2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Uncertainty Aware Texture Classification and Mapping Using Soft Tactile Sensors

Alexander Amini[1,*], Jeffrey I. Lipton[1,2,*], Daniela Rus[1]

***Abstract*— Spatial mapping of surface roughness is a critical enabling technology for automating adaptive sanding operations. We leverage GelSight sensors to convert the problem of surface roughness measurement into a vision classification problem. By combining GelSight sensors with Optitrack positioning systems we attempt to develop an accurate spatial mapping of surface roughness that can compare to human touch, the current state of the art for large scale manufacturing. To perform the classification, we propose the use of Bayesian neural networks in conjunction with uncertainty-aware prediction. We compare the sensor and network with a human baseline for both absolute and relative texture classification. To establish a baseline, we collected performance data from humans on their ability to classify materials into 60, 120, and 180 grit sanded pine boards. Our results showed that the probabilistic network performs at the level of human touch for absolute and relative classifications. Using the Bayesian approach enables establishing a confidence bound on our prediction. We were able to integrate the sensor with Optitrack to provide a spatial map of sanding grit applied to pine boards. From this result, we can conclude that GelSight with Bayesian neural networks can learn accurate representations for sanding, and could be a significant enabling technology for closed loop robotic sanding operations.**

I. Introduction

We believe that high resolution soft robotic touch sensors are the key to enabling robots to offload the dull, dirty, and dangerous job of sanding from humans. Sanding is of critical importance to manufacturing and maintenance in the aerospace industry, automotive industry, and carpentry. Components may need to be sanded after forming, before painting, or for buffing. Sanding is not a safe task for humans. The process generates particulate matter which is dangerous to breathe [1], the tools generate vibrations that can cause joint and nerve damage [2], and the work often places people in uncomfortable positions that can lead to falls [3]. Attempts to eliminate human workers from sanding are often constrained by the lack of feedback on surface roughness. It is difficult to know where to sand an object just by looking at a surface. This is especially true when copies of parts can have variations, when doing repairs, or in low-volume production environments. Therefore, humans perform this step because their sense of touch is highly developed, enabling them to find small regions in need of further processing.

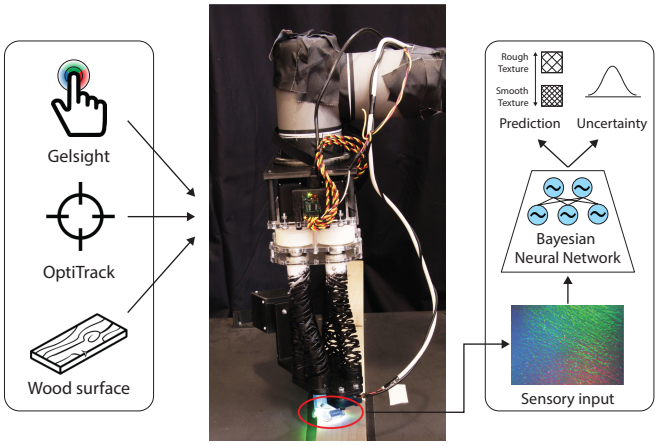

Fig. 1. **Large area texture prediction.** Our system leverages Gelsight optical touch sensing to learn complex representations of surface texture. By additionally modelling epistemic uncertainty, our algorithm can provide feedback to inform automated robotic sanding pipelines.

Current surface roughness feedback systems in manufacturing rely on fixed infrastructure or spot inspections. Fixed sensors can work for many factory inspections, but are too limiting for some products. Airplane components, for instance, would be too large to fit inside an array for stationary laser profilometers. Other structures are manufactured and finished on sight. Sanding flooring for instance would require a deployed scanning system for inspection. Finally, other instances, such as refining a chair, require getting into tight spaces. For all of these applications for automated sanding, the ability to spatially map surface roughness would be a key enabling technology. Such a technology would need to be lightweight, fast, and integrated with positioning data to successfully close the perception-control feedback loop.

To solve the perception problem of adaptive sanding systems, we propose a perceptual system consisting of Gelsight sensory inputs, a convolutional neural network (CNN) for learning representations of surface roughness, and a soft robotic actuator for palpating the Gelsight onto surfaces. We developed and trained two Bayesian CNN models for classifying and comparing surface roughness of sanded wood based on Gelsight images, and compared their performance to human touch through a user study. Furthermore, we deal with large amounts of noise in real-world soft robotic perception scenarios by formulating a novel uncertainty-aware inference algorithm. We demonstrate that our system

* These authors contributed equally to this work.

Support for this work was given by the National Science Foundation (NSF) under grant numbers 1830901 and 1644558 and Toyota Research Institute (TRI).

[1]Computer Science and Artificial Intelligence Lab, MIT, 32 Vassar St, Cambridge, MA 02139, USA

[2] Mechanical Engineering Department, University of Washington, 3900 E Stevens Way NE, Seattle, WA 98195, USA

can classify grit (i.e., roughness) with 82% accuracy and relative change in grit with 80% accuracy, outperforming both human vision and human touch baselines. We integrated the Gelsight+CNN system with handed shearing auxetic actuators [4] and mounted it onto a UR5 arm. This yielded an entirely electric soft robotic system which can scan large surfaces, enable adaptive sanding pipelines, and easily be deployed on mobile robots.

The contributions of this paper can be summarized as follows:

1) Design of a CNN based classifier for the absolute and relative surface roughness of sanded wood based on tactile Gelsight images;
2) Formulation of a novel uncertainty-aware prediction algorithm to leverage Bayesian uncertainty estimates of our model for inference;
3) Evaluation of our algorithms against the performance of humans on comparable tactile classification tasks;
4) Integration of our system on a robotic pipeline for capturing Gelsight scans over large areas.

II. Related Work

A. Surface roughness sensing

A variety of surface roughness sensors exist for traditional robotic manufacturing processes [5]. One of the most popular surface roughness scanners is a portable surface roughness sensor, wherein a light and delicate contact probe along a line of a surface [6]. Using this to measure roughness over a large area would be time-consuming. Laser profilometers can be uses to measure the height variations of a surface, but require a stationary and stable reference location for mounting [7]. 3D structured light systems also require a stationary and stable mounting location for accurate sensing [8]. More recently there has been interest in eliminating the need for a stationary base to enable the scanning of large surfaces of concrete with drones [9]. While this ultrasonic technique is accurate enough for determining wear on canals and infrastructure, it cannot provide enough detail for manufacturing work and still require a stationary base while scanning only along a line, similar to a traditional profilometer.

B. Soft robotic touch

Many current soft touch sensors, such as Ionogel, focus on highly detailed force and pressure measurements. Others, like BioTac, are used to identify many characteristics of the objects they touch (i.e. estimating friction coefficients or identifying textures) [10], [11]. Ionogels, in particular, are useful for implementing more robust grasp algorithms via the deformation these sensors perceive when they come in contact with an object [12]. Despite the high accuracy of these pressure and force sensors, they cannot give further information about the roughness and texture of the objects they grasp or touch.

The GelSight sensor, by contrast is an optical touch sensor that incorporates a reflective membrane covering a deformable silicone rubber piece with an embedded camera and optical system [13], [14]. The camera is able to view the external contact geometry and textures of an object via the deformation of the outer rubber piece. Because the GelSight sensor is an optical sensor, methods from computer vision can be applied to tactile problems. Previous work involving GelSight sensors include using the sensor to estimate the hardness of various objects [15] and to differentiate between various types of clothing textures [16].

C. Uncertainty aware learning

Reliable estimation of uncertainty of any automated system is crucial for real-world deployement, especially in safely critical scenarios. Uncertainty estimation of deep learning algorithms is a heavily studied field of research, both in terms of a prediction's aleatoric (statistical) uncertainty as well as the epistemic (inference) uncertainty. While aleatoric uncertainty can be modeled by learning to output the parameters of a probability distribution [17]–[21], epistemic uncertainty describes the uncertainty of the model's prediction [22].

Bayesian deep learning [23] aims to explicitly model epistemic uncertainty by placing probability distributions over every weight in the network instead of treating them as deterministic scalars [22]. Tractably training such networks generally relies on approximations of these distributions through sampling [24], [25], ensembling [26], [27], or probabilistic backpropagation [28], [29]. Such approaches have been successfully applied to tasks in computer vision [22], [25], sequential modeling [30], and guided reinforcement learning [31]. In this work, we leverage dropout sampling uncertainty estimation in the context of touch-based sensing to fuse predictions with uncertainty and achieve superior predictive performance, but prefer more direct, non-sampling based approaches to uncertainty estimation such as evidential deep learning [32], [33] for future work.

III. Methodology

A. Problem formulation

Adaptive robotic sanding roughly consists of two parts: (1) a robotic sander which performs the sanding operation and (2) a perception system that classifies the roughness of areas on the board. The perception system perceives the board and provides feedback to the robot sander to inform where and how much to continue sanding. This process continues in a closed loop to enable an adaptive sanding solution. In this paper, we focus on engineering the perception system using a machine learning algorithm trained directly on images from soft tactile sensors.

In the following subsection, we outline the problem formulation of this machine learning perception problem. Given a paired dataset of observations, $\boldsymbol{X}$, from the Gelsight touch based sensor, and property labels of the surface, $\boldsymbol{Y}$, we aim to learn a functional mapping, f, parameterized by weights, $\boldsymbol{W}$ such that $\boldsymbol{Y} \approx f(\boldsymbol{X}; \boldsymbol{W})$. More concretely, we aim to solve the following minimization problem between the true labels and our inferred predictions,

$$\min_{\boldsymbol{W}} \mathcal{L}\Big(\boldsymbol{Y}, f(\boldsymbol{X}; \boldsymbol{W})\Big). \tag{1}$$

where $\mathcal{L}(\boldsymbol{Y}, \hat{\boldsymbol{Y}})$ is the objective or cost function that we will optimize. In this work, we explore two types of classification problems:

1) **Absolute classification**. Given a single Gelsight scan as input, the network is trained to output a discrete probability distribution over the possible standing grit levels. We consider three evenly spaced grit levels: 60, 120, 180 (Fig. 2A).
2) **Relative classification**. Given two Gelsight scans as input, the network is trained to output the relative difference in grit levels. Again we consider a discrete output space, where the grit in one of the scans can either be equal to, less than, or greater than the grit in the other scan.

B. Data collection

Our dataset consisted of Gelsight scans collected from individual boards of pine wood. Each board was sanded individually to one of the three grit levels using varying levels of sand paper grit (60, 120, or 180 grit). Separate boards were used for training, validation, and test sets. Gelsight scans were collected manually by a human operator (for training) or a soft robotic manipulation system (for testing). In total, our system consisted of a motion-tracked Gelsight sensor attached to a soft robotic end effector on a Universal Robotics (UR) 5 robotic arm. We leverage handed shearing auxetics (HSAs) [4], [34] with servo control as our soft robotic end effector for additional compliance of the tactile sensor as it comes in contact with the surface. While spatial localization information was not used for training, we integrate our system within a motion capture environment for providing test-time feedback during testing. Please refer to Sec. V-B for additional details on spatial testing.

For training, each of the collected scan images was also augmented in order to curate our dataset (Fig. 2B). First, to avoid losing spatial resolution on larger input scans, every scan is split into four quadrant patches and saved separately. Next, each of the patches is randomly rotated and reflected about the coordinate axes to allow the network to learn rotational symmetry. Finally, the scans are converted into HSV colorspace such that the hue can be randomly perturbed before converting back to the original RGB space. Since the hue perturbation amount is a continuous random variable, we perform this augmentation online during training, in contrast to patching and rotating, which are done before training. The resulting dataset consists of over 3,000 unique Gelsight images from the three grit categories. Additional data was collected (and augmented) for a validation set to evaluate training progress.

C. Human user study

To establish a baseline comparison to our automated system, we evaluate human performance on both of the desired tasks (absolute and relative texture classification). We evaluate the performance of humans on completing these tasks using either (1) images of the Gelsight scans (as in the proposed system) or (2) their own sense of human touch. While we gauge (1) as a fair baseline to our system, we also consider (2) as a "gold-standard" baseline to compare our system against the highly evolved human sense of touch.

For each of the tests, participants were provided a brief training period until they deemed that they were sufficiently comfortable with solving the task. The testing phase consisted of 20 randomly selected test samples for each of the tasks; no supervisory feedback was given to the participants until the completion of all tests. A total of 10 participants were chosen for this study.

IV. Learning Algorithm

A. Uncertainty estimation

Traditionally, neural networks are trained with deterministic, scalar weights. In contrast, Bayesian deep neural networks place probability distributions over every weight in the network [24], thus making the network stochastic and capable of capturing its uncertainty. These approaches aim to learn the full posterior, $P(\boldsymbol{W}|\boldsymbol{X}, \boldsymbol{Y})$. In practice, it is infeasible to directly compute this posterior from vast observational data and thus many sampling based techniques have emerged as accurate and computationally efficient ways of estimating this posterior [24], [26].

Dropout sampling [35] is frequently performed during training of deep neural networks to avoid overfitting by placing independent identically distributed (i.i.d.) Bernoulli random variables over every neuron to either "drop" or "keep" it on that iteration. Beyond the training regime, it has also been shown that dropout sampling also approximates a probabilistic deep Gaussian process [24]. Using a dropout-based approximation of the posterior, $q(\boldsymbol{W})$, we get a predictive distribution,

$$q(\boldsymbol{Y}|\boldsymbol{X}) = \int P(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{W}) q(\boldsymbol{W})\, d\boldsymbol{W}. \tag{2}$$

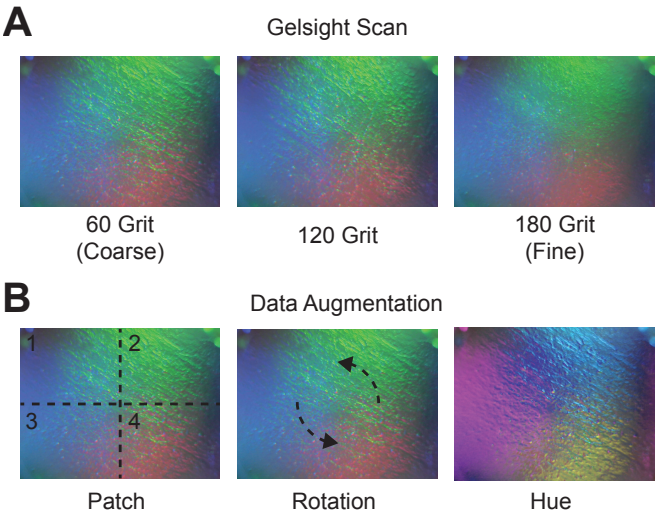

Fig. 2. **Dataset and augmentation**. Random sample scans (A) on three varying surface materials ranging from coarse (left) to fine (right). To augment the training dataset, each scan also underwent train-time augmentation to encourage translational, rotational, and hue invariance.

Given K stochastic forward passes through the network using dropout, $\{\boldsymbol{W}_i\}_{t=1}^{K}$, we define the final prediction, $\hat{\boldsymbol{Y}}$, and estimated uncertainty, $\hat{\boldsymbol{\sigma}}^2$, of the network as,

$$\hat{\boldsymbol{Y}} = E[\boldsymbol{Y}|\boldsymbol{X}] = \frac{1}{K}\sum_{t=1}^{K} f(\boldsymbol{X};\boldsymbol{W}_t) \tag{3}$$

$$\hat{\boldsymbol{\sigma}}^2 = Var[\boldsymbol{Y}|\boldsymbol{X}] = \frac{1}{K}\sum_{t=1}^{K} f(\boldsymbol{X};\boldsymbol{W}_t)^2 - E[\boldsymbol{Y}|\boldsymbol{X}]^2. \tag{4}$$

Using this formulation of estimating predictive uncertainty, we propose two uncertainty-aware models designed to learn to solve our absolute and relative texture classification tasks.

B. Models and training

Since Gelsight texture scans are efficiently stored as 2D RGB images, our models utilize convolutional layers to learn 2D visual features [36], [37]. The underlying backbone of both models can be roughly split up into two parts (Fig. 3A): a convolutional feature extractor and a learned predictor. The feature extractor takes as input a single Gelsight scan and returns a learned feature vector representation of that scan, while the predictor takes a feature vector as input and outputs a discrete probability distribution over the possible task classes using a final softmax activation layer,

$$\hat{\boldsymbol{Y}}_i = \frac{\exp(\boldsymbol{a}_i)}{\sum_j \exp(\boldsymbol{a}_j)}, \qquad i = \{1,2,3\} \tag{5}$$

where $\boldsymbol{a}$ is the associated activation vector of logits before applying the softmax operator. Finally, we train the network end-to-end using a cross entropy loss function between the predicted ($\hat{\boldsymbol{Y}}_i$) and target ($\boldsymbol{Y}$) distributions (both with 3 discrete classes),

$$\mathcal{L}(\boldsymbol{Y}, \hat{\boldsymbol{Y}}) = -\sum_{i=1}^{3} \boldsymbol{Y}_i \log(\hat{\boldsymbol{Y}}_i). \tag{6}$$

Fig. 3A, provides additional details on the individual layers and hyperparameters of each layer in both of these modules. Additionally, as discussed in Sec. IV-A, dropout ($p = 0.7$) is also used throughout all parts of the model to allow for test-time uncertainty estimation.

In absolute texture classification (Fig. 3B), the end-to-end model feeds the learned features from a single scan input into the predictor module, which is trained to predict the absolute grit of the texture (60, 120, 180). In relative classification, we propose the use of a dual input head, where features are learned from both scans simultaneously and then combined into a single feature vector before entering the predictor module. Since we want to enforce consistency in the learned features across both of the two feature extractors, we explicitly share the weights during training and testing between these two heads.

All models are implemented in TensorFlow [38], trained using the Adam optimizer [39] with learning rate 10^{-4} and batch size of 64, and terminated using early-stopping based on the held-out validation set. Models were trained independently five times for statistically significant results.

C. Uncertainty integration

Using our trained model and predictive uncertainty estimation techniques described in the previous two subsections, we can fuse uncertainty estimates with our predictions to enable uncertainty-aware inference. This is extremely important in the context of soft robotics. Since the primary sensor was controlled using a soft end-effector for extra compliance, we compromise accurate manipulation in space. Thus, during deployment, such systems are subject to encountering noisy scans, due to the sensor not being fully pressed down, part of the sensor being off the region of interest, etc. In this subsection, we outline an algorithm for obtaining reliable predictions on potentially noisy tactile scans by combining the predictions over sub-parts of the scan based on the uncertainty of each sub-part.

Given a single scan, X, we split the image into S sub-patches, obtained by sliding a 64×64 sliding window over the scan. In practice, we found that using $S = 25$ (i.e., 5 sub-patches in both the x-y dimensions, yielding 25 sub-patches), provided a desirable trade-off between spatial overlap with memory storage. Using $S > 25$ provided additional spatial resolution at the cost of memory, but yielded similar results as $S = 25$. We denote the collection of S sub-patches as $\boldsymbol{x}$. Next, we run K stochastic inference passes of all sub-patches, $\boldsymbol{x}$, through our Bayesian network to generate predictions, $\hat{\boldsymbol{y}}$, and uncertainties, $\hat{\boldsymbol{\sigma}}^2$ (Eq. 3, 4). At this point, one could manually define an uncertainty threshold at which sub-patch predictions should be discarded; however, this requires manually tuning a viable threshold to your dataset. Furthermore, an identified threshold on one dataset may not be valid on a slightly different dataset. Therefore, we propose a more flexible approach to integrate all sub-patch predictions together taking into account their associated uncertainty. This is done by representing the final prediction $\hat{Y}$ as the weighted average of all sub-patch predictions, weighted by their inverse-uncertainty $\hat{\boldsymbol{\tau}}$,

$$\hat{Y} = \frac{\sum_{s=1}^{S} \hat{\boldsymbol{y}}_s \cdot \hat{\boldsymbol{\tau}}_s}{\sum_{s=1}^{S} \hat{\boldsymbol{\tau}}_s} \tag{7}$$

where $\hat{\boldsymbol{\tau}} = \hat{\boldsymbol{\sigma}}^{-2}$ is the inverse-uncertainty (also known as precision). Thus, predictions with lower uncertainty (higher precision) will be heavily weighted in the final prediction, while high uncertainty patches will not contribute significantly. Alg. 1 outlines this algorithm in pseudo-code for the absolute texture classification problem. For relative classification, we adopt a similar algorithm where one of the inputs is patched as described above while the second input is kept constant and treated as a baseline scan that we want to compare against. In future sections, we will discuss the effect of the number of dropout samples and number of baseline candidate scans on both of these algorithms.

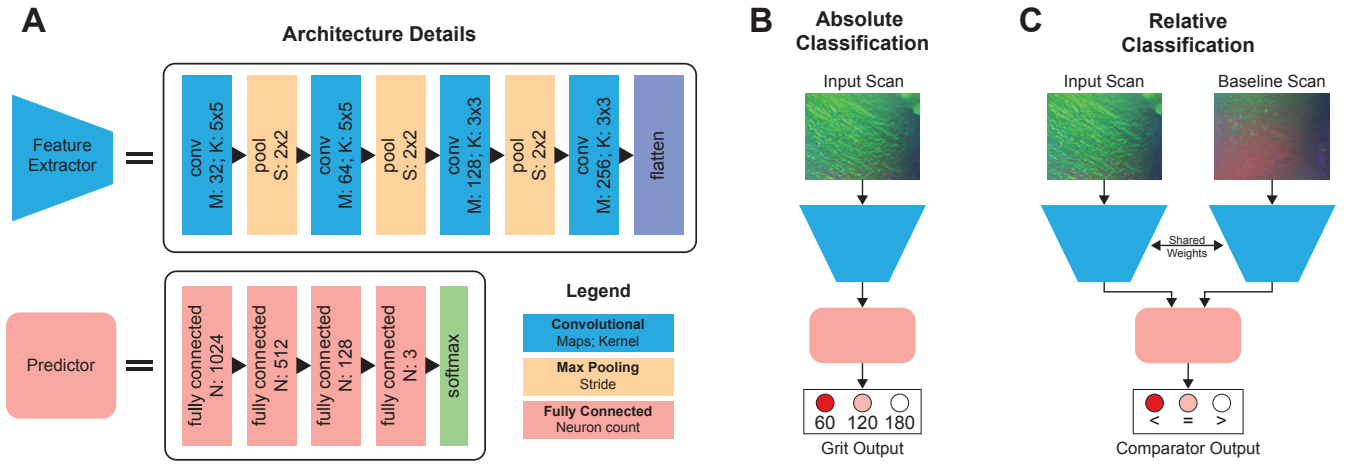

Fig. 3. **Model architecture**. (A) Architecture and layer details of the feature extractor and predictor modules. End-to-end pipelines for both absolute (B) and relative (C) texture classification. In both cases, the model is trained to output a discrete probability distribution for that task.

Algorithm 1 Uncertainty aware prediction from tactile sensing

$\boldsymbol{x} \leftarrow \{X_s\}_{s=1}^{S}$ ▷ Split input into sub-patches
for $t = 1 \dots K$ **do**
  $\boldsymbol{W}_t \leftarrow Dropout(p)$ ▷ Stochastic sampling
  $\hat{\boldsymbol{y}}_t = f(\boldsymbol{x}; \boldsymbol{W}_t)$
end for
$\hat{\boldsymbol{y}} = \frac{1}{K}\sum_t \hat{\boldsymbol{y}}_t$ ▷ Predictive mean. Eq 3.
$\hat{\boldsymbol{\sigma}}^2 = \frac{1}{K}\sum_t \left(\hat{\boldsymbol{y}}_t{}^2 - \hat{\boldsymbol{y}}^2\right)$ ▷ Uncertainty. Eq 4.
$\hat{\boldsymbol{\tau}} = \hat{\boldsymbol{\sigma}}^{-2}$

$\boldsymbol{\omega} = \hat{\boldsymbol{\tau}} / \sum_s \hat{\boldsymbol{\tau}}_s$ ▷ Standardized precision weights
$\hat{Y} = \sum_s \hat{\boldsymbol{y}}_s \cdot \boldsymbol{\omega}_s$ ▷ Precision weighted mean. Eq 7.
return $\hat{Y}$

D. Spatially informed prediction

Our inference pipeline is also fully integrated with high precision motion tracking using OptiTrack localization sensors. Thus, we demonstrate that while our uncertainty-aware prediction system can be used to classify textures on the surface, our localization setup enables mapping those predictions in coordinate space according to the pose of the end effector.

As discussed in Sec. III-B, our system consists of a motion tracked Gelsight sensor attached to a soft robotic end effector on a UR5 robotic arm. Full integration of the sensing system with localization information allows for sensing predictions to directly inform robotic sanding systems where to place attention. This integration allows the sensing system to be directly placed within a larger perception-planning-control loop for automated robotic sanding.

V. Results

A. Learning evaluation

Our models, both for absolute and relative texture classification, are trained on a dataset of tactile scans of varying textures (Sec. III-B). In addition to our training set, a validation set of scans from different boards are used to monitor the progress of training on unseen data. This also allows us to apply early stopping to optimization and avoid overfitting on our training data. Fig. 4 illustrates the evolution of our training algorithm on both the absolute (A) and relative (B) texture classification task. Results show averages over 5 trials (blue line), with error bars (cyan bands) depicting standard deviation. Note that these curves illustrate the validation set accuracy on unseen Gelsight scans, not on previously seen training data. For additional comparison, we also plot horizontal lines for the accuracy of the human on both tasks using either tactile (purple) or visual Gelsight (orange) sensing.

Our models exhibit significant increases in performance over the human baselines using the equivalent Gelsight sensing for both absolute (+39%) and relative (+30%). In addition, we discovered that our methods also outperformed human touch classification again in both absolute (+13%) and relative (+12%). We acknowledge that human touch performance is heavily dependent on training and that it is very possible that individuals trained to perform this task every day will be potentially more competitive when compared to our algorithm.

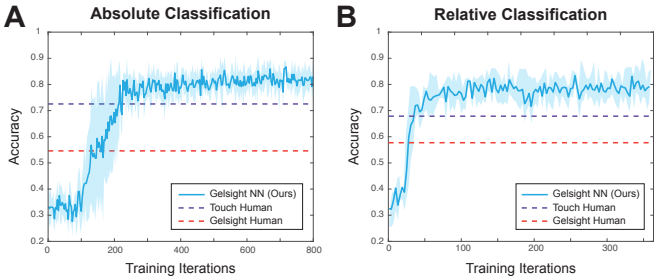

Fig. 4. **Loss evolution results**. Comparison of test set accuracies on learned absolute (A) and relative (B) models (cyan) against human touch (purple) and human visual sensing through Gelsight (orange). One standard deviation error bands are also visualized for statistical significance.

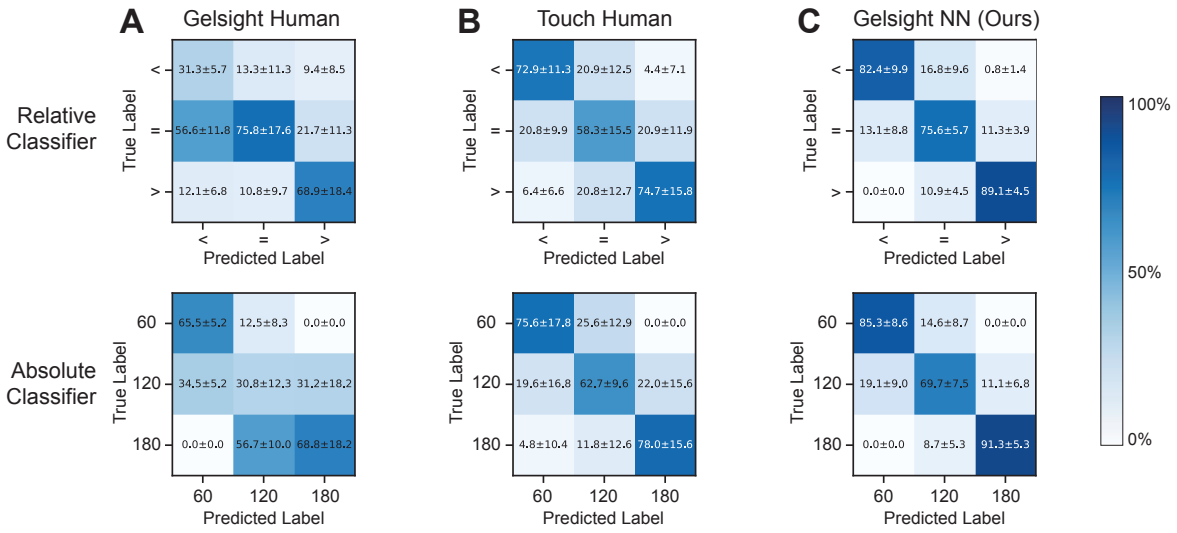

Fig. 5. **Class confusion matrices**. (C) Confusion matrices for class-wise accuracies of each models and human studies is also provided for both learning tasks. Color intensities depict the mean accuracy, with one standard deviation also presented in each cell.

Fig. 5 shows confusion matrices for each of the three methods on both tasks as a means to depict class-specific accuracies. Columns in an individual confusion matrix correspond to a predicted label while rows represent the true label. A perfect classifier will have a 100% along the diagonal (i.e., all predictions are correct), and 0% on all elements off the diagonal (i.e., no missed predictions).

Several interesting results emerge from these confusion matrices. First, the accuracy difference between human sensing with Gelsight (A) and human touch (B) is stark. Human classification using Gelsight suffers significantly with predicting the 120 (middle) class in the absolute task, with predictions roughly random (approx. $1/3$ correct). Conversely, on the relative task, it was easiest to classify equivalent textures but harder to distinguish which grit was greater.

When observing the results from human touch (B) and our proposed system (C) classification, we see that both absolute and relative classification have strong performance on the extreme class cases (60, 180 and "<", ">"), but struggle on the central cases (120 and "="). A desirable property of human touch classification was that while the main diagonal was the strongest, the two off-diagonals were consistently next highest in accuracy. This means that if the human (or model) did not predict the correct answer, they were rarely no more than one class off in their prediction. This is an important property that is also shared with our NN results. Overall, our method achieved average accuracies of over 82% on both classification tasks while outperforming human touch on every one of the individual class predictions.

B. Online inference and uncertainty estimation

Beyond offline evaluations on a cleaned validation set, we are also interested in how the system performs in a test setting, where poor Gelsight scans are not filtered out beforehand, and the system has to reason about its own uncertainty when determining a final prediction. To accomplish this, we measure accuracy of our system on an unseen "spatial" test set where the ground truth label is known based on the location of the end effector at the moment the scan is collected. In Sec. IV-C, we outlined the algorithm (Alg. 1 for fusing the predictions of the network on sub-parts of an input scan based on the network's uncertainty of each part. In this section, we explore the performance of this uncertainty-aware prediction algorithm and empirically further motivate its use over a classical deterministic neural network solution.

1) Effect of increasing samples: A central aspect of Bayesian deep learning is that model uncertainty can be estimated by conducting stochastic forward passes through the network (thus, obtaining samples of the weight distributions) and measuring the variance of the output. Since we leverage dropout to obtain these stochastic samples, a deterministic NN baseline can be easily achieved by using a dropout probability of 0 (no dropout) at inference time. In this case, only a single sample is needed since every pass through the network yields the same output (at the consequence of lacking uncertainty estimates).

Figure 6 illustrates the effect of greater numbers of dropout samples on the accuracy of our system. As the number

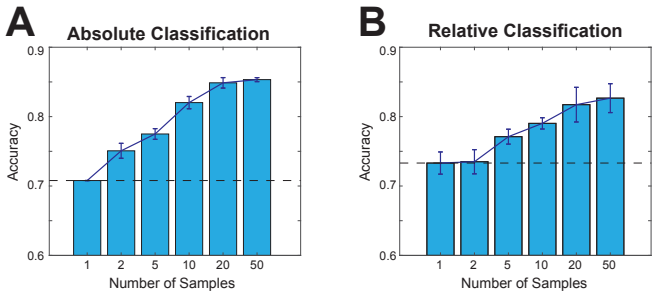

Fig. 6. **Effect of number of uncertainty samples on accuracy**. For both absolute (A) and relative (B) texture classification, we evaluate the utility of increased stochastic sampling. We also compare against performance of a deterministic neural network with no measure of uncertainty (illustrated with dotted-black line).

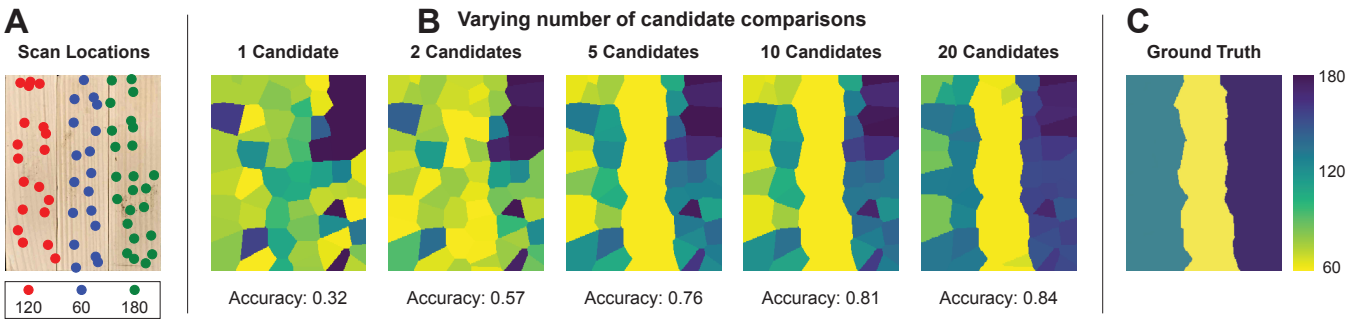

Fig. 7. **Spatially aware touch sensing**. By integrating our system into a full motion capture environment, we enable high precision inference to be provided back as spatial feedback. (A) We collect raw Gelsight scans on three different boards of varying texture. With our relative classification network we predict the relative texture of each scan with respect to a random labeled candidate from the database. Based on the results from the relative classifier, an absolute grit is inferred. We demonstrate that using greater numbers of baseline candidates significantly improves performance (B) in comparison to the ground truth surface grits (C).

of sampling passes increases, the accuracy of the uncertainty estimate also increases, which in turn improves the accuracy of the uncertainty-aware prediction algorithm. We demonstrate that for both absolute and relative classification the uncertainty-aware prediction with sufficient number of samples (greater than five) outperforms the deterministic, non-uncertainty-aware prediction.

2) Spatial testing: Combining the sensing system with a motion capture environment (Sec. III-B) allowed us to evaluate how the network's predictions fall spatially on a textured area in 3D space. Since every scan is captured along with the 6DOF pose of the sensor, we can map the model predictions spatially over large areas. Furthermore, since in our test set, we have ground truth pose to texture classification information, we are able to further evaluate our models in the context of this spatial mapping problem.

Fig. 7A illustrates the test boards and spatial positions of Gelsight scans using our system. The color of each point represents the ground truth grit at that scanned location. Using the absolute classifier with uncertainty-aware weighting, we can achieve a texture classification accuracy of 85.3%, as expected given our results on the validation set. However, in this section we also investigate how we can use the relative classifier for large area spatial mapping problems.

For relative texture classification, at every scan location, we input the current scan at that location along with a "candidate" baseline patch of known grit. This candidate patch can be directly taken from the training set (since we already have labels for those scans) or from a separate dataset of annotated candidates. Thus, the model will predict the grit between the current scan (of unknown grit) and another candidate scan of known grit. Comparing against several candidates from various different grit classes will inform the prediction as to what grit the current scan is. For example, if we compare the current scan to a candidate of known grit 120, and the model predicts that the current scan is very likely of greater grit, then we can infer that the scan is most likely of grit 180 (higher than 120).

However, even with uncertainty aware weighted predictions, if the random candidate sample scan is of poor quality, then the final predictions could also be of poor quality. Thus, we found that using multiple candidate samples and averaging over the results can drastically improve performance. In some sense, humans operate in a very similar manner for the same task. When comparing the roughness of two surfaces, we do not place our fingers down statically in a single place, but rather slide our fingers spatially across the surface collecting many comparisons to ensemble and bootstrap our prediction with.

In Fig. 7B we visualize the effect of adding more candidate samples while continuing to leverage our uncertainty weighting algorithm to filter out high uncertainty regions of the scan. We observe that a greater number of candidates, like dropout samples, results in an increase in accuracy with a plateau eventually coming once enough candidates (i.e., 20) have been considered. The heat maps are visualized by spatially coloring the regions based on the nearest scan prediction. Accuracy of the predictions are computed with respect to the ground truth grit classifications (Fig. 7C).

VI. Conclusion

In this paper, we proposed an automated system for learning, directly from raw tactile sensory scans, texture classification of wooden sanding surfaces. Our solution enables adaptive robotic sanding where the robot can adapt its sanding levels based on feedback on its progress. We leverage soft tactile sensing using the Gelsight sensor to perceive touch and propose novel uncertainty-aware learning algorithm to classify absolute grit of a single scan or relative grit change between two scans. Furthermore, we develop a novel uncertainty aware prediction algorithm which leverages the model's uncertainty in its prediction in order to improve its inference procedure. We experimentally demonstrate, through human user study experiments, that our models consistently outperform humans at comparable tasks across a range of varying texture levels. Our results show that the concept works on pine over a range of 60-180 grit. In the future, we plan to expand the number grits, and the number of materials to find the limits of the sensor hardware and explore how this solution scales to more categories of classification. We believe that higher resolution on the camera hardware, softer gels, and smoother

Gelsight coatings will enable finer resolution sensing and thus increase our discriminative ability at higher resolution grits. Our study demonstrates the promise of this technique for enabling closed loop feedback sanding operations. Such a future closed loop sanding system could help liberate humans form the dull, dirty, and dangerous job of sanding.

ACKNOWLEDGEMENT

The authors would like to thank Ava Soleimany, Sandra Q. Liu, Edward Adelson for their help in this research. Support for this work was given by the National Science Foundation (NSF) and Toyota Research Institute (TRI).

REFERENCES

[1] A. T. Saber, I. K. Koponen, K. A. Jensen, N. R. Jacobsen, L. Mikkelsen, P. Møller, S. Loft, U. Vogel, and H. Wallin, "Inflammatory and genotoxic effects of sanding dust generated from nanoparticle-containing paints and lacquers," *Nanotoxicology*, vol. 6, no. 7, pp. 776–788, 2012.

[2] S. C. Shen and R. A. House, "Hand-arm vibration syndrome: What family physicians should know," *Canadian Family Physician*, vol. 63, no. 3, pp. 206–210, 2017.

[3] P. Spielholz, S. Bao, and N. Howard, "A practical method for ergonomic and usability evaluation of hand tools: a comparison of three random orbital sander configurations," *Applied occupational and environmental hygiene*, vol. 16, no. 11, pp. 1043–1048, 2001.

[4] J. I. Lipton, R. MacCurdy, Z. Manchester, L. Chin, D. Cellucci, and D. Rus, "Handedness in shearing auxetics creates rigid and compliant structures," *Science*, vol. 360, no. 6389, pp. 632–635, 2018.

[5] A. R. A. Besari, M. D. b. M. Palil, R. Zamri, and A. S. Prabuwono, "A review of novel sensing techniques for automatic polishing robot system," in *National conference design and concurrent engineering*, pp. 353–357, 2008.

[6] A. M. Kabir, A. V. Shembekar, R. K. Malhan, R. S. Aggarwal, J. D. Langsfeld, B. C. Shah, and S. K. Gupta, "Robotic finishing of interior regions of geometrically complex parts," in *ASME 2018 13th International Manufacturing Science and Engineering Conference*, American Society of Mechanical Engineers Digital Collection, 2018.

[7] H. Kosler, U. Pavlovčič, M. Jezeršek, and J. Možina, "Adaptive robotic deburring of die-cast parts with position and orientation measurements using a 3d laser-triangulation sensor," *Strojniški vestnik-Journal of Mechanical Engineering*, vol. 62, no. 4, pp. 207–212, 2016.

[8] R. Majchrowski, M. Grzelka, M. Wieczorowski, Ł. Sadowski, and B. Gapiński, "Large area concrete surface topography measurements using optical 3d scanner," *Metrology and Measurement Systems*, vol. 22, no. 4, pp. 565–576, 2015.

[9] K. Okajima, S. Nagaoka, M. R. Islam, R. Ito, and K. Watanabe, "Development and testing of a surface roughness measurement device based on aerial ultrasonic reflections," *Paddy and Water Environment*, pp. 1–9, 2019.

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

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

[12] R. L. Truby, R. K. Katzschmann, J. A. Lewis, and D. Rus, "Soft robotic fingers with embedded ionogel sensors and discrete actuation modes for somatosensitive manipulation," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 322–329, IEEE, 2019.

[13] R. Li and E. H. Adelson, "Sensing and recognizing surface textures using a gelsight sensor," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1241–1247, 2013.

[14] W. Yuan, R. Li, M. A. Srinivasan, and E. H. Adelson, "Measurement of shear and slip with a gelsight tactile sensor," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 304–311, IEEE, 2015.

[15] W. Yuan, M. A. Srinivasan, and E. H. Adelson, "Estimating object hardness with a gelsight touch sensor," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 208–215, IEEE, 2016.

[16] W. Yuan, Y. Mo, S. Wang, and E. H. Adelson, "Active clothing material perception using tactile sensing and deep learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8, IEEE, 2018.

[17] D. A. Nix and A. S. Weigend, "Estimating the mean and variance of the target probability distribution," in *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, vol. 1, pp. 55–60, IEEE, 1994.

[18] C. M. Bishop, "Mixture density networks," 1994.

[19] A. Amini, G. Rosman, S. Karaman, and D. Rus, "Variational end-to-end navigation and localization," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8958–8964, IEEE, 2019.

[20] I. Gilitschenski, R. Sahoo, W. Schwarting, A. Amini, S. Karaman, and D. Rus, "Deep orientation uncertainty learning based on a bingham loss," in *International Conference on Learning Representations*, 2019.

[21] G. Costante and M. Mancini, "Uncertainty estimation for data-driven visual odometry," *IEEE Transactions on Robotics*, 2020.

[22] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?," in *Advances in neural information processing systems*, pp. 5574–5584, 2017.

[23] H. Wang and D.-Y. Yeung, "Towards bayesian deep learning: A survey," *arXiv preprint arXiv:1604.01662*, 2016.

[24] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, pp. 1050–1059, 2016.

[25] A. Amini, A. Soleimany, S. Karaman, and D. Rus, "Spatial uncertainty sampling for end-to-end control," *arXiv preprint arXiv:1805.04829*, 2018.

[26] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Advances in neural information processing systems*, pp. 6402–6413, 2017.

[27] T. Pearce, M. Zaki, A. Brintrup, N. Anastassacos, and A. Neely, "Uncertainty in neural networks: Bayesian ensembling," *arXiv preprint arXiv:1810.05546*, 2018.

[28] J. M. Hernández-Lobato and R. Adams, "Probabilistic backpropagation for scalable learning of bayesian neural networks," in *International Conference on Machine Learning*, pp. 1861–1869, 2015.

[29] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural networks," *arXiv preprint arXiv:1505.05424*, 2015.

[30] M. Fortunato, C. Blundell, and O. Vinyals, "Bayesian recurrent neural networks," *arXiv preprint arXiv:1704.02798*, 2017.

[31] I. Osband, C. Blundell, A. Pritzel, and B. Van Roy, "Deep exploration via bootstrapped dqn," in *Advances in neural information processing systems*, pp. 4026–4034, 2016.

[32] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential deep learning to quantify classification uncertainty," in *Advances in Neural Information Processing Systems*, pp. 3179–3189, 2018.

[33] A. Amini, W. Schwarting, A. Soleimany, and D. Rus, "Deep evidential regression," *arXiv preprint arXiv:1910.02600*, 2019.

[34] L. Chin, J. Lipton, R. MacCurdy, J. Romanishin, C. Sharma, and D. Rus, "Compliant electric actuators based on handed shearing auxetics," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 100–107, IEEE, 2018.

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[36] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, pp. 396–404, 1990.

[37] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[38] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, 2016.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.