Event-Driven Visual-Tactile Sensing and Learning for Robots

Anonymous Authors
Somewhere, Someplace

***Abstract*—This work contributes an event-driven visual-tactile perception system, comprising a novel biologically-inspired tactile sensor and multi-modal spike-based learning. Our biologically-inspired fingertip tactile sensor, NeuTouch, scales well with the number of taxels thanks to its event-based nature. Likewise, our Visual-Tactile Spiking Neural Network (VT-SNN) enables fast perception when coupled with event sensors. We evaluate our visual-tactile system (using the NeuTouch and Prophesee event camera) on two robot tasks: container classification and rotational slip detection. On both tasks, we observe good accuracies relative to standard deep learning methods. We have made our visual-tactile datasets freely-available to encourage research on multi-modal event-driven robot perception, which we believe is a promising approach towards intelligent power-efficient robot systems.**

***Index Terms*—Event-Driven Perception, Multi-Modal Learning, Tactile Sensing, Spiking Neural Networks.**

I. Introduction

Many everyday tasks require multiple sensory modalities to perform successfully. For example, consider fetching a carton of soymilk from the fridge [1]; humans use vision to locate the carton and can infer from a simple grasp how much soymilk the carton contains. They are also able to use their sense of sight and touch to lift the object without letting it slip. These actions (and inferences) are performed robustly using a power-efficient neural substrate — compared to popular deep learning approaches for using multiple sensor modalities in artificial systems, human brains require far less energy [2], [3].

In this work, we take crucial steps towards efficient visual-tactile perception for robotic systems. We gain inspiration from biological systems, which are *asynchronous* and *event-driven*. In contrast to resource-hungry deep learning methods, event-driven perception forms an alternative approach that promises power-efficiency and low-latency — features that are ideal for real-time mobile robots. However, event-driven systems remain under-developed relative to standard synchronous perception methods [4], [5].

This paper makes multiple contributions that advance event-driven *visual-tactile* perception. First, to enable richer tactile sensing, we contribute the 39-taxel *NeuTouch fingertip sensor*. Compared to existing commercially-available tactile sensors, NeuTouch's neuromorphic design enables scaling to a larger number of taxels while retaining low latencies.

Next, we investigate multi-modal learning with NeuTouch and the Prophesee event camera. Specifically, we develop a *visual-tactile spiking neural network* (VT-SNN) that incorporates both sensory modalities for supervised-learning tasks. Different from conventional deep artificial neural network (ANN) models [6], SNNs process discrete spikes asynchronously and thus, are arguably better suited to the event data generated by our neuromorphic sensors. In addition, SNNs can be used on efficient low-power neuromorphic chips such as the Intel Loihi [7].

Our experiments center on two robot tasks: object classification and (rotational) slip detection. In the former, we tasked the robot to determine the type of container being handled and amount of liquid held within. The containers were opaque with differing stiffness, and hence, both visual and tactile sensing are relevant for accurate classification. We show that relatively small differences in weight (≈ 30g across 20 object-weight classes) can be distinguished by our prototype sensors and spiking models. Likewise, the slip detection experiment indicates rotational slip can be accurately detected within 0.08s (visual-tactile spikes processed every ≈ 1ms). In both experiments, SNNs achieved competitive (and sometimes superior) performance relative to ANNs with similar architecture.

Taking a broader perspective, event-driven perception represents an exciting opportunity to enable power-efficient intelligent robots. This work suggests that an "end-to-end" event-driven perception framework is promising and warrants further research. We have made the data from our two experiments (and a third dataset that expands the number of grasped items to 36 different objects using the same protocol) freely-available to the research community[1]. To our knowledge, these represent the first publicly-available *event-based* visual-tactile datasets, and we hope their availability will spur research on event-driven robotics. To summarize, our primary contributions are:

- NeuTouch, a scalable event-based tactile sensor for robot end-effectors;
- A Visual-Tactile Spiking Neural Network that leverages multiple event sensor modalities;
- Systematic experiments demonstrating the effectiveness of our event-driven perception system on object classification and slip detection, with comparisons to conventional ANN methods;
- Visual-tactile event sensor datasets comprising more than 50 different object classes across our experiments, which also includes RGB images and proprioceptive data from the robot.

[1]Access URL blinded for review.

II. Background and Related Work

In the following, we give a brief overview of related work on visual-tactile perception for robotics, and event-driven sensing and learning. Both areas are broad and thus, we focus on the core concepts and provide links to articles that cover these topics in greater detail.

A. Visual-Tactile Perception for Robots

General recognition of the importance of multi-modal sensing for robotics has led to innovations both in sensing and perception methods. Of late, there has been a flurry of papers on combining vision and touch sensing, e.g., [8]–[13]. However, work on visual-tactile learning of objects dates back to (at least) 1984 when vision and tactile data was used to create a surface description of primitive objects [14]; in this early work, tactile sensing played supportive role for vision due to the low resolution of tactile sensors at the time.

Recent advancements in tactile technology [15] have encouraged the use of tactile sensing for more complex tasks, including object exploration [16], shape completion [17], and slip detection [18], [19]. One popular sensor is the BioTac; similar to a human finger, it uses textured skin, allowing vibration signatures to be used for high accuracy material and object identification and slip detection [20]. The BioTac has also been used in visual-tactile learning, e.g., [9] combined tactile data with RGB images to recognize objects via deep learning. Other recent works have used the Gelsight [21] — an optical-based tactile sensor — for visual-tactile slip detection [10], [22], grasp stability, and texture recognition [23]. Very recent work has used unsupervised learning to generate neural representations of visual-tactile data (with proprioception) for reinforcement learning [11]. Compared to the prior work above, we do not leverage synchronous sensors or conventional deep learning methods. Rather, our sensor and learning method are both event-driven.

B. Event-based Perception: Sensors and Learning

Work on event-based perception has focused primarily on vision (see [24] for a comprehensive survey). The emphasis on vision can be attributed both to its applicability across many tasks, as well as the recent availability of event cameras such as the DVS and Prophesee Onboard; unlike conventional optical sensors, event cameras capture pixel changes asynchronously.

Event-based sensors have been successfully used in conjunction with deep learning techniques [24]. The binary events are first converted into real-valued tensors, which are processed downstream by deep ANNs. This approach generally yields good models (e.g., for motion segmentation [25], optical flow estimation [26], and car steering prediction [27]), but at high compute cost.

Neuromorphic learning, specifically Spiking Neural Networks (SNNs) [4], [28], provide a competing approach for learning with event data. Similar to event-based sensors, SNNs work directly with discrete spikes and hence, possess similar characteristics, i.e., low latency, high temporal resolution and low power consumption. Historically, SNNs have been hampered by the lack of a good training procedure. Gradient-based methods such as backpropagation were not available because spikes are non-differentiabile. Recent developments in effective SNN training [29]–[31] and the nascent availability of neuromorphic hardware (e.g., IBM TrueNorth [32] and Intel Loihi [7]) have renewed interest in neuromorphic learning for various applications, including robotics. SNNs do not yet consistently outperform their deep ANN cousins on pseudo-event image datasets, and the research community is actively exploring better training methods for real event-data.

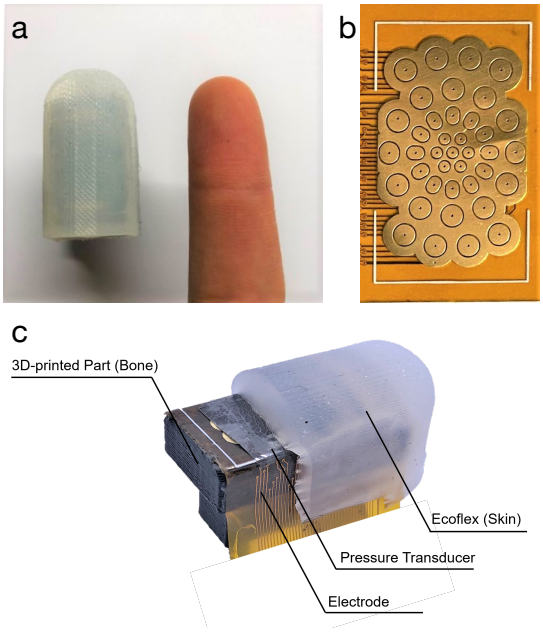

Fig. 1. (a) NeuTouch compared to a human finger. (b) Spatial distribution of the 39 taxels on NeuTouch. (c) Cross-sectional view of NeuTouch and constituent components. NeuTouch performs tactile sensing using an electrode layer with 39 taxels and a graphene-based piezoresistive thin film that is embedded beneath the protective Ecoflex "skin".

In this work, we develop a visual-tactile SNN trained using SLAYER [29], a recent (approximate) backpropagation method for SNNs. There has been relatively little work on multi-modal SNNs and prior work has focused on audio-video data [33], [34] (e.g., for emotion detection [35]). Very recent work [36] has investigated turn-taking in human-robot collaboration using a range of non-neuromorphic sensors (Kinect video, EEG etc.). In contrast, we develop a multi-modal SNN for visual and *tactile* event data; note that the latter is an "active" sense with different characteristics from audio.

III. NeuTouch: An Event-based Tactile Sensor

Although there are numerous applications for tactile sensors (e.g., minimal invasive surgery [37] and smart prosthetics [38]), current tactile sensing technology lags behind vision. In particular, current tactile sensors remain difficult to scale and integrate with robot platforms. The reasons are twofold: first, many tactile sensors are interfaced via time-divisional multiple access (TDMA), where individual taxels are *periodically* and *sequentially* sampled. The serial readout

nature of TDMA inherently leads to an increase of readout latency as the number of taxels in the sensor is increased. Second, high spatial localization accuracy is typically achieved by adding more taxels in the sensor; this invariably leads to more wiring, which complicates integration of the skin onto robot end-effectors and surfaces.

Motivated by the limitations of the existing tactile sensing technology, we developed a novel Neuro-inspired Tactile sensor (NeuTouch) for use on robot end-effectors (Fig. 1). The structure of NeuTouch resembles a human fingertip: it comprises "skin", and "bone", and has a physical dimension of $37 \times 21 \times 13$ mm. This design facilitates integration with anthropomorphic end-effectors (for prosthetics or humanoid robots) and standard multi-finger grippers; in our experiments, we use NeuTouch with a Robotiq 2F-140 gripper. We focused on a fingertip design in this paper, but alternative structures can be developed to suit different applications.

Tactile sensing is achieved via a layer of electrodes with 39 taxels and a graphene-based piezoresistive thin film. The graphene-based pressure transducer forms an effective tactile sensor [39], [40] due to its high Young's modulus, which helps to reduce the transducer's hysteresis and response time. We employed a 3D-printed component to serve the role of the fingertip bone, and Ecoflex 00-30 (Ecoflex) to emulate skin for NeuTouch. The Ecoflex offers protection for the electrodes for a longer use-life and amplifies the stimuli exerted on NeuTouch. The latter enables more tactile features to be collected, since the transient phase of contact (between object and sensor) encodes much of the physical description of a grasped object, such as stiffness or surface roughness [41]. The NeuTouch exhibits a slight delay of $\approx$ 300ms when recovering from a deformation due to the soft nature of Ecoflex. Nevertheless, our experiments in Secs. VI and VII show this effect did not impede the NeuTouch's sensitivity to various tactile stimuli.

Compared to existing tactile sensors, NeuTouch is *event-based* and scales well with the number of taxels; NeuTouch can accommodate 240 taxels while maintaining an exceptionally low constant readout latency of 1ms for rapid tactile perception [42]. We achieve this by leveraging upon the Asynchronously Coded Electronic Skin (ACES) platform [42] — an event-based neuro-mimetic architecture that enables *asynchronous* transmission of tactile information. With ACES, the taxels of NeuTouch mimic the function of the fast-adapting (FA) mechano-receptors of a human fingertip, which capture dynamic pressure (i.e., dynamic skin deformations) [43]. FA responses are crucial for dexterous manipulation tasks that require rapid detection of object slippage, object hardness, and local curvature.

Crucially, transmission of the stimuli information is in the form of asynchronous spikes (i.e., electrical pulses), similar to biological systems; data is transmitted by individual taxels *only when necessary* via *single common conductor* for signalling. This is made possible by encoding the taxels of NeuTouch with unique electrical pulse signatures. These signatures are robust to overlap and permit multiple taxels to transmit data *without*

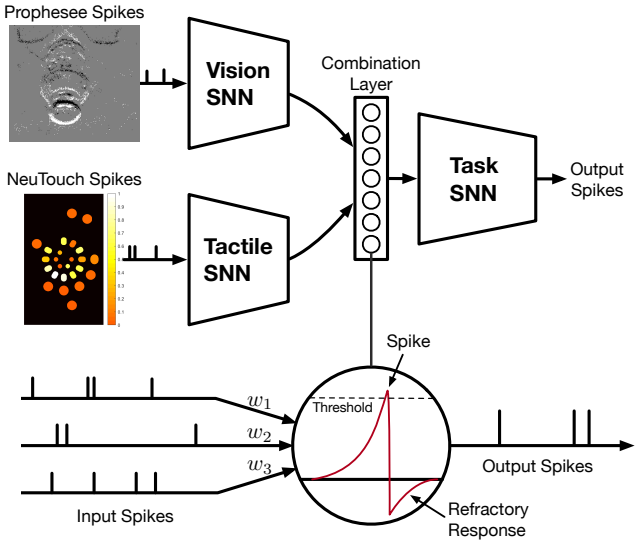

Fig. 2. The Visual-Tactile Spiking Neural Network (VT-SNN) which comprises two "spiking encoders" for each modality. The spikes from these two encoders are combined via a fixed-width combination layer and propagated to a task SNN that outputs a task-specific output spike-train. VT-SNN employs the Spike Response Model (SRM) neuron that integrates incoming spikes and spikes when a threshold is breached.

specific time synchronization. Therefore, stimuli information of all the activated taxels can be combined and propagated upstream to the decoder via a single electrical conductor. This yields lower readout latency and simpler wiring. The decoder correlates the received pulses (i.e., the combined pulse signatures) against each taxel's known signature to retrieve the spatio-temporal tactile information. In the next section, we describe how the decoded tactile event data is used for learning and classification.

IV. Visual-Tactile Spiking Neural Network (VT-SNN)

As motivated in the introduction, the successful completion of many tasks is contingent upon using multiple sensory modalities. In this work, we focus on touch and sight, i.e., we fuse tactile and visual data from NeuTouch and an event-based camera via a spiking neural model. This Visual-Tactile Spiking Neural Network (VT-SNN) enables learning and perception using both these modalities, and can be easily extended to incorporate other event sensors.

Model Architecture. From a bird's-eye perspective, the VT-SNN employs a simple architecture (Fig. 2) that first encodes the two modalities into individual latent (spiking) representations, that are combined and further processed through additional layers to yield a task-specific output.

We now detail the precise network structures used in our experiments, but VT-SNN may use alternative network structures for the Tactile, Vision and Task SNNs. The Tactile SNN employs a fully connected (FC) network consisting of 2 dense

spiking layers[2]. It has an input size of 156 (two fingers, each with the 39 taxels with a positive and negative polarity channel per taxel) and a hidden layer size of 32. The Vision SNN uses 3 layers; the first layer is a pooling layer with kernel size and stride length of 4. The pooled spike train is passed as input to a 2-layer FC architecture identical to the Tactile SNN. The tactile and vision encoders have output sizes of 50 and 10, respectively[3]. The encoded spike-trains of both modalities are concatenated, and passed into a dense spiking layer that generates output spikes. Note that the output dimensionality is dependent on the task: 20 for container & weight classification, and 2 for rotational slip classification. The model architectures are agnostic to the size of the input time dimension, and the same model architectures are used in both classification tasks.

Neuron Model. We use the Spike Response Model (SRM) [29], [44]. In the SRM, Spikes are generated whenever a neuron's internal state ("membrane potential") $u(t)$ exceeds a predefined threshold φ. Each neuron's internal state is affected by incoming spikes and a refractory response:

$$u(t) = \sum w_i(\epsilon * s_i)(t) + \nu * o)(t) \tag{1}$$

where w_i is a synaptic weight, $*$ indicates convolution, $s_i(t)$ are the incoming spikes from input i, $\epsilon(\cdot)$ is the response kernel, $\nu(\cdot)$ is the refractory kernel, and $o(t)$ is the neuron's output spike train. In words, incoming spikes $s_i(t)$ are convolved with a response kernel $\epsilon(\cdot)$ to yield a spike response signal that is scaled by a synaptic weight w_i.

Model Training. We optimized our spiking networks using SLAYER [29]. As mentioned in Sec. II, the derivative of a spike is undefined, which prohibits a direct application of backpropagation to SNNs. SLAYER overcomes this problem via a stochastic spiking neuron approximation to derive an approximate gradient, and a temporal credit assignment policy to distribute errors. SLAYER trains models "offline" on GPU hardware. Hence, the spiking data needs to be binned into fixed-width intervals during the training process, but the resultant SNN model can be run on neuromorphic hardware. We used a straight-forward binning process where the (binary) value for each bin window V_w was 1 whenever the total spike count in that window $\sum_w S$ exceeded a threshold value S_{min}:

$$V_w = \begin{cases} 1 & \sum_w S \geq S_{\text{min}} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Following [29], class prediction is determined by the number of spikes in the output layer spike train; each output neuron is associated with a specific class and the neuron that generates the most spikes represents the winning class. We trained the model by minimizing the loss:

$$\mathcal{L} = \frac{1}{2}\sum_{t=0}^{T}\left(\sum \mathbf{s}^o(t) - \sum \tilde{\mathbf{s}}^o(t)\right)^2 \tag{3}$$

[2]In preliminary experiments, we also tested convolutional layers but it resulted in poorer performance.

[3]Several different dimension sizes were tested and a 50-10 encoding gave the best results.

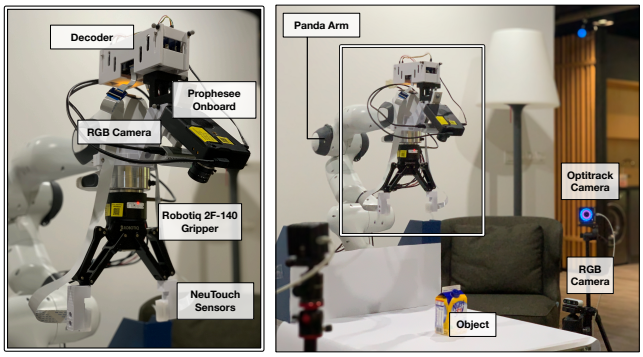

Fig. 3. Robot Experiment Setup. (**Left**) Close-up of the Franka Emika Panda arm and sensors; a NeuTouch sensor was attached on each Robotiq gripper finger. The Prophesee event and Realsense cameras were mounted on the arm and pointed towards the center of the gripper's grasp area. Our prototype ACES decoder was mounted on top of the arm's control panel. (**Right**) A view of the object classification experiment showing three OptiTrack cameras (11 were used, and out of scene), the RGB scene camera, and the object (soy milk carton) to be grasped and lifted.

which captures the difference between the observed output spike count $\sum \mathbf{s}^o(t)$ and the desired spike count $\sum \tilde{\mathbf{s}}^o(t)$ for output neuron o. Appropriate counts have to be specified for the correct and incorrect class and are application-specific hyperparameters. We tuned them manually for each task. In our experiments, setting the positive class count to $\approx 50\%$ of the maximum number of spikes (across each input within the considered time interval) worked well. For example, if the maximum number of incoming spikes across the neurons was 300, then we set a desired spike count of 150 for the positive class. The negative class counts were set to a low value (5 for our experiments).

From initial trials, we observed that training solely with the classification loss above on our data led to rapid over-fitting and poor performance on a validation set. We explored several different techniques to mitigate this issue (e.g., ℓ_1 regularizers and dropout), but found a simple ℓ_2 regularization term led to the best results:

$$\hat{\mathcal{L}} = \mathcal{L} + \gamma_2\|\theta\|_2^2 \tag{4}$$

where θ comprise the network weights. For all our SNN models, we optimized $\hat{\mathcal{L}}$.

V. Robot and Sensors Setup

In this section, we describe the robot hardware setup used across our experiments (Fig. 3). We used a 7-DoF Franka Emika Panda arm with a Robotiq 2F-140 gripper and collected data from four primary sensors types: NeuTouch, Prophesee Onboard, RGB cameras, and the Optitrack motion capture system. The latter two are non-event sensors and their data streams were not used in VT-SNN.

NeuTouch Tactile Sensor. We mounted two NeuTouch sensors to the Robotiq 2F-140 gripper and the ACES decoder on the Panda arm (Fig. 3, left). To ensure consistent data, we performed a sensor warm-up before each data collection session and obtained baseline results to check for sensor drift.

Specifically, we repeated 100 cycles of: closing the gripper onto a flat stiff object (the '9 hole peg test' from the YCB dataset [45]) for 3 seconds, opening the gripper, and pausing for 2 seconds. We then collected a set of benchmark data, i.e., 20 repetitions of closing the gripper onto the same '9 hole peg test' for 3 seconds. Throughout our experiments, we periodically tested for sensor drift by repeating the closing test on the '9 hole peg test' and then examining the sensor data; no significant drift was found throughout our experiments.

Prophesee Event Camera. Event-based vision data was captured using the Prophesee Onboard[4]. Similar to the tactile sensor, each camera pixel fires asynchronously and a positive (negative) spike is obtained when there is an increase (decrease) in luminosity. The Onboard was mounted on the arm and pointed towards the gripper to obtain information about the object of interest (Fig. 3). Although the camera has a maximum resolution of 640 x 480, we captured spikes from a cropped 200 x 250 rectangular window to minimize noise from irrelevant regions. The event camera bias parameters were tuned following recommended guidelines[5] and we use the same parameters throughout all experiments. During preliminary experiments, we found the Onboard to be sensitive to high frequency ($\geq$ 100Hz) luminosity changes; in other words, flickering light bulbs triggered undesirable spikes. To counter this effect, we used six Philips 12W LED White light bulbs mounted around the experiment setup to provide non-flickering lighting.

RGB Cameras. We used two Intel RealSense D435s to provide additional non-event image data[6]. The first camera was mounted on the end-effector with the camera pointed towards the gripper (providing a view of the grasped object), and the second camera was placed to provide a view of the scene. The RGB images were used for visualization and validation purposes, but not as input to our models; future work may look into the integration of these standard sensors to provide even better model performance.

OptiTrack. The OptiTrack motion capture system was used to collect object movement data for the slip detection experiment. We attached 6 reflective markers on the rigid parts of the end-effector and 14 markers on the object of interest. Eleven OptiTrack Prime 13 cameras were placed strategically around the experimental area to minimize tracking error (Fig. 3, right); each marker was visible to most if not all cameras at any instance, which resulted in continuous and reliable tracking. We used Motive Body v1.10.0 for marker tracking and manually annotated the detected markers. From initial tests, we found the OptiTrack to give reliable position estimates with error $\leq$ 1mm at 120Hz.

Further Details. In addition to the above sensors, we also collected proprioceptive data for the Panda arm and Robotiq gripper; these are not currently used in our models but can be included in future work. Additional information (including specific parameter settings, 3D-printed attachments, and multi-node data collection) is available in the online Supplementary Material[7].

VI. Container & Weight Classification

Our first experiment applies our event-driven perception framework — comprising NeuTouch, the Onboard camera, and the VT-SNN — to classify containers with varying amounts of liquid. Our primary goal was to determine if our multi-modal system was effective at detecting differences in objects that were difficult to isolate using a single sensor. Note that our objective was *not* to derive the best possible classifier; indeed, we did not include proprioceptive data which would likely have improved results [11], nor conduct an exhaustive (and computationally expensive) search for the best architecture. Rather, we sought to understand the potential benefits of using both visual and tactile spiking data using a reasonable setup.

A. Methods and Procedure

Objects. We used four different containers: an aluminium coffee can, a plastic Pepsi bottle, a cardboard soy milk carton and a metal tuna can (Fig. 5). These objects have different degrees of hardness; the soy milk container was the softest, and the tuna can was the most rigid. Because of size differences, each container was filled with differing amounts of liquid; the four objects contained a maximum of 250g, 400g, 300g, and 140g, respectively[8]. For each object, we collected data for $\{0\%, 25\%, 50\%, 75\%, 100\%\}$ of the respective maximum amount. This resulted in 20 object classes comprising the four containers with five different weight levels each.

Robot Motion. The robot would grasp and lift each object class fifteen times, yielding 15 samples per class. Trajectories for each part of the motion was computed using the MoveIt Cartesian Pose Controller [46]. Briefly, the robot gripper was initialized 10cm above each object's designated grasp point. The end-effector was then moved to the grasp position (2 seconds) and the gripper was closed using the Robotiq grasp controller with a force setting of 1 (4 seconds). The gripper then lifted the object by 5cm (2 seconds) and held it for 0.5 seconds.

Data Pre-processing. For both modalities, we selected data from the grasping, lifting and holding phases (corresponding to the 2.0s to 8.5s window in Figure 4), and set a bin duration of 0.02s (325 bins) and a binning threshold value $S_{\text{min}} = 2$. We used stratified K-folds to create 5 splits; each split contained 240 training and 60 test examples with equal class distribution.

Classification Models. We compared the SNNs against conventional deep learning, specifically Multi-layer Perceptrons (MLPs) with Gated Recurrent Units (GRUs) [47]. We trained each model using (i) the tactile data only, (ii) the visual data

[4]Details available at https://www.prophesee.ai.

[5]https://support.prophesee.ai/portal/kb/articles/bias-tuning

[6]The infrared emitters were disabled as they increased noise for the event camera and hence, no depth data was recorded.

[7]To be made available after review.

[8]The can did not have a cover and we filled it with a packet of rice to avoid spills and possible liquid damage. The tuna can was placed with the open side facing downwards so, the rice was not visible.

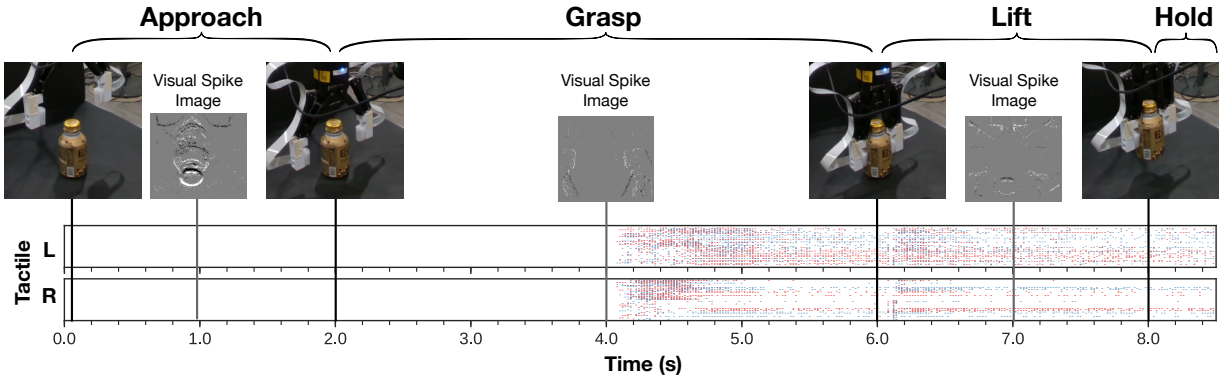

Fig. 4. Data collection procedure for the container & weight classification task. The robot end-effector starts at 10 cm above the grasp point. It then *approaches* the coffee can, stopping at the 2-second mark. The gripper then starts to close, *grasping* it at around the 4-second mark, resulting in tactile spikes. At the 6-second mark, the robot *lifts* it by 5cm above the table. The robot then holds it in the same position till the 8.5-second mark. For container and weight classification, we only use data starting from the 2-second mark.

Fig. 5. (**Left**) Containers used for container classification task: coffee can, plastic soda bottle, soy milk carton, and metal tuna can. (**Right**) The objects in our expanded dataset (36 object classes with various visual and tactile profiles) collected using the same protocol.

TABLE I
CONTAINER & WEIGHT CLASSIFICATION: AVERAGE ACCURACY WITH STANDARD DEVISION IN BRACKETS

Model	Tactile	Vision	Combined
SNN	0.677 (0.049)	0.660 (0.056)	0.840 (0.045)
ANN (MLP-GRU)	0.497 (0.065)	0.504 (0.084)	0.446 (0.084)

only, and (iii) the combined visual-tactile data. Note that the SNN model on the combined data corresponds to the VT-SNN. When training on a single modality, we use Visual or Tactile SNN as appropriate. We implemented all the models using PyTorch. The SNNs were trained with SLAYER to minimize spike count differences [29] and the ANNs were trained to minimize the cross-entropy loss using RMSProp. All models were trained for 500 epochs. Source code implementing our models is available online at http://[blindedforreview].

B. Results and Analysis

Model Comparisons. The test accuracy of the models are summarized in Table I. The multi-modal SNN model achieves the highest score of 84%, which is more than 15% better than the single-modality variants. Using either modality results in comparable performance. We were initially surprised by the accuracy attained by the vision-only model; we expected performance $\approx$ 20% since since the containers were easily distinguishable by sight, but the weight category was not. However, a closer examination of the data showed that (i) the Pepsi bottle was not fully opaque and the water level was observable by Onboard on some trials, and (ii) the Onboard was able to see object deformations as the gripper closed, which revealed the "fullness" of the softer containers.

Figure 6 gives an instructive example showing the advantage of fusing both modalities. It shows the output spikes from the different SNN models for a coffee can with 100% weight. The models trained on tactile and vision data are uncertain of the container and the weight category respectively. We see the tactile model is unable to discern between tuna can and coffee can. On the other hand, the vision model correctly predicts the container but is unsure about the weight category. The combined visual-tactile model incorporates information from both the modalities and is able to predict the correct class (both container and weight categories) with high certainty.

The SNN models performed far better than the ANN models, particularly for the combined visual-tactile data. We had expected comparable performance since MLP-GRU models are known to perform well on a variety of tasks. The poor performance was possibly due to the relatively long sample durations (350 time-steps) and the large number of parameters in the ANN models, relative to the size of our dataset.

Early Classification. Instead of waiting for all the output spikes to accumulate, we can perform early classification based on the number of spikes seen up to time t. Fig. 7 shows the accuracies of the different models over time. The combined visual-tactile model achieves the highest accuracies overall, but between $0.5 - 3.0$s, the vision model was already able to distinguish between certain objects. We posit this was due to small movements as the gripper closed, which resulted in changes perceived by the Onboard camera. As expected, tactile spikes do not emerge until contact is made with the object at $\approx$ 2s. Potentially, future work may look into event-based "attention" models that are able to actively choose (or weight) modalities at different time instances.

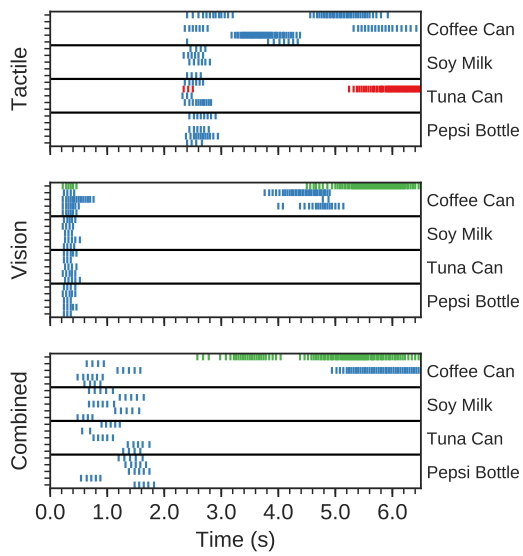

Fig. 6. Output spikes for the models trained with different modalities with correct and incorrect predictions in green and red, respectively. The weight categories are arranged from 0% to 100% (bottom to top) for each container. The tactile model is unable to distinguish between a coffee can and a tuna can while the vision model is uncertain about the weight. The combined visual-tactile model predicts the correct class with high certainty.

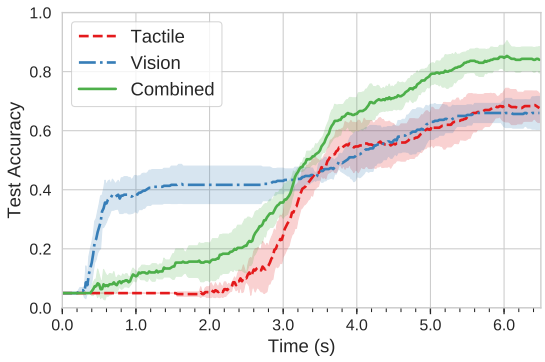

Fig. 7. Container and weight classification accuracy over time. Lines show average test accuracy and shaded regions represent the standard deviations. Vision-only classification results in higher early accuracy as visual spikes are obtained as the gripper is closing, and tactile events arise only upon contact with the object. Combining both vision and tactile event data via our VT-SNN results in significantly higher accuracy, compared to using each modality separately.

VII. Rotational Slip Classification

In this second experiment, we task our perception system to classify rotational slip, which is important for stable grasping; stable grasp points can be incorrectly predicted for objects with center-of-mass that are not easily determined by sight, e.g., a hammer or irregularly-shaped items. Accurate detection of rotational slip will allow the controller to re-grasp the object and remedy poor initial grasp locations. However, to be effective, slip detection needs to be performed accurately and rapidly.

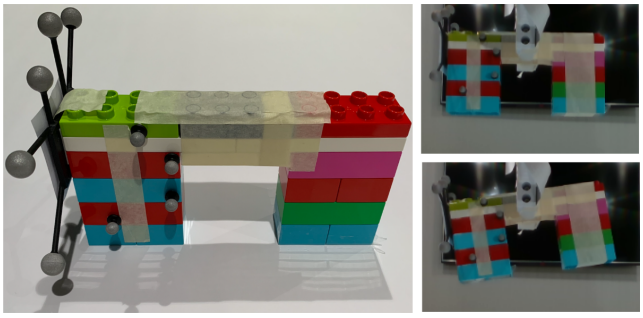

Fig. 8. (**Left**) Object for the Slip Classification Task with attached OptiTrack markers. (**Right**) Object during a stable grasp (top) and unstable grasp with rotational slip (bottom); the bottom object has additional mass transferred to the left leg (from the right leg) that causes it to rotate during lifting. Note that both objects had equal mass.

TABLE II
Slip Detection Accuracy

Model	Tactile	Vision	Combined
SNN	0.860 (0.042)	1.0 (0.0)	1.0 (0.0)
ANN (MLP-GRU)	0.790 (0.066)	1.0 (0.0)	1.0 (0.0)

A. Method and Procedure

Objects. Our test object was constructed using Lego Duplo blocks (Fig. 8) with a hidden 10g mass in each leg. The "control" object was designed to be balanced at the grasp point. To induce rotational slip, we modified the object by transferring the hidden mass from the left leg to the right. As such, the stable and unstable objects were visually identical and had the same overall weight.

Robot Motion. The robot would grasp and lift both object variants 50 times, yielding 50 samples per class. Similar to the previous experiment, motion trajectories were computed using the MoveIt Cartesian Pose Controller [46]. The robot was instructed to close upon the object, lift by 10cm off the table (in 0.75 seconds) and hold it for an additional 4.25 seconds. We tuned the gripper's grasping force to enable the object to be lifted, yet allow for rotational slip for the off-center object (Fig. 8, right).

Data Preprocessing. Instead of training our models across the entire movement period, we extracted a short time period in the lifting stage. The exact start time was obtained by analyzing the OptiTrack data; specifically, we obtained the baseline orientation distribution (for 1 second or 120 frames) and defined rotational slip as an orientation larger (or smaller) than 98% of the baseline frames lasting more than four consecutive OptiTrack frames. We found slip occurred almost immediately during the lifting. Since we were interested in rapid detection, we extracted a 0.15s window around the start of the lift, and set a bin duration of 0.001s (150 bins) with binning threshold $S_{\text{min}} = 1$. Again, we used stratified K-folds to obtain 5 splits, where each split contained 80 training examples and 20 testing examples.

Classification Models. The model setup and optimization

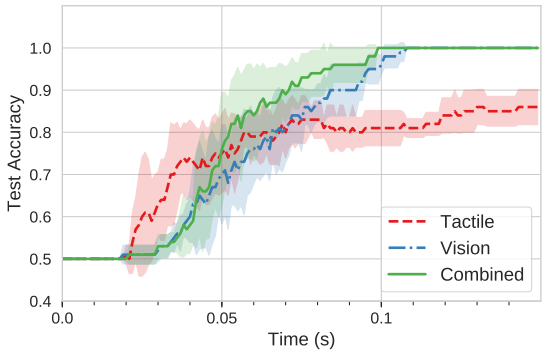

Fig. 9. Slip classification accuracy over time. Lines show average test accuracy and shaded regions represent the standard deviations. Classification using tactile only data results in higher early accuracy, but vision-only classification is more accurate as sensory data is accumulated. Combining both modalities results in higher accuracy, especially after 0.05s.

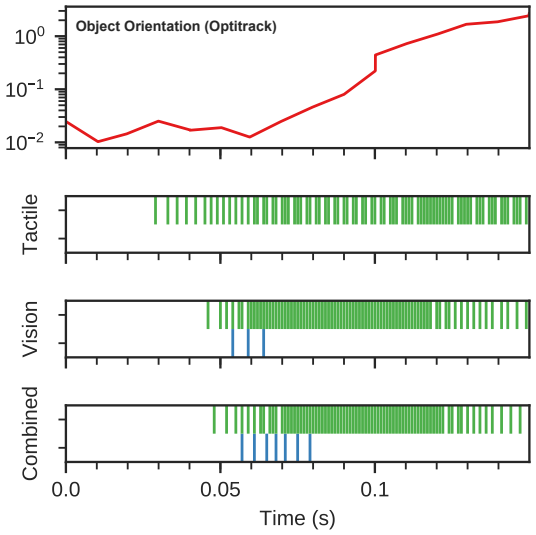

Fig. 10. Output spikes for the slip detection models trained with different modalities with correct prediction class in green. All models predict this sample correctly, but the tactile model spikes earlier compared to the other two models.

procedure are identical to the previous task, with 3 slight modifications. First, the output size is reduced to 2 for the binary labels. Second, the sequence length for the ANN GRUs were set to 150, the number of time bins. Third, the SNN's desired true and false spike counts were set to 80 and 5 respectively. Again, we compare SNN and ANN models using (i) the tactile data only, (ii) the visual data only, and (iii) the combined visual-tactile data.

B. Results and Analysis

Model Comparisons. The test accuracy of the models are summarized in in Table II. For both the SNN and ANN, both the vision and multi-modal models achieve 100% accuracy. This suggests that vision data is highly indicative of slippage, which is unsurprising as rotational slip would produce a visually distinctive signature. Using only tactile events, the SNN and ANN achieving 86% and 79% accuracy respectively; we hypothesise that the soft Ecoflex skin deforms in the same direction as the rotational slip, potentially causing the tactile sensors to generate a less distinctive signature. If so, it would interesting to examine alternative skin materials that can mitigate this issue.

Early Slip Detection. Similar to the previous analysis on early container classification, Fig. 9 summarizes slip test accuracy at different time points. The object starts being lifted at approximately 0.02s, and we see that by 0.1s, the multi-modal VT-SNN is able to classify slip perfectly. Again, we see vision and touch possess different accuracy profiles; tactile-only classification is more accurate early on (between $0.025 - 0.05$s), while vision-based classification is better after $\approx$ 0.6s. In Fig. 10, we see that the tactile model spikes earlier than the other models. Fusing both modalities results in an accuracy profile similar to vision but shifted towards higher accuracies. Again, performance may be improved by encouraging the multi-modal SNN to weigh the tactile-signals appropriately over time.

VIII. Conclusion

In this work, we propose an event-based perception framework that combines vision and touch to achieve better performance on two robot tasks. In contrast to conventional synchronous systems, our event-driven framework asynchronously processes discrete events and as such, can achieve high temporal resolution and low latency, with low power consumption.

We contributed NeuTouch, a novel neuromorphic event tactile sensor, and VT-SNN, a multi-modal spiking neural network that learns from raw unstructured event data. Experimental results on container & weight classification, and rotational slip detection show that combining both modalities is important for achieving high accuracies. The SNN models achieve better performance compared to similarly-structured ANNs, and are capable of early classification based on emitted spikes. These promising results suggest that the event-driven paradigm is a promising line of enquiry; we believe event-based sensing and learning will form essential parts of next-generation real-time autonomous robots that are power-efficient. We hope that the results in this paper and our datasets will encourage research in this area.

References

[1] A. Billard and D. Kragic, "Trends and challenges in robot manipulation," *Science*, vol. 364, no. 6446, p. eaat8414, 2019.

[2] D. Li, X. Chen, M. Becchi, and Z. Zong, "Evaluating the energy efficiency of deep convolutional neural networks on cpus and gpus," 10 2016, pp. 477–484.

[3] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in NLP," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, 2019, pp. 3645–3650. [Online]. Available: https://doi.org/10.18653/v1/p19-1355

[4] M. Pfeiffer and T. Pfeil, "Deep Learning With Spiking Neurons: Opportunities and Challenges," *Frontiers in Neuroscience*, vol. 12, no. October, 2018.

[5] S.-C. Liu, B. Rueckauer, E. Ceolini, A. Huber, and T. Delbruck, "Event-driven sensing for efficient perception: Vision and audition algorithms," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 29–37, 2019.

[6] Y. A. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[7] M. Davies, N. Srinivasa, T. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, Y. Liao, C. Lin, A. Lines, R. Liu, D. Mathaikutty, S. McCoy, A. Paul, J. Tse, G. Venkataramanan, Y. Weng, A. Wild, Y. Yang, and H. Wang, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, January 2018.

[8] J. Sinapov, C. Schenck, and A. Stoytchev, "Learning relational object categories using behavioral exploration and multimodal perception," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 5691–5698.

[9] 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)*. IEEE, 2016, pp. 536–543.

[10] J. Li, S. Dong, and E. Adelson, "Slip detection with combined tactile and visual information," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7772–7777.

[11] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8943–8950.

[12] J. Lin, R. Calandra, and S. Levine, "Learning to identify object instances by touch: Tactile recognition via multimodal matching," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3644–3650.

[13] H. Liu, F. Sun *et al.*, "Robotic tactile perception and understanding," 2018.

[14] P. Allen, "Surface descriptions from vision and touch," in *Proceedings. 1984 IEEE International Conference on Robotics and Automation*, vol. 1. IEEE, 1984, pp. 394–397.

[15] S. Luo, J. Bimbo, R. Dahiya, and H. Liu, "Robotic tactile perception of object properties: A review," *Mechatronics*, vol. 48, pp. 54–67, 2017.

[16] 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.

[17] J. Varley, D. Watkins, and P. Allen, "Visual-tactile geometric reasoning," in *RSS Workshop*, 2017.

[18] J. Reinecke, A. Dietrich, F. Schmidt, and M. Chalon, "Experimental comparison of slip detection strategies by tactile sensing with the biotac® on the dlr hand arm system," in *2014 IEEE international Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 2742–2748.

[19] Y. Bekiroglu, R. Detry, and D. Kragic, "Learning tactile characterizations of object-and pose-specific grasps," in *2011 IEEE/RSJ international conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 1554–1560.

[20] Z. Su, K. Hausman, Y. Chebotar, A. Molchanov, G. E. Loeb, G. S. Sukhatme, and S. Schaal, "Force estimation and slip detection/classification for grip control using a biomimetic tactile sensor," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2015, pp. 297–303.

[21] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.

[22] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine, "More than a feeling: Learning to grasp and regrasp using vision and touch," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3300–3307, 2018.

[23] 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.

[24] G. Gallego, T. Delbr, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, K. Daniilidis, D. Scaramuzza, S. Leutenegger, and A. Davison, "Event-based Vision : A Survey," Tech. Rep., 2018.

[25] A. Mitrokhin, C. Ye, C. Fermuller, Y. Aloimonos, and T. Delbruck, "EV-IMO: Motion Segmentation Dataset and Learning Pipeline for Event Cameras," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[26] A. Z. Zhu and L. Yuan, "EV-FlowNet: Self-Supervised Optical Flow Estimation for Event-based Cameras," in *Robotics: Science and Systems*, 2018.

[27] A. I. Maqueda, A. Loquercio, G. Gallego, N. Garc\'\ia, and D. Scaramuzza, "Event-based vision meets deep learning on steering prediction for self-driving cars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5419–5427.

[28] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, "Deep learning in spiking neural networks," *Neural Networks*, vol. 111, pp. 47–63, 2019. [Online]. Available: https://doi.org/10.1016/j.neunet.2018.12.002

[29] S. B. Shrestha and G. Orchard, "Slayer: Spike layer error reassignment in time," in *Advances in Neural Information Processing Systems*, 2018, pp. 1412–1421.

[30] G. Bellec, F. Scherr, E. Hajek, D. Salaj, R. Legenstein, and W. Maass, "Biologically inspired alternatives to backpropagation through time for learning in recurrent neural nets," *arXiv preprint arXiv:1901.09049*, 2019.

[31] M. Akrout, C. Wilson, P. Humphreys, T. Lillicrap, and D. B. Tweed, "Deep learning without weight transport," in *Advances in Neural Information Processing Systems*, 2019, pp. 974–982.

[32] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014. [Online]. Available: https://science.sciencemag.org/content/345/6197/668

[33] S. Chevallier, H. Paugam-Moisy, and F. Lemaître, "Distributed processing for modelling real-time multimodal perception in a virtual robot." in *Parallel and Distributed Computing and Networks*, 2005, pp. 393–398.

[34] N. Rathi and K. Roy, "Stdp-based unsupervised multimodal learning with cross-modal processing in spiking neural network," *IEEE Transactions on Emerging Topics in Computational Intelligence*, pp. 1–11, 2018.

[35] E. Mansouri-Benssassi and J. Ye, "Speech emotion recognition with early visual cross-modal enhancement using spiking neural networks," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[36] T. Zhou and J. P. Wachs, "Spiking neural networks for early prediction in human-robot collaboration," *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1619–1643, 2019. [Online]. Available: https://doi.org/10.1177/0278364919872252

[37] J. Konstantinova, A. Jiang, K. Althoefer, P. Dasgupta, and T. Nanayakkara, "Implementation of tactile sensing for palpation in robot-assisted minimally invasive surgery: A review," *IEEE Sensors Journal*, vol. 14, no. 8, pp. 2490–2501, 2014.

[38] Y. Wu, Y. Liu, Y. Zhou, Q. Man, C. Hu, W. Asghar, F. Li, Z. Yu, J. Shang, G. Liu *et al.*, "A skin-inspired tactile sensor for smart prosthetics," *Science Robotics*, vol. 3, no. 22, p. eaat0429, 2018.

[39] Q.-J. Sun, X.-H. Zhao, Y. Zhou, C.-C. Yeung, W. Wu, S. Venkatesh, Z.-X. Xu, J. J. Wylie, W.-J. Li, and V. A. Roy, "Fingertip-skin-inspired highly sensitive and multifunctional sensor with hierarchically structured conductive graphite/polydimethylsiloxane foams," *Advanced Functional Materials*, vol. 29, no. 18, p. 1808829, 2019.

[40] J. He, P. Xiao, W. Lu, J. Shi, L. Zhang, Y. Liang, C. Pan, S.-W. Kuo, and T. Chen, "A universal high accuracy wearable pulse monitoring system via high sensitivity and large linearity graphene pressure sensor," *Nano Energy*, vol. 59, pp. 422–433, 2019.

[41] T. Callier, A. K. Suresh, and S. J. Bensmaia, "Neural coding of contact events in somatosensory cortex," *Cerebral Cortex*, vol. 29, no. 11, pp. 4613–4627, 2019.

[42] W. W. Lee, Y. J. Tan, H. Yao, S. Li, H. H. See, M. Hon, K. A. Ng, B. Xiong, J. S. Ho, and B. C. Tee, "A neuro-inspired artificial peripheral nervous system for scalable electronic skins," *Science Robotics*, vol. 4, no. 32, p. eaax2198, 2019.

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

[44] W. Gerstner, "Time structure of the activity in neural network models," *Physical review E*, vol. 51, no. 1, p. 738, 1995.

[45] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: Using the yale-cmu-berkeley object and model set," *IEEE Robotics Automation Magazine*, vol. 22, no. 3, pp. 36–52, Sep. 2015.

[46] D. Coleman, I. Sucan, S. Chitta, and N. Correll, "Reducing the barrier to entry of complex robotic software: a moveit! case study," *arXiv preprint arXiv:1404.3785*, 2014.

[47] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734.