Analysis and classification of MoCap data by Hilbert space embedding-based distance and multikernel learning

J. D. Pulgarin-Giraldo[1,2(✉)], A. M. Alvarez-Meza[3], S. Van Vaerenbergh[4], I. Santamaría[4], and G. Castellanos-Dominguez[2]

[1] G-BIO Research Group, Universidad Autónoma de Occidente, Cali, Colombia.
[2] Signal Processing and Recognition Group, Universidad Nacional de Colombia, Manizales, Colombia.
jdpulgaring@unal.edu.co
[3] Faculty of Engineering, Universidad Tecnológica de Pereira, Pereira, Colombia.
[4] Dept. of Communications Engineering, University of Cantabria, Santander, Spain.

Abstract. A framework is presented to carry out prediction and classification of Motion Capture (MoCap) multichannel data, based on kernel adaptive filters and multi-kernel learning. To this end, a Kernel Adaptive Filter (KAF) algorithm extracts the dynamic of each channel, relying on the similarity between multiple realizations through the Maximum Mean Discrepancy (MMD) criterion. To assemble dynamics extracted from all MoCap data, center kernel alignment (CKA) is used to assess the contribution of each to the classification tasks (that is, its relevance). Validation is performed on a database of tennis players, performing a good classification accuracy of the considered stroke classes. Besides, we find that the relevance of each channel agrees with the findings reported in the biomechanical analysis. Therefore, the combination of KAF together with CKA allows building a proper representation for extracting relevant dynamics from multiple-channel MoCap data.

Keywords: Multichannel data, kernel adaptive filters, maximum mean discrepancy, center kernel alignment.

1 Introduction

In human action recognition using MoCap data, the primary efforts are directed at extracting adequately robust dynamics to model the movements accomplished under given actions [1]. In practice, the models are mostly oriented to classify accurately executed actions, accounting for the relevance of the extracted feature sets but voiding the contribution of the body segments and articulations (i.e., channel relevance). One of the restraints to assess the channel relevance is the need of developing spatial filtering methods that may provide an adequate interpretation of biomechanical generation.

To deal with this issue, compact, meaningful *dictionaries* or *codebooks*, that match physiological principles are built. To this end, Kernel Adaptive Filters

(KAFs) are widely employed in time-series prediction task that enables encoding the salient elements of signals [2], avoiding the segmentation step within the feature extraction stage of human action recognition [3]. Furthermore, the combination of multiple dynamic models by kernels methods can be implemented through different feasible approaches like CKA proposed in [4].

Provided a set of output labels, the supervised CKA algorithm employs a distance that measures the dissimilarity/similarity between each basis kernel and the target kernel, yielding the combination weights that estimate the relevance of each input kernel. In channel relevance tasks of MoCap multichannel time series, however, construction of adequate basis kernel sets, which must be in independent from each other, is still a challenging issue.

Here, to reveal the contribution of channels involved in each action execution, a channel relevance methodology is presented to improve the performance of prediction and classification tasks using MoCap multichannel data. Initially, from input data, the Kernel Adaptive Filter build a *codebook* set as well a vector of predicted outputs, which are further mapped in a Reproducing Kernel Hilbert Space. Relying on the similarity between multiple realizations through the Maximum Mean Discrepancy criterion, we construct a basis kernel per channel. Then, CKA aligns the whole basis kernel set, using the label set. As a result, we find that the relevance of each channel agrees with the findings reported in the biomechanical analysis. Therefore, the combination of KAF together with CKA allows building a proper representation for extracting relevant dynamics from multiple-channel MoCap data.

2 Theoretical framework

2.1 Dynamical channel model encoded by kernel adaptive filtering

We assume a scenario in which a set of J time series $\mathbf{x}_j[t]$ are obtained from sensor measurements, with $j = 1, \ldots, J$. For each time series, T time steps are available, i.e. $t = 1, \ldots, T$. We collect the entire set of measurements in the matrix $\boldsymbol{X} \in \mathbb{R}^{J \times T}$, which contains the J time series as its rows as follows:

$$\boldsymbol{X} = \begin{bmatrix} x_1[1] & x_1[2] & \ldots & x_1[T] \\ x_2[1] & x_2[2] & \ldots & x_2[T] \\ \vdots & \vdots & \ddots & \vdots \\ x_J[1] & x_J[2] & \ldots & x_J[T] \end{bmatrix} \tag{1}$$

Thorough this paper, we assume that multiple sets are available, where the n-th set is represented as $\boldsymbol{X}^n$, with $n = 1, \ldots, N$. Also, to indicate that a time series belongs to a particular set n, we use notation $\mathbf{x}_j^n[t]$.

With the aim of modeling properly each time series $\mathbf{x}_j$, its dynamic behavior is represented through Kernel Adaptive Filters (KAFs) so that the problem non-linearities can be represented as a kernel expansion in terms of the training data:

$$f(\mathbf{x}_j) = \sum_{r=1}^{R} \alpha_r \kappa(\mathbf{x}_j[r], \mathbf{x}_j), \tag{2}$$

where α_r is built using kernel least-mean-square algorithms (KLMS). Here, we employ KAFs that enable tracking of non-stationary data with nonlinear relationships. Among KAF algorithms, we are interested in those that construct a dictionary set or *codebook* composed of R elements, each one including the most representative data points learned from the quantization process.

2.2 Model construction and similarity measure

The KRLS tracker introduced in [5],assumes a set of ordered input-output pairs $\{\mathbf{x}_j[t], y_j[t]\}$ in which the input data is taken as the time-embedded version of the series with L lags, $\mathbf{x}_j[t] = [x_j[t], x_j[t-1], \ldots, x_j[t-L+1]]$, and the desired output is the next sample, $y_j[t] = x_j[t+1]$. In addition to the obtained channel predictor (see Eq. (2)), we get a codebook $\mathbf{c}_j[r]$ and their estimated latent function outputs or desired values $d[r]$, applying the KRLS tracker [5]. Consequently, we define a model associated to each time series as $\mathcal{M}_j = \{\mathbf{c}_j[r], d_j[r],\ r = 1, ..., R\}$.

Further, we perform the similarity measure between models. Namely, let us consider two different models $\mathbf{p}_r = (\mathbf{c}_p[r], d_p[r])$ and $\mathbf{q}_r = (\mathbf{c}_q[r], d_q[r])$. The elements of each model or model samples, as given by KRLST, are not ordered. Therefore, any permutation or reordering of the elements represents the same model. Bearing this in mind, we interpret each model as a cluster of points in the input space. We now define a mapping from the set of models $\mathcal{Z}$ to a RKHS as $\Phi : \mathcal{Z} \longrightarrow \mathcal{H}$, which maps $\{\mathbf{p}_r\}_{r=1}^R \longmapsto \{\Phi(\mathbf{p}_r)\}_{r=1}^R$. A model can be interpreted as a distribution function $\mathcal{P}$ from which R realizations are available. Then, to define a distance between models we resort to the Maximum Mean Discrepancy (MMD) defined by Gretton in [6]. Given two models $\mathcal{P}$ and $\mathcal{Q}$, the MMD criterion computes the distance between them as

$$\mathfrak{d}^2(\mathcal{P}, \mathcal{Q}) = \left\| \frac{1}{R} \sum_{r=1}^{R} \Phi(\mathbf{p}_r) - \frac{1}{R} \sum_{r=1}^{R} \Phi(\mathbf{q}_r) \right\|_2^2 . \tag{3}$$

Assuming a separable model that decouples the influence of the input and the output [7], the distance between models in Eq. (3) can be rewritten in terms of kernel matrices as

$$\mathfrak{d}^2(\mathcal{P}, \mathcal{Q}) = \frac{1}{R^2} \left(\mathbf{d}_p^T \boldsymbol{K}_{pp} \mathbf{d}_p + \mathbf{d}_q^T \boldsymbol{K}_{qq} \mathbf{d}_q - 2\mathbf{d}_p^T \boldsymbol{K}_{pq} \mathbf{d}_q \right), \tag{4}$$

where $\boldsymbol{K}_{pq}(r, r') = \exp(-\|\mathbf{c}_p[r] - \mathbf{c}_q[r']\|^2 / 2\sigma_c^2)$, and $\mathbf{d}(r, r') = \mathbf{d}[r]\mathbf{d}[r']$ is a linear kernel for the output of each model.

2.3 Relevance assessment by multikernel learning

Let $\boldsymbol{X}^n \in \mathbb{R}^{J \times T}$, $n = 1, \ldots, N$ be a labeled set of J-dimensional time series. For the n-th multichannel time series we have a collection of J models that we denote as $\{\mathcal{M}_j[n]\}_{j=1}^J$. Let us denote as $\mathbf{K}_j$ the $N \times N$ kernel matrix that measures the (di)similarities for the j-th channel between the N time series

in the training data set. The element (n, m) of this kernel matrix is given by $\mathbf{K}_j(n,m) = \exp - \left(\frac{\mathfrak{d}^2(\mathcal{M}_j[n],\mathcal{M}_j[m])}{2\sigma_{\mathfrak{d}}^2}\right)$, where $\mathfrak{d}^2(\mathcal{M}_j[n], \mathcal{M}_j[m])$ is the pairwise distance between models described in Section 2.2 (Eq. (4)).

To combine the information from the J channels we propose to use a multi-kernel constructed as follows

$$\hat{\boldsymbol{K}} = \sum_{j=1}^{J} \alpha_j \boldsymbol{K}_j, \tag{5}$$

where the weights α_j $j = 1, \ldots, J$ are yet to be determined. To find informative weights that allow us to quantify the relevance of individual channels, we propose to use a centered kernel alignment procedure [4]. The basic idea is to find the optimal α_j^* maximizing the alignment between the multikernel matrix $\boldsymbol{K}$ and the target kernel matrix $\boldsymbol{K_l} = \boldsymbol{l}\boldsymbol{l}^T$, which is calculated from the known label classes $\boldsymbol{l} = \{l[i]\}_{i=1}^N$. For a given set of weights α_j, the centered correlation or alignment between matrix kernels $\boldsymbol{K}$ and $\boldsymbol{K_l}$ is given by

$$\rho\left(\boldsymbol{K}, \boldsymbol{K_l}; \alpha\right) = \frac{\langle \boldsymbol{HKH}, \boldsymbol{HK_lH} \rangle}{\|\boldsymbol{HKH}\|_F \, \|\boldsymbol{HK_lH}\|_F}, \qquad \rho \in [0, 1] \tag{6}$$

where $\boldsymbol{H}{=}\boldsymbol{I} - N^{-1}\mathbf{1}\mathbf{1}^\top$ is a centering matrix, $\boldsymbol{I}{\in}\mathbb{R}^{N\times N}$ is the identity matrix, $\mathbf{1}{\in}\mathbb{R}^N$ is an all-ones vector, and notations $\langle\cdot,\cdot\rangle$ and $\|\cdot,\cdot\|_F$ stand for the inner product and the Frobenius norm, respectively.

Then, the optimal relevance weights are $\alpha^* = \text{argmax } \rho(\boldsymbol{K}, \boldsymbol{K_l}, \alpha)$ subject to the constraint $||\alpha^*|| = 1$. This problem is solved by the Centered Kernel Alignment (CKA) algorithm [4].

3 Experimental setup

3.1 Database description

The data were collected from 17 high-performance tennis players of the Caldas-Colombia tennis league. Infrared videography with 23 optical markers was collected from six cameras to acquire sagittal, frontal, and lateral planes and skeleton and multichannel time series were estimated in Optitrack Arena®. All subjects were encouraged to hit the ball with the same velocity and action just as they would in a match. They were instructed to hit one series continuously by 30 seconds of each indicated stroke. The strokes indicated in each record were: serve, forehand, backhand, volley, backhand volley and smash.

3.2 MoCap data preprocessing

Let $\boldsymbol{U}{\in}\mathbb{R}^{T\times(J\times D)}$ be a multi-channel input matrix that holds T frames and $J{\times}D$ channels, where J is the number of joints of the body model. Each $\boldsymbol{U}_j = \left\{\mathbf{u}_{ij}{\in}\mathbb{R}^D : i{\in}T\right\}$ assembles time behavior of D-dimensional body-joint j. Initially, all channels are centered respect to the limb center. Then, to describe

the time behavior of the j-th body-joint from $\boldsymbol{U}_j$, we perform a dimensional reduction stage from $\mathbb{R}^D \rightarrow \mathbb{R}$ to obtain a compact representation of its time behavior. In this case, from the covariance matrix $\boldsymbol{W} \in \mathbb{R}^{D \times D}$ we consider only the first principal eigenvector $\mathbf{w}_1$, obtained from the first column of the covariance matrix. Then, we obtain the linear projection $\mathbf{x}_j = \boldsymbol{U}_j \mathbf{w}_1$, where $\mathbf{w}_1 \in \mathbb{R}^{D \times 1}$.

3.3 Model estimation and similarity measure

We compute each model $\mathcal{M}_j$ through a KRLST algorithm with parameters set as follows: forgetting factor 1, time embedding $L = 6$, codebook size $R = 50$, regularization parameter $\lambda = 10^{-6}$, a Gaussian kernel with σ calculated as the median value of channel $\mathbf{x}_j$ and the initial codebooks are built directly from the input time series $\mathbf{x}_j \in \mathbb{R}^{T \times 1}$. Each model is validated doing a simple task: predict $x(t+1)$ from data available up to time t.

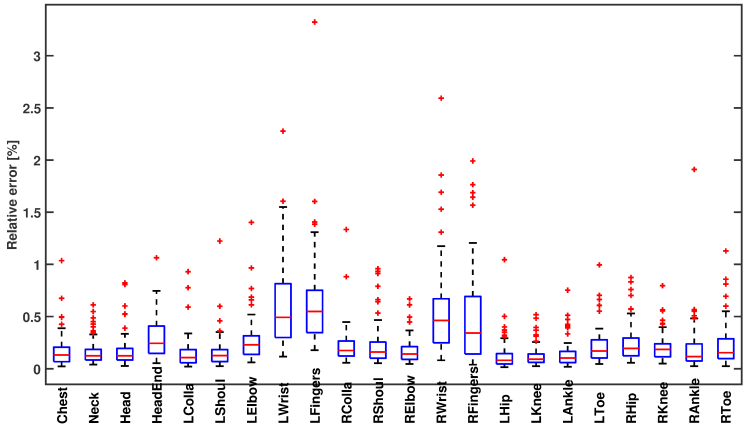
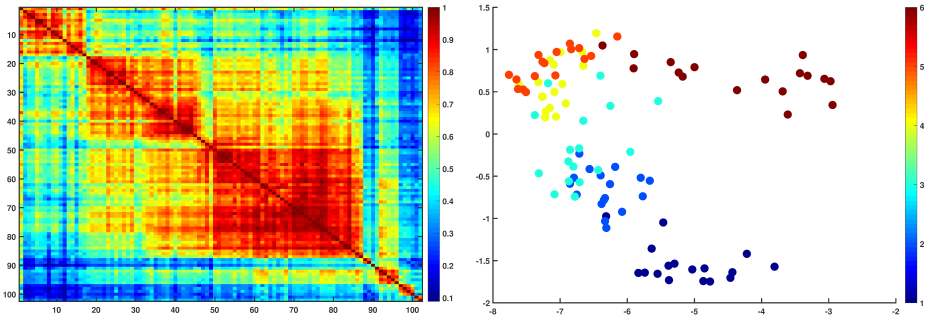


Fig. 1. Relative error results for each joint model $\mathcal{M}_j^n$ estimated over N records with six different classes

Fig. 1 shows the mean prediction error in each channel j for all sets of multichannel data, in this case, N=102. Although the number of outliers looks high, it shows a low and regular mean error, which is significant due to the high variability of both: inter-subject and inter-class variability. Besides, our approach works with the 30 seconds full-long one take videos where several and continuous actions were recorded. There are approximately 12 to 16 strokes in each individual record. It is worth saying that segmentation and selection of actions are not required in our modeling process.

Besides, our proposed functional $\mathfrak{d}^2$ allows us to construct a kernel similarity measure $\kappa(\mathcal{M}_j[n], \mathcal{M}_j[m])$ which highlights each group of actions without previous information about the classes. In Fig. 2(a) we can see the block diagonal structure of the Gram matrix $\boldsymbol{K}$ constructed over records of the right wrist joint.

In fact, KPCA 2D-embedding in Fig. 2(b) shows the separability between groups of records that are colored according to its true label.



(a) Kernel similarity matrix (b) KPCA 2D-embedding

Fig. 2. Model similarity comparison for right wrist body-joint over 102 records. In both plots, most classes of 17 strokes records are distinguishable.

4 Relevance and classification results

Once the multikernel $\hat{\boldsymbol{K}}$ from Eq. (5) is constructed it allows us compare multi-channel data, so that we can apply any kernel-based classifier. In this work, we use a kernel nearest neighbor (KNN). The KNN classifier finds the k samples in the training dataset closest to test data (with maximum similarity) and carries out majority vote. Classification performance and relevance are computed using a cross-validation scheme.

Fig. 3 shows the attained α values in a boxplot. Particularly, the body joints at the end of the limbs are the most relevant. These channels highlight the difference between the six classes of action executed. Nonetheless, the variability observed in the most relevant channels implies a strong dependency in the execution, namely, the angle of the racquet in the hit moment varies with the wrist and fingers channels relation.

Regarding to the classification results, as can be seen in Fig. 4(a), accuracies over 90% are attained for a number of nearest neighbors ranging from 1 to 9. In Fig. 4(b), the lowest results must be analyzed in confrontation with the action, where backhand presents low ball speeds after the impact and it were closer to speeds obtained in volley strokes executions. Nevertheless, each record classified contains 12 to 16 continuously stroke executions without segmentation, so the confused actions depend of execution's speed after 30 seconds.

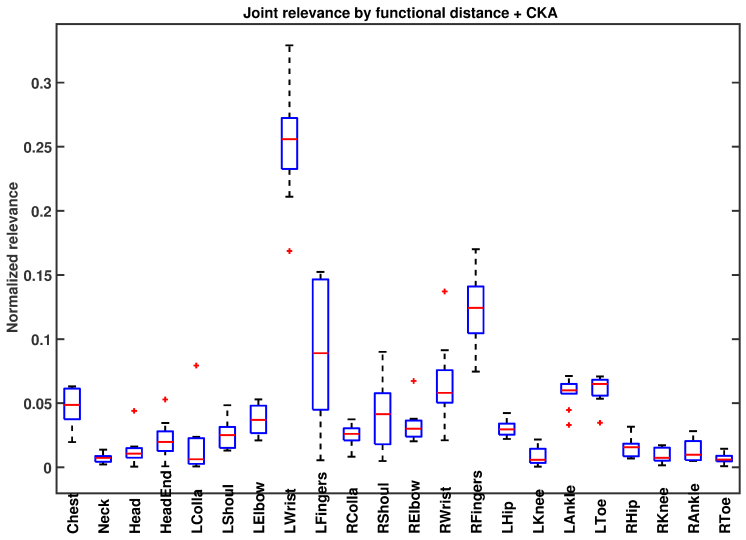
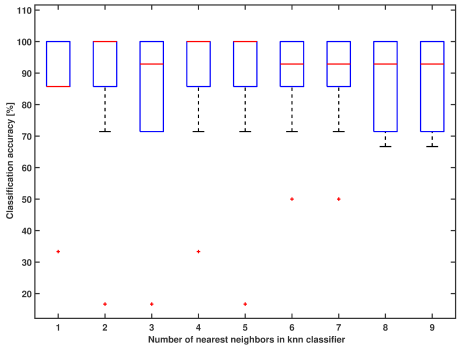


Fig. 3. Relevance body joint analysis in six activities. 10 folds in cross-validation were used over 102 records



(a) Classification performance versus number of nearest neighbors in KNN classifier classifier

	Serve	Forehand	Backhand	Volley	BH volley	Smash
Serve	100	0	0	0	0	0
Forehand	0	100	0	0	0	0
Backhand	0	0	100	0	0	0
Volley	0	0	0	100	0	0
BH volley	0	0	0	5.88	94.12	0
Smash	0	0	0	5.88	5.88	88.24

(b) Confussion matrix with three nearest neighbors. Accuracy results in %

Fig. 4. Classification results in six activities. 10 folds in cross-validation were used over 102 records.

4.1 Discussion and concluding remarks

The proposed framework for MoCap multichannel analysis presents a methodology that first: obtains an appropriate and individual representation of the dynamic of each channel; and second: this channel representation based on KAFs allows us to combine similarity between several realizations. In fact, this framework easily matches with a multikernel algorithm as CKA, which merges multiple channels into just one kernel that can be used in classification tasks. It can be

seen that CKA reveals the most significant channels in a set of actions, and these results are congruent with biomechanic theory in tennis actions execution [8].

This framework should be expanded to analyze optimal number and placement of sensors in human action recognition tasks, regardless of its source: optical markers, inertial sensors or depth cameras. Besides, human motion action involves an interaction between all body segments: every action has a biomechanical chain that produces it, so relevance of channels must give information about the most relevant body segments involved across the time. The results encourage us to develop an algorithm for biomechanical chain generation without kinetic information, just from skeleton representations of actions.

As future work, this framework must be validated in larger action datasets, as well as be evaluated in assessment motor disorders to check whether relevance shows alterations in specific body segments or articulations.

Acknowledgments. This work is supported by the project 36075 and mobility grant 8401 funded by Universidad Nacional de Colombia sede Manizales, by program "Doctorados Nacionales 2014" number 647 funded by COLCIENCIAS, as well as PhD financial support from Universidad Autónoma de Occidente.

Bibliography

[1] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition. *J. Vis. Comun. Image Represent.*, 25(1):24–38, Jan 2014.

[2] S. Van Vaerenbergh and I. Santamaría. A comparative study of kernel adaptive filtering algorithms. In *2013 IEEE DSP/SPE Meeting*, pages 181–186, August 2013. Software available at `https://github.com/steven2358/kafbox/`.

[3] J. D. Pulgarin-Giraldo, A. M. Alvarez-Meza, L. G. Melo-Betancourt, S. Ramos-Bermudez, and G. Castellanos-Dominguez. A similarity indicator for differentiating kinematic performance between qualified tennis players. In *CIARP*, volume 10125 of *Lecture Notes in Computer Science*, pages 309–317, November 2016.

[4] C. Cortes, M. Mohri, and A. Rostamizadeh. Algorithms for learning kernels based on centered alignment. *J. Mach. Learn. Res.*, 13(1):795–828, March 2012.

[5] S. Van Vaerenbergh, M. Lazaro-Gredilla, and I. Santamaria. Kernel recursive least-squares tracker for time-varying regression. *IEEE Trans. Neural Netw. Learning Syst.*, 23(8):1313–1326, August 2012.

[6] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *J. Mach. Learn. Res.*, 13:723–773, March 2012.

[7] M. A. Álvarez, L. Rosasco, and N. D. Lawrence. Kernels for vector-valued functions: A review. *Found. Trends Mach. Learn.*, 4(3):195–266, March 2012.

[8] J. Landlinger, S. Lindinger, T. Stoggl, H. Wagner, and E. Muller. Key factors and timing patterns in the tennis forehand of different skill levels. *J. Sports Sci. Med.*, 9:643–651, December 2010.