Learning Dynamic Tactile Sensing with Robust Vision-based Training

Oliver Kroemer, Christoph H. Lampert, and Jan Peters

Abstract—**Dynamic tactile sensing is a fundamental ability for recognizing materials and objects. However, while humans are born with partially developed dynamic tactile sensing and master this skill quickly, today's robots remain in their infancy. The development of such a sense requires not only better sensors, but also the right algorithms to deal with these sensors' data. For example, when classifying a material based on touch, the data is noisy, high-dimensional and contains irrelevant signals as well as essential ones. Few classification methods from machine learning can deal with such problems.**

In this paper, we propose an efficient approach to inferring suitable lower-dimensional representations of the tactile data. In order to classify materials based on only the sense of touch, these representations are autonomously discovered using visual information of the surfaces during training. However, accurately pairing vision and tactile samples in real robot applications is a difficult problem. The proposed approach therefore works with weak pairings between the modalities. Experiments show that the resulting approach is very robust and yields significantly higher classification performance based on only dynamic tactile sensing.

I. Introduction

The sense of touch has a fundamental role in most human manipulation tasks, where it serves a variety of purposes. A particularly important type of tactile sensing is *dynamic tactile sensing*. The impressive abilities of this sense are straightforward to observe [1]. For example, when a blindfolded person has an object placed in the palm of their hand, and they do not move their hand nor the object, it is very difficult to recognize the object. The size and weight of the object can be determined, but important properties such as the object's material and precise shape cannot. If one instead slides the object over the skin, one can quickly determine the object and the material [1]. Developing this ability for robots offers many future possibilities.

Dynamic tactile sensing relies on the motion between the skin and the object to induce vibrations and deformations in the skin, which it then uses to infer object and material properties [2]. This type of sensing can be used to determine various properties of a surface, including texture, hardness, roughness, and friction [3], [4]. These properties can be used for tasks such as object identification and determining suitable contact points for grasps.

Dynamic tactile sensing also obtains information about the manipulation task. Vibrations are induced in the finger when it makes or breaks contact with objects, or when incipient slip occurs [5]. These signals help coordinate the fingers, and allow humans to finely regulate the contact forces depending on the object's surface properties [4]. One can also detect the vibrations created when a held object is in contact with another object. Such signals are crucial for dexterously using tools. Humans can even use rigid objects as probes to determine the fine texture of surfaces [6].

The sense of touch should however not be seen in complete isolation, but rather as part of a multimodal system. When recognizing materials and objects, humans often combine touch with vision and even audition [7], [6]. Several studies have shown that the human brain even employs multisensory models of objects [7]. By using such a shared model, humans can transfer knowledge about an object from one sensory modality to another [8]. This sharing of information is especially useful when one sense can not be used. For example, experiments with both vision and touch have shown that humans rely more on touch when the texture has small details that are difficult to see [6]. Dynamic tactile sensing can thus be combined with other senses for more accurate information and additional robustness [9].

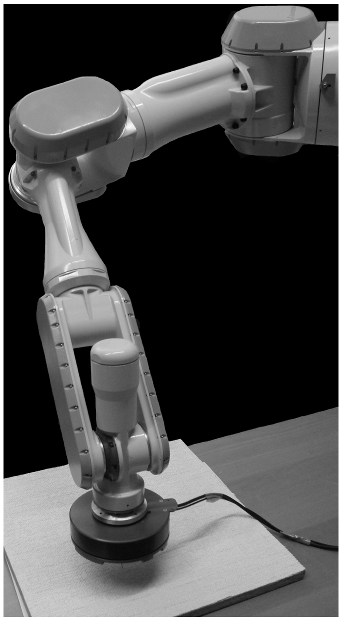


Figure 1. Robot learning about materials by stroking and visually inspecting different surfaces

Given the various benefits of using tactile information in manipulation tasks, there is a considerable interest in equipping robots with such capabilities [10], [11], [12]. The need for robust manipulation skills is especially important for service robots in unstructured environments [13]. A variety of tactile sensors are required to create a complete tactile sensor suite, as discussed in the review paper of Dahiya et al. [14]. As one part of tactile sensing, a dynamic tactile sensor usually only mimics the fast afferent nerves (FA) in human fingers. Human fingers have two types of fast afferent nerves in their

O. Kroemer and J. Peters are with the Max Planck Institute for Biological Cybernetics, Germany. (email: {oliverkro, jan.peters}@tuebingen.mpg.de)

C. H. Lampert is with IST Austria (Institute of Science and Technology Austria), Klosterneuburg. (email: chl@ist.ac.at)

This paper has 12.8MB of downloadable supplementary material available at http://ieeexplore.ieee.org, provided by the authors. The document includes 17 images of the textures used in the experiments, as well as a glossary of machine learning terms.

fingers, i.e., FA-I and FA-II. Type I afferents have a well-localized receptive field and are densely spaced on the skin [15]. Examples of sensors that mimic type I afferents are tactile arrays [12], [16]. Type II afferents have a larger receptive field, and therefore cannot localize the source of the vibrations as well. FA-II afferents are used to sense the vibrations in held objects during manipulation tasks, and are particularly important for tool usage [14]. Due to their large receptive fields, FA-II sensors often struggle to differentiate between various sources of vibrations. Apart from the contact with the object, vibrations also come from other sources [17], [11], such as the robot's own vibrations and deformations of the skin as the finger flexes.

A crucial ability of FA-II nerves is sensing temporal characteristics, such as those involved in recognizing a surface by stroking it. In this paper, we want to reproduce this ability to recognize materials. As a testbed for our proposed algorithms, we have created a basic sensor that represents a primitive technical counterpart to an FA-II type mechanoreceptor. The design is based on a microphone with a probe on its membrane, and was inspired by the work on haptography of Kuchenbecker et al. [18].

The raw time-series data received from the dynamic tactile sensor consists of the detected vibrations. This signal will usually serve as the input for a classifier with task-specific labels. However, classification of tactile data is a difficult task, since a time-series needs to be represented as a high-dimensional data point to capture the details of the signal. Classification in high-dimensional spaces is however prone to *overfitting*, due to "the curse of dimensionality" [19]. The overfitting results in the classifier often performing poorly when applied to new data. This problem can be addressed using *dimensionality reduction* approaches which project the data into lower-dimensional feature spaces. The goal is to discard information that is not relevant, such as noise or redundant information.

As previously discussed, additional sources of vibrations are often present in the signal together with the desired tactile signal. For good performance, the classifier needs to automatically determine the relevant parts of the signal. We therefore take a human-inspired approach and transfer knowledge from the vision modality.

In this paper, we present approaches for combining vision and tactile information to improve the performance of dynamic tactile sensors. The focus of this paper is on service robots that need to perform assorted tasks. However, the proposed approach is applicable to a wide range of robots with hand-eye systems. The proposed approach is based on *Maximum Covariance Analysis (MCA)* [20], which is a machine learning method for dimensionality reduction using sets of paired data. The MCA method is described in Section II-B. However, MCA requires perfect pairings between tactile and visual samples, which is often a problem for robot systems in unstructured environments [21], [22]. We therefore propose *Mean Maximum Covariance Analysis (μMCA)* and using *Weakly-paired Maximum Covariance Analysis (WMCA)* for robotic applications. These methods are more robust and only require weak pairings between the modalities. After learning, the tactile sensor can be used independently of the vision system, while retaining its improved performance. Thus, the resulting system can be used even when conditions are not suitable for visual inspection, e.g., dim lighting, occluded surfaces, perspective distortion, and even damaged cameras.

Our initial work and evaluations of the WMCA algorithm were presented in [23]. The novel contributions of this paper include the μMCA method and a more robust implementation of WMCA based on concepts from deterministic annealing [24]. These methods are presented in Section III and compared through a series of benchmarking experiments in Section IV. The experiments show that the proposed methods are robust and allow the robot to accurately discriminate between materials by only stroking them.

II. Formalization in a Multimodal Dimensionality Reduction Setting

In this section, we formulate the problem in a machine learning framework (Section II-A) and give a brief review of multimodal dimensionality reduction methods (Section II-B).

A. Problem Statement

Our goal is to have a robot accurately discriminate between different surfaces by only stroking them. We initially allow the robot to learn about textures by both stroking and visually inspecting them. The robot should subsequently transfer the additional visual information to improve its knowledge of tactile sensing. As a result, the tactile sensor's independent performance should also improve.

We now repose the problem in a general machine learning framework. The problem involves reducing the dimensionality of a sensor's data such that the relevant tactile information is retained. Not all dimensionality reduction methods are suitable for our robot application. We must therefore first select an appropriate type of method.

Dimensionality reduction algorithms are either inductive or non-inductive. Inductive methods create a function f that can map the data $\mathbf{X}$ onto a lower dimensional representation $\hat{\mathbf{X}}$. Inductive methods include *PCA* [25], *kernelPCA* [26] and *autoencoder networks* [27]. Non-inductive methods, such as *probabilistic latent semantic analysis (pLSA)* [28], and *Isomap* [29], also compute a lower-dimensional representation $\hat{X}$ from X, but do not provide a mapping function f.

Robots continue to collect more data as they explore their, often changing, environments. The mapping function f of inductive methods can be used to reduce the dimensionality of the sensor's data as it is received. We therefore require an inductive method.

Definition 1 (Inductive Dimensionality Reduction) *Let $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n) \subset \mathbb{R}^{d\times n}$ be a set of data vectors. Inductive dimensionality reduction procedures take the input $\mathbf{X}$, and output a functional mapping $f : \mathbb{R}^d \to \mathbb{R}^q$ with $q < d$. The lower dimensional representation of $\mathbf{X}$ is given by $\hat{\mathbf{X}} = (\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_n)$, i.e., $\hat{\mathbf{x}}_i = f(\mathbf{x}_i)$.*

We can further divide inductive dimensionality reduction techniques into discriminative and generative methods. Discriminative techniques, such as *linear discriminant analysis*

(LDA) [30] and *canonical correlation analysis (CCA)* [31], [32], identify lower-dimensional representations that are suitable for one specific task, e.g. classification into a predefined set of classes. These techniques discard all information that is irrelevant for this particular task. While the new representations $\hat{\mathbf{X}}$ are very good for this task, they tend to be unsuitable for other tasks. In contrast, generative dimensionality reduction techniques find lower-dimensional data representations that are suited for various subsequent tasks. Intuitively, generative dimensionality reduction techniques are a form of lossy data compression methods.

Service robots will face a large range of tasks, which makes it difficult to predefine a set of suitable labels. The robots will also encounter new objects and materials as they explore their unstructured environments. If the robot discards information based only on its current set of labels, it may discard information pertinent to new materials and objects. We therefore focus on generative methods.

Having decided on using generative inductive methods, we must determine how to transfer the visual information into the tactile domain. The key to combining visual and tactile information is that both contain spatial data, such as texture, about objects and materials [7]. The senses of vision and touch are otherwise very distinct, and thus the additional sources of vibrations and noise in the tactile modality will be excluded from the visual data. We can therefore use the visual information to determine which parts of the tactile signal are relevant to the textured surface.

Audio signals can also be used to distinguish between textured surfaces [6]. Therefore, an alternative approach would be to combine the tactile sensing with hearing. However, a robot's audio sensors may also detect other vibrations, such as those from the robot's motors. These vibrations would then be present in both sensing modalities, and would therefore be incorrectly regarded as relevant for tactile sensing. To avoid this error, we use vision as our second sensor modality.

In order to automatically extract the relevant information from the vision data, we make use of *multimodal dimensionality reduction*. The general goal of multimodal dimensionality reduction is to compute new representations of the high-dimensional data samples that lie in lower-dimensional feature spaces. In comparison to unimodal dimensionality reduction, we expect the availability of multiple data representations to give a better indication of the relevant parts of the signal, and which parts can be suppressed. We formalize this concept in the following definition.

Definition 2 (Multimodal Dimensionality Reduction) *Let* $\mathbf{X}^1 = (\mathbf{x}^1_1, \ldots, \mathbf{x}^1_{n^1}) \subset \mathbb{R}^{d^1 \times n^1}, \ldots, \mathbf{X}^m = (\mathbf{x}^m_1, \ldots, \mathbf{x}^m_{n^m}) \subset \mathbb{R}^{d^m \times n^m}$ *be* m *different data sets from potentially different spaces. Inductive dimensionality reduction techniques are multimodal if they take inputs* $\mathbf{X}^1, \ldots, \mathbf{X}^m$, *and output functions* $f_1 : \mathbb{R}^{d^1} \to \mathbb{R}^q, \ldots, f_m : \mathbb{R}^{d^m} \to \mathbb{R}^q$ *for all data domains.*

Each of the m different modalities must have its own independent mapping function f based only on the modality's own data. This part of the definition is crucial, as it will allow the tactile sensor to be used on its own. Thus, if the robot is in a dark room or cannot position the object to allow for visual inspection, the robot can still use the transferred visual information for improved tactile sensing.

The canonical way to construct multimodal algorithms is to use the dependencies between *paired* samples. Two samples are strongly paired if their sensors acquired them from the same source. For example, consider a tactile sensor moving a short distance across a textured surface. The tactile reading acquired during this motion would be strongly paired with an image of the surface area swept by the tactile sensor. Acquiring perfectly paired samples across modalities is often problematic in practice, especially in unstructured environments. Any inaccuracies in moving the object or the cameras for visual inspection will result in incorrect pairings. The different sensors may also have different numbers of samples that need to be paired. For example, while cameras can quickly acquire data from large surface areas, tactile sensors obtain information from their relatively small contact region with the surface. We therefore only assume weakly-paired data [23].

Definition 3 (Weakly-Paired Multimodal Data) *A collection of data sets* $\mathbf{X}^1, \ldots, \mathbf{X}^m$ *is weakly paired, if each* $\mathbf{X}^i$ *is split into* g *groups as*

$$\mathbf{X}^i = (\mathbf{X}^i_1, \ldots, \mathbf{X}^i_g) \in \mathbb{R}^{d^i \times n^i},$$

where each group of samples is given by

$$\mathbf{X}^i_h = (\mathbf{x}^i_{h,1}, \ldots, \mathbf{x}^i_{h,n^i_h}) \in \mathbb{R}^{d^i \times n^i_h},$$

with $n^i = \sum_{l=1}^{g} n^i_l$. *When* $n^i_l = 1$ *for all* $i = 1, \ldots, m$ *and* $l = 1, \ldots, g$ *the data sets are fully paired with strong pairings. When* $g = 1$, *all samples are weakly paired together, which means that they are all unpaired.*

A weak pairing implies that a group of samples from one modality is paired to a group of samples in another modality. While strong pairings require samples to be obtained from the same source, weak pairings only require the samples to be acquired from similar sources. Hence, the robot can acquire samples from various regions of a textured surface and group these together. Alternatively, a robot could weakly pair one tactile sensor reading to multiple images of the nearby surface. In both of these examples, the samples can subsequently be used to infer suitable strongly-paired data. Ultimately, the condition of weakly-paired data is a relaxation of the standard fully-paired requirement, and is therefore easier for robots to fulfill.

The samples used for learning the dimensionality reductions should be acquired under conditions suitable for both visual inspection as well as tactile sensing. The conditions for visual inspection can be ignored only after the mapping functions have been learned.

Although our focus is on combining visual and tactile information, the described problem framework is quite common in robotics. The algorithms described in this paper were therefore designed to work with weak pairings between a variety of sensors. However, different mapping functions are obtained for a sensor when it is combined with different types of sensors.

```
MAXIMUM COVARIANCE ANALYSIS
INPUT:
    Data covariance matrix $\mathbf{X}\mathbf{X'}^T \in \mathbb{R}^{d \times d'}$
    Desired output dimensionality $q$
COMPUTE MAPPINGS:
    Compute Singular Value Decomposition of $\bar{\mathbf{X}}\bar{\mathbf{X}}'^T$
        $\mathbf{U}\mathbf{S}\mathbf{V}^T = \mathrm{svd}(\bar{\mathbf{X}}\bar{\mathbf{X}}'^T)$ where $\mathbf{U} \in \mathbb{R}^{d \times d}$, $\mathbf{V} \in \mathbb{R}^{d' \times d'}$
    Find $q$ largest elements in $\mathbf{S} \in \mathbb{R}^{d \times d'}$
        Set $\mathbf{W}$ to corresponding $q$ columns of $\mathbf{U}$
        Set $\mathbf{W}'$ to corresponding $q$ columns of $\mathbf{V}$
OUTPUT:
    Projection matrices $\mathbf{W}$ and $\mathbf{W}'$
```

Figure 2. Implementation of MCA algorithm

The features regarded as relevant are those that both sensors observe of the source, and any features found only in one of the modalities will usually be suppressed.

B. Introduction to Multimodal Dimensionality Reduction

This section gives a brief review of linear multimodal dimensionality reduction methods, including MCA. To simplify the notation, we restrict the discussion to two sensor modalities, i.e., $\mathbf{X} \in \mathbb{R}^{d \times n}$ and $\mathbf{X}' \in \mathbb{R}^{d' \times n'}$.

Linear dimensionality reduction functions can be written as $f(\mathbf{x}) = \mathbf{W}^T\mathbf{x}$ for a matrix $\mathbf{W} \in \mathbb{R}^{d \times q}$, and $f'(\mathbf{x}') = \mathbf{W}'^T\mathbf{x}'$ for a matrix $\mathbf{W}' \in \mathbb{R}^{d' \times q}$. The lower dimensional representations are thus $\hat{\mathbf{X}} = \mathbf{W}^T\mathbf{X}$ and $\hat{\mathbf{X}}' = \mathbf{W}'^T\mathbf{X}'$. The orthogonal matrices $\mathbf{W}$ and $\mathbf{W}'$ contain the basis vectors of the q-dimensional subspaces.

A popular generative dimensionality reduction technique is *principal component analysis (PCA)*. PCA finds a lower-dimensional representation that retains as much of the original signal's variance as possible. Given that other sources of vibrations may also have large variances, PCA is not a suitable approach for our purposes. The multimodal counterpart to PCA is *maximum covariance analysis (MCA)* [20].

MCA assumes that the data is fully paired, i.e., for every sample in $\mathbf{X}$ there is exactly one strongly paired sample in $\mathbf{X}'$. The data sets $\mathbf{X}$ and $\mathbf{X}'$ are centered by subtracting their means from all of their samples. MCA then optimizes the objective function $\max_{\mathbf{W},\mathbf{W}'} \mathrm{tr}\left[\mathbf{W}^T\mathbf{X}\mathbf{X}'^T\mathbf{W}'\right]$, where $\mathrm{tr}[.]$ is the standard *matrix trace operator*, to determine suitable projection matrices $\mathbf{W}$ and $\mathbf{W}'$. The objective function can be rewritten with $\mathrm{tr}\left[\mathbf{W}^T\mathbf{X}\mathbf{X}'^T\mathbf{W}'\right] = \sum_{p=1}^{q}\left[\mathbf{W}^T\mathbf{X}\right]_p^T\left[\mathbf{W}'^T\mathbf{X}'\right]_p$, where the operator $[.]_p$ extracts the pth column of the matrix, and $q \leq n$. Thus MCA maximizes the covariances between the low dimensional representations $\hat{\mathbf{X}}$ and $\hat{\mathbf{X}}'$. The standard MCA method requires strong one-to-one pairings between the modalities, and therefore $n = n'$. An implementation of MCA is given in Fig. 2.

MCA comes from the same family of standard statistical methods as PCA, LDA, and CCA. It also forms the basis for *partial least squares (PLS)* regression [33]. The PCA, LDA, CCA, and PLS techniques have all been kernelized into nonlinear versions [34], [35], [26]. The methods presented in this paper can also be kernelized (Section III-C). *Kernel canonical correlation analysis (kernelCCA)* [36] is amongst the most common methods for multimodal dimensionality reduction, but it is not generative. Furthermore, kernelCCA requires the tuning of a regularization parameter for each modality. Alternative approaches include *multimodal pLSA* [37] and *Hilbert-Schmidt dependence maximization* [38], but these require more careful experimental setups and are computationally more demanding. In contrast, the classical methods, and our proposed methods, can be implemented with standard matrix operations.

Even though MCA is a strong method for multimodal dimensionality reduction, robots in unstructured scenarios often cannot provide the required fully-paired data. In the following section, we show how to overcome this limitation, and make use of weakly-paired data.

III. MAXIMUM COVARIANCE ANALYSIS ALGORITHMS FOR MULTIPLE ROBOT SENSOR MODALITIES

In this section, we explain μMCA and WMCA for robot applications. These methods incorporate vision information to create an improved representation of the tactile data. *Sensor fusion* is another process that combines data from multiple sensors to improve performance and the accuracy of measurements [39], [9]. The data from sensors can be combined directly using *data fusion,* or classified separately and then combined with *classifier fusion* [40]. These approaches rely on always having access to both sensor modalities, while the methods proposed in this section only require both modalities during the learning phase. After learning with the proposed methods, the sensors can be used independently. Hence, tactile sensing performance is improved even when the conditions are unsuitable for visual inspection, or when the camera is currently allocated to performing another task. A fundamental problem of combining tactile and vision data is self-occlusion; i.e., the hand used for tactile sensing blocks visual inspection. The proposed methods are well-suited for such situations.

Self-supervised learning is another framework that only requires both sensor modalities during the learning phase. In self-supervised learning, the robot uses one modality to generate the labels for the classification problem of another sensor modality [41], [42]. A large amount of information from the supervising modality is lost during these procedures, as the data is reduced to a single value. The methods proposed in this section use the entire signal of both sensors to improve the classification performance. In this manner, the proposed methods can share information between different materials at the level of individual features.

Self-supervised methods are sensitive to errors in the pairings between modalities [21], [22]. The μMCA and WMCA methods overcome this problem by automatically inferring strong pairings from the weakly-paired groups. The lower dimensional representations found by self-supervised methods are usually only suited for the task they were trained on [41].

In the remainder of this section, we present the proposed μMCA (Section III-A) and a robust implementation of WMCA (Section III-B) for robotic applications, as well as extensions to nonlinear problems (Section III-C) and multiple sensor modalities (Section III-D). We present straightforward algorithms for

```
MEAN MAXIMUM COVARIANCE ANALYSIS
INPUT:
    Weakly-paired data from sensors one $\mathbf{X}$ and two $\mathbf{X}'$
        $\mathbf{X}$ has $n_h$ samples $\mathbf{x}_{h,1\ldots n_h}$ in group $h = 1 \ldots g$
        $\mathbf{X}'$ has $n'_h$ samples $\mathbf{x}'_{h,1\ldots n'_h}$ in group $h = 1 \ldots g$
    Desired output dimensionality $q \leq \min(\{g, d, d'\})$
INITIALIZATION:
    $\bar{\mathbf{X}} = (\bar{\mathbf{x}}_1, \ldots, \bar{\mathbf{x}}_g) \subset \mathbb{R}^{d \times g}$ with means $\bar{\mathbf{x}}_{1\ldots g} = 0$
    $\bar{\mathbf{X}}' = (\bar{\mathbf{x}}'_1, \ldots, \bar{\mathbf{x}}'_g) \subset \mathbb{R}^{d' \times g}$ with means $\bar{\mathbf{x}}'_{1\ldots g} = 0$
COMPUTE MAPPINGS:
    for $h = 1$ to $g$
        for $i = 1$ to $n_h$
            Update $\bar{\mathbf{x}}_h \Rightarrow \bar{\mathbf{x}}_h + (\mathbf{x}_{h,i} - \bar{\mathbf{x}}_h)(i+1)^{-1}$
        for $i = 1$ to $n'_h$
            Update $\bar{\mathbf{x}}'_h \Rightarrow \bar{\mathbf{x}}'_h + (\mathbf{x}'_{h,i} - \bar{\mathbf{x}}'_h)(i+1)^{-1}$
    Obtain $\mathbf{W}$ and $\mathbf{W}'$ from MCA($\bar{\mathbf{X}}\bar{\mathbf{X}}'^T$, $q$)
OUTPUT:
    Projection matrices $\mathbf{W}$ and $\mathbf{W}'$
```

Figure 3. Implementation of μMCA algorithm

both μMCA and WMCA to guide the reader through using these methods. These algorithms can be implemented with standard matrix toolboxes.

A. Mean Maximum Covariance Analysis (μMCA)

When using different types of sensors, it is common to obtain different numbers of samples from them. For example, vision sensors can easily obtain information about large parts of a surface, while tactile sensors are limited to the regions they make contact with. Thus, there will usually be many visual samples weakly-paired to a few tactile samples. Rather than selecting a single visual sample for each tactile sample, μMCA combines the information from all of these samples.

The μMCA method assumes that each of the g groups, as specified in Definition 3, represents a series of observations of the same surface. The variations within each group can then be modeled as a standard Gaussian model, i.e., $\mathbf{x}_{i,j} \sim N(\bar{\mathbf{x}}_i, (\boldsymbol{\sigma}_i)^2)$ and $\mathbf{x}'_{i,j} \sim N(\bar{\mathbf{x}}'_i, (\boldsymbol{\sigma}'_i)^2)$. The mean values $\bar{\mathbf{x}}_i \in \mathbb{R}^d$ and $\bar{\mathbf{x}}'_i \in \mathbb{R}^{d'}$ are thus suitable representations of the ith surface group, and can be strongly paired together.

Service robots should generally be autonomous and automatically gather the information they require. We therefore assume that additional prior information is not available. Given a set of collected samples, the robot should fit a model of the surface that best represents this data. We therefore propose a *maximum likelihood* estimation to determine the values of $\bar{\mathbf{x}}_i$ and $\bar{\mathbf{x}}'_i$ that best represent the collected samples.

Given the centered and weakly-paired data $\mathbf{X}$ and $\mathbf{X}'$, the μMCA method solves

$$\max\nolimits_{\mathbf{W},\mathbf{W}'} \operatorname{tr}\left[\mathbf{W}^T \bar{\mathbf{X}}\bar{\mathbf{X}}'^T \mathbf{W}'\right], \tag{1}$$

where $\bar{\mathbf{X}} = (\bar{\mathbf{x}}_1, \ldots, \bar{\mathbf{x}}_g) \subset \mathbb{R}^{d \times g}$ with group means $\bar{\mathbf{x}}_h = (n_h)^{-1} \sum_{j=1}^{n_h} \mathbf{x}_{h,j}$, and $\bar{\mathbf{X}}' = (\bar{\mathbf{x}}'_1, \ldots, \bar{\mathbf{x}}'_g) \subset \mathbb{R}^{d' \times g}$ with group means $\bar{\mathbf{x}}'_h = (n'_h)^{-1} \sum_{j=1}^{n'_h} \mathbf{x}'_{i,j}$. This problem can be solved using the μMCA algorithm shown in Fig. 3. When q is small, the singular value decomposition can be efficiently computed using techniques based on random projections [43]. Intuitively, μMCA uses the groups of samples to estimate archetypes that are more representative of the surface than any one sample. Since the rank of the $\tilde{\mathbf{X}}\tilde{\mathbf{X}}'^T$ matrix is limited by the number of groups g, the output dimensionality is limited to $q \leq g$. The μMCA algorithm has a computational complexity of $\mathcal{O}(g^3)$.

The sequential updates of the group means in Fig. 3 allows new data to be easily incorporated. Hence, the memory requirements of μMCA depend on the number of groups and not the number of samples. The μMCA approach is therefore suitable for large amounts of data.

B. Weakly-Paired Maximum Covariance Analysis (WMCA)

While μMCA combined samples into more informative representations, WMCA's approach is to infer strong pairings between individual samples in a weakly-paired group. Inferring strong pairings is done by including a $n \times n'$ pairing matrix $\mathbf{\Pi}$. The elements of the pairing matrix are either one or zero $\mathbf{\Pi} \in \{0,1\}^{n \times n'}$. A one in the ith row and the jth column implies a pairing between the ith sample of the first modality and the jth sample of the second modality. Each sample is only paired to at most one sample in the other modality, i.e., $\sum_{i=1}^{n} \mathbf{\Pi}_{i,j} \leq 1$ for all $j = 1, \ldots, n'$ and $\sum_{j=1}^{n'} \mathbf{\Pi}_{i,j} \leq 1$ for all $i = 1, \ldots, n$. Assuming that the samples are ordered according to their weakly-paired groups, the pairing matrix will have a block diagonal structure $\mathbf{\Pi} = \operatorname{diag}(\mathbf{\Pi}^1, \ldots, \mathbf{\Pi}^g)$. This structure ensures that samples are only paired within their own group.

Given the described pairing matrix, WMCA optimizes

$$\max\nolimits_{\mathbf{W},\mathbf{W}',\mathbf{\Pi}} \operatorname{tr}\left[\mathbf{W}^T \mathbf{X}\mathbf{\Pi}\mathbf{X}'^T \mathbf{W}'\right], \tag{2}$$

to determine projection matrices $\mathbf{W}$ and $\mathbf{W}'$, where the trace operator $\operatorname{tr}[.]$ sums the diagonal elements of the matrix. The optimization of (2) requires both continuous optimization for $\mathbf{W}$ and $\mathbf{W}'$, and combinatoric optimization for $\mathbf{\Pi}$. There is therefore no single closed form solution to this optimization. Furthermore, it is a high-dimensional non-convex problem, such that finding the global optimum with a numeric procedure is usually impossible. We can, however, efficiently find a locally optimal solution by *alternating maximization*, as shown in Fig. 4. Step one can be efficiently solved using the same singular value decomposition methods used for μMCA. To efficiently solve the linear assignment problem in step two, we suggest using the Hungarian algorithm [44] or LAPJV [45]. In this manner, we can apply WMCA to data with thousands of dimensions. The computational complexity of WMCA is given by $\mathcal{O}(\min(\{nn'^2, n^2n'\}))$.

In both steps of the algorithm, we maximize the same objective function, which will thus increase monotonically with the number of iterations. Given that the objective function has an upper bound, the algorithm is guaranteed to converge to a local maximum. Unfortunately, the objective function will often have multiple local maxima. Hence, WMCA may converge to a local maximum with a relatively low covariance. In order to avoid many local maxima of poor quality, we

```
WEAKLY-PAIRED MAXIMUM COVARIANCE ANALYSIS
INPUT:
    Weakly-paired data from sensors one $\mathbf{X}$ and two $\mathbf{X}'$
    Desired output dimensionality $q \leq \min(\{n, n', d, d'\})$
INITIALIZATION:
    $\eta = 1$
    $\hat{\mathbf{\Pi}} = \mathrm{diag}(\hat{\mathbf{\Pi}}^1, \ldots, \hat{\mathbf{\Pi}}^g)$ and $\mathbf{\Pi} \rightarrow \hat{\mathbf{\Pi}}$ wherein
    $[\hat{\mathbf{\Pi}}^h]_{i,j} = \min(n_h, n'_h)^{-1} \forall i = 1, \ldots, n_h, j = 1, \ldots, n'_h$
ANNEALING WMCA:
    while $\eta \geq 0$
        Run Alternating Maximization
        Reduce $\eta$
ALTERNATING MAXIMIZATION:
    while trace value of $\mathbf{W}^t\mathbf{X}\mathbf{\Pi}\mathbf{X}'^t\mathbf{W}'$ increases
        Step 1) Maximize with respect to $\mathbf{W}$ and $\mathbf{W}'$:
            Obtain $\mathbf{W}$ and $\mathbf{W}'$ from MCA$(\mathbf{X}\mathbf{\Pi}\mathbf{X}'^T, q)$
        Step 2) Maximize with respect to $\mathbf{\Pi}$:
            Set all elements of $\mathbf{\Pi}$ to zero
            for $h = 1$ to $g$
                Compute the cost matrix $\mathbf{C} = [\mathbf{X}'^t_h\mathbf{W}'\mathbf{W}^t\mathbf{X}_h]^t$
                Solve linear assignment problem for $\mathbf{C}$
                Set elements of $\mathbf{\Pi}$ to 1 for assigned pairings
        Anneal) Relax pairings:
            $\mathbf{\Pi} \rightarrow \eta\hat{\mathbf{\Pi}} + (1 - \eta)\mathbf{\Pi}$
OUTPUT:
    Projection matrices $\mathbf{W}$ and $\mathbf{W}'$
```

Figure 4. Implementation of WMCA with annealing

propose incorporating concepts from *deterministic annealing* [24].

The annealing process for WMCA is shown in Fig. 4. The annealing introduces the mean pairing matrix $\hat{\mathbf{\Pi}}$, which pairs together the groups' means. The pairing matrix $\mathbf{\Pi}$ is a mix between the assignments found in step 2 and this mean pairing matrix $\hat{\mathbf{\Pi}}$. The mixing is controlled by parameter η, which is initially set to one and monotonically decreases to zero.

Intuitively, a larger value for the parameter η makes the data points within each group more correlated. When $\eta = 1$, all of the data points are effectively equal to their respective group's mean. Applying the alternating maximization results in the globally optimal $\mathbf{W}$ and $\mathbf{W}'$ when $\eta = 1$. The manner in which η decreases is known as the *cooling schedule*. The additional local maxima gradually emerge as η decreases. Since the results of each maximization are used to initialize the next one, the alternating maximization continuous to track the best local maximum as η decreases. When $\eta = 0$, the true objective function is recovered. The annealing does not guarantee that the global maximum is recovered. However, the annealing process is a systematic and efficient approach to avoiding many poor local maxima.

The idea of treating unknown correspondences as latent variables and optimizing over them has been used in previous applications, including the classical k-means [46] algorithm and the optimization in [38]. However, in both of these cases the assignments are between sample and clusters, not between samples in different data modalities.

```
PROCESSING OF NEW TACTILE DATA
Input:
    Tactile sensor data $\mathbf{Y}$
    Labels $\mathbf{L}$ of training data OR the number of clusters $c$
Learning:
    Determine $\mathbf{W}$ with WMCA or $\mu$MCA
Processing:
    Project $\mathbf{Y}$ using $\hat{\mathbf{Y}} = \mathbf{W}^t\mathbf{Y}$
    If labels $\mathbf{L}$ are given, supervised learning:
        Sort $\hat{\mathbf{Y}}$ with labels into $\hat{\mathbf{Y}}_{train}$, and rest into $\hat{\mathbf{Y}}_{test}$
        Train Nearest Neighbor classifier with $\mathbf{L}$ and $\hat{\mathbf{Y}}_{train}$
        Apply classifier to $\hat{\mathbf{Y}}_{test}$
    Else, unsupervised learning:
        apply $k$-means clustering with $c$ clusters
Output:
    Labels for $\hat{\mathbf{Y}}_{test}$ OR cluster assignments for $\hat{\mathbf{Y}}$
```

Figure 5. Example method for applying learned mappings to new data

Given the projection matrices $\mathbf{W}$ and $\mathbf{W}'$ from either μMCA or WMCA, we apply them to new tactile data, as suggested in Fig. 5.

C. Kernelization for Nonlinear Problems

Nonlinear dimensionality reduction techniques are often more powerful than linear ones, as they can create more diverse dimensionality reduction functions. μMCA and WMCA can be made into nonlinear techniques by *kernelization*, and thus applied to problems in robotics that cannot be solved using linear representations. As the necessary steps are very similar to those for deriving kernelPCA [47] from PCA, we only outline them here. We refer the reader to [26] for a more detailed description of kernelization.

For kernelization, we require positive definite and symmetric similarity measures between samples, called kernel functions, that we denote by $k : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ and $k' : \mathbb{R}^{d'} \times \mathbb{R}^{d'} \rightarrow \mathbb{R}$. Any such kernel function corresponds to an inner product in a latent Hilbert space, and induces a latent feature map from the original data domain to this space [26]. The kernelized methods thus consist of mapping the input data into the latent Hilbert spaces and performing the corresponding linear method on the resulting data sets.

For example, the kernelized form of (2) becomes

$$\max_{\mathbf{A},\mathbf{A}',\mathbf{\Pi}} \ \mathrm{tr}\left[\mathbf{A}\bar{\mathbf{K}}\mathbf{\Pi}\bar{\mathbf{K}}'\mathbf{A}'^T\right], \tag{3}$$

where $\bar{\mathbf{K}}$ and $\bar{\mathbf{K}}'$ are the centered kernel matrices. $\bar{\mathbf{K}}$ is computed by forming the kernel matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$ as $[\mathbf{K}]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ and then centering it using the formula $\bar{\mathbf{K}} = \mathbf{K} - \frac{1}{n}\mathbf{1}_n\mathbf{K} - \frac{1}{n}\mathbf{K}\mathbf{1}_n + \frac{1}{n^2}\mathbf{1}_n\mathbf{K}\mathbf{1}_n$, where $\mathbf{1}_n$ denotes the $n \times n$ matrix in which all elements are 1. $\bar{\mathbf{K}}'$ is computed from kernel k' in the analogous way. Centering the kernels ensures that the implicitly defined feature vectors have zero mean in the latent feature space. One can solve (3) with an alternating optimization similar to the one described in Section III-B. In contrast to $\mathbf{W}, \mathbf{W}'$, the matrices $\mathbf{A} \in \mathbb{R}^{n \times q}$ and $\mathbf{A}' \in \mathbb{R}^{n' \times q}$ are not orthogonal matrices, but are orthogonal in the latent feature space, i.e., $\mathbf{A}^T\mathbf{K}\mathbf{A} = \mathbf{I}$ and $\mathbf{A}'^T\mathbf{K}'\mathbf{A}' = \mathbf{I}$, where $\mathbf{I}$

is the *identity matrix* of size $q \times q$. We obtain the rows of $\mathbf{A}$ and $\mathbf{A}'$ from a generalized eigenvalue problem:

$$\begin{pmatrix} 0 & \mathbf{K\Pi K'} \\ \mathbf{K'\Pi}^t\mathbf{K} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{a} \\ \mathbf{a'} \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{K} & 0 \\ 0 & \mathbf{K'} \end{pmatrix} \begin{pmatrix} \mathbf{a} \\ \mathbf{a'} \end{pmatrix}. \qquad (4)$$

Equation (4) can be efficiently solved for q eigenvectors using the *power method* [48]. Ultimately, the kernelized methods provide reduction functions $f : \mathbb{R}^d \to \mathbb{R}^q$ and $f' : \mathbb{R}^{d'} \to \mathbb{R}^q$ by setting $f(\mathbf{x}) = \mathbf{A}^T\mathbf{K}(\mathbf{x})$ with $\mathbf{K}(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_1), \ldots, k(\mathbf{x}, \mathbf{x}_n))^T$ and $f'(\mathbf{x}') = \mathbf{A}'^T\mathbf{K}'(\mathbf{x}')$ with $\mathbf{K}'(\mathbf{x}') = (k'(\mathbf{x}', \mathbf{x}'_1), \ldots, k'(\mathbf{x}', \mathbf{x}_{n'}))^T$.

Kernelization usually requires more computation time, but can also reduce them in certain situations. When solving for $\mathbf{A}$ and $\mathbf{A}'$, the matrix $\mathbf{K\Pi K}$ is of size $n \times n'$ instead of $d \times d'$. Thus, if the number of samples is less than the input dimensionalities, the computation is faster in the kernelized form. To perform the optimization, one uses linear kernels $k(\mathbf{x}, \tilde{\mathbf{x}}) = \mathbf{x}^T\tilde{\mathbf{x}}$ and $k'(\mathbf{x}', \tilde{\mathbf{x}}') = \mathbf{x}'^T\tilde{\mathbf{x}}'$ and obtains the linear solutions as $\mathbf{W} = \mathbf{A}^T\mathbf{X}$ and $\mathbf{W}' = \mathbf{A}'^T\mathbf{X}'$.

D. Incorporating Additional Sensor Modalities

To keep the notation simple, we have been describing μMCA and WMCA for only two sensor modalities. An extension to more than two data sources is straightforward by reformulating the objective function as the sum of all pairwise covariances between the modalities. The linear μMCA objective function thus becomes

$$\max_{\mathbf{W}^1,\ldots,\mathbf{W}^m} \operatorname{tr}\big[\sum_{i,j=1}^{m} \mathbf{W}^i\bar{\mathbf{X}}^{iT}\bar{\mathbf{X}}^j\mathbf{W}^{jT}\big], \qquad (5)$$

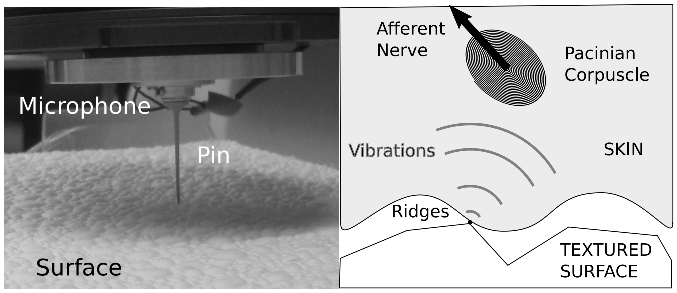
which can be solved as an eigenvalue problem. For WMCA, (2) becomes

$$\max_{\substack{\mathbf{W}^1,\ldots,\mathbf{W}^m \\ \mathbf{\Pi}^{1,2},\ldots,\mathbf{\Pi}^{m-1,m}}} \operatorname{tr}\big[\sum_{i,j=1}^{m} \mathbf{W}^i\mathbf{X}^{iT}\mathbf{\Pi}^{i,j}\mathbf{X}^j\mathbf{W}^{jT}\big], \qquad (6)$$

with the convention that $\mathbf{\Pi}^{i,i} = 0$ and $\mathbf{\Pi}^{i,j} = \mathbf{\Pi}^{j,iT}$. The WMCA problem can again be solved by an alternating maximization approach. The step of finding the projection directions is solvable as an eigenvalue problem. Finding the sample pairings requires solving $0.5m(m-1)$ linear assignment problems. The quadratic scaling in the number of modalities m does not pose a practical problem. Unless the sensor suite is highly redundant, usually only a few sensor modalities will produce related samples. Using multiple modalities to supervise one sensor also suffers from diminishing returns.

IV. Robot Experiments with Dynamic Touch and Vision

Three experiments were performed to show that the μMCA and WMCA methods are useful for learning dynamic tactile sensing. The first experiment tests the robot's performance on the supervised classification and the unsupervised clustering of tactile data. The second experiment evaluates the system's ability to generalize between materials, and involves classifying materials that it had not encountered during the learning phase. The final experiment investigates the robustness to incorrectly paired data. In all of these experiments, we assume that both tactile and visual information is available for learning the dimensionality reduction, but only the tactile sensor is available during the testing stage.



A. Tactile Sensor B. Human FA-II

Figure 6. A) The robot's tactile sensor. B) Diagram of how type II fast afferent nerves obtain tactile information (based on [2]). Both the sensor's pin and the human skin are compliant and move along the surfaces. When making and breaking contact with the surface, vibrations are created at the human's epidermal ridges and the tip of the sensor's pin. These vibrations are transferred through the skin and the pin respectively. When the vibrations reach the pacinian corpuscle, this mechanoreceptor transfers the signal to the human nervous system. Similarly, when the pin's vibrations reach the microphone's membrane, the microphone transfers the signal to the robot.

A. Tactile Sensor and Surface Materials

In order to explore various textured surfaces, we equipped a Mitsubishi PA-10 robotic arm with a single basic tactile sensor. The experimental setup is shown in Fig. 1. The aim of the experiments is to test the data processing procedure. We therefore used a straightforward oscillator-based design for the sensor. The dynamic tactile sensor consists of a compliant pin that makes contact with the surface, and a capacitor microphone that can detect the pin's vibrations at 44.1 kHz. Mechanisms in the human finger tip resemble this structure, as shown in Fig. 6. In particular, the sensor acts similar to an FA-II afferent, and the pin can be seen as either a part of the finger or as an object held by the robot. Given the compliance of the plastic pin, the location of the contact point with the surface could not be precisely determined. This sensor design is similar to other dynamic tactile sensors, such as the "whisker" sensor [49], [50]. The resulting apparatus is a suitable platform for testing the proposed WMCA and μMCA algorithms and showing that they can be applied to dynamic tactile sensors. Given that humans can discriminate between textures by probing them with a stylus [6], a single dynamic tactile sensor should be sufficient to perform the task.

The experiments were run on a set of 26 surfaces of 17 different materials. A common trait of these surfaces is that they have rich multi-scale textures. For example, a mosaic has the coarse texture set by the placement of the tiles, as well as the fine texture created by the surface of the tiles and cement (see Fig. 7, the supplementary information contains additional information on the materials). The data set includes materials that are similar and thus difficult to discriminate, as well as materials that are distinct and thus hard to generalize between.

The robot acquired samples by sliding the tactile sensor in a straight line across the surfaces. In this manner, each textured

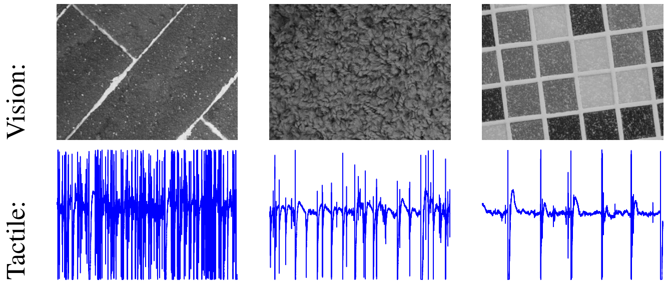


Figure 7. Examples of the multimodal data. The top images show the vision data while the bottom images show the corresponding time series of the tactile sensor signals. The x-axes of the tactile sensor plots represent time, while the y-axes represent the signal's magnitude. The samples for the plots were recorded over a four second time span.

surface was probed in five different regions. The robot used similar task-space movements for each region. If very different movements had been used, the data would require additional preprocessing to compensate for the different velocity profiles. Experiments have shown that humans also need to take into account the relative velocity between the finger and surface to accurately discriminate between textured surfaces [51]. After the robot had explored a surface with the tactile sensor, the object was repositioned 20 cm in front of the robot's camera for visual inspection. Four pictures were taken of each surface with different in-plane-rotations. The resulting grayscale images have resolutions of 512×768, as shown in Fig. 7. The pictures were taken in a well lit room.

B. Tactile and Visual Features

The information from both the tactile sensor and the camera were preprocessed to obtain suitable feature spaces. The robot probed five different surface regions from each of the 26 surfaces, resulting in 130 time series of tactile data. Textures are characterized by repeated local features. We therefore propose using a *bag-of-features* model [52], [53], which represents each region by a normalized histogram of local features. Local features are found by dividing each time series into 450 segments of 50ms, with 12.5ms overlaps between segments. In order to make the local features invariant to changes in phase and amplitude, each time segment was centered and its cepstrum was computed. The power cepstrum of a signal $\mathbf{z}$ is given by $C(\mathbf{z}) = |F(\log(|F(\mathbf{z})|^2))|^2$, where the function F is the Fourier transform, and describes the harmonic structure of the signal. It is often used to discriminate between different sources of acoustic signals [54]. Intuitively, the cepstrum represents the differences in the sound made by a brass and a string instrument playing the same note. In order to generate the desired histograms, we need to partition the cepstrum space. Hence, we partition the cepstrums into 1000 groups using k-means clustering. By using 1000 clusters, we ensure that the resulting feature vectors are sparse. Each of the $n = 130$ probed regions in $\mathbf{X}$ is thus represented as a normalized histogram of $d = 1000$ partitions, which indicate the relative occurrences of local cepstrum features.

The vision data was obtained by segmenting each of the 104 images into 32 equally-spaced strips. Each strip is three pixels wide. Similar to the regions probed by the tactile sensor, each strip is represented using a bag-of-features model. Along each strip, we compute *local binary patterns* over 3×3 pixel regions using *uniform patterns*, as suggested by Ojala et al. [55]. These 58 local features, shown in Fig. 8, are invariant to shifts in grayscale and rotations. Each of the $n' = 3328$ strips in $\mathbf{X}'$ is thus represented by a normalized histogram of $d' = 58$ partitions, which indicate the relative frequency of the local binary patterns.

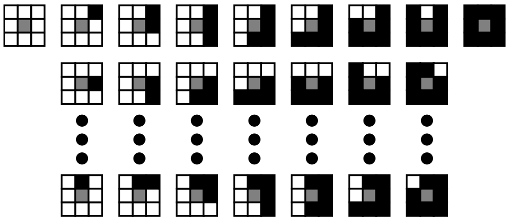


Figure 8. The 58 vision filters used to represent the textured images. Each 3×3 box represents a uniform binary pattern. The **grey** middle pixel defines the threshold value of the patch. A **black** pixel indicates that it is darker than the threshold, while a **white** pixel indicates that it is lighter or identical.

For both the image and tactile data, the feature dimensions were normalized to have zero mean and unit variance. This normalization step reduces the artifacts caused by having some histogram partitions being more populated than others.

C. Testing Performance, Ability to Generalize, and Robustness

Three experiments were run to compare the proposed μMCA and WMCA algorithms. The experiments' tasks were also performed with the standard PCA approach as well as the naive approach of not using any dimensionality reduction. The PCA method gives a baseline for using dimensionality reduction without the multi-modal data. The WMCA method used a ten step cooling schedule to reduce η from one to zero. The dimensionality reduction methods' only hyperparameter is the number of output dimensions q. The experiments were repeated for each output dimensionality in the range 1 to 55.

Each experiment consists of a learning phase and a testing phase. The learning phase corresponds to a robot exploring different object surfaces in a setting that allows for both visual and tactile inspection. The robot subsequently learns a mapping matrix $\mathbf{W}$ using one of the dimensionality reduction methods. The set of data used during the learning phase is known as the *learning set*.

The testing phase corresponds to a robot sorting different materials using only data from the tactile sensor. Visual inspection is not possible during the testing phase. The classification and clustering of the surfaces is performed, as described in Fig. 5, with the mappings $\mathbf{W}$ from the learning phase. The set of data used during the testing phase is known as the *testing set*. The classification tasks were evaluated using *a leave-one-out* scheme, i.e., we removed a data vector $\mathbf{x}_i$ from the testing set, trained a classifier on the remaining data, classified the removed vector $\mathbf{x}_i$, and then reinserted the data vector into the testing set. We repeated this procedure for each data vector in the testing set. The leave-one-out scheme makes efficient use of the available data and gives confident classification results.

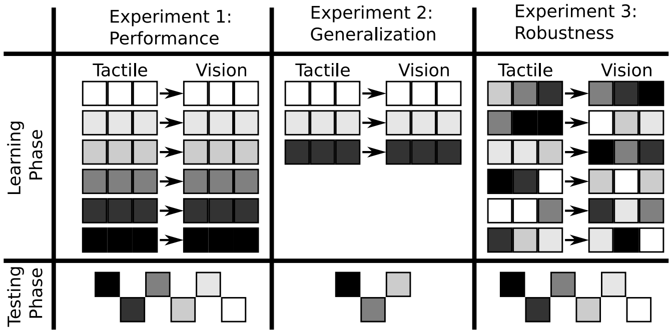


Figure 9. An illustration of the three experimental setups. The **top row** shows how the data was structured for the learning phase. Each small square represents the data from one surface region, and adjoining squares are grouped together. The shading of the squares indicates the materials that the sample was obtained from. The arrows indicate groups of samples that are weakly paired together between tactile and vision modalities. The **bottom row** indicates the materials that the learned system was tested on. Each square represents a type of material tested in the classification and clustering tasks. Testing data is limited to tactile data and, therefore, does not contain any groups or weak pairings. This figure does not show the true number of samples and materials used in the experiments.

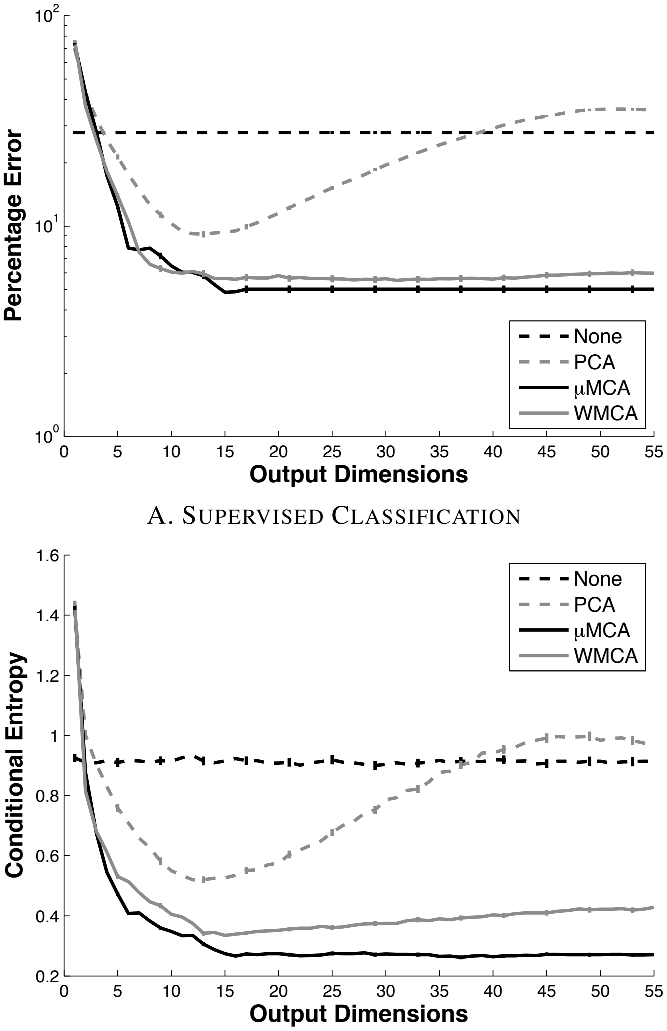
The labels used for classification are defined as the material from which the data was obtained.

The materials and groupings used to generate the learning and testing sets were altered for each of the three experiments in order to test different aspects of the dimensionality reduction algorithms. An overview of how the data was allocated to the learning and testing sets is shown in Fig. 9.

The first experiment investigates the performance at classifying and clustering surfaces. The learning set is generated by randomly selecting half of the tactile and visual data for each of the 17 materials. All of the data taken from the same textured surface is weakly paired together such that $g = 17$. The testing set consists of the other half of the tactile data. Thus, the learning and training sets both include examples from all 17 materials. For the clustering experiment, the number of clusters is set to the number of materials $c = 17$, and would otherwise need to be estimated from the data [56]. Additionally, the time required to learn the dimensionality reduction was recorded for each method.

The second experiment tests the ability to generalize to new materials. The learning set consists of the tactile and visual data from 10 randomly selected materials. All of the data taken from the same textured surface is weakly paired together such that $g = 10$. The testing consists of the tactile data from the seven materials excluded from the learning set. Hence, the learning and training sets consist of different materials. This experiment demonstrates how information can be transferred between related tasks using dimensionality reduction [57].

The third experiment tests the robustness to incorrectly paired data, which is a common problem for self-supervised approaches [21], [22]. Similar to the first experiment, the learning set is generated by randomly selecting half of the tactile and visual data for each of the 17 materials. However, rather than forming groups of the same material, the data is randomly allocated to the $g = 17$ groups. Hence, each weakly-paired group contains a mix of different materials. The testing set is the same as in the first experiment, and consists of the other half of the tactile data. Thus, the learning and training sets both include examples from all 17 materials. This situation is contrived and represents a worst case scenario that is unlikely to occur in practice.



A. Supervised Classification

B. Unsupervised Clustering

Figure 10. The performance of the tested methods for different numbers of output dimensions. Plot A shows the results from a classification problem. This plot uses a log scale for the y-axis. Plot B shows the results from a clustering experiment. In both plots, a lower value indicates a better performance. Error bars are also plotted, indicating +/- two standard errors of the mean.

Each experiment was run 500 times for each output dimensionality. For each run, A different seed value was used to initialize the randomization.

D. Results

The first experiment's classification and clustering results are shown in Fig. 10A and Fig. 10B respectively. The conditional entropy indicates how much information about the true material label is given by the cluster it has been assigned to. It is therefore a suitable measure of clustering performance [58]. The μMCA method achieved the best performance in both the supervised classification task, with an accuracy of 95.15%, and

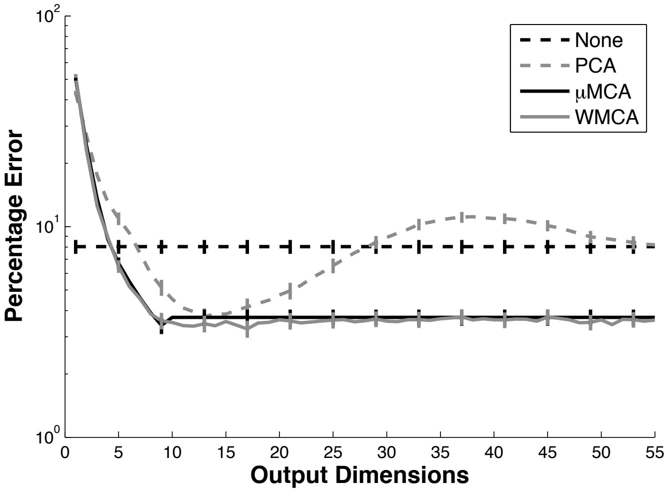


Figure 11. The graph shows the classification error incurred when classifying seven textures that were excluded from the learning set. The error bars indicate +/- two standard errors of the mean.

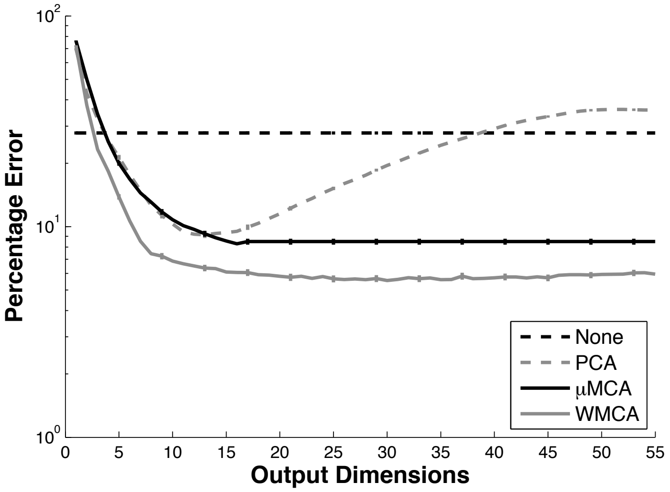


Figure 12. This graph shows the effects on classification performance when WMCA and μMCA are trained on incorrectly-paired data. Each weakly-paired group consists of a mix of materials, rather than a single material. The error bars indicate +/- two standard errors of the mean.

the unsupervised clustering task, with a conditional entropy of 0.262. The WMCA method achieved a similar classification accuracy, but a conditional entropy of only 0.335 for the clustering task. The unimodal PCA approach performed considerably worse than the multimodal approach with a best classification accuracy of 90.85% and a conditional entropy of 0.520. The naive approach gives a benchmark accuracy 72.14% of and a conditional entropy of 0.900. Both WMCA and μMCA display plateau structures of similar performance for a wide range of output dimensions.

The mean times required to compute matrix $\mathbf{W}$ are

WMCA	μMCA	PCA	Naive
1617ms	22ms	19ms	0ms

when run on a 3.0 GHz Intel Duo Core processor in python. The time required by μMCA can be decomposed into 7ms for computing the group means, and 15ms for computing the mapping matrix $\mathbf{W}$ from these means.

The results of the second experiment are shown in Fig. 11. These error rates are lower than in the first experiment, as this classification task only uses seven classes rather than 17. The WMCA, μMCA, and PCA approaches achieved similar classification accuracies of approximately 96.5%. The naive approach obtained an accuracy of 92.0%. The standard deviations in this experiment are approximately one and a half times as great as in the first experiment.

The results of the robustness experiment are shown in Fig. 12. The μMCA method's classification accuracy is similar to that of PCA. The WMCA method, with annealing, achieves performance levels similar to those of the first experiment.

E. Discussion

The results show that the use of visual data in the dimensionality reduction significantly improves the performance of the system. When the number of output dimensions increases, each method is selecting additional directions in the input space to keep. If the signals in these directions contain information relevant for tactile sensing, the performance improves. When the performance of a method decreases, it is including signals that are irrelevant to the tactile sensing, even though they have a high variance. Such signals could be caused by additional factors in the tactile modality, such as the vibrations of the robot [17], [11].

The PCA approach performs the best around $q = 16$ output dimensions. Deviations from this value lead to worse performance. In contrast, the WMCA method uses the vision information to determine which dimensions are relevant. By actively trying to exclude irrelevant signals, WMCA creates a plateau of good performance around the optimal q value. Hence, the WMCA method is less sensitive to changes in q and easier to tune.

By performing MCA on the group means $\bar{\mathbf{x}}$ and $\bar{\mathbf{x}}'$, the μMCA method automatically omits the dimensions describing variations within the groups. The resulting low-dimensional representations therefore contains less noise, which leads to better performance. These representations are especially well-suited for representing cluster centers, as shown by the clustering task's results. The μMCA method's plateau structure is the result of its limited output dimensionality $q \leq g$. Similar to WMCA, the μMCA method uses the vision data to include the relevant dimensions first. Hence, the final dimensions added tend to be the worst and decrease performance levels.

Both WMCA and μMCA perform well in the classification and clustering tasks of the first experiment. However, the standard errors of the mean, as shown in Fig. 10, indicate that μMCA's performance is significantly better. The WMCA method also requires considerably more computation time than μMCA and PCA. However, most applications will not require the learning to be performed in real time.

The second experiment shows that the abilities of μMCA and WMCA to generalize to new materials is similar to that of PCA. The good performance in this experiment suggests that the dimensionality reductions keep most of the pertinent information. The additional vision samples that WMCA did not find a pairing for may therefore be removed to save memory. The standard deviations are larger in this experiment because the performance is affected by the similarity between the learning and testing data sets. If the learning set includes materials similar to those in the testing set, the methods perform better.

Although the groups in the third experiment contained large amounts of incorrect data, the WMCA automatically found good pairings between samples. This result suggests that WMCA can be used with more complicated vision data and still find good pairings. Unlike WMCA, the μMCA method could not find suitable low-dimensional representations due to the incorrect data.

Since μMCA is less robust to incorrect data, it requires a more structured environment for the learning phase. The environment should allow for surfaces to be easily inspected through both vision and touch. The inspected surfaces should be easy to identify in the images and should ideally be large and flat. Since the μMCA method only requires weakly-paired samples, the objects may be freely manipulated by the robot between the tactile and vision inspections. Given these conditions, the environment should effectively resemble an infant's playpen.

The additional robustness of WMCA allows it to learn in more complicated environments. The experiments suggest that WMCA can handle situations such as having multiple objects in an image, and visually inspecting surfaces from multiple angles. The images must still contain some good data, but the robot is also allowed to collect some incorrect data while exploring. The WMCA may therefore be able to learn in everyday environments, as long as the conditions allow for both tactile and visual inspection of surfaces. The ability to learn by inspecting everyday objects is however beyond the scope of this paper, and will need to be thoroughly tested in the future.

In the future, the tactile data will also need to be preprocessed to take into account the velocity of the tactile sensor. Currently, moving the sensor at a different velocity has the same effect as scaling the textured surface. A similar problem occurs in the vision modality when a surface is observed at an angle. By preprocessing the data to make it invariant to such changes, the robot will be able to learn in even more complicated situations.

Once the dimensionality reduction has been learned with either μMCA or WMCA, the tactile sensor can be used in a wide range of situations. The tactile sensing will still benefit from the multimodal learning phase, even if the conditions do not allow for visual inspection.

V. Conclusion

Dynamic tactile sensing represents an important form of feedback when performing manipulation tasks. These sensors will therefore be vital for the many tasks that service robots may encounter. However, the data from tactile sensors is usually high dimensional and can contain vibrations from spurious sources. Hence, the data is difficult to use for discriminating between different surfaces.

In this paper, we presented the μMCA and WMCA methods for using tactile sensors to accurately and robustly classify textured surfaces. These methods use a second sensor modality, i.e. vision, during the learning phase to determine suitable lower-dimensional representations of the tactile data. The proposed approach relies on both sensors observing the relevant information from the environment, i.e. the texture of a surface. Any additional information is only observed by one of the modalities. For example, the surface's color is only seen by the camera and the robot's vibrations are only detected by the tactile sensor. Hence, the relevant part of the data is correlated between the modalities. A common problem when using multimodal data is the need to perfectly pair the data samples across modalities. The proposed methods were therefore designed to work with groups of weakly-paired data.

The μMCA method uses a maximum likelihood estimate to create a model of each group from its samples. The estimated means of the tactile data are paired with those of the vision data for each group. Subsequently, a maximum covariance analysis is applied to recover the relevant dimensions. The experiments show that the μMCA approach performs well in both classification and clustering tasks. The mapping to lower-dimensions can also be quickly learned from a set of samples.

The WMCA method uses an iterative maximization procedure to automatically determine suitable pairings within the groups. The final pairings lead to a dimensionality reduction mapping that is guaranteed to locally maximize the covariance between the modalities. In order to systematically converge on a good local maximum, the proposed WMCA implementation uses concepts from deterministic annealing. The experiments show that this approach is very robust and can even handle heavily mixed groups.

After learning a mapping to a lower dimensionality, the vision modality is no longer required. The tactile sensor can therefore be used in conditions where visual inspection in not possible, while still benefiting from the multimodal learning.

The experiments have shown that the methods can learn good dimensionality reduction mappings from only weakly-paired data obtained in semi-structured environments. The proposed algorithms can also be used with a variety of other sensors that acquire related samples. For example, a camera could be trained with a laser scanner to determine visual features that indicate depth-of-view, or a microphone to determine vision features related to audio sources. Therefore, the μMCA and WMCA methods are widely applicable in the field of robotics.

References

[1] J. Hawkins and S. Blakeslee, *On Intelligence*. Times Books, October 2004.

[2] J. Scheibert, S. Leurent, A. Prevost, and G. Debregeas, "The role of fingerprints in the coding of tactile information probed with a biomimetic sensor.," *Science*, vol. 323, no. 5920, pp. 1503–6, 2009.

[3] R. S. Johansson and J. R. Flanagan, "Coding and use of tactile signals from the fingertips in object manipulation tasks," *Nat Rev Neurosci*, vol. 10, no. 5, pp. 345–359, 2009.

[4] R. S. Johansson and G. Westling, "Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects," *Experimental Brain Research*, vol. 56, no. 3, pp. 550–564, 1984.

[5] J. R. Flanagan, M. C. Bowman, and R. S. Johansson, "Control strategies in object manipulation tasks.," *Curr Opin Neurobiol*, vol. 16, pp. 650–659, December 2006.

[6] S. J. Lederman and R. I. Klatzky, *Multisensory Texture Perception*, pp. 107 – 122. The MIT Press, 2004.

[7] S. Lacey, C. Campbell, and K. Sathian, “Vision and touch: Multiple or multisensory representations of objects?,” *Perception*, vol. 36, no. 10, pp. 1513 – 1521, 2007.

[8] F. N. Newell, M. O. Ernst, B. S. Tjan, and H. H. BÃŒlthoff, “Viewpoint dependence in visual and haptic object recognition,” *Psychological Science*, vol. 12, pp. 37–42, 2001.

[9] P. K. Allen, A. T. Miller, P. Y. Oh, and B. S. Leibowitz, “Integration of vision, force and tactile sensing for grasping,” *Int. J. Intelligent Machines*, vol. 4, pp. 129–149, 1999.

[10] B. S. Eberman and J. K. S. Jr., “Application of change detection to dynamic control contact sensing,” *Int. Journal Robotics Research*, vol. 13, no. 5, pp. 369–394, 1994.

[11] J. S. Son, E. A. Monteverde, and R. D. Howe, “A tactile sensor for localizing transient events in manipulation,” in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 471–476, 1994.

[12] G. Heidemann and M. Schöpfer, “Dynamic tactile sensing for object identification,” in *Proc. IEEE Int. Conf. Robotics and Automation*, (New Orleans, USA), pp. 813–818, IEEE, IEEE, 2004.

[13] D. Kragic and H. I. Christensen, “Biologically motivated visual servoing and grasping of real world tasks,” in *Intl Conf. on Intelligent Robotics and Systems (IROS)*, (Las Vegas, NV), IEEE, Oct. 2003.

[14] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini, “Tactile sensing: from humans to humanoids,” *Trans. Robotics*, vol. 26, no. 1, pp. 1–20, 2010.

[15] H. P. Saal, S. Vijayakumar, and R. S. Johansson, “Information about complex fingertip parameters in individual human tactile afferent neurons,” *Journal of Neuroscience*, vol. 29, pp. 8022–8031, June 2009.

[16] D. Johnston, P. Zhang, J. Hollerbach, Z. Hollerbach, and S. Jacobsen, “A full tactile sensing suite for dextrous robot hands and use in contact force control,” in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 3222–3227, 1996.

[17] R. D. Howe and M. R. Cutkosky, “Sensing skin acceleration for slip and texture perception,” in *Proc. Int. Conf. Robotics and Automation*, pp. 145–150, 1989.

[18] K. J. Kuchenbecker, J. Fiene, and G. Niemeyer, “Improving contact realism through event-based haptic feedback,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, pp. 219–230, March 2006.

[19] R. E. Bellman, *Adaptive control processes - A guided tour*. Princeton, New Jersey, U.S.A.: Princeton University Press, 1961.

[20] L. R. Tucker, “An inter-battery method of factor analysis,” *Psychometrika*, vol. 23(2), no. 2, 1958.

[21] B. Sofman, E. L. Ratliff, J. A. D. Bagnell, J. Cole, N. Vandapel, and A. T. Stentz, “Improving robot navigation through self-supervised online learning,” *Journal of Field Robotics*, vol. 23, December 2006.

[22] D. Kim, J. Sun, S. Min, O. James, M. Rehg, and A. F. Bobick, “Traversability classification using unsupervised on-line visual learning for outdoor robot navigation,” in *Proc. Int. Conf. Robotics and Automation*, 2006.

[23] C. H. Lampert and O. Kroemer, “Weakly-paired maximum covariance analysis for multimodal dimensionality reduction and transfer learning,” in *Proc. European Conf. Computer Vision*, pp. 1–15, 09 2010.

[24] K. Rose, “Deterministic annealing for clustering, compression, classification, regression, and related optimization problems,” in *Proceedings of the IEEE*, pp. 2210–2239, 1998.

[25] K. Pearson, “On lines and planes of closest fit to systems of points in space,” *Philosophical Magazine Series 6.*, vol. 2(11), no. 11, pp. 559–572, 1901.

[26] B. Schölkopf and A. J. Smola, *Learning with kernels*. MIT Press, 2002.

[27] G. Hinton and R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 313(5786), no. 5786, p. 504, 2006.

[28] T. Hofmann, “Probabilistic latent semantic indexing,” in *Proc. ACM SIGIR Conf. Research and development in information retrieval*, (New York, NY, USA), pp. 50–57, ACM, 1999.

[29] J. B. Tenenbaum, V. Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290(5500), no. 5500, p. 2319, 2000.

[30] R. A. Fisher, “The use of multiple measurements in taxomic problems,” *Ann. Eugenics*, vol. 7, pp. 179–188, 1936.

[31] F. R. Bach and M. I. Jordan, “A probabilistic interpretation of canonical correlation analysis.” Tech. Rep. 688, Department of Statistics, University of California, Berkeley, 2005.

[32] H. Hotelling, “Relation between two sets of variates,” *Biometrika*, vol. 28, pp. 322–377, 1936.

[33] H. Wold, “Estimation of principal components and related models by iterative least squares,” *Multivariate Analysis*, vol. 1, pp. 391–420, 1966.

[34] G. Baudat and F. Anouar, “Generalized discriminant analysis using a kernel approach,” *Neural computation*, vol. 12(10), no. 10, pp. 2385–2404, 2000.

[35] R. Rosipal and L. J. Trejo, “Kernel partial least squares regression in reproducing kernel Hilbert space,” *Journal Machine Learning Reasearch*, vol. 2, pp. 97–123, 2002.

[36] D. Hardoon, S. Szedmak, and J. Shawe-Taylor, “Canonical correlation analysis: an overview with application to learning methods,” *Neural Computation*, vol. 16(12), no. 12, pp. 2639–2664, 2004.

[37] R. Lienhart, S. Romberg, and E. Hörster, “Multilayer pLSA for multimodal image retrieval,” in *Conf. Image Video Retreival*, pp. 1–8, 2009.

[38] M. Blaschko and A. Gretton, “Learning taxonomies by dependence maximization,” *Proc. Conf. Neural Information Processing Systems*, 2009.

[39] D. L. Hall and J. Llinas, *Handbook of Multisensor Data Fusion*. CRC Press, June 2001.

[40] I. Halatci, C. a. Brooks, and K. Iagnemma, “A study of visual and tactile terrain classification and classifier fusion for planetary exploration rovers,” *Robotica*, vol. 26, no. 6, pp. 767–779, 2008.

[41] A. Angelova, L. Matthies, D. Helmick, and P. Perona, “Dimensionality reduction using automatic supervision for vision-based terrain learning,” in *Proc. Robotics: Science and Systems Conf.*, 2007.

[42] L. Matthies, M. Turmon, A. Howard, A. A. B. Tang, E. Mjolsness, J. Mulligan, and G. Grudic, “Learning for autonomous navigation: Extrapolating from underfoot to the far field,” in *NIPS Workshop Machine Learning Based Robotics in Unstructured Environments*, 2005.

[43] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala, “Latent semantic indexing: A probabilistic analysis,” *Computer and System Sciences*, vol. 61(2), no. 2, pp. 217–235, 2000.

[44] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, vol. 2, 1955.

[45] R. Jonker and A. Volgenant, “A shortest augmenting path algorithm for dense and sparse linear assignment problems,” *Computing*, vol. 38(4), no. 4, pp. 325–340, 1987.

[46] J. MacQueen, “Some methods for classification and analysis of multivariate observations.” 5th Berkeley Symposium on Mathematics, Statistics, and Probability, 1967.

[47] B. Schölkopf, A. J. Smola, and K.-R. Müller, “Kernel principal component analysis,” in *Proc. Int. Conf. Artificial Neural Networks*, pp. 583–588, 1997.

[48] G. H. Golub and C. F. Van Loan, *Matrix computations*. Johns Hopkins Univ. Press, 1996.

[49] M. Fend, “Whisker-based texture discrimination on a mobile robot,” in *Advances in Artificial Life - Proceedings of the 8th European Conference on Artificial Life (ECAL)*, Lecture Notes Artificial Intelligence, pp. 302–312, Springer Verlag Berlin, Heidelberg, 2005.

[50] S. N’Guyen, P. Pirim, and J.-A. Meyer, “Tactile texture discrimination in the robot-rat psikharpax,” in *Proc. Int. Conf. Bio-Inspired Systems and Signal Processing*, (Valencia, Spain), 2010.

[51] M. Hollins, A. Fox, and C. Bishop, “Imposed vibration influences perceived tactile smoothness.,” *Perception*, vol. 29, no. 12, pp. 1455–65, 2000.

[52] J. Zhang, S. Lazebnik, and C. Schmid, “Local features and kernels for classification of texture and object categories: a comprehensive study,” *International Journal of Computer Vision*, vol. 73, p. 2007, 2007.

[53] A. Schneider, J. Sturm, C. Stachniss, M. Reisert, H. Burkhardt, and W. Burgard, “Object identification with tactile sensors using bag-of-features,” in *Proc. Int. Conf. Intelligent robots and systems*, (Piscataway, NJ, USA), pp. 243–248, IEEE Press, 2009.

[54] B. Logan, “Mel frequency cepstral coefficients for music modeling,” in *Proc. Int. Symp. Music Information Retrieval*, 2000.

[55] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *Pattern Analysis and Machine Intelligence*, vol. 24(7), no. 7, pp. 971–987, 2002.

[56] D. Pelleg and A. Moore, “X-means: Extending K-means with efficient estimation of the number of clusters,” in *Proc. Int. Conf. Machine Learning*, pp. 727–734, 2000.

[57] S. J. Pan, J. T. Kwok, and Q. Yang, “Transfer learning via dimensionality reduction,” in *Proc. Conf. AAAI*, pp. 677–682, 2008.

[58] A. Rosenberg and J. Hirschberg, “V-measure: A conditional entropy-based external cluster evaluation measure,” in *Proc. Conf. Empirical Methods in Natural Language Processing*, pp. 410–420, 2007.