The Grenoble System for the Social Touch Challenge at ICMI 2015

Viet Cuong Ta, Wafa Johal, Maxime Portaz, Eric Castelli, Dominique Vaufreydaz

The Grenoble System for the Social Touch Challenge at ICMI 2015

Viet-Cuong Ta[1,2,3], Wafa Johal[4,5], Maxime Portaz[2,3,5],
Eric Castelli[1], Dominique Vaufreydaz[2,3,5]

[1]International Research Institute MICA, HUST-CNRS/UMI 2954-Grenoble INP

[2]Prima/LIG, CNRS

[3]Inria

[4]MAGMA/LIG, CNRS

[5]University of Grenoble-Alpes

October 12, 2015

Abstract

New technologies and especially robotics is going towards more natural user interfaces. Works have been done in different modality of interaction such as sight (visual computing), and audio (speech and audio recognition) but some other modalities are still less researched. The touch modality is one of the less studied in HRI but could be valuable for naturalistic interaction. However touch signals can vary in semantics. It is therefore necessary to be able to recognize touch gestures in order to make human-robot interaction even more natural. We propose a method to recognize touch gestures. This method was developed on the CoST corpus and then directly applied on the HAART dataset as a participation of the Social Touch Challenge at ICMI 2015. Our touch gesture recognition process is detailed in this article to make it reproducible by other research teams. Besides features set description, we manually filtered the training corpus to produce 2 datasets. For the challenge, we submitted 6 different systems. A Support Vector Machine and a Random Forest classifiers for the HAART dataset. For the CoST dataset, the same classifiers are tested in two conditions: using all or filtered training datasets. As reported by organizers, our systems have the best correct rate in this year's challenge (70.91% on HAART, 61.34% on CoST). Our performances are slightly better that other participants but stay under previous reported state-of-the-art results.

Keywords

Multimodal perception, gesture recognition, touch challenge

1 Introduction

Up to now, research in natural human-robot interaction has mainly focused on visual and audio modalities. However, few studies have given attention to the touch modality of interaction. Some well known robots, such as Aibo (1999), Paro (2001), Nao (2006) or Reeti (2011) are equipped with touch sensors. Some researchers have investigated skin-like sensing, i.e. lot of sensors spread all over the robot body [1, 2, 3]. For a pet robot, with fur, such as Paro, it would be interesting to be equipped with a sensors matrix and to be able to recognize the gestures. For cobots, touching information may increase co-working performance.

Some other approaches tackle squared grid pressure sensors to sense socials gestures from touching information. The CoST [4] and the HAART [5] corpora were recorded in order to study these gestures. This paper focuses on touch gestures recognition and provides new features and recognition systems to enable recognizing up to 14 gestures from the CoST and 7 gestures from the HAART database. The next section develops on the state-of-the art of social touch sensing, and introduces the two dataset we have worked on.

We then introduce our recognition method that was presented for participation to the Social Touch Challenge at ICMI2015[1]. Finally, we show our results and discuss them.

2 Related Works

2.1 Research Context

Touch gestures recognitions is a trending research topic with recent patent such as Google's on grid of touch sensors [6] on flat displays. As recently showed by Cooney et al. [7], touch is one of the main channels of communication of affection in human-robot interaction (HRI). These results suggest more focus of the research in recognizing affective touch gestures and improve human-robot touch communication. In a survey [8], the authors propose to categorize what they call Tactile HRI according to the covers of the sensor and according to king of physical interaction aiming to interfere with the robot's operation (i.e. grab to stop the movement for instance), to be part of the interaction as a way to communicate (i.e. pat to signify satisfaction) or to learn from tactile inputs (i.e. reinforce behaviors by a pat). In this framework, more and more researches focus on recognizing social touch for the robot to react to the user. Sometimes, sensors are spread on a robot to recognize social touch [3].

2.2 Touch Challenge

2.2.1 The CoST dataset

The Corpus of Social Touch (CoST) dataset was introduced in [4]. It is a collection of 14 different social gestures performed under three variations: gentle, normal and rough. The data collection has been made via a sensor grid place on a mannequin arm. 31 participants participated to the data acquisition. The collected data consist in an 8x8 grid of pressure intensity at 135 fps. For the first recognition of the 14 gestures of the CoST, authors [4] chose to compute 28 features. They used Gaussian Bayesian and SVM classifiers to constitute the baseline of recognition of gestures form the 8x8 sensors data. The accuracy had an overall maximum of 54% with the Bayesian classifiers. This paper reported also the difficultly to recognize some gestures from others like the gentle gestures. The curvature of the arm on which the sensors are placed could also play a role in the lower recognition rate.

2.2.2 The HAART dataset

The aim for recording the HAART corpus [5] was to be able to sense emotion through gestures. Authors collected data from 16 participants at 50fps. They choose 9 gestures for the participants to perform and recorded each gesture for 2 seconds. When trained on all participants the recognition has an accuracy of 86% and when trained on individuals the authors' recognition system reaches 94%. The reader must note that for the challenge, in order to be consistent with the CoST corpus, the 10x10 pressure sensor data has been trimmed to the 8x8 central pressure cells. Thus, for the HAART task, we are not strictly in condition reported by previous researches [2] as they used more pressure cells (100 versus 64).

3 Methodology

The team involved in the touch challenge for the Grenoble system is multidisciplinary. People have background in several audio and video signal processing (speech recognition, visual tracking, multimodal perception...), Human Robot Interaction and indoor localization. Interest for such a group is to try alternative approaches for the challenge coming from multiple scientific communities.

The first step of our methodology was to decide between HAART and CoST tasks which is the more difficult one. Factually, the CoST corpus has twice more classes. Preliminary simple classification experiments confirmed that the same system performs better on the HAART task. For the challenge, we concentrated our efforts on building the best classification system for the CoST task. Then, we applied recipes directly on the HAART corpus.

[1] http://www.utwente.nl/ewi/touch-challenge-2015/

Next steps of our methodology are detailed in the following sections. Results using Dynamic Time Warping DTW are reported in 3.3.1. Submitted challenge results using Support Vector Machine and Random Forest are detailed in section 4.

3.1 Looking deeply at the data

Our analysis of the HAART and CoST corpora started with a deep look at the data. Several information where extracted. We computed usual statistics *i.e.* *min*, *max*, *mean*, *median*, and *standard deviation* of each cell of the pressure sensor for each corpus and for each class. No real differences between *min* and *max* values were noticeable for both corpora. The distribution for each pressure sensor value for CoST gestures is quite uniformly distributed on $[0; 1023]$ (see fig. 1) (99% of values are in the $[0, 770]$ interval). HAART statistics (*mean=7.20*, *median=3*, *std-dev=16.15*) significantly differ from CoST ones (*mean=159.62*, *median=106*, *std-dev=159.37*). Mean and standard deviation from the HAART corpus are lower. 99% of HAART pressure values are in $[0, 75]$. Figure 1 shows some peculiarities of the CoST data. Distribution of pressure values presents a periodic variation $p{=}{\sim}30$: all values do not have the same probability. This result is the same for the 64 sensors. One hypothesis could be that internal quantification algorithm of each pressure sensor is responsible but we cannot conclude definitely. This statistical analysis showed that HAART and CoST corpora have notable disparities.

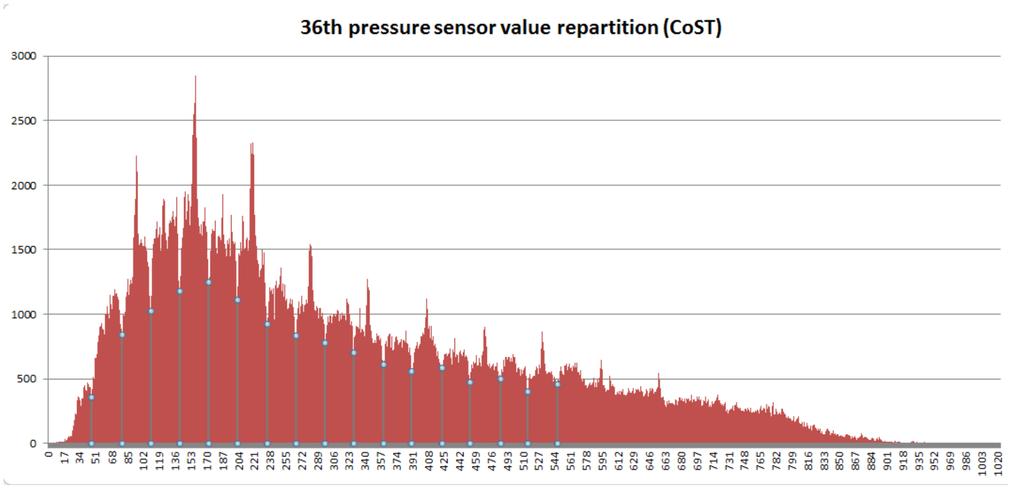

Figure 1: Distribution of pressure values for of the 36th sensor (central) of the CoST corpus. Blue vertical lines highlight periodic variation of the values.

In order to accurately understand what the gestures are, we used several visual representations (see fig. 2). The 2D visualization represents power of each cell of the 8x8 sensor for a frame. Concatenating then, we built gesture videos. These videos showed that external sensors are less frequently activated. Some visual patterns seem to appear on the 8x8 sensor, humanly recognizable but not for all gestures. We also looked at temporal representation of all training sequences. There are a lot of variations among participants and gestures. For a same touch gesture, intra-participant variations are less important than inter-participant variations. One can find examples of these variations for a simple *roughgrab* on figure 2 and 3 of a previous article about the CoST corpus [4]. We can only make the assumption that even with a specific request, people have different interpretation of a gesture. This hypothesis was consolidated by results presented in previous study on the HAART corpus [5]. Using gestures, the person recognition accuracy was at least equal to 79%. On some gestures (*contact*, *pat*, *tickle*), the accuracy was over 90%.

We learned a lot analyzing data from both corpora. We took benefits from this knowledge to design our classification system.

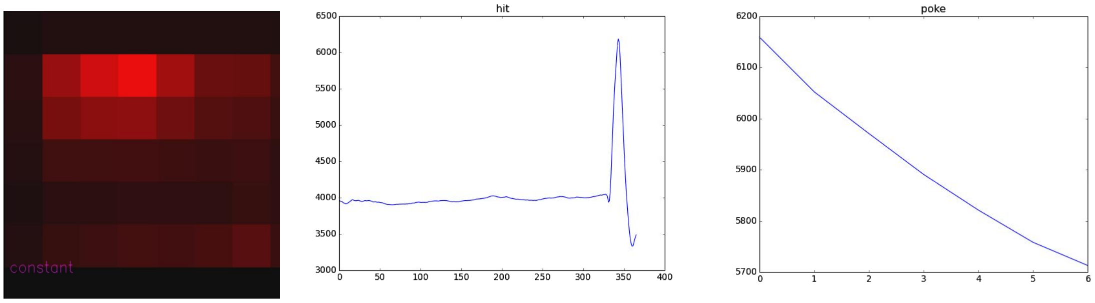

Figure 2: Visualization of data. At left, 2D visualization of a 8x8 sensor for one frame (*constant* example, max value in red). The middle and right figures are temporal representation of respectively *hit* and *poke* examples.

3.2 Filtering training subsets

An important problem emerged when looking at temporal representations of gestures: bad segmentation for some gestures in the training data. On figure 2, the central example depicted one too long *hit* gesture. The first 320 frames, more than 2.3 seconds[2], do not belong to a *hit* touch and should have been removed. On the *poke* example, the sequence is contrarily too short and does not contain actual information about the gesture.

Armed with this experience, we decided to select data for the training of our system in order to increase performance of our classification system. We tried to remove automatically abnormal training sequences. As seen, these gestures were mainly too long or too short sequences. We used statistics on frames, filtering outlier gestures within classes. Several thresholds on length of sequences, median, mean, or maximal pressure values, and selection percentage were unsuccessfully tested. There was no evident automatic filter that significantly increases performance of the classification system at this time.

With the certitude that removing outliers can lead to performance enhancement, we backed down to a manual selection. We did a two pass selection on the data, made by one researcher of our challenge group. The selection criteria were not only based on length of sequences, but also on shapes. For instance, some *pat* gestures looks more like *tap* or *grab*. From the CoST corpus, we removed 164 gestures (∼4.5%). We refer to this new training corpus as "filtered" or "clean" later in this article.

As it is a time consuming task, the training selection was only done on the CoST training corpus. The new filtered trained corpus can be retrieve asking to the authors. In section 5, we will see that each classification systems on the CoST corpus, we submitted two results: one on *filtered data* and one on *all training data*.

3.3 Experiments

3.3.1 Dynamic Time Warping experiments

Based on our background in speech processing and on literature in gesture recognition, we tried several Dynamic Time Warping (DTW) approaches. Dynamic Time Warping is used for many time sequence matching problems like speech or gesture recognition [9, 10, 11, 12]. In a similar context to this challenge, DTW have been used for emotion recognition using pressure sensor keyboard [13]. Dynamic Time Warping selects among n models the best candidate, i.e. the model that best matches the input sequence. DTW only needs a feature set and a distance function to process data sequences. Uncountable variants of DTW can be found in literature. Using temporal representation depicted in section 3.1, we implemented several DTW flavors based on models selection or optimization [10], or multidimensionnal DTW (MD-DTW) [14, 12, 15]. MD-DTW does not perform well on the Cost task (∼40% of accuracy). Model selection or optimization strategy can elicit better models. As presented in [11], our selection strategy chooses normalized models that minimize intra-class and maximize inter-class distances: 90% of original models are kept. The best score using models selection is ∼47%.

[2] The frame rate of the CoST corpus is 135Hz.

None of these global approaches provide more than 50% of accuracy. As we said before, outside pressure cells are less activated than center ones. Not all lines or columns of the 8x8 sensor have the same temporal shape. We did experiments using DTW per lines, per columns and per both axis. A majority vote provides the most detected class, or if there is an ambiguity, the class with the lowest distance score. Finally, we managed to almost equals original results previously reported [4] on the CoST corpus (our best score=52%) using DTW per lines and per columns with model selection.

3.3.2 Machine learning approach

As DTW did not perform as expected, we needed to choose among machine learning algorithms. They are many candidates that cannot all be tested. The choice of the underlying classification algorithms has been driven mainly by our expertise and previous results on the corpora. Support Vector Machine (SVM) has demonstrated performance in previous tasks [16]. Random Forest (RF) proved their adequacy on the HAART corpus [5]. The later sections of this article will detail experiments and results using SVM and RF.

4 Features set

This section details our feature list, the feature selection method and the evaluation of the final feature set.

4.1 Features Extraction

From our research background, previous works [4, 5, 11, 13, 17] and our analysis of the data, we picked a list of features to cater gesture recognition. For easy tracking of the number of features and the way it affects the classification capacity of the model, we divided the features into three groups: *global* features, *channel-based* features, *sequence* of average pressure features.

4.1.1 Global features (Group 1)

The 40 features in this group represent overall statistics of the gestures. Some of them are adapted from [4], some came from our deep look at the representation of the gestures (see 3.1):

- Number of frames (i.e. length of the gesture);
- Average pressure on the 64 channels over all frames;
- Maximum value of pressure of all channels over all frames;
- Percentage *no signal* frames. A frame is considered as *no signal* if its average value over all 64 channels is below a threshold t. We set $t=50$ for both datasets;
- Mean pressure over all frames of each column;
- Mean pressure over all frames of each row;
- Variations around a specific value c. *Zero-Crossing Rate* (ZCR) is a time-domain feature still widely used in many speech applications [18]. It is measures variations around the central value (zero in case of speech) of the temporal signal and provides clues information about speech activity. Inspired by this feature and our experiments using DTW, we implemented what can be called *c-crossing.* It considers the 64 channels as 1-dimensional temporal signal by taking the mean value over 64 channels for each frame. This group of feature includes the number of crossing around c, the percentage of the sequence above c, the average length of sequence above c, and the average slope of each crossing. As it was impossible to fixed c for all gestures, we took 5 different values for c: 35th, 50th, 65th, 80th and 95th-percentile of the maximal value.

4.1.2 Channel-based features (Group 2)

Channel-based features are extracted on each channel. Basically, they ignore the temporal relationship of consecutive frames and focus on the spatial relationship between different channels. There are a total of 192 features in this group:

- Average pressure of each channel over all frames;
- Average pressure variation of each channel over all frames. First, it takes the absolute difference of each consecutive pair of frames. Then, we add the mean of all the differences through the time per channel;
- Percentage of time when a channel has pressure more than a fixed threshold T: For each channel, we compute the number of times where it reaches above T. It is then divided by the number of frames to get the percentage value. These features reflect the strong active regions of the channels. In our experiments, T is set relatively as the 90-percentile of all the value in the gesture.

4.1.3 Sequence features (Group 3)

In this group of features, we want to extract the changes over time of the gestures. These changes should be captured on each individual channel. However, it will drastically increase the size of the feature set. We used the sequence of average pressure over 64-channel for each frame. There are a total of 41 features in this group:

- Fast Fourier transform based features: these features are the results of applying the Fast Fourier operator [19] to the sequence of average pressure. The operator extracts the first 16 highest frequencies values, taking only the real part.
- Discrete cosine transform based features: these features are the results when apply the discrete cosine transform [20] to the sequence of average pressure over 64 channels. We used the discrete cosine transform type 2. The operator extracts the first 25 value. In the case where the length of the gesture is less than 25 frames, the missing values are set to 0. This will set each feature vector have an equal length.

The total set of 273 features is extracted on the original frames. On each individual frame, we applied a Sobel operator to extract local variation of the pressure [21]. We get a new sequence of Sobel frames for each gesture. As this new sequence has the same structure (size, number of channel per frame) of the original one, we can use the same process to extract another 273 features. Our final feature set contains 546 features in total.

4.2 Feature selection

From the process above, there may have noisy, under-correlated or redundant features. For example, the outside channels may contain noise from the extraction process rather than providing valuable information. Hence, we performed feature selection to improve the performance of the system. From 546 feature vector above, we ran a Random Forest model (from the Scikit-learn library [22]), then selecting the 30% highest weighted features with a 3-fold cross-validation on training data (see more details on cross-validation in 4.3.2). The choice of a 3-fold was made to provide a similar relative proportion of the training and test sets in the competition. After this selection step, we got a 164 feature set.

In previous researches in multimodal perception [16], we successfully applied a feature reduction technique from bioinformatics research domain. The Minimum Redundancy Maximum Relevance [23] (MRMR) has the advantage of giving the more relevant features instead of building new features from the observed ones. On our computers, we cannot compute a global selection, but an in-group partial selection. The reduced feature set using MRMR did not improve significantly our results, we kept the previous selection.

Features	Acc. on Train	Acc. on Test
Group 1	58.22%	52.17%
Group 1 + 2	66.56%	56.81%
Group 1 + 2 + 3	66.89%	58.96%
Adding Sobel Frame	68.61%	60.10%
164 features	69.77%	61.34%

Table 1: Incremental feature set performance on the CoST dataset.

4.3 Feature evaluation

As said, the same feature selection scheme can be applied on the CoST and HAART datasets. For time constraint reasons, we choose to rely on only the CoST dataset to estimate the effectiveness of the feature set.

4.3.1 Auto-validation

One preliminary validation was to check whatever we have, among selected features, enough information about the data. The first test was to train a system and to test it against the whole training dataset. This test is not dedicated to check performance but to acknowledge our previous choices and our algorithm implementation. For SVM and RF systems, using the 164 feature set, auto-validation provides 100% of accuracy. With this result, we can go ahead and use our system on the challenge test data.

4.3.2 Cross-validation

Cross-validation is used to validate performance of a system against a corpus. We tried a *k-fold* cross validation on the filtered CoST corpus. In this method, the dataset is partitioned in k subsets. One subset is kept for testing and the *k-1* others are used for training the model. This splitting process is repeated k times so that each subset is used once for testing against others subsets. Interest of this method resides in two aspects: a clear separation of training and testing parts for evaluation; a validation of all data, subset by subset, to check performance in regards of data distribution within the corpus. As gestures are very user-dependent [5], we implemented a specific *k-fold*. Our folding strategy does not split using gestures and/or variants information but using user ids. Thus, it is impossible to have gestures from the same user both in train and test corpora. Our *3-fold* result is presented on last line of second column of table 1. The best cross-validation score on the CoST task is 69.77%. At this time, it seemed a good improvement from previous results [4, 17].

4.3.3 Contribution of each feature group

To verify the previous score, we ran a Random Forest model with cross validation on the clean training set. We added features group by group to check whether the recognition rate increase or not. After the challenge, with the test data labels, we did the experiment again on the test data. The results are presented in the table 1.

The experiment started with only group 1 as the base features. We kept expanding the size of feature vectors by adding features of group 2, then features of group 3. At last, we tried 164 feature set. By looking at the *k-fold* testing on the training data, each group generates a clear improvement to the starting one. We can conclude that valuable features are present in all groups. The selection method is useful as it proposes an absolute ∼1% accuracy improvement. One can note that there is a high correlation between accuracy in both testing scheme but there is a clear drop of accuracy on testing data.

Additional information can be found on figures 3 and 4. The first one gives additional statistical properties, notably distribution of each feature group, in the 164 feature set. The second provides importance of each channel to classify the gesture types in term of selected channel features. As we expected, the channels around the center contain the highest amount of information.

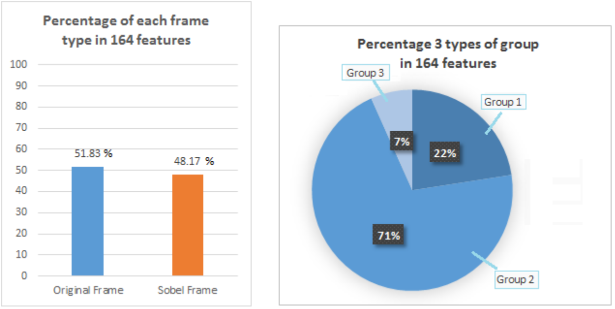

Figure 3: Contribution of each feature group to the final feature set.

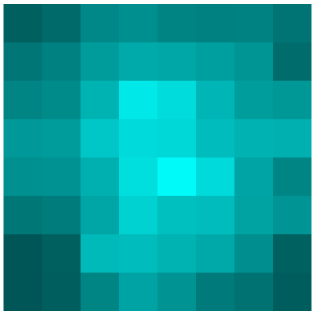

Figure 4: Importance of the 64 channels. The highest one are colored with the brightest color.

5 Experimental Results

Random Forest were used for their ability to handle large sets of features and to include irrelevant features. Support Vector Machine (SVM) were used as a baseline for comparison with previous results on classifying gestures on the CoST dataset [4]. SVM used *rbf* kernels and standard parameters from the Scikit-learn library. Results using RF and SVM are depicted in this section. In all tables, RF results have a light gray background.

5.1 Actual results for the challenge

After release of the test data, we ran our classification systems on both corpora. For the HAART corpus, we applied directly our SVM and RF systems. For CoST, we used two versions of our systems: one using all training data, one trained on the manually filtered version (see 3.2).

Results on the CoST task for the all flavors of our systems can be found in table 2. At first look, these results confirmed our hypothesis: filtering training data can be of interest. Either for SVM or RF, the classification result is better than using the whole training set. In this table, the best result (61.34%) is a Random Forest system trained on the filtered training set. We cannot definitely conclude that RF is better that SVM. Indeed, feature selection was done using a RF paradigm. This may explain why RF performs better using this final feature set. The matrices (tables 5 and 6) show confusions among our results. There is no outstanding difference in these results between SVM and RF systems, in both training scheme. The list of main confusion groups is (*squeeze*, *grab*), (*tap*, *pat*, *slap*, *hit*), (*rub*, *stroke*) and (*scratch*, *tickle*). For all conditions, the worst performance classification concerns *rub*, *pat*, *scratch* and *squeeze* ($\leq$ 50%). *Rub* is the more difficult gesture to recognize. The best recognize score is for *poke*, followed by *stroke*, *tickle* and *massage*. These results are coherent with previous reported confusion matrices [4] even if the distribution is strictly not the same.

For the HAART task, one can see results in table 3. Again, the RF performs slightly better than SVM. Looking at confusion matrix (see table 4), contrarily to the previous results, there are differences between the SVM and RF classification system. Indeed, even if the global scores are comparable (*SVM=68.52%*, *RF=70.91%*), the inner values for classes notably vary. There is an absolute improvement for the *constant* and the *rub* classes, respectively +34.29% and +30.56% of accuracy. At the same time, the correct rate of some other classes decreases: *pat* −11.11%, *scratch* −25.00% and *tickle* −8.33%. *notouch* and *stoke* remain almost stable. At the writing time of this article, we have no hypothesis to explain these results. We will need more experiments to make assumptions.

5.2 Discussion

Gesture classification does not present the same accuracy for all gestures. Even if auto-validation provides perfect results, it seems that information is missing to fully differentiate the gestures. Our confusion matrices are coherent with the literature. This problem is thus the same for all reported systems. We can place our classification results within the context of the challenge looking at tables 7 and 8. Performances of our

	CoST task	
Training data	manually selected subset	all
SVM	59.91%	60.51%
Random Forest	**61.34%**	60.81%

Table 2: Overall results on the CoST task.

	HAART task
SVM	68.52%
Random Forest	**70.91%**

Table 3: Overall results on the HAART task.

	constant		notouch		pat		rub		scratch		stroke		tickle	
constant	**22**	**34**	0	1	0	0	0	0	0	0	0	0	0	0
notouch	1	1	**36**	**35**	0	0	0	0	0	0	0	0	0	0
pat	0	0	0	0	**30**	**26**	0	1	0	0	1	1	0	0
rub	0	0	0	0	0	4	**15**	**28**	2	13	1	4	1	5
scratch	12	0	0	0	1	0	17	3	**27**	**18**	3	0	24	21
stroke	0	0	0	0	5	5	3	3	1	0	**31**	**31**	0	4
tickle	0	0	0	0	0	1	1	1	6	5	0	0	**11**	**6**

Table 4: Confusion matrix for the HAART task. In white, SVM results (correct rate 68.52%). In light gray, Random Forest results (correct rate 70.91%).

classifiers are slightly better than systems proposed by the other participants. Our best scores for both corpora remain below the state-of-the-art results. Nevertheless, the previous results on the HAART corpus [5] are not strictly in the condition of the challenge. The 86% of accuracy is obtained using a 100-fold cross-validation paradigm. Moreover, these results are obtained on a 10x10 pressure sensor. Even if we showed (see figures 2 and 4) that outside channels less contribute to the classification, they are not useless. These facts may explain the huge gap (> 15%) between challenge participants and HAART state-of-the-art results. Still, we can compare our system using the available 8x8 pressure data we have and a 100-fold cross-validation. Our score is 76.84%. Difference with state-of-the-art result on the CoST corpus [24] is less important (< 5%). Conditions of this result are similar to the challenge condition.

6 Conclusion

This paper detailed a method to recognize touch gestures on CoST and HAART corpora. We provide new features and classification system to recognize gestures on a grid touch sensor. Our process was developed on the CoST corpus and then directly applied on the HAART task. For the challenge, we submitted several classifiers. A Support Vector Machine and a Random Forest systems for HAART. For the CoST task, the

	grab	hit	massage	pat	pinch	poke	press	rub	scratch	slap	squeeze	stroke	tap	tickle
grab	**83**	0	4	0	4	0	13	2	0	0	52	0	0	0
hit	0	**71**	0	10	0	0	0	0	0	35	0	0	8	0
massage	0	0	**80**	0	0	0	1	9	6	0	2	0	0	6
pat	0	9	0	**48**	1	0	0	0	0	3	0	0	23	0
pinch	2	1	0	1	**82**	2	7	0	0	0	6	2	0	1
poke	0	8	0	2	14	**107**	8	0	0	0	1	0	5	2
press	0	0	0	2	9	3	**82**	11	0	0	0	1	1	0
rub	1	0	14	0	0	0	1	**47**	17	0	0	11	1	5
scratch	8	0	1	0	0	0	0	7	**50**	0	0	3	0	12
slap	0	14	5	18	1	0	0	0	2	**66**	0	3	17	0
squeeze	26	0	6	0	7	0	7	1	1	0	**58**	0	0	0
stroke	0	0	1	2	0	0	0	32	9	0	0	**94**	0	4
tap	0	17	0	37	2	8	1	0	2	16	0	2	**64**	6
tickle	0	0	9	0	0	0	0	11	33	0	0	4	1	**84**

Table 5: Confusion matrix for the CoST task with SVM using all training data (correct rate 60.51%)

	grab	hit	massage	pat	pinch	poke	press	rub	scratch	slap	squeeze	stroke	tap	tickle
grab	**80**	0	4	0	6	0	14	3	2	0	52	0	0	0
hit	0	**74**	0	13	0	0	0	0	0	41	0	1	13	0
massage	1	0	**91**	0	0	0	1	9	3	0	0	1	0	3
pat	0	6	0	**57**	1	1	0	0	0	2	0	0	23	1
pinch	2	2	0	0	**79**	1	4	0	1	0	8	6	0	2
poke	0	5	0	3	7	**99**	4	0	0	1	1	1	3	0
press	1	0	0	2	13	6	**88**	9	0	0	0	0	2	0
rub	3	0	8	0	0	0	1	**41**	10	0	0	14	0	4
scratch	6	0	7	0	0	0	0	8	**56**	1	0	0	0	15
slap	0	21	0	18	2	0	0	0	0	**64**	0	5	19	0
squeeze	27	0	0	0	12	0	8	1	1	0	**58**	0	0	0
stroke	0	0	1	2	0	0	0	32	11	0	0	**86**	0	4
tap	0	11	0	25	0	13	0	0	0	11	0	1	**60**	3
tickle	0	1	9	0	0	0	0	17	36	0	0	5	0	**88**

Table 6: Confusion matrix for the CoST task with Random Forest using manually selected training data (correct rate 61.34%)

System description	Correct rate
Neural network	35%
Correlation and moments based feature extraction + SVM classification	50%
Random Forests	61%
Decision trees	63%
Multiboost	65%
RandomForest	67%
Ensemble of DNN-HMM, Gaussian HMM and spectral features	68%
Anonymized system (SVM and RF, see table 3)	**from 68.52% to 70.91%**
→*state-of-the-art* [5]	*86%*

Table 7: Results of all other systems for the HAART task as reported by the challenge organizers. In italic, state-of-the-art result from [5]. In bold, our systems. (The system description was provided by the participants).

System description	Correct rate
ConvNet	45%
Neural network	45%
Ensemble of DNN-HMM, Gaussian HMM and spectral features	47%
Correlation and moments based feature extraction + SVM classification	53%
→*state-of-the-art* [4, 24]	*54%*
SVC OneVsRest	54%
Forest Bagging	57%
Multiboost	58%
RandomForest	59%
Anonymized system (SVM and RF, 2 training datasets, see table 2)	**from 59.91% to 61.34%**
→*state-of-the-art* [17]	*64.6%*

Table 8: Results of 8 best other systems for the CoST task as reported by the challenge organizers. In italic, previous state-of-the-art results from [4, 17, 24], In bold, our systems. (The system description was provided by the participants).

same classifiers are tested in two conditions: using all training set or our manually filtered one. As reported by organizers, our systems have the best correct rate in this year challenge (70.91% on HAART, 61.34% on CoST). However, our performances are slightly better that other participants and under previous reported state-of-the-art results.

In conclusion, one can say that our statistical analysis of the grid touch sensor and our classification method contribute to social touch classification. Lessons learned and depicted in this article can be combined with information from other participants. This may lead to a next step improvement in social touch recognition domain.

References

[1] M. D. Cooney, S. Nishio, and H. Ishiguro, "Recognizing affection for a touch-based interaction with a humanoid robot," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 1420–1427, IEEE, 2012.

[2] A. Billard, A. Bonfiglio, G. Cannata, P. Cosseddu, T. Dahl, K. Dautenhahn, F. Mastrogiovanni, G. Metta, L. Natale, B. Robins, *et al.*, "The roboskin project: Challenges and results," in *Romansy 19–Robot Design, Dynamics and Control*, pp. 351–358, Springer, 2013.

[3] H. Knight, R. Toscano, W. D. Stiehl, A. Chang, Y. Wang, and C. Breazeal, "Real-time social touch gesture recognition for sensate robots," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 3715–3720, IEEE, 2009.

[4] M. M. Jung, "Touching the Void - Introducing CoST : a Corpus of Social Touch," in *Proceedings of the 16th International Conference on Multimodal Interaction*, pp. 1–8, 2014.

[5] A. Flagg and K. MacLean, "Affective touch gesture recognition for a furry zoomorphic machine," in *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction*, pp. 25–32, ACM, 2013.

[6] W. Hillis, "Bounding box gesture recognition on a touch detecting interactive display," May 18 2010. US Patent 7,719,523.

[7] M. D. Cooney, S. Nishio, and H. Ishiguro, "Importance of Touch for Conveying Affection in a Multimodal Interaction with a Small Humanoid Robot," *International Journal of Humanoid Robotics*, vol. 12, p. 1550002, 2015.

[8] B. D. Argall and A. G. Billard, "A survey of tactile human–robot interactions," *Robotics and Autonomous Systems*, vol. 58, no. 10, pp. 1159–1176, 2010.

[9] W. Abdulla, D. Chow, and G. Sin, "Cross-words reference template for dtw-based speech recognition systems," in *TENCON 2003. Conference on Convergent Technologies for the Asia-Pacific Region*, vol. 4, pp. 1576–1579 Vol.4, Oct 2003.

[10] V. Niennattrakul, D. Srisai, and C. A. Ratanamahatana, "Shape-based template matching for time series data," *Knowledge-Based Systems*, vol. 26, pp. 1–8, 2012.

[11] B. Hartmann and N. Link, "Gesture recognition with inertial sensors and optimized dtw prototypes," in *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*, pp. 2102–2109, Oct 2010.

[12] N. Gillian, R. B. Knapp, and S. O?Modhrain, "Recognition of multivariate temporal musical gestures using n-dimensional dynamic time warping," in *Proc of the 11th Int'l conference on New Interfaces for Musical Expression*, 2011.

[13] H.-R. Lv, Z.-L. Lin, W.-J. Yin, and J. Dong, "Emotion recognition based on pressure sensor keyboards," in *Multimedia and Expo, 2008 IEEE International Conference on*, pp. 1089–1092, IEEE, 2008.

[14] G. Ten Holt, M. Reinders, and E. Hendriks, "Multi-dimensional dynamic time warping for gesture recognition," in *Thirteenth annual conference of the Advanced School for Computing and Imaging*, 2007.

[15] S. Uchida and H. Sakoe, "A monotonic and continuous two-dimensional warping based on dynamic programming," in *icpr*, p. 521, IEEE, 1998.

[16] D. Vaufreydaz, W. Johal, and C. Combe, "Starting engagement detection towards a companion robot using multimodal features," *Robotics and Autonomous Systems*, pp. –, 2015.

[17] S. van Wingerden, T. J. Uebbing, M. M. Jung, and M. Poel, "A neural network based approach to social touch classification," in *Proceedings of the 2014 workshop on Emotion Recognition in the Wild Challenge and Workshop*, pp. 7–12, ACM, 2014.

[18] X. Pan, H. Zhao, Y. Zhou, C. Fan, W. Zou, Z. Ren, and X. Chen, "A preliminary study on the feature distribution of deceptive speech signals," *Journal of Fiber Bioengineering and Informatics*, vol. 8, no. 1, pp. 179–193, 2015.

[19] P. Duhamel and M. Vetterli, "Fast fourier transforms: a tutorial review and a state of the art," *Signal processing*, vol. 19, no. 4, pp. 259–299, 1990.

[20] J. Makhoul, "A fast cosine transform in one and two dimensions," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 28, no. 1, pp. 27–34, 1980.

[21] R. C. Gonzalez and R. E. Woods, "Digital image processing reading," *MA: Addison-Wesley*, 1992.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[23] P. Hanchuan, L. Fuhui, and D. Chris, "Feature Selection Based on Mutual Information : Criteria of Max-Dependency, Max-Relevance and Min-Redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[24] M. M. Jung, "Towards social touch intelligence: Developing a robust system for automatic touch recognition," in *Proceedings of the 16th International Conference on Multimodal Interaction*, ICMI '14, (New York, NY, USA), pp. 344–348, ACM, 2014.