DenseTact 2.0: Optical Tactile Sensor for Shape and Force Reconstruction

Won Kyung Do, Bianca Jurewicz, and Monroe Kennedy III

***Abstract*— Collaborative robots stand to have an immense impact on both human welfare in domestic service applications and industrial superiority in advanced manufacturing with dexterous assembly. The outstanding challenge is providing robotic fingertips with a physical design that makes them adept at performing dexterous tasks that require high-resolution, calibrated shape reconstruction and force sensing. In this work, we present DenseTact 2.0, an optical-tactile sensor capable of visualizing the deformed surface of a soft fingertip and using that image in a neural network to perform both calibrated shape reconstruction and 6-axis wrench estimation. We demonstrate the sensor accuracy of 0.3633*mm* per pixel for shape reconstruction, 0.410*N* for forces, 0.387$N \cdot mm$ for torques, and the ability to calibrate new fingers through transfer learning, which achieves comparable performance with only 12% of the non-transfer learning dataset size.**

I. INTRODUCTION

Robots must be able to manipulate objects with dexterity comparable to human performance in order to be effective collaborators in environments designed for humans. This requires both the physical design of a robotic fingertip that can accommodate complex objects as well as the modeling of the contact region relating deformation to calibrated shape and force measurements. Many robotic fingertip designs exist with various strengths and weaknesses with representative examples that include piezoelectric [1], optical [2]–[4], resistance [5], capacity [6] and hall effect [7]. Robotic fingertips can be broadly categorized in terms of transduction based sensors, where an electrical signal is caused by deformation and used to provide information of shape and force, versus optical-based sensors, where an image of the fingertip is observed and the deformation of the soft fingertip is correlated to shape and forces. For all of these sensors, the objectives are high contact resolution and calibration for both shape and forces. The challenge is obtaining high-resolution calibration for the shape and forces. Previous work explored high-resolution shape calibration for a vision based sensor [8], but a primary limitation was the inability to sense forces as well. A comparison of the proposed sensor and competitive sensor models is presented in Table I with comparison metrics for sensor resolution, design shape, and force sensing modality.

To obtain the shape from the interior of an optical tactile sensor, the incidence angle of the interior projected light with the surface of the sensor must be used to approximate the normal to the surface, and with these normal's the sensor shape can be constructed from Poisson integration either directly or through an approximation method that leverages neural networks [8]. To obtain forces, some studies leverage an inverse FEM model [9], use of an expressing marker, or leverage a skeletal structure for precise estimation via image input [10]. For all of these approaches, the deformation of the sensor must be tracked to correlate to observed forces. If a soft fingertip is used without skeletal structure [10], [11], then the force distribution approximation may require a huge computational load due to the nature of the hyper-elastic material of the sensor. This can be mitigated with data-driven models, which can approximate forces from a single deflected image. To track the interior of the sensor, markers must adorn the interior in order to observe normal, shear and torsional deflection. However simple patterns such as dots can suffer the effects of aliasing under large deflection [12]. To accommodate this, we propose the use of a randomized, continuous pattern deposited on the surface of the sensor for tracking large deflection without aliasing.

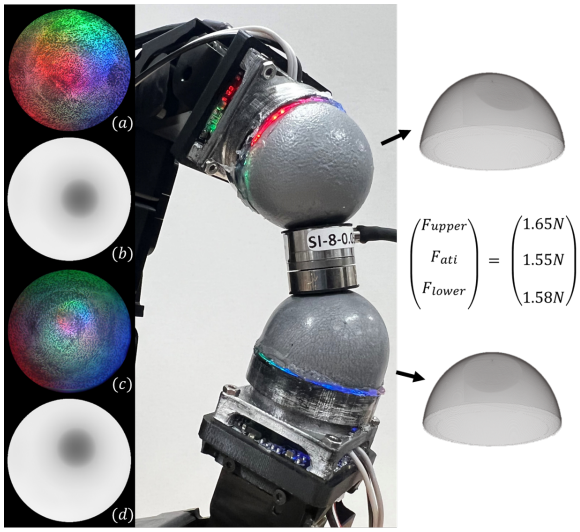

Fig. 1: **DenseTact 2.0.** Sensors are pinching ATI Nano sensor™. Image (a) and (c) shows the image taken from upper and lower sensors. Image (b) and (d) shows the corresponding resultant depth images. Right pointcloud represents the 3D reconstructed surface of sensor. Right middle matrix shows the estimated force from DenseTact 2.0s and ATI sensor.

Our contributions are as follows: 1) We present the physical design of the DenseTact 2.0 which has upgrades in modularity, lighting design, and is approximately 60% the size of DenseTact 1.0 [8] with a novel surface patterning

Authors are members of the ARMLab in the Mechanical Engineering Department, Stanford University, Stanford, CA 94305, USA. {wkdo, biancalj, monroek}@stanford.edu. The first author is supported by a fellowship from the Kwanjeong Educational Foundation. This work is supported by the National Science Foundation under Grant 2142773. Youtube link for DenseTact 2.0: https://youtu.be/5S74w0iSPz8

Name	Resolution	Force range	Shape
Gelslim 3.0 [11]	640×480	unspecified	full
TaTa [13]	1280×720	×	full
Skin sensor [7]	0.1 mm	$0 \sim 3N, 1$	partial
Softbubble [14]	224×171	×	full
Omnitact [15]	400×400	×	partial
NeuTouch [16]	39	×	×
Romero [17]	640×480	×	full
Optofiber-sensor [18]	61 fibers	$0.03 \sim 8N, 5$	×
GelTip [19]	×	unspecified	partial
Digit [20]	640×480	×	partial
Insight [10]	1640×1232	$0.03 \sim 2N, 5$	partial
DenseTact 1.0 [8]	800×600	×	full
DenseTact 2.0	1024×768	$-11 \sim 3N, 6$	full

TABLE I: **Related Work.** Table shows resolution of sensor, sensing range and dimension of the force, and availability of shape reconstruction. 'Partial' means the sensor does not estimate the depth of the entire sensing area, or estimates only the position of contact.

deposition technique. 2) We present the combination of a calibrated high-resolution shape reconstruction with a calibrated 6-axis wrench estimation. 3) We provide a comparison study between leading monocular depth estimation models with our model applied to both shape reconstruction and force estimation. 4) We show the effectiveness of transfer learning of our model for faster and more efficient training of future sensors with consistent geometry for calibration and deployment.

The paper is organized as follows: Sec. II presents the sensor design of DenseTact 2.0, Sec. III presents the data preparation for force and shape estimation, IV presents the method of estimating force and shape with modeling, Sec. V shows results for the shape reconstruction model and force estimation model, and the conclusion and future work is discussed in Sec. VI.

II. DenseTact 2.0 Sensor Design

A. Design Criteria

A tactile sensor with a highly-deformable gel has a clear advantage for the vision-based approach. Gel deformation not only enables collecting the information of the contact object, but also easily tracks features, even with small indentation. In order to extract as much geometrical and force information from the single image, the sensor requires more attractive features. Furthermore, the in-hand manipulation is more prone to happen with a compact sensor size. To deal with these issues, we augmented the design of DenseTact [8] with following features: 1) Reduced sensor size while maintaining highly-curved 3D shape. 2) Modular design using off-the-shelf materials for easy assembly and resource-efficiency. 3) Enriched features with randomized pattern on the surface for force estimation.

B. Gel Fabrication with Randomized Pattern

The fabrication process of the gel follows three steps: 1. making a gel base [8], 2. printing a randomized pattern on the surface of the gel, and 3. covering the gel with a reflective surface.

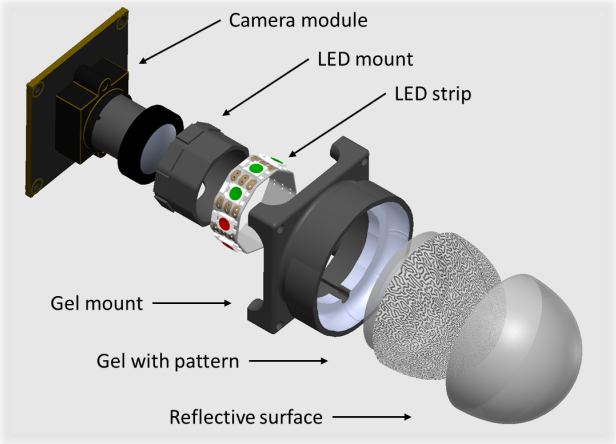

Fig. 2: **DenseTact Design**. Exploded view of DenseTact 2.0. The gel mount reflects the light from LED while the gel is covered with the pattern and reflective surface.

1) Gel Base: The material of the gel is the same material (49.7 shore OO hardness) and similar mold shape as in [8], while the compact hemispherical shape has a 31.5mm diameter. Compared to the DenseTact 1.0, we increased the contact area between the gel mount and lens to be more durable - the contact area - volume ratio of the DenseTact 1.0 is $0.0707mm^{-1}$ ($Area\ /\ Vol = 3,264.3mm^2\ /\ 46,173mm^3$), and the ratio of DenseTact 2.0 is $0.1229mm^{-1}$ ($Area\ /\ Vol = 1,443.4mm^2\ /\ 11,746mm^3$).

2) Randomized Pattern on the Gel Surface: The randomized pattern can hold more information for extracting features from a single image, such as continuous deformation output or non-aliasing problem. Marker-based approaches seen in most tactile sensors are hard to deal with the aliasing problem with large deformation. One other approach such as the use of a randomly colored pattern [21] enables intrinsic features to follow, but it is only applicable for sensors with planar surfaces, and the RGB channel can interfere with the pattern itself. Furthermore, the pattern requires to have a unique pattern to avoid the aliasing problem and maintain a balanced density between the pattern and background to extract the feature from the surface deformation.

To create the unique pattern, we first distribute points on the 2D planar surface using the voronoi stippling technique [22] and randomly connect all points. Connecting the array of points can be considered as the Traveling Salesman Problem (TSP), a classic algorithm for finding the shortest route to connect a finite set of points with known positions. We connect all points with a TSP solver [23], convert the solution as an image file, and extract the unique pattern using 8,192 points on a $25mm \times 25mm$ size square.

We printed a stamp plate of the randomized pattern using a laser cutter with a depth of 0.03mm. Next, we spread an ink on the plate, where the ink is composed of a silicone base with black ink (Smooth-on Psycho Paint™ and pigment, the ratio of silicone base to ink is 5:1). Then we scrape the ink on the plate so that the ink only remains on the imprinted part of the stamp. Next, we press the cured gel onto the ink

and distribute the pattern evenly by contacting all parts of the surface only once. The result of the printed pattern is shown in input images at Fig. 1.

3) Reflective Surface: The reflective surface is made from a mixture of silicone paint and silver silicone pigment with the ratio of 2:1. 0.5% of the solution of Thivex thickening solution is added to the mixture. Then, the mixture is placed in a vacuum chamber to remove any air bubbles that may be present from mixing the materials together. The method of applying the reflective surface improved from versions DenseTact 1 to 2.0 as the application time reduced from 2 hours to 30 minutes. This is done by leveraging a paint dipping technique as opposed to airbrushing. To execute this method, a suction cup is used to grip the gel, which is then dipped into the silicone ink mixture. The gel is dipped in the ink a total of three times and a heat gun is used to cure the paint after each dip. With this method, users can easily repair from possible abrasion created through extended gel usage by dipping the gel into the ink solution whenever necessary.

C. Sensor Fabrication

The bottom part of the sensor contains a camera, LED mount, LED strip, and a gel mount covered with mirror-coating. The sensor's exploded view is shown in Fig. 2.

1) Illumination with Mirror-Coated Wall: The major requirement for a vision-based sensor is illumination. Because of the compact size of the sensor and the LED being a point light source with a limited angle of light emission, the LED strip with a single RGB channel LED had a limitation when the sensor became smaller. Therefore, we implemented the new illumination system with a mirror-coated wall while still maintaining the simple assembly feature.

Instead of using 3 LED lights as in [8] or other tactile sensors, we utilized 9 LED lights (3 LEDs for each color-red, green, and blue) from an LED strip (Adafruit Mini Skinny NeoPixel ™) while controlling the intensity of each LED. As shown in Fig. 2, the LED surrounds the camera while facing outside. An equivalent distance between each LED with more lights allows the sensor to get an equal distribution of lights. Furthermore, the increased brightness makes the sensor more resistant to external lights.

The 3D-printed gel mount reflects the lights to the gel through the mirror-coated surface. To develop the mirror-like effect on the side, we flattened the surface of the gel mount with XTC-3D™ and coated it with the mirror-coating spray. Finally, the lights on the LED pass through the opposite side of the gel (see the input image in Fig. 1).

2) Sensor Assembly: We modularized the sensor into three parts - gel with gel mount and lens, LED module, and camera module. Each module is easily replaceable while the other modules remain intact. The gel, gel mount and lens are firmly attached through sil-poxy adhesive™ and Loctite Powergrab™ Crystal Clear adhesive. The gel module and LED module is fixed through the 4 screws with camera module. The user can simply unscrew and replace either the camera, LED, or gel module. Since the sensor has more contact area-volume ratio, the durability increased even with the modularized design.

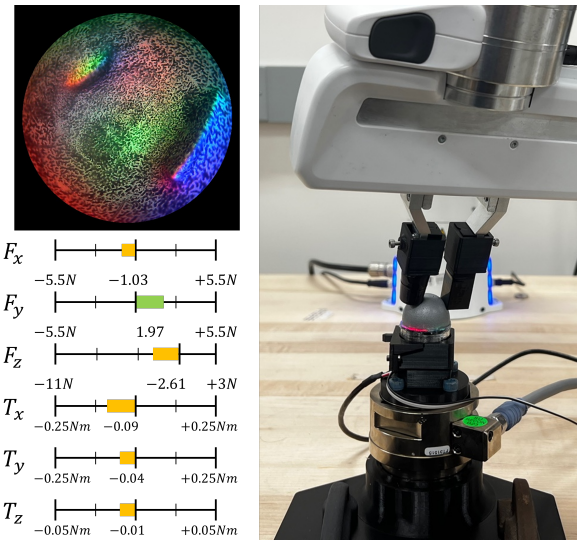

Fig. 3: **Force Data Collection.** Densetact 2.0 has been pushed under an ATI Gamma sensor with the Franka™ arm, where the grippers are covered with indenters.

We chose to use the camera module Sony IMX179 (30fps) and M12 size lens with the field of view 185 deg degree for easy replacement. The distance of focal length of the camera is manually set to optimize the focal length for the expected deformation. The final size of the sensor including the camera is $W \times D \times H = 32 \times 32 \times 43mm$ with the weight 34g. The cost of the sensor became cheaper because of the smaller LED strip ($3.75), gel part ($3), and camera LED mount ($1.5) with the same price of the camera system ($70).

III. Data Preparation for Force and Shape Estimation

A. Data Collection Process for Shape Reconstruction

The dataset for shape estimation has been collected in a similar manner as the DenseTact 1.0 with more autonomy [8]. While utilizing the CNC machine with a stepper motor for precise movement, we implemented an encoder for the stepper motor and a limit switch for an autonomous procedure. The sensor is attached to the stepper motor side with a mount. The mount ensures the center of the sensor is aligned with the rotational center of the stepper motor.

In order to collect more datasets in one process, 21 indenters, a Stl model which covers the entire sensor surface, are placed in a 3×7 grid on the plate of the CNC machine. Each row contains the same shape of indenters, each with a different orientation along a random axis. The rotation axis is aligned with the center of each indenter and placed in the xy plane. Therefore, each row shows different orientations by rotation on the x and y axes, while the z axis rotational difference is provided from the stepper motor. As a result, each data collection procedure generates up to 8,400 image datapoints (21 indenters $\times$ 400 steps/rev) without human intervention.

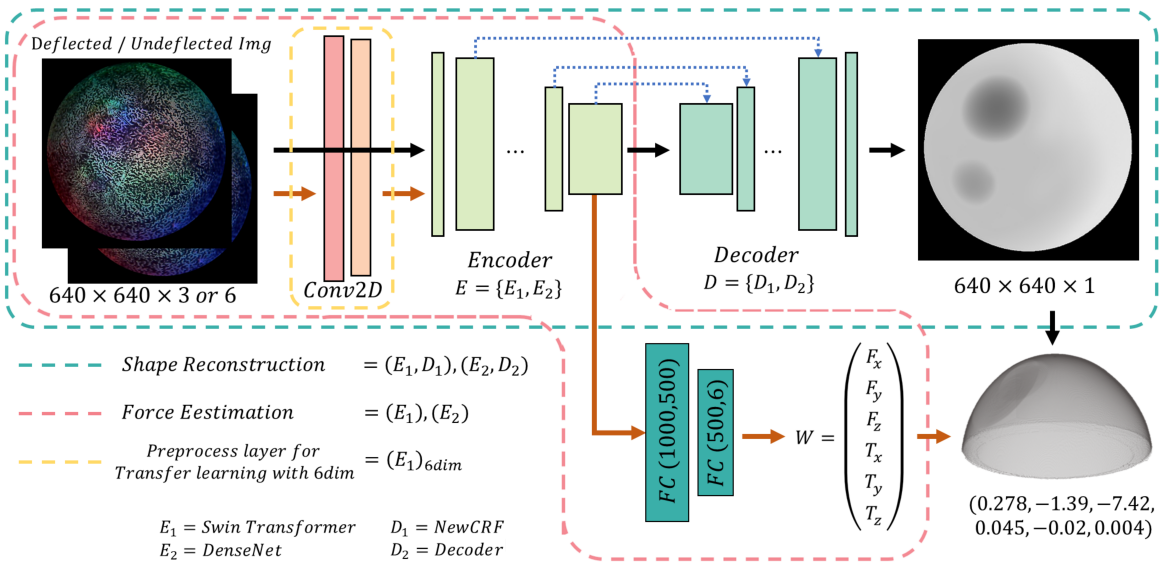

Fig. 4: **DenseTact 2.0 Algorithm**. The sensor interior is the input to the encoder-decoder network for shape estimation model (blue box). Force estimation models (red box) consist of encoder and fully-connected layers. Transfer learning model with 6 dimension (includes yellow box) takes deflected and undeflected image as an input. (Model references: E_1- [24], E_2- [25], D_1- [26], D_2 - [27])

Consideration of the bulging effect is a major improvement in the data collection process. Since the silicone gel is an incompressible, and hyper-elastic material, deformation causes bulging on regions not in contact with the indentor. For shape reconstruction, we account for this in the generated contact shape Stl file. In this way, the sensor is exposed to a more natural deformation. The size of captured image from camera has been increased into $(1024 \times 768 \times 3)$, which leads the final image size into $(640 \times 640 \times 3)$. After collecting the input image, the depth is reprocessed from the corresponding Stl model through a Gaussian Process with a ray casting algorithm [8]. The Stl files are available on github [1].

The default distance of the sensor in the dataset is 15.5mm, and its radial distance ranges from 12.23mm to 16.88mm, while the opposite side of the sensor bulges when deformed, resulting in a larger depth value than the hemisphere radius. Allowing for a margin around 0.05mm, we normalized the depth value from 12.23mm to 16.88mm (4.64mm range) into 0-255 pixel values for ground truth depth images. Finally, the 1 pixel increment corresponds to a 0.0182mm increment in depth value. We collected a dataset for two sensors - the sensor 1 has 38,909 training and 1,000 test configurations, and the sensor 2 has 20,792 training and 1,000 test configurations. Test configurations for each sensor are recorded with an unseen indenter from the training dataset. The datasets have a total 8.7GB and 6.8GB size for the sensor 1 and 2.

B. Dataset Collection for Force Estimation

The force dataset has been generated through randomly pushing sensors with the Franka arm. This method allowed us to collect the dataset with no constraints on the pose of the franka arm. The right image in Fig. 3 shows the configuration of the force dataset collection where the DenseTact 2.0 is mounted on the ATI Gamma sensor (SI-65-5). We created 10 different objects to push the sensor and collected the dataset while either attaching an object on each gripper finger or by gripping an object. The set of objects includes cylindrical shapes, spherical shapes and daily objects such as nuts. All joint positions including the position of the gripper fingers were recorded during dataset generation in the rate of 1,000 paths per second. The recorded motion of the Franka arm for calibration makes multiple sensor calibration easier by requiring less human intervention.

During online dataset collection, we filter out duplicated sequential images which do not show significant change using Peak signal-to-noise ratio (PSNR) as a similarity metric. The ring buffer collects the image up to 5 current images and applies the following threshold - $PSNR(Img_{curr}, Img_{prev,i}) < 0.9$, where $i = 1, \ldots, 5$. The dataset has been collected within the range specified in the left part of the Fig. 3, where the unit of force and torque are N and $N \cdot m$. the left image and force distribution in Fig. 3 shows the collected input image and corresponding force and torque data. The final dataset has been normalized between each force and torque range. The dataset has been collected for two sensors - the sensor 3 has 38,909 training force points and 1,000 test points, and the sensor 4 has 20,792 training points and 1,000 test points. Test points are collected with different shape as the pushing objects used in each training points. Total size of each dataset is 7.5GB and 2.7GB, respectively.

IV. Algorithms for Shape and Force Reconstruction

A. Algorithms for Shape Reconstruction

While the randomized pattern on the surface adds more features for continuously tracking surface movement, recon-

[1] https://github.com/armlabstanford/DenseTact

structing the sensor surface requires learning features such as the deflected part's location, or surface normal based on the LED position. The position of the random pattern also gives the dynamic movement of the sensor, which requires the networks to learn more features from a single image. Therefore, we compared two network models to reconstruct the shape of the sensor surface.

1) Network with Swin Transformer and NeWCRF: The Vision Transformer (ViT) is a well-known model from transformer-based architecture for image classification [28]. While ViT splits an image into patches and train position embedding for each image patch, the Swin transformer builds the feature maps hierarchically with lower computational complexity because of a localized self-attention layer [24]. Our input image contains closely-related information relation between neighbor pixels. Therefore, the path embedding with hierarchical feature maps between each layer can better connect information between the indented and opposite parts.

Once the input image has been trained with swin transformer as the encoder part, the decoder is also important for correlated embeddings. Recently, models using a classification model to boost the performance of depth estimation, such as binsformer [29] or adpative bins [30], perform well with the monocular depth estimation. However, Neural window FC-CRFs (NeWCRF) reaches the same performance by applying Conditional Random Field (CRF) on the decoder part to regress the depth map by utilizing fully-connected CRFs on each split image part (window) [26]. Therefore, we chose the Swin Transformer with NeWCRF decoder among the state of the art models for monocular depth estimation.

As shown in the green part of Fig. 4, the network gets input as $640 \times 640 \times 3$. Without using the pretrained model, we normalized both the input and ground truth depth images from 0 to 1 (ground truth originally 0-255). We utilized 4 swin-transformer blocks on the encoder part using window size 20, the number of patches from one image. The predicted model is compared with the ground truth using the Scale-Invariant Logarithmic loss (SILog loss) with 0.85 as variance minimizing factor [31]. The model is trained for 21 epochs with the batch size of 8 while the learning rate starts from 2×10^{-5} on the 4 Nvidia A4000 GPUs. The model took about 36 hours for training.

2) Densetact Net Position: The above model is compared with the Network from [8] without resizing the image. As shown in Fig. 4, the network consists of an encoder as Densenet [25], and a simple decoder with skipped connections [27]. The final result has been upsampled by the upsampling layer to get the $640 \times 640 \times 1$ as an output depth image. Unlike the above model, the input and ground truth are un-normalized. The network is trained without any prior or pretrained model and used the reciprocal of the depth for structural similarity loss.

By comparing the above model with ours, we can show 1) whether the random pattern blocks the estimation result and 2) how many model parameters are enough to estimate the depth or force estimation. The training runs for 25 epochs with batch size 8. The learning rate is set to 1×10^{-4}, where the model took about 16 hours for training.

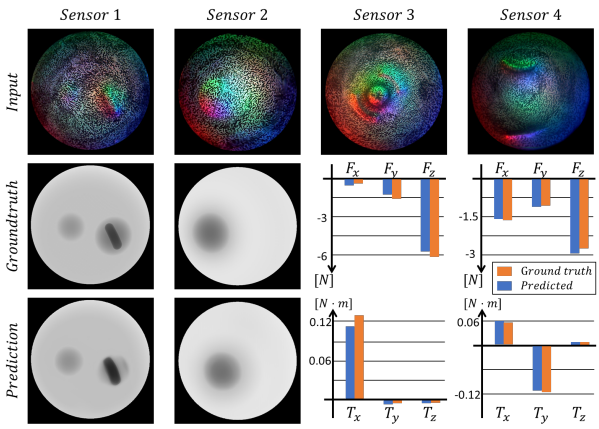

Fig. 5: **Shape and Force Reconstruction Performance.** Examples from test set for sensor shape and force reconstruction.

B. Algorithm for Force Estimation

The network model for force estimation utilized each encoder part of the above two models. The network structure for force estimation is illustrated in Fig. 4. After passing either the Swin transformer encoder or the Densenet-based encoder, two fully-connected layers shrink the channel size from 1,000 to 500, and from 500 to 6, which corresponds to the 6 wrench inputs. The learning rate starts decreasing from 2×10^{-5}. Both models are trained with batch size 8 for 22 epochs. The training has been done for each sensor dataset.

C. Data-efficient Training with Transfer Learning

One major disadvantage of the data-driven approach in tactile sensing estimation is that the data collection process is much longer than the modeling-based approach. However, the pre-trained model from multiple sensor datasets can significantly reduce the burden for the calibration process. We proposed a two network structure for transfer-learning model for our sensor.

The first model, 3-dim model, has a network identical to the Densenet-based encoder, which takes input as $640 \times 640 \times 3$. The training datasets are combined for both sensors, while the test sets are also combined from each sensor's test dataset. Finally, the total number of configuration training data points is 59,701 and for the test dataset is 1,200, where 600 data points are extracted from each sensor.

The second model, 6-dim model, simultaneously takes the undeflected image and deflected image of the sensor so that the input becomes $640 \times 640 \times 6$. The blue part in Fig. 4 indicates the layer applied for the model with 6-dimensional inputs. Two blocks of convolutional, batchnorm, and relu layers reduces the number of channels from 6 to 4, and from 4 to 3. Finally, the input has been passed to the same Densenet-based encoder. Both models are trained for 30 epochs with a batch size of 16. After getting the pretrained model, we compared the loss of each model by using a small dataset of the another, *unseen* sensor input for both force and

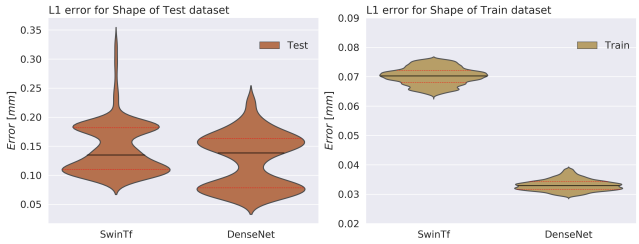

Fig. 6: **Violin plot of shape reconstruction model.** Left and right image compares L1 error for model SwinTF and DenseNet on test and train dataset, respectively.

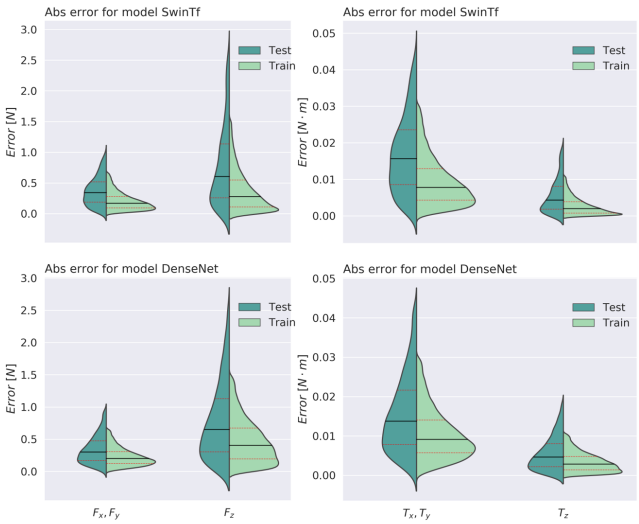

Fig. 7: **Violin plot of force estimation model.** Upper and lower images show the absolute error of train and test dataset from model SwinTF and DenseNet. Left and right side shows the plot of $F_{x,y}, F_z$, $T_{x,y}, T_z$, respectively.

position dataset. The size of the small dataset is 4,643 and the dataset is pushed with a single object.

V. Results and Discussion

A. Validation of Models

Fig. 6 and Fig. 7 show the network evaluation with violin plots for position and force for all training and test datasets, respectively. The qualitative performance of the model is shown in Fig. 5. The mean of the L2 loss for SwinTF model, a shape model with encoder as Swin transformer and decoder as NeWCRF, with shape test set is $(sen_1, sen_2) = (0.370mm, 0.584mm)$, $err_{tot} = 0.4769mm$, and the L2 loss for the DenseNet model, a shape model with encoder as DenseNet and decoder as skipped decoder, is $(sen_1, sen_2) = (0.282mm, 0.445mm)$, $err_{tot} = 0.3633mm$. Both results show that the DenseNet model outperforms the SwinTF model.

The force model also shows that the DenseNet model, a force model with encoder as DenseNet, performs slightly better than the SwinTF model, a model with encoder as Swin transformer. The total absolute mean error of force of each sensor on the test set for the SwinTF model is $(sen_3, sen_4) = (0.426N, 0.436N)$, and force error for the DenseNet model is $(sen_3, sen_4) = (0.409N, 0.410N)$. For the torque, the absolute mean errors are $(sen_3, sen_4) = (0.416N \cdot mm, 0.438N \cdot mm)$ for the SwinTF model, and $(sen_3, sen_4) = (0.395N \cdot mm, 0.377N \cdot mm)$ for the DenseNet model. In conclusion, DenseTact 2.0 performs the shape reconstruction with an absolute mean error of 0.3633mm with the DenseNet model, and performs force estimation with an absolute mean error of $(e_{force}, e_{torque}) = (0.410\text{N} , 0.387\ N \cdot mm)$.

The result of transfer learning indicates that the model that gets 3-dimensional input works similarly for both big datasets and small datasets. A 3-dim model trained with a small force dataset on top of a pretrained model has an absolute mean error of normalized wrench of 0.05163, whereas the 6-dim model with a pretrained model is 0.05143. However, the 3-dim model converged faster (converged at 7,200 steps) than the 6-dim model (converged at 8,400 steps). 3-dim model with a position dataset has converged faster (converged at 6,900 steps) with an rms error of 0.2275mm, than the 6-dim model (converged at 8,100 steps) with rms error of 0.2394mm.

B. Discussion

The results above demonstrate that the randomized pattern on DenseTact 2.0 performs well with both shape reconstruction and force estimation. The SwinTF model for shape reconstruction has about 270 million parameters, while the DenseNet model for shape has 44 million parameters. The forward path for the SwinTF model takes 0.124s per step, while the DenseNet model takes 0.04s per step on Nvidia 3090 GPU. Therefore, the Densetact input requires fewer parameters for training and works better with the DenseNet model. The state of the art model for monocular depth estimation might perform well on the daily objects and general images, but the input image for DenseTact 2.0 requires solving the relation between each pixel by estimating the position of the LEDs and tracking the pattern deflection.

Both transfer-learning models demonstrated that the model works with a smaller dataset, which is about 10% of the original training dataset size. The training time only took 1 hours and 1.8 hours for force and position, respectively, for 30 epochs with a batch size 10 on Nvidia A4000 GPU. Considering that the data calibration process can be done easily with the ATI force sensor, the calibration time is comparable with the model-based sensors.

VI. Conclusion

This paper presents DenseTact 2.0, a compact-size calibrated shape and force sensor with an very soft gel, which is capable of reconstructing surface shape and force estimation with high resolution. The modularized design and compact size of DenseTact 2.0 enables versatile in-hand manipulation as well as easy assembly. We leverge a marker deposition algorithm which avoids aliasing under large sensor deformations. We benchmarked multiple models and show the benefit of using transfer learning which allows us to use 12% of the original dataset size. Our future direction is estimating force distribution from the single image output while applying the transfer learning feature as well.

REFERENCES

[1] N. Wettels, J. A. Fishel, Z. Su, C. H. Lin, G. E. Loeb, and L. SynTouch, "Multi-modal synergistic tactile sensing," in Tactile sensing in humanoids—Tactile sensors and beyond workshop, 9th IEEE-RAS international conference on humanoid robots, 2009.

[2] S. Dong, W. Yuan, and E. H. Adelson, "Improved GelSight tactile sensor for measuring geometry and slip," in IEEE International Conference on Intelligent Robots and Systems, vol. 2017-Septe. IEEE, sep 2017, pp. 137–144. [Online]. Available: http://ieeexplore.ieee.org/document/8202149/

[3] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," The International Journal of Robotics Research, vol. 0, no. 0, p. 02783649211027233, 2021. [Online]. Available: https://doi.org/10.1177/02783649211027233

[4] F. R. Hogan, M. Bauza, O. Canal, E. Donlon, and A. Rodriguez, "Tactile Regrasp: Grasp Adjustments via Simulated Tactile Transformations," IEEE International Conference on Intelligent Robots and Systems, pp. 2963–2970, 2018.

[5] M. Cheng, X. Huang, C. Ma, and Y. Yang, "A flexible capacitive tactile sensing array with floating electrodes," Journal of Micromechanics and Microengineering, vol. 19, no. 11, p. 115001, 2009.

[6] T. M. Huh, H. Choi, S. Willcox, S. Moon, and M. R. Cutkosky, "Dynamically Reconfigurable Tactile Sensor for Robotic Manipulation," IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 2562–2569, 2020.

[7] Y. Yan, Z. Hu, Z. Yang, W. Yuan, C. Song, J. Pan, and Y. Shen, "Soft magnetic skin for super-resolution tactile sensing with force self-decoupling," Science Robotics, vol. 6, no. 51, p. eabc8801, 2021.

[8] W. K. Do and M. Kennedy, "Densetact: Optical tactile sensor for dense shape reconstruction," in 2022 International Conference on Robotics and Automation (ICRA), 2022, pp. 6188–6194.

[9] D. Ma, E. Donlon, S. Dong, and A. Rodriguez, "Dense Tactile Force Estimation using GelSlim and inverse FEM," Proceedings - IEEE International Conference on Robotics and Automation, vol. 2019-May, pp. 5418–5424, aug 2019.

[10] H. Sun, K. J. Kuchenbecker, and G. Martius, "A soft thumb-sized vision-based sensor with accurate all-round force perception," Nature Machine Intelligence, vol. 4, no. 2, pp. 135–145, Feb. 2022. [Online]. Available: https://www.nature.com/articles/s42256-021-00439-3

[11] I. Taylor, S. Dong, and A. Rodriguez, "Gelslim3. 0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger," arXiv preprint arXiv:2103.12269, 2021.

[12] Y. Du, G. Zhang, Y. Zhang, and M. Y. Wang, "High-resolution 3-dimensional contact deformation tracking for fingervision sensor with dense random color pattern," IEEE Robotics and Automation Letters, vol. 6, no. 2, pp. 2147–2154, 2021.

[13] S. Li, X. Yin, C. Xia, L. Ye, X. Wang, and B. Liang, "Tata: A universal jamming gripper with high-quality tactile perception and its application to underwater manipulation," in 2022 International Conference on Robotics and Automation (ICRA). IEEE, 2022, pp. 6151–6157.

[14] A. Alspach, K. Hashimoto, N. Kuppuswarny, and R. Tedrake, "Soft-bubble: A highly compliant dense geometry tactile sensor for robot manipulation," RoboSoft 2019 - 2019 IEEE International Conference on Soft Robotics, pp. 597–604, 2019.

[15] A. Padmanabha, F. Ebert, S. Tian, R. Calandra, C. Finn, and S. Levine, "OmniTact: A Multi-Directional High-Resolution Touch Sensor," in Proceedings - IEEE International Conference on Robotics and Automation. Institute of Electrical and Electronics Engineers Inc., may 2020, pp. 618–624.

[16] T. Taunyazov, W. Sng, H. H. See, and B. Lim, "Event-Driven Visual-Tactile Sensing and Learning for Robots," Robotics: Science and Systems, 2020.

[17] B. Romero, F. Veiga, and E. Adelson, "Soft, Round, High Resolution Tactile Fingertip Sensors for Dexterous Robotic Manipulation," May 2020, arXiv:2005.09068 [cs]. [Online]. Available: http://arxiv.org/abs/2005.09068

[18] D. Baimukashev, Z. Kappassov, and H. A. Varol, "Shear, Torsion and Pressure Tactile Sensor via Plastic Optofiber Guided Imaging," IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 2618–2625, apr 2020.

[19] D. F. Gomes, Z. Lin, and S. Luo, "Geltip: A finger-shaped optical tactile sensor for robotic manipulation," in 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020, pp. 9903–9909.

[20] M. Lambeta, P.-W. Chou, S. Tian, B. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer, et al., "Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation," IEEE Robotics and Automation Letters, vol. 5, no. 3, pp. 3838–3845, 2020.

[21] Y. Zhang, X. Chen, M. Y. Wang, and H. Yu, "Multidimensional tactile sensor with a thin compound eye-inspired imaging system," Soft Robotics, 2021.

[22] N. P. Rougier, "[re] weighted voronoi stippling," The ReScience journal, vol. 3, no. 1, 2017.

[23] D. Applegate, R. Bixby, V. Chvatal, and W. Cook, "Concorde tsp solver," 2006.

[24] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10 012–10 022.

[25] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.

[26] W. Yuan, X. Gu, Z. Dai, S. Zhu, and P. Tan, "Neural window fully-connected crfs for monocular depth estimation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 3916–3925.

[27] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," arXiv preprint arXiv:1812.11941, 2018.

[28] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.

[29] Z. Li, X. Wang, X. Liu, and J. Jiang, "Binsformer: Revisiting adaptive bins for monocular depth estimation," arXiv preprint arXiv:2204.00987, 2022.

[30] S. F. Bhat, I. Alhashim, and P. Wonka, "Adabins: Depth estimation using adaptive bins," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 4009–4018.

[31] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," Advances in neural information processing systems, vol. 27, 2014.