PVT^3: A Pruned Video-Vision Transformer for Tactile Texture Classification

Yanjia Ouyang[1,3], Ruihan Gao[1,2], Zhiping Lin[3], Yan Wu[1]

***Abstract*— With the increased traction in deploying robots in less structured environments, the perceptual capacity of a robot to understand the environments needs to be scaled both in depth and breath. The sense of touch provides a robot the ability to feel the environment while physically interacting with it. One important property of a touch feeling is the understanding of different textures which can be seen as a unique representation distilled from the complex spatial-temporal interactions between the mechanoreceptors and the environment. In this paper, we propose a light-weight pure PVT^3, Transformer-based architecture to model the texture representation. PVT^3 uses the Video-Vision Transformer as the backbone with multi-dimensional model pruning to limit the model complexity and size without sacrificing the performance. The PVT^3 model is tested on texture datasets collected on three different tactile sensors each with a distinct collection method. Our results show that the proposed model not only achieves higher accuracy, but also smaller model size as compared to a state-of-the-art tactile texture model on all three datasets.**

I. INTRODUCTION

In an unstructured environment such as a natural scene or a human daily living environment, it is difficult for human and robots alike to depend on single sensing modality to understand and manipulate the environment. The sense of touch is indispensably one important sensing modality to complement vision and audition in the low-latency understanding of material properties [1], [2], surface textures [3], [4], detailed shapes [5], [6], [7] and manipulability of the interacted objects [8], [9]. The last decade has seen more extensive research in tactile perception and tactile-guided manipulation owing to both the development of new sensor technologies [10], [11], [12] and the maturity of advanced machine learning algorithms [13], [14], [15].

One unique feature of tactile sensing against conventional sensory modalities such as vision and audition is that tactile sensing is an interactive sensing modality by nature. It observes the environment only when it manipulates. Although there have been many attempts in classifying such environment properties, such as surface textures by collecting tactile texture datasets, many follow strict control protocols in order for their proposed model to work well [14]. However, it is intriguing that the tactile understanding of the environment is not significantly impacted by how the environment is manipulated. For example, so long as our fingers are moving across a surface regardless of speed, moving pattern, exerted force and so on, we can tell the texture of the surface fairly accurately. This seems to suggest that there is a strong self-attention to correlate between the spatial-temporal tactile signals in representing the touch properties.

To tackle the spatial-temporal correlation embedded in tactile data, existing works attempt to provide modelling treatments the spatial and temporal components respectively and integrate as an overall learning framework to optimise for model performance. In [16], the spatial information is compressed using statistical aggregates to allow a sparse Gaussian Process to model the temporal signatures. The use of Convoluntional Neural Network (CNN) to model the spatial dimension has been a recent common practice while treatments to the temporal dynamics are different. [17] directly concatenates the spatial-temporal data as a 2-dimensional array to be learned by a CNN. [14] uses a Long-Short Term Memory (LSTM) to model the temporal dimension while [18] uses temporal attention mechanisms to localise the temporal signatures. Recurrent autoencoder structures are used to extract a latent tactile representation with LSTM as the backbone in [19] and Gated Recurrent Unit (GRU) hierarchically stacked with LSTM in [20] to enhance generalisability across datasets collected from different tactile modalities. However, given the limited supervising signal from class labels, these models with a long chain of learning pipeline may not generalise well especially if the spatial resolution is to increase. Moreover, by modelling the spatial and temporal aspects independently, it is difficult to provide global-level of attention mechanism for the model to pick up the most important spatial-temporal features. Attempts in vision field demonstrate the potential of using transformer architecture to learn spatial and temporal information simultaneously with embedded attention layer. [21] proposes a pure-transformer based models for video classification with three variants focusing on different ways of factorising components of the transformer encoder over the spatial and temporal dimensions. It extends capabilities of transformer to video processing but still depends on image-pretrained models that are unavailable yet for tactile sensing.

Another drawback of previous works is large model size that affects both training time and inference speed. Recurrent neural networks (RNN) takes input sequences step by step and suffer from vanishing gradients and slow training. Moreover, given the relatively small size of available tactile datasets, concatenating different network modules also

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

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

[2]Robotics Institute, SCS, Carnegie Mellon University, USA. Email: ruihang@andrew.cmu.edu

[3]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Email: {e180169,EZPLin}@ntu.edu.sg

introduces difficulties of tuning each block without intermediate supervision. Since transformer enables fast training speed by parallelizing sequential data and solves long-range dependencies with multi-head attention, we leverage on its capabilities and propose PVT^3, a pure-transformer based model for processing multi-modal tactile data processing. It uses the Video-Vision Transformer as the backbone and maintains a compact model size using multi-dimensional model pruning.

In summary, the main contributions of this paper are as follows:

- We propose the architecture PVT^3 for modelling texture representation with a ViViT [21] as the backbone and model pruning for complexity optimization.
- We demonstrate and benchmark the generalization capabilities of PVT^3 on three tactile datasets, each collected on a sensor of unique modality and different trajectories of exploratory motion.
- Besides two open-source datasets, we collect and release a new dataset of GelSight [11] sensor rolling on 45 textures, which provides additional perspective for texture representation learning.

The rest of this paper is organised as follows. We introduce the related work of video transformer and ways of model pruning in Section II, describe the methodology in Section III, then present experiments, results and discussions in Section IV, and finally draw conclusions in Section V.

II. RELATED WORK

A. Video Transformer

Transformer was first proposed to solve machine translation problem [22], and subsequently applied to audio processing [23]. It is a sequence-to-sequence model that uses encoder-decoder structure and embeds attention layers to compute the relation between each part of the time-series data. Featuring its strong capability of learning global correlation, transformer is also applied to image classification tasks and one of the prominent models is ViT [24], which directly applies a pure transformer to sequences of image patches. Works further extend the sequential information to temporal dimension and propose similar transformer architecture for video processing, e.g. ViViT model [21], which studies efficient ways to factorize the spatial and temporal component in input data.

As discussed in [24], transformer structures have less focus on local information. Compared to CNN, the main component of self-attention in transformer does not enforce many invariant properties. This property may hinder the performance of the model on small datasets since minimum inductive bias is provided by the transformer structure. [25], [26] propose to include extra feature selection block and build a hybrid architecture to enhance model capability for computer vision tasks. For application in tactile sensing, since taxel density and distribution are typically very sparse, and sensor drifting occurs unavoidably at individual taxel level, CNN treatment may not be uniformly beneficial to tactile understanding with all sensors.

B. Model Pruning

Despite its advantages in dealing with sequential data, transformer is often criticized for its large number of model parameters and memory requirement. Especially for data of long sequences, larger models are often required to fully process the necessary information. To tackle this, a number of approaches have been proposed to reduce the model size and speed up computation, including quantization [27], [28], knowledge distillation [29] and model pruning [30], [31], [32], [33].

Quantization utilize matrix factorization, vector factorization or other algorithms to lower bitwidths when storing tensors. By reducing the bitwidths to 8 bits or smaller, memory will be saved by 4 times or higher[28]. Knowledge distillation can make small models have the same performance as large models. The large model will be replaced by a small model with a completely different structure, saving dozens of times the number of parameters.

Le Cun et al.[34] introduced pruning to reduce model complexity by removing connections. Typically, specific algorithms are used to define pruning criteria, and the compressed model will be repeatedly pruned and retrained to maintain performance. Two types of pruning are unstructured pruning and structured pruning. The former has more degrees of freedom and can prune more parameter weights. The latter changes the model structure and therefore reduces the model size.

Liu et al. [35] have demonstrated that for CNN models, pruning can be effective in four levels, namely weight-level, kernel-level, channel-level and layer-level. Drawing inspiration from [35], transformer-based architectures can also consider these pruning strategies to improve on model efficiency. Weight-level pruning is easy to implement and very flexible. However, such kind of unstructured pruning can only increase model sparsity but hard to speed up training unless using special software or hardware implementation[36]. Layer-level pruning[37] in the contrary does not require any further changes in software or hardware, but it is not often used for shallow neural network. Kernel-level and Channel-level pruning are both structural pruning but in different positions. In transformer model, MHSA and MLP layers consume most of the computing efforts. Trimming off excessive heads, patches or dimensions seems like a good idea to trim down the model while maintaining high performance[32][30][31].

III. METHODOLOGY

In this section, we begin by giving a brief description of the backbone structure of vision transformer and then introduce several ways of model pruning to reduce the model size. Finally, we describe details of our proposed mechanism to select features to be pruned. The model structure diagram is shown in Figure1.

A. Video-Vision Transformer Backbone

Vision transformer takes in image inputs and split them into fixed-size small patches. Those patches are then con-

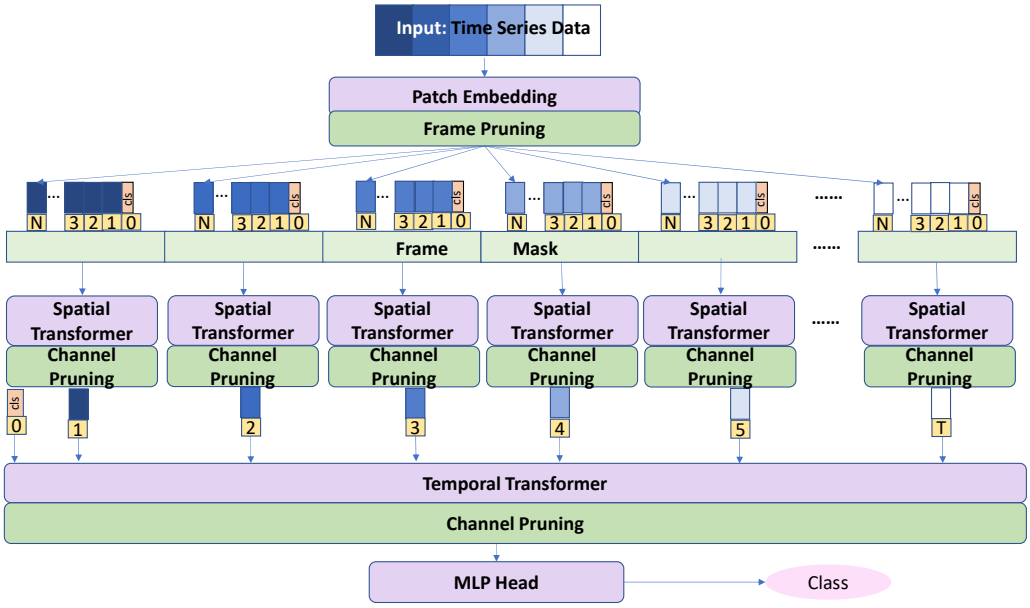

Fig. 1. Overview of the PVT^3 model architecture. It consists of a factorised encoder version of the Video-Vision Transformer as the backbone inserted two types of model pruning at both channel and frame levels.

verted into a 1-D vector using fully connection layer and passed to a classic transformer encoder block. Inside this block, there are two important layers, Multi-Head Self-Attention(MHSA) and Multi-Layer Perceptron(MLP). The MHSA layer is the characteristic constituent of transformer to perform information exchange among patches. In details, each patch is multiplied with three transpose matrices to generate query($\mathbf{Q}$), key($\mathbf{K}$) and value($\mathbf{V}$) that are then be used to get attention score between patches and to output attention vectors as follows [22]:

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = Softmax(\mathbf{Q}\mathbf{K}^T/\sqrt{d})\mathbf{V} \quad (1)$$

The MLP layer contains fully connected layers with activation to transform attention vectors to latent vectors for next encoder or decoder layer.

Since tactile perception arises mainly through active interaction, the pure attention structure in ViViT [21] is hypothesised to be able to pick up the correlations in the changes spatial-temporal signals. Since the exact hierarchy in interactions between spatial and temporal components are unclear in the neuroscience literature, we will assume that a Factorised Encoder model which is similar to existing tactile inference approaches [14], [18], [20] is able to capture the spatial-temporal dynamics while limiting the computational cost well.

Two transformers are stacked to form a spatial-temporal transformer, while each of them closely follows the vision transformer structure. Data are passed to two transformers separately. The spatial transformer only considers attention score between patches comes from the same time frame, while the temporal transformer relates all the frames using the selected feature outputs from spatial transformer. Output of temporal transformer is passed to a MLP that determines the final classification result.

Although two transformers contain more layers and introduce a more complex model, the overall computational power requirement is smaller because fewer number of correlations between pairs of patches are computed. As mentioned in [21], model with one spatial-temporal transformer has a complexity of $O((n_t * n_h * n_w)^2)$, while model with stacked two transformers only has a complexity of $O((n_h * n_w)^2 + (n_t)^2)$, where n_h, n_w, n_t represent the number of patch divisions in height, width, time dimensions, respectively.

B. Transformer Pruning

To further save computational cost and promote efficient training on smaller datasets, pruning techniques can be used as a treatment to the transformer backbone to attain fewer floating point operations(FLOPs). In our model, we insert two types of pruning selection. Channel selection is inserted at the MHSA and MLP layers while frame selection is inserted before the space transformer.

Both ways of layer selection are implemented by adding a mask on the features to prune. The masks are initialized to matrices of one and are trainable parameters in the model. After training, the mask values are viewed as the importance scores for each feature. At the pruning stage, a threshold will be set according to how many percent of the parameters will be pruned. All mask values below this threshold will be set to zero and therefore, the corresponding features will have zero impact on the final classification output.

1) Channel selection: As mentioned earlier, MHSA and MLP layers execute lots of operations. In MHSA layers, matrices **Q K V** will be calculated and saved for every single patch. Then, each set of **Q K V** will be evaluated with other sets of **Q K V** to get the attention score. Therefore we decide to prune out some values in **Q K V** matrices for both spatial and temporal transformers. Also, we select the dimension in fully connection layers to create more sparsity. As there are so many features in those layers, pruning a large portion will not affect the overall accuracy.

2) Frame selection: Each tactile sample has many time frames, depending on the sensor frequency and interaction duration for each dataset. It has been demonstrated in the experiments in [14] that for a model to learn the texture representation on a given tactile signature duration, only a fractional segment of the recorded time signal is needed. This suggests that there exists redundancy in the temporal domain of the signal which means that pruning at frame level at the very beginning of our model will be beneficial to significantly reduce the temporal features entering the model. The exact proportion of pruning may depend on the complexity of the textures present, the sensor frequency and the recorded duration.

IV. EXPERIMENTS

This section presents our datasets, training results and pruning results in details.

A. Tactile Texture Datasets (TTD)

We evaluate our proposed model on three distinct datasets on tactile texture classification, namely TTD-RoBoSkin, TTD-BioTac and TTD-GelSight. These datasets are publicly available. Each dataset was collected with a unique tactile sensor mounted on a robot arm and interacted on common homogeneous material surfaces. Due to different sensor design and mechanisms, they collectively provide a wide range of temporal resolution and spatial distribution of texture representation for us to benchmark the effectiveness of the proposed model. The materials that were used for experiments are shown in Fig3.

1) TTD-RoboSkin Dataset [14]: The forearm patches of the iCub RoboSkin [38] tactile sensor were used to record the capacitive force readings when the iCub robot rotationally sweeps its forearm on different materials. This dataset contains 20 different materials with 50 samples in each class. Each sample records 75 frames and each frame is processed into a 6×10 gray-scale image.

2) TTD-BioTac Dataset [19], [20]: The SynTouch BioTac [39] tactile sensor has 19 electrodes. It was installed as an extended end-effetor on a KUKA LBR iiwa 14 robot arm to record the sensor readings within a time window of 400 frames for a lateral sliding motion with controlled speed and contact force. Two sub-datasets are available containing 20 materials (TTD-BioTac20) and 50 materials (TTD-BioTac50) respectively. Although the 50 materials contains all the 20 materials, the two datasets were not collected at the same time which potential differences in sensor characteristics due to wear and tear and liquid pressure. Please refer to [19] and [20] for a more detailed description of the data collection setup.

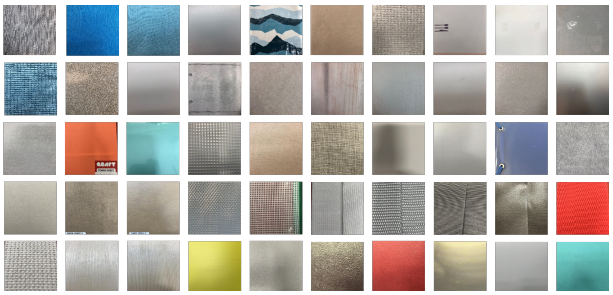

Fig. 2. Snapshots of 50 materials used in TTD-BioTac [20] with a proportion of which were used in TTD-RoboSkin and TTD-GelSight. Materials include: carpet net, cotton, bath towel, leather fake, polypropileno, felt, soft material2, paper2, polypropileno smooth, polypropylene thin, soft material1, cork, eva, paper1, fiber board, wood hard, styrofoam, sponge soft, foam, metal, light purple felt, orange foam sheet, green transparent file folder, white grid, brown wooden paper, green grid cloth, white soft plastic board, white plastic board opposide, blue transparent file folder, purple crepe paper, grey cardboard, wooden board, arclic board, blue gauze, plastic grid bag, white elastic belt, bright white non-elastic belt, bright grey ribbon belt, bright gold ribbon belt, red strap belt, griege strap belt, wooden board slice, wooden board slice opposide, bright yellow board, white cform plain, gold-red paper, bright gold foil, white rice paper, green plasticish paper

3) TTD-GelSight: In addition to the above-mentioned publicly available datasets, we also collected and released a new dataset using the Gelsight sensor [11]. However, because the gel pad is made of elastomer and vulnerable to abrasion caused by lateral friction on the surface, this dataset is instead collected by having the sensor rolling across the material surfaces using a spline trajectory instead of long-distance sliding motion. Specifically, the GelSight sensor is mounted on a KUKA iiwa robot arm connected by a 3D printed end-effector and is controlled to slowly contact the materials from the top until reaching a predefined force threshold and then to rotate clockwise by 1 degree, anticlockwise 2 degrees, and finally clockwise 1 degree back to the center position. Each time frame size is 480×640 and each sample has a total 300 to 400 time frames. Finally we construct a dataset of 45 materials with 50 samples for each material.

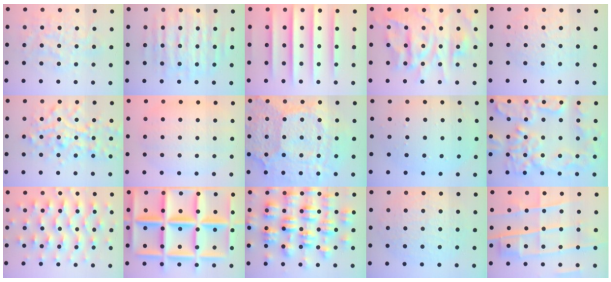

Fig. 3. 15 kinds of materials outputs from GelSight

B. Results without Model Pruning

1) Benchmark against the Baseline: We first validatedthe training result for TTD-RobotSkin and TTD-BioTac20

datasets to benchmark on the performance of Factorised Encoder based ViViT with recurrent autoencoder (RAEC) [19] used as the baseline model. When choosing the model parameters, we set the model to have similar parameter size with RAEC which is about 0.1 million trainable parameters. The model is trained for 300 epochs for each k-fold cross validation set. The results are shown in the first three rows of Table I.

TABLE I
COMPARISON OF TEST ACCURACY FOR 4-FOLD CROSS VALIDATION

Average Test Accuracy		
Model \ Dataset	TTD-RoboSkin	TTD-BioTac20
RAEC(independent)	0.9114	0.9531
RAEC(joint)	0.9323	0.9601
PVT^3 without pruning	**0.9224**	**0.9659**
PVT^3 with RANDOM pruning	**0.9425**	**0.9798**

As compared to RAEC with independent training, the transformer outperforms for both datasets, improving the classification accuracy by about 1.2%. As compared to RAEC with joint learning, the transformer performs better for TTD-BioTac20 dataset but slightly lower for RoboSkin dataset. This is in line with the hypothesis for the joint training to provide information to relatively noisier dataset to improve model performance [19]. The results demonstrate that the transformer backbone is able to better extract spatial-temporal correlations in the tactile texture signals as compared to the baseline.

2) Performance on Two TTD-BioTac Datasets: Considering potential sensor deterioration and setup discrepancy, the two datasets are supposed to contain similar representation but with different scaling and bias. By benchmarking the performance of mixing the two sub-datasets together, we can study the capturability of the important spatial-temporal information present in both datasets by the transformer backbone. We train the model on each dataset both individually and together. The results are shown in Table II. It can be seen that the proposed model maintains a reasonably robust performance despite the distribution shift in data and the presence of imbalancedly higher number of training examples in 20 of the material classes.

3) GelSight Dataset: GelSight is an image-based tactile sensor and provides very high resolution data for local contact region. It is highly sensitive to spatial resolution. While the captured data from a Gelsight sensor on some materials can be differentiated by naked human eyes, others may still share similar features which are almost visually indistinguishable. However, the high spatial resolution can also be problematic for training a representation model with a long architecture pipeline such as the REAC due to the weak supervisory signal at the end of the pipeline. This is especially so when the dataset is insufficiently large. To test a scaled-down version of the Gelsight captured spatial information, we extracted a 240 $\times$ 320 pixels ROI for each frame and the mid-25 frames from the dataset for model training. Background subtraction from the raw GelSight data was performed to normalise the dataset prior to training. Due to computational resource constraints, we trained on the first 15 classes Fig. 3 of the 45 materials with both the ViViT backbone and the RAEC model. The testing results are 89.3% and 16.0% accuracy for ViViT backbone and RAEC model respectively. This suggests that even for the reduced size of the spatial dimension from GelSight, the long RAEC pipeline with two independent models in RAEC to model the temporal and spatial components is unable to make use of the supervising signal from the label while ViViT which integrates the attention for both domains can still effectively pick up the correlations from the labels.

TABLE II
PERFORMANCE OF VIVIT BACKBONE ON BOTH TTD-BIOTAC DATASETS

BioTac 50 and 20 Classes	
Dataset	Average Test Accuracy
TTD-BioTac20	0.966
TTD-BioTac50	0.947
Combined	0.903

C. PVT^3 Performance

To tackle noisy data and to extract representative feature for efficient training, we conducted a preliminary study to prune out part of trainable parameters directly by random unstructured pruning and then retrain the model. As this random pruning will set some weights to zero, it will have a similar effect as dropout. By re-initializing optimized function, the accuracy increased for both TTD-RoboSkin and TTD-BioTac datasets as shown in the last row of Table I. The simple random unstructured pruning enlarges the model sparsity. Similar to dropout, but in a much larger space, it leads to better generalizability. Meanwhile, we re-initialze the optimization function for very 100 epochs. This two approaches help to fine tune the model. As such, same approach by re-initialzing optimization function when training the PVT^3 with pruning selection version was used subsequently.

As briefly mentioned above, the proposed model can maintain good performance after certain level of random pruning. This implies possible redundancy in the original model architecture and suggests potential improvement and acceleration by model pruning. In this section, we explore different model size and present the results of iterative pruning.

1) Channel Selection: We use the TTD-RoboSkin dataset as an example to demonstrate the parameter size of channel selection for the model pruning. Each input sample has a shape of $H \times W \times T = 6 \times 10 \times 75$, where 75 represents the length of time sequences. The hyper parameter of the pruned version of PVT^3 creates a total number of 64272 trainable parameters. Input are split into 4 patches and embedded to 1-D tensor of length 20. Concatenating the four tensors and adding on the space classification token will create a [5$\times$20] shape tensor to feed in spatial transformer. The

TABLE III
CHANNEL SELECTION RESULTS

Channel Selection Pruning		
Dataset	Test Accuracy before pruning	Test accuracy after pruning
RoboSkin fold-1	0.925	0.94
RoboSkin fold-2	0.935	0.94
RoboSkin fold-3	0.945	0.945
RoboSkin fold-4	0.935	0.95
BioTac20 fold-1	0.975	0.965
BioTac20 fold-2	0.97	0.98
BioTac20 fold-3	0.965	0.965
BioTac20 fold-4	0.98	0.975

Channel Pruning layer in spatial transformer will prune out specific percent of $\mathbf{Q}$ $\mathbf{K}$ $\mathbf{V}$ matrices with each matrix size being [5×20×64] where 64 is set as a hyper-parameter. Our mask will mask out some values in the last dimension and create sparsity in $\mathbf{Q}$ $\mathbf{K}$ $\mathbf{V}$ matrices. While the input shape to temporal transformer is [76×20], where 76 comes from the 75 time frames and 1 temporal classification token, the $\mathbf{Q}$ $\mathbf{K}$ $\mathbf{V}$ matrices size will change to [76×20×64]. Since time frames is larger than 64, we put the mask on the first dimension. Other than pruning $\mathbf{Q}$ $\mathbf{K}$ $\mathbf{V}$ matrices, pruning is also applied to fully connection layer inside two transformer blocks. Those masks will prune out some values in the 20 dimensions.

The results are shown in TableIII. It can be observed that the performance gain for the TTD-RoboSkin dataset is more significant than that of the TTD-BioTac dataset. This is in line with the higher level of noisy present in the TTD-RobotSkin dataset. By leveraging on pruning, some noise is suppressed.

2) Frame Selection: The full datasets contain 400 time frames for TTD-BioTac and 75 time frames for TTD-RoboSkin, taking eight and three seconds in real time. Although they have been proved sufficient to distinguish different materials, it is hard for implementing real-time inference. Inspired by [14] which suggests that a selection of frames can achieve rather good performance within half of the time, we add a frame selection layer after the patch embedding layer before the inputs are taken by transformers. The results for frame pruning is shown in Table IV.

It is can be seen that pruning has a minor effect on the TTD-BioTac dataset. The accuracy only start to decrease when over 80 percent of data is not selected. However, TTD-RoboSkin dataset shows a sharp drop before reaching the 50 percent as in [14]. Three possible reasons can account for this decrease in accuracy. Firstly, the overall information in one frame is not sufficient for classification tasks. In [14], sensor outputs from both dynamic touch and sliding motions are used to build the classification model and multi-modal data can represent a richer set of material properties, while here we only have access to the single modality of sliding. Secondly, the most important features for classification may be embedded in the time series by amount of change between consecutive frames and thus pruning too many data may disrupt the correlation between frames. Last but not least,

TABLE IV
FRAME PRUNING WITH DIFFERENT RATIO

TTD-BioTac and TTD-RoboSkin Frame Pruning		
Prune Rate	TTD-BioTac	TTD-RoboSkin
30 percent	0.97	0.915
40 percent	0.96	0.895
50 percent	0.945	0.65
60 percent	0.94	0.545
70 percent	0.93	0.63
80 percent	0.895	\

the TTD-BioTac dataset can bear a larger ratio of pruning partially because it originally has a larger temporal resolution, 100 Hz compared to about 50 Hz for TTD-RoboSkin dataset. The speed of motion of the sensor is also much slower for TTD-BioTac than its counterpart, 2.5 cm/s VS about $5^{\circ}/s$. In summary, we claim that our model is robust to certain level of pruning that may depend on the modality, temporal resolution and consistency between frames of the original dataset.

V. CONCLUSION

In this paper, we presented the PVT^3 model based on ViViT backbone with channel pruning techniques to represent and classify textures from tactile sensory readouts. We demonstrated that transformer architecture can capture the rich information hidden in spatial-temporal correlations. As compared to the baseline, PVT^3 yielded better results. In addition, the proposed model adapted well to datasets of different sizes without additional processing. Such end-to-end model is easy to transfer to different datasets and therefore can be applicable to other tactile perception tasks. The pruning layers added to the model gave an easy access to the minimal number of parameters needed and can be used as a benchmark for further model compression.

One potential future work is the patch size selection for different input size. For inputs with irregular shape or when information is not concentrated, e.g. BioTac data consisting 19 electrodes readings for each frame, patch size selection can be crucial for accurate results. In our experiment, we only explore feeding the original input to the transformer, but other feature selection methods can be tested for better performance.

Another direction is to explore transformer with smaller computational power, which may require optimization on specific software or hardware. Corresponding pruning algorithms can be developed to take full advantage of model architecture and memory storage. There is also a possible way to transfer transformer structure to spiking neural network (SNN). We noticed some works have been down in this field [40], but it seems that the proposed spiking transformer network preserves too much original transformer structure and the advantages of SNN are not fully utilized. Further improvements is needed to build an energy efficient spiking transformer model with high classification accuracy.

REFERENCES

[1] 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)*. IEEE, 2018, pp. 2722–2727.

[2] T. Taunyazov, L. S. Song, E. Lim, H. H. See, D. Lee, B. C. Tee, and H. Soh, "Extended tactile perception: Vibration sensing through tools and grasped objects," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 1755–1762.

[3] 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.

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

[5] 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)*. IEEE, 2015, pp. 1155–1160.

[6] 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.

[7] G. Li, S. Liu, L. Wang, and R. Zhu, "Skin-inspired quadruple tactile sensors integrated on a robot hand enable object recognition," *Science Robotics*, vol. 5, no. 49, p. eabc8134, 2020. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abc8134

[8] 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.

[9] 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.

[10] N. Wettels, V. J. Santos, R. S. Johansson, and G. E. Loeb, "Biomimetic tactile sensor array," *Advanced Robotics*, vol. 22, no. 8, pp. 829–849, 2008. [Online]. Available: https://doi.org/10.1163/156855308X314533

[11] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, 2017. [Online]. Available: https://www.mdpi.com/1424-8220/17/12/2762

[12] N. F. Lepora, "Soft biomimetic optical tactile sensing with the tactip: A review," *IEEE Sensors Journal*, vol. 21, no. 19, pp. 21 131–21 143, 2021.

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

[14] 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)*. IEEE, 2019, pp. 4269–4275.

[15] 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.

[16] H. Soh, Y. Su, and Y. Demiris, "Online spatio-temporal gaussian process experts with application to tactile classification," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 4489–4496.

[17] Y. Gao, L. A. Hendricks, K. J. Kuchenbecker, and T. Darrell, "Deep learning for tactile understanding from visual and haptic data," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 536–543.

[18] G. Cao, Y. Zhou, D. Bollegala, and S. Luo, "Spatio-temporal attention model for tactile texture recognition," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 9896–9902.

[19] 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)*, 2020, pp. 10 907–10 913.

[20] R. Gao, T. Tian, Z. Lin, and Y. Wu, "On explainability and sensor-adaptability of a robot tactile texture representation using a two-stage recurrent networks," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 1296–1303.

[21] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," 2021.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[23] L. Dong, S. Xu, and B. Xu, "Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5884–5888.

[24] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[25] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "Cvt: Introducing convolutions to vision transformers," 2021.

[26] S. d'Ascoli, H. Touvron, M. Leavitt, A. Morcos, G. Biroli, and L. Sagun, "Convit: Improving vision transformers with soft convolutional inductive biases," 2021.

[27] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush, "OpenNMT: Open-source toolkit for neural machine translation," in *Proc. ACL*, 2017. [Online]. Available: https://doi.org/10.18653/v1/P17-4012

[28] A. Bhandare, V. Sripathi, D. Karkada, V. Menon, S. Choi, K. Datta, and V. Saletore, "Efficient 8-bit quantization of transformer neural machine language translation model," 2019.

[29] D. Jia, K. Han, Y. Wang, Y. Tang, J. Guo, C. Zhang, and D. Tao, "Efficient vision transformers via fine-grained manifold distillation," 2021.

[30] M. Zhu, Y. Tang, and K. Han, "Vision transformer pruning," 2021.

[31] Y. Tang, K. Han, Y. Wang, C. Xu, J. Guo, C. Xu, and D. Tao, "Patch slimming for efficient vision transformers," 2021.

[32] P. Michel, O. Levy, and G. Neubig, "Are sixteen heads really better than one?" 2019.

[33] F. Lagunas, E. Charlaix, V. Sanh, and A. M. Rush, "Block pruning for faster transformers," 2021.

[34] Y. L. Cun, J. S. Denker, and S. A. Solla, *Optimal Brain Damage*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990, p. 598–605.

[35] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning efficient convolutional networks through network slimming," 2017.

[36] S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M. A. Horowitz, and W. J. Dally, "Eie: Efficient inference engine on compressed deep neural network," 2016.

[37] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Weinberger, "Deep networks with stochastic depth," 2016.

[38] 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*. IEEE, 2010, pp. 2212–2217.

[39] 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)*. IEEE, 2012, pp. 1122–1127.

[40] E. Mueller, V. Studenyak, D. Auge, and A. Knoll, "Spiking transformer networks: A rate coded approach for processing sequential data," in *2021 7th International Conference on Systems and Informatics (ICSAI)*, 2021, pp. 1–5.