2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

A Neuromorphic System for the Real-time Classification of Natural Textures

George Brayshaw[1], Benjamin Ward-Cherrier[2] and Martin J. Pearson[3]

***Abstract*— Tactile exploration of surfaces is a key component of everyday life, allowing us to make complex inferences about our environments even when vision is occluded. The emergence of biomimetic neuromorphic hardware in recent years has furthered our ability to create biologically plausible sensing solutions. While these platforms continue to improve in regards to latency and power consumption, within recent literature on tactile texture classification there is an emphasis on accuracy at the expense of real-time processing. In order for these tactile sensing systems to find use outside of experimental laboratory environments, it is key to design systems capable of capturing and processing data in real-time. Within this paper we present a system for the real-time classification of texture using a neuromorphic tactile sensor, a spiking neural network and a novel decision making algorithm. Our real-time system achieves classification accuracies of 94% on a dataset of 11 natural textile textures. Furthermore our system is capable of identifying textures at human-level performance in as little as 84ms. Additionally, benchmarking our system across CPU, GPU and Loihi2 hardware platforms resulted in a 96% reduction in power consumption on the neuromorphic platform. This system out-performed previous work by the authors and the state of art, both in terms of accuracy and classification speed.**

I. INTRODUCTION

Tactile texture discrimination is critical for enabling robots and prosthetic hands to dexterously manipulate objects and identify surfaces. However, performing fine-grained texture classification in real-time poses challenges due to the high spatial and temporal resolution required, as well as the need for efficient processing of spatio-temporal tactile data. Bio-inspired neuromorphic approaches offer a promising pathway to address these challenges by mimicking the neural coding and computation underlying rapid texture discrimination in the human somatosensory system.

In the human tactile system, textural information is encoded into streams of electro-chemical impulses, or spikes, along afferent nerve fibers [1]. Neuromorphic engineering seeks to model this spike-based neural representation and processing for efficient hardware implementations. Recent studies have developed neuromorphic texture discrimination systems using dynamic tactile sensors and spiking neural networks (SNNs), inspired by the speed and efficiency of biological touch sensing [2].

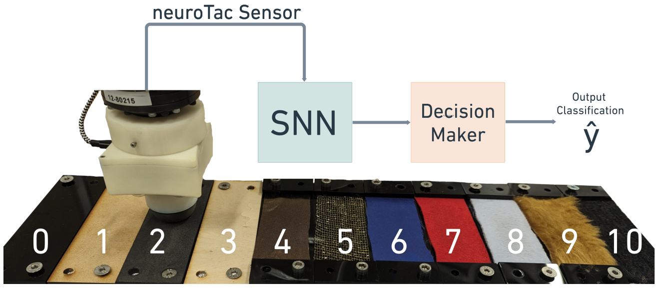


Fig. 1: Overview of our real-time classification system. Tactile data from the neuroTac sensor is passed through a pooling layer before being processed by a pre-trained 3-layer SNN. The output of the network is monitored over time by our tunable decision making algorithm which outputs a classification at each time step based on prior spike trains. Each block, including each individual network layer, processes in parallel. Textures comprising our texture dataset are shown with associated labels. From 0-10; Acrylic, MDF, Foam, Plywood, Microdot, Mesh, Felt, Satin, Fleece, Fake Fur, Wool

Here, we focus on applying neuromorphic technologies to texture detection and identification. Current literature surrounding texture classification with neuromorphic touch sensors has achieved high classification accuracy [3]–[5], often applying traditional machine learning classifiers in conjunction with spike-based approaches [6], [7]. While accuracy is critical, existing work has focused more on achieving the high classification performance provided by more conventional approaches and less on achieving real-time latency comparable to human perception.

Neuromorphic systems and algorithms are particularly well-suited to the rapid processing of information. Therefore, we propose within this paper a real-time tactile texture processing system utilising SNNs and a novel decision making algorithm (Fig. 1). Although classification performance remains of the utmost importance within this work, the ability of a system to classify this tactile information in a meaningful time frame is also key as we look toward applying these systems in real-world settings.

Our contributions are as follows:

- An end-to-end spiking system capable of classifying tactile texture data on neuromorphic hardware.
- A simple, tunable decision making algorithm for neuromorphic classification systems.
- An evaluation of power usage for our system running on different hardware configurations.

This work was partially supported by the EPSRC Centre for Doctoral Training in Future Autonomous and Robotic Systems (FARSCOPE) at the Bristol Robotics Laboratory and a Royal Academy of Engineering Research Fellowship on ”Shared autonomy neuroprosthetics: Bridging the gap between artificial and biological touch.” (RF\202021\20\171).

[1] Department of Aerospace Engineering, University of Bristol, United Kingdom `george.brayshaw@brl.ac.uk`

[2] Department of Engineering Mathematics, University of Bristol, United Kingdom `b.ward-cherrier@bristol.ac.uk`

[3] Bristol Robotics Laboratory, University of the West of England, United Kingdom `martin.pearson@brl.ac.uk`

979-8-3503-8457-4/24/$31.00 ©2024 IEEE

II. RELATED WORKS

Neuromorphic hardware innovations have looked to achieve the parallelism and power efficiency demonstrated by the mammalian brain by mimicking its computational processes and data sparsity. Event cameras showcase how data sparsity results in lower power consumption with Brandli et al. [8] reporting a 10 times decrease in power usage when compared to standard frame based cameras. Meanwhile systems such as SpiNNaker [9] and Loihi2 [10] provide hardware platforms for the deployment of spiking neural networks on architectures purpose built to imitate the parallelism of the brain. Deployment to these hardware platforms is key as running neuromorphic algorithms on traditional hardware such as GPUs has shown to provide little to no decrease in power consumption when compared to conventional methods [11].

The reduction in power consumption and processing latency afforded by this neuromorphic hardware underpins its utility within the domain of robotic systems [12], [13]. The potential for SNNs to achieve online learning, without forgetting previously trained states [14], also supports the adoption of neuromorphics within robotics, with excellent results presented for online learning during adaptive vision tasks [15].

The neuroTac [6] sensor looks to leverage the power efficiency of event cameras for use in neuromorphic tactile sensing. The high temporal resolution of our neuroTac sensor, able to update at 1KHz, enables it to detect the subtle, high frequency tactile features that are required for natural texture classification. Previous work by the authors reported that the use of simple machine learning algorithms was enough to accurately classify a range of artificial and textile textures, providing similar or improved performance ($\approx 70\%$ accuracy) to the Hierarchy of Event-Based Time-Surfaces (HOTS) neuromorphic pattern recognition system [16], [17].

Rongala et al. [18] have previously presented a system able to classify a set of textures with high accuracy ($90-97\%$), across a range of applied forces and movement velocities using a neuromorphic tactile sensor and Izhikevich neuron models. This system achieved high classification performance, but could not run in real-time due to the need to preprocess a given sample before using a k-nearest neighbours algorithm on features such as spike rate and covariance of the inter spike intervals (ISIs).

The works discussed thus far have achieved high classification accuracy for their respective texture datasets but are unable to classify in real-time. Rasouli et al [19] created a novel platform for the classification of textures in real-time. Their proposed system converted incoming sensor data into spikes for classification using an extreme learning machine. The system achieved a classification accuracy of 92% across a dataset of 10 3D printed textures. The system was also able to achieve this classification accuracy with under 1 second of input data. However it is limited by its inability to process spatio-temporal data, an important part of texture classification [20].

In work presented by Taunyazov et al. [21] neural encoding techniques are used to convert time series data from a number of different tactile sensors into spiking outputs. These outputs are then used to train and test a SNN for texture classification, boasting high classification performance of $\leq 94\%$, achieving these highly accurate classifications after only 500ms of input data. The conversion of sensor data into spikes using neural encoding is interesting given that it allows non-natively spiking sensors to interface with SNNs, however utilising neuromorphic sensing systems end-to-end as we do here helps to further decrease power consumption and increase information throughput by removing the need for an encoding step.

Within the field of robotics, with resources such as power and weight often restrained within real-world applications, neuromorphic hardware seems an obvious solution. This is provided that these neuromorphic systems still perform the tasks well and in a timely manner. Neuromorphics, though novel, appear to provide little in the way of improving classification performance when compared to traditional deep learning methods. While most research reports comparable performance to standard approaches, as discussed above, less focus has been on achieving the real-time latencies required for interactive robotics. Additionally, few studies have benchmarked neuromorphic solutions against conventional methods for speed and power efficiency. There remain opportunities to optimise spike-based neuromorphic pipelines for low-latency tactile texture recognition applicable to real-world robotic manipulation and prosthetics.

III. METHOD

A. Experimental Setup & Dataset

To acquire our dataset, we used a similar experimental setup as in our prior work [17]. Spiking data is collected using a neuroTac tactile sensor attached to a robotic arm. In Smith et al [22], the forces used by human subjects during texture exploration tasks is measured at $F = 1.51N \pm 0.5N$. Within our experiment we applied the following downward forces, which lie towards the lower bounds of this range; $0.8N, 1N, 1.2N$ to both encourage generalisation across a range of forces and due to the robustness of the current neuroTac tip design. This downwards force is verified by a force sensor connected to the arm. Once this downward force was applied to the texture, recording began whilst the sensor was moved by the arm across the texture. Moving the sensor across each texture at a constant velocity of 12 mm/s yielded 5000 ms of spiking data for each sample. Our dataset is comprised of 11 real-world fabrics and materials that you would expect to encounter regularly, seen in Fig. 1. The training set is comprised of 50 samples per texture, per force, yielding $n = 1650$ samples total.

B. Preprocessing & Offline Training

Our spiking networks were built and trained using Lava, an open source deep learning library built atop the PyTorch framework. Lava incorporates SLAYER [23] in order to train SNNs using a process akin to gradient descent via back propagation. This framework enables both the rapid, GPU-accelerated, training of SNNs as well as their deployment to Intel's neuromorphic hardware platform Loihi2, or an adjacent simulated environment, for inference.

In order to deploy to Loihi2, the input dimensionality of our neuroTac sensor (240x180) had to be reduced. To achieve this, we first applied a crop to the 2D space before employing a technique similar to that presented by Rizzo et al. [24], whereby the large input space provided by the DAVIS240 camera within the neuroTac is downsampled to a more practical size for input to our network. However, unlike the cited approach, we do not employ a standalone network to perform this pooling operation, instead using the following simplified algorithm:

A pooling window of size k_w x k_h moves across the camera frame with stride $stride$, with padding applied where required. Spikes are summed within this window at each time step. If a threshold number of spikes (M) is exceeded within a window, this results in an output spike within the output mapping.

This method is used, along with prior cropping of the input space, to reduce the size of the neuroTac output to a square 40x40 output. This square output from this pooling step is flattened for the networks input layer. The following pooling parameters are used within this work; $k_w = 4$, $k_h = 4$, $stride = 4$, $M = 1$.

Our networks are comprised of 3 densely connected layers of current-based (CUBA) LIF neurons [25]. An input layer of size 1600, followed by a single hidden layer of size 450 and a final output layer of size 11, giving a single neuron for each possible output classification. The size of the hidden layer was chosen from the result of a series of optimisations using a Tree-structured Parzen Estimator approach (TPE), implemented with the Hyperopt Python package [26].

During training, the output class is given by the output neuron with the highest spike rate over the duration of the sample.

In order to acquire a value for loss to back propagate through the network during training, Lava SLAYER uses a target-based loss function that aims to produce a high spike rate for the correct class and low rate for incorrect classes. During training we set a true spike rate of 0.8 and a false rate of 0.02. This translates as an ideal spike train with the correct output neuron spiking during 80% of time steps, with the other incorrect output neurons spiking during only 2% of time steps. These rates were hand tuned to produce the most accurate classifiers.

Each network was trained for 100 epochs and utilised k-fold cross validation ($k = 10$) to reduce overfitting. After training, the network was evaluated on a separate testing dataset made up of 50 samples per texture, with randomised downwards forces, to quantify overall classification accuracy.

C. Real-time Classification Algorithm

As previously discussed, Lava allows for both the training and running of SNNs in either a simulated hardware emulator or their deployment directly to the Loihi2 neuromorphic chip. For convenience, the classification and timing results presented in this paper were taken from the simulated hardware environment, with the power usage data taken from Loihi2 hardware via the Intel Neuromorphic Research Cloud (INRC).

When deployed, a classification is output from our decision maker process at every 1 ms timestep, allowing for inferences to be made before the completion of the entire 5000 ms trial. In order to gauge the progress of this real-time classification, a metric for confidence was defined. Confidence in the system's current classification (c) is measured by the percentage of total spikes (S_T) achieved by the highest spiking output neuron (n_{max}), after a set number of spikes (ϕ). This offset (ϕ) was added to filter out large fluctuations in confidence caused by low spike counts that are observed early in each trial and is representative of the system's temporal evidence accumulation. The formalisation for our decision making algorithm is given in equations 1- 4.

$$S_T = \sum_{n=0}^{N} S_n \tag{1}$$

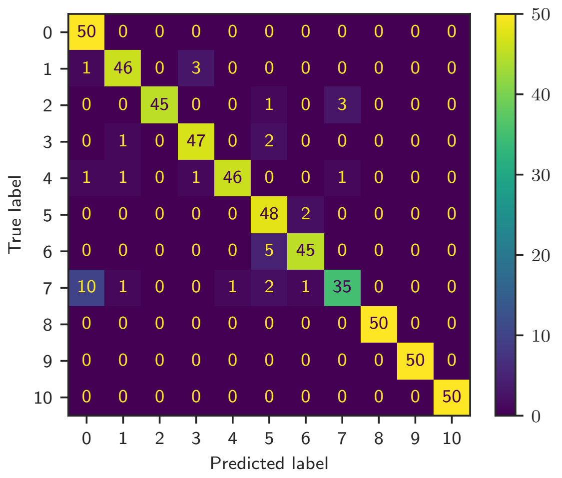
$$S_{max} = max\{S_0, ..., S_N\} \tag{2}$$

$$c = \begin{cases} 0 & S_T \leq \phi \\ \frac{S_{max}}{S_T} & S_T > \phi \end{cases} \tag{3}$$

$$c \geq \theta \Rightarrow \hat{y} = \arg\max\{S_0, ..., S_N\} \tag{4}$$

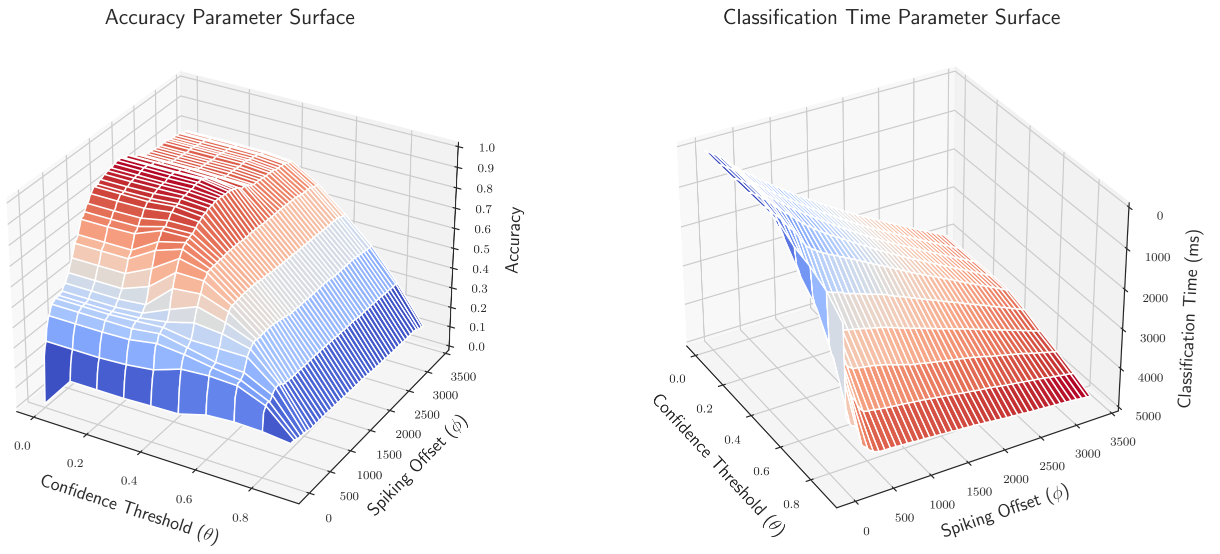
Where N is the number of output classes, in the case of our networks $N = 11$. S_n is the number of spikes output by neuron n and S_{max} is the number of spikes output by the highest spiking neuron.

Once the system's confidence has overcome a threshold value (θ), the system outputs a final classification for its current sample. This is formalised by equation 4 where $\hat{y}$ is our output classification.



True label \ Predicted label	0	1	2	3	4	5	6	7	8	9	10
0	50	0	0	0	0	0	0	0	0	0	0
1	1	46	0	3	0	0	0	0	0	0	0
2	0	0	45	0	0	1	0	3	0	0	0
3	0	1	0	47	0	2	0	0	0	0	0
4	1	1	0	1	46	0	0	1	0	0	0
5	0	0	0	0	0	48	2	0	0	0	0
6	0	0	0	0	0	5	45	0	0	0	0
7	10	1	0	0	1	2	1	35	0	0	0
8	0	0	0	0	0	0	0	0	50	0	0
9	0	0	0	0	0	0	0	0	0	50	0
10	0	0	0	0	0	0	0	0	0	0	50

Fig. 2: Confusion Matrix for the testing of our 5000ms trained network.


Accuracy Parameter Surface

Classification Time Parameter Surface

Fig. 3: ϕ and θ surfaces showing how parameters vary the accuracy and time taken for a classification with our 5000 ms trained classifier.

D. Temporal Performance Metrics

In order to appraise the temporal performance of our system we define two metrics. These two metrics are given by the mean number of time steps required by a given classifier to classify each testing sample at, or above, a threshold accuracy. Time-to-peak (t_p) is the time for the classifier to achieve its maximum accuracy (Equation 5) and time-to-human (t_h) represents the time to achieve a human baseline performance of 65% accuracy (Equation 6). This human baseline is derived from a literature review into human texture classification performance within [17], [27]–[29].

$$t_p = min\, t(\theta, \phi) : Acc(\theta, \phi) = Acc_{max} \tag{5}$$

$$t_h = min\, t(\theta, \phi) : Acc(\theta, \phi) \geq 65 \tag{6}$$

Where t is the time step and θ and ϕ are the tunable parameters from the decision making algorithm defined in Equations 1- 4.

E. Power Metrics

The power usage of a neuromorphic system is an important metric to report, with superior power consumption being one of the main arguments for developing these systems.

To compare the energy performance of our system we measured its power consumption when deployed across three different hardware configurations; onboard a Loihi2, an AMD Ryzen Threadripper 2920X CPU running a simulated Loihi2 environment and an NVIDIA RTX A4500 GPU running inference in PyTorch.

To collect data on Loihi2 performance, the Lava framework contains a suite of profiling tools. CPU and GPU performance was logged using Running Average Power Limit Energy (RAPL) and NVIDIA *pynvml* tools respectively.

IV. RESULTS

A. Accuracy and Classification Performance

Our network achieved accuracy, recall, precision and f1 scores of 0.94 across our testing dataset. Fig. 2 shows the confusion matrix of our network testing. The network is shown in Fig. 2 to achieve between 100% (50/50) and 70% (35/50) in classification accuracies across the 11 textures tested.

B. Temporal Classification Performance

After acquiring the results shown above for overall classification performance, we looked to analyse the temporal performance of our classifier.

Once our network is deployed within our real-time classifier we make classifications using the algorithm described in section III-C. By varying the spiking offset (ϕ) and confidence threshold (θ) parameters of this real-time classification we are able to further tune the accuracy and speed of our networks. Fig. 3 presents the classification and temporal performance of our 5000ms classifier when varying these ϕ and θ parameters to affect the speed/accuracy trade off. The figure demonstrates how an increase in spiking offset (ϕ) tends to aid the accuracy of the system, however providing a larger initial delay also incurs a reduction in temporal performance. Confidence threshold values (θ) exceeding ≈ 0.4 begin to degrade overall performance in both metrics as confidence of the system seems to rarely rise beyond these values.

Values of ϕ and θ were optimised for peak and human performance using Equations 5 and 6 respectively. These

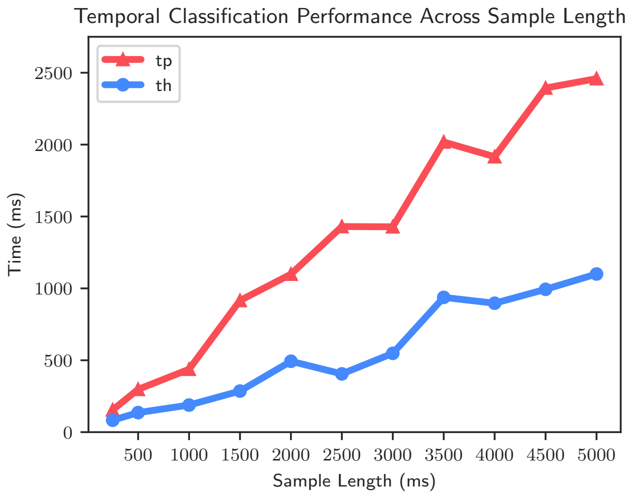


Fig. 4: The temporal performance of our networks are appraised via the metrics introduced in section III-D. An approximately linear relationship is shown to emerge between the length of sample used to train the network and its temporal performance.

parameters and their associated t_p and t_h values and shown in Table I.

C. Reducing Time for Classification

Prior work with neuroTac texture data [17] demonstrated that the initial 1000ms of a 5000ms sample contains enough information to classify textures. To verify the efficacy of reducing training sample length for SNNs we trained a series of networks with increasingly reduced sample lengths. These reduced samples were created by temporally cropping the input data used to train and test the networks to a given length.

Table I shows the classification performance of classifiers trained using different lengths of input sample. A fairly steady performance is shown across all metrics for sample sizes of 250-5000ms with all classifiers achieving accuracy scores of ≥ 0.85.

Fig. 4 demonstrates how changing this sample length affects the temporal performance metrics. An almost linear relationship appears between sample length and both t_p and t_h. Therefore, a reduction in sample length coincides with a reduction in both t_p and t_h while not significantly affecting accuracy.

D. Power Usage

All power results are presented as average energy consumption across the entire test set. The idle power of each of the systems is also shown to give a better understanding of typical consumption during deployment. Idle consumption of each platform is averaged over 5 minutes under negligible load. Fig. 5 shows this platform-wise comparison.

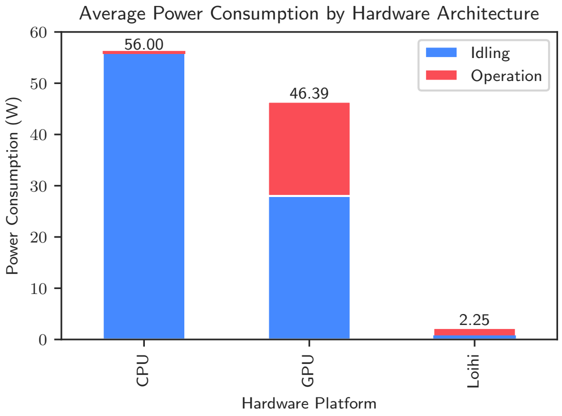


Fig. 5: Bar plot illustrating the variation in power efficiency among the three platforms examined. The baseline power consumption during idle periods was registered at $56W$ for the CPU, $28W$ for the GPU, and $0.5W$ for Loihi2. Furthermore, the additional power consumption while in operation, exceeding the idle levels, was determined to be $0.005W$, $18W$, and $1.75W$ for the CPU, GPU, and Loihi2 devices, respectively.

As shown, the overall power consumed by the Loihi2 platform is low in comparison to the less specialised hardware platforms, providing an overall power consumption reductions of $\approx$ 96% and $\approx$ 95% when compared to respective CPU and GPU implementations. The idling power of the Loihi2 is negligible at $0.5W$, with most of its power consumption occurring when processing incoming information ($2.25W$). The utility of the CPU and GPU systems for remote or low-power systems are both hindered by their relatively high overall consumption ($56.005W$ and $46.39W$). Although a reduction in usage is seen for the GPU when compared to our CPU implementation, it again is a large increase in consumption when compared to the Loihi2. It is worth noting that the operating power of the CPU system is fairly consistent with its idling power ($56.005W$ and $56W$), indicating that the CPU efficiently processes the required information but that its power overhead is high.

V. DISCUSSION AND FUTURE WORK

Neuromorphic hardware provides an excellent avenue for the real-time sensing and processing of tactile information. This system is an important step towards including neuromorphic sensors and processing within the control loop of robotic systems. Our texture classification networks showcase flexibility by their ability to not only achieve rapid, real-time classifications but also adjust their speed-accuracy trade-off, via the ϕ and θ parameters. Peak accuracies of 94% on our dataset exceeded those achieved using non-spiking classifiers with the same sensor [6], [17] and demonstrated the system's ability to identify subtly distinct, naturally occurring textures. This classification performance also compares well to other State-Of-Art (SOA) works on tactile texture classification discussed within section II.

Our network also demonstrated rapid texture identification, achieving respective minimum values of $84ms$ and $154ms$ for t_h and t_p on our dataset. This reduces the minima

TABLE I: Threshold and Offset parameters when optimising for either human (t_h) or peak (t_p) classification accuracy. The tuning of the decision making algorithm allows for emphasis to be placed on highly accurate, slower classification or slightly lower accuracy, rapid classifications.

Sample Length (ms)	250	500	1000	1500	2000	2500	3000	3500	4000	4500	5000
Peak Accuracy	0.93	0.85	0.93	0.94	0.92	0.94	0.93	0.90	0.89	0.91	0.94
t_p (ms)	154	298	437	916	1099	1430	1428	2018	1916	2394	2459
Threshold θ	0.3	0.3	0.4	0.4	0.3	0.3	0.3	0.2	0.2	0.2	0.3
Offset ϕ	150	200	300	625	850	1150	1025	1675	1450	1900	1800
t_h (ms)	83	135	188	286	493	405	548	937	897	994	1100
Threshold θ	0.2	0.4	0.4	0.3	0.3	0.2	0.3	0.3	0.2	0.3	0.3
Offset ϕ	75	75	125	175	325	325	350	675	650	750	675

achieved in prior work [17] by 79% ($t_h : 400ms \rightarrow 84ms$) as well as the minima achieved for this level of classification performance within both [19] and [21].

The time taken to reach an output classification is of the utmost importance when optimising real-time performance, although it often comes at the expense of accuracy. Across all sample lengths, maintaining a confidence value of $\theta \leq 0.4$ appears to be essential for maintaining classification performance at or above our human baseline (65%). Increasing the spike offset (ϕ), and thus the amount of information acquired before making a classification, improved accuracy across all sample lengths. This however comes at the expense of a higher time required for classification. This trade off demonstrates the versatility of our presented classification algorithm by achieving both high classification accuracies and low classification latencies. The simplicity of the classifier also grants the potential for online tuning via a feedback mechanism. While this is not investigated during this work it is an avenue for future work.

Training our networks using shorter periods of data has little impact on the accuracy of our classifiers, whilst vastly reducing the time taken to make classifications. This provides further evidence to support statements from the authors' prior neuroTac work hypothesising that the events generated by the sensor's initial overcoming of static friction provides enough information to accurately classify texture [17]. This hypothesis was tested further here, reducing our minimum sample length from $1000ms$ to $250ms$ in this work, and to our knowledge we have created the fastest tactile texture classification system capable of this level of accuracy.

Interestingly, an unintended effect of our sample length optimisation was a large reduction in network training times. This will affect future works with Lava SLAYER by reducing development time and power expended during the training of our classifiers.

As presented within our benchmark above, the Loihi2 system is shown to provide comparable performance to the same system implemented on more traditional computing hardware. Both the overall power usage of the system during inference and the baseline idling power usage is shown to outperform both a GPU and CPU system. It must be noted that the networks deployed to the Loihi2 during these experiments were large in size, utilising 68 of the 120 neuromorphic cores available on the target chip. A reduction in model size or a reduction in the number of bits per weight in our model may lead to lower operating power consumption. Although the networks presented within this work are large and densely connected, smaller more nuanced networks are planned for future work. It is also worthy of note that robotic tactile interaction with the environment is inherently intermittent and sparse in nature. In other words, tactile sensors are typically not in contact with a surface, and therefore idle, for most of the time. This operational characteristic further leverages the use of low power, event driven sensing and processing as proposed here.

Work presented within this paper has been undertaken on the NRC, providing remote access to Loihi2 hardware. If this system is to be implemented in conjunction with real-world manipulators, hardware-in-the-loop is required. In future work we plan to directly integrate a physical Loihi2 chip in the loop for live classifications.

VI. CONCLUSIONS

The work presented within this paper showcases the viability of real-time texture classification using an entirely neuromorphic pipeline. Our work achieves comparable classification performance to the SOA (94% accuracy), while achieving considerable improvements in classification speeds (79% reduction) and power consumption ($\leq 96\%$ reduction).

REFERENCES

[1] Z. Yi, Y. Zhang, and J. Peters, "Biomimetic tactile sensors and signal processing with spike trains: A review," *Sensors and Actuators A: Physical*, vol. 269, pp. 41–52, 2018.

[2] C. Bartolozzi, "Neuromorphic circuits impart a sense of touch," *Science*, vol. 360, no. 6392, pp. 966–967, 2018.

[3] U. B. Rongala, A. Mazzoni, and C. M. Oddo, "Neuromorphic artificial touch for categorization of naturalistic textures," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 4, pp. 819–829, 2017.

[4] S. Sankar, D. Balamurugan, A. Brown, K. Ding, X. Xu, J. H. Low, C. H. Yeow, and N. Thakor, "Texture discrimination with a soft biomimetic finger using a flexible neuromorphic tactile sensor array that provides sensory feedback," *Soft Robotics*, vol. 8, no. 5, pp. 577–587, 2021, pMID: 32976080.

[5] A. K. Gupta, R. Ghosh, A. N. Swaminathan, B. Deverakonda, G. Ponraj, A. B. Soares, and N. V. Thakor, "A neuromorphic approach to tactile texture recognition," in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2018, pp. 1322–1328.

[6] B. Ward-Cherrier, N. Pestell, and N. F. Lepora, "Neurotac: A neuromorphic optical tactile sensor applied to texture recognition," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2654–2660.

[7] H. Nguyen, L. Osborn, M. Iskarous, C. Shallal, C. Hunt, J. Betthauser, and N. Thakor, "Dynamic texture decoding using a neuromorphic multilayer tactile sensor," in *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2018, pp. 1–4.

[8] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, "A 240 × 180 130 db 3 µs latency global shutter spatiotemporal vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014.

[9] S. B. Furber, D. R. Lester, L. A. Plana, J. D. Garside, E. Painkras, S. Temple, and A. D. Brown, "Overview of the spinnaker system architecture," *IEEE Transactions on Computers*, vol. 62, pp. 2454–2467, 2013.

[10] M. Davies, A. Wild, G. Orchard, Y. Sandamirskaya, G. A. F. Guerra, P. Joshi, P. Plank, and S. R. Risbud, "Advancing neuromorphic computing with loihi: A survey of results and outlook," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 911–934, 2021.

[11] G. Parpart, S. R. Risbud, G. T. Kenyon, and Y. Watkins, "Implementing and benchmarking the locally competitive algorithm on the loihi 2 neuromorphic processor," *arXiv preprint arXiv:2307.13762*, 2023.

[12] A. D. Rast, S. V. Adams, S. Davidson, S. Davies, M. Hopkins, A. Rowley, A. B. Stokes, T. Wennekers, S. Furber, and A. Cangelosi, "Behavioral learning in a cognitive neuromorphic robot: an integrative approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 12, pp. 6132–6144, 2018.

[13] F. Galluppi, C. Denk, M. C. Meiner, T. C. Stewart, L. A. Plana, C. Eliasmith, S. Furber, and J. Conradt, "Event-based neural computing on an autonomous mobile platform," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 2862–2867.

[14] T. Zhang, X. Cheng, S. Jia, C. T. Li, M.-m. Poo, and B. Xu, "A brain-inspired algorithm that mitigates catastrophic forgetting of artificial and spiking neural networks with low computational cost," *Science Advances*, vol. 9, no. 34, p. eadi2947, 2023.

[15] E. Hajizada, P. Berggold, M. Iacono, A. Glover, and Y. Sandamirskaya, "Interactive continual learning for robots: a neuromorphic approach," in *Proceedings of the International Conference on Neuromorphic Systems 2022*, 2022, pp. 1–10.

[16] X. Lagorce, G. Orchard, F. Galluppi, B. E. Shi, and R. B. Benosman, "Hots: a hierarchy of event-based time-surfaces for pattern recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 7, pp. 1346–1359, 2016.

[17] G. Brayshaw, B. Ward-Cherrier, and M. Pearson, "Temporal and spatio-temporal domains for neuromorphic tactile texture classification," in *Neuro-Inspired Computational Elements Conference*, 2022, pp. 50–57.

[18] U. B. Rongala, A. Mazzoni, and C. M. Oddo, "Neuromorphic artificial touch for categorization of naturalistic textures," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 4, pp. 819–829, 2015.

[19] 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.

[20] A. K. Gupta, A. Nakagawa-Silva, N. F. Lepora, and N. V. Thakor, "Spatio-temporal encoding improves neuromorphic tactile texture classification," *IEEE Sensors Journal*, vol. 21, no. 17, pp. 19 038–19 046, 2021.

[21] T. Taunyazov, Y. Chua, R. Gao, H. Soh, and Y. Wu, "Fast texture classification using tactile neural coding and spiking neural network," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9890–9895.

[22] A. M. Smith, G. Basile, J. Theriault-Groom, P. Fortier-Poisson, G. Campion, and V. Hayward, "Roughness of simulated surfaces examined with a haptic tool: effects of spatial period, friction, and resistance amplitude," *Experimental brain research*, vol. 202, no. 1, pp. 33–43, 2010.

[23] S. B. Shrestha and G. Orchard, "Slayer: Spike layer error reassignment in time," *Advances in neural information processing systems*, vol. 31, 2018.

[24] C. P. Rizzo, C. D. Schuman, and J. S. Plank, "Neuromorphic downsampling of event-based camera output," in *Proceedings of the 2023 Annual Neuro-Inspired Computational Elements Conference*, 2023, pp. 26–34.

[25] S. Cavallari, S. Panzeri, and A. Mazzoni, "Comparison of the dynamics of neural interactions between current-based and conductance-based integrate-and-fire recurrent networks," *Frontiers in neural circuits*, vol. 8, p. 12, 2014.

[26] J. Bergstra, D. Yamins, and D. Cox, "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures," in *International conference on machine learning*. PMLR, 2013, pp. 115–123.

[27] B. P. Delhaye, E. W. Schluter, and S. J. Bensmaia, "Robo-psychophysics: Extracting behaviorally relevant features from the output of sensors on a prosthetic finger," *IEEE transactions on haptics*, vol. 9, no. 4, pp. 499–507, 2016.

[28] S. Chun, I. Hwang, W. Son, J.-H. Chang, and W. Park, "Recognition, classification, and prediction of the tactile sense," *Nanoscale*, vol. 10, no. 22, pp. 10 545–10 553, 2018.

[29] A. Amini, J. I. Lipton, and D. Rus, "Uncertainty aware texture classification and mapping using soft tactile sensors," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4249–4256.