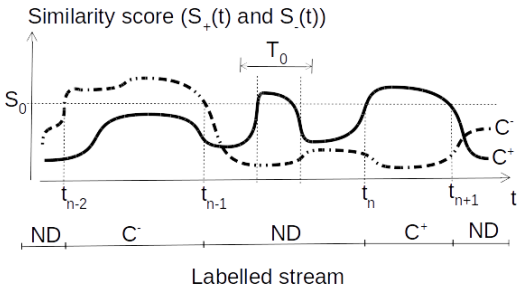

Figure 2. Stream labelling for a two categories (C^+ and C^-) problem

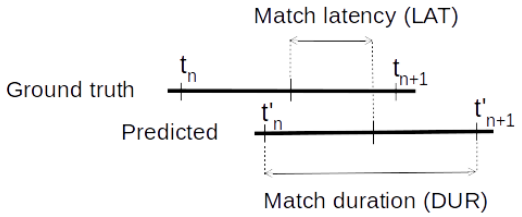

Figure 3. Matching process between ground truth and predicted labels

5.2. Experimental conditions and Results

For the HMM and HCRF models, following an heuristic approach, the maximal number of iterations relating to the training process was set to 300 and the number of hidden states was initialized to 6, which is adequate for the complexity of the gestures that we consider. The SVM implementation that we have used is a modified version of libsvm (28) able to cope with stream processing. To evaluate comparatively the proposed classification methods, we have performed a subject cross validation experiment consisting in 100 tests: for each test, 10 subjects have been randomly drawn for training and the remaining 10 subjects have been retained for testing. Only the isolated gesture dataset is used for training the classifier. The ν parameter of the KDTW kernel as well as the SVM meta parameter (RBF bandwidth σ and C) are optimized using a leave one subject out cross-validation on the train isolated gesture dataset. Nevertheless, the gesture sequence dataset is used also for deriving the T_0 and S_0 meta parameters required, for all models, to complete the stream recognition task. To that end, we have carried out, for each test, a 10-folds cross validation scheme such that the classification error rate is minimized on the sequences belonging to the test fold. Note that once T_0 and S_0 have been set up during training, the labelling algorithm (LA) can be performed on-line during testing.

Tables 2 and 3 give, for the six tested recognition methods and the two considered tasks, the average, evaluated on the 100 tests, of the assessment measures presented above. These results show that the kernelization of DTW (KDTW) associated to a SVM classifier significantly outperforms for the both tasks all the other models. Furthermore KDTW-SVM is quite robust and adapted to the streaming task, with only 3% error rate loss comparatively to the isolated gesture recognition task. It also outperforms others state-of-the-art methods (13)(14) (29) applied to similar skeleton-based data sets and treating the motion data as undivided whole set. The main limitation of this model, comparatively to the others, is its latency. As already reported in the literature (see for instance (18) for isolated gesture recognition) DTW does not couple well with SVM (the likely cause being its indefiniteness). HCRF performs quite well on the isolated gesture task but is outperformed by the 1NN classifiers on the streaming task. The HMM is the poorer model on the two considered tasks, although its latency is minimal.

TABLE 2. Isolated gestures recognition assessment measures

Method	ERR mean ‖ std	PRE	REC	F1
1NN DTW	.134 ‖ .012	.869	.866	0.867
1NN KDTW	.128 ‖ .016	.876	.972	.874
HMM	.318 ‖ .049	.776	.727	.750
HCRF	.128 ‖ .039	.898	.891	.894
SVM DTW	.146 ‖ .015	.871	.854	.862
SVM KDTW	.051 ‖ .015	.952	.949	.951

TABLE 3. Gestures spotting and recognition in stream assessment measures

Method	ERR mean ‖ std	PRE	REC	F1	LAT	DUR
1NN DTW	.204 ‖ .015	.868	.799	.832	20.9	39.9
1NN KDTW	.189 ‖ .016	.878	.817	.846	20.9	39.7
HMM	.404 ‖ .024	.721	.600	.654	4.8	42.4
HCRF	.244 ‖ .043	.840	.757	.796	16.8	50.8
SVM DTW	.291 ‖ .086	.783	.714	.741	21.0	50.5
SVM KDTW	.107 ‖ .021	.947	.893	.920	25.4	41.6

6. Conclusion

We have proposed a framework dedicated to the assessment of gesture spotting and recognition in stream. Despite its dedicated nature, we believe that this framework is quite general purpose and can be used to benchmark continuous pattern detection and recognition in stream. The case study we have carried out using this benchmark consolidate earlier results obtained on isolated gesture (18) recognition and extend them to stream processing condition. Results show that the regularized kernelized version of the DTW measure (KDTW) is particularly efficient on the tested stream data comparatively to the other experimented recognition models (DTW, HMM, HCRF). As a perspective we plan to enhance our excerpt of the NATOPS dataset in the near future to complete the 24 gesture vocabulary.

Acknowledgments

The authors would like to thank Thales Optronics for their partial support of this research work.

References

[1] I. Laptev, “On space-time interest points,” Int. J. Comput. Vision, vol. 64, no. 2-3, pp. 107–123, Sep. 2005.

[2] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies." in CVPR. IEEE, 2008.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. of the 2005 Conf. on Computer Vision and Pattern Recognition, Vol. 1, ser. CVPR '05. Washington, DC, USA: IEEE Comp. Soc., 2005, pp. 886–893.

[4] F.-F. Li and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in Proc. of the Conference on Computer Vision and Pattern Recognition, Vol. 2, ser. CVPR '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 524–531.

[5] Y. Song, D. Demirdjian, and R. Davis, "Tracking body and hands for gesture recognition: NATOPS aircraft handling signals database," in IEEE Intern. Conf. on Automatic Face and Gesture Recognition (FG 2011), Santa Barbara, USA, March 2011, 2011, pp. 500–506.

[6] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," International Journal of Computer Vision, vol. 29, pp. 5–28, 1998.

[7] V. N. Vapnik, Ed., The Nature of Statistical Learning Theory. Springer, 1995.

[8] M. Müller, A. Baak, and H.-P. Seidel, "Efficient and robust annotation of motion capture data," in Proc. of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, ser. SCA '09. New York, NY, USA: ACM, 2009, pp. 17–26.

[9] S. Wang, Y. Yang, Z. Ma, X. Li, C. Pang, and A. G. Hauptmann, "Action recognition by exploring data distribution and feature correlation." in CVPR. IEEE, 2012, pp. 1370–1377.

[10] D. Gong, G. G. Medioni, S. Zhu, and X. Zhao, "Kernelized temporal cut for online temporal segmentation and recognition," in Computer Vision - ECCV 2012 - 12th Eur. Conf. on Computer Vision, Florence, Italy, October 7-13, 2012, Proc., Part III, 2012, pp. 229–243.

[11] T. Hofmann, B. Schölkopf, and A. Smola, "Kernel methods in machine learning," Annals of Statistics, vol. 36, no. 3, pp. 1171–1220, Jun. 2008.

[12] A. Gretton, K. Fukumizu, Z. Harchaoui, and B. Sriperumbudur, "A fast, consistent kernel two-sample test," in Advances in Neural Information Processing Systems 22. Red Hook, NY, USA: Curran, 2009, pp. 673–681.

[13] X. Zhao, X. Li, C. Pang, X. Zhu, and Q. Z. Sheng, "Online human gesture recognition from motion data streams," in Proc. of the 21st ACM Intern. Conf. on Multimedia, ser. MM '13. New York, NY, USA: ACM, 2013, pp. 23–32.

[14] X. Zhao, X. Li, C. Pang, Q. Z. Sheng, S. Wang, and M. Ye, "Structured streaming skeleton – a new feature for online human gesture recognition," ACM Trans. Multimedia Comput. Commun. Appl., vol. 11, no. 1s, pp. 22:1–22:18, Oct. 2014.

[15] Y. Yang, Z. Ma, A. G. Hauptmann, and N. Sebe, "Feature selection for multimedia analysis by sharing information among multiple tasks," IEEE Trans. Multimedia, vol. 15, no. 3, pp. 661–669, 2013.

[16] Z. Liu, "Msr action recognition datasets and codes, http://research.microsoft.com/en-us/um/people/zliu/actionrecorsrc/," 2014.

[17] Y. Song, L.-P. Morency, and R. Davis, "Multi-view latent variable discriminative models for action recognition," in Proceedings of the 25th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, June 2012.

[18] P.-F. Marteau, S. Gibet, and C. Reverdy, "Down-sampling coupled to elastic kernel machines for efficient recognition of isolated gestures," in 22nd Intern. Conf. on Pattern Recognition, ICPR, Stockholm, Sweden, 2014, 2014, pp. 363–368.

[19] L. R. Rabiner, "Readings in speech recognition," A. Waibel and K.-F. Lee, Eds. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990, ch. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, pp. 267–296. [Online]. Available: http://dl.acm.org/citation.cfm?id=108235.108253

[20] P. M. Baggenstoss, "A modified baum-welch algorithm for hidden markov models with multiple observation spaces," IEEE Trans. Speech and Audio Processing, vol. 9, no. 4, pp. 411–416, 2001. [Online]. Available: http://dx.doi.org/10.1109/89.917686

[21] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell, "Hidden conditional random fields," IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 10, pp. 1848–1852, Oct. 2007. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2007.1124

[22] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in Proc. Workshop on Computational Learning Theory, COLT '92. New York, NY, USA: ACM, 1992, pp. 144–152.

[23] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273–297.

[24] V. M. Velichko and N. G. Zagoruyko, "Automatic recognition of 200 words," International Journal of Man-Machine Studies, vol. 2, pp. 223–234, 1970.

[25] H. Sakoe and S. Chiba, "A dynamic programming approach to continuous speech recognition," in Proc. of the 7th Intern. Congress of Acoustic, 1971, pp. 65–68.

[26] M. Cuturi, J.-P. Vert, O. Birkenes, and T. Matsui, "A Kernel for Time Series Based on Global Alignments," in Proc. of ICASSP'07. Honolulu, HI: IEEE, April 2007, pp. II–413 – II–416.

[27] P.-F. Marteau and S. Gibet, "On Recursive Edit Distance Kernels with Application to Time Series Classification," IEEE Trans. on Neural Networks and Learning Systems, pp. 1–14, Jun. 2014.

[28] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," ACM Trans. Intell. Syst. Technol., vol. 2, no. 3, pp. 27:1–27:27, May 2011.

[29] C. Ellis, S. Z. Masood, M. F. Tappen, J. J. Laviola, Jr., and R. Sukthankar, "Exploring the trade-off between accuracy and observational latency in action recognition," Int. J. Comput. Vision, vol. 101, no. 3, pp. 420–436, Feb. 2013.