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

Figure 1. PressureVision++ leverages *contact labels* to estimate fingertip pressure during interaction with diverse surfaces. Our method enables mixed reality devices to use everyday surfaces as touch-sensitive interfaces, as demonstrated with a surface drawing application.

Abstract

Touch plays a fundamental role in manipulation for humans; however, machine perception of contact and pressure typically requires invasive sensors. Recent research has shown that deep models can estimate hand pressure based on a single RGB image. However, evaluations have been limited to controlled settings since collecting diverse data with ground-truth pressure measurements is difficult. We present a novel approach that enables diverse data to be captured with only an RGB camera and a cooperative participant. Our key insight is that people can be prompted to apply pressure in a certain way, and this prompt can serve as a weak label to supervise models to perform well under varied conditions. We collect a novel dataset with 51 participants making fingertip contact with diverse objects. Our network, PressureVision++, outperforms human annotators and prior work. We also demonstrate an application of PressureVision++ to mixed reality where pressure estimation allows everyday surfaces to be used as arbitrary touch-sensitive interfaces. Code, data, and models are available online.[1]

1. Introduction

People frequently interact with their surroundings by applying pressure with their hands. Machine perception of hand contact pressure has been used for activity recognition [56], ergonomics [49], user interfaces [51], and other applications. Most approaches use physical pressure sensing arrays. These sensors, however, may be expensive or impractical to mount to hands or natural objects.

Recently, PressureVision [18] showed that computer vision can be used to estimate hand pressure from a single RGB image. As opposed to physical pressure sensors, cameras provide a scalable, low-cost method to sense contact and pressure, opening the door to broad application. While their model performs well with diverse hands, performance was only explored in an idealized, controlled environment. Their capture environment used artificial lighting, high-end machine vision cameras, and was trained and tested on simple, flat, rigid surfaces. Based on their results, it is not evident that vision-based pressure estimation is possible in less constrained settings.

Collecting training data for diverse surfaces might enable PressureVision [18] to generalize more broadly, but data collection is challenging (Figure 2). Each RGB image in the training data requires ground truth pressure from a high-resolution pressure sensor. This requirement severely limits data collection for diverse surfaces, since mounting pressure sensors alters the appearance and properties of the surface. Additionally, human labelers have difficulty identifying contact and pressure from images.

We present a novel approach that enables training data for visual hand pressure estimation to be captured for unal-

[1]https://pressurevision.github.io/

tered surfaces found in the wild. Instead of instrumenting the surface, our approach relies on people's manual dexterity and highly sensitive perception of touch [32]. We prompt participants to make contact with a surface in a specific way or move their hands close to a surface without making contact. The prompts serve as *contact labels*, which are a form of weak label [45]. A contact label consists of the regions of the hand that are in contact with a surface and the level of applied force.

We collect a dataset, ContactLabelDB, which captures 51 participants applying pressure to surfaces with their hands. The dataset contains *fully labeled data*, which captures participants interacting with a pressure sensor. However, the sensor reduces the diversity of data that can be collected. We also capture *weakly labeled data*, which is captured without a pressure sensor but contains greater diversity.

Training our network, PressureVision++, on RGB images paired with contact labels results in higher performance on diverse surfaces, outperforming prior work and generalizing to surfaces that are not represented in the fully labeled training data.

Finally, we demonstrate an application of PressureVision++ to mixed reality. Visual pressure estimation allows using everyday surfaces as touch-sensitive user interfaces. We demonstrate a variety of interfaces, including a touch-sensitive keyboard that allows users to quickly type by touching a table surface. Participants type faster and prefer our keyboard when compared to a commercially released pose-based keyboard included with the Meta Quest 2 headset.

In summary, we make the following contributions:

- We present PressureVision++, a deep model for pressure estimation that leverages contact labels to learn from data with and without ground truth pressure labels.
- We collect ContactLabelDB, a dataset of RGB images with 51 participants interacting with 100+ surfaces.
- We demonstrate the utility of PressureVision++ with applications to mixed reality.
- We release our models, data, and code.

2. Related Work

Physical Sensors for Pressure Sensing: Sensors to measure pressure may be mounted to human hands. Glove-based sensors have been developed by researchers [7, 56] and are commercially available [48, 58]. However, sensors mounted to the hand are expensive, interfere with tactile perception, and impact manual dexterity. Further, gloves occlude the surface of the hand, which interferes with data collection for visual models intended for bare hands.

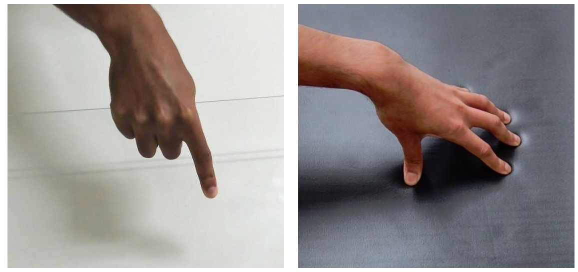

Figure 2. Instrumenting surfaces with pressure sensors without altering their properties is challenging. For example, pressure sensors must be transparent in order to instrument glass, and must be stretchable in order to instrument a deformable mat.

Various types of pressure sensors have been developed which can be mounted on objects, including capacitive sensors [2, 12, 20], force-sensitive resistors [6, 44, 47], flexible sensors [4, 34, 58], and fabric-based sensors [39]. However, even flexible pressure sensors have difficulty in conforming to the complex geometry of everyday objects. Mounting pressure sensors to objects also fundamentally alters their visual appearance and mechanical properties, reducing the diversity of data that can be captured.

Visual Hand Pressure Estimation: The net forces applied by a hand to a known object can be estimated by observing the object's pose over time and calculating the forces that would result in this trajectory [14, 36, 46, 47, 50]. These methods can infer contact that is occluded or out-of-view, but they fail for static objects like tabletops. Contact estimates based on mesh geometry are also sensitive to precision since contact depends on millimeter scale displacements [19].

A number of approaches have demonstrated that visual cues can be used to estimate hand pressure, including fingertip color changes [10, 40, 41], soft tissue deformation [31], and cast shadows [27, 28, 30]. In contrast to this prior work, our method uses an external camera to view the whole hand from a distance and deep learning to take advantage of multiple types of cues. Our method builds on PressureVision [18].

Deep Learning with Weak Labels: In cases where full labels are not available, approaches have been developed to still use partially or *weakly* labeled data. Prior work has used semantic segmentation as a motivating task, where generating per-pixel labels requires significant time from human annotators [13]. Techniques have been developed to leverage faster annotations, including image-level labels [1, 8, 35] and point labels [3]. Most similar to our paper

Dataset	Modality	Frames	Partici-pants	Objects / Surfaces	Contact	Pressure	Pose	Whole-Hand	Natural Objects
OakInk [62]	RGBD	230k	12	100	Inferred from pose	×	✓	✓	✓
DexYCB [9]	RGBD	582k	10	20	Inferred from pose	×	✓	✓	✓
HO-3D [22]	RGBD	78k	10	10	Inferred from pose	×	✓	✓	✓
GRAB [57]	Pose	1.6M	10	51	Inferred from pose	×	✓	✓	×
ContactPose [5]	RGBD	3.0M	50	25	Thermal imprint	×	✓	✓	×
PressureVisionDB [18]	RGB	3.0M	36	2	Pressure sensor	✓	×	✓	×
ContactLabelDB (ours)	RGB	2.9M	51	106	Pressure sensor	✓	×	×	✓

Table 1. Several hand/object datasets infer contact from pose. However, this requires accurate pose tracking and hand/object models, limiting the quality of the inferred contact. ContactPose [5] captures the heat left by hands grasping objects. ContactLabelDB features ground truth pressure measurements, a large number of participants, and interaction with natural objects, but only captures fingertip contact.

is work that leverages image-level labels and an adversarial loss to transfer segmentation models to new domains [45]. In contrast to weak labels applied by human annotators after data collection, our method prompts human behavior while data is being collected.

Hands in Mixed Reality: Modern mixed reality devices increasingly rely on hand tracking as an input modality. Commercially available devices use monochrome cameras [23, 24] and depth sensors [21] to estimate 3D hand pose.

In order to sense contact with the environment, a variety of hand-mounted physical sensors have been proposed, including IMUs [43, 54], electrical current injection [33, 63], and acoustic sensing [16]. Most similar to our work is research in sensing contact between fingers and flat surfaces from depth cameras [17, 53, 59]. However, depth cameras add cost, draw high power, and may not work on reflective surfaces or in brightly illuminated scenes. Time-of-flight cameras may have error on materials such as human skin [26]. In comparison, our work senses pressure from only RGB cameras, which are non-invasive and low-cost.

3. Data Collection

This section describes the capture of ContactLabelDB, a dataset of 51 participants making contact with diverse surfaces and objects.

3.1. Contact Labels

During all data capture sequences, the participant is prompted to make contact with a surface using a specific combination of fingertips and to achieve a target force level, for example: "press ring finger at a low force". The participant performs the requested action while data is collected. This prompt has a one-to-one correspondence with a contact label (Figure 3).

As shown in Figure 3, a contact label W is represented as a vector with 6 elements. The first 5 elements indicate the presence or lack of contact at each of the 5 fingertips. The sixth element indicates if the participant was prompted to exert a low, high, or unspecified force with the fingers in contact.

We represent a contact label $W \in \mathbb{Z}^6$ as follows:

$$w_i|_{0\leq i\leq 4} \in \{0,1\} \equiv \{\textit{no contact, contact}\}$$
$$w_i|_{i=5} \in \{-1,0,1\} \equiv \{\textit{unspecified, low, high force}\}$$

For all data collection procedures, we prompted participants to press one of eight combinations of fingertips onto a surface. For each combination, we prompted the participant to apply a low force, a high force, or to slide with unspecified force. Additionally, we prompted participants to make "no contact" by hovering the specified fingertips just above the surface.

While PressureVisionDB [18] captured contact with the entire hand including the palm, in this work, we only capture fingertip contact. This decision was made since prompting contact with other parts of the hand is complex, and fingertips are sufficient for many downstream tasks. During pilot studies, we also found that participants apply similar pressures between fingertips unless explicitly prompted not to.

3.2. Collection Method

We collect two types of data: *fully labeled* data and *weakly labeled* data. For both types of sequences, we collect contact labels by prompting the subject with a specific instruction. For the fully labeled sequences, we additionally collect ground truth pressure labels using a high-resolution pressure sensing array [44] (Figure 3).

When data is collected with a ground truth pressure sensor, participants press and release their hand multiple times on the surface to capture the onset and termination of contact. Frames with no pressure detected are assigned a "no-contact" label. When data is collected without a pressure sensor, the participant maintains contact throughout the duration of the recording.

3.3. Data Splits

Our training data comes from 37 participants, while our testing data comes from 14 participants who are not present

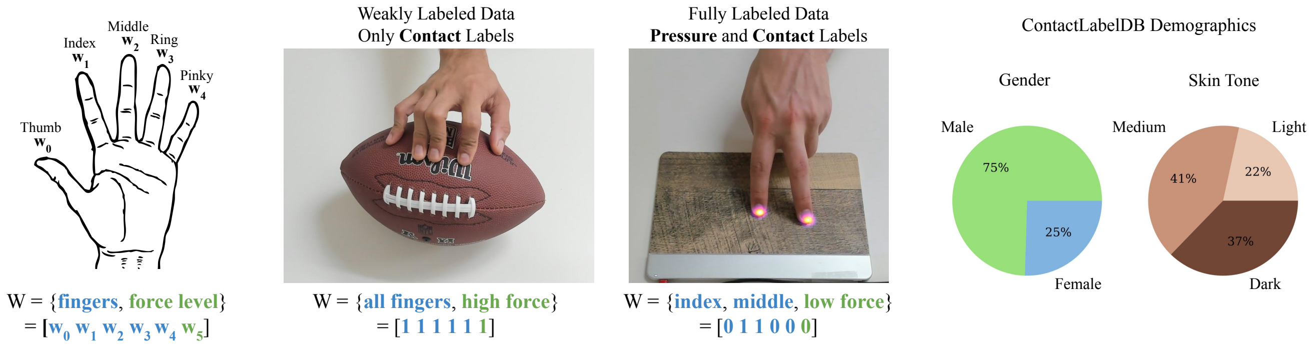

Figure 3. We represent the contact labels as a six-dimensional vector. The first five elements are binary values indicating which fingers are prompted to be in contact, and the last element indicates the prompted force level. Fully labeled data has *both* pressure and contact labels, while weakly labeled data has *only* contact labels. Participants with a range of genders and skin tones were recruited for our study.

in the training data. Since weakly labeled data collection does not require the pressure sensor, a much greater diversity of data can be collected. We collect a total of 2.9M frames: 0.5M fully labeled frames and 2.4M weakly labeled frames.

Our training data consists of a **fully labeled training set** where participants interacted with solid-colored overlays on a pressure sensor. We also collect a **weakly labeled training set** where participants interacted with diverse surfaces and natural objects instead of a pressure sensor.

We desire to evaluate our approach on interactions with the natural world. However, we face the same problem as during training data collection: it is difficult to collect ground truth pressure in diverse environments. We collect a **fully labeled testing set** where participants interact with textured overlays not seen in the training set on a pressure sensor. We also collect a **weakly labeled testing set** where participants interact with diverse surfaces and natural objects, many of which were not present in the training set.

Images were captured with multiple consumer-grade webcams at 30 FPS and 1080p resolution. We conducted data collection in 20 environments with different lighting conditions. We used a Sensel Morph [44] pressure sensing array.

3.4. Ethics

Approval to conduct this study was obtained from an Institutional Review Board (IRB). We recruited a diverse set of 51 participants (Figure 3). All participants gave informed consent and were compensated for their time. We measured skin tone with a Pantone X-Rite RM200 spectrocolorimeter, and participants self-reported gender.

4. Network Architecture

We create a network, PressureVision++, (Figure 4) to take a single RGB image, I, as input and output a pressure image, $\hat{P} = f(I)$. For fully labeled data, each RGB image is paired with a ground-truth pressure image obtained by projecting the output of a pressure-sensing array into the image using a homography transform. The output pressure $\hat{P}$ is in *image space*, such that the input and output images are the same shape and can be superimposed (Figure 5).

4.1. Pressure Estimation

To estimate pressure, PressureVision++ uses a binned representation and performs classification across bins. The pressure range is split into $N_B = 9$ logarithmically spaced bins divided across the pressure range, including one zero bin. Pressure estimation uses a *structure-aware cross-entropy* loss L_p [42, 55]. Unlike regular cross-entropy, the structure-aware loss penalizes large errors more than small errors. For each pressure pixel over the image x, y, the loss is computed over all bin indices $b \in B$ using the ground truth index k_b and the estimated probability for each bin $\rho_{x,y}(b)$.

$$L_p = -\sum_{x,y}\sum_{b} e^{-|b-k_b|} log(\rho_{x,y}(b)) \tag{1}$$

L_p is only computed when fully labeled data is available.

4.2. Contact Label Estimation

In addition to estimating a pressure image, PressureVision++ performs the auxiliary task of estimating the contact label $\hat{W}$. The contact label classifier predicts $\hat{W}$ given the features F at the network bottleneck (Figure 4). The addition of the contact label classifier ensures that the features generated by the encoder are discriminative to the set of fingers in contact and the force level. The classifier pools features and uses a 2-layer MLP to estimate the contact label collected in Section 3.2. This classifier is trained with a binary cross-entropy loss L_w. When the amount of force is not specified in the ground truth contact label, this portion of the loss is masked out.

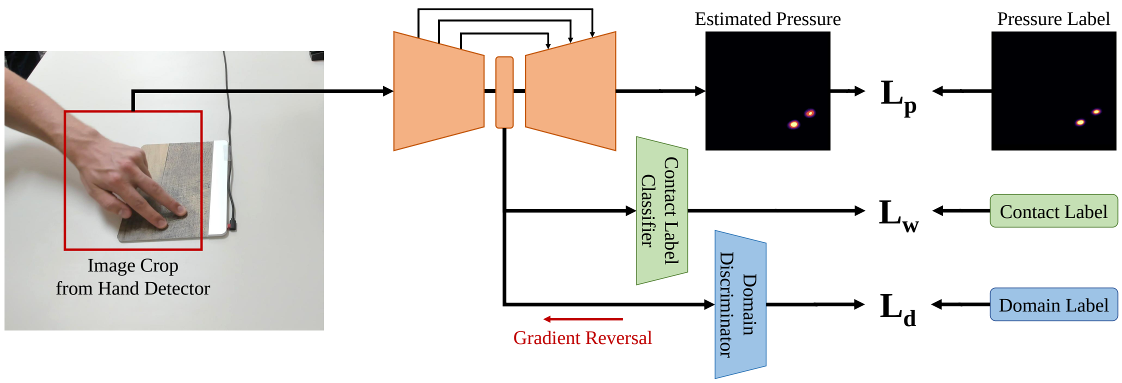

Figure 4. PressureVision++ architecture. First, hand crops are generated using the bounding boxes estimated by an off-the-shelf hand detector. The crops are passed into an encoder-decoder network to estimate pressure for each pixel in the input image. Two classification heads are attached to the bottleneck of the network; one is trained to estimate the contact label, and the other uses an adversarial loss to reduce the shift between fully labeled and weakly labeled domains.

4.3. Adversarial Domain Adaptation

Following prior work in domain adaptation [15], we apply an additional feature alignment loss using a domain discriminator D. This loss is *unsupervised*, as it does not leverage contact label information. This loss attempts to minimize the distance between the distributions of features generated from two domains. The discriminator estimates if the image is from the fully labeled or weakly labeled domain, and when backpropagating, gradients are reversed upstream of the domain discriminator [15]. For image features from the fully labeled domain F_f and weakly labeled domain F_w, the domain loss function L_d is:

$$L_d = -log(D(F_f)) - log(1 - D(F_w)) \tag{2}$$

4.4. Training Details

As hands often only take up a small part of the image, PressureVision++ operates on crops of the hand. We use MediaPipe [38] to produce bounding boxes which are used to generate hand crops. Hand crops are resized to 448x448 pixels before being sent to PressureVision++.

PressureVision++ uses an SE-ResNeXt-50 encoder [25, 29, 60] and an FPN decoder [37, 61], and is trained end-to-end using the following loss function:

$$L = L_p + \lambda_1 L_w + \lambda_2 L_d \tag{3}$$

5. Evaluation

We consider two types of evaluations: *contact* and *pressure* evaluations, following prior work [18]. Contact is a binary quantity indicating if the hand and object are touching, while pressure is a scalar indicating the magnitude of force. A binary contact image $\hat{C}$ is generated by thresholding each pressure pixel in $\hat{P}$ at $P_{th} = 1\ kPa$.

- **Contact Accuracy:** The estimated contact image $\hat{C}$ is used to determine if *any* contact is estimated. Accuracy is calculated by counting the percentage of video frames for which $\hat{C}$ corresponds with the contact label.
- **Contact IoU:** Intersection-over-union (IoU) is computed between the ground truth contact image C and estimated contact image $\hat{C}$.
- **Volumetric IoU:** An extension of Contact IoU that considers the *magnitude* of pressure. 2D pressure images are viewed as 3D pressure volumes, where the height of the volume is equal to the magnitude of pressure at that pixel. Intersection-over-union is computed using these volumes.

For the same reasons that collecting fully labeled training data on diverse surfaces is difficult, collecting fully labeled testing data also presents challenges. We evaluate both the *fully labeled* and *weakly labeled* test sets. However, due to the lack of pressure measurements in the weakly labeled test set, only contact accuracy is computed. For more details and evaluations, refer to the supplementary material.

5.1. Performance Compared to Baselines

We compare our method against three baselines as shown in Table 2.

- **Zero Guesser:** The zero guesser outputs a zero-pressure image and provides a reference for Contact Accuracy due to the large number of frames with no contact.
- **Human Annotator:** Annotators from Amazon Mechanical Turk perform a binary classification on 10,000 images to determine if any part of the hand is touching the surface.

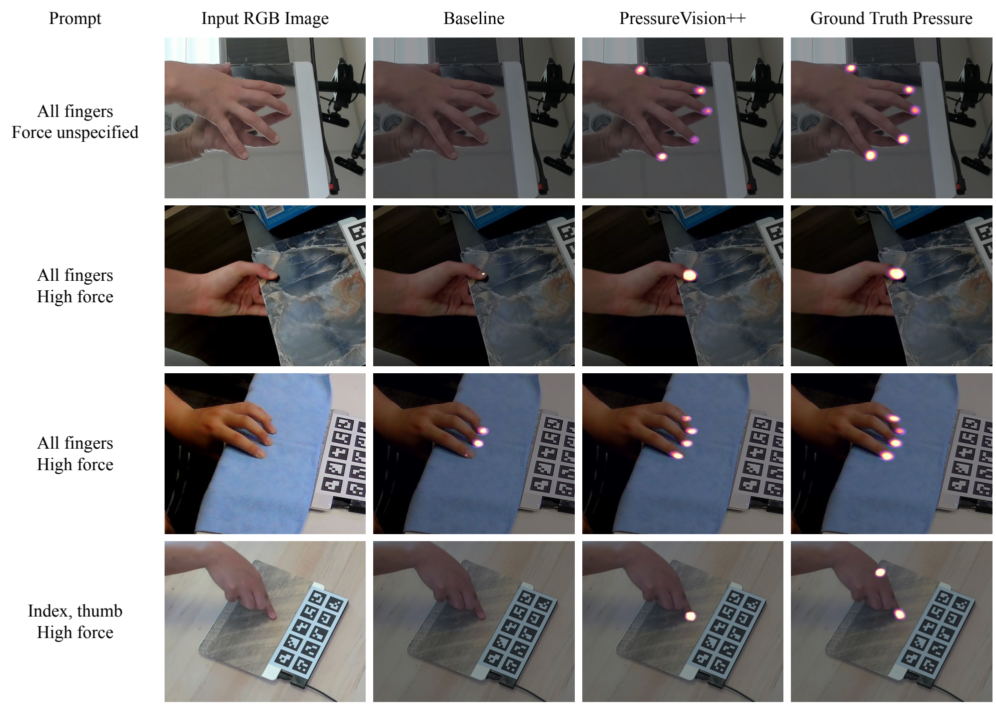

Figure 5. Results on the fully labeled test set. The baseline column is PressureVision++ trained without either the domain loss or contact label loss. The bottom row shows a common failure mode where pressure is not estimated for occluded parts of the hand.

	Fully Labeled Test Set			Weakly Labeled Test Set
Method	**Contact Acc.**	**Contact IoU**	**Volumetric IoU**	**Contact Acc.**
Zero Guesser	53.4%	0.0%	0.0%	24.9%
Human Annotator	78.4%	-	-	**80.5%**
PressureVision [18]	72.7%	15.2%	11.3%	53.5%
PressureVision++ (ours)	**89.3%**	**41.9%**	**27.5%**	**80.5%**

Table 2. Performance compared to a PressureVision baseline [18] and human annotators.

- **PressureVision:** The network from [18] is retrained on our fully labeled data. This method does not use contact labels.

PressureVision++ significantly outperforms prior work, improving on all metrics. Examples from the fully labeled test set are shown in Figure 5, and examples from the weakly labeled test set are shown in Figure 6. PressureVision++ estimates fingertip pressure on diverse surfaces, including textured, deformable, and curved surfaces. Our method adapts to unseen surfaces in the test set by leveraging diverse weakly labeled data. We observe one common failure mode where pressure is not estimated for occluded fingertips (Figure 5).

Human-Annotated Contact: We investigated the possibility of using non-expert human labelers recruited from Amazon Mechanical Turk to identify contact from images. In this evaluation, workers performed a binary classification on 10,000 frames to identify whether the hand is in contact or not. This is an easier task than generating contact labels or estimating pressure.

We observe that non-expert annotators had difficulty distinguishing near-contact from contact (Table 2). When contact accuracy is computed, we observe that PressureVision++ outperforms the human annotators on the fully labeled test set, and performs similarly on the weakly labeled test set. This result suggests that our approach may exceed human performance under ideal conditions, but is not as robust when tested on more diverse data.

5.2. Ablating Weakly Labeled Data

Table 3 illustrates the impact of weakly labeled data on PressureVision++'s performance. With neither the domain loss nor contact label loss, the weakly labeled dataset is unused. With only the domain loss L_d, the weakly labeled

Figure 6. Results on the surfaces in the weakly labeled test set, *none* of which are included in the fully labeled training set. PressureVision++ produces qualitatively accurate results on highly textured, curved, and compliant surfaces. All images except the top row are zoomed in to show detail. The bottom right shows a failure case where pressure is underestimated on an object.

L_d	L_w	**Volumetric IoU**
		14.9%
✓		17.5%
	✓	25.5%
✓	✓	**27.5%**

Table 3. The domain loss L_d and contact label loss L_w enable training on weakly labeled data which improves performance significantly.

dataset is used in an unsupervised way. Finally, the contact label loss L_w leverages the contact labels collected. We find that both losses significantly contribute to performance. However, the contact label loss has the largest effect, with this alone improving volumetric IoU by +71%.

This large performance improvement demonstrates the value of contact labels for hand pressure estimation. Weakly labeled data is easy to collect, yet significantly increases performance on diverse surfaces.

6. Applications in Mixed Reality

Modern mixed reality devices increasingly rely on hand tracking as a primary input modality. The Meta Quest and Microsoft HoloLens product lines use 3D pose estimation to allow hands to interact with mid-air interfaces. However, mid-air interfaces are fatiguing, and virtual objects do not provide tactile feedback. A study by Cheng *et al*. [11] compared mid-air interfaces to tabletop interfaces. Participants interacting on the tabletop were more accurate and reported less exertion and improved comfort.

PressureVision++ presents a natural way to extend hand tracking to detect interactions with surfaces. Our system only requires a low-cost, externally mounted RGB camera and may enable more accurate, lower-exertion input.

6.1. Hardware Setup

We use a Meta Quest 2 headset for our demos. The Quest operates in passthrough mode which allows users to "see through" the headset by using the device's onboard cameras to provide a mixed reality experience. However, we do not use these onboard cameras for pressure estimation.

We mount a single RGB camera to a table (Figure 1) and run PressureVision++ on a desktop computer. The pressure estimation system runs at approximately 50 FPS with an RTX 3090 GPU, which includes first running the hand detector [38] and then running PressureVision++ once for each detected hand. Pressure information is sent over WiFi to the headset. See the supplementary materials for more details.

6.2. Touch-Sensitive Interfaces

PressureVision++ enables the creation of touch-sensitive user interfaces which can be attached to tabletops, walls,

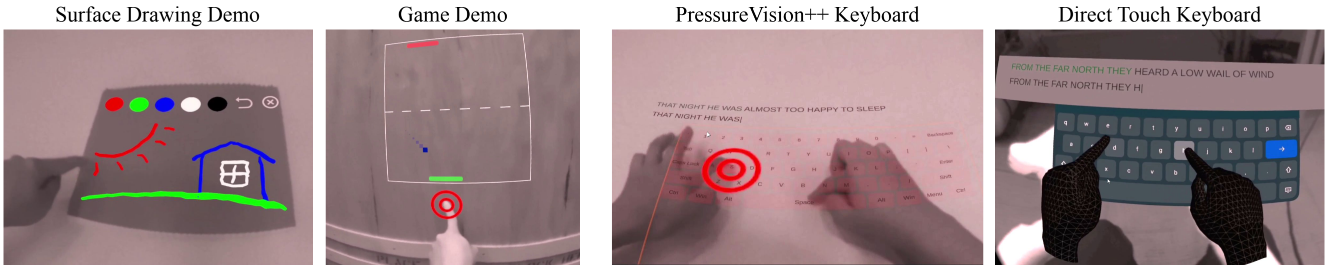

Figure 7. PressureVision++ allows touch-sensitive interfaces to be placed on objects. We show drawing on the back of a notebook and playing a game on a vertical wooden surface. The PressureVision++ Keyboard allows users to type by touching keys on a tabletop. This is compared to the Direct Touch Keyboard which uses 3D hand poses to allow users to press keys of a mid-air keyboard.

or objects in the environment (Figure 7). We developed a pressure-sensitive drawing application that allows users to paint on a surface simply by dragging their finger. The amount of pressure they apply controls the width of the brush stroke. Sample drawings are shown in Figures 1 and 7. We also design a game similar to Pong where the user controls the paddle position by dragging their finger and show a user interacting with this game on a vertical wooden surface.

6.3. Touch Typing

To demonstrate the accuracy and responsiveness of our approach, we evaluate PressureVision++ on a typing task. We develop a touch-sensitive keyboard using our system which allows users to tap on a tabletop to enter text.

PressureVision++ Keyboard: The headset projects a keyboard layout onto a tabletop. When the system detects contact, a red marker is drawn (Figure 7) to aid the user. We implement a debouncing filter that requires 2 subsequent frames to trigger the onset or termination of a keypress, reducing spurious keystrokes.

Baseline Keyboard: The Quest 2 headset performs 3D hand pose estimation using its four monochrome cameras [24]. The Meta Direct Touch Keyboard is a text entry method that places a floating keyboard in front of the user and allows users to press keys using their index fingers (Figure 7). As this keyboard is built-in to the Quest 2 operating system, we use it as a baseline.

User Study: To evaluate the typing methods, we recruited 10 participants, none of whom participated in the collection of ContactLabelDB. The two typing methods were presented in random order. Participants attempted to quickly and accurately type a prompt sentence. Typing speed was calculated in net words per minute [52]. After completing the typing test, participants gave open-ended responses comparing the two systems. They also selected which keyboard they preferred.

Results: We found that participants typed 78% faster and 9 of 10 participants favored the PressureVision++ Keyboard over the Direct Touch Keyboard (Table 4). In their open-ended responses, 7 participants mentioned that they found the tactile feedback of interacting with a real surface helpful, and 5 mentioned that they found the PressureVision++ keyboard less tiring to use.

Typing Method	Net WPM	Preferred
Direct Touch	14.4	1/10
PressureVision++ (ours)	**25.8**	**9/10**

Table 4. Participants type faster with the PressureVision++ Keyboard as compared to the Direct Touch Keyboard. After trying both, 9 of 10 participants preferred our method.

Overall, these results suggest that surface interactions enabled by PressureVision++ have advantages over the mid-air interfaces enabled by pose estimators.

Limitations: We observe that during 5-finger typing, fingertips are often occluded and users make hand poses that are not represented in our dataset. As a result, we instructed participants to only type with their index fingers, which are more reliably detected. Additionally, compared to the Direct Touch Keyboard, our approach is not limited to the monochrome egocentric cameras and the mobile processor that the headset uses for hand pose estimation. However, these limitations may be overcome in future work.

7. Conclusion

Training deep models to visually estimate the pressure applied by fingertips relies on ground-truth pressure measurements that are difficult to obtain. We presented PressureVision++ which uses more easily obtained contact labels collected by prompting participants to achieve specific types of contact. Leveraging this weakly supervised data improves pressure estimation on diverse surfaces and outperforms prior methods. PressureVision++ additionally enables interactions with natural surfaces in mixed reality.

References

[1] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4981–4990, 2018. 2

[2] Karan Ahuja, Paul Streli, and Christian Holz. TouchPose: Hand pose prediction, depth estimation, and touch classification from capacitive images. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pages 997–1009, 2021. 2

[3] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *European Conference on Computer Vision (ECCV)*, pages 549–565. Springer, 2016. 2

[4] Raunaq Bhirangi, Tess Hellebrekers, Carmel Majidi, and Abhinav Gupta. Reskin: versatile, replaceable, lasting tactile skins. *Conference on Robot Learning (CoRL)*, 2021. 2

[5] Samarth Brahmbhatt, Chengcheng Tang, Christopher D. Twigg, Charles C. Kemp, and James Hays. ContactPose: A dataset of grasps with object contact and hand pose. In *European Conference on Computer Vision (ECCV)*, August 2020. 3

[6] Samarth Manoj Brahmbhatt. *Grasp contact between hand and object: Capture, analysis, and applications*. PhD thesis, Georgia Institute of Technology, 2020. 2

[7] Gereon H Büscher, Risto Kõiva, Carsten Schürmann, Robert Haschke, and Helge J Ritter. Flexible and stretchable fabric-based tactile sensor. *Robotics and Autonomous Systems*, 63:244–252, 2015. 2

[8] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8991–9000, 2020. 2

[9] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, et al. Dexycb: A benchmark for capturing hand grasping of objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9044–9053, 2021. 3

[10] Nutan Chen, Göran Westling, Benoni B Edin, and Patrick van der Smagt. Estimating fingertip forces, torques, and local curvatures from fingernail images. *Robotica*, 38(7):1242–1262, 2020. 2

[11] Yi Fei Cheng, Tiffany Luong, Andreas Rene Fender, Paul Streli, and Christian Holz. Comfortable user interfaces: Surfaces reduce input error, time, and exertion for tabletop and mid-air user interfaces. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 150–159. IEEE, 2022. 7

[12] Frederick Choi, Sven Mayer, and Chris Harrison. 3d hand pose estimation on conventional capacitive touchscreens. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction*, MobileHCI '21, New York, NY, USA, 2021. Association for Computing Machinery. 2

[13] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. 2

[14] Kiana Ehsani, Shubham Tulsiani, Saurabh Gupta, Ali Farhadi, and Abhinav Gupta. Use the force, Luke! Learning to predict physical forces by simulating effects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 224–233, 2020. 2

[15] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016. 5

[16] Jun Gong, Aakar Gupta, and Hrvoje Benko. Acustico: surface tap detection and localization using wrist-based acoustic tdoa sensing. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pages 406–419, 2020. 3

[17] Norberto Adrian Goussies, Kenji Hata, Shruthi Prabhakara, Abhishek Amit, Tony Aube, Carl Cepress, Diana Chang, Li-Te Cheng, Horia Stefan Ciurdar, Mike Cleron, et al. Learning to detect touches on cluttered tables. *arXiv preprint arXiv:2304.04687*, 2023. 3

[18] 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. 1, 2, 3, 5, 6

[19] Patrick Grady, Chengcheng Tang, Christopher D. Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C. Kemp. ContactOpt: Optimizing contact to improve grasps. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1471–1481, 2021. 2

[20] Anhong Guo, Robert Xiao, and Chris Harrison. Capauth: Identifying and differentiating user handprints on commodity capacitive touchscreens. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces*, pages 59–62, 2015. 2

[21] Hung-Jui Guo and Balakrishnan Prabhakaran. Hololens 2 technical evaluation as mixed reality guide. *arXiv preprint arXiv:2207.09554*, 2022. 3

[22] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. HOnnotate: A method for 3D annotation of hand and object poses. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3196–3206, 2020. 3

[23] Shangchen Han, Beibei Liu, Randi Cabezas, Christopher D. Twigg, Peizhao Zhang, Jeff Petkau, Tsz-Ho Yu, Chun-Jung Tai, Muzaffer Akbay, Zheng Wang, Asaf Nitzan, Gang Dong, Yuting Ye, Lingling Tao, Chengde Wan, and Robert Wang. Megatrack: monochrome egocentric articulated hand-tracking for virtual reality. *ACM Transactions on Graphics (ToG)*, 39:87, 2020. 3

[24] Shangchen Han, Po-chen Wu, Yubo Zhang, Beibei Liu, Linguang Zhang, Zheng Wang, Weiguang Si, Peizhao Zhang,

Yujun Cai, Tomas Hodan, et al. Umetrack: Unified multi-view end-to-end hand tracking for vr. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 3, 8

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 5

[26] Ying He, Bin Liang, Yu Zou, Jin He, and Jun Yang. Depth errors analysis and correction for time-of-flight (tof) cameras. *Sensors*, 17(1):92, 2017. 3

[27] Helen H Hu, Amy A Gooch, Sarah H Creem-Regehr, and William B Thompson. Visual cues for perceiving distances from objects to surfaces. *Presence: Teleoperators & Virtual Environments*, 11(6):652–664, 2002. 2

[28] Helen H Hu, Amy A Gooch, William B Thompson, Brian E Smits, John J Rieser, and Peter Shirley. Visual cues for imminent object contact in realistic virtual environments. In *Proceedings Visualization 2000.*, pages 179–185. IEEE, 2000. 2

[29] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7132–7141, 2018. 5

[30] Geoffrey S. Hubona, Philip N. Wheeler, Gregory W. Shirah, and Matthew Brandt. The relative contributions of stereo, lighting, and background scenes in promoting 3d depth visualization. *ACM Transactions on Computer Humam Interaction*, 6(3):214–242, 1999. 2

[31] Wonjun Hwang and Soo-Chul Lim. Inferring interaction force from visual information without using physical force sensors. *Sensors*, 17(11):2455, 2017. 2

[32] Roland S. Johansson and J. Randall Flanagan. Coding and use of tactile signals from the fingertips in object manipulation tasks. *Nature Reviews Neuroscience*, 10:345—-359, 2009. 2

[33] Wolf Kienzle, Eric Whitmire, Chris Rittaler, and Hrvoje Benko. Electroring: Subtle pinch and touch detection with a ring. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2021. 3

[34] Hong-Ki Kim, Seunggun Lee, and Kwang-Seok Yun. Capacitive tactile sensor array for touch screen application. *Sensors and Actuators A: Physical*, 165(1):2–7, 2011. 2

[35] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *European Conference on Computer Vision (ECCV)*, pages 695–711. Springer, 2016. 2

[36] Zongmian Li, Jirí Sedlár, Justin Carpentier, Ivan Laptev, Nicolas Mansard, and Josef Sivic. Estimating 3d motion and forces of person-object interactions from monocular video. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8640–8649, 2019. 2

[37] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2117–2125, 2017. 5

[38] 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. 5, 7

[39] Yiyue Luo, Yunzhu Li, Pratyusha Sharma, Wan Shou, Kui Wu, Michael Foshey, Beichen Li, Tomás Palacios, Antonio Torralba, and Wojciech Matusik. Learning human–environment interactions using conformal tactile textiles. *Nature Electronics*, 4(3):193–201, 2021. 2

[40] Stephen A Mascaro and H Harry Asada. Photoplethysmograph fingernail sensors for measuring finger forces without haptic obstruction. *IEEE Transactions on Robotics and Automation*, 17(5):698–708, 2001. 2

[41] Stephen A. Mascaro and H. Harry Asada. Measurement of finger posture and three-axis fingertip touch force using fingernail sensors. *IEEE Transactions on Robotics and Automation*, 20(1):26–35, 2004. 2

[42] Francisco Massa, Renaud Marlet, and Mathieu Aubry. Crafting a multi-task CNN for viewpoint estimation. *British Machine Vision Conference (BMVC)*, 2016. 4

[43] Manuel Meier, Paul Streli, Andreas Fender, and Christian Holz. Tapid: Rapid touch interaction in virtual reality using wearable sensing. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pages 519–528. IEEE, 2021. 3

[44] Morph. Sensel Morph haptic sensing tablet. `www.sensel.com/pages/the-sensel-morph`, Last accessed on 2020-02-25. 2, 3, 4

[45] Sujoy Paul, Yi-Hsuan Tsai, Samuel Schulter, Amit K Roy-Chowdhury, and Manmohan Chandraker. Domain adaptive semantic segmentation using weak labels. In *European Conference on Computer Vision (ECCV)*, pages 571–587. Springer, 2020. 2, 3

[46] Tu-Hoa Pham, Abderrahmane Kheddar, Ammar Qammaz, and Antonis A. Argyros. Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2810–2819, 2015. 2

[47] Tu-Hoa Pham, Nikolaos Kyriazis, Antonis A. Argyros, and Abderrahmane Kheddar. Hand-object contact force estimation from markerless visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2883–2896, 2017. 2

[48] Pressure Profile Systems. PPS TactileGlove. 2

[49] M Reinvee, K Jansen, et al. Utilisation of tactile sensors in ergonomic assessment of hand–handle interface: a review. *Agron. Res*, 12(3):907–914, 2014. 1

[50] Grégory Rogez, James S Supancic, and Deva Ramanan. Understanding everyday hands in action from RGB-D images. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3889–3897, 2015. 2

[51] Jung-Sim Roh, Yotam Mann, Adrian Freed, David Wessel, UC Berkeley, A Freed, A Street, and D Wessel. Robust and reliable fabric, piezoresistive multitouch sensing surfaces for musical controllers. In *Conference on New Interfaces for Musical Expression*, pages 393–398, 2011. 1

[52] Timothy A Salthouse. Effects of age and skill in typing. *Journal of Experimental Psychology: General*, 113(3):345, 1984. 8

[53] Vivian Shen, James Spann, and Chris Harrison. Farout touch: Extending the range of ad hoc touch sensing with depth cameras. In *Proceedings of the 2021 ACM Symposium on Spatial User Interaction*, pages 1–12, 2021. 3

[54] Yilei Shi, Haimo Zhang, Kaixing Zhao, Jiashuo Cao, Mengmeng Sun, and Suranga Nanayakkara. Ready, steady, touch! sensing physical contact with a finger-mounted imu. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(2):1–25, 2020. 3

[55] Hao Su, Charles R Qi, Yangyan Li, and Leonidas J Guibas. Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3d model views. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2686–2694, 2015. 4

[56] Subramanian Sundaram, Petr Kellnhofer, Yunzhu Li, Jun-Yan Zhu, Antonio Torralba, and Wojciech Matusik. Learning the signatures of the human grasp using a scalable tactile glove. *Nature*, 569(7758):698–702, 2019. 1, 2

[57] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[58] Tekscan. Pressure mapping, force measurement and tactile sensors. www.tekscan.com/products-solutions/sensorsLast, Last accessed on 2020-02-25. 2

[59] Robert Xiao, Julia Schwarz, Nick Throm, Andrew D. Wilson, and Hrvoje Benko. MRTouch: adding touch input to head-mounted mixed reality. *IEEE Transactions on Visualization and Computer Graphics*, 24(4):1653–1660, 2018. 3

[60] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5987–5995, 2017. 5

[61] Pavel Yakubovskiy. Segmentation models Pytorch, 2020. 5

[62] Lixin Yang, Kailin Li, Xinyu Zhan, Fei Wu, Anran Xu, Liu Liu, and Cewu Lu. Oakink: A large-scale knowledge repository for understanding hand-object interaction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20953–20962, 2022. 3

[63] Yang Zhang, Wolf Kienzle, Yanjun Ma, Shiu S Ng, Hrvoje Benko, and Chris Harrison. Actitouch: Robust touch detection for on-skin ar/vr interfaces. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, pages 1151–1159, 2019. 3