On Explainability and Sensor-Transferability of a Robot Tactile Texture Representation Using a Two-Stage Recurrent Networks

Ruihan Gao*[1], Tian Tian*[2], Zhiping Lin[2], Yan Wu[1]

***Abstract*— The ability to simultaneously distinguish objects, their materials, the associated physical properties and more is one fundamental function of the sense of touch. Recent advances in both the development of tactile sensors and machine learning techniques allow ever more accurate modelling of robotic tactile sensations with increasing complexities. However, many state-of-the-art (SotA) approaches focus solely on constructing black-box models to achieve ever higher classification accuracy. Moreover, each type of tactile sensor produces a unique spatial-temporal data format, making most of the SotA models unable to transfer across sensors. In this work, we propose an explainable and sensor-transferrable recurrent networks (ESTRAN) model for tactile texture representation. The ESTRAN model consists of a two-stage recurrent networks fed by a sensor-specific header network. The first stage of the ESTRAN makes use of the GRUs to decouple sensor-specific information and split the tactile sensations into different frequency response bands similar to our human touch receptors while the second stage codes the overall temporal signature as an LSTM autoencoder. We infuse the latent representation with categorical labels of texture properties (e.g. rough, smooth) to aid representation learning and provide explainability to the latent space. The ESTRAN model is tested on texture datasets collected with two different tactile sensors. Our results show that the model not only achieves higher accuracy, but also provides transferability across the sensors with different sampling frequencies, data formats and texture classes. The addition of the crudely obtained categorical property labels offers a practical approach to enhance the interpretability of the latent space and improve the overall performance of the model.**

I. INTRODUCTION

Being one of the most important sensory modalities in physical interaction, the sense of touch has increasingly been studied for robotic applications in unstructured environments. Tactile sensing is used either as a complimentary or as an integrative sensing modality to vision to infer properties of the environment (e.g. glass v.s. transparent plastic) such as distinguishing textures [1], [2], [3], [4], [5], [6], recognising objects [7], [8], [9] and estimating object poses [10], [11]. It is also explored to enhance interaction control in slip detection [12], dexterous manipulation [13], [14], [15] and compliant interaction [16], [17].

*This research is supported by the Agency for Science, Technology and Research (ASTAR) under its AME Programmatic Funding Scheme (Project #A18A2b0046).

*Authors with equal contributions.

[1]Robotics & Autonomous Systems Department, A*STAR Institute for Infocomm Research, Singapore. Email: {gao_ruihan,wuy}@i2r.a-star.edu.sg

[2]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Email: {tian0090,EZPLin}@ntu.edu.sg

Specifically for texture classification, works done can be mainly divided into two streams: end-to-end learning and inference based on hand-craft features. End-to-end learning are more often applied to texture image classification [18], which gains a boost following the development of Convoluted Neural Network (CNN) [19], [20], [21], [22]. However, these methods are limited to camera-based tactile sensors and can provide little information about the abstract characteristics of the materials. The other stream focuses on extracting hand-crafted features with customized formulas. For example, [2] extracts components of different vibration frequencies based on Fourier coefficients; roughness, traction, and fineness are formulated by sensor pressure readings and motor currents in [3]. Others also work on the discrimination of one specific material property through different hardware designs [23], [24]. Although models with hand-crafted features are naturally more explainable, they are, however, sensitive to the customized formulation and may not be easily generalizable beyond the demonstrated datasets.

Despite the extensive works, most of the proposed methods are sensor-specific. Unlike other sensing modalities which have standardised their data representation formats, that of tactile sensory outputs varies significantly across different sensors which themselves also vary in sensing mechanisms (e.g. capacitive, optical, barometric, piezoresistive), contact medium, shapes and spatial distributions. This also contributes to the level of difficulty to generalise models to other applications which use different sensors.

Inspired by the domain adaptation approach, [25] proposes a weakly supervised recurrent autoencoder framework in an attempt to extract the common features in surface texture properties for classification purpose from data collected from heterogeneous sensors with varied protocols. By minimizing the mean square error between the latent vectors of two heterogeneous datasets, a joint training approach is able to align the latent representation for each class and to improve inference performance over model learned from individual sensor. It thus provides some evidence on the benefits of releasing and using tactile datasets collected on different sensors. However, the unified common latent representation remains obscure to human understanding and the joint training process requires the datasets from different sensors to have the same classes and sample sizes which might not be practical for real-life applications. Moreover, since only the latent representation is common, the learned model cannot be reused on other sensors outside the joint training process. Complete joint training for each pair of datasets is required which further limits the reusability of the framework.

In this work, we propose the explainable and sensor-transferrable recurrent networks (ESTRAN) model, a hierarchical learning-based temporal information encoding scheme with additional categorical labels infused at the latent space in an attempt to address the above issues. This approach splits the processing of the temporal signature of a tactile signal into two stages. The outer stage performs sensor- or application- specific preprocessing, which is drawn from inspiration of the human mechanoreceptors, to extract low-level frequency-based features. The outputs are then transmitted to the inner-stage for sensor-independent temporal signature modelling. This hierarchical approach, thus, allows transfer learning to take place across datasets collected with different sensors, sampling frequencies and setups. To build explainability into the model, we also propose to add further weak supervision at the latent space, by infusing categorical property labels that can be inexpensively obtained from qualitative common-sense knowledge bases or human intuition. The added explainability, in turn, improves the model's learning effectiveness and provides more interpretable classification results. In summary, this paper makes four primary contributions:

- We propose ESTRAN, a hierarchical learning model that is agnostic to sensor sampling frequency and data length;
- The model decouples the sensor-specific feature extraction task from the texture classification one allowing the possibility to perform transfer learning between heterogeneous datasets without the need to completely retrain a fresh model with external dataset.
- With the addition of categorical labels of texture properties, the model achieves higher learning efficiency and enhances the explainability of the representation learning.
- To further validate the efficacy and efficiency of the proposed approach, we expand the number of textures from 20 to 50 and collect twice the number of samples per class to that of our released dataset on the BioTac sensor for training and validation.

The rest of this paper is organized as follows. We introduce the methodology of the proposed framework in Section II, then present the experiments, results and discussions in Section III, and finally draw conclusions in Section IV.

II. METHOD

The overall framework of the ESTRAN is shown in Fig. 1a. It consists of 4 basic components, the Header Network, the Outer Recurrent Stage, the Intermediate Relay and the Inner Recurrent Stage. The MLP/CNN Header Network is used to localise and enrich the features arising from the groups of activated taxels due to a tactile event. The Outer Recurrent Stage (ORS), which uses Gated Recurrent Units (GRUs) operating at distinct frequencies as inspired by human mechanoreceptors, extracts features responding at different temporal scales. The Intermediate Relay integrates the ORS outputs into a unified feature representation before being encoded by the Inner Recurrent Stage (IRS). The IRS consists of a Long Short-Term Memory Variational Autoencoder (LSTM-VAE) with classifiers at the latent space for texture property infusion and texture classification. We map the whole latent representation to the texture labels while assigning the categorical properties to dedicated neurons of the latent vector with a linear mapping and activation. The following sections will introduce the modules in details.

A. The Header Network

The task of texture classification via sliding motion typically involves a small group of taxels around the point of contact between the sensor and the surface while the vast number of other taxels stay dormant. It is thus practical to firstly localise and enrich the spatial tactile features before temporal learning takes place. Thus, the introduction of a header network acting as a spatial attention mechanism can produce a more compact overall model which in turn, can learn faster. A Convolutional Neural Network (CNN) can be used if the taxels are fairly regularly distributed and the total number of taxels is big (e.g. iCub RoboSkin [26]). Conversely, a Multi-Layer Perceptron (MLP) can be employed (e.g. BioTac [27]). Specifically, a convolutional kernel size of (3, 5) is applied to RoboSkin data of input size (6,10) followed by a max pooling as proposed in [5]; a linear layer with input size 19 and output size 18 is applied to BioTac readings of 19 electrodes. Overall, the sensor-specific header network maps the raw input readings at each time step into a fixed-sized output (18 in our case) and accomplishes spatial compression.

B. The Outer Recurrent Stage

The output of the header network at the sensorś output frequency will be fed into a parallel number of GRUs to extract responses at different temporal resolutions, similar to the motivation of clockwork Recurrent Neural Network (RNN) [28]. GRU is empirically chosen based on computational efficiency and performance. We also introduce a residual factor when initialising the GRU cell in between receiving consecutive segments of one sequence, i.e.

$$h_0(n+1) = h_T(n) * r \tag{1}$$

where n represents the index of segments, T represents the length of the preceding input segment to a receptor, and r represents the scaling number of residual.

To balance the temporal resolution and computational efficiency, we implement only two GRU units, namely "fast" GRU (F-GRU) and "slow" GRU (S-GRU), and determine the exact frequency empirically based on the sampling frequency of two sensors used in the experiments. For example, if the input signal is $100Hz$ and if the GRU is sensitive to $20Hz$ stimuli, every 5 consecutive time samples will form a "complete" sequence as inputs to this GRU. The hidden state of the 5^{th} time step will be transmitted to the Intermediate Relay. More details of parameter tuning are illustrated in Section III.

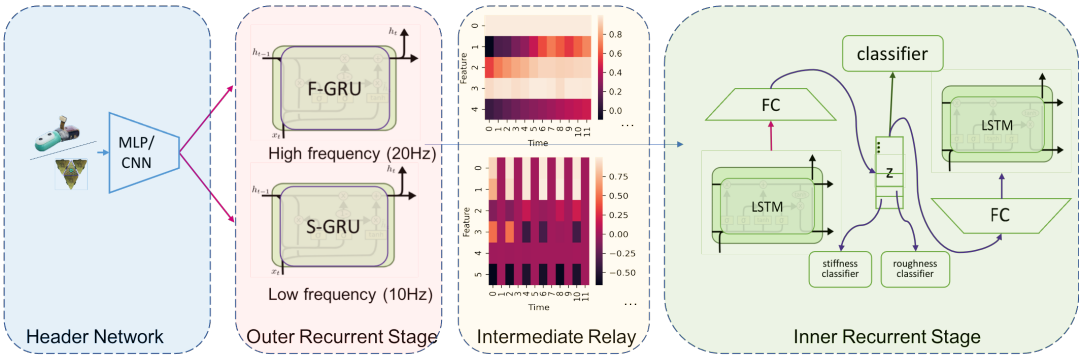


Fig. 1: Overview of the proposed ESTRAN model which encodes the tactile signal into a hierarchical recurrent network representation. Its latent space, which is used to perform texture classification, is infused with categorical texture property labels.

C. Intermediate Relay

As the GRUs at ORS output responses at different frequencies, we use F-GRU as the pacemaker, setting the input frequency for processing at IRS. S-GRU output is aligned to F-GRU output at the closest time-step. For all other time-steps, S-GRU outputs are treated as zero. The yellow box in Fig. 1a shows a sample feature map for F-GRU receptor and S-GRU, respectively. The horizontal axis represents time scale, and the vertical axis represents features. Columns of uniform color in S-GRU feature map represent the zero-padding when F-GRU outputs but S-GRU does not.

D. The Inner Recurrent Stage

The IRS consists of three components: the LSTM to model the sensor-invariant temporal dynamics, the variational autoencoder to mimic sensory imagery and the classifiers to texture property infusion and texture discrimination.

1) LSTM: receives input signals from the Intermediate Relay. The hidden state at the last time step is extracted as the output feature and is passed to a fully connected (FC) layer to obtain the latent space vector. The hidden size and latent representation size follow the implementation in [25], i.e. 90 and 40, respectively.

2) Variational autoencoder: is to reconstruct the sensory imagery. Practical consideration in having the autoencoder is to provide added reconstruction constraints to improve classification performance. We reconstruct the signal only at IRS to learn a sensor-agnostic representation and use a Gaussian prior for the latent space.

3) Classifier: A standard linear layer is implemented to map the latent representation to a distribution over C texture classes. Multi-class cross-entropy loss is minimized between output vector and ground truth texture label y.

To enhance the explainability of the proposed model, we provide categorical labels of the texture properties at the latent representation. In this work, roughness and stiffness are chosen for illustration based on human common sense and previous texture classification work using texture properties [29], [30]. We use -1, 0, 1 as a coarse grading, e.g. for roughness, -1 for smooth, 1 for rough, and 0 for moderate or medium-scale. Such labellings can be inexpensively obtained from knowledge bases or human intuition. For each property, we map one neuron of the latent vector to the property label via a linear transformation with activation such that the categorical labels does not impose unbalanced scaling or discretize the latent space. The property labels are obtained by averaging over human common sense. To provide unbiased benchmark, ten participants (five males and five females) are invited to label the material properties in terms of roughness and stiffness. These qualitative labels are then averaged to obtain the final categorical label for each texture.

E. Implementation and Evaluation Metrics

The proposed model is trained and tested on texture classification and property infusion in two groups of experiments: end-to-end training on individual datasets and transfer learning between heterogeneous sensor datasets. A common loss function Loss is used. It is a weighted sum of texture classification loss L_{tcl}, property loss for roughness L_{pr} and stiffness L_{ps}, reconstruction loss L_r, and KL divergence loss L_{kl}:

$$\begin{aligned}\text{Loss} = w_{tcl} * L_{tcl} + w_{pr} * L_{pr} + w_{ps} * L_{ps} \\ + w_r * L_r + w_{kl} * L_{kl}\end{aligned} \tag{2}$$

where w_{tcl}, w_{pr}, w_{ps}, w_r, and w_{kl} are the weights for texture classification loss, roughness property loss, stiffness property loss, reconstruction loss, and KL divergence loss, respectively. Details of the model parameter choices are described in EXPERIMENTS.

The texture classification uses standard cross-entropy loss and aims to maximize the probability of the ground-truth label y_i given a data point x_i.

$$L_{tcl} = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}\mathbb{I}(c, y_i)\log(p_{x_i,c}) \tag{3}$$

where N is the number of data points, C is the number of classes, $p_{x_i,c}$ is the predicted probability of input x_i being of class c.

For property infusion, we map one neuron of the latent vector to a property label to a linear layer with activation

and minimize the mean square error between the mapping output and the property label. The neurons chosen for dedicated property learning are the first neuron (index 0) for roughness property and the second neuron (index 1) for stiffness property, without loss of generality:

$$\begin{aligned} \hat{z_{0_i}} &= z_{0_i} * a_0 + b_0 \\ \hat{z_{1_i}} &= z_{1_i} * a_1 + b_1 \\ L_{pr} &= \frac{1}{N}\sum_{i=1}^{N}(\hat{z_{0_i}} - y_{r_i})^2 \\ L_{ps} &= \frac{1}{N}\sum_{i=1}^{N}(\hat{z_{1_i}} - y_{s_i})^2 \end{aligned} \tag{4}$$

where z_{j_i} is the index j neuron of the latent vector z of the i^{th} sample, $\hat{z_{j_i}}$ represents corresponding mapping output of z_{j_i} after a linear transformation and activation, a_j and b_j are learnable parameters of linear transformations, and y_{r_i} and y_{s_i} are the roughness label and stiffness label for the i^{th} sample, respectively.

The reconstruction loss is set to be the mean square error between input and the reconstructed input, and aims to maintain fidelity to input data. In this work, the reconstruction loss is implemented at two layers in the IRS processing, i.e. input to the LSTM and input to the FC layer of variational autoencoder.

$$L_r = \frac{1}{N}\sum_{i=1}^{N}(h_i - \hat{h}_i)^2 \tag{5}$$

where h_i represent the hidden units at the input to the LSTM and input to the FC layer of variational autoencoder, while $\hat{h}_i$ represent the corresponding reconstructed units, respectively.

The KL-divergence loss is implemented for the variational autoencoder and is minimized to reduce the disparity between the encoder's distribution $q_\theta(z|x)$ and the true posterior distribution $p(z)$ [31].

$$\begin{aligned} L_{kl} &= \frac{1}{N}\sum_{i=1}^{N} D_{KL}(q_\theta(z|x_i)\|p(z)) \\ &= \frac{1}{N}\sum_{i=1}^{N}(-0.5 * (1 + \log(\sigma_i^2) - \mu_i^2 - \sigma_i^2)) \end{aligned} \tag{6}$$

where q represents the encoder network, which makes an inference about z based on input x_i, and μ and σ are the output of the approximated distribution.

Final classification performance is measured by:.

$$\text{Acc} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y_i})^2 \tag{7}$$

III. EXPERIMENTS

This section presents the experimental setup, the experiments, results and discussions. In summary, we performed a benchmark study against the RAEC model in [25] using a number of datasets. To further understand the effect of the explainable property infusion, a variant version of the RAEC model was also implemented with texture property infusion at its latent space.

A. Datasets

Apart from using the datasets released in [25] for benchmarking, we collected a new set of data of 50 texture classes with 50 samples for each class. This dataset is collected under the same data collection protocol for the KUKA iiwa robot attached with the BioTac sensor. Twenty of the classes overlap with the existing dataset. The additional textures range from cardboard to luggage belt. Snapshots of all the textures are shown in Fig. 2 and a detailed documentation of each material is accessible online*. The rest of this paper follows the naming convention in Table I. In particular, c20icub and c20BT refers to the existing datasets on RoboSkin and BioTac respectively; c20BTcombined combines the data of the 20 common classes with that of the previous dataset; c50BT is the complete set of newly collected data of the 50 textures.

Fig. 2: Snapshots of the 50 materials.

TABLE I: Dataset specifications

Dataset	No. of classes	Sensor	No. of samples per class
c20icub	20	RoboSkin	50
c20BT	20	BioTac	50
c20BTcombined	20	BioTac	100
c50BT	50	BioTac	50

B. Preliminaries

In this work, we heuristically determined the output size of the Header Network to be 18, and $r = 1$ for the residual constant in GRUs. The loss weights are chosen empirically as follows so that the loss terms are of similar order of magnitude: $w_{tcl} = 1$, $w_r = 0.001$, $w_{kl} = 0.0005$. For w_{pr} and w_{ps}, since the datasets vary in data formats and therefore have different sensitivity to the properties, while a common set of property weights for all datasets can be found for good model performance, we performed grid search to determine the optimal property weights for each dataset as shown in

*https://github.com/RuihanGao/Bio-inpired-MTR-TRAN.git

TABLE II: The optimal property weights for the datasets.

Dataset	Model	w_{pr}	w_{ps}
c20icub	ESTRAN	1	1
c20icub	RAEC	0.5	1
c20BT	ESTRAN	0.2	0.5
c20BT	RAEC	0.2	1
c20BTcombined	ESTRAN	0.2	0.5
c20BTcombined	RAEC	1	2
c50BT	ESTRAN	1	1
c50BT	RAEC	1	1

TABLE III: Preliminary results of different combinations of fast(F)/slow(S) GRU operating frequencies

F-GRU/Hz \ S-GRU /Hz	1	2	5	10
20	0.845	0.840	0.865	**0.875**
40	0.865	0.860	0.860	0.870
60	0.780	0.770	0.865	0.860
80	0.810	0.765	0.790	0.805
100	0.815	0.825	0.77	0.785

Table II. The exact property contributes to the models will be presented in the subsequent section. For ORS, based on the sampling frequencies of the sensors (BioTac at $100Hz$, RoboSkin at $50Hz$), the sampling range of $20-100Hz$ and $1-10Hz$ for F-GRU and S-GRU respectively are used to search for the frequency pair with the best performance. Each combination runs for 20 epochs on c20 dataset. Based on the results shown in Table III, the combination of GRU frequencies are chosen to be $20Hz$ and $10Hz$ with hidden size of 100 and 200 respectively.

C. Experiment 1: End-to-end training on individual datasets

We benchmark the performance of the proposed ESTRAN model against the RAEC model using the same latent dimension in [25]. Ablation study is conducted to determine the individual contribution of additional GRU units and property labels. The results are shown in Table IV. The dataset is randomly split for a 5-fold validation, with the ratio of train, validation, and test size set to 6:2:2. All experiments are run for 100 epochs.

In Table IV, the first row represents the performance of the original RAEC model and the last row represents that of the proposed ESTRAN model with property labels. In general, the full ESTRAN model outperforms all other model combinations in all datasets.

The contribution of the ORS layer with frequency response modelling can be seen from the results comparison between row 1 and row 3. We can see that the ORS layer improves performance for all datasets while it is particularly significant for datasets collected from the BioTac sensors. As the data collection process for BioTac sensor is so much stricter than that of the RoboSkin on iCub, the frequency response for the BioTac datasets is expected to involve much less noisy. Moreover, as BioTac operates at a higher frequency than RoboSkin, the frequency-based response at different bands can capture more enriched characteristics of textures.

To understand the contributions of the inexpensively obtained texture properties, we can observe performance accuracy for each model toggled with property infusion. In short, performance accuracy improves with the addition of property infusion. This is expected as more information has been provided to the model although the information provided is crudely obtained. However, the RoboSkin dataset enjoys greater benefit from property labels. This is because the weak supervision at the latent space provides more distillation for the noisy RoboSkin dataset collected without strict force or velocity control.

Comparison among the last three columns (c20BT, c20BTcombined, and c50BT) demonstrates that the performance improvement achieved by the proposed model is rather invariant to the change in number of data samples (c20BT vs c20BTcombined) and change in the number of classes (c20BT vs c50BT).

Since comparable results can be achieved on a few datasets, we take c20BT as an example to show the confusion matrix and visualization of the latent space.

Fig. 3 shows the illustrative confusion matrix of ESTRAN model trained on c20BT dataset with property labels. Five folds of data are aggregated to demonstrate the overall performance on the whole dataset. The result is nearly an identity matrix with few scattered samples.

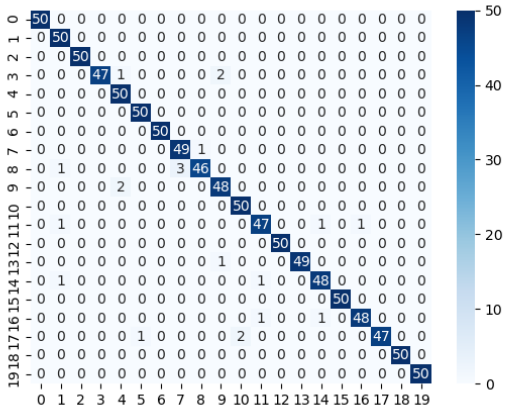


Fig. 3: Confusion matrices of models trained on c20BT

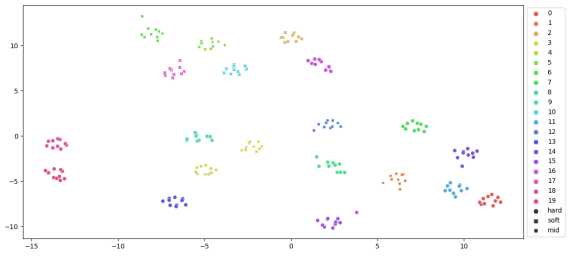
Since tactile sensor readings are multi-dimensional and sequential in time domain, the t-distributed stochastic neighbor embedding (t-SNE) plots are generated to visualise the distribution of the extracted features in the latent space. Fig. 4 shows the sample plots of c20BT dataset for ESTRAN model, with and without property labels. Colors represent different materials and the shapes represent the texture properties. It is shown that the different textures can be clearly separated in the latent space and they can be grouped better with additional property labels.

D. Experiment 2: End-to-end training with drastically reduced latent space dimension

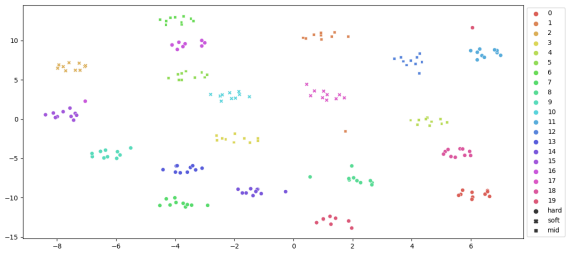
When the latent space is large, it is possible that similar information is redundantly encoded by several neurons.

TABLE IV: The mean (standard deviation) % accuracy of end-to-end training results on individual datasets with dimension of latent space = 40.

Model	with property?	c20icub	c20BT	c20BTcombined	c50BT
RAEC	False	88.5 (4.41)	88.3 (1.57)	90.8 (2.25)	89.3 (1.27)
RAEC	True	90.7 (3.86)	90.0 (2.60)	92.8 (1.14)	91.0 (2.01)
ESTRAN	False	89.8 (3.91)	94.9 (1.63)	95.9 (0.73)	94.0 (2.24)
ESTRAN	True	**91.7 (4.85)**	**95.6 (2.02)**	**96.7 (0.67)**	**95.0 (1.51)**



(a) t-SNE plot for ESTRAN with property labels



(b) t-SNE plot for ESTRAN without property labels

Fig. 4: Visualization of latent space vectors by t-SNE

Therefore, pushing one neuron to learn a specific property may not offer great advantages. To further investigate on the information gain from the crudely obtained property labels, we drastically reduce the latent space to one-tenth $(1/10)$ of the original dimension size and observe how performance changes when the model is trained with and without property labels. Moreover, as the ORS provides a simpli The results are shown in Table V.

We can observe from both Tables IV and V that when the latent space size is drastically reduced, the performance of most models deteriorate as expected. This is especially true for the base RAEC model. For the ETRAN model, the performance for the last two datasets seem to produce even better results without the crude property labels while the dimensionality is drastically reduced. However, this is inconclusive as the performances are well within 1 standard deviation from each other. One possible explanation is that the BioTac datasets have high signal-to-noise ration, by adding the crude property labels when the number of neurons are reduced to a small fraction, the confusion from these noisy labels has much stronger influence to the model performance. As such, the converse seems to also hold for the RoboSkin dataset which brings the model performance back on track.

E. Experiment 3: Transfer learning on heterogeneous sensor datasets

One of the problems that we aim to tackle by introducing GRUs at the Outer Recurrent Stage is that the RAEC approach only shares the common representation at the latent space between sensors. It does not provide a common model to be used or transferred. Therefore, we treat GRUs as sensor-based adaptors and that split the input data into different frequency bands and decouple the sensor-specific information. With the reconstruction loss and classification loss added to the LSTM variational autoencoder, the Inner Recurrent Stage is dedicated to learn common temporal signatures that are sensor-agnostic. This will allow the Inner Recurrent Stage model to be transferred across heterogeneous tactile sensors. To examine the transferrability of the proposed model, we first train the ESTRAN model on BioTac dataset. The ESTRAN model for the RoboSkin dataset is then initialised with the Inner Recurrent Stage copied from that of the BioTac dataset model, leaving the header network and the Outer Recurrent Stage to for training. Since the GRUs represent the sensor-specific conversion, they are assumed to be able to learn and adapt well for data mapping. Should transferrability occurs, the RoboSkin dataset will be able to learn with similar performance results with the IRS fixed at the model parameters used for the BioTac dataset.

For the BioTac model, we follow the aforementioned parameter setting and evaluate the transferrability for models trained using c20BT. c20BT has the same number of samples and same texture classes as c20icub and is used to benchmark the sensor plasticity under the same amount of input information. Fig. 5 shows the speed of convergence of models 1) trained from scratch 2) transferred from models pre-trained on BioTac dataset.

The model transferred from c20BT achieves a test accuracy of 92.3%, about 0.5% increase compared to the baseline model trained on c20icub from scratch. Fig. 5 also shows that the transferred model (yellow line) maintains a sightly higher position than the baseline model (blue line), which demonstrates that the higher-level temporal texture representation is indeed sensor-invariant and can be used to facilitate training across heterogeneous datasets.

IV. CONCLUSION

In summary, this work proposed ESTRAN, a two-stage recurrent networks for tactile texture representation learning with enhanced explainability and transferrability across heterogeneous datasets. It consists of a sensor-specific header

TABLE V: The mean (standard deviation) % accuracy of end-to-end training results on individual datasets with dimension of latent space = 4.

Model	with property?	c20icub	c20BT	c20BTcombined	c50BT
RAEC	False	84.8 (9.56)	85.4 (2.44)	88.4 (2.67)	78.5 (0.87)
RAEC	True	88.3 (4.34)	85.6 (2.73)	87.9 (2.09)	83.1 (2.06)
ESTRAN	False	88.7 (6.31)	92.3 (4.00)	**96.2 (1.19)**	**94.3 (2.08)**
ESTRAN	True	**90.9 (4.71)**	**95.3 (1.47)**	95.2(1.34)	93.5 (2.79)

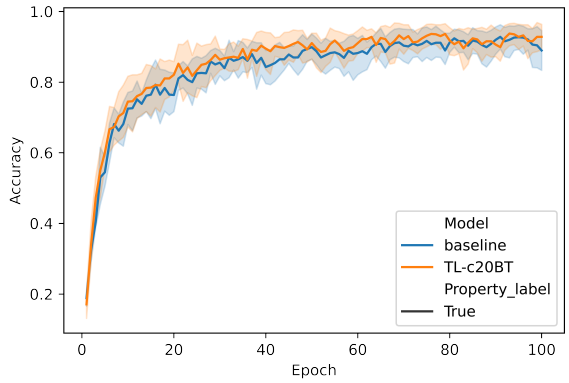


Fig. 5: Comparison of the speed of convergence for transfer learning

network, an outer recurrent stage for resampling, and an inner recurrent stage to learn sensor-invariant temporal signatures. The results show a reasonable improvement in classification accuracy and indicate robustness across different textures. We demonstrated that by driving the explainable labels to influence on the learning of the latent neurons towards the physical meanings, the overall model is able to improve performance accuracy. With the introduction of a sensor-dependent ORS layer, we also demonstrated that transfer learning is able to take place between heterogeneous tactile sensor datasets. As a result, strict experimental conditions that improve learning results as well as strict constraint for joint training are no longer required under the ESTRAN model, which allows for higher generalizability and is more pragmatic for real-life applications. Moreover, we have also released a larger tactile texture dataset for the community.

Future work includes the investigation into making the latent vector even more explainable by introducing an increasing number of inexpensively obtained property labels, having cross validations with more types of sensors and a greater range of textures to test the efficacy of the ESTRAN model. Moreover, combination of different exploratory movements, class-incremental learning, and real-time active exploration would be attempted for a more comprehensive approach for texture perception.

REFERENCES

[1] C. W. Fox, B. Mitchinson, M. J. Pearson, A. G. Pipe, and T. J. Prescott, "Contact type dependency of texture classification in a whiskered mobile robot," *Autonomous Robots*, vol. 26, no. 4, pp. 223–239, 2009.

[2] N. Jamali and C. Sammut, "Material classification by tactile sensing using surface textures," in *2010 IEEE International Conference on Robotics and Automation*, pp. 2336–2341, IEEE, 2010.

[3] J. A. Fishel and G. E. Loeb, "Bayesian exploration for intelligent identification of textures," *Frontiers in neurorobotics*, vol. 6, p. 4, 2012.

[4] S. Luo, W. Yuan, E. Adelson, A. G. Cohn, and R. Fuentes, "Vitac: Feature sharing between vision and tactile sensing for cloth texture recognition," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2722–2727, IEEE, 2018.

[5] T. Taunyazov, H. F. Koh, Y. Wu, C. Cai, and H. Soh, "Towards effective tactile identification of textures using a hybrid touch approach," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 4269–4275, IEEE, 2019.

[6] T. Taunyazov, W. Sng, H. H. See, B. Lim, J. Kuan, A. F. Ansari, B. C. Tee, and H. Soh, "Event-driven visual-tactile sensing and learning for robots," *arXiv preprint arXiv:2009.07083*, 2020.

[7] M. Kaboli, R. Walker, G. Cheng, *et al.*, "In-hand object recognition via texture properties with robotic hands, artificial skin, and novel tactile descriptors," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pp. 1155–1160, IEEE, 2015.

[8] J. Hoelscher, J. Peters, and T. Hermans, "Evaluation of tactile feature extraction for interactive object recognition," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pp. 310–317, IEEE, 2015.

[9] H. Liu, Y. Yu, F. Sun, and J. Gu, "Visual–tactile fusion for object recognition," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 996–1008, 2016.

[10] J. Bimbo, S. Luo, K. Althoefer, and H. Liu, "In-hand object pose estimation using covariance-based tactile to geometry matching," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 570–577, 2016.

[11] N. Kuppuswamy, A. Castro, C. Phillips-Grafflin, A. Alspach, and R. Tedrake, "Fast model-based contact patch and pose estimation for highly deformable dense-geometry tactile sensors," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1811–1818, 2019.

[12] J. W. James, N. Pestell, and N. F. Lepora, "Slip detection with a biomimetic tactile sensor," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3340–3346, 2018.

[13] C.-H. King, M. O. Culjat, M. L. Franco, C. E. Lewis, E. P. Dutson, W. S. Grundfest, and J. W. Bisley, "Tactile feedback induces reduced grasping force in robot-assisted surgery," *IEEE transactions on haptics*, vol. 2, no. 2, pp. 103–110, 2009.

[14] H. Van Hoof, T. Hermans, G. Neumann, and J. Peters, "Learning robot in-hand manipulation with tactile features," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pp. 121–127, IEEE, 2015.

[15] Y. Chebotar, K. Hausman, Z. Su, G. S. Sukhatme, and S. Schaal, "Self-supervised regrasping using spatio-temporal tactile features and reinforcement learning," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1960–1966, IEEE, 2016.

[16] M. Fritzsche, N. Elkmann, and E. Schulenburg, "Tactile sensing: A key technology for safe physical human robot interaction," in *Proceedings of the 6th International Conference on Human-robot Interaction*, pp. 139–140, 2011.

[17] A. Roncone, M. Hoffmann, U. Pattacini, L. Fadiga, and G. Metta, "Peripersonal space and margin of safety around the body: learning visuo-tactile associations in a humanoid robot with artificial skin," *PloS one*, vol. 11, no. 10, p. e0163713, 2016.

[18] K. I. Kim, K. Jung, S. H. Park, and H. J. Kim, "Support vector machines for texture classification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 11, pp. 1542–1550, 2002.

[19] V. Andrearczyk and P. F. Whelan, "Using filter banks in convolutional neural networks for texture classification," *Pattern Recognition Letters*, vol. 84, pp. 63–69, 2016.

[20] S. S. Baishya and B. Bäuml, "Robust material classification with a tactile skin using deep learning," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8–15, IEEE, 2016.

[21] W. Yuan, Y. Mo, S. Wang, and E. H. Adelson, "Active clothing material perception using tactile sensing and deep learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4842–4849, IEEE, 2018.

[22] G. Cao, Y. Zhou, D. Bollegala, and S. Luo, "Spatio-temporal attention model for tactile texture recognition," *arXiv preprint arXiv:2008.04442*, 2020.

[23] L. Qin and Y. Zhang, "Surface roughness discrimination using unsupervised machine learning algorithms," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 854–857, IEEE, 2017.

[24] A. Parvizi-Fard, N. Salimi-Nezhad, M. Amiri, E. Falotico, and C. Laschi, "Sharpness recognition based on synergy between bio-inspired nociceptors and tactile mechanoreceptors," *Scientific reports*, vol. 11, no. 1, pp. 1–13, 2021.

[25] R. Gao, T. Taunyazov, Z. Lin, and Y. Wu, "Supervised autoencoder joint learning on heterogeneous tactile sensory data: Improving material classification performance," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (Las Vegas, USA), IEEE, 2020.

[26] A. Schmitz, M. Maggiali, L. Natale, B. Bonino, and G. Metta, "A tactile sensor for the fingertips of the humanoid robot icub," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2212–2217, IEEE, 2010.

[27] J. A. Fishel and G. E. Loeb, "Sensing tactile microvibrations with the biotac—comparison with human sensitivity," in *2012 4th IEEE RAS & EMBS international conference on biomedical robotics and biomechatronics (BioRob)*, pp. 1122–1127, IEEE, 2012.

[28] J. Koutnik, K. Greff, F. Gomez, and J. Schmidhuber, "A clockwork rnn," in *International Conference on Machine Learning*, pp. 1863–1871, PMLR, 2014.

[29] M. Rasouli, Y. Chen, A. Basu, S. L. Kukreja, and N. V. Thakor, "An extreme learning machine-based neuromorphic tactile sensing system for texture recognition," *IEEE transactions on biomedical circuits and systems*, vol. 12, no. 2, pp. 313–325, 2018.

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

[31] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.