Supplementary - PressureVision++: Estimating Fingertip Pressure from Diverse RGB Images

Patrick Grady[1], Jeremy A. Collins[1], Chengcheng Tang[2], Christopher D. Twigg[2], Kunal Aneja[1], James Hays[1], Charles C. Kemp[1]

[1]Georgia Institute of Technology, [2]Meta Reality Labs

1. Introduction

This supplementary document provides additional details and results that were not included in the main paper. Section 2 provides additional details surrounding the data collection, Section 3 provides additional details surrounding the network architecture and training, Section 4 provides additional details about the mixed reality applications, and Section 5 provides additional results.

2. Data Collection Details

2.1. Actions

Participants were prompted with actions from a list of prompts, shown in Table 9. Most of the actions were repeated across four force levels: low force, high force, slide (force unspecified), and no contact. Due to the highly varying frictional properties of each surface, we did not prompt a force level during the slide prompt. Not all participants completed all actions.

2.2. Data Collection Hardware

To record pressure, a Sensel Morph [5] pressure sensor was used. This sensor records a 105×185 pressure image. To vary the sensor's appearance, various commercially available adhesive vinyl coverings were applied to the sensor's active area. The location and lighting were also changed to vary exposure (and thus the amount of motion blur), hue, and saturation of the images.

Data was captured from seven consumer-grade webcams, including four Logitech Brio 4K webcams, one Dell Ultrasharp 4k webcam, one Elgato Facecam 1080p webcam, and one Lumina 4k webcam. All streams were recorded at 1080p and 30 FPS, and later down-sampled to 15 FPS due to the large size of the dataset.

Most of the data was captured under unaltered room lighting, however some was collected in a room illuminated with smart LED bulbs which randomly changed brightness, providing a greater diversity of lighting. The data collection took place in twenty different environments.

For recordings with the ground truth pressure sensor, the cameras were spatially calibrated with an ArUco board [1]. The cameras were temporally aligned with pressure sensor readings with a specialized tool. When pressed against the pressure sensor, the tool would illuminate, allowing the pressure readings and camera frames to be aligned.

2.3. Dataset Statistics

Participants	51
Cameras	7
Objects	106
Locations	20
Resolution	1920x1080
Framerate	15 FPS
Total Frames	2.9M
Full Train Frames	182k
Weak Train Frames	1805k
Full Val Frames	21k
Weak Val Frames	72k
Full Test Frames	305k
Weak Test Frames	509k
Mean force, high force prompt	19.6N
Mean force, low force prompt	3.6N

Table 5. ContactLabelDB Statistics.

We show additional information about the dataset in Table 5.

During data collection, participants were prompted to apply high and low forces. Figure 9 shows the distribution of total applied forces as measured in the fully labeled dataset. Pressure data is integrated over the sensor area to calculate total force. This plot includes data from all sequences, meaning that the data is representative of one-finger contact as well as five-finger contact. Generally, we find a consistent difference between the two classes, and

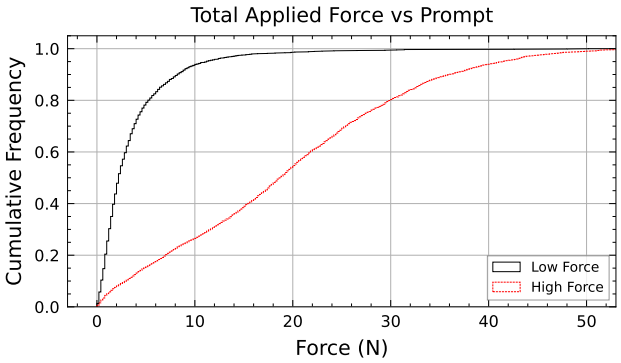

Figure 9. Cumulative distribution of total applied force versus prompt. During data collection, participants apply higher forces when prompted to apply "high force" as opposed to "low force".

participants apply higher forces when prompted in the "high force" case.

3. Network and Evaluation Details

3.1. Training Details

PressureVision++ takes cropped images of the hand as input. An off-the-shelf hand detector, MediaPipe [4], generates hand bounding boxes on all images. The bounding box is used to generate a crop of the hand for PressureVision++. We discarded frames where the hand detector did not find a hand. When the participant interacted with reflective surfaces, the hand detector frequently detected both the hand and its reflection. In cases when two hands were detected, the hand towards the top of the image was chosen.

PressureVision++ was implemented in PyTorch [6] and used network implementations from the *segmentation-models-pytorch* project [8]. PressureVision++ was trained with batches of 28 images: 14 fully-labeled images and 14 weakly-labeled images. Batch size was set to fully utilize the memory of an RTX 3090 GPU. The network was optimized with the Adam optimizer [3] for 300k iterations. The learning rate was 0.001 for the first 100k iterations, and 0.0001 thereafter. Training data is augmented with horizontal flips, color jitter, random rotations, scaling, and translations.

The complete loss function is:

$$L = L_p + \lambda_1 L_w + \lambda_2 L_d \tag{1}$$

We choose $\lambda_1 = 0.01$ and $\lambda_2 = 0.001$.

3.2. Cross-Dataset Generalization

In table 6, we show cross-dataset generalization results when PressureVision++ is tested and trained on PressureVisionDB [2] and ContactLabelDB. Although ContactLabelDB contains more diversity in terms of objects, it appears that the model trained on ContactLabelDB and tested on PressureVisionDB performs worse than the model trained on PressureVisionDB and tested on ContactLabelDB.

We hypothesize that this is because PressureVisionDB was captured in very harsh, artificial lighting conditions. These extreme lighting conditions are not captured in ContactLabelDB, which instead captures normal indoor lighting environments. We believe that models trained on ContactLabelDB generalize poorly to the extreme lighting captured in PressureVisionDB. During real-world testing, we find models trained on ContactLabelDB generalize much better to real-world scenarios.

Train \ Test	**PV-DB**	**CL-DB**
PV-DB [2]	41.3%	9.2%
CL-DB (ours)	2.3%	27.5%

Table 6. Cross-dataset results comparing PressureVisionDB (PV-DB) to ContactLabelDB (CL-DB).

3.3. Accuracy of Estimated Contact Labels

PressureVision++ produces two outputs for every input image: a pressure image and a contact label. The main paper analyzes the accuracy of the estimated pressure image, and this section evaluates the accuracy of the estimated contact label.

We compare the performance of the pressure estimate to the contact label estimate. We report the following metrics, which are computed over both the fully labeled and weakly labeled test sets:

- Contact Accuracy (pressure image) uses the estimated pressure image to determine if any contact is present across the entire image. This is compared to the ground truth contact. This is the same metric reported in Section 5 of the main paper.
- Contact Accuracy (contact label) uses the estimated contact label to determine if any contact is present across 5 fingers. This is compared to the ground truth contact.

Contact Accuracy (pressure image)	83.7%
Contact Accuracy (contact label)	86.1%

Table 7. Contact Accuracy compared between pressure estimates and contact label estimates.

We find that the pressure-based contact accuracy and contact-label-based contact accuracy perform similarly, with the contact-labeled-based estimate performing slightly better.

Contact Label Segment	Accuracy
Thumb	89.6%
Index	87.8%
Middle	90.8%
Ring	92.4%
Pinky	92.5%
Force	77.5%

Table 8. Per-finger and force accuracy.

We report per-finger contact label accuracy in Table 8. Force accuracy uses the estimated contact label to determine if the hand applies a high or low force. This is compared to the ground truth force level as prompted to the participant. The force accuracy is generally lower than the other segments of the contact label, suggesting that estimating the quantity of force is a more difficult task than the binary presence of contact.

4. Applications in Mixed Reality

4.1. Surface Interactions

In order to align coordinate frames between the RGB camera and the Meta Quest 2 headset, we designed a custom calibration tool (Figure 10). The calibration tool features an ArUco board [1] to estimate the pose of the RGB camera used for pressure estimation. The pose of the headset is calibrated by attaching a controller to the calibration tool. A calibration procedure is performed at the beginning of each session.

In order to calculate precise touch locations, the peaks of the pressure blobs are found with a local maxima detector. A custom application is developed for the Quest headset using Unity and the Oculus Integration Toolkit.

4.2. Net WPM Metric

For typing speed evaluations, words per minute (WPM) [7] is calculated by dividing the number of characters typed (including letters, spaces, and punctuation), c, by 5 to arrive at the number of words typed. Time t is measured in seconds between the first keystroke and pressing "Enter" to complete the sentence.

$$WPM = \frac{c/5}{t/60} \tag{2}$$

However, the WPM metric does not consider errors in typing. In our evaluations, we report net words per minute (Net WPM) [7], which modifies the standard WPM metric to factor in errors. A single character error (insertion, deletion, or substitution) results in the subtraction of 5 characters, or one word. Where e is the number of single-character errors, Net WPM can be calculated as:

$$NetWPM = \frac{c/5 - e}{t/60} \tag{3}$$

Figure 10. Calibration tool used to align coordinate frames between the RGB camera and the Meta Quest 2 headset. The controller is rigidly connected to the ArUco board.

4.3. Typing User Study

For the typing user study, 10 participants were recruited who did not participate in the collection of ContactLabelDB and who were not familiar with the research. The order of presentation of the two keyboards was randomized. Before collecting data, participants were allowed to practice typing with that keyboard for as long as they wanted.

After the study, participants were given a free-form text box to explain their perceived advantages and disadvantages of each typing method. They also rated which keyboard they preferred on a scale of 1 "strongly prefer Direct Touch Keyboard" to 5 "strongly prefer PressureVision++ Keyboard". The average score was 4.4, with only one participant not preferring the PressureVision++ keyboard.

We hypothesize as to the reasons why participants preferred the PressureVision++ Keyboard. For the Direct Touch Keyboard, due to the noise in pose estimation, to prevent false keystrokes, participants must press each key very deeply. Additionally, users generally must look at their hands to find the correct key since it is difficult to memorize the location of mid-air keys. For the PressureVision++ Keyboard, users only have to hover their fingers a few millimeters above the surface, and since the surface allows them to rest their hands and provides a reference point, they can type without looking at their hands. The most common error that participants made with the PressureVision++ keyboard was pressing a key adjacent to the desired key, resulting in single-character errors. We hypothesize that a simple autocorrect system would be able to correct these errors easily and improve typing speed.

5. Additional Results

Additional results are shown in Figures 11, 12, 13, and 14.

References

[1] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014. 1, 3

[2] Patrick Grady, Chengcheng Tang, Samarth Brahmbhatt, Christopher D. Twigg, Chengde Wan, James Hays, and Charles C. Kemp. PressureVision: estimating hand pressure from a single RGB image. *European Conference on Computer Vision (ECCV)*, 2022. 2

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*, 2015. 2

[4] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. Mediapipe: A framework for perceiving and processing reality. In *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019*, 2019. 2

[5] Morph. Sensel Morph haptic sensing tablet. `www.sensel.com/pages/the-sensel-morph`, Last accessed on 2020-02-25. 1

[6] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 2

[7] Timothy A Salthouse. Effects of age and skill in typing. *Journal of Experimental Psychology: General*, 113(3):345, 1984. 3

[8] Pavel Yakubovskiy. Segmentation models Pytorch, 2020. 2

Action	Force Level
Index, fingers	{Low, high, slide, no contact}
Thumb	{Low, high, slide, no contact}
Index and thumb	{Low, high, slide, no contact}
Index and middle	{Low, high, slide, no contact}
Middle	{Low, high, slide, no contact}
Ring	{Low, high, slide, no contact}
Pinky	{Low, high, slide, no contact}
All fingers	{Low, high, slide, no contact}
Press fingers sequentially	{Low, high}

Table 9. Participants were prompted according to this list of actions, e.g., *press thumb, low force*. Not all participants completed the entire list of actions.

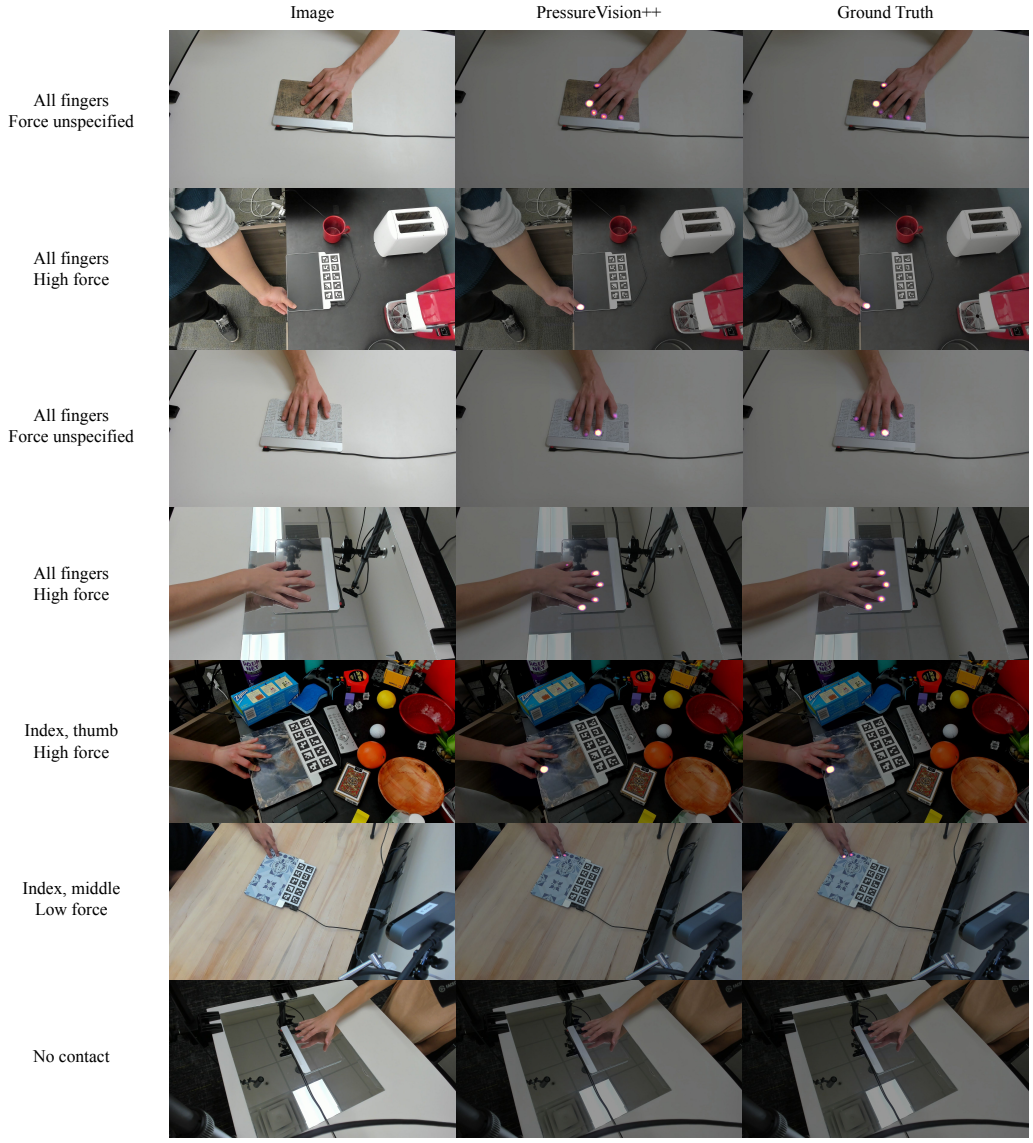

Figure 11. Results from the fully labeled test set where ground truth pressure is measured by a pressure sensor. Testing participants are held out from the training sets.

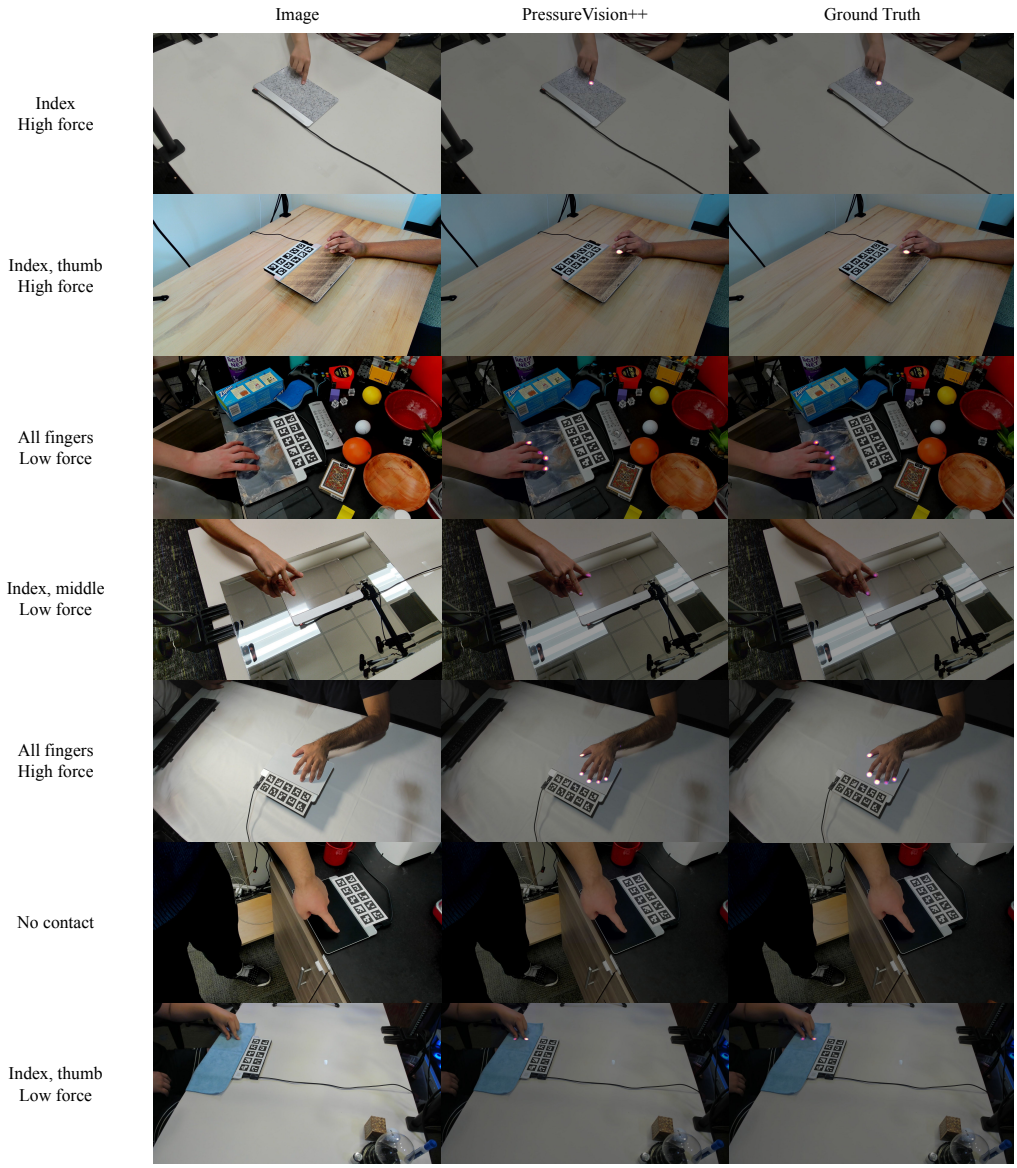

Figure 12. Results from the fully labeled test set where ground truth pressure is measured by a pressure sensor. Testing participants are held out from the training sets.

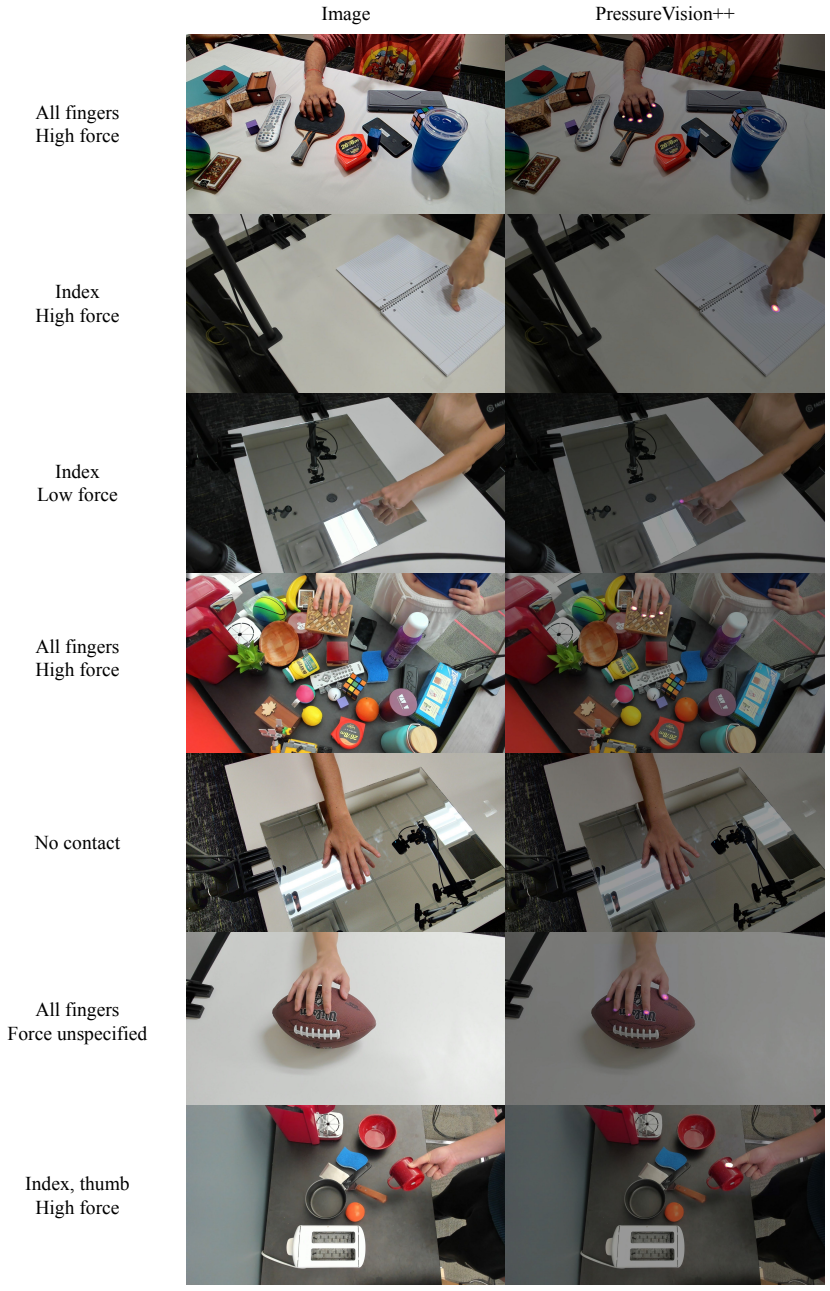

Figure 13. Results from the weakly labeled test set where no ground truth pressure is available. Testing participants are held out from the training sets.

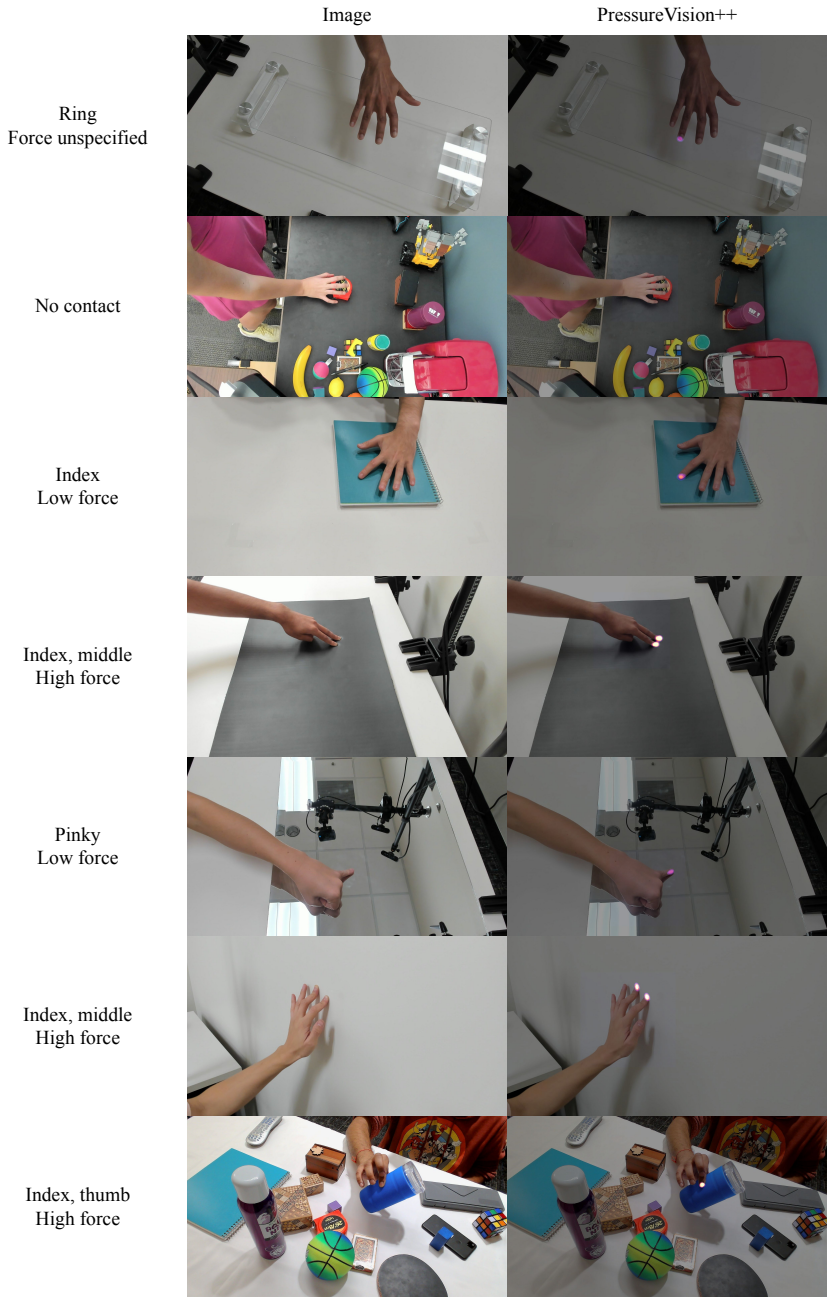

Figure 14. Results from the weakly labeled test set where no ground truth pressure is available. Testing participants are held out from the training sets.