Tactile Image-to-Image Disentanglement of Contact Geometry from Motion-Induced Shear

Anonymous Author(s)
Affiliation
Address
email

Abstract: Robotic touch, particularly when using soft optical tactile sensors, suffers from distortion caused by motion-dependent shear. The manner in which the sensor contacts a stimulus is entangled with the tactile information about the geometry of the stimulus. In this work, we propose a supervised convolutional deep neural network model that learns to disentangle, in the latent space, the components of sensor deformations caused by contact geometry from those due to sliding-induced shear. The approach is validated by reconstructing unsheared tactile images from sheared images and showing they match unsheared tactile images collected with no sliding motion. In addition, the unsheared tactile images give a faithful reconstruction of the contact geometry that is not possible from the sheared data, and robust estimation of the contact pose that can be used for servo control sliding around various 2D shapes. Finally, the contact geometry reconstruction in conjunction with servo control sliding were used for faithful full object reconstruction of various 2D shapes. The methods have broad applicability to deep learning models for robots with a shear-sensitive sense of touch.

Keywords: Robotic Touch, Disentanglement, Shear, Object Reconstruction

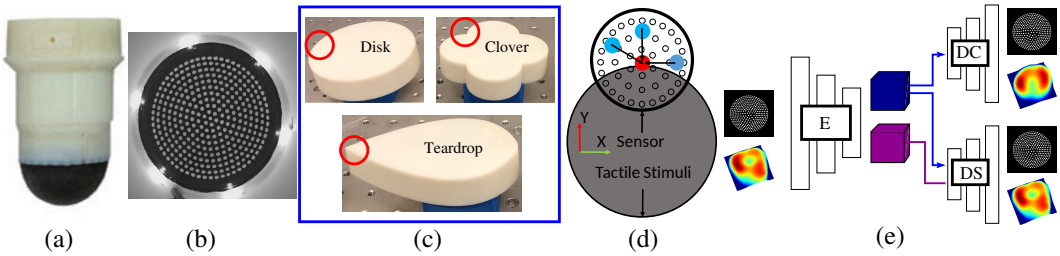

Figure 1: **Experiment setup and model architecture.** (a) TacTip sensor, (b) Sensor internal surface, (c) 3D-printed tactile stimuli with approximate locations of data collection (red circles), (4) Schematic of data collection, showing target location (red dot) and initial contact location (blue dots). Canonical (tapping) data was collected by vertically contacting the target; sheared (sliding) data was collected by sliding to the target, and (5) Schematic of the *Disentangled UnshearNet* model architecture with representative input/output tactile images and contact reconstructions.

1 Introduction

A fundamental aspect of tactile sensing is that it measures the environment via physical contact. Therefore, the manner in which the sensor contacts a stimulus is entangled with the tactile information about that stimulus. For example, when sliding or rubbing fingers across tactile stimuli, soft tactile sensors are inevitably distorted by motion-dependent shear, making the sensor response history-dependent and generalization hard in tactile tasks. This issue is prevalent in camera-based optical tactile sensors that use the lateral motion of markers to infer contact geometry, such as the TacTip [1], ChromaTouch [2], GelForce [3] and the optical tactile sensor described in [4], but may occur also in other shear-sensitive tactile sensors. Generally, this shear distortion of the sensor response when sliding along surfaces would make it difficult to perform tactile-dependent tasks such as shape reconstruction and continuous object exploration or manipulation.

Submitted to the 5th Conference on Robot Learning (CoRL 2021). Do not distribute.

Therefore, an important capability is to separate the deformation due to the manner of contact from that due to the stimulus geometry. For example, when continuously exploring an object surface, the sliding-induced shear would need to be removed from tactile image to extract the local contact shape. Previous work on extracting pose information from contact to continuously slide a tactile sensor over complex objects used a deep convolutional neural network trained to map between shear-distorted inputs and the desired (pose) outputs, which was necessary for sliding exploration over a range of complex novel objects [5, 6, 7]. Here, we suggest, an alternate disentanglement based approach that learns to remove motion-induced shear from raw sheared sensor response that has several advantages over previous task-specific approach of training insensitivity to shear, in that: (a) it is computationally and data efficient by only requiring the 'unshearing' of the tactile data to be learnt once instead of relearning for each tactile task, such as when applied to full object reconstruction; (b) the models can then be learnt more easily on unsheared data, which we demonstrate by predicting local object pose; moreover, in some cases no model need be learnt at all, which we demonstrate with contact surface reconstruction using a Voronoi tessellation from the tactile marker data; and (c) the methods are extensible to including other downstream variables of interest appropriate to the task, rather than having to completely retrain new models from scratch.

Our approach to disentangling environmental factors of variation and object attributes in tactile sensing is based on work in computer vision where disentanglement is considered important to improve the generalizability of machine learning models, for example to aid in learning of robust object representations for object classification under novel scenarios [8]. Inspired from this, here we propose a supervised convolutional neural network model that learns to disentangle the components of sensor deformation caused by the contacted object geometry from those due to motion-induced shear. The models were trained using paired data from discrete (tapping) and continuous (sliding) touch, with an image-to-image 'Disentangled UnshearNet' model architecture that outputs unsheared tactile images from sheared images (Fig. 1). Our validation: (1) quantifies that the unsheared images from continuous motion match their paired tactile images from discrete taps (Table 1); (2) demonstrates that local contact geometry reconstructed from unsheared images using Voronoi match their paired tapping counterparts over several stimuli shapes (Fig. 4, top row); (3) infers pose from the sheared tactile images by re-using a convolutional neural network model trained on discrete (tapping) data, which was then applied to controlling robust sliding exploration around several planar objects (Fig. 4, middle row); and (4) combines the two previous steps to attain faithful full-object reconstruction of several planar shapes (Fig. 4, bottom row).

2 Background

Measuring strain is a fundamental component of tactile sensing, as emphasised by Platkiewicz, *et al* [9]: 'the pressure distribution at the surface of a tactile sensor cannot be acquired directly and must be inferred from the [strain] field induced by the touched object in the sensor medium.'

There are many mechanisms whereby normal strain and shear strain are measured by tactile sensors. Taxel-based tactile sensors such as the iCub fingertip [10] and BioTac [11] directly measure normal strain. Nevertheless, this measurement of normal strain will be affected by shearing of the soft sensor medium, although the effects on sensor output will be much smaller than those from normal pressure.

Camera-based optical tactile sensors can be very sensitive to shear because they transduce deformation of the soft sensor surface into a lateral image. For optical tactile sensors such as the TacTip biomimetic optical sensor used here [13, 1] and other marker-based sensors [2, 4], the surface shear will also distort the apparent geometry of the contact. This distortion is because the normal indentation is represented in the local shear strain of the markers, upon which the global shear strain is superimposed. This contrasts with optical tactile sensors that reconstruct the indentation field from reflection, such as the GelSight [14], where shear will shift the imprint on the tactile image but otherwise the imprint shape should be relatively unaffected.

This effect of shear complicates tactile perception under general motions against an object, and restricts the application of touch on challenging tasks such as tactile exploration and surface shape reconstruction. Studies have been limited to discrete contacts where the sensor taps discretely over the object to minimize shear [16, 17, 19], or to rigid planar tactile sensors [20, 21] that do not shear.

A recent body of work on the TacTip sensor used here focused on mitigating the effects of motion-dependent shear during tactile servo control to slide around planar shapes [5, 18] and complex 3D objects [6, 7]. The shear-dependence problem was addressed in three studies that used a deep

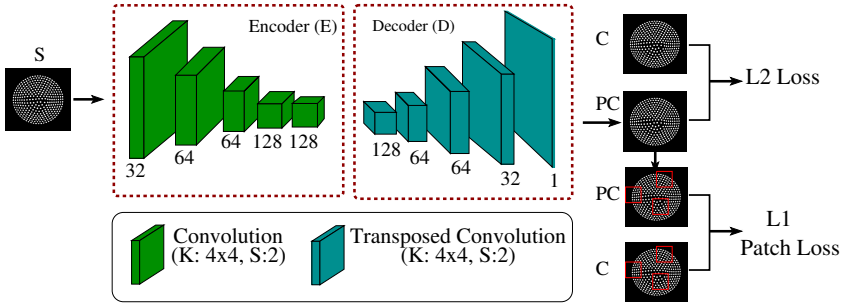

Figure 2: **Entangled UnshearNet architecture:** A shear-distorted tactile image S was processed by the encoder E and decoder D (kernel K, stride S) with output PC: the unsheared counterpart of input S post removal of sliding-induced distortion. The match between predicted unsheared data PC and canonical data C was enforced by training with a direct comparison between PC and C along with comparing corresponding patches (20×20 pix.) taken from PC and C to enforce local compliance.

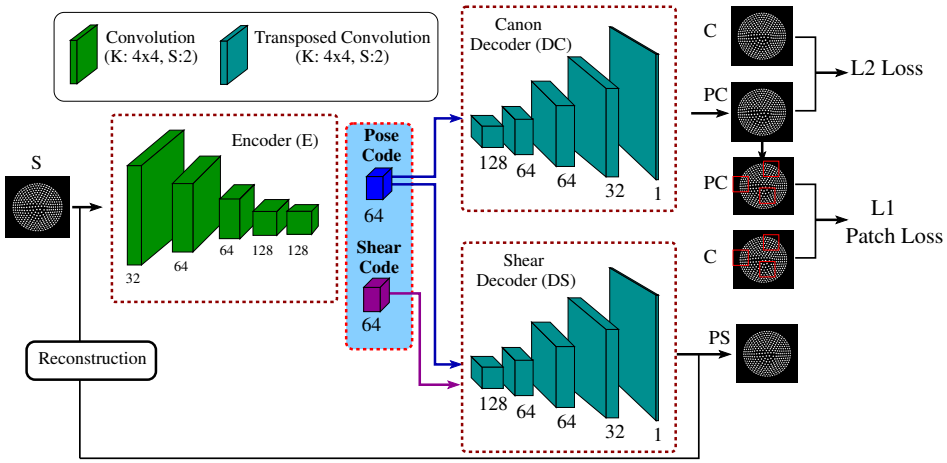

Figure 3: **Disentangled UnshearNet architecture:** Labelling as in Fig. 2, with two decoders DC and DS. The first decoder has output PC: the unsheared counterpart of input S post removal of sliding-induced distortion; the second decoder DS reconstructed the input S, denoted PS.

convolutional neural network to predict pose by training insensitivity to shear into the prediction network [5, 6, 7]. While effective, this has a drawback that procedures for learning shear-invariance needs to be introduced into the training for each individual task, and thus would not generalize easily to new tasks. Another study did not use deep learning, but instead dimensionally reduced the tactile data [18], where the the pose components naturally separated from the shear components.

This work pursues an alternate approach using deep learning to remove the effect of motion-dependent shear from the sheared tactile image, offering several advantages that were summarized in the introduction. Our approach to disentangling environmental factors of variation and object attributes in tactile sensing is based on work in computer vision where disentanglement is considered important to improve the generalization of machine learning models, for example to aid in learning of robust object representations for object classification under novel scenarios [8]. Once disentangled, different environmental factors and object attributes can be recombined to generate coherent novel concepts thus extending knowledge to previously unobserved scenarios [8]. For example, synthesizing sensor responses to novel stimuli by combining the stimuli attributes and factor of variation without actually observing them to plan action under previously unobserved conditions. In the case of tactile sensing considered here, the stimuli attributes could be novel contact geometry and the factors of variation the motion-induced shear.

3 Methods

3.1 Experimental Setup

The setup in this study followed recent experiments with the TacTip biomimetic optical tactile sensor [5, 18, 6, 7] where it is mounted as an end effector on a robot arm (Fig. 1). The TacTip is biomimetic in that its skin morphology is based on the layered dermal and epidermal structure of the human fingertip [13, 1]; the tactile sensor measures the shear displacement of an array of papillae pins on the inner side of the sensing surface caused by the skin deformation.

The TacTip can be designed with varied morphologies dictated by the application [1]. The TacTip sensor (Fig. 1 (a) & (b)) used in this work had a soft rubber-like hemispherical dome (40 mm radius, TangoBlack+) manufactured using multi-material 3D printing. Its inner side was covered with an array of 331 pins with hard white tips arranged in a concentric circular grid. The dome was filled with an optically-clear silicone gel to give it a compliance similar to a human fingertip. The final assembly of sensors is done by hand that introduces variability in the response in other otherwise same sensors. A camera was used to collect tactile images of the internal surface of the dome. For more details, we refer to the references above.

The tactile stimuli used here were three 3D-printed (ABS plastic) planar shapes: a circular disk, clover and teardrop (Fig. 1 (c)). These objects were chosen to give different edge shapes for application of our methods, and were fastened securely to the workspace to prevent accidental motion. An additional four complex planar shapes, including two acrylic shapes with distinct frictional properties to the remaining 3D-printed shapes (spiral 1 & 2, Fig. 4) were used to test for generalizability.

3.2 Data Collection

3.2.1 Paired Canonical and Sheared Data

This work proposes a supervised deep learning model *Disentangled UnshearNet* to remove the effect of motion-induced global shear in the tactile images, caused by the friction between the sensor skin and its contacted surface during contact. We emphasise this motion-induced shear is different from the skin deformation due to local shear from the geometry of the stimulus imprint.

Our method requires paired tactile data with minimal global shear taken from vertically tapping onto the stimulus (referred to as canonical data) and sliding data with random global shear (referred to as sheared data). The sheared data is collected after sliding randomly and is paired with a tactile image from the canonical data by using the same relative poses between the sensor and the stimuli (Fig. 1). The raw tactile data was cropped, sub-sampled to a 256×256-pixel region and finally converted to binary images using an adaptive threshold to minimise sensitivity to internal lighting conditions.

In total, data for 200 distinct poses were collected for each of the three stimuli (Fig. 1 (c)). To generate poses, the sensor location relative to the target contact location (red circle, Fig. 1 (d)) was sampled randomly from a uniform distribution spanning the range [-5, 5] mm in the two lateral directions (along the x- and y-axes), [-45, -45] deg in yaw (θn_Z) and [-6, -1] mm in depth (z-axis). Ranges of the target poses and perturbations to generate canonical and sheared data were chosen to ensure safe contact of the sensor with the stimulus (Fig. 1 (c)).

Canonical Data: The canonical data was taken by vertically tapping onto the stimulus, to give a representative sample of tactile data with minimal global shear just due to the stimulus geometry. This gives target (reference) data that the sheared data from sliding contacts should be restored to. The dataset, in total, had 30,000 samples: 50 instances of each of the 200 poses recorded previously for each of the three stimuli shapes (Fig. 1 (c)). To generate instances, the indentation depth (z-axes) was varied randomly between [-1, 1] mm from the original indentation depth.

Sheared Data: The sheared data accords to a canonical tactile image distorted by global shear from lateral sliding motion of the sensor on the stimulus. Thus, the sheared data does not give contact-only information akin to canonical data that represented the geometry of the contacted stimulus, but has a significant component due to the contact motion.

The dataset, in total, had 90,000 samples: 150 instances of each of the 200 random poses recorded previously for each of the three stimulus shapes (Fig. 1 (c)). To generate instances for each pose, the lateral position offset for sliding to the target pose was varied randomly between [-2.5, 2.5] mm in lateral direction(s) (along the x- or y-axes or both), using the procedure shown in Fig. 1 (d).

Training and Test Data: The canonical and sheared datasets were partitioned randomly into training, validation and test sets (ratio 60:20:20); i.e. from the 200 unique poses for each stimulus shape (Fig. 1 (c)), 120 poses were assigned to a training set and 40 poses each to validation and testing sets. Overall the canonical and sheared training set contained 18,000 and 54,000 samples respectively, while the validation and testing sets contained 6,000 and 18,000 samples respectively. To generate paired canonical and transformed data for training, each transformed tactile image was paired randomly with one of the canonical instances having the same pose.

3.2.2 Pose-labelled Data

To train pose prediction network *PoseNet*, a separate set of pose data was collected instead of reusing the previously collected pose data to train *Entangled & Disentagled UnshearNet*. This was done to not overfit the training set and to ensure performance of the trained *UnshearNet* models despite potential dataset shifts due to a host of factors, including drift in sensor characteristics over time or with continued use due to wear and tear of the tactile sensing surface.

The dataset had, in total, 30,000 samples: 10,000 samples from each of the three object shapes (Fig. 1 (c)). To generate contact poses, the sensor location relative to the initial contact location was sampled randomly from a uniform distribution spanning the range [-5, 5] mm laterally (x- and y-axis directions), [-45, -45] deg in yaw (θn_Z) and [-6, -1] mm in depth (z-axis). Ranges of the target poses, as before, were chosen to ensure safe contact of the sensor without causing damage. As discussed above, all tactile images were cropped, sub-sampled and thresholded to 256×256-pixel to binary (black/white) tactile images. The dataset was partitioned randomly into training , validation and test sets in the ratio 60:20:20 as above, containing 18,000, 6,000 and 6,000 samples respectively.

3.3 Image-to-Image Models for Shear Removal

To achieve our goal of removing the effects of shear induced by the sliding motion, we trained two models based on the encoder-decoder framework – *Entangled UnshearNet* (Fig. 2) and *Disentangled UnshearNet* (Fig. 3). The former model (baseline) learns the mapping between the input sheared data and its corresponding canonical (tap) data directly. The later model instead, first disentangles the contact-only (pose) sensor response component from the sliding-induced global shear component in the latent space of the encoder E; followed by accurate reconstruction of the input sheared image and its corresponding canonical (tap) image using decoders *DS* (sheared or sliding) and *DC* (canonical or tapping). We hypothesized that a more robust mapping between sheared and canonical (tap) data can be learned by disentangling the contact-only and sliding-induced components of sensor response. The mathematical formulation of the *Disentangled UnshearNet* is listed below:

The encoder E splits the latent space into two components $\mathbf{z} = (\mathbf{z_p}, \mathbf{z_s}) \in Z_p \times Z_s$, given paired canonical $\{x_c\}_{i=1}^N$ and shear-transformed training samples $\{x_s\}_{i=1}^N$, where $x_c \in X_c$ and $x_s \in X_s$ are in the sets of canonical and sheared data, respectively.

The decoder *DC* takes as input only the pose part of the latent code $\mathbf{z}_p$ and reconstructs the canonical sample corresponding to the input sheared sample. In combination with the encoder E, it thus learns a mapping $M : X_t \rightarrow X_c$ from the sheared to canonical data.

The other decoder *DS*, takes as input the full latent code $\mathbf{z}$ and reconstructs the input sheared sample. In combination with the encoder E, it thus acts as an auto-encoder and learns a mapping $N : X_s \rightarrow X_s$ from the sheared data to itself.

Our objective function has three components:
(1) *L2 loss* for matching the generated canonical samples $M(x_t)$ to their targets x_c;
(2) *L1 patch loss* for matching random patches extracted from generated canonical images $M(x_t)$ and corresponding targets x_c for enforcing image similarity locally as changes in sensor response due to change in contact conditions are predominately local in nature;
(3) *Reconstruction loss* for matching the generated sheared samples $N(x_t)$ to their target input sheared samples x_s.

The objective functions are:

$$\mathbb{L}_{\text{rec}}(E,DS) = \mathbb{E}_{x_s \sim X_s}[\|x_t - DS(z_p, z_s)\|_2], \tag{1}$$

$$\mathbb{L}_{\text{sup}}(E,DC) = \mathbb{E}_{x_s \sim X_s, x_c \sim X_c}[\|x_c - DC(z_p))\|_2], \tag{2}$$

$$\mathbb{L}_{\text{patch}}(E,DC) = \mathbb{E}_{x_s \sim X_s, x_c \sim X_c}[\|\text{crops}(x_c) - \text{crops}(DC(z_p)))\|_1], \tag{3}$$

where $(z_p, z_s) = E(x_s)$ for all $x_s \in X_s$. The function crops() selects a patch of size 20×20 pix from $M(x_t)$ and x_c. The objective functions L_{rec}, L_{sup} and L_{patch} represent sheared reconstruction loss, canonical supervised loss and canonical supervised patch loss respectively.

The final objectives for Encoder E and Decoders DC, DS are:

$$\mathbb{L}_E = \mathbb{L}_{\text{rec}}(E,DS) + \mathbb{L}_{\text{sup}}(E,DC) + \lambda \cdot \mathbb{L}_{\text{patch}}(E,DC), \tag{4}$$

$$\mathbb{L}_{DC} = \mathbb{L}_{\text{sup}}(E,DC) + \lambda \cdot \mathbb{L}_{\text{patch}}(E,DC), \tag{5}$$

$$\mathbb{L}_{DS} = \mathbb{L}_{\text{rec}}(E,DS), \tag{6}$$

where λ is the relative loss scale factor.

Details of the specific model architectures and their training are given in Appendices A & B.

4 Experimental Results

4.1 Shear Removal from Tactile Images

We first performed an ablation study to demonstrate the separation of latent representations by the *Disentangled UnshearNet* into pose and shear codes respectively. For details see Appendix D.

The first test of the performance of our approach for removing motion-induced global shear from tactile images used the multi-scale structural similarity index (SSIM) [25] to compare tactile images from the shear-transformed test data set with the paired canonical and the *Disentangled (Entangled) UnshearNet*-generated unsheared tactile images. All three images represent the same pose.

To compare, we computed the average SSIM between the canonical (tap)-sheared and canonical (tap)-unsheared images in several distinct scenarios (stimuli: disk edge, clover corner, teardrop corner). The unsheared images were structurally closer to the target canonical (tap) samples than their sheared counterparts. The overall SSIM value for unsheared samples was 0.93 (0.90) in comparison to 0.32 for sheared samples, with closeness to unity indicating a good match (Table 1). Performance was similar for all stimuli considered with *Disentangled UnshearNet* achieving closer match to canonical samples than its *Entangled* counterpart.

Table 1: **Multi-Scale Structural Similarity Index (MS-SSIM)**

Image Similarity Index				
	Disk Edge	**Clover Corner**	**Teardrop Corner**	**Overall**
Tap Data & Tap Data	1	1	1	1
Tap Data & Shear Data	0.33	0.31	0.33	0.32
Tap Data & Unsheared (Entangled)	0.91	0.90	0.90	0.90
Tap Data & Unsheared (Disentangled)	0.94	0.92	0.92	0.93

4.2 Reconstructed Contact Geometry

As stated in the introduction, an application of shear removal is to aid reconstruction of the stimulus geometry from the tactile images. This is validated by using a Voronoi-based method to infer an approximate indentation field, using the method introduced in Cramphorn *et al* [12] on tactile data from the TacTip. The Voronoi method transforms the displacements of the tactile markers into areas of hexagonal cells tessellating the grid of the markers [12, Fig. 4]. The change in Voronoi areas of the cells represents the skin distortion, which correlates with the indentation due to contact geometry.

For display purposes, we fit a 3D surface to the (x, y) centroid coordinates using the Voronoi cell areas as the corresponding height values (Fig. 4, top row). The fitted surface from the original canonical tactile images taken under tapping visibly represents the shape of the indentation on the tactile sensor, as expected (Fig. 4, top row). The same stimulus under sliding motion is visibly distorted by shear, resulting in an indentation field that does not resemble the stimulus shape (Fig. 4, top row).

Evidently, the indentation fields reconstructed from the model-generated unsheared tactile images under sliding motion resemble those from the canonical data (Fig. 4, top row; last column, all stimuli). Again, this confirms the successful removal of sliding-induced distortion from the tactile image.

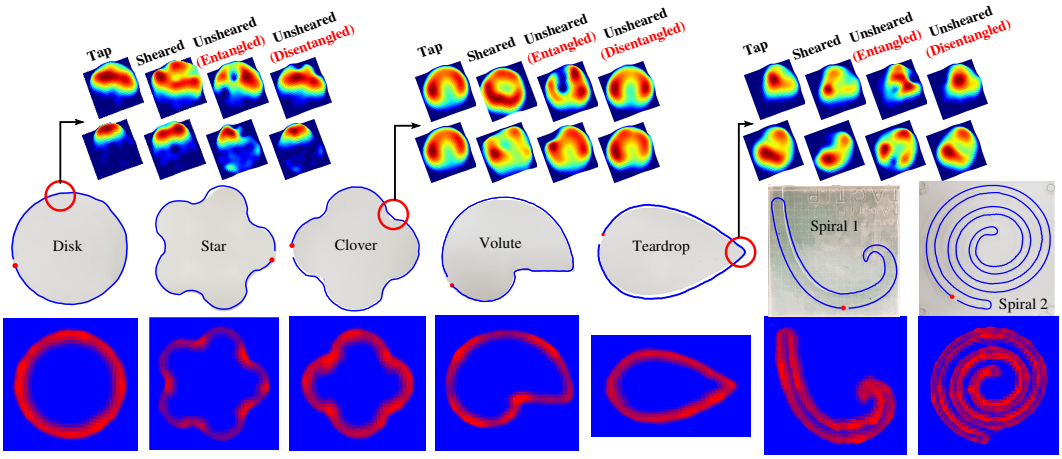

Figure 4: **Contact geometry reconstructions, contour following and shape reconstructions.** Top row: Local contact geometry reconstruction useing (x, y)-centroid values and the Voronoi cell areas as z values to fit a 3D surface using cubic interpolation. The approximate contact geometry can be recovered from sheared data using the unsheared images generated by the two types of *UnshearNet* (entanged and disentangled), but not from the original sheared data. Middle row: Trajectories under robust sliding using pose estimation post removal of sliding-induced shear, overlaid on the objects, with pose estimation applied to the output of the *Disentangled UnshearNet* (red dot: starting point of trajectory). Bottom row: Full object reconstructions from combining the unsheared local contact geometry reconstructions along the sliding trajectories. High-resolution figures are in Appendix E.

4.3 Validation on Shape Exploration

To further confirm that our model can remove shear from transformed images, we trained a prediction network (PoseNet) on stimulus pose parameters: lateral position (x-horizontal, τ_X) and in-plane orientation (yaw, θn_Z), using canonical tactile images from vertical tapping contacts. PoseNet performance was quantified with the mean-absolute error (MAE) between predicted pose and target when input data were canonical, sheared and model-generated unsheared tactile images respectively using both the entangled and disentangled networks (results in Table 2). On the sheared data, the lowest prediction error was obtained on unsheared samples with the disentangled network, with the entangled network about 50% higher, and on the original sheared samples about 5× poorer. Again, the *UnshearNet* managed to successfully correct the distortion in sensor response cause by slide-induced global shear, with the disentangled representation giving the best performance.

Another application of shear removal that also tests the real-time performance of UnshearNet is to use the above pose prediction network to servo control smoothly over varied object shapes during sliding motion. The robot successfully traced contours across all three shapes by a combination of the use of *UnshearNet* for global shear removal, *PoseNet* for prediction of object pose from unsheared images (Fig. 4, second row) and a simple control policy (see Appendix C). This clearly indicates that the proposed *Disentangled UnshearNet* is well suited for correcting the sliding-induced shear-distorted sensor response for real-time servo control applications.

4.4 Full Object Reconstruction

As a final demonstration of our model's utility to remove shear, we reconstructed full object shapes by combined use of contact geometry estimation (section 4.3) and sliding shape exploration (section 4.4), both of which are adversely affected by motion-induced shear (Fig. 4, bottom row).

Table 2: **Mean Absolute Error (MAE) in Pose Predictions**

Pose Prediction Error				
	Tap Data	**Sheared Data**	**Unsheared Data (Entangled)**	**Unsheared Data (Disentangled)**
horizontal, τ_x (mm)	0.43	2.72	0.95	0.64
yaw, θn_Z (degrees)	2.13	22.20	6.73	4.38

To attain full object reconstructions, we fused together the contact information extracted from unsheared images corresponding to sliding contacts (acquired during shape exploration) and linearly interpolated them over a rectangular grid (Fig. 4, bottom row). These results clearly demonstrates the effectiveness of our approach in learning the sheared-unsheared mapping that can then later be reused for multiple downstream tasks. In addition, the faithful shape reconstructions along with the shape exploration across multiple planar objects show that the model generalizes to novel situations.

5 Conclusion

In this work, we proposed a tactile image-to-image supervised convolutional neural network model to remove the effects of motion-induced global shear from the tactile sensor response while preserving the sensor deformations due to the spatial geometry of the contacted stimuli. Two *UnshearNet* models were considered: one based on a basic encoder-decoder structure with an entangled pose and shear latent space, and a second that disentangled these components in the model latent space (Figs 2, 3).

This approach was validated in three ways: (1) Comparing similarity of tactile images collected by vertically tapping without a shearing motion to those collected with an unknown shearing perturbation, for which the unsheared data passed through the *Disentangled UnshearNet* gave a good match according to the structural similarity index (Table 1). (2) Reconstructing the indentation field from the contact geometry using sheared tactile images passed through the *Disentangled UnshearNet* (Fig. 4, top row), via a (Voronoi) method used previously to reconstruct the sensor deformation [12]. The removal of motion-induced shear was apparent in the model-generated unsheared tactile images, with the contact geometry only visible after use of the UnshearNet with the original sheared image obscuring the contact shape. (3) Predicting the local pose of part of an object in a shear-insensitive manner with a prediction network trained only on canonical (tapping) data (Table 2). This approach was then applied to smooth servo control around several planar objects (Fig. 4, middle row).

The combination of smooth servo control and local shape reconstruction was later used for full object reconstruction (Fig. 4, middle row), which remains an problem under active investigation in the field [23, 24]. Our approach improves upon related work [5, 6] that instead trained insensitivity to shear into the prediction network itself. An obvious drawback of this approach, even though it works well for smooth servo control, is that it does not extend readily to predict other tactile dimensions such as parameters of the local contact geometry. In that case, new labelled data would need collecting to train a new model for each intended task. Our approach, of shear removal from tactile images, instead means that a single model based on the canonical (tapping) data can be re-used.

The present study considered only planar objects in 2D and two components of pose. As recent results have shown how to extend the *PoseNet* to 3D surfaces (3 pose components) or 3D edges (5 pose components) for tactile servo control [5, 6], we expect the present results will extend to those situations, although questions remain on the best way to do this with the large number of degrees of freedom. Although the proposed disentanglement-based approach to remove shear from sensor response has been tested on TacTip optical tactile sensor, it should hold well for other soft optical sensors. The models, however, would have to be retrained as the sensor response depends on the frictional and elastic properties of the tactile surface, which varies across sensor types.

One limitation of our current approach is that we limited our training to just translation shear. However, smooth servoing over the objects necessarily introduces rotational shear, as the servo control rotates the sensor while continuously sliding over the object surface. However, the fact that the methods performed well both in controlling the sensor and reconstructing the object geometry in the presence of this rotational shear shows that some unshearing carries over from translation to rotational shear at least for planar objects. Our expectation is that bringing rotational shear into the training will become important when extending the methods to 3D objects. Another limitation of the supervised approach investigated here is the requirement of paired canonical and sheared data, which complicates the data collection. Thus, a semi-supervised approach would be a worthwhile extension to this work, by obviating the need for paired data collection. Overall, we expect the methods developed here can can be applied more widely to various other exploration and manipulation tasks involving soft tactile sensors that are adversely affected by motion-induced shear.

References

[1] B. Ward-Cherrier, N. Pestell, L. Cramphorn, B. Winstone, M. E. Giannaccini, J. Rossiter, and N. F. Lepora," The TacTip Family: Soft Optical Tactile Sensors with 3D-Printed Biomimetic Morphologies," Soft Robotics, Apr 2018, pp. 216-227.

[2] X. Lin, L. Willemet, A. Bailleul and M. Wiertlewski, "Curvature sensing with a spherical tactile sensor using the color-interference of a marker array," 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 2020, pp. 603-609.

[3] Kazuto Kamiyama, Kevin Vlack, Terukazu Mizota, Hiroyuki Kajimoto, Naoki Kawakami, and Susumu Tachi. Vision-based sensor for real-time measuring of surface traction fields. *IEEE Computer Graphics and Applications*, 25(1):68–75, January 2005.

[4] C. Sferrazza and R. D'Andrea," Design, Motivation and Evaluation of a Full-Resolution Optical Tactile Sensor," Sensors, vol. 19(4), pp. 928, February 2019.

[5] N. F. Lepora, A. Church, C. de Kerckhove, R. Hadsell and J. Lloyd, "From Pixels to Percepts: Highly Robust Edge Perception and Contour Following Using Deep Learning and an Optical Biomimetic Tactile Sensor," in IEEE Robotics and Automation Letters, vol. 4, no. 2, pp. 2101-2107, April 2019.

[6] N. F. Lepora and J. Lloyd, "Optimal Deep Learning for Robot Touch: Training Accurate Pose Models of 3D Surfaces and Edges," in IEEE Robotics & Automation Magazine, vol. 27, no. 2, pp. 66-77, June 2020.

[7] N. F. Lepora and J. Lloyd. Pose-Based Servo Control with Soft Tactile Sensing. December 2020.

[8] Y.Bengio. Deep Learning of Representations: Looking Forward. June 2013.

[9] Jonathan Platkiewicz, Hod Lipson, and Vincent Hayward. Haptic Edge Detection Through Shear. *Scientific Reports*, 6(1):23551, September 2016.

[10] Nicholas Wettels, Veronica J. Santos, Roland S. Johansson, and Gerald E. Loeb. Biomimetic Tactile Sensor Array. *Advanced Robotics*, 22(8):829–849, January 2008.

[11] Alexander Schmitz, Perla Maiolino, Marco Maggiali, Lorenzo Natale, Giorgio Cannata, and Giorgio Metta. Methods and Technologies for the Implementation of Large-Scale Robot Tactile Sensors. *IEEE Transactions on Robotics*, 27(3):389–400, June 2011.

[12] L. Cramphorn, J. Lloyd and N. F. Lepora, "Voronoi Features for Tactile Sensing: Direct Inference of Pressure, Shear, and Contact Locations," 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, 2018, pp. 2752-2757.

[13] C. Chorley, C. Melhuish, T. Pipe, and J. Rossiter, "Development of a Tactile Sensor Based on Biologically Inspired Edge Encoding," in 2009 International Conference on Advanced Robotics, 2009, pp. 1–6.

[14] W. Yuan, S. Dong, and E. Adelson, "GelSight: High-Resolution Robot Tactile Sensors for Estimating Geometry and Force," Sensors, vol. 17, no. 12, p. 2762, Nov. 2017.

[15] H. Liu, K.C. Nguyen, V. Perdereau, J. Bimbo, J. Back, M. Godden, L.D. Seneviratne and K. Althoefer, "Finger contact sensing and the application in dexterous hand manipulation," Auton Robot 39, pp. 25–41, 2015.

[16] N. F. Lepora, K. Aquilina and L. Cramphorn, "Exploratory Tactile Servoing With Active Touch," in IEEE Robotics and Automation Letters, vol. 2, no. 2, pp. 1156-1163, April 2017.

[17] U. M.-Hernandez, T. J. Dodd, M. H. Evans, T. J. Prescott, and N. F. Lepora, "Active sensorimotor control for tactile exploration," Robotics and Autonomous Systems, vol. 87, pp. 15–27, 2017.

[18] K. Aquilina, D. A. W. Barton and N. F. Lepora, "Shear-invariant Sliding Contact Perception with a Soft Tactile Sensor," 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 2019, pp. 4283-4289.

[19] C. Lu, J. Wang and S. Luo, ”Surface Following using Deep Reinforcement Learning and a GelSight Tactile Sensor”, arXiv:1912.00745v1[cs.RO], December 2019.

[20] Q. Li, C. Schurmann, R. Haschke, and H. Ritter, “A control framework for tactile servoing,” Robotics: Science and Systems, 2013.

[21] Z. Kappassov, JA.C. Ramon and V. Perdereau, ”ZMP Features for Touch Driven Robot Control via Tactile Servo,” in International Symposium on Experimental Robotics, Springer, 2016, pp. 234–243.

[22] D.P. Kingma and J.L. Ba,” Adam: A method for stochastic optimization,” arXiv preprint, arXiv:1412.6980v9 [cs.LG], January, 2017.

[23] Suresh, S., Bauza, M., Yu, K. T., Mangelson, J. G., Rodriguez, A. and Kaess, M. (2020). Tactile SLAM: Real-time inference of shape and pose from planar pushing. arXiv preprint arXiv:2011.07044.

[24] Bauza, M., Canal, O. and Rodriguez, A. (2019). Tactile mapping and localization from high-resolution tactile imprints. In 2019 International Conference on Robotics and Automation (ICRA) (pp. 3811-3817).

[25] Z. Wang, E. P. Simoncelli and A. C. Bovik, ”Multiscale structural similarity for image quality assessment,” The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, 2003, pp. 1398-1402 Vol.2, doi: 10.1109/ACSSC.2003.1292216.

APPENDIX

A: Network Architectures

Entangled UnshearNet: The encoder E compresses the input 256×256-pixel tactile image using five convolutional (Conv) layers, each followed by batch normalization (BN) and rectified linear unit (ReLU) activation layers respectively (architecture in Fig. 2). The output of the last Conv layer ($8 \times 8 \times 64$) is passed to the decoder D which upsamples it to an output 256×256-pixel canonical image PC. Similarly to the Conv layers, all transposed convolutional layers (T-Conv) were followed by BN and ReLU activation layers except the output layer which used sigmoid activation function instead (see also Fig. 2).

This model was used as a baseline to evaluate the performance of proposed *Disentangled UnshearNet* model on its effectiveness in removing distortion in sensor response caused by sliding-induced global shear. For details of the training, please see Appendix B.

Disentangled UnshearNet: The encoder has the same inputs and architecture as the *Entangled UnshearNet*, except the the output of the 5th Conv layer is followed by two additional Conv layers, one each for the two latent codes: Pose ($8 \times 8 \times 64$) and Shear ($8 \times 8 \times 64$).

One decoder DC takes as input pose latent code and upsamples it to 256×256-pixel canonical output tactile image PC. All T-Conv layers were followed by BN and ReLU activation layers except the output layer which used sigmoid activation layer instead (architecture shown in Fig. 3).

The other decoder DS takes as input both pose and shear codes, merges them and upsamples to an output 256×256-pixel sheared image PS (intended to match the encoder input). Apart from the above differences, DS has the same architectures as DC. For details of the training, please see Appendix B.

PoseNet: This model takes as input 256×256 image, compresses it to extract features using the Conv part which are then combined using fully connected (FC) part to predict continuous-value pose components at the output. In total, the network consists of five convolution layers, two max pooling (MP) layers following 2nd and 4th Conv layers; the CONV layers were followed by BN and ReLU activation layers respectively. The 1st FC layer used ReLU activation while the output FC layer used sigmoid activation. For details of the architecture and training see Fig. 5 and Appendix B.

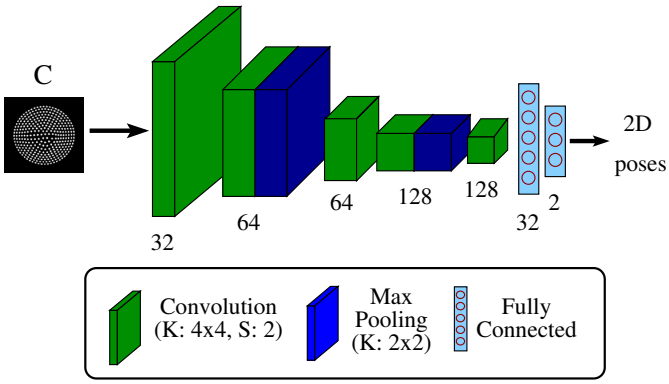

Figure 5: **PoseNet schematic:** This network takes as input, a tapping binary sensor image, C. The convolution part compresses input to extract relevant features which are then combined by fully connected part to predict continuous-valued object pose parameters at output.

B: Training Details

All the inputs (binary 256×256-pixel images) and outputs (binary 256×256-pixel images for image-to-image models, pose parameters for image-to-pose model) were scaled to the range $[0,1]$. All networks weights were initialized from a zero-centered normal distribution with 0.02 standard deviation. Both image-to-image models were trained on data collected from all three stimuli shapes (Fig. 1 (c)) as discussed in section 3.2.

For image-to-image models, we used a batch size of 32. All convolutional/transposed convolutional layers used L1/L2 regularization (10e-4) along with random image shifts, 1-2% of image size to prevent overfitting. The *L2 loss* computed across the entire image and *L1 patch loss* – computed between 100 random crops of size 20×20 of generated unsheared images and corresponding canonical images – were used to train the networks in the ratio 10:1. In case of *Disentangled UnshearNet*, the encoder (E) and shear decoder (DS) were trained using reconstruction loss.

For image-to-pose model, we used a batch size of 256. Like image-to-image models, all layers used L1/L2 regularization (10e-4) along with random image shifts, 1-2% of image size to prevent overfitting. The network was trained via the *L2 loss* computed between predicted and target pose parameters.

We used ADAM optimizer [22] with learning rate of 0.0001, $\beta 1$ = 0.5 and $\beta 2$ = 0.999. We used learning rate scheduling for training. For image-to-image models, the learning rate was reduced to one-fifth and one-tenth after 40 and 80 epochs. The models were trained, in total, for 100 epochs. For image-to-pose model, the learning rate was reduced to half and one-tenth after 100 and 200 epochs respectively. The model was trained for 250 epochs. For all models, the model with best validation accuracy was used for testing.

Finally, training and optimization of the networks was implemented in the Tensorflow 2.0 library on a NVIDIA GTX 1660 (6 GB memory) hosted on a Ubuntu machine.

C: Control Policy for Continuous Contour Following

Local pose estimation allows a robot to maintain contact while safely moving over the object, thus enabling complex robot-object interactions. To demonstrate continuous 2D contour following, we used a simple control policy with following two aims: 1) keep the sensor normal to object surface while in motion and 2) at every time step t, move the sensor tangentially along the surface by a predefined step (0.5 mm in this case). To achieve these aims, a discrete-time proportional-integral (PI) controller was implemented to output a change in the pose of the sensor ($\Delta p(t)$) in its reference frame

$$\Delta p(t) = K_p e(t) + K_i \sum_{t'=0}^{t} e(t')$$

where K_p and K_i are diagonal gain matrices with proportional gain of 0.5 and integral gains of 0.3 and 0.1 for translations and rotations respectively. e was error between the predicted pose and a reference normal to the edge.

D: Disentanglement of Latent Representations

An ablation study was used to verify the separation of latent representations in the *Disentangled UnshearNet* into pose and shear codes respectively. To do this, we passed the 'Shear Code' to the unsheared reconstruction decoder (DC) instead of the 'Pose Code'. As expected, this led to a severe degradation in performance with mean-square error (MSE) between the ground truth images C (tap) and unsheared images PC increasing to 0.22, an order of magnitude higher than original of 0.023. In similar fashion, the SSIM index dropped to 2% when using the Shear Code to reconstruct the unsheared images PC instead of the original 93% when the Pose Code was used. Likewise, asimilar degradation was observed when the 'Shear Code' was replaced by the Pose Code to reconstruct the sheared input S. The mean-squared error between S and the sheared output PS increased to 0.1, which was 50-times the original of 0.002 when both 'Pose and Shear Codes were used to reconstruct PS. The SSIM index showed a similar trend with the similarity dropping from 99.5% to only 11%. This shows that the Shear Code is indeed encoding relevant information required for successful reconstruction of sheared input S. These results clearly demonstrate that the *Disentangled UnshearNet* successfully disentangles the latent representations as desired.

E: High Resolution Results

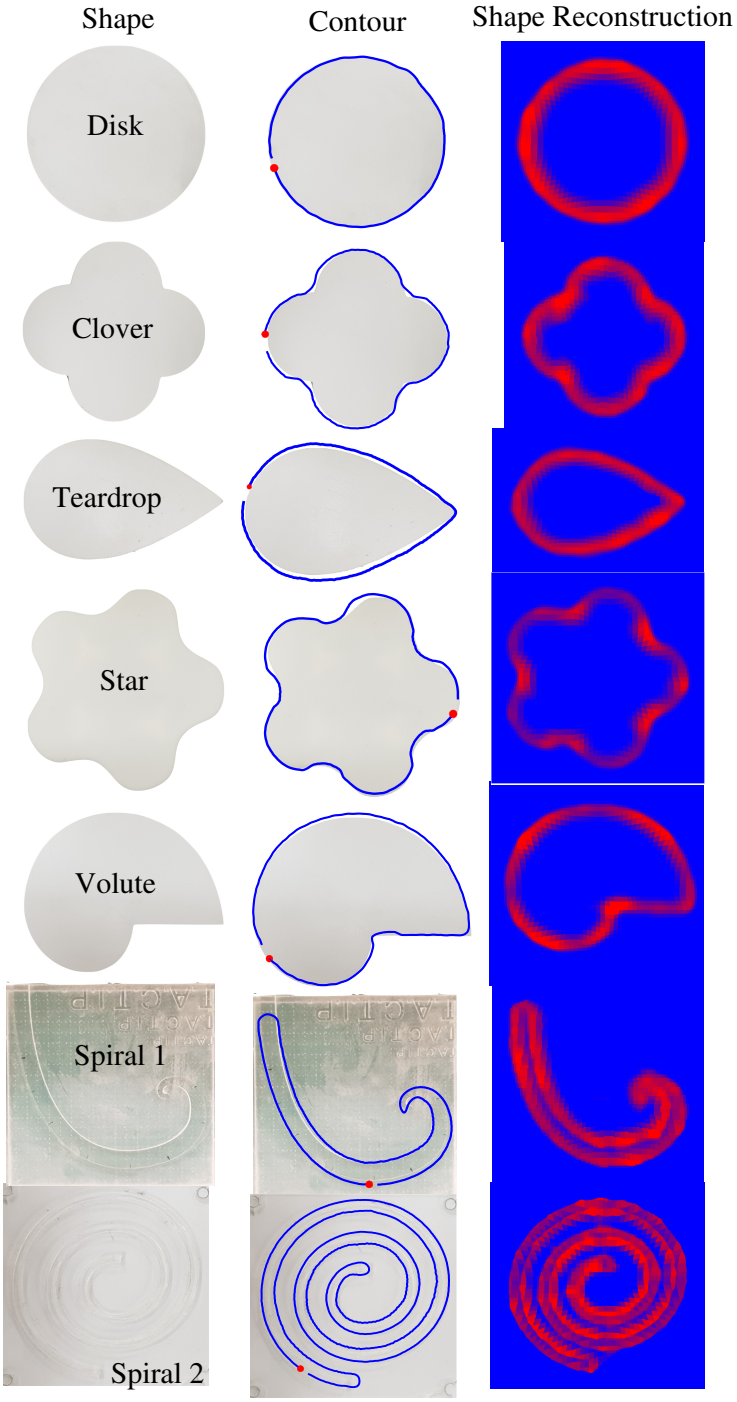

Figure 6: **Contour following & Shape Reconstruction.** Left: Stimuli shapes, Middle: Robust sliding across shapes in left row post removal of shear using *Disentangled UnshearNet*. Red dot shows the starting point/initial contact. Right: Full shape reconstruction post removal of shear using *Disentangled UnshearNet.*