Self-Supervised Learning for Tactile Classification

Jonas Fischer
Technical University of Munich
Munich, Germany
jonasetienne.fischer@tum.de

Zakariea Sharfeddine
Technical University of Munich
Munich, Germany
zakariea.sharfeddine@tum.de

***Abstract*—Tactile sensing is essential for robots to classify materials in real-world environments. While models like TacNet-II achieve high accuracy with supervised learning, they rely on costly labeled data. We explore self-supervised learning methods (Autoencoder, Masked Autoencoder and CNN-based JEPA) using a shared TacNet-II encoder. Our approach introduces time-based masking adapted to tactile signals and evaluates the impact of self-supervised learning on downstream classification. Results show improved accuracy and sample efficiency.**

I. Introduction

Efficient tactile sensing is a critical capability for intelligent robots, especially when it comes to tasks that require precise interaction with different materials. Accurate material classification enables robots to reliably grasp, manipulate, and interact with objects, which is essential for effective assistance in household environments.

Recent advances, such as TacNet-II by Tulbure and Bäuml [1], have demonstrated impressive results in classifying 36 different household materials using supervised learning, achieving accuracies up to 95%. However, supervised approaches require extensive labeled datasets, which are notoriously difficult, expensive, and time-consuming to obtain in robotics.

To accelerate the development and deployment of robotic learning systems, it is crucial to reduce the dependency on labeled data. Self-supervised learning (SSL) methods have emerged as a promising alternative, capable of learning robust representations from unlabeled data. In particular, Joint-Embedding Predictive Architecture (JEPA), introduced by Assran et al. [2], has shown superior performance by learning abstract, semantically meaningful representations.

In this work, we investigate the effectiveness of JEPA adapted specifically to CNN-based architectures (CNN-JEPA [3]) and time-series data for tactile sensing tasks. We explore how TacNet-II can benefit from pretraining with self-supervised methods with the goal to improve its sample efficiency and reduce labeling requirements. Specifically, we compare CNN-JEPA with established SSL techniques, including Masked Autoencoders (MAE [4]) and classical autoencoders (AE [5]), using different splits of pretraining and downstream tasks. By systematically analyzing these splits, we evaluate the impact of self-supervised pretraining on downstream tactile classification performance.

II. Related Work

Tactile sensing is essential for enabling autonomous robots to interact with physical environments, particularly for tasks like material classification. Early approaches, such as those by Fishel and Loeb [6], relied on multi-modal sensors like BioTac and handcrafted features, achieving high accuracy in controlled environments but poor generalization in real-world settings [7]. Deep learning methods, including HapticNet [8], later improved performance by learning features directly from data. These approaches also explored various tactile modalities such as acceleration [9], vibration [10], and thermal signals [11]—but were often constrained to rigid sensors or lab setups, limiting their real-world applicability.

To address real-world applicability of tactile material classification, Tulbure and Bäuml introduced TacNet-II [1], an enhanced convolutional neural network (CNN) architecture specifically designed to handle tactile data from a flexible tactile skin. This commercially available sensor features a 4×4 taxel array and can easily be attached to curved robotic surfaces. TacNet-II significantly improved upon its predecessor by incorporating recent CNN architectural enhancements, such as batch normalization and adversarial training, achieving 86% classification accuracy with one sweep motion on a challenging dataset of 36 household materials.

To mitigate the reliance on labeled data, SSL methods have emerged as promising alternatives. Classical reconstruction-based methods, such as Masked Autoencoders (MAE [4]), reconstruct missing input patches in pixel space. TI-MAE [12] extends MAE to time-series data by incorporating temporal positional encodings and transformer-based decoders.

JEPA, introduced by Assran et al. [2], is a promising self-supervised architecture that predicts abstract representations of masked regions within input images. Unlike reconstruction-based approaches, JEPA operates in a learned abstract representation space rather than pixel space, leading the model to learn generalized and semantically meaningful representations. Compared to MAE, JEPA requires higher computation per iteration but achieves significantly faster convergence, highlighting its computational efficiency and robust performance [3].

While JEPA was initially developed for transformer architectures, adapting it to CNN-based architectures poses challenges, such as handling masked inputs and ensuring a sufficient receptive field. Kalapos and Gyires-Tóth introduced

CNN-JEPA [3], which addresses these challenges and even outperforms I-JEPA with ViT-Small and ViT-Base by a large margin on ImageNet-100. Recent results show that CNN-JEPA is competitive with state-of-the-art CNN-based self-supervised methods such as BYOL, SimCLR, and VICReg while requiring fewer computational resources [3].

Our work builds upon these developments by exploring CNN-based JEPA for self-supervised learning on tactile data, aiming to address the limitations of supervised tactile classification methods.

III. Methods

A. Autoencoder

An Autoencoder (AE) [5] is a neural network architecture composed of two parts: an Encoder E and a Decoder D. The Encoder compresses the input x into a latent representation $E(x) = z$ that tries to captures the most salient features of the data. The Decoder then attempts to reconstruct the original input from this latent variable. During pretraining, both the Encoder and Decoder are optimized by minimizing the L2 reconstruction loss between the input and its reconstruction:

$$\mathcal{L}_{\text{AE}} = \frac{1}{N} \sum_{i=1}^{N} \|x_i - D(z_i)\|_2^2 , \tag{1}$$

where N is the number of samples in the dataset, x_i denotes the i-th input, and $z_i = E(x_i)$ is its corresponding latent representation.

B. Masked-Autoencoder

He et al. [4] proposed a variant of the Autoencoder called the Masked Autoencoder (MAE). The architecture is similar to the standard Autoencoder, with the key difference that a binary mask m_i is applied to the input x_i, such that only a subset of the input taxel are visible to the Encoder. The Encoder processes the masked input, $z_i = E(x_i \cdot m_i)$, and in the latent space, a learnable vector p is inserted at positions corresponding to the masked input dimensions. The Decoder reconstructs the full input based on this partially masked representation. The model is trained to minimize the L2 reconstruction error over the masked input regions:

$$\mathcal{L}_{\text{MAE}} = \frac{1}{N} \sum_{i=1}^{N} \|(x_i - D(z_{i,masked})) \cdot \tilde{m}_i\|_2^2 , \tag{2}$$

where $z_{i,\text{masked}} = z_i \cdot m_i + \tilde{m}_i \cdot p$,

where m_i is the binary mask and $\tilde{m}_i$ the flipped mask for the i-th input,. By focusing reconstruction loss only on masked regions, MAE encourages the Encoder to learn generalizable and high-capacity representations [4].

C. Joint-Embedding Predictive Architecture

Assran et al. [2] introduced JEPA, a self-supervised learning framework that learns semantic representations without relying on handcrafted data augmentations or explicit pixel reconstruction. Unlike MAE, which reconstruct masked regions directly in pixel space, JEPA operates entirely in latent representation space, predicting abstract representations of masked patches from visible context regions. This abstraction encourages the encoder to produce semantically rich, high-level features rather than low-level pixel details.

The architecture consists of two encoders: a context encoder E_C, which processes the input sequence masked with a context mask, and a target encoder E_T, which processes the full (unmasked) input. A set of target patches is then sampled, and a predictor network P, conditioned on positional tokens p, is trained to predict the latent representations of these target patches given the output of the context encoder. The JEPA model minimizes the L2 loss between the predicted and actual target representations and updates only the context encoder and predictor via stochastic gradient descent (SGD). The loss is computed exclusively over the selected target patches.

$$\mathcal{L}_{\text{JEPA}} = \frac{1}{N} \frac{1}{K} \sum_{i=1}^{N} \sum_{k=1}^{K} \|\tilde{z}_{i,k} - P(z_i, p_k)\|_2^2 , \tag{3}$$

where $\tilde{z}_{i,k}$ is the k-th target patch, z_i is the latent representation of the masked context input x_i, and p_k is the positional information of the k-th target patch, and K number of target patches. The target encoder's parameters are updated via an exponential moving average (EMA) of the context encoder's weights to stabilize training and avoid representation collapse [2].

D. Time-based Masking

An important design choice in both MAE and JEPA is the masking strategy. While Li et al. [13] propose time-based masking for learning representations from time series, and Assran et al. [2] introduce multi-block masking for improved performance, we combine both approaches by applying a time-based multi-block masking strategy.

MAE masking: For the MAE input, we randomly sample N binary time blocks to create a mask m, which we apply to the input x via element-wise multiplication. We mask between 75–80% of the input sequence. The same mask is also applied to the latent representation, where masked tokens are replaced with a learnable parameter p.

JEPA masking: The context mask is sampled in the same way as in MAE, using randomly selected binary time blocks. Additionally, we independently sample L binary time blocks to construct the target mask, masking 15–20% of the encoded input sequence. Following Assran et al. [2], we prevent trivial predictions by masking all taxels in the context input that are not masked in the target mask.

IV. Experiments

A. Data

For conducting our experiments, we primarily used the TactMat dataset introduced by Tulbure and Bäuml [14], the same dataset employed for evaluating the TacNet-II model [1].

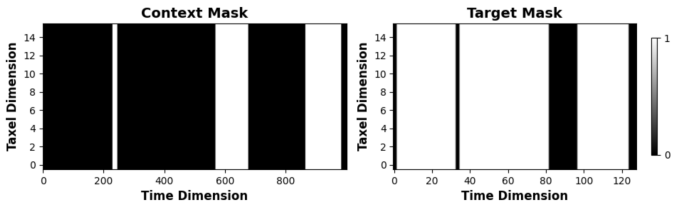

Fig. 1. Illustration of time-based multi-block masking for JEPA, using two context blocks (70-75% masked) and three target blocks (15-20% masked)

The TactMat dataset consists of tactile recordings from 36 common household materials. Each material has 100 individual samples, and each sample contains spatio-temporal tactile signals captured by a flexible 4×4 taxel array sensor. Specifically, the dataset dimensions are as follows:

- **Materials:** 36 different household materials
- **Samples per material:** 100 samples
- **Timesteps per sample:** 1000 timesteps
- **Sensor resolution:** 4×4 taxel array

Additionally, for our final experiment aimed at evaluating the impact of self-supervised pretraining using a larger dataset, we utilized another unlabeled tactile dataset consisting of ∼6200 samples. Each sample in this additional dataset also contains 1000 timesteps recorded with the same 4×4 taxel array sensor. However, this dataset differs from TactMat in terms of experimental setup, material selection, and tactile exploration motions. Due to the absence of labels, we exclusively used this dataset for the pretraining phase.

B. Model Architecture

We reimplemented TacNet-II [1] with minor adjustments. The model processes spatio-temporal tactile signals of shape $1000 \times 4 \times 4$ and follows a CNN-based architecture consisting of three convolutional layers with batch normalization and ReLU activation, each followed by max pooling to progressively downsample the time dimension. The final representation is flattened and passed through three fully connected layers with dropout for classification. The complete architecture is illustrated in Figure 2.

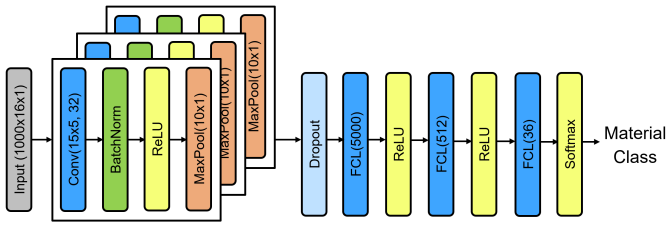

Fig. 2. Our implementation of the TacNet-II architecture [1]. The network processes a 1000×16 spatio-temporal tactile signal with a sequence of convolutional, pooling, and fully connected layers for final classification.

We reuse the TacNet encoder as a shared backbone across all self-supervised learning methods: AE, MAE, and JEPA. This ensures a fair comparison and isolates the impact of the self-supervised objective.

C. CNN-JEPA Adaptation

To adapt JEPA to tactile data and CNNs, we follow the approach of Kalapos et al. [3]. Our CNN-JEPA model uses two copies of the TacNet encoder: a context encoder and a target encoder. The context encoder has its first layers converted to sparse operations, computing only unmasked spatial positions (i.e., where the binary mask m is 1), while the target encoder processes the full (unmasked) input and is updated solely via a slow exponential moving average (EMA) of the context encoder's weights, with the EMA momentum gradually increasing from 0.996 to 1.0 during training. This slow update mitigates representational loss by keeping the target encoder stable. A predictor, composed of depthwise separable convolutions, maps the context features to a prediction of the target features. The architecture is illustrated in Figure 3.

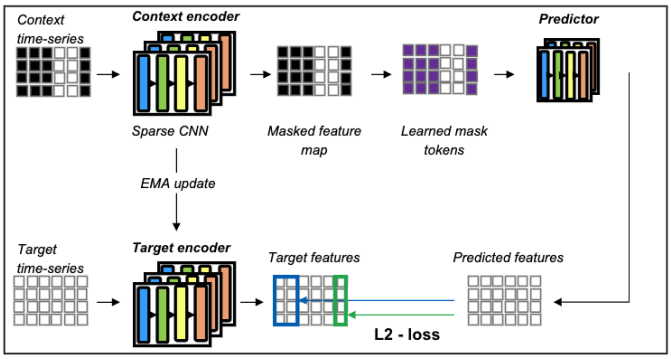

Fig. 3. CNN-JEPA architecture adapted from Assran et al. [2] and Kalapos et al. [3]. The context encoder processes masked time-series inputs. A predictor estimates masked latent features, while the target encoder (updated via EMA) provides the target features. The model is trained to minimize the L2 distance in latent space.

Downsampling trick: The TacNet encoder compresses the input from shape $B \times 1 \times 1000 \times 16$ to $B \times 128 \times 1 \times 16$, effectively collapsing the time dimension. To transfer the mask m to the latent representation z, we need to downsample it. We hypothesize that the channel dimension in the latent space encodes temporal information. Therefore, to apply the mask in latent space, we first downsample m along the time dimension and then reshape the time and channel dimensions, before we apply the target mask to the encoded input.

D. Experiment Setup

For evaluation, we use a fixed test set of 10 samples per class from the TactMat dataset. The remaining data is split into pretraining and downstream using three different ratios: 50/50, 80/20, and 95/5. Each downstream set is further split into 85% training and 15% validation.

To evaluate sample efficiency (in this case the ability to extract useful representations even from imperfect, less informative data), we also conducted a second experiment: pretraining the models on an unlabeled dataset of ∼6200 samples (from the same sensors but a different setup), followed by downstream training on the full labeled TactMat training set.

TABLE I
TRAINING DETAILS

Parameters	*No-Pretraining*	*JEPA*	*MAE*	*AE*
Pretraining Epochs	0	150		
Finetuning Epochs	400			
Weight decay	1e-3			
Dropout	0.3	0.5	0.5	0.3
Encoder LR	1e-4	1e-4	5e-5	1e-4
Classifier LR	1e-4			
Masking Ratio	-	60%	70%	-

E. Training Details

All models were fine-tuned for 400 epochs with a weight decay of 1×10^{-3}. For JEPA and MAE, we performed 150 epochs of pre-training using time-based multi-block masking.

During fine-tuning, we observed that pre-training with JEPA and MAE enabled the use of higher dropout rates (0.5) without causing underfitting. In contrast, models without pre-training or pretrained with autoencoders exhibited underfitting when regularization increased. This pretraining contributed to more stable model training. Furthermore, MAE benefited from a lower encoder learning rate compared to the classifier learning rate, while other setups used the same rate for both components. Table I summarizes the key hyperparameters and training configurations used across different pre-training strategies.

Data Augmentation: As part of our investigation into sample efficiency, we experimented with augmentations such as adding Gaussian noise, flipping sensors in the spatial dimension and masking the initial timesteps. These had no noteworthy effect on classification accuracy.

V. RESULTS

When splitting the training data into 50% for self-supervised pretraining and 50% for fine-tuning, JEPA outperforms the non-pretrained baseline by 4.7 percentage points and MAE by 8.3 percentage points. With an 80%/20% split, JEPA again outperforms the baseline by 5.2 percentage points, and MAE by 7.2 percentage points. In a few-shot setting, where only four samples remain for fine-tuning, MAE achieves a marginal improvement of 1.1 percentage points over the non-pretrained model, likely within the range of statistical noise, whereas JEPA yields a more substantial gain of 4.8 percentage points. In all three scenarios, the autoencoder-pretrained model performs worse than the non-pretrained baseline. This is likely due to the limited dataset size, which causes the AE to memorize input samples rather than learn meaningful, transferable representations. In contrast, MAE and JEPA successfully learn generalizable and informative features, even when pretrained on small datasets, as demonstrated in these experiments.

When extending pretraining to an additional dataset, all self-supervised approaches improve over the baseline. AE and JEPA each achieve a 2.9 percentage point performance gain, while MAE improves performance by 3.8 percentage points.

Overall, both MAE and JEPA consistently improve performance across all experiments, with MAE yielding the highest accuracy in most settings. A summary of results across all splits is presented in Table II.

TABLE II
TACTILE CLASSIFICATION ACCURACY (%) UNDER DIFFERENT FINETUNING SPLITS

Model	**Unlabeled dataset / 100**	**50/50**	**80/20**	**95/5**
No Pretraining	78.3	65.3	42.8	22.2
JEPA	81.1	70.0	48.0	**27.0**
MAE	**82.2**	**73.6**	**50.0**	23.3
AE	81.1	63.9	35.0	20.0

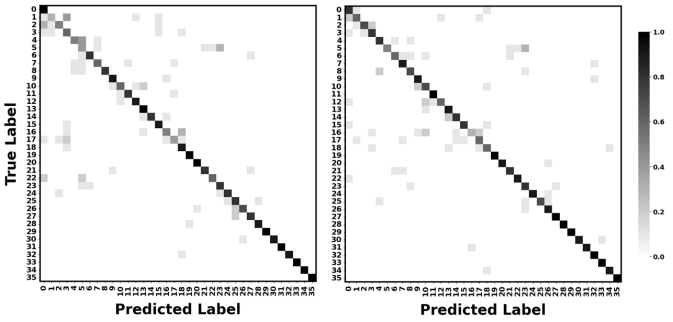

Fig. 4. Normalized confusion matrices for the 36-class material classification task. Left: baseline model trained from scratch. Right: model pretrained with CNN-JEPA and fine-tuned on TactMat. Pretraining improves prediction accuracy, particularly for previously high-error classes (top left corner in both matrices)

VI. CONCLUSION

In this work, we demonstrated the effectiveness of self-supervised learning for tactile classification tasks. By adapting JEPA and MAE to the existing CNN-based TactNet architecture and applying a multi-block masking strategy tailored for time-series data, both methods successfully learn meaningful representations for tactile time series. This leads to improved TactNet performance in tactile classification without requiring additional labeled data.

VII. FUTURE RESEARCH

Due to the TactNet encoder architecture, we downsampled and reshaped the mask, hypothesizing that the channel dimension captures temporal information. Future work should explore models that preserve the temporal dimension to assess whether this improves the effectiveness of JEPA and MAE for tactile time-series classification. Preserving temporal structure could enable the encoder to better learn sequential patterns during self-supervised learning.

Additionally, we apply pre-training to a relatively small dataset. JEPA has been shown to benefit from training on larger and more diverse datasets [2]. With access to more data, a larger model could be employed, which may further enhance JEPA's performance [2]. Future research

should investigate whether JEPA can outperform MAE in a self-supervised setting when trained on a significantly larger tactile time-series dataset with a higher-capacity encoder.

ACKNOWLEDGMENT

This project is part of the course "Advanced Deep Learning for Robotics" at TUM. We thank the Chair of Learning AI for Dextrous Robots (Prof. Bäuml) and our supervisor Felix Kroll for their guidance and support.

REFERENCES

[1] A. Tulbure and B. Bäuml, "Superhuman performance in tactile material classification and differentiation," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*. IEEE, 2018.

[2] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas, "Self-supervised learning from images with a joint-embedding predictive architecture," 2023. [Online]. Available: https://arxiv.org/abs/2301.08243

[3] A. Kalapos and B. Gyires-Tóth, "Cnn-jepa: Self-supervised pretraining convolutional neural networks using joint embedding predictive architecture," in *2024 International Conference on Machine Learning and Applications (ICMLA)*. IEEE, Dec. 2024, p. 1111–1114. [Online]. Available: http://dx.doi.org/10.1109/ICMLA61862.2024.00169

[4] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. B. Girshick, "Masked autoencoders are scalable vision learners," *CoRR*, vol. abs/2111.06377, 2021. [Online]. Available: https://arxiv.org/abs/2111.06377

[5] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul 2006.

[6] J. Fishel and G. Loeb, "Bayesian exploration for intelligent identification of textures," *Frontiers in neurorobotics*, vol. 6, p. 4, 06 2012.

[7] D. Xu, G. E. Loeb, and J. A. Fishel, "Tactile identification of objects using bayesian exploration," in *2013 IEEE international conference on robotics and automation*. IEEE, 2013, pp. 3056–3061.

[8] Y. Gao, L. A. Hendricks, K. J. Kuchenbecker, and T. Darrell, "Deep learning for tactile understanding from visual and haptic data," 2016. [Online]. Available: https://arxiv.org/abs/1511.06065

[9] H. Zheng, L. Fang, M. Ji, M. Strese, Y. Özer, and E. Steinbach, "Deep learning for surface material classification using haptic and visual information," *IEEE Transactions on Multimedia*, vol. 18, 12 2015.

[10] A. Gómez Eguíluz, I. Rano, S. Coleman, and T. Mcginnity, "Reliable robotic handovers through tactile sensing," *Autonomous Robots*, vol. 43, 10 2019.

[11] W. Böttcher, P. Machado, N. Lama, and T. Mcginnity, "Object recognition for robotics from tactile time series data utilising different neural network architectures," 07 2021.

[12] Z. Li, Z. Rao, L. Pan, P. Wang, and Z. Xu, "Ti-mae: Self-supervised masked time series autoencoders," 2023. [Online]. Available: https://arxiv.org/abs/2301.08871

[13] ——, "Ti-mae: Self-supervised masked time series autoencoders," 2023. [Online]. Available: https://arxiv.org/abs/2301.08871

[14] A. Tulbure and B. Bäuml, "Tactmat dataset: Dlr's robotic tactile material classification dataset," 2018, accessed 5 June 2025. [Online]. Available: https://dlr-alr.github.io/dlr-tactmat/