# A Neural Network Based Approach to Social Touch Classification

Siewart van Wingerden, Tobias J. Uebbing, Merel M. Jung, Mannes Poel
Human Media Interaction Group
University of Twente, P.O. Box 217, 7500 AE, Enschede, The Netherlands
siewart@gmail.com, t.uebbing@me.com, m.m.jung@utwente.nl, m.poel@utwente.nl

## ABSTRACT

Touch is an important interaction modality in social interaction, for instance touch can communicate emotions and can intensify emotions communicated by other modalities. In this paper we explore the use of Neural Networks for the classification of touch. The exploration and assessment of Neural Networks (NNs) is based on the Corpus of Social Touch established by Jung et al. This corpus was split in a train set (65%) and test set (35%), the train set was used to find the optimal parameters for the NN and for training the final model. Also different feature sets were investigated; the basic feature set included in the corpus, energy-histogram and dynamical features. Using all features led to the best performance of 64% on the test set, using a NN consisting of one hidden layer with 46 neurones. The confusion matrix showed the expected high confusion between pat–tap and grab–squeeze. A leave-one-subject-out approach lead to a performance of 54%, which is comparable with the results of Jung et al.

## Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Haptic I/O; I.5.2 [**PATTERN RECOGNITION**]: Design Methodology—*Classifier design and evaluation*

## Keywords

Social Touch; Touch Gesture Recognition; Neural Networks

## 1. INTRODUCTION

Interpersonal touch can communicate emotions and can intensify emotions communicated by other modalities [8]. Furthermore, touch can influence the emotions, attitude and social behavior of the person that is touched [5]. Humans are able to understand the meaning of social touch such as emotions [8]. However, robots and other interfaces should be able to understand social touch as well. As the amount of robots operating in close proximity to humans, for example in health care, increases there is a need for automatic understanding of social touch [4]. Thus far, scarce research has been carried out to create suitable devices for capturing and classifying social touch to enable social intelligent interaction [3, 9, 10, 11, 13, 14, 15]. Envisioned applications for these interfaces are: robot therapy [3, 15], remote communication [10] and interactive stuffed animals [10].


*ERM4HCI'14*, November 16, 2014, Istanbul, Turkey.

ACM 978-1-4503-0124-4/14/11 ...$15.00.
http://dx.doi.org/10.1145/2668056.2668060

The focus of this paper is to explore a Neural Network (NN) based approach for the classification of social touch gestures. Furthermore, the classification performance of various feature sets inspired by video classification is tested. As a benchmark we use the 'Corpus of Social Touch' (CoST) introduced by Jung et al. [9].

The rest of the paper is organized as follows. Section 2 gives an overview of the related work. Afterwards, in Section 3 the methodology for exploring the NN based approach is explained followed by the description of various feature sets in Section 4. The classification results are presented in Section 5. Subsequently, the possible influencing factors for the different performance indicators are discussed in Section 6. In the end, conclusions are drawn in Section 7 and in Section 8 advice for future work on social touch is given.

## 2. RELATED WORK

Machine Learning has been applied to classify similar social touch data in the past. This section starts by listing previous work on the topic of social touch recognition. Also, research on video classification is presented because of similarities in data representation.

In the introduction paper of CoST, Jung et al. [9] used two algorithms to classify 14 touch gestures. Using leave-one-subject-out cross-validation, Bayesian classifiers yielded a mean accuracy of 54% while Support Vector Machines (SVMs), using a linear kernel, obtained an accuracy of 53%.
Chang et al. [3] built a custom gesture recognition engine for classification of four touch gestures (stroke, slap, poke, and pat) performed on different body areas of a robotic lap pet. Depending on the sensor density of the body location, an accuracy between 11% and 77% was found.
Boosting ('LogitBoost') was used by Silvera-Tawil et al. with leave-one-subject-out cross-validation for the classification of touch applied on an artificial sensitive skin wrapped around a mannequin arm [13, 14]. Nine touch gestures (pat, push, scratch, slap, stroke, tap, pull, squeeze and 'no touch') were classified with a mean accuracy of 71% between multiple

subjects [13]. Also, a single human classifier was used to classify between subjects, resulting in an accuracy of 90%. In a follow up study, six emotions (anger, disgust, fear, happiness, sadness and surprise) and six social messages (acceptance, affection, animosity, attention-getting, greeting and rejection) were classified in addition to a 'no touch' category [14]. Average accuracies for emotions were 47% using the LogitBoost algorithm and 52% for human classification while social messages were recognized with a mean accuracy of 50% for the algorithm and 62% for human classification. Classification of the touch data between both studies showed links between the use of touch gestures and the expression of social touch, for example: squeeze was an often used to express fear.
Naya et al. [11] classified five touch gestures (pat, scratch, slap, stroke and tickle) performed on a $44 \times 44$ sensor grid. Combining Fisher's linear discriminant method with the k-nearest neighbors (k-NN) algorithm resulted in an overall accuracy of 87%.
NNs were tested by Stiehl and Breazeal [15] on a data set of nine touch gestures (contact, pat, pet, poke, rub, scratch, slap, squeeze and tickle) performed on an arm of a robotic animal companion. The NN selected contained 3 layers and 100 layer nodes at a learning rate of 0.001 using the log-sigmoid transfer function. Slap could not be classified well because of it's short duration. For the other eight touch gestures the model showed good performance for finding true negatives (94-98%) but varying results for true positive classification (23-84%).
Nakajima et al. [10] let subjects perform seven touch gestures (grasp, hug, press, punch, rub, slap and 'no-touch') on a balloon interface. Classification using SVMs with the radial basis function kernel resulted in an overall accuracy of 75% between multiple subjects.

To be able to more accurately recognize touch gestures and the emotional meaning of touch, more research is needed into the characteristics of different forms of touch. One approach is to look at new ways to extract describing features from touch data. The CoST data contains pressure values on an $8 \times 8$ grid updated at a fixed rate, which could be compared to the data representation of a low-resolution grayscale video. In a survey, Brezeale and Cook [1] distinguish between five feature classes for visual based video classification. Three of these classes contain features which are potentially useful for social touch classification. First, color-based features, which are created by binning the occurrences of colors into a histogram to describe color distribution and illumination (e.g. [16]). Second, features based on objects in the video, which can track trajectories and derive colors (e.g. [12]). Lastly, motion-based features, which are based on (approximated) movement of the camera and objects. Also, Euclidean pixel distance between sequential frames can be used for classification. Often derivatives are used to obtain optical flow (e.g. [6]). Besides the features used by Jung et al. [9] additional features will be extracted inspired by the features found in video classification.

# 3. METHODOLOGY

This section describes the data set used, the methodology for division of train and test set, exploring the feature space, and tuning the hyper-parameters of the Feedforward NN.

## 3.1 The data set: CoST

For the exploration of the use of NNs for touch classification we used CoST, a data set developed by Jung et al. [9]. This corpus contains 14 touch gestures, see Table 1, performed by 31 subjects on an $8 \times 8$ grid of pressure sensors. Resulting in 64 channels of data per frame refreshing at 135 hertz. The data values given by each channel are integers ranging between 0 and 1023. Each subject performed each gesture 6 times and in 3 variations: gentle, normal and rough. In this research we only focus on the rough variant of the 14 gestures. More details on the data set can be found in [9].

| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
|---|---|---|---|---|---|---|
| *grab* | *hit* | *massage* | *pat* | *pinch* | *poke* | *press* |
| **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| *rub* | *scratch* | *slap* | *stroke* | *squeeze* | *tap* | *tickle* |

**Table 1: Description of the 14 touch gestures contained in CoST.**

## 3.2 Train and test set

The entire data set of 2602 samples is split into a test (35%, 910 samples) and a training set (65%, 1692 samples). The training set is divided in 10 different folds of equal size used for 10-fold cross-validation. Each fold and set had (almost) the same proportional amount of samples for each gesture as the original data set. The folds were used for determining optimal hyper-parameters, such as number of hidden layers and number of neurons in each hidden layer. These sets were kept fixed over all experiments.

## 3.3 Feature selection

The features taken into consideration for this research are based on the features defined in [9] (the basic feature set) and extended by new features described in Section 4. The feature selection is subset based, each feature subset is added to the basic feature set and the classification performance was measured using a Feedforward NN with one hidden layer and the number of hidden neurons equal to the number of features.

## 3.4 Neural Network architecture

For the touch gesture classification task a Feedforward NN architecture is used. For training the scaled conjugate gradient training method was applied. The default training values for second derivative approximation ($\sigma = 5.0 \times 10^{-5}$) and Hessian indefiniteness ($\lambda = 5.0 \times 10^{-7}$) were selected. The maximum number of training iterations was set to 1000, however after 6 subsequent increases of the error on the validation set the training is terminated. The hyper-parameters – number of hidden layers and number of hidden neurons per layer – are determined using the 10-fold cross-validation described above. The number of hidden layers was varied between 1 and 10 and the number of hidden neurons between 1 and 40 with a step size of 10 below 100 and a step size of 50 in the range 100-400. Moreover for each hidden layer the number of hidden neurons is the same. The best overall architecture, determined by the best performance in the 10-fold cross-validation on the train set, was 1 hidden layer with 10-80 neurons. For each feature set the number of hidden neurons was determined by an exhaustive search in this range.

# 4. FEATURE SETS

This section describes the feature sets used for classification.

**Basic features** - The basic features provided alongside of the raw data include average pressure (1), maximum intensity (2), variability in pressure (3), average pressure per column (4-11) and per row (12-19), the size of the contact area (20, 21), temporal peaks in channels exceeding a certain threshold (22, 23), temporal displacement of the center of mass (centroid) (24-27) and the duration of the gesture (28). A more detailed specification of the basic features can be found in the paper introducing CoST [9].

Besides the basic features, additional features based on the those found in video classification have been created. All of these were derived from the provided raw data. These new features can be put in three distinct classes:

**Histogram-based features** - The histogram based features are closely related to the in section 2 described color-based features. Instead of three values, a single value of intensity for each sensor channel was obtained in the 0-1023 range and put into bins. The amount of bins was varied from 2 to 32, the highest accuracy for 10-fold cross-validation on the train set was obtained using 8 bins (M = 63.2%, SD=0.4%). Although, it should be mentioned that any amount of bins between 2 and 9 provided similar results. Using 8 bins, this leads to non-overlapping bins of size 128, starting with the first bin at pressure levels 0 to 128 until the last bin with a range of 896 to 1024. The average amount of channels per bin over all frames was saved as features (29-36).

**Motion-based features** - The motion based features are closely related to the in section 2 described object features from video data using motion and object trajectory. Spatial peaks and the center of mass are used to find trajectories and count the number of spatial peaks as features similar to object features. A spatial peak is found whenever a single channel has a value higher than all direct horizontal, vertical and diagonal neighboring channels in the pressure sensor grid. Additionally this channel must have a value above 25% (chosen over 0% and 50%) of the mean of all channels in the frame with the highest summed pressure within the entire gesture. This was applied to reduce false positives caused by noise. The first feature derived is the average amount of spatial peaks found over all frames in the entire gesture (37). Secondly, the variance in the amount of peaks per frame over the entire gesture was calculated (38). Thirdly, the distance of each peak to the center of mass in a frame, averaged over all peaks in all frames was obtained (39). Finally, the mean relative velocity of each peak to the velocity of the center of mass averaged over all peaks in all frames in the entire gesture was derived (40).

**Derivative-based features** - Three features were chosen based on motion features used in video, using temporal and spatial derivatives between the channels of frames and neighboring channels. Three final features were obtained from the raw data, using derivatives. Two derivatives were obtained from differences between channels within a frame, defined as the mean absolute difference in the x-direction, and y-direction averaged over the entire gesture (41, 42). Another derivative was found between frames within a channel, the mean absolute difference between the current and previous frame for each channel between all frames, averaged over all frames (43).

**Temporal features** - From all features described previously, including the basic features, non-averaged variants were derived. Those were saved as sequential data of the same length as the entire gesture or as length minus one for derivatives. This was not applicable for features 21 (maximum contact area), 28 (gesture duration) and 37 (variability in amount of spatial peaks). Feature 3 (pressure variability) was calculated as the variability within a single frame among channels.

**Mean segmented temporal features** - Another feature set was generated based on the temporal feature derived from the features in the basic feature set by scaling down the total amount of frames to 10 for each recorded gesture. This was done by taking the average value of the frames in steps of 10% over the entire gesture for each feature. The goal of this approach is to normalize the length of all gestures to a uniform length. Additionally this makes the data suitable for non-sequential classification. Thus after this procedure every former feature was split into 10 features, resulting in a total of 260 features in this feature set. The value of 10 frames was chosen to limit redundancy and dimensionality of the feature set.

# 5. RESULTS

Using only the basic features, the network was able to reach an accuracy of 60.2% (SD=3.2%) using 46 neurons on the hidden layer. By adding the histogram-based features to the basic features an accuracy of 61.6% (SD=2.8%) was found using 43 neurons. Adding only the derivative-based features to the basic features gave an accuracy of 63.5% (SD=4.0%) with 45 neurons. Using only the motion-based features in addition showed an accuracy of 60.4% (SD=1.9%) at 53 neurons. Finally, using all feature sets in the following order: basic, histogram, motion, derivative, provided the highest accuracy found using NN of 64.6% (SD=3.9%) with 46 neurons. Additionally, the mean segmented temporal features provided an accuracy of 53.8% (SD=3.5%) at 22 neurons.

Applying the best result, using all features at 46 neurons on the test set, after training the entire training set (all 10 folds) gave a result of 64.0%. In Table 2 the confusion matrix corresponding to all features applied on the test set can be found. The performance per touch gesture of the NNs using different feature sets, as described above, is compared in Table 3.

In order to be able to compare the results with earlier work we applied leave-one-subject-out cross-validation as well, using all samples. This was applied on two feature sets: the basic feature set and all feature sets combined. Since no tuning was applied, the number of 46 neurons in a single layer network was used, as found in 10-fold cross-validation for both feature sets. This yielded similar results to earlier work: basic feature set 54.9% (SD=13.7%) and all feature sets combined 54.1% (SD=15.0%).

**Table 2: Confusion Matrix of the best result using NN on all feature sets of the test set (mean accuracy = 64.0%). Predicted classes in columns, actual classes in rows. The final row states the fraction of correctly classified gestures samples for each gesture.**

| Gesture | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| grab (1) | 36 | 0 | 4 | 0 | 1 | 0 | 2 | 0 | 1 | 0 | 0 | 20 | 0 | 1 | 65 |
| hit (2) | 0 | 33 | 0 | 6 | 2 | 2 | 0 | 0 | 0 | 12 | 0 | 0 | 9 | 1 | 65 |
| massage (3) | 0 | 0 | 52 | 0 | 0 | 0 | 2 | 3 | 1 | 0 | 2 | 2 | 0 | 3 | 65 |
| pat (4) | 0 | 6 | 0 | 32 | 1 | 1 | 0 | 1 | 0 | 3 | 1 | 0 | 20 | 0 | 65 |
| pinch (5) | 0 | 0 | 1 | 0 | 49 | 10 | 3 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 65 |
| poke (6) | 0 | 4 | 0 | 0 | 4 | 47 | 2 | 0 | 0 | 0 | 0 | 0 | 8 | 0 | 65 |
| press (7) | 3 | 0 | 2 | 0 | 6 | 3 | 42 | 0 | 0 | 0 | 2 | 6 | 1 | 0 | 65 |
| rub (8) | 0 | 0 | 6 | 4 | 0 | 0 | 0 | 36 | 4 | 1 | 9 | 0 | 1 | 4 | 65 |
| scratch (9) | 0 | 0 | 2 | 2 | 0 | 0 | 1 | 2 | 37 | 0 | 1 | 1 | 0 | 19 | 65 |
| slap (10) | 0 | 10 | 0 | 11 | 0 | 0 | 0 | 0 | 0 | 37 | 1 | 0 | 5 | 1 | 65 |
| stroke (11) | 0 | 0 | 0 | 2 | 0 | 0 | 1 | 13 | 0 | 2 | 44 | 0 | 2 | 1 | 65 |
| squeeze (12) | 15 | 0 | 1 | 0 | 6 | 0 | 2 | 0 | 0 | 0 | 0 | 41 | 0 | 0 | 65 |
| tap (13) | 0 | 3 | 1 | 11 | 1 | 5 | 1 | 1 | 0 | 1 | 1 | 0 | 39 | 1 | 65 |
| tickle (14) | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 7 | 0 | 0 | 0 | 0 | 57 | 65 |
| total | 54 | 56 | 69 | 69 | 70 | 68 | 56 | 56 | 50 | 56 | 61 | 72 | 85 | 88 | 910 |
| corr. class. | 0.55 | 0.51 | 0.80 | 0.49 | 0.75 | 0.72 | 0.65 | 0.55 | 0.57 | 0.57 | 0.68 | 0.63 | 0.60 | 0.88 | 0.64 |

Three gesture subsets were evaluated as well using 10-fold cross-validation, maintaining the original 10 folds partitioning of the training data. NNs were trained using all feature sets as described above (results are omitted from Table 3). A subset using only pat, poke, slap and stroke provided an overall accuracy of 71.2% (SD=7.0%) using 42 neurons. Another subset of pat, scratch, slap, stroke and tickle yielded an accuracy of 60.0% (SD=6.4%) using 14 neurons. Lastly, the subset of hit, poke, press, rub, stroke and squeeze was tested obtaining an accuracy of 69.8% (SD=4.0%) at 45 neurons.

# 6. DISCUSSION

In the following the previously presented results are discussed.

The best result on the training set provides an accuracy of 64.6% combining all feature sets. This is an increase of 4.5% over the basic feature set. This increase can be attributed to the addition of new features, mainly the derivatives feature set (M=63.5%, SD=4.0%). However, adding all other feature sets a slightly higher accuracy (M=64.6%, SD=3.9%, 10-fold cv) was obtained. Moreover, the results on the test set provided the similar result of 64.0%, while having more varying results between gestures. Further exploration is necessary to specify which features in the feature sets have led to the increase in performance.

Using only the mean segmented temporal features an accuracy of 53.8% (SD=3.5%) was achieved, making it not advisable to use the temporal aspect of the data within CoST using a Feedforward NN.
Analyzing the results found in Table 3 it can be seen that improvements were found on the press gestures (71.9% accuracy) over the basic set (48.6%) an increase of 13.2%. This was also found for squeeze (resp. 57.0% and 47.9%, diff. 9.1%), and slap (resp. 57.0% and 52.1%, diff. 7.4%). However tap (resp. 50.4% and 46.8%, diff. 1.6%) and hit (resp. 66.1% and 66.9%, diff. 0.8%) performed slightly worse.
The confusion matrix in Table 2, allows us to compare the results of the touch gestures classified on the test set using all feature sets. This shows confusion between pat (4) and tap (13), this can be expected due to the similarity in performing the gesture. Furthermore, the grab (1) and squeeze (12) and the hit (2) and slap (10) gestures have high correlations. Again, these pairs are performed in a similar manner. These findings resemble those previous reported on this dataset [9].

In order to create comparability with previous reported results on the CoST data, leave-one-subject-out cross-validation was applied [9]. Using all feature sets, an accuracy of 54% was found which is comparable to results reported by Jung et al. [9] (Bayes: 54%; SVM: 53%). Nevertheless, it must be mentioned that the hyper-parameters were tuned on the 10-folds of the train set. This implies that no separate tuning was done for each fold, hence the data was tuned on the whole dataset including the data of the subject left out. It should be mentioned that the standard deviation of leave-one-subject-out cross-validation is quite high compared to using 10 folds (15.0% vs. 3.9%). Moreover, preliminary tests showed a higher accuracy (M=56%, SD=19%) for higher numbers of layers using all feature sets, but not for the basic feature set. However, tuning on the entire data set in this way is not advisable, since this will train the model on the test set as well.

Finally, classification results on gesture subsets led to higher accuracies, but still relatively low compared to related studies. The first subset (pat, poke, slap and stroke) was classified with an accuracy of 71.2% using NN, compared to an accuracy of 79% reported by [9] using a Bayesian classifier on the same dataset, while a similar data set classified these gestures with an accuracy up to 77% [3]. A second gesture set (pat, scratch, slap, stroke and tickle) obtained an accuracy of 60.0%, and 69% by [9], while [11] reported an accuracy of 87.3% on a similar dataset. A third subset (hit, poke, press, rub, stroke and squeeze) was classified with an accuracy of 69.8% using NN while [9] reported an accuracy of 77% using a Bayesian classifier. Results between studies

**Table 3: This table shows the average results per gesture separately for all feature sets using the training set on the NN classifier ($k = 10$) and the Bayesian classifier (LOSO, $k = 31$) by Jung et al. [9], followed by the feature sets using NN (LOSO, $k = 31$).**

| gest. | all | basic | deri. | hist. | mot. | seg. | Bayes | all | basic |
|---|---|---|---|---|---|---|---|---|---|
| grab | 0.74 | 0.69 | 0.75 | 0.74 | 0.72 | 0.73 | 0.65 | 0.72 | 0.69 |
| hit | 0.66 | 0.67 | 0.69 | 0.69 | 0.66 | 0.47 | 0.62 | 0.24 | 0.60 |
| mass. | 0.76 | 0.72 | 0.75 | 0.72 | 0.72 | 0.60 | 0.68 | 0.54 | 0.67 |
| pat | 0.39 | 0.34 | 0.41 | 0.39 | 0.41 | 0.35 | 0.23 | 0.30 | 0.34 |
| pinch | 0.69 | 0.66 | 0.70 | 0.70 | 0.68 | 0.71 | 0.67 | 0.67 | 0.60 |
| poke | 0.70 | 0.69 | 0.71 | 0.74 | 0.69 | 0.70 | 0.70 | 0.76 | 0.71 |
| press | 0.72 | 0.59 | 0.68 | 0.64 | 0.61 | 0.60 | 0.62 | 0.62 | 0.54 |
| rub | 0.59 | 0.53 | 0.51 | 0.53 | 0.47 | 0.50 | 0.41 | 0.47 | 0.40 |
| scrat. | 0.57 | 0.52 | 0.54 | 0.59 | 0.57 | 0.45 | 0.52 | 0.48 | 0.47 |
| slap | 0.65 | 0.57 | 0.65 | 0.60 | 0.63 | 0.56 | 0.50 | 0.63 | 0.69 |
| stroke | 0.73 | 0.73 | 0.66 | 0.73 | 0.74 | 0.43 | 0.59 | 0.64 | 0.70 |
| squee. | 0.57 | 0.48 | 0.55 | 0.50 | 0.40 | 0.41 | 0.35 | 0.42 | 0.48 |
| tap | 0.49 | 0.50 | 0.52 | 0.34 | 0.44 | 0.40 | 0.42 | 0.35 | 0.30 |
| tick. | 0.79 | 0.74 | 0.77 | 0.71 | 0.72 | 0.63 | 0.59 | 0.75 | 0.66 |
| mean: | 0.65 | 0.60 | 0.64 | 0.62 | 0.60 | 0.54 | 0.54 | 0.54 | 0.55 |
| sd: | 0.11 | 0.12 | 0.11 | 0.13 | 0.12 | 0.13 | 0.14 | 0.16 | 0.13 |

using different datasets of similar gesture sets are still difficult to compare because of differences in the way the touch data is obtained and classification protocols. Although results on the CoST data set reported by Jung et al. [9] were higher on these gesture subsets than the reported accuracies using NN in this paper, results on the whole gesture set using the same cross-validation method were similar.

# 7. CONCLUSION

The experiments carried out showed that Feedforward NNs are a suitable machine learning model to classify the CoST data based on structural features averaged over time. In experiments new features sets were added to a basic feature set provided by Jung et al. [9], the derivative feature set performed best (63.5%) out of all newly generated feature sets. However, usage of all features sets combined led to an even slightly higher accuracy (64.6 %). Classification on the mean segmented temporal feature set resulted in slightly lower accuracies of 53.8%. Based on these results, using the temporal aspect of the data with Feedforward Neural Networks seems inadvisable. Analysis of the performance of several machine learning algorithms on the separate gestures imply that potentially ensemble learning such as a committee of models could reach higher accuracies than just one single machine learning algorithm. Thus, future classification experiments with different machine learning models and techniques on the data of the 'Corpus of Social Touch' is worthwhile.

# 8. FUTURE WORK

Although, the use of temporal features did not increase performance on a Feedforward NN, temporal features should still be considered in future work on the CoST dataset. Hidden Markov Models (HMMs), which are often used for the classification of temporal data for example in speech recognition [17] could be explored to take advantage of the temporal aspect of touch gesture data. Preliminary experiments with HMMs on the temporal feature set showed promising results. The tools used[1] during those experiments required that the data was discretized into bins. The span of the eight bins for each feature covered the complete value range of each feature over the complete data set. Parameters were optimized using 10-fold cross-validation on the training set with randomly initialized transition and emission matrices. Classification of the test set with a HMM system containing 2 hidden states yielded a maximum accuracy of 60.5% on a subset of three gestures and 84.6% on a subset of two gestures.

Taking advantage of the high accuracies achieved in the classification of gesture subsets with Feedforward NN and HMMs in a committee of models seems even more worthwhile considering the preliminary results with HMMs. Also, related to this approach of ensemble learning a further analysis of dependencies and relations between gestures can be useful to create subsets. Further improvements in accuracy with the existing feature sets and Feedforward NNs could be reached by a more detailed analysis of the effects of feature sets or specific features within the sets on the classification performance.

Besides further research to improve the accuracy of touch gesture recognition, more research is needed on the use of touch gestures to express emotions. Links between touch gestures and emotion such as the use of squeeze to express fear identified by Silvera-Tawil et al. [14], could improve the recognition of human emotional expressions by social intelligent systems. The fusion of unimodal inputs such as emotions conveyed by speech, facial expressions or body language into one emotion representation for a more robust human-like emotion recognition system has been repeatedly proposed in the last years [2, 7].

[1]Kevin Murphy's Matlab toolbox - http://www.cs.ubc.ca/ murphyk/Software/HMM/hmm.html

## 9. ACKNOWLEDGEMENTS

This publication was supported by the Dutch national program COMMIT.

## 10. REFERENCES

[1] D. Brezeale and D. J. Cook. Automatic video classification: A survey of the literature. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(3):416–430, 2008.

[2] R. A. Calvo and S. D'Mello. Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on Affective Computing*, 1(1):18–37, 2010.

[3] J. Chang, K. MacLean, and S. Yohanan. Gesture recognition in the haptic creature. In *Proceedings of the International Conference EuroHaptics, (Amsterdam, The Netherlands)*, pages 385–391. 2010.

[4] T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and autonomous systems*, 42(3):143–166, 2003.

[5] A. Gallace and C. Spence. The science of interpersonal touch: An overview. *Neuroscience & Biobehavioral Reviews*, 34(2):246–259, 2010.

[6] X. Gibert, H. Li, and D. Doermann. Sports video classification using HMMS. In *Proceedings of the International Conference on Multimedia and Expo (ICME), (Baltimore, MD)*, pages II–345–348, 2003.

[7] H. Gunes and M. Pantic. Automatic, dimensional and continuous emotion recognition. *International Journal of Synthetic Emotions (IJSE)*, 1(1):68–99, 2010.

[8] M. J. Hertenstein, J. M. Verkamp, A. M. Kerestes, and R. M. Holmes. The communicative functions of touch in humans, nonhuman primates, and rats: a review and synthesis of the empirical research. *Genetic, Social, and General Psychology Monographs*, 132(1):5–94, 2006.

[9] M. M. Jung, R. Poppe, M. Poel, and D. K. J. Heylen. Touching the void – introducing CoST: Corpus of Social Touch. In *Proceedings of the international conference on Multimodal interfaces (ICMI), (Istanbul, Turkey)*. in press.

[10] K. Nakajima, Y. Itoh, Y. Hayashi, K. Ikeda, K. Fujita, and T. Onoye. Emoballoon a balloon-shaped interface recognizing social touch interactions. In *Proceedings of Advances in Computer Entertainment (ACE), (Boekelo, The Netherlands)*, pages 182–197. 2013.

[11] F. Naya, J. Yamato, and K. Shinozawa. Recognizing human touching behaviors using a haptic interface for a pet-robot. In *Proceedings of the International Conference on Systems, Man, and Cybernetics (SMC), (Tokyo, Japan)*, volume 2, pages 1030–1034, 1999.

[12] M. Roach, J. S. Mason, and M. Pawlewski. Motion-based classification of cartoons. In *Proceedings of the International Symposium on Intelligent Multimedia, Video and Speech Processing, (Hong Kong)*, pages 146–149, 2001.

[13] D. Silvera-Tawil, D. Rye, and M. Velonaki. Interpretation of the modality of touch on an artificial arm covered with an eit-based sensitive skin. *The International Journal of Robotics Research*, 31(13):1627–1641, 2012.

[14] D. Silvera-Tawil, D. Rye, and M. Velonaki. Interpretation of social touch on an artificial arm covered with an eit-based sensitive skin. *International Journal of Social Robotics*, pages 1–17, 2014.

[15] W. D. Stiehl and C. Breazeal. Affective touch for robotic companions. In *Prodeedings of the international conference on Affective Computing and Intelligent Interaction (ACII), (Beijing, China)*, pages 747–754. 2005.

[16] B. T. Truong and C. Dorai. Automatic genre identification for content-based video categorization. In *Proceedings of the International Symposium on Pattern Recognition, (Barcelona, Spain)*, volume 4, pages 230–233, 2000.

[17] A. P. Varga and R. K. Moore. Hidden markov model decomposition of speech and noise. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP), (Albuquerque, NM)*, pages 845–848, 1990.