Making a Socially Assistive Robot Companion Touch Sensitive*

Steffen Müller and Horst-Michael Gross

Technische Universität Ilmenau, Neuroinformatics and Cognitive Robotics Lab, 98693 Ilmenau, Germany,
steffen.mueller@tu-ilmenau.de,
http://www.tu-ilmenau.de/neurob

Abstract. Socially assistive robots are in the focus of research for a while. These robots are to be in close interaction with humans and try to communicate in a natural way. One intuitive modality for interaction is touch. This paper describes the design of our service robot intended for use in private homes of elderly people. We combined capacitive touch sensors at the stiff parts of the robot's shell with a pressure sensitive textile array sensor on the flexible back of our robot. A detailed description of the construction and properties of our pressure sensitive matrix sensor is presented. Furthermore, a maximum likelihood classification algorithm is presented, which is capable of online classification without explicit segmentation of touch gesture events in before.

Keywords: resistive pressure sensor, mobile robot, capacitive sensor

1 Introduction

The work presented here is part of the SYMPARTNER[1] project, which aims at developing a service robot intended to live together with elderly people in their private homes [1]. This project continues our previous work on socially assistive robots [5, 6, 14]. Like its predecessors, the new designed robot is capable of autonomous navigation in the apartment and, therefore, can bring specially designed services to the user. These services comprise calendar management with active reminders, communication via video telephony, encouraging people, entertaining by means of presenting various media and social companionship. One function, which is also relevant for the interaction in a sitting position is the ability to stop the robot's autonomous navigation in order to allow the users to position the robot manually according to their demands. For that purpose, it is necessary to enable the robot to notice when people push or grab it, which has been realized by means of capacitive touch sensors. A pressure sensitive patch on

* This work has received funding from the German Federal Ministry of Education and Research (BMBF) to the project SYMPARTNER (grant agreement no. 16SV7218).

[1] SYMbiosis of PAul and RoboT companion for Emotion sensitive caRe (www.sympartner.de)

the back of the robot additionally can be used for receiving unspecific feedback to the robot in form of petting gestures or small slaps. The robot will show an emotional reaction expressed by means of facial animations and sound outputs in that case. This functions is intended to reinforce the relationship of the user and the robot. In our ongoing work the feedback via touch gestures will be used for adaptation of the robot's interaction behavior.

The remainder of this paper is structured as follows: First, related work in the field of haptic human robot interaction will be discussed. Then, the robot and the tactile sensor hardware will be described in detail followed by an introduction to our real-time classification approach for touch gestures on the sensitive patch on the back of the robot.

2 Related Work

Tactile sensors have a wide range of application and come along with different qualities of the signal gained. On the one hand, there are systems providing only a touch signal (capacitive sensors), and on the other hand there are force or pressure sensors. Detectable forces thereby can be in perpendicular direction only or the amount of shearing forces at the surface is detectable as well. Furthermore, the position of a touch event can be distinguished more or less accurate, whereby the amount of independent touch areas is of interest for a system designer. Similarly, there is a variety of physical measurement principles available like optical, capacitive, and resistive effects as possible sources for information [16]. Additionally, [2] proposed a method for using changes in the magnetic field generated by bending small magnetized fibers on a surface. Also acoustic surface waves have been used for recognizing changes in the transmission characteristics, if an object (e.g. a hand) gets in contact with the surface. An important aspect for practical application of such systems in a robotic prototyping application is the manufacturing process. Commercially available touch sensor systems often are not suitable for the curvy shape of a robot or can only be adapted with high costs. Nevertheless, there are several projects explicitly dealing with the development of complex artificial skin [9]. For application at a human hand prosthesis, these systems comprise a variety of modalities like pressure, temperature, and humidity. On mobile assistive robots these modalities would be interesting too, but at the moment these techniques may break the budget of many robotics projects. A survey on artificial skin and tactile sensing for socially interactive robots can be found in [15].

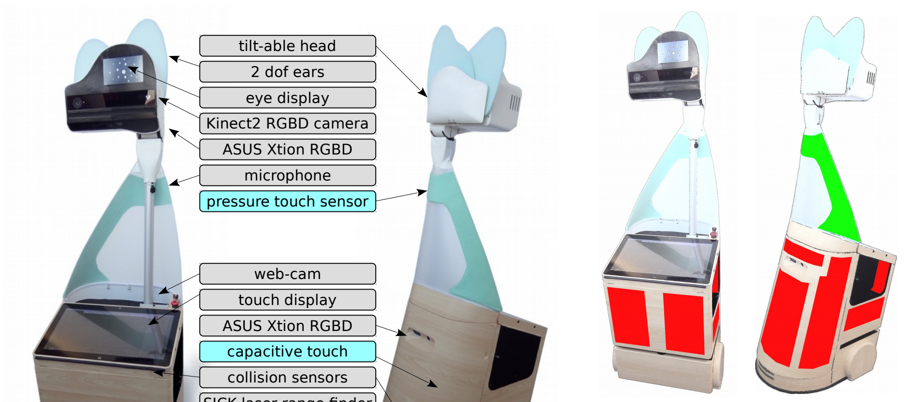
A popular approach to overcome the shape restrictions is the usage of a flexible textile material, able to cover non-planar and even flexible surfaces [13]. Especially for recognition of social touch gestures on artificial creatures, such low cost solutions are widely used [3, 4].

With the availability of pressure array sensors a competition for gesture classification and recognition methods emerged [8, 7]. Open questions here are: What is a useful set of distinguishable gestures? and Which features and classifier are best suited for the data? In a former project, we also developed a pressure sensitive patch of fur [11], that could be mounted on the convex head of our service robots Max and Tweety. With a simple Bayesian classifier, we were able to classify chunks of pressure data. This approach and also other popular methods [4] have the problem, that touch signals first have to be segmented before a classification, which leads to a delay for the recognition depending on the size of the window to be classified.

In this paper, we propose a sliding window classifier, which is able to overcome that drawback and is able to detect an event as soon as the pattern makes a distinction from other classes possible.

In addition to the pressure array sensor for touch gesture recognition, a capacitive touch technique has been used on our previous robots as well [10] in order to enable them for motor assisted pushing of the robot. In contrast to that, our new platform can be moved with ease, and a simple binary signal for stopping the robot's autonomous movements is sufficient.

3 Robot Hardware



(red) placement of eleven capacitive touch electrodes, (green) position of the pressure sensitive array sensor

Fig. 1: Overview on the robot's sensor systems mainly used for autonomous navigation and interaction with a user.

Fig. 1 shows our robot platform with the sensors available. The robot has been designed by University of Siegen [17] and was manufactured by Metralabs GmbH Ilmenau. Its mobility is based on a differential drive, and the robot has battery capacity for about 4 hours of mobile operation, where two dual core mobile PCs and a variety of active sensors have to be provided with power.

Together with its charging station, the robot is able to autonomously recharge itself. With this capability, an around-the-clock operation can be guaranteed. The footprint of approximately 45 x 55 cm is chosen to be as small as possible in order to allow for navigation in narrow indoor environments. For navigation purposes, the system is equipped with a SICK laser range scanner at a hight of 20 cm, two ASUS Xtion RGB-D cameras on the back and on the head facing downwards. For people detection, a Kinect2 RGB-D sensor is mounted at a tiltable head. Interaction is mainly supported by a 15' touch display and sound outputs. Head animations using a second display and two movable ears are used to communicate emotions and internal states of the robot. To complete the human interface, the robot has capacitive touch sensors at the base and a pressure sensitive textile sensor array at the back side of the neck (see Fig. 1 right side). Both of these sensors are controlled by a circuit board (see sec. 5.2) connected to the robot's PC via USB. In the following, these sensor systems will be described in more detail.

4 Capacitive Touch Sensors

First part of the robot's tactile interface is a set of capacitive touch sensors, which are mounted at the inner side of the 3d printed outer shell. As mentioned before, we only need a binary signal to stop the robot's movement, if a contact is detected. Therefore, the low spatial resolution of only eleven panels (see red faces in Fig. 1 right side) is sufficient. The sensitive areas are on the back, on the front, on both sides, and in the inner sides of the compartment beneath the display, which is the first choice for grabbing the robot to pull it from a sitting position. The sensor panels are made from a self adhesive copper foil directly attached to the plastic parts. The subdivision of the sensitive surface helps to keep the areas of the individual sensors small, which is necessary, since the relative change of capacitance when touched by a human hand gets smaller if the area of the electrode is large (see Fig. 2). By means of that, we could apply a standard IC (AT42QT1111) for reading the capacitance values into the microcontroller for evaluation.

In software, there is a continuously working calibration implemented. This allows to find the individual quiescent values for each of the panels by means of a moving average filter. The sensors are quite sensitive, such that the robot also gets signals when it is moving close to obstacles depending on the material (see next subsection and Fig. 2). For recognizing a touch event, we introduced a simple threshold for the difference of the actual signal and the average, that is high enough to avoid false detections in a normal home environment.

4.1 Properties of the Capacitive Sensors

The capacitive sensors respond to different materials and objects with specific characteristics. An experiment, where hands, legs, and a couch have been placed in different distances to the sensor surfaces, shows this property (see Fig. 2).

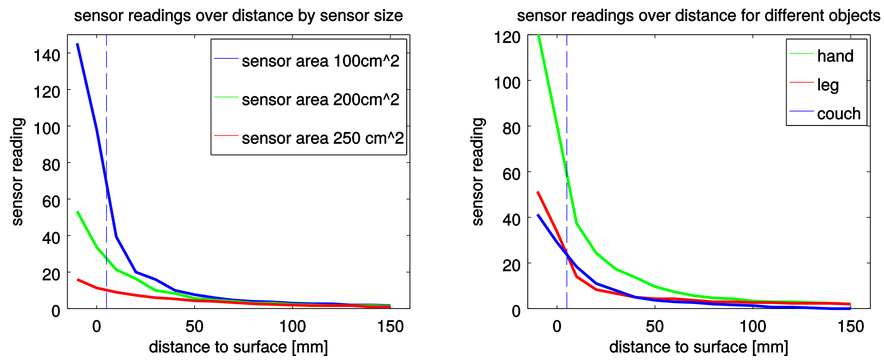


Fig. 2: Characteristics of capacitive sensors regarding material of the interacting object (right) and size of the electrodes (left); Values left of the dashed line are after contact with increasing pressure and thus increasing contact area.

This makes it difficult to derive the actual distance or pressure values from the sensor readings. The curves also show, that after contact (left of the dotted line) the pressure or contact area increases the signal further.

The comparison of the sensor responses at the various electrodes of different size shows that smaller electrodes give larger signals since the huge electrodes already have a higher capacitance even in untouched conditions. Differences in electrode size and sensitivity can be individually adjusted by means of the values of the reference capacitors on the circuit board.

The graphs of Fig. 2 shows, that it is possible to find a threshold for recognizing contact with a hand without reacting to close by situated furniture and legs, which is very important for all autonomous navigation tasks.

5 Pressure Sensor Array

The more interesting part of the tactile interface is the pressure sensitive matrix sensor. Like [13] and others we used textile materials for building the pressure sensor, since the treatment of such material is relatively easy and does not require any special equipment. Regarding the construction of resistive matrix sensors, we already had some experience from former projects [11], but for this robot we tried new materials and increased the number of channels and thus the spatial resolution of the sensor.

The layered structure of the matrix sensor basically consists of two conductive electrode layers one implementing the rows and the other one for the columns (see Fig. 3). The conductive material for these layers is an emf shielding fleece (100dB RF Shielding-Fleece Aaronia X-Dream™ by AARONIA AG), which has a coating with conductive adhesive at the back. In between these electrodes, a piezoresistive layer of Eeontex™ stretch fabric forms the sensitive elements at the crossing points of the rows and columns. The Eeontex™ material has a surface resistivity of $20k\Omega/cm^2$. In addition to these active layers, the matrix consists of two layers of iron fleece to give the sensor its structure and the wiring by means of enamelled copper wire. The sensor is wrapped in a cover from cotton fabric to realize the haptics and color specified by the designers. The completed sensor finally is sewed onto the $2mm$ thick back piece made from flexible transparent plastic.

5.1 Manufacturing the Sensor

In our former approach [11], the electrodes were made from copper coated nylon fabric sewed at a carrier layer of fabric. For the much smaller electrodes of our new construction, manual sewing was not a useful option anymore. Therefore, we have chosen a bonding solution. The electrodes were cut out, and the matrix layout can easily be arranged by fixating the strips at a sticky worktop with the adhesive side facing upwards. The electrodes have a width of about $13mm$ with a gap of $4-5mm$ (see Fig. 3). This gap is necessary to limit the crosstalk in the matrix, since there are no diodes that would allow to address individual cells. There also exists a fabric material, which already has a stripe pattern of conductive and insulating regions. Unfortunately, for our layout parallel stripes were not suitable, and the bonding of wires to that stretchy material is also not easy. In other setups of resistive textile matrix sensors, a spacer layer can be found between the electrodes and the piezoresistive material [4]. This exists to completely disconnect cells that are not touched. Unfortunately, this spacer increases the minimum pressure that is recognizable by the sensor, so we left that out accepting the resulting cross talk effects.

After laying out the electrodes, the wiring bond to the electrodes by means of self adhesive copper foil. After the wiring, a patch of iron fleece can be sticked to the arranged electrodes and the layout gets finalized by ironing. Fig. 3 on the left shows the row and column electrodes bond to the carrier. Fig. 3 on the right side shows the resulting sensor matrix with the piezoresistive layer in between.

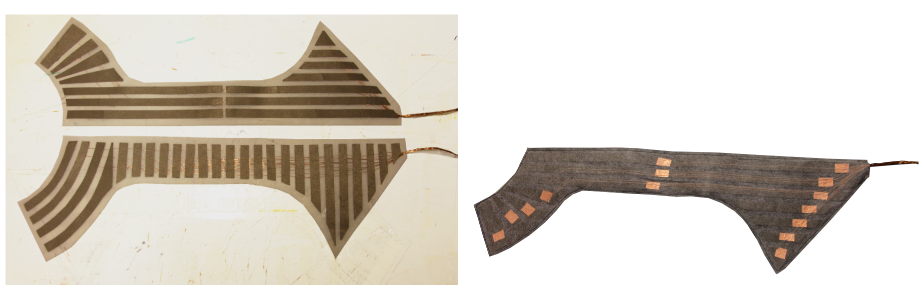


Fig. 3: (left) inner sides of the electrodes assemblies of the pressure sensitive matrix sensor, (right) sensor array sewed together.

5.2 Electronic Circuit

The electronics for reading out the sensors is based on an ATMEGA328P microcontroller. The resistive matrix is addressed via two analog multiplexers (74HD4067), one for the columns and one for the rows (see Fig. 4). This allows for 16 by 16 cells of spatial resolution, whereby only a part of this amount is used due to the irregular shape of the sensor area on the robot. A reference resistor is used to get a varying signal at the analog digital converter of the microcontroller. The microcontroller also communicates to the capacitive touch controller (AT42QT1111) via SPI and sends out the captured and preprocessed sensor values to the PC via USB.

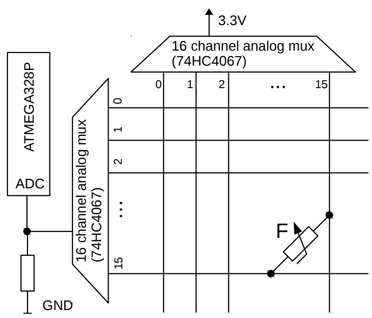


Fig. 4: Principle of measuring the pressure sensitive array.

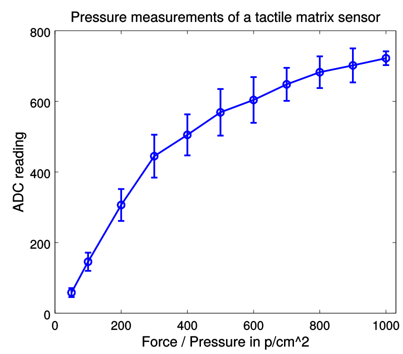


Fig. 5: Characteristics of the pressure matrix sensor resulting ADC values for 10 repetitions with a force on a $1cm^2$ area.

5.3 Properties of the Matrix Sensor

We did a series of experiments with a prototype of the matrix sensor to find out the characteristics of the specific materials. Fig. 5 shows the dependency of voltage at the adc input over the force (simple weight) put on a 1 by 1 cm patch on the matrix sensor. The graph shows that the sensor has its best resolution in the region below $400p$, which corresponds well to the forces occurring during natural touch gestures. According to [15], these are in a range from 0.3N to 10N. The sensor also shows correlating signals at very low forces below $50p/cm^2$, but they are difficult to distinguish from the changing quiescent value. That leads to a drawback of the setup using loose layers of flexible materials. After a pressure event, the sensor needs some time to get back to its resting state, which additionally can change in a limited range. That is the reason for the necessity of a run-time calibration that can deal with such unsteady signals. Alternatively, classification working on the temporal derivatives of the sensor values can be applied.

One further effect of a resistive matrix is a crosstalk between cells in the rows and columns of active cells. Because of the high resistance of unpressed matrix cells ($\approx 1M\Omega$) compared to the values of a pressed cell ($\approx 30k\Omega$), the amount of crosstalk activations can be neglected for our application.

6 Signal Processing and Online Gesture Classification

6.1 Calibration and Preprocessing

The microcontroller is programmed to read out each of the 256 matrix elements $\hat{a}_{i,j}$ at an internal rate of $100Hz$ with 10bit resolution. These raw signals at first are low pass filtered using a moving average filter in order to reduce the noise. Afterwards, they get sub-sampled and transmitted to the PC with a rate of $20Hz$ (now called $a^t_{i,j}$). As mentioned before, the resting state of the array sensor can change over time. Therefore, the individual minimum values $m_{i,j}$ are tracked over time using the following equation for temporal smoothing / low-pass filtering:

$$m^t = min\{a^t, m^{t-1} + \tau(a^t - m^{t-1})\} \quad (1)$$

Here the time constant τ defines the adaptation speed. Knowing the individual minimum values, for each sensor cell a normalized activation value $r^t_{i,j}$ is computed by subtracting the minimum and scaling it by a constant gain $g_{i,j}$.

$$r^t_{i,j} = min\{max\{g_{i,j}(a^t_{i,j} - m^t_{i,j}), k\}, 1\} \quad (2)$$

This simple scaling is reasonable, since the characteristic is nearly linear in the region below $300p/cm^2$. The values finally are cropped to $[k, 1]$ range and get further processed in the classification algorithm. The lower limit $k = 0.08$ is to cancel out noise and crosstalk activations.

6.2 Gesture Classification

To enable the recognition of touch gestures, in our previous approach [11] an event detection was employed, that captured fixed size sequences of the signals (4 sec) starting with excitation of an activity threshold. Because of this, the classification was always a bit delayed especially for short gestures (slap). This approach had been used to ensure that the pattern is always aligned to the window in the same way, and thus the variance in the data was reduced.

In order to overcome this delay problem, we now apply a sliding window classifier, which therefore has to deal with more variance in the patterns, because they are not longer aligned to the boundaries of the segmented classification window. An event detection is realized on a sample-based decision. We have chosen a probabilistic classifier that can provide a posterior probability distribution of the actual gesture classes given the features $\mathbf{f}^t$ of the current window of the signals. Only if one class has a significant probability, the event is triggered. The advantage of this approach is, that gestures can be recognized as soon as the patterns are significantly different to the other classes independent of the length of the gestures. For the evaluation in this paper, we used a window length of 3 seconds, which equals 60 samples to be processed.

As shown by [8], Bayesian classifier and more complex classifiers like SVMs can reach similar results. Therefore, we expect reasonable results despite the usage of such simple methods like Gaussian Mixture Models (GMM). Simpler models additionally have the advantage of requiring less data for training compared to complex models.

For each of the gesture classes $c \in C$, a Mixture of Gaussians model $p(\mathbf{f}|C=c) \propto \sum_{k=1}^{n} w_k \mathcal{N}(\mathbf{f}|\mu_k^c, \mathbf{\Sigma}_k^c)$ is learned from a training dataset by means of the Expectation Maximization (EM) algorithm in order to implement a maximum likelihood classifier. In our application, the inference of $p(C|\mathbf{f}^t)$ is done at the rate of the incoming pressure values ($20Hz$), where the a-priori probability of the gestures $p(C)$ was supposed to be uniform. If the probability of one class raises above a threshold and the class id did change, a gesture event is send to the application.

Social interaction gestures could successfully be classified based on the area, pressure, and their dynamics [12]. Also [4] defined a sliding window and extracted statistical features min, max, median, mean, and variance of pressure, and centroid position. Therefore, we also rely on a set of statistical features of the pressure signals. For computing features for a time window, at first basic feature values are extracted from the activation matrix of each time step $r_{i,j}^t$ in the window:

- the maximum activation $f_1^t = max_{i,j}\{r_{i,j}^t\}$
- the sum of the cells activations $f_2^t = \sum_{i,j}\{r_{i,j}^t\}$
- the number of active cells (which corresponds to the area) $f_3^t = |\{r_{i,j}^t > k\}|$
- the center of activation in the two dimensions of the sensor surface $f_4^t = (\sum_{i,j} i\ r_{i,j}^t)/f_2^t$ and $f_5^t = (\sum_{i,j} j\ r_{i,j}^t)/f_2^t$

To derive a feature vector for a time window $t \in \{t_0, \ldots t_{59}\}$ the maximum, mean and variance of the raw features f_1^t, f_2^t and f_3^t over all time steps in the window are computed. This provides the first nine features. Additionally, the number of time steps with at least one active cell is used as a feature describing the duration of a gesture. To encode the movement of the contact area, the distance of the activation centers of consecutive time steps is accumulated as a further feature followed by the variance of the 2d center of activation. One drawback of this approach is that the features do not describe where in the time window the patterns take place. This can be improved in future, e.g. by subdividing the window. Unfortunately, this would increase the dimensionality of the feature vector. Concluding, we have twelve features describing the signals in the sliding time window with a length of 3 seconds.

6.3 Evaluation and Results

For the scenario of our application, four gesture classes should be distinguished. These are (i) a **stroke** gesture that is used for praising the robot, (ii) a **tickling** gesture to provoke an emotional reaction, (iii) a **pushing** gesture, which actually is not intended for communication but occurs when the robot is positioned manually, and (iv) a **slapping** used to objurgate the robot for its behavior. There are other datasets having much more classes [8], but [7] showed, that fine granular gestures are not to distinguish robustly e.g. scratch and tickling.

Since touch gesture data depends on the actual sensor and its configuration on the robot, we captured an own gesture datasets with gestures performed by 8 persons. During the recording, the desired class was given in before, and there had to be a break of at least 4*sec* between consecutive gesture events. All the dataset comprises 200 events in 88,800 samples. For the training, the demanded class label simply is assigned to the feature vectors of all timestamps of a session also including silence intervals in between. By means of that, all class models contain the silence and transition features and, therefore, a balanced probability distribution will result in the classification process, if there are not characteristic features for one particular gesture (see Fig. 7 in between the events). For validation, the actual events have been segmented by means of an activity threshold on the raw pressure data to yield a ground truth (see bottom line of Fig. 7). For all further evaluations, a cross-validation was used, where each time the data of one person was used for validation and the other 7 datasets for training.

per sample cross-validation results

	stroke	tickle	push	slap	silence
stroke	**5822**	**161**	**93**	**53**	6
tickle	**34**	**1761**	**19**	**68**	97
push	**93**	**3**	**5931**	**89**	0
slap	**0**	**99**	**26**	**773**	49
not recognized	17171	12882	9490	19839	14285

TPR of all samples: 0.32
TPR of detected samples: 0.95
(bold faced submatrix)

confusion matrix of detected events

ground truth	detection: stroke	tickle	push	slap
stroke	34	4	2	1
tickle	6	52	7	8
push	6	1	37	3
slap	0	13	1	25

Correct: 148/200 = 0.74
False detections: 52/200 = 0.26
Misses: 51/200 = 0.25

Fig. 6: Confusion matrices for cross-validation results.

First, a parameter grid search was done, where the number of components and the probability threshold have been optimized. Best performance for event detection was found at a threshold of 0.99 and $n = 5$ mixture components. Fig. 6 (left) shows the resulting confusion matrix for a sample based classification. When the probability threshold was not exceeded by any of the classes, samples got count as not recognized. The true positive rate for sample-based classification is rather low with only 32%. Nevertheless, the correct classification rate of all samples with a significant probability is 95%, which seems promising for the event detection, because for the duration of a gesture not every sample needs to be classified, but one needs to have a high probability and the correct class. An evaluation based on the events showed satisfactory results (see Fig. 6 right). 74% of the 200 events could be detected correctly, while 25% were not detected at all. Additionally, 26% false detections occurred. Fig. 7 shows the classifier results for a part of the data. It can be seen, that the tickling and slapping gestures often do not reach high significance, which is due to the short and weak activation, which also can be part of the other gestures, too.

The actual classification results on unrestricted gesture datasets containing a variety of people seem to be too insufficient in terms of absolute numbers, which is caused by the inherent inter-personal variance of the data. Nevertheless, in a real application on a robot, which directly gives a feedback to touch gestures, a learning effect can be observed at the users. People change their touch patterns until the robot shows the intended reaction. This is similar to a human-human or human-pet situation. Therefore, we do not try to collect a huge dataset that covers a great variance of gestures. Such training data would reduce the distinctness of the gesture classes. Instead, we successfully applied the model containing only a limited but consistent set of samples, that gives the robot a character in terms of specific patterns that lead to an appropriate and consistent reaction.

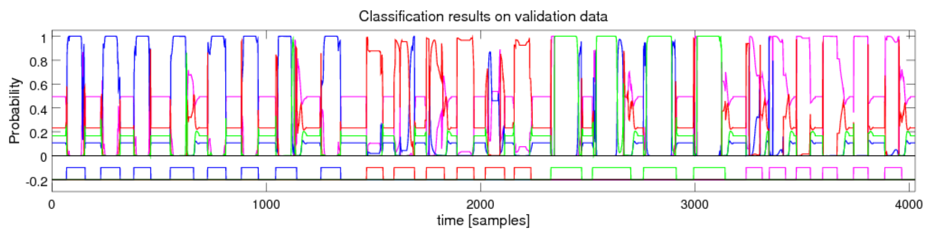


Fig. 7: Output probabilities for the classes stroke (blue), tickling (red), pushing (green), and slapping (magenta) over a validation dataset, the ground truth events are shown below.

7 Conclusion

With our experiments we showed, that it is possible to realize a multi-modal touch interface for a mobile assistance robot with very inexpensive materials and construction processes. Furthermore, we found that the classification of touch gestures in practice is acceptable. The pressure matrix system is able to distinguish coarse classes of different gestures independent of the position they are performed.

In future robotic projects, we plan to apply the pressure matrix solution again, while we will benefit from the easy production process and the high configurability of the approach. One further aspect of interest is the crosstalk effect in resistive matrix sensors in multi-touch situations. We currently work on an efficient computationally compensation but also try to combining the capacitive sensors and the pressure sensors to mask out the phantom activations in the pressure matrix.

References

1. http://www.sympartner.de (accessed April, 2018)
2. Alfadhel, A., Khan, M.A., Cardoso, S., Leitao, D., Kosel, J.: A magnetoresistive tactile sensor for harsh environment applications. Sensors 16(5), 650 (2016)
3. Cang, L., Bucci, P., MacLean, K.E.: Cuddlebits: friendly, low-cost furballs that respond to touch. In: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction. pp. 365–366. ACM (2015)

4. Cang, X.L., Bucci, P., Strang, A., Allen, J., MacLean, K., Liu, H.: Different strokes and different folks: Economical dynamic surface sensing and affect-related touch recognition. In: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction. pp. 147–154. ACM (2015)
5. Gross, H.M., Mueller, St., Schroeter, Ch., Volkhardt, M., Scheidig, A., Debes, K., Richter, K., Doering, N.: Robot companion for domestic health assistance: Implementation, test and case study under everyday conditions in private apartments. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS). pp. 5992–5999. IEEE (2015)
6. Gross, H.M., Schroeter, Ch., Mueller, St., Volkhardt, M., Einhorn, E., Bley, A., Langner, T., Merten, M., Huijnen, C., van den Heuvel, H., van Berlo, A.: Further progress towards a home robot companion for people with mild cognitive impairment. In: IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC). pp. 637–644. IEEE (2012)
7. Jung, M.M., Cang, X.L., Poel, M., MacLean, K.E.: Touch challenge'15: Recognizing social touch gestures. In: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction. pp. 387–390. ACM (2015)
8. Jung, M.M., Poppe, R., Poel, M., Heylen, D.K.: Touching the void–introducing cost: Corpus of social touch. In: Proceedings of the 16th International Conference on Multimodal Interaction. pp. 120–127. ACM (2014)
9. Kim, J., Lee, M., Shim, H.J., Ghaffari, R., Cho, H.R., Son, D., Jung, Y.H., Soh, M., Choi, C., Jung, S., et al.: Stretchable silicon nanoribbon electronics for skin prosthesis. Nature communications 5, 5747 (2014)
10. Mueller, St., Schroeter, Ch., Gross, H.M.: Low-cost whole-body touch interaction for manual motion control of a mobile service robot. In: Int. Conf. on Social Robotics (ICSR). LNCS, vol. 8239, pp. 229–238. Springer (2013)
11. Mueller, St., Schroeter, Ch., Gross, H.M.: Smart fur tactile sensor for a socially assistive mobile robot. In: Int. Conf. on Intelligent Robotics and Applications (ICIRA). LNAI, vol. 9246, pp. 49–60. Springer (2015)
12. Naya, F., Yamato, J., Shinozawa, K.: Recognizing human touching behaviors using a haptic interface for a pet-robot. In: Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on. vol. 2, pp. 1030–1034. IEEE (1999)
13. Pan, Z., Cui, H., Zhu, Z.: A flexible full-body tactile sensor of low cost and minimal connections. In: Systems, Man and Cybernetics, 2003. IEEE International Conference on. vol. 3, pp. 2368–2373. IEEE (2003)
14. Schroeter, Ch., Mueller, St., Volkhardt, M., Einhorn, E., Huijnen, C., van den Heuvel, H., van Berlo, A., Bley, A., Gross, H.M.: Realization and user evaluation of a companion robot for people with mild cognitive impairments. In: IEEE Int. Conf. on Robotics and Automation (ICRA). pp. 1145–1151. IEEE (2013)
15. Silvera-Tawil, D., Rye, D., Velonaki, M.: Artificial skin and tactile sensing for socially interactive robots: A review. Robotics and Autonomous Systems 63, 230–243 (2015)
16. Weiss, K., Worn, H.: Resistive tactile sensor matrices using inter-electrode sampling. In: Industrial Electronics Society, 2005. IECON 2005. 31st Annual Conference of IEEE. pp. 6–pp. IEEE (2005)
17. Welge, J., Hassenzahl, M.: Better than human: About the psychological superpowers of robots. In: International Conference on Social Robotics. pp. 993–1002. Springer (2016)