2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Daejeon Convention Center
October 9-14, 2016, Daejeon, Korea

Robust Material Classification with a Tactile Skin Using Deep Learning

Shiv S. Baishya Berthold Bäuml

***Abstract*— Attaching a flexible tactile skin to an existing robotic system is relatively easy compared to integrating most other tactile sensor designs. In this paper we show that material classification purely based on the spatio-temporal signal of a flexible tactile skin can be robustly performed in a real world setting. We compare different classification algorithms and feature sets, including features adopted and extended from previous works in tactile material classification and that are based on the signal's Fourier spectrum. Our convolutional deep learning network architecture, which we also present here, is directly fed with the raw 24000 dimensional sensor signal and performs best by a large margin, reaching a classification accuracy of up to 97.3%.**

I. Introduction

The sense of touch is a prerequisite for a robot to perform dexterous manipulation tasks. E.g., a pressure sensitive tactile sensor with high spatial resolution (in the range of mm) is needed to sense and interpret the contact situation to robustly grasp geometrically complex objects or to perform in-hand manipulation. If, in addition, the sensor exhibits a high temporal resolution (sample rate of some 100 Hz) and/or provides other modalities like temperature the robot's skills can be advanced to detect slippage during grasping or identification of an object's material simply by touching it.

In the last years an increasing interest in tactile sensing has shown up in the robotics community and led to a plethora of different sensor principles and designs (see Dahiya et al. [1] and Stassi et al. [2] for detailed reviews). A key differentiating criteria is how easily a tactile sensor can be integrated into a robotic system. The two extremes are on the one side "bulky" sensors which need to replace parts of the robotic structure, e.g., the BioTac sensor [3] from SynTouch [4] or Hosoda's finger tip sensor [5] which replace at least a phalanx of a finger. On the other side are thin ($<$1 mm) and flexible tactile skins which can be easily attached to the surface of a robot, e.g., even on top of a soft finger tip; examples are iCub's finger tip sensors [6], the pressure mapping sensors from Tekscan [7], DLR's elastic artificial skin [8][1], or a soft skin covering the Shadow Hand [9]. In between lie quasi-skin like stiff sensors made from printed circuit board modules [10] (width of some cm and thickness $<$1 cm), which can only be attached to rather modestly curved robot arms and bodies.

The bulky and quasi-skin sensors typically provide, beside pressure sensing, additional modalities like absolute temperature, heat flow, or an accelerometer, whereas the flexible skin sensors can provide spatio-temporal pressure signals only.

The authors are with DLR Institute of Robotics and Mechatronics, Münchnerstr. 20, 82234 Wessling, Germany.
{shiv.baishya,berthold.baeuml}@dlr.de

[1]This skin is commercially available from http://www.tacterion.com

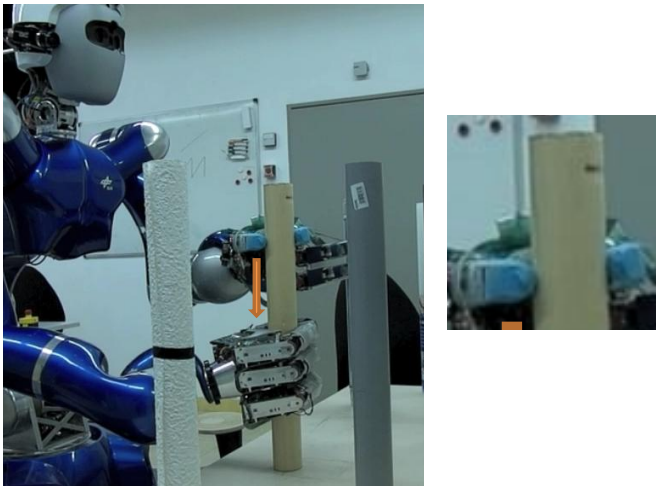

Fig. 1. DLR's *Agile Justin* [11] performing a sweeping motion to identify the material of the tube. Agile Justin is equipped with two DLR Hand-II [12] (all 12 joints are torque controlled). To the soft finger tip of the thumb of the left hand a flexible tactile skin with a 4×4 taxel array is attached providing a spatio-temporal pressure signal at 750 Hz sample rate. On the right a close-up of the contact situation is depicted.

In this paper we address the task of tactile material classification, i.e., objects of identical shape but different materials shall be identified by performing an exploration motion with a tactile sensor on the objects' surface. Up to now, material classification has only been reported using the bulky and stiff quasi-skin tactile sensor types, presumably due to the richer set of sensor modalities they provide (see Sec. I-B for a discussion). Here, we show that tactile material classification can also be performed using a flexible tactile skin.

A. Contributions

- We show that material classification purely based on the spatio-temproal signal of a flexible tactile skin is feasible and describe how the skin can be easily and quickly attached on a robot's soft finger tip.
- We analyze and discuss the problem of additional variability in the skin's signal in a real world application and on a real robot compared to a well-controlled test bench setup.
- We transfer learning methods which have been previously reported for material classifications with "bulky" sensors and which are based on the temporal Fourier spectrum to our tactile skin setup.
- For the first time we apply deep learning to tactile

978-1-5090-3762-9/16/$31.00 ©2016 IEEE

material classification resulting in superior performance compared to the spectral feature learning based methods, esp. with regard to robustness in a real world setting.

B. Related Work

1) Tactile Material Classification: Fishel and Loeb [13] report to classify 117 different materials with an accuracy of 95.4% using their BioTac sensor in a test bench setup with precisely controlled contact force and traction. They use Bayesian classification on three features: traction (from motor currents), roughness (the overall power of the Fourier spectrum from 20 to 700 Hz from the pressure sensor), and fineness (the centroid of the Fourier spectrum).

However, this impressive discrimination performance could not be transferred to more realistic robotic setups. As the same authors report in Xu et al. [14], using the BioTac mounted on a Shadow Hand, they could reach a classification accuracy of 99% – but only for 10 different materials and performing multiple exploration motions based on a Bayesian exploration framework. They kept only the spectral roughness feature but added the more "static" features compliance (one pressure electrode divided by joint angle) as well as temperature and thermal flow.

This result is in accordance with findings from Hölscher et al. [15]. They use a BioTac sensor on a PA10 robot and performed a comprehensive comparison of diverse feature sets and learning methods for the discrimination of 49 object surfaces. The result show that not elaborate features that depend on the detailed time structure of the signals but rather simple and "static" features (most prominent temperature and thermal flow) perform best. A linear SVM (support vector machine) in combination with multiple exploration motions reached an accuracy as high as 97%.

As a flexible tactile skin can not provide the modalities temperature and thermal flow, in a first step, we only adopt and extend the spectral features (see Sec. III-A).

Sinapov et al. [16] use a finger nail with an accelerometer as a tactile sensor and performed 5 different scratch motions (different velocities and directions) for discriminating 20 object surfaces. Instead of extracting low dimensional features they directly use the down-sampled (5 temporal and 25 frequency bins) spectrogram (time varying frequency response) of the 400 Hz sensor signal. Using a SVM classifier they reach an accuracy of 65.7% for a single scratching motion and 80% using all 5 motions.

We adopt a similar spectrogram based classification method to the tactile skin sensor in Sec. III-B.

Kaboli et al. [17] and Chathuranga et al. [18] address the problem of making tactile material classification based on high resolution time signals more robust. The former use the 3-axis accelerometers (250 Hz) of their stiff quasi-skin mounted on a Nao robot's body and the latter use a soft 3-axis force measuring tactile sensor (1 kHz) in a test bench setup where the effect of varying applied force and velocities is systematically studied. Both use correlation based features to dramatically reduce the high dimensional time series (some 100 values) to a low dimensional vector ($<$10 values) which is then fed into a SVM for classification.

Instead of using hand-crafted features, in this paper we address the robustness problem by learning robust features automatically from samples by applying deep learning (Sec. IV)

2) Deep Learning on Tactile Data: To our best knowledge, deep learning has never been applied to tactile material classification based on high resolution spatio-temporal signals (more than 100 Hz) before.

Schmitz et al. [19] used deep learning to recognize objects of different shape and softness with the Twendy-One Robot based on the static sensor readings (tactile, force/torque and joint angle sensors) after having grasped an object. Madry et al. [20] applied a deep hierarchical feature learning method to purely tactile spatio-temporal (sampling rate 10 to 100 Hz) sensor readings while grasping objects. Both works report significantly higher recognition accuracy compared to, e.g., SVM or shallow neural networks.

Büscher et al. [9] used a recurrent deep neural network in combination with a hand equipped with their soft tactile skin in a rather unusual material classification setting: the hand slowly squeezes a bag filled with a given material. From the recorded tactile and joint angle signals (100 Hz sample rate) a recognition accuracy of 79% could be reached.

II. Sensor and Experimental Setup

Our material classification task is motivated by our "Mars Habitat" demonstrator where the mobile humanoid robot Agile Justin [11] autonomously builds up a scaffold structure from single tubes. The robot should be able to discriminate tubes made of different materials by simply sweeping with its fingers over them, e.g., in case the camera view is obstructed.

A. Sensor Setup

We use one sensor patch of a Grip VersaTek® sensor 4256E from Tekscan[2]. The sensor is made from two thin flexible polyester sheets with a printed matrix of piezoresistive ink cells in between. It is only 0.1 mm thin and the patch with 4×4 taxels (size 2.5 mm) is 1.6 cm $\times$ 1.6 cm wide and the pressure range goes from 0.34 kPa up to 345 kPa. An important feature is the high sample rate of 750 Hz for reading out the full sensor. We integrated the sensor in our hard realtime software framework aRDx [21].

We attached the sensor patch to the tip of the thumb of the robot's left hand (a DLR Hand-II [12]) by using a double-sided adhesive tape (see Fig. 1 and 2). The robot fingertip is made of soft silicon and has a flat contact surface with the sensor. To increase the grip while sweeping over a surface we added a layer of latex cut off from a thin laboratory glove.

This simple attachment method can be performed easily and quickly and results in a stable position of the sensor relative to the finger's kinematics during usage. But when it is necessary to remove and reattach the sensor, e.g., for

[2]http://www.tekscan.com/4256E-pressure-sensor

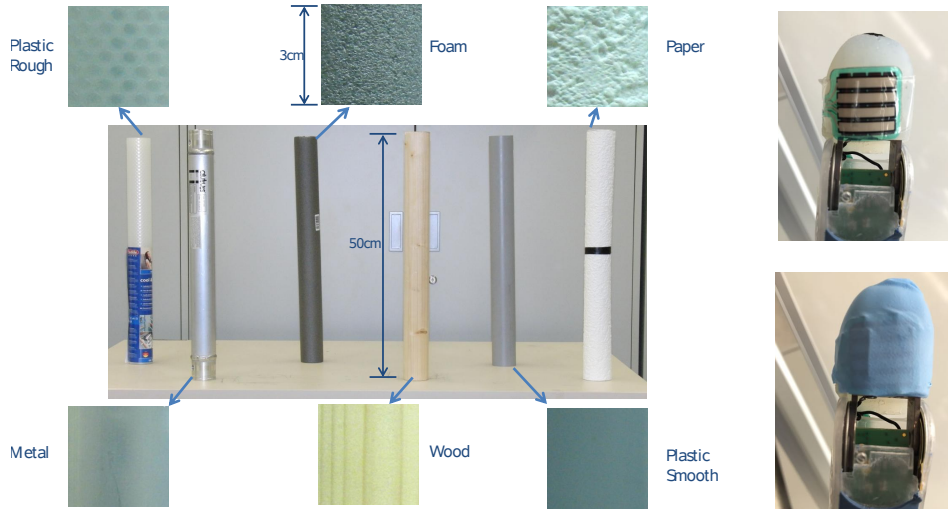

Fig. 2. *Left:* The $C = 6$ tubes made of different materials but with identical geometry (50 cm $\times$ 5 cm). The close-ups show the surface structure in more detail. *Plastic Rough* is a rolled-up nubby plastic mat; *Foam* is a elastic isolation foam tube; *Paper* is an ingrain wallpaper; *Plastic Smooth* is a plastic drain pipe; *Wood* is a wooden rod; *Metal* is a polished aluminum tube. *Right:* The soft silicon tip of the DLR Hand-II's thumb with the attached tactile skin sensor patch (top) and the additional piece of a thin latex laboratory glove (bottom).

repairs, the error for repositioning[3] the sensor on the soft finger tip can be as large as 1 mm.

B. Data Sets

To evaluate the material classification performance of the tactile skin we use only its spatio-temporal signal and no other sensors of the robot. The benchmark experiment is to discriminate $C = 6$ tubes made of different materials but having the same geometry (see Fig. 2).

The procedure for exploring a given tube is performed autonomously by the robot: grasp the tube with the right hand to stabilize it; grasp the tube with the thumb (equipped with the tactile sensor) and the index finger of the left hand and wait for 2 s; slide down along the tube with the left hand at a constant velocity of 3 cm/s for 3 s; release the tube. While sliding down, the contact force of the left hand's thumb is controlled to stay constant at about 1 N with a precision of about 20% using the hand's joint torque sensors.

For material classification we use the $T = 2$ s in the middle of the sliding motion resulting, given the sample rate of $f = 750$ Hz, in a $1500 \times 4 \times 4 = 24000$ dimensional spatio-temporal signal per sample (see Fig. 3). We name the signal s_n^m with n running over the $N_\mathrm{T} = 1500$ temporal and m over $M_\mathrm{S} = 4 \times 4$ spatial dimensions[4]. The value range of s_n^m is 0 to 255 digits, with larger values meaning larger pressure is exerted on the taxel.

We record three tube data sets with increasing classification difficulty, i.e., going from a lab like controlled environment to a real world setting. For each data set each tube is explored multiple times and after each exploration motion the tube is rotated randomly to cover the inhomogeneity of each tube in the recorded samples.

[3]We marked the sensor's outline on the finger tip for repositioning.

[4]For notational simplicity we write m for running over both spatial dimensions individually or when running over the flattened one dimensional vector. Which semantics applies becomes clear from context.

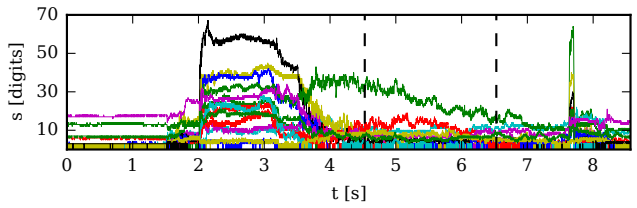

Fig. 3. Signal clipping (4x4 taxels sweep over metal tube). Only the middle part of the sweep motion (interval marked by the two vertical dashed lines) is used for learning.

1) T-LAB: This data set is recorded with the sensor attached once to the finger tip, i.e., without repositioning. Each tube is sampled 40 times resulting in $N_\mathrm{LAB} = 240$ samples in total.

2) T-REAL: In this more realistic data set the sensor was removed and reattached multiple times but with the best effort to have a high repositioning precision. For each of the 9 sensor positions, each tube was sampled 10 times resulting in $N_\mathrm{REAL} = 540$ samples in total.

3) T-HARD: In contrast to T-REAL for this data set the sensor was repositioned once more but with an offset of about 4 mm, i.e., a full taxel row. This leads to a significant change of the spatio-temporal signal. 10 samples for each of the tubes were recorded resulting in $N_\mathrm{HARD} = 60$ samples in total.

C. Cross-Validation

To assess the performance of a classification method we use n-fold cross validation to compute an estimate of the sensors classification accuracy and the confusion matrix.

For the T-LAB data set we use standard randomized n-fold cross validation where the data set is randomly split into n subsets and then the classifier is trained n times with $(n-1)$ subsets and tested against the one remaining subset. We then report the average accuracy as well as the minimal and maximal accuracy over the n runs as error bars.

As described in Sec. II-A in a real world usage scenario the tactile skin gets repositioned from time to time and it is not feasible to retrain a classifier (including executing the exploration motions) each time. Therefore the classifier has to be robust against sensor repositioning. To asses this robustness we use the T-REAL data set and perform a variant of the leave-one-out cross-validation which we call "leave-one-position-out" (LOPO) cross-validation. In LOPO cross-validation the classifier is trained with the samples from 8 sensor positions and tested against the 9th position.

In contrast, using naive randomized n-fold cross-validation with the T-REAL data set would significantly overestimate the classification performance as we will show in Sec. III-D.

To assess the robustness of a classifier against massive repositioning errors we train the classifier with all 9 positions from the T-REAL set and test it against the T-HARD set.

III. Classification with Spectral Features

In this section we present two feature sets – low dimensional L-features and high dimensional H-features – for material classification which are computed from the temporal Fourier spectrum of the spatio-temporal skin signal. Both sets are inspired by features previously used with other tactile sensor types.

A. L-Features

The three L-Features are directly derived from the spectral features reported in Fishel and Loeb [13]: roughness and fineness.

The discrete time Fourier transform of the signal s_n^m is defined as

$$\tilde{s}_q^m = \frac{1}{\sqrt{N_T}} \sum_n^{N_T} \mathrm{e}^{-\mathrm{i}\omega_q t_n} s_n^m, \tag{1}$$

with $t_n = n/f$ and $\omega_q = q \cdot 2\pi f / N_\mathrm{T}$.

The *roughness* is defined as the logarithm of the overall power of the temporal Fourier spectrum in the intervall $f_A = 20\,\mathrm{Hz}$ to $f_B = 375\,\mathrm{Hz}$ and summed over all taxels:

$$r = \log \sum_m^{M_S} \frac{1}{B-A} \sum_{q=A}^{B} \left| \tilde{s}_q^m \right|^2. \tag{2}$$

The *fineness* is defined as the centroid of the spectrum in a given interval. Unlike Fishel et al. we use not only one but two centroid features. One, c_1, from 20 Hz to 375 Hz (resembles the original fineness) and an additional low frequency centroid, c_2, from 2 Hz to 20 Hz. The centroid values of the individual taxels are combined in a weighted sum where each taxel contributes corresponding to its temporal mean activation:

$$c_i = \sum_m^{M_S} \sum_{q=A_i}^{B_i} w^m \tilde{s}_q^m \omega_q, \tag{3}$$

with the taxel weights

$$w^m = \frac{\sum_n^{N_T} s_n^m}{\sum_m^{M_S} \sum_n^{N_T} s_n^m}. \tag{4}$$

The three dimensional L-feature set is then $\{r, c_1, c_2\}$.

B. H-Features

The H-features are based on the short-time Fourier transform (STFT) [22] of the sensor signal, i.e, the frequency response of the sensor over time. This method is similar to Sinapov et al. [16] but we use a higher number of time and frequency bins and extend it to the multi-taxel signal. For each taxel the power of the spectrum is computed for each time bin $t_l = l\Delta t$ and frequency bin $\omega_q = q\Delta\omega$ and then the taxel values are combined in a weighted sum using the weights from (4) resulting in

$$p_{l,q} = \sum_m^{M_S} w_m \left| \frac{1}{\sqrt{N_T}} \sum_n^{N_T} \mathrm{e}^{-\mathrm{i}\omega_q t_n} s_n^m h(t_n - t_l) \right|^2. \tag{5}$$

$h(t)$ is the Dirichlet window function with a window size d.

Tuning the parameters using the T-LAB data set resulted in $\Delta t = 0.05\,\mathrm{s}$, $\Delta\omega = 2\pi \cdot 20\,\mathrm{Hz}$, $d = 0.1\,\mathrm{s}$, and cut-off frequencies f_A =2 Hz and $f_\mathrm{B} = 375\,\mathrm{Hz}$.

From this follows that the H-feature set $p_{l,q}$ is $40 \times 19 = 760$ dimensional. Because of the large differences in the frequency responses the H-features are normalized over the q index before they are fed into the classification algorithms.

C. Classification Algorithms

We apply three well-known classification algorithms [23], namely Gaussian classification, k-nearest neigbours (kNN), and support vector machine (SVM) to the L-features and kNN and SVM to the H-feature.

We tuned the the parameters of the classifiers with the T-LAB data set:

- kNN: k = 4;
- SVM: *scheme*: one against one; *kernel type*: "radial basis function" (RBF) with *kernel parameter* $\gamma = 1/N$.

D. Results

Fig. 4 depicts the average classification accuracy for the T-LAB data set using 10-fold cross validation. The accuracy is $> 80\%$ for all combinations of feature sets and classifiers and clearly higher than the random guessing accuracy of $1/C = 16.7\%$, proving that material classification is feasible using the flexible tactile skin.

The accuracy using the L-features is almost the same for all three classification algorithms and substantially lower than for the H-feature set. This suggests that all of the information about the material class the L-features contain can be extracted by the classification algorithms but the extreme reduction of the original 24000 dimensional signal to a 3 dimensional feature set drops important information.

The classifier using the H-features and SVM performs best with a high average accuracy of 96% with only a small variance over the test runs. kNN with H-features comes close with 90% but the results are less consistent with a 3 times larger min-max variation. This suggests that both classification algorithms could take advantage from the richer information encoded in the 760 dimensional H-features but SVM is more robust.

Fig. 5 shows the performance for the T-REAL data set using LOPO cross-validation. The accuracy degrades for all classifiers and severely drops even for the best classifier (again SVM with H-features) by 10% to only 86%. The confusion matrix and a discussion of which materials where hardest to discriminate is provided in Sec. V-C.

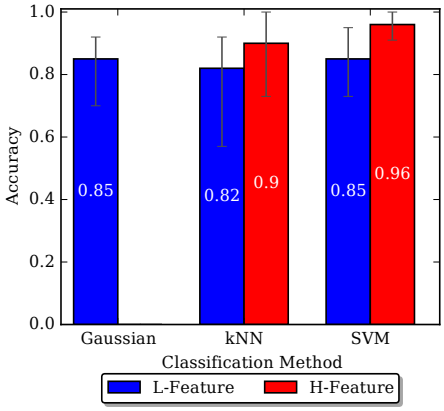

Fig. 4. Accuracy comparison for the T-LAB data set using randomized 10-fold cross validation.

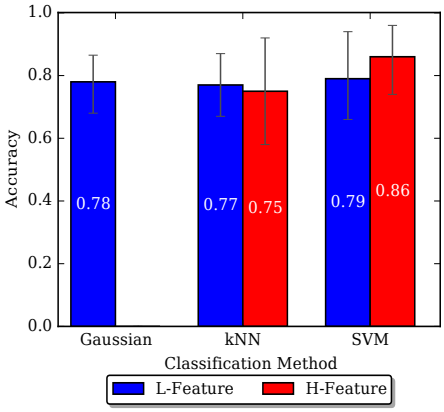

Fig. 5. Accuracy comparison for the T-REAL data set using LOPO cross validation.

As discussed in Sec. II-C, for a comprehensive evaluation of the classification performance that could be expected in real world applications it is important to use the correct cross-validation method. As is shown in Fig. 6, 9-fold random cross-validation would severely overestimate the classification performance.

Finally, Fig. 7 shows that all classifiers lack robustness against moderately large repositioning errors by training them with T-REAL and testing them against T-HARD. Even for the best performing classifier SVM with H-features the accuracy drops down to 70%.

In summary, material classification with a flexible tactile skin is feasible; the high dimensional features include more information which can be used by the SVM classification method; all classifiers are not robust against sensor repositioning.

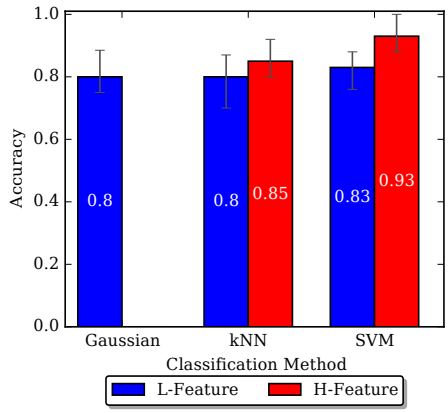

Fig. 6. Accuracy comparison for the T-REAL data set using randomized 9-fold cross validation.

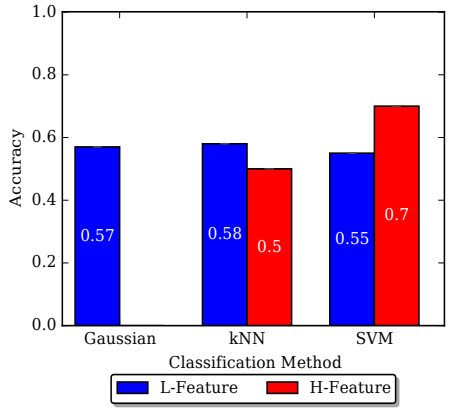

Fig. 7. Accuracy comparison for the T-HARD data set (trained using the T-REAL data set).

IV. Classification with Deep Learning

A. Motivation

As the results from Sec. III-D suggest is it not feasible to construct low dimensional feature vectors from the spatio-temporal sensor signal which robustly exhibit high accuracy in real world settings, ideally, a classification algorithms should be able to work directly on the raw 24000 dimensional spatio-temporal signal without any preprocessing. This way, no information included in the sensor signal is lost before but can in principle be extracted for classification.

An interesting family of such algorithms are deep learning neural networks [24] which have been shown to have superior performance application with high dimensional input like image recognition [24] or speech recognition [25].

In a sense, deep learning not only learns a classifier but also automatically extracts the features from the training data provided [26]. However, a prerequisite for the application of deep learning is to provide a training set that is large enough to allow the extraction of the important class invariants. For our material classification setup with the sensor integrated on the robot this is no principled problem as we can automatically perform the data collection.

B. Architecture

In our tactile signal we expect to see strong temporal correlations with repeating local patterns in a hierarchical

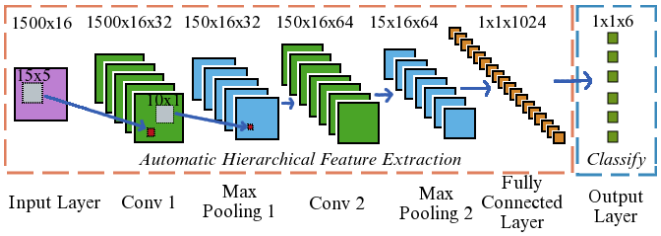

Fig. 8. CNN system architecture.

order of scales, similar to that in image recognition tasks. So we chose a convolutional neural network (CNN).

We reshape the 3D $1500\times4\times4$ spatio-temporal signal into a 2D 1500×16 matrix by flattening the spatial dimension as we do not expect to have significant spatial correlations in the tactile signal. A technical reason is that the used deep learning framework, Google Tensorflow 0.5 [27], did only support (efficient) 2D convolutions, like most openly available deep learning frameworks [5].

The system architecture of our CNN is shown in Fig. 8. The input layer receives the reshaped signal as a 2D matrix. This layer is followed by two convolution layers (CLs) composed of Rectified Linear Unit (ReLU) neurons.

The convolution operation in each of the two CLs is performed using the respective fix sized feature set. The cardinalities of these feature sets are 1×32 and 32×64 respectively[6]. Each feature $\boldsymbol{F}$ is of the shape (M, N). The l^{th} CL performs the following operation:

$$
\begin{aligned}
{}^{l}Y^{k}_{m,n} &= \max\left(0, a + {}^{l}b_k\right), \\
a &= \sum_{k'}^{K_{l'}} \sum_{m'=0}^{M-1} \sum_{n'=0}^{N-1} {}^{l}F^{k',k}_{m',n'} \; {}^{l'}X^{k'}_{m'+m,n'+n}.
\end{aligned}
\tag{6}
$$

The input layer can be considered as the (pseudo) 0^{th} CL with the input signal as the only feature response. K_l is the number of feature responses of the l^{th} CL. Hence, K_0, K_1 and K_2 are 1, 32 and 64 respectively. ${}^{l}Y^{k}_{m,n}$ is a ReLU neuron at the location (m, n) of the k^{th} feature response of the l^{th} CL. ${}^{l}\boldsymbol{F}^{k',k}$ is an element of the set containing $K_{l'} \times K_l$ features where $l' = l - 1$. This set and the K_l-sized bias vector ${}^{l}\boldsymbol{b}$ belong to the l^{th} CL.

Each CL is followed by a max-pooling layer (MPL)

$$
{}^{l}X^{k'}_{m,n} = \max_{m'=0}^{P-1} \max_{n'=0}^{Q-1} {}^{l}Y^{k'}_{m'+Pm,n'+Qn}. \tag{7}
$$

The MPLs work in (P, Q) sized non-overlapping windows. For both convolution and max-pooling zero padding is used to handle under- and over-flow of indices.

The output of the second MPL is flattened into a vector $\boldsymbol{x}$ and fed into a fully connected layer (FCL)

$$
y_i = \max\left(0, {}^{\mathrm{F}}W^i_j \, x_j + {}^{\mathrm{F}}b_i\right) \text{ for } i = 1 \ldots D, \tag{8}
$$

with the weight matrix ${}^{\mathrm{F}}\boldsymbol{W}$ and bias vector ${}^{\mathrm{F}}\boldsymbol{b}$. D is the size of the FCL.

[5]This was true when initially writing the paper, now, e.g., Tensorflow 0.8 supports also 3D convolution.

[6]A more complex architecture with three CLs with sets of 1×64, 64×128 and 128×256 features did not increase the accuracy.

TABLE I

CNN PARAMETERS.

Training Iterations	4000
Dropout, p_d	0.75
Feature size, (M, N)	$(15, 5)$
Max-pool window size, (P, Q)	$(10, 1)$
FCL size (in neurons), D	1024

An usual practice to reduce over-fitting during training is to apply dropout. During dropout with a given probability p_d each y_i is independently scaled up by $1/p_d$ or, with probability $1 - p_d$ set to 0. However, in our case dropout had no significant effect on the networks performance.

The resulting vector is then forwarded to the output layer (OL)

$$
z_i = {}^{\mathrm{O}}W^i_j x_j + {}^{\mathrm{O}}b_i \text{ for } i = 1 \ldots C, \tag{9}
$$

where ${}^{\mathrm{O}}\boldsymbol{W}$ is the weight matrix and ${}^{\mathrm{O}}\boldsymbol{b}$ is the bias vector. During testing, the index i corresponding to the maximum z_i is used as the predicted label.

C. Cost Function and Optimization

For training, a softmax function

$$
s_i(\boldsymbol{z}) = \frac{\mathrm{e}^{z_i}}{\sum_{i'}^{C} \mathrm{e}^{z_{i'}}} \text{ for } i = 1 \ldots C \tag{10}
$$

is applied to convert the unscaled output vector $\boldsymbol{z}$ from the OL into a probability distribution.

The cross-entropy cost is defined as

$$
c(\boldsymbol{Z}) = -\frac{1}{B} \sum_{j}^{B} \sum_{i}^{C} L^j_i \log s_i(\boldsymbol{Z}^j), \tag{11}
$$

with $\boldsymbol{Z}$ as the output of the CNN and $\boldsymbol{L}$ the ground truth labels for a set of B input signals.

The Adam Optimizer [28], a first-order gradient-based optimizer, is used to minimize c.

D. Parameters

A comprehensive search over the parameter space was performed to obtain the parameters in Table I. E.g., to estimate a good value for the number of training iterations, the cost function c was plotted. Fig. 9 shows that c flattens asymptotically to the order of 10^{-4} after 3300 iterations without further improvement.

CNNs for image processing typically use small window sizes (around 5x5 for convolution and 2x2 for max-pooling). However, we note that to extract prominent features from the high frequency tactile sensor data a bigger window (along the temporal direction) was required.

The Adam optimizer was used with a learning rate of 0.0001, an exponential decay rate $\beta_1 = 0.9$ for the 1^{st} moment estimates, an exponential decay rate $\beta_2 = 0.999$ for the 2^{nd} moment estimates, and a numerical stability constant $\epsilon = 10^{-8}$.

Note that batches of 50 input signals collected in a sequential manner from the training set were used for the training iterations. After a complete run through, the training set was randomly shuffled and the process was resumed.

To preserve taxel dimension the max-pooling window was chosen to be of unit length along that dimension.

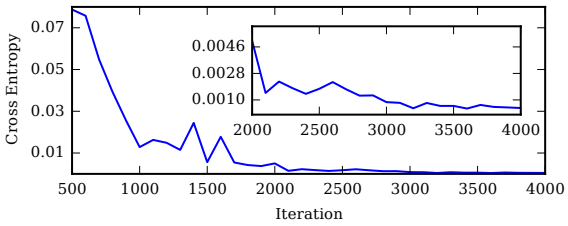

Fig. 9. Cross entropy cost function during training the CNN.

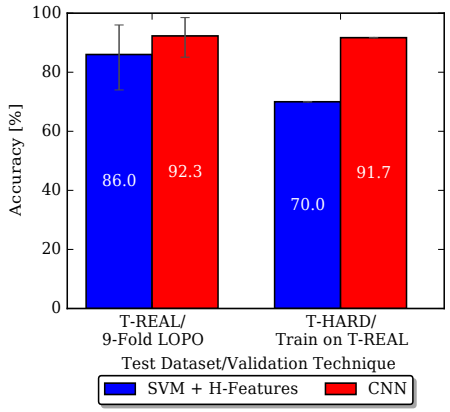

Fig. 10. Accuracy comparison between SVM + H-features and CNN for the different data sets using specified cross validation techniques.

V. Results

A. Accuracy

From Fig. 10 we see that the CNN perform better for both, the T-REAL and T-HARD data sets, especially outperforming the best classifier (SVM using H-Features) significantly achieving 91.7% in the T-HARD dataset.

This high performance of the CNN is surprising at first sight considering the large number of to be trained weights ($\approx 10^7$) versus the small amount of training data (≈ 600). But the results show that all relevant information needed for classification is included in the training data and that it can be extracted without overfitting due to the regularization properties of the stochastic deep learning algorithms in combination with the network architecture.

B. Confusion Matrix

One can gain interesting insights into the network by studying the confusion matrix, C, in Fig. 11. Among the six material classes, *metal*, *wood* and *plastic smooth* have relatively low classification rates with metal being the worst. This can be attributed to the fact that these three materials share two properties - a smooth surface and hard structure. Smooth surface implies low macro vibrations and hard structure directly influences the pressure on the tactile sensors. Note that the *wooden* tube however has some added fine line structures on its surface.

It can also be seen that these three classes are strongly misclassifed into each other that could be once again explained based on the properties they share.

Finally, the matrix is almost symmetric, which implies that each pair of materials is mutually confused in the same way by the network. For example, the highest contribution among

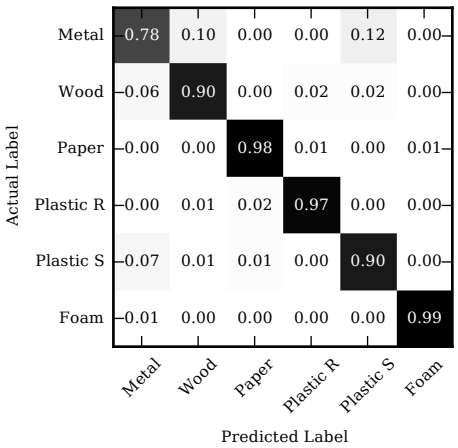

Fig. 11. Confusion matrix for CNN on T-REAL dataset using LOPO cross validation.

the falsely predicted classes for *metal* is from *plastic smooth* and vice versa.

Paper, *plastic rough* and *foam* have clearly higher classification rate for the CNN. *Foam* has the highest classification rate which might be explained by its unique features: highest elasticity and surface structure between *smooth* and *coarse*.

C. Differences from SVM in Classification Pattern

One would derive significantly different observations from the confusion matrix for SVM using H-Features in Fig. 12. Firstly, *metal wood* and *paper* have relatively low classification rates with paper being slightly worse. The tube rolled out of coarse wallpaper has neither a smooth surface nor a hard structure - different from the other two tubes.

Secondly, *metal*, and *wood* are strongly misclassified as *plastic smooth* but the reverse is not true. Apparently, *plastic smooth* is classified better than the other two although they all share the two properties mentioned before. Perhaps, unlike the CNN, the SVM does not depend heavily on texture and hardness for classification.

Lastly, the matrix in this case is not symmetric except while discerning between *paper* and *plastic rough*. Interestingly their tubes have relatively the roughest surfaces among all, one is harder than the other but still deformable with admissible pressure.

D. Multiple Exploration Movements

Using the information from more than one exploration movement, the performance of the classifier can be improved. Multiple exploration movement is simulated by choosing more than one samples per material randomly from the test set. The most occurring predicted class in the group wins. For equally divided classes the one from the first sample wins. We simulate this on the T-REAL dataset using LOPO cross validation (Fig. 13) reaching a accuracy of 97.3% in the case of two sweeps (which would take only 6 s to execute in a real application scenario).

VI. Conclusions and Future Work

We have shown that material classification using only the spatio-temporal signal of a tactile skin is feasible with a high

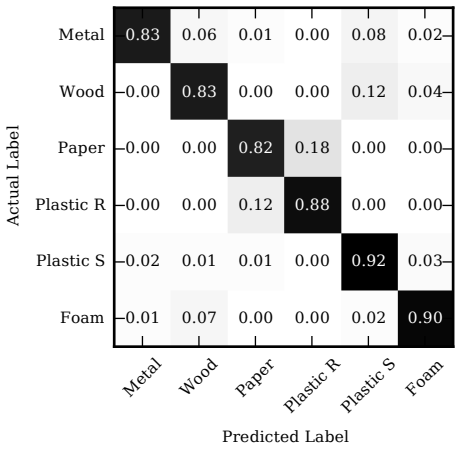

Fig. 12. Confusion matrix for SVM + H-features on T-REAL dataset using LOPO cross validation.

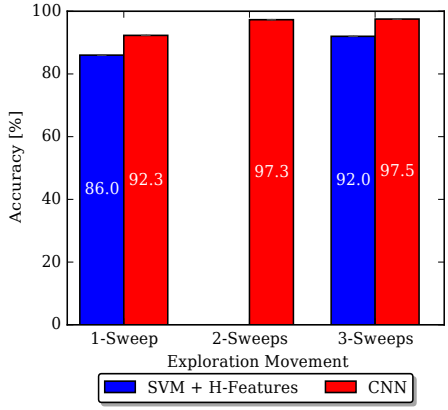

Fig. 13. Accuracy comparison between SVM + H-features and CNN on the T-REAL dataset using LOPO cross validation and different number of exploration sweeps.

accuracy of 97.3% (CNN and two sweeps) in a real world setting.

The robustness of the classifier against repositioning errors of the sensor is essential for using the tactile skin in real applications. We have shown that the robustness increases when increasing the dimensionality of the input data provided to the classification algorithm – if the algorithm is capable of extracting the information. The convolutional deep learning network fed directly with the raw 24000 dimensional spatio-temporal signal performed best by a large margin.

In future work we will apply deep recurrent neural networks to the tactile skin signals as they are esp. suited for high dimensional temporal data and have shown promising results in speech recognition [25]. We will evaluate if adding other sensor modalities which are easily available in our robotic setup, e.g., joint angle and torque sensors, does increase the classification performance. And finally, we will test the new methods in a more challenging task by enlarging the number of materials.

Acknowledgments

We thank FangYi Zhi for collecting the sample data and for the pre-experiments with the tactile skin and DLR's *Agile Justin* hardware team for their indispensable technical support.

References

[1] R. S. Dahiya *et al.*, "Directions toward effective utilization of tactile skin: A review," *IEEE Sensors Journal*, vol. 13, no. 11, 2013.
[2] S. Stassi, V. Cauda, G. Canavese, and C. F. Pirri, "Flexible tactile sensing based on piezoresistive composites: A review," *Sensors*, vol. 14, pp. 5296–5332, 2014.
[3] N. Wettels, V. Santos, R. Johansson, and G. Loeb, "Biomimetic tactile sensor array," *Advanced Robotics*, vol. 22, no. 8, pp. 829–849, 2008.
[4] SynTouch. [Online]. Available: http://www.syntouchllc.com
[5] K. Hosoda, Y. Tada, and M. Asada, "Anthropomorphic robotic soft fingertip with randomly distributed receptors," *Robotics and Autonomous Systems*, 2006.
[6] A. Schmitz *et al.*, "A tactile sensor for the fingertips of the humanoid robot iCub," in *Proc. IEEE International Conference on Intelligent Robots and Systems*, 2010.
[7] Tekscan. [Online]. Available: https://www.tekscan.com
[8] M. Strohmayr, H. Wörn, and G. Hirzinger, "The DLR artificial skin step I: Uniting sensitivity and collision tolerance," in *Proc. IEEE International Conference on Robotics and Automation*, 2013.
[9] G. Büscher *et al.*, "Augmenting curved robot surfaces with soft tactile skin," in *Proc. IEEE International Conference on Intelligent Robots and Systems*, 2015.
[10] P. Mittendorfer and G. Cheng, "Humanoid multimodal tactile-sensing modules," *IEEE Trans. Robot.*, vol. 27, no. 3, pp. 401–410, 2011.
[11] B. Bäuml *et al.*, "Agile Justin: An upgraded member of DLR's family of lightweight and torque controlled humanoids," in *Proc. IEEE International Conference on Robotics and Automation*, 2014.
[12] J. Butterfaß, M. Grebenstein, H. Liu, and G. Hirzinger, "DLR-Hand II: Next generation of a dextrous robot hand," in *Proc. IEEE International Conference on Robotics and Automation*, 2001, pp. 109–114.
[13] J. Fishel and G. Loeb, "Bayesian exploration for intelligent identification of textures," *Frontiers in Neurorobotics*, vol. 6, no. 4, pp. 1–20, 2012.
[14] D. Xu, G. Loeb, and J. Fishel, "Tactile identification of objects using Bayesian exploration," in *Proc. IEEE International Conference on Robotics and Automation*, 2013.
[15] J. Hölscher, J. Peters, and T. Hermans, "Evaluation of tactile feature extraction for interactive object recognition," in *Proc. IEEE-RAS International Conference on Humanoid Robots*, 2015.
[16] J. Sinapov, V. Sukhoy, R. Sahai, and A. Stoytchev, "Vibrotactile recognition and categorization of surfaces by a humanoid robot," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 488–497, 2011.
[17] M. Kaboli, P. Mittendorfer, V. Hugel, and G. Cheng, "Humanoids learn object properties from robust tactile feature descriptors via multi-modal artificial skin," in *Proc. IEEE/RAS International Conference on Humanoid Robots*, 2015.
[18] D. S. Chathuranga *et al.*, "Robust real time material classification algorithm using soft three axis tactile sensor: Evaluation of the algorithm," in *Proc. IEEE International Conference on Intelligent Robots and Systems*, 2015.
[19] A. Schmitz *et al.*, "Tactile object recognition using deep learning and dropout," in *Proc. IEEE-RAS International Conference on Humanoid Robots*, 2014.
[20] M. Madry, L. Bo, D. Kragic, and D. Fox, "ST-HMP: Unsupervised spatio-temporal feature learning for tactile data," in *Proc. IEEE International Conference on Robotics and Automation*, 2014.
[21] T. Hammer and B. Bäuml, "The communication layer of the ardx software framework: Highly performant and realtime deterministic," *Journal of Intelligent and Robotic Systems*, 2015.
[22] L. Cohen, *Time-Frequency Analysis*. Prentice-Hall, 1995.
[23] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
[24] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012.
[25] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82—97, 2012.
[26] Y. Xu *et al.*, "Deep learning of feature representation with multiple instance learning for medical image analysis," in *Proc. IEEE Int. Conf on Acoustics, Speech and Signal Processing*, 2014.
[27] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," Google Research, Tech. Rep., November 2015.
[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conference on Learning Representations*, 2015.