Identification of Multimodal Human-Robot Interaction Using Combined Kernels

Saith Rodriguez, Katherín Pérez, Carlos Quintero, Jorge López, Eyberth Rojas and Juan Calderón

Abstract In this paper we propose a methodology to build multiclass classifiers for the human-robot interaction problem. Our solution uses kernel-based classifiers and assumes that each data type is better represented by a different kernel. The kernels are then combined into one single kernel that uses all the dataset involved in the HRI process. The results on real data shows that our proposal is capable of obtaining lower generalization errors due to the use of specific kernels for each data type. Also, we show that our proposal is more robust when presented to noise in either or both data types.

1 Introduction

Interacting robots have been considered to play significant roles within the human society long before they actually existed. Applications range from education, service assistance, rescue and entertainment among many others [1]. Challenges in each specific application usually vary depending on the proximity of interaction between the robots and the humans (or other robots), the support of effective social interactions

S. Rodriguez (✉) · K. Pérez (✉) · C. Quintero (✉) · J. López (✉) · E. Rojas (✉) · J. Calderón (✉)
Universidad Santo Tomás, Bogotá, Colombia
e-mail: carlosquinterop@usantotomas.edu.co

S. Rodriguez
e-mail: saithrodriguez@usantotomas.edu.co

K. Pérez
e-mail: andrea.perez@usantotomas.edu.co

J. López
e-mail: jorgelopez@usantotomas.edu.co

E. Rojas
e-mail: eyberthrojas@usantotomas.edu.co

J. Calderón
University of South Florida, Tampa, FL, USA
e-mail: juancalderon@mail.usf.edu

V. Snášel et al. (eds.), *Innovations in Bio-Inspired Computing and Applications*,
Advances in Intelligent Systems and Computing 424,
DOI 10.1007/978-3-319-28031-8_23

and the robot's autonomy. The RoboCup initiative [2], for example, contains at least two competitions that require high levels of human-robot interaction (HRI), namely the RoboCup Rescue League and the RoboCup @ Home League [3]. In both cases, especially in the @ Home League it is important to provide the robots with the capabilities of understanding human interactions in different levels; from the most simple instructions to more complicated communications.

Human communications are complex and may be multimodal [1], i.e., when a human says something, she could also use gestures at the same time. These gestures may be meaningful, superfluous or redundant. As an example, a human may issue a spoken command of: "hand me those keys" and at the same time hand pointing to the key's physical location. Other example is a human issuing the order: "come here" and gesturing such order with her hands. In the former case, the instruction requires both data types; the spoken command and the visual gesture to execute the expected action. In the latter, only one of them is necessary, as the other is redundant. However, the usage of both sources of communication may help the robot to assert that the given instruction was clearly understood. Furthermore, having access to this redundant information, the interaction system may be more robust to noise in one data source. For instance, imagine that the robot is made to interact within a place with controlled light conditions but with high levels of auditive noise. In such scenario, the spoken issue may be polluted with noise, while the gesture may be easier to detect.

In this paper, we show the proposal and development of a methodology based on statistical learning that builds classifiers for the multimodal human-robot interaction problem. Our proposal uses kernel-based classifiers, specifying one kernel per data type that aims at exploiting the data type similarity and combining them into a single classifier. We use One-class classifiers in order to capture the probability distribution of each instruction avoiding the problem of retraining when a new instruction requires to be added. The main contributions of this paper are as follows:

- Modeling the detection of multimodal human-robot interaction as a multiclass statistical classification problem.
- Proposal of a suitable scalable architecture for a multiclass classifier based on One-class classifiers.
- Development of combined kernels by using data from different sources, each one with an appropriate kernel.
- Construction of a database of multimodal instructions (audio and video) with several test subjects and feature extraction.
- Validation of the proposed model using real data that shows improved classification when exposed to noise in any of the sources.

This paper starts by showing the work done by other researchers related to the HRI problem using statistical learning and combining input from several sources. Then, we describe the steps of our methodology, from the data acquisition step until the construction of a multiclass classifier using combined kernels. Finally, we show the implementation details of the methodology and its results when compared to two standard statistical learning classifiers and showing the behavior of our proposal in the presence of noise. Finally, we conclude and show future work.

2 Related Work

Many different strategies have been proposed in order to recognize instructions and gestures given by humans over the last years aiming at attaining a more natural communication between humans and robots. In this sense, [4] presents a proposal in which a telepresence robot is capable of heading its attention to places where a human is indirectly pointing by using gestures. They use a Kinect in order to recognize the gestures and head tracking as well as a microphone to achieve voice detection and localization of the audio source. Other proposals apply machine learning techniques to recognize only visual gestures, only voice commands or combined information sources, such as in [4]. In this work, the authors obtain data from a Kinect to recognize gestures using human body joints as movement markers.

On the other hand, in [5, 6] the authors show how to implement a gesture classifier capable of identifying signals and gestures with no meaning and others with well known meaning. This project also uses a Kinect for data acquisition. They consider four sets of gestures, each with 300 examples made by different subjects. In addition, they recorded a data set made of random movements, in order to be included in the validation process to assess the precision of the model. In [7], the authors present an audio classification system that uses SVM and RBFNN. In this work, the authors propose extraction of a number of audio features using Mel-frequency cepstral coefficients (MFCCs). The extraction of related coefficients is fundamental for classification systems as mentioned in [8].

In [9], the authors presented a classification system for five categories analyzing visual features. These features are the base of the works presented in [9], and is of high importance in the field of classification of directions given by gestures and audio. The authors propose a combination of data from gestures and audio to classify content in five categories (news, advertisement, sport, serial and movie). As the other mentioned works, they employ MFCC coefficients to extract audio features, however, for gestures they used color histograms to video segmentation. With the compiling all these features, they assemble a classifier using Support Vector Machines allowing them to build a classifier to every category for audio data and also for video data and through a combination method of weighted sum of audio and video, obtain a category for the processed information.

3 Proposed Methodology

We have designed a methodology that aims at solving the problem of multimodal human-robot interaction, initially for three instructions given by the human, namely: up, down and unknown. The human shall provide an instruction composed of a gesture (body movements) and a spoken word that the robot should be able to identify and execute. In the following sections we show a brief description of each step in the proposed methodology.

3.1 Data Acquisition

The first step in the proposed methodology consists on the construction of a database that contains examples of humans performing gestures and spoken instructions for the "Up" and "Down" classes. For the class "Up", the oral instruction is the verbal spanish word "arriba", while for the class "Down" the instruction is the word "abajo". The gestures for each class correspond to the act of moving both hands upward, for the class "Up" and downward for the class "Down".

Overall, we performed the experiment with 21 subjects who executed 20 repetitions for each instruction for a total of 420 observations for each class. From this dataset, the 90 % is used in the training phase and the remaining 10 % is used to validate the performance of each classifier. Additionally, the validation dataset also includes 80 samples of instructions that do not correspond to either class. Note that these 80 additional observations are not used in the training process of the classifiers, but only as a validation set to assess the performance of the various multiclass classifiers shown henceforth. On one hand, for the "audio" dataset that contains the oral instructions, we acquired the raw data using a microphone. On the other hand, the "video" dataset that corresponds to the gestures performed by the humans for both classes was taken using the Kinect.

3.2 Feature Extraction

3.2.1 Video

Using the Kinect, we have captured a set of frames taken at 30 fps per observation. At each frame, the Kinect provides spatial coordinates of the human skeleton joints for each subject, while the subject performed gestures. In this context, one observation corresponds to the execution of the complete gesture performed by the subject from the start of the data acquisition, following the skeleton tracking until no more movements are detected. For computational tractability and also to avoid the curse of dimensionality, we have only included the information corresponding to the tip of both hands. Figure 1 shows an example of a "Up" gesture performed by one subject and the skeleton detection executed by the Kinect.

The chosen set of features for each observation is composed of the following values:

- (x, y) coordinates of the final and initial positions of the gesture for the tip of the hands
- Mean velocity of the complete gesture for each hand
- Angle of the vector that points from the initial position to the final position for each hand

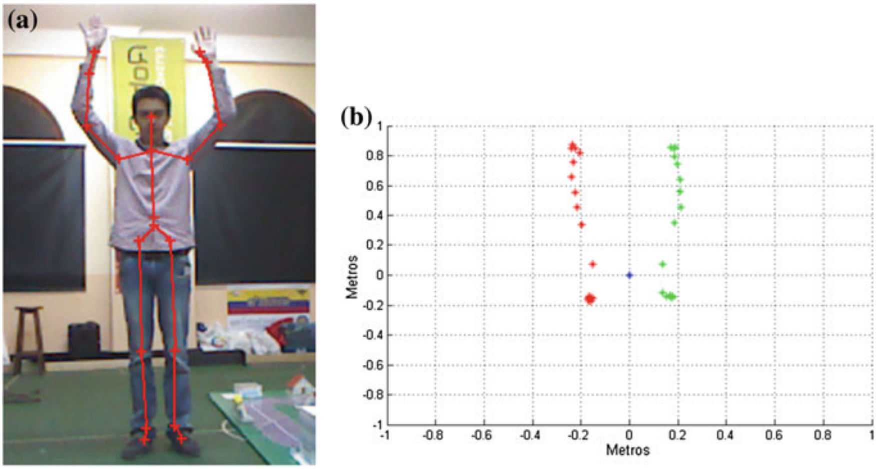


Fig. 1 **a** Example of Kinect skeleton tracking at the end of the "up" gesture for one subject. **b** Example of coordinate trajectory for the tip of both hands when performing the "up" gesture

3.2.2 Audio

We have characterized the audio dataset using the Mel Frequency Cepstral Coefficients (MFCC). According to [8], the first 13 initial coefficients are sufficient to properly represent the original signal without losing important information. The procedure consists on dividing the entire audio signal in time windows of 15ms each where the Fourier coefficients are computed. Then, we compute an estimation of the power spectral density by using triangular overlapping windows, then, the we take the log of the power at each Mel frequency and finally take the Discrete Cosine Transform. The amplitude of such spectrum are the MFCCs. Each coefficient is then averaged over the time windows obtaining 13 coefficients that describe the complete signal.

4 SVM Classifier with Combined Kernels

This section contains the different stages followed to build an appropriate SVM-multiclass classifier using combined kernels. First, Sect. 4.1 shows the methodology we used to build multiclass classifiers based on multiple one-class classifiers. Afterwards, we evaluate the performance of using these classifiers with three different kernels for each type of data, i.e., audio and video. Subsequently, we show how to build a new classifier that uses a kernel that is the result of the combination of the two kernels that have shown improved performance for each data type. Finally, the result is compared and validated with multiclass classifiers that use a single kernel.

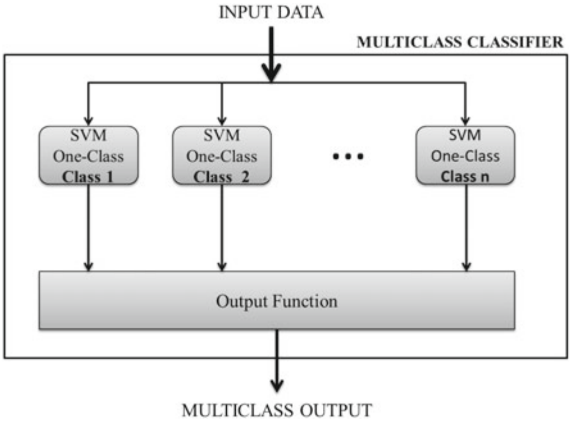
4.1 Multiclass Classifier Based on SVM One-Class Classifiers

The multiclass classifiers built for the task at hand were based on multiple SVM one-class classifiers. The reason to follow this approach, in contrast with building common multiclass classifiers, is mainly inspired on the specific problem that we are tackling, and is twofold:

- In the context of human-robot interaction, the potential amount of comands issued by a human that a robot would be required to identify may be high, which may cause the number of classes in a common multiclass problem to grow rapidly. This would require a potentially high amount of samples per class. Besides, every time a new class were added (i.e., a new instruction were included in the human-robot communication), the multiclass problem would require retraining, which is, in general, a hard process.
- In addition to the set of instructions that the robot should be able to identify and execute, we should also consider the fact that there is a need of an additional class where the robot identifies a given instruction as “invalid”. This class encompasses all instructions for which the robot should perform no action. It is not feasible to collect data that represent the underlying distribution of such class since it contains all actions (potentially infinite) but the ones identifiable by the robot.

For these reasons, we proposed to build multiclass classifiers based on SVM one-class classifiers. Figure 2 shows our implementation of a multiclass classifier based solely on one-class classifiers. In this architecture, the complete dataset is used as input to each one-class classifier. Each individual classifier outputs its own hypothesis recognizing only the portion of the data that belongs to such class (i.e., each classifier is capable of identify data points that belong to one single class) which then needs to be combined into an output function that generates a single multiclass output. For the particular case, one observation belongs to the “invalid” class

Fig. 2 Multiclass classifier based on one class SVM



if any of the two following situations is true: (**a**) more than one classifier claims the observation belongs to its own class or (**b**) all classifiers label the observation as not belonging to their own class (i.e., the instructions is not recognized by any classifier). In situations where only one classifier recognize the observation as belonging to its class (and everyone else rejected it), the observation is assigned the label of the class that correspond to the one-class classifier that recognized it as part of its probability distribution. All the SVM multiclass classifiers built henceforth used this approach of combining several one-class classifiers.

4.2 Kernel Selection

The selection of the kernel is a highly relevant process and is at the center of the methodology proposed here. The idea is that each data source (i.e., audio and video) will be better represented using a kernel that exploit its similarities. This means that potentially the use of an appropriate kernel in a specific type of data will attained improved performance over other general kernels. In our methodology, we require an identification of the kernel for each type of data that captures the most representative similarity for such type of data. In other applications, special kernels may be constructed for specific types of data.

For this selection step we have split our dataset in two disjoint sets, one containing the audio data and other with the video data, and built a multiclass classifier for each using the polynomial, gaussian and sigmoidal kernels. The result of this process is one classifier for each data type for each kernel. Then, each classifier is evaluated using the error on the validation data and finally keeping the kernel for each data type that delivers the minimum validation error.

4.3 Construction of the Combined Kernel

The next step consists on finding a new kernel by combining the kernels that showed improved performance for each data type. According to [11], it is possible to combine several kernels by means of a linear combination of the individual kernels. The rationale behind this idea is that different kernels could be using inputs from representations of different sources. For this reason, they have different measures of similarity which correspond to different kernels. This procedure allows us to combine multiple information sources as required in HRI applications. Equation (1) shows the calculation of the combined kernel as a linear combination of the audio and video kernels. The coefficients of the linear combination requires a careful tuning process.

$$\kappa = \omega_a \kappa_a + \omega_v \kappa_v \tag{1}$$

5 Implementation and Results

We have used the LibSVM library [14] for the implementation of the SVM one-class classifiers using the precomputed kernel option. For the construction of the multiclass classifiers, we have performed cross-validation in which the entire dataset is split in two disjoint sets: one for the training process and one for validation. The performance of each classifier is measured as the accuracy attained in the validation set after the training process has been completed using only training data.

5.1 Kernel Selection

The performance of each SVM One-class classifiers (and hence that of the multiclass classifier) using each kernel heavily depends on a proper tuning of the kernel parameters. For this task, we have performed a grid search over the kernel parameters space for each classifier together with the cross validation procedure in order to asses the performance of each kernel combination in the classification process. Table 1 shows the generalization errors for each One-class classifier (Up and Down), for each data type (audio and video) and each kernel and the error obtained when building the multiclass classifier based on the One-class classifiers.

By using this experiment, we have shown that the kernel that provides the best representation for the audio type of data is the **Sigmoidal kernel** (with a generalization error of 28.05 %), while that for the video data is the **Gaussian kernel** (with a generalization error of 5.49 %). These kernels will be used for each data type in the kernel combination process below.

Table 1 Generalization errors of each One-class classifier for each kernel and for each type of data and generalization errors of the multiclass classifier using the three chosen kernels for each type of data

	Audio			Video		
	Polynomial (%)	Gaussian (%)	Sigmoidal (%)	Polynomial (%)	Gaussian (%)	Sigmoidal (%)
Up	21.34	16.46	19.51	3.05	2.44	2.44
Down	16.46	17.07	9.76	5.49	3.05	4.27
Multiclass	36.58	29.87	**28.05**	8.53	**5.49**	6.71

5.2 Construction of the Multiclass SVM Using Combined Kernel

Our proposed classifier is one multiclass SVM with the following characteristics:

- It is composed of several One-class SVM classifiers; one per each instruction. In our case study, we have two classifiers; one for the "up" class and other for the "down" class.
- Each One-class classifier is constructed using a combined kernel classifier using the sigmoidal kernel for the audio data and the gaussian kernel for the video data.

Overall, we solved two One-class combined kernel classifiers; one for each class which means that each SVM problem contained five parameters that needed to be tuned, namely: the regularization parameter ν, the linear combination coefficients ω_a and ω_v and the kernel parameters for the sigmoidal and gaussian kernel γ_s and γ_g respectively. The procedure followed to tune such parameters was cross validation as described before.

After the tuning process, our multiclasss SVM classifier that uses combined kernels attained a generalization error of 3.66 % showing that it is possible to classify two human instructions using two different data sources with low generalization errors. In the following section we implement two standard multiclass classifiers and compared their results with ours.

5.3 Validation

In order to validate the results of our proposal, we have constructed two standard multiclass classifiers using Support Vector Machines and Neural Networks (NN) with the training dataset. On one hand, for the NN, we have chosen a multilayer perceptron (MLP) architecture with one hidden layer using sigmoidal activation functions. The only parameter that needs to be decided is the number of neurons in the hidden layer. In the case of the SVMs, we need to decide which kernel will be used and the values of the kernel parameters as well as the γ value.

The NN that presented the lowest error was one that uses 6 neurons in its hidden layer while the SVM with lowest generalization error was the one using sigmoidal kernel. These two classifiers will be used later to compare the performance of the classifier that combines kernels. The neural network with one hidden layer with 6 neurons attained 12.01 % of generalization error while a multiclass SVM using one sigmoidal kernel attained 10.37 %. These results show that the classifier built by using combined kernels showed improved performance when compared to two standard classifiers.

5.4 Analysis of Noise in Generalization

We have performed one additional experiment that aims at exposing our methodology to situations where the dataset is in the presence of noise. For this, we have included additive gaussian noise to the raw data, i.e., to the audio file and to the skeleton structure from the Kinect and performed three different tests, namely:

- Noise is added only to the audio dataset with different variance levels $\sigma \in [0\ 0.1]$
- Noise is added only to the video dataset with different variance levels $\sigma \in [0\ 0.05]$
- Noise is added to both, audio and video dataset with different variance levels.

Figure 3a, b, c show the results of such experiments. It is important to notice that the SVM classifier that uses combined kernels has lower generalization errors for the different noise variances in the audio dataset, showing that such classifier is more resilient to noise in the audio channel. For the noise in the video dataset we can see that the NN and the SVM with combined kernels achieve similar results. However, when the noise is presented in the complete dataset, the SVM that uses combined kernel, shows improved performance. This result shows that it is possible to build robust classifiers by exploiting each data type similarities by choosing appropriate kernels and then combining them into one single kernel.

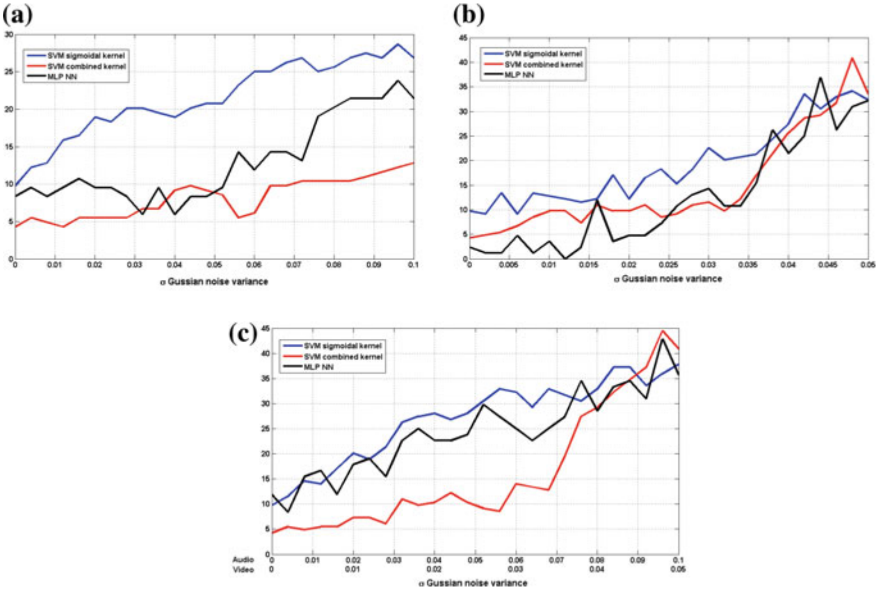


Fig. 3 **a** Generalization error of the three classifiers when exposed to noise in the audio dataset. **b** Generalization error of the three classifiers when exposed to noise in the video dataset. **c** Generalization error of the three classifiers when exposed to noise in the audio and the video dataset

6 Conclusions

In this work we have shown a methodology designed to build classifiers capable of identifying between several human instructions given using oral commands and gestures. The methodology is based on statistical learning and uses support vector machines with special kernels per data type that are combined into one single kernel by means of a linear combination. The classifier supports the addition of new instructions without the need to retraining. This is obtained by building classifiers based on several SVM One-class classifiers, one per each class. This methodology also solves the problem of acquiring data for unknown instructions for which it may be unfeasible to collect representable data. Finally, we have shown experimentally that our proposal attain improve performance over standard classifiers by using combined kernels which have also shown greater robustness when the data are corrupted with noise.

References

1. Goodrich, M., Schultz, A.: Human-robot Interaction: a survey. Found. Trends Hum.-Comput. Interact. **1**, 203–275 (2007)
2. Behnke, S., Veloso, M., Visser, A., Xiong, R. (eds.): RoboCup 2013: Robot World Cup XVII, LNCS. Springer, Berlin (2014)
3. Van Beek, L., Chen, K., Holz, D., Matamoros, M., Rascon, C., Rudinac, M., Ruiz del Solar, J., Sugiura, K., Wachsmuth, S.: RoboCupHome 2015: Rule and Regulations (2015)
4. Bhattacharya, S., Czejdo, B., Perez, N.: Gesture classification with machine learning using kinect sensor data. In: Third International Conference on Emerging Applications of Information Technology, pp. 348–351. IEEE Press, Kolkata (2012)
5. Kinect Gesture Recognition for Interactive System. http://cs229.stanford.edu/proj2012/ZhangDuLiKinectGestureRecognitionforInteractiveSystem.pdf
6. Huang, J., Lee, C., Ma, J.: Gesture Recognition and Classification using the Microsoft Kinect. Final Project CS229 Machine Learning. Stanford University, Stanford (2012)
7. Dhanalakshimi, P., Palanivel, S., Ramaligam, V.: Classification of audio signals using SVM and RBFNN. Expert Syst. Appl. **36**, 6069–6075 (2009)
8. Sarachaga, G., Sartori, V., Vignoli, L.: Identificacin Automtica de Resumen en Canciones. Proyecto de fin de carrera, Universidad de la Repblica, Uruguay (2006)
9. Suresh, V., Mohan, C., Kumaraswamy, R., Yegnanarayana, B.: Content-based video classification using support vector machines. In: Pal, N.R., et al. (eds.) ICONIP 2004. LNCS, vol. 3316, pp. 726–731. Springer, Heidelberg (2004)
10. Subashini, K., Palanivel, S., Ramalingam, V.: Audio-Video based classification using SVM and AANN. Int. J. Comput. Appl. **53**(18), 43–49 (2012)
11. Gönen, M.M., Alpaydin, E.: Multiple Kernel Learning Algorithms. J. Mach. Learn. Res. **12**, 2211–2268 (2011)
12. Manevitz, L., Yousef, M.: One-Class SVMs for document classication. J. Mach. Learn. Res. **2**, 139–154 (2001)
13. Scholkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Technical report, Microsoft Research MSR-TR-99-87 (1999)
14. Chang, C.C., Lin, C.J.: LIBSVM : a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 27:1–27:27 (2011)