Interactive Hand Pose Estimation using a Stretch-Sensing Soft Glove

OLIVER GLAUSER and SHIHAO WU, ETH Zurich, Switzerland
DANIELE PANOZZO, New York University, USA
OTMAR HILLIGES and OLGA SORKINE-HORNUNG, ETH Zurich, Switzerland

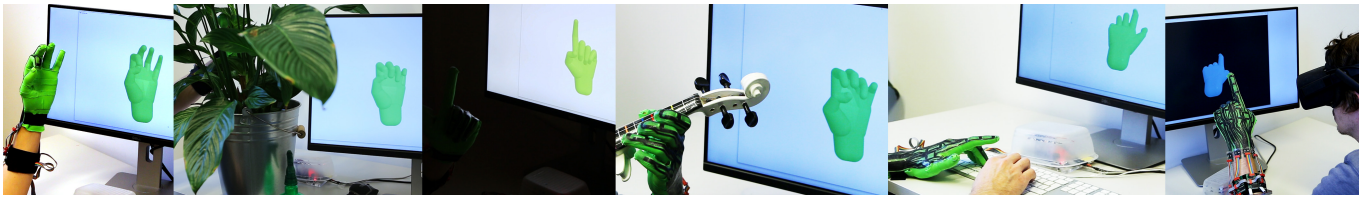

Fig. 1. Our stretch-sensing soft glove captures hand poses in real time and with high accuracy. It functions in diverse and challenging settings, like heavily occluded environments or changing light conditions, and lends itself to various applications. All images shown here are frames from recorded live sessions.

We propose a stretch-sensing soft glove to interactively capture hand poses with high accuracy and without requiring an external optical setup. We demonstrate how our device can be fabricated and calibrated at low cost, using simple tools available in most fabrication labs. To reconstruct the pose from the capacitive sensors embedded in the glove, we propose a deep network architecture that exploits the spatial layout of the sensor itself. The network is trained only once, using an inexpensive off-the-shelf hand pose reconstruction system to gather the training data. The per-user calibration is then performed on-the-fly using only the glove. The glove's capabilities are demonstrated in a series of ablative experiments, exploring different models and calibration methods. Comparing against commercial data gloves, we achieve a 35% improvement in reconstruction accuracy.

CCS Concepts: • **Human-centered computing** → **Interaction devices**; • **Computing methodologies** → **Motion capture**; *Machine learning*; Computer graphics.

Additional Key Words and Phrases: hand tracking, data glove, sensor array, stretch-sensing

ACM Reference Format:
Oliver Glauser, Shihao Wu, Daniele Panozzo, Otmar Hilliges, and Olga Sorkine-Hornung. 2019. Interactive Hand Pose Estimation using a Stretch-Sensing Soft Glove. *ACM Trans. Graph.* 38, 4, Article 41 (July 2019), 15 pages. https://doi.org/10.1145/3306346.3322957

1 INTRODUCTION

Hands are our primary means to manipulate physical objects and communicate with each other. Many applications such as gaming, robotics, biomechanical analysis, rehabilitation and emerging human-computer interaction paradigms such as augmented and virtual reality (AR/VR) critically depend on accurate means to recover the full hand pose even under dexterous articulation. These challenging applications require that a hand tracking solution fulfills the following requirements: 1) it must be *real-time*, 2) it should work in a variety of environments and settings, and 3) it should be minimally invasive in terms of user instrumentation.

In many applications, hand pose is recovered via commercial motion capture systems (MoCap) such as Vicon [2019], but these require expensive infrastructure and markers placed on the user. Marker-less approaches to the task of hand pose estimation include multiple cameras [Ballan et al. 2012; Tompson et al. 2014; Oikonomidis et al. 2011b], or more recently, a single depth camera [Oberweger and Lepetit 2017; Oberweger et al. 2015; Tang et al. 2014; Wan et al. 2016] or even monocular camera [Spurr et al. 2018; Iqbal et al. 2018; Cai et al. 2018; Mueller et al. 2018; Zimmermann and Brox 2017]. Despite this significant progress, vision-based methods require externally mounted cameras with the whole hand visible in the image. This limitation presents a practical barrier for many applications, in particular those where heavy occlusions can be expected, such as while interacting with an object, wearing gloves or other items of clothing or while working in cluttered environments. Thus camera-based techniques are limited to applications with a controlled environment and impose physical constraints on immersive user experiences.

Mounting sensors directly onto the user's hand removes the need for direct line-of-sight and can improve robustness and reliability. Not surprisingly, a variety of glove-like devices have been proposed in research (e.g. [Chossat et al. 2015]) and are available commercially (e.g., [Cyb 2019; Man 2019]). Such approaches typically leverage inertial measurement units (IMUs), bend sensors, strain sensors or combinations thereof to capture local bone transformations. While potentially accurate, placing a sufficient amount of sensing elements on a glove in order to capture all the degrees-of-freedom (DoFs) of the hand is challenging due to space constraints. Hence, most existing solutions use fewer sensors than there are DoFs in the human hand. This inherently restricts the reconstruction fidelity.

Authors' addresses: Oliver Glauser; Shihao Wu, ETH Zurich, Switzerland, {oliver.glauser, shihao.wu}@inf.ethz.ch; Daniele Panozzo, New York University, USA, panozzo@nyu.edu; Otmar Hilliges; Olga Sorkine-Hornung, ETH Zurich, Switzerland, {otmar.hilliges, olga.sorkine}@inf.ethz.ch.

0730-0301/2019/7-ART41
https://doi.org/10.1145/3306346.3322957

We propose an intrinsic (i.e., without the need for external sensing) hand-pose estimation approach in a thin, unobtrusive form-factor. Our approach leverages two key observations: 1) it has recently become feasible to produce soft, stretchable sensor arrays entirely from silicone [Araromi et al. 2015; Glauser et al. 2019], and 2) modern data-driven techniques can be leveraged to map the resulting sensor readings (which are no longer trivially related to bone transformations) to hand poses. The combination of these two observations leads to our contribution: a soft, self-sensing glove, consisting of an over-complete sensor array (i.e., more sensing elements than DoFs) that can accurately reconstruct hand poses without an optical setup and requiring only minimal calibration. Furthermore, our glove is thin and easy to put on and take off without sacrificing a tight adaptive fit that is crucial for high repeatability.

The proposed glove senses local stretch magnitude exerted on the embedded silicone sensors by measuring their capacity changes. These stretch-driven sensors are small, soft and low-cost. However, the fabrication process proposed in [Glauser et al. 2019] has only been shown to capture simple cylindrical shapes at a frame rate of 8 Hz. Our main hardware contribution is a much more elaborate sensor design in the form of a wearable glove, which requires several improvements to the fabrication process, including integration of the sensor array with a textile cut pattern, as well as a redesign of the readout scheme to enable querying the glove at 60 Hz.

Since the stretch sensors are not in a one-to-one relation with the degrees of freedom of the hand, the reconstruction of the pose is a highly involved task. While Glauser et al. [2019] use an out-of-the-box deep neural network that maps capacitance to 3D vertex positions for this purpose, we discover that a data representation based on prior knowledge of geometric neighborhood and spatial correspondence, both in the input and output domain, allows a neural network to more efficiently discover the inter-dependencies between the joints in the human hand and in consequence outperforms several baseline architectures.

Attaining a sufficiently large and diverse training data corpus for hand pose estimation is a notoriously difficult problem due to the absence of ground-truth acquisition approaches. While this is particularly severe in the case of (2D) image-based approaches (where no instrumentation whatsoever may be used), we observe that our glove design is so unobtrusive, that it is invisible to a depth-camera. This allows us to leverage a state-of-the-art model-fitting based hand tracking approach [Tkach et al. 2017] to capture a large training dataset consisting of one million samples from 10 subjects of time-synchronized sensor readings and the corresponding joint-angle configurations, including a set of shape parameters per person, which we release to the public domain[1] to foster future research.

To validate the utility and performance of our data capture and regression setup, we carry out extensive experiments using different calibration regimes, varying from employing a personalized model for a specific hand, to applying our model to different users with significant variation in hand shapes and sizes. The quality of our reconstruction deteriorates gracefully, offering different calibration options depending on the accuracy required by the application. Finally, we compare with two commercial gloves, demonstrating that our solution shows substantial improvement in reconstruction accuracy (35%), which we believe may have a major impact in real-world applications, especially when paired with the low cost and simple fabrication of our device.

[1] https://igl.ethz.ch/projects/stretch-glove

2 RELATED WORK

The majority of hand pose reconstruction methods are based on either an external vision setup or a set of sensors embedded into a data glove. Most gloves employ sensors from three categories: IMUs (inertial measurement units), bend (flex) sensors, and strain (stretch) sensors. For a complete overview we refer to the surveys [Dipietro et al. 2008; Rashid and Hasan 2018]. Other work has used wrist worn IR cameras [Kim et al. 2012] or magnetic sensing [Chen et al. 2016] for hand pose estimation, and capacitive wrist bands [Truong et al. 2018] or electromyography (EMG) [Saponas et al. 2009] for gesture recoginition. In the following, we summarize the works most closely related to our data glove.

Camera based tracking. A variety of vision based approaches for the problem of hand pose estimation have been proposed in the computer vision and graphics literature (cf. [Erol et al. 2007]). Marker based MoCap approaches (e.g., Vicon [2019]) require multiple, calibrated cameras and, compared to the full-body case, marker occlusions are a more severe problem. In consequence, learning based approaches to marker labelling under occlusion have been proposed [Han et al. 2018]. However, the need for multiple cameras restricts the applicability of such approaches. Wang and Popović [2009] propose a marker-like glove, requiring only one RGB camera. With the widespread availability of consumer grade depth cameras, single sensor solutions have received intense attention [Sharp et al. 2015; Sun et al. 2015; Tagliasacchi et al. 2015; Tang et al. 2014, 2015, 2013; Taylor et al. 2016; Wan et al. 2017, 2016; Zhang et al. 2016]. Depth based approaches can be categorized into model fitting based methods (e.g., [Oikonomidis et al. 2011a; Tkach et al. 2016]) and per-frame classification [Sun et al. 2015; Tang et al. 2014; Wan et al. 2017; Tang et al. 2015]. Moreover, many hybrid approaches that initialize a coarse hand pose estimate via discriminative approaches and then refine this via minimization of some error functional have been proposed [Sridhar et al. 2013; Tkach et al. 2016; Taylor et al. 2016; Tkach et al. 2017; Taylor et al. 2017]. Others deploy convolutional neural networks (CNNs) to regress 3D hand poses from depth images [Oberweger et al. 2015; Oberweger and Lepetit 2017; Sinha et al. 2016; Ge et al. 2017; Tang et al. 2014; Wan et al. 2017] or even from just a single RGB image [Simon et al. 2017; Spurr et al. 2018; Mueller et al. 2018; Cai et al. 2018; Zimmermann and Brox 2017].

In contrast to vision-based approaches, our work relies only on intrinsic sensor readings and, once trained, requires no additional external infrastructure, opening the door to usage scenarios where traditional motion capture approaches are not applicable.

IMU sensor gloves. IMUs consist of a 3-axis accelerometer, a 3-axis gyroscope and a 3-axis magnetometer. Gloves based on 15 [Fang et al. 2017], 16 [Connolly et al. 2018], or 18 [Lin et al. 2018] IMUs have been suggested to recover hand pose. The work of von Marcard et al. [2017] leverages 6 IMUs together with an offline optimization to recover full-body pose, and Huang et al. [2018] use a bi-directional

RNN to learn this mapping from synthetic data and reconstruct full-body poses in real time. One major drawback of IMUs in the context of hand pose estimation is their rigidity and bulkiness compared to the size of human fingers.

Bend sensor gloves. Bend (flex) sensors have been very successfully applied in commercial products like the CyberGlove [Cyb 2019], the VPL Glove [VPL 2019], the 5DT glove [5DT 2019] or the recent ManusVR glove [Man 2019], with the latter also employing two IMUs. [VPL 2019] and [5DT 2019] are equipped with optical flex sensors. Some glove designs leverage off-the-shelf flex sensors [Gentner and Classen 2008; Zheng et al. 2016; K Simone et al. 2007], whereas others focus on designing novel, soft bend sensors [Kramer et al. 2011; Shen et al. 2016; Ciotti et al. 2016]. Typically such gloves feature between 5 and 22 (CyberGlove) sensors, whereas the human hand has at least 25 DoFs. A larger amount of sensing elements is difficult to place and typically increases the complexity of the glove design and consequently the manufacturing cost (cf. CyberGlove [2019]) and may hinder dexterous and natural hand movements. In contrast, our design consists of a single sheet of silicone composite, and the amount of sensing elements is only limited by the surface area and space for routing of connecting leads. We compare with two state-of-the-art gloves [Man 2019; Cyb 2019] in Sec. 5.

Strain sensor gloves. Elastic strain sensors have the potential to allow for very slim and comfortable gloves. Starting with [Lorussi et al. 2005], many different strain sensor gloves, glove parts, or novel sensors tailored for hand capture have been proposed. Most of the presented strain sensor gloves are resistive [O'Connor et al. 2017; Michaud et al. 2016; Hammond et al. 2014; Lorussi et al. 2005; Park et al. 2017; Ryu et al. 2018; Chossat et al. 2015], either using a piezoresistive material, an elastic conductive yarn or conductive liquid channels. Liquid sensors are superior in terms of hysteresis, but their fabrication is often highly involved. Gloves based on capacitive stretch sensors, similar to ours, [Atalay et al. 2017] or video demos by commercial stretch sensor manufacturers [Str 2019; Ban 2018], combine the advantages of a slim form factor, no hysteresis, and softness. At most 15 strain sensors are used for a full glove by [Park et al. 2017], including abduction sensors. This is still significantly less than the amount of DoFs of a full hand; therefore many of the suggested designs are only demonstrated in the context of gesture recognition [Ryu et al. 2018; O'Connor et al. 2017; Hammond et al. 2014; Lorussi et al. 2005] and are not suitable for continuous full hand pose estimation. Some works show pose capture of a part of the hand [Michaud et al. 2016; Park et al. 2017]. Only [Park et al. 2017] and [Chossat et al. 2015] (11 sensors) demonstrate the capture of a full hand, but without evaluating the resulting accuracy. Our sensor design incorporates almost three times as many strain sensors as the closest comparison, and to the best of our knowledge, we are the first to demonstrate the feasibility of accurate, continuous reconstruction of full hand poses from strain sensors alone.

Stretchable sensor arrays. Glauser et al. [2019] extend the capacitive strain sensor concept of [O'Brien et al. 2014; Atalay et al. 2017] and simplify the fabrication method from [Araromi et al. 2015] to achieve dense area-stretch sensor arrays. They demonstrate how

Fig. 2. Our glove consists of a full soft composite of a stretchable capacitive silicone sensor array and a thin custom textile glove (green).

their stretch array sensors, combined with a learned prior, can capture dense surface deformation of simple, cylindrical human body parts like a wrist, elbow or bulging biceps. For an in-depth discussion on different capacitive strain (stretch) sensor types and their fabrication, we refer to [Glauser et al. 2019].

Calibration. To provide reasonable accuracy, appropriate calibration is crucial for data gloves [Kessler et al. 1995], due to specific sensor characteristics and the large variations in shape of different hands. Calibration is often equivalent to finding model parameters, such as gain, offset, or adjusting cross-coupling effects of a custom hand deformation model. Min-max pose calibration [Menon et al. 2003], ground truth calibration [Chou et al. 2000], and inverse kinematics (IK) calibration [Griffin et al. 2000; Hu et al. 2004] are among the most common approaches. Wang and Neff [2013] elegantly combine all three calibration methods to build a Gaussian process regression model, allowing to reconstruct joint-angles with high accuracy. Menon et al. [2003] and Chou et al. [2000] fit a hand-sensor model to individual users. The work of [Menon et al. 2003] assumes specific joint angles for poses to be performed by the user, while [Chou et al. 2000] track a set of markers to overcome the fixed angle assumption. Kahlesz et al. [2004] and Steffen et al. [2011] introduce models mapping from several sensors to one pose parameter to reduce cross-coupling effects. Griffin et al. [2000] ask the user to pose the hand while the thumb and one fingertip touch, and fit model parameters by minimizing fingertip distances. Hu et al. [2004] extend this method by applying a vision system, tracking fingertip positions, and Zhou et al. [2010] extract user-specific calibration parameters from a single image via a ANN. Fischer et al. [1998] use a neural network to learn a mapping from sensor readings to fingertip positions.

We propose a simple yet effective per-user calibration procedure: First, a non-personalized model is trained to map from sensor readings to pose parameters. For new hands, minimal and maximal capacitance values per sensor are captured and used to normalize sensor readings. Note that this is different from the classic min-max calibration, where specific joint-angles or poses are assumed to correspond to the min and max values (e.g., [Menon et al. 2003]). We discuss the calibration details in Sec. 5.

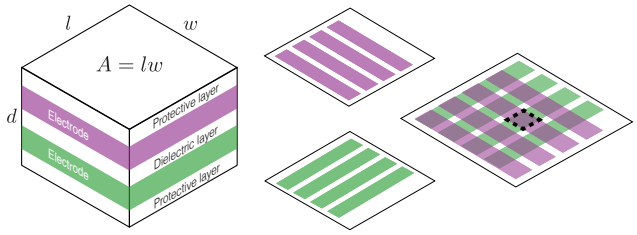

Fig. 3. Left: A capacitive silicone stretch sensor consists of 5 layers. When it is deformed, its capacitance changes. Right: Conductive strip patterns (magenta and green) are embedded into the two electrode layers. Wherever they overlap a local capacitor, which we call sensor cell, is formed. One such sensor cell is marked with a dashed line.

3 COMPOSITE CAPACITIVE GLOVE

The goal of our work is to develop a thin and lightweight glove that is comfortable to wear, yet delivers high pose reconstruction accuracy without requiring elaborate calibration or a complex setup.

Fig. 2 illustrates our final design, consisting in a dense stretch sensor array. It is easy to put on, unobtrusive to wear, and manufacturable at a low-cost (material cost around 15 USD, not including 60 USD for prototype electronics). At the heart of our data glove lies a silicone based stretch sensor array, specifically designed for the purpose of reconstructing dexterous hand articulations. Our design features 44 individual stretch sensors on a hand-shaped silicone sensor array, attached to an elastic textile to form a thin form factor glove. The total weight is just 50 g and its thickness is only 1.2 mm, making it comfortable to wear even for extended use. Our glove adapts well to a range of hand sizes and shapes: one single size fits the hand of all members of our research group.

The sensor is a composite material, consisting of a textile layer paired with conductive and non-conductive silicone layers, fabricated following a procedure inspired by [Glauser et al. 2019], but adapted to the more complex geometry and motion of the hand.

3.1 Sensor design

Sensor. Capacitive stretch sensors are appealing since they are based on the principle of a shape-changing capacitor, which, unlike many resistive sensors, does not suffer from hysteresis. A capacitor is formed by two conductive plates with an overlapping area A, separated by a dielectric (see Fig. 3 left). Any change in shape: width w, length l or distance between the plates d, leads to a change in capacitance $C = \epsilon_r \epsilon_0 A/d = \epsilon_r \epsilon_0 lw/d$, where ϵ_r and ϵ_0 are constants. Therefore, the area of a capacitor can be estimated by continuously measuring the capacitance:

$$\left(\frac{A}{A^0}\right)^2 = \frac{A}{A^0}\frac{A}{A^0} = \frac{A}{A^0}\frac{d^0}{d} = \frac{\epsilon_r \epsilon_0 \frac{A}{d}}{\epsilon_r \epsilon_0 \frac{A^0}{d^0}} = \frac{C}{C^0}, \tag{1}$$

assuming volume conservation holds, i.e., $V = V^0 \Leftrightarrow Ad = A^0 d^0 \Leftrightarrow A/A^0 = d^0/d$.

Sensor arrays. Traditionally, capacitive strain sensors are fabricated individually and connected to a pair of conductive traces for read-out [O'Brien et al. 2014]. To increase the number of sensors placed in a certain area, Glauser et al. [2019] arrange the traces in a grid structure. As shown in Fig. 3 (right), a local capacitor, also called sensor cell, is formed wherever two traces overlap, and each pair of traces overlaps at most once. In consequence, the number of required leads is the sum of the number of grid rows and columns, instead of the product. For the example in Fig. 3, only 8 (4+4) instead of 17 ($4 \cdot 4 + 1$ for ground) leads are required. This space-efficient design allows us to place as many as seven sensors on thin objects like fingers. And for our 44 sensors on the glove, only 27 leads in 2 layers are needed, compared to 45 with a non-matrix approach, a reduction of 42.5%.

Readout scheme. The matrix layout means that sensor cells cannot be read directly. Furthermore, Glauser et al. [2019] experimentally verified that simple variants of scanning schemes commonly used in mutual capacitive touchscreens cannot be applied. Instead, they introduced a time-multiplexed readout scheme, where for each of the measurements a voltage is applied to a subset of traces, while the remaining leads are connected to ground. This way, a temporary (compound) capacitor is formed, whose capacitance is measured. There exists a linear relationship between the compound capacitance values $\mathbf{C}_m$ and the desired individual capacitor values $\mathbf{C}_c$ [Glauser et al. 2019]:

$$\mathbf{M}\,\mathbf{C}_c = \mathbf{C}_m\,. \tag{2}$$

The rows of the rectangular matrix $\mathbf{M}$ encode all possible measurement combinations, and it transforms the sensor cell capacitances $\mathbf{C}_c$ into the measured combined capacitances $\mathbf{C}_m$. The rows of matrix $\mathbf{M}$ are formed by iteratively connecting one trace from the top and one trace from the bottom layer as source electrode, with all remaining traces connected as the ground electrode. Our glove layout has 15 traces in the bottom layer and 12 traces in the top layer, resulting in $180 = 15 \cdot 12$ rows in $\mathbf{M}$. Each row corresponds to one measurement, so that the 44 sensor cells in our glove design require 180 measurement combinations. The linear system above is overdetermined by design (for better robustness) and is solved in the least-square sense. Following [Glauser et al. 2019], we obtain the capacitance by measuring the charging time. However, the procedure and choice of resistors used in the original configuration, would lead to an insufficient readout rate of only 5 Hz in our setting.

The sensor readout scheme in [Glauser et al. 2019] neglects the lead resistance. Therefore, high charging resistors (56 kOhm and 470 kOhm) are required in their case to achieve physically accurate stretch reading. In our case, the readings only need to be repeatable but do not necessarily directly correspond to physically meaningful stretch values. This allows us to use lower charging resistors (47 kOhm and 220 kOhm), improving the readout rates. Further, instead of solving for $\mathbf{C}_c$ every full cycle of combined measurements (180 updates), we solve every 16 updates. We experimentally found that this setup provides good sensor readings, and that more frequent solving has a negative impact on the frame rate due to the limits of the micro-controller-host communication bottleneck. Our readout scheme has a capture rate of about 60 Hz. To filter out noise in the readings, we mean-filter the last five frames of $\mathbf{C}_m$ before solving for $\mathbf{C}_c$.

The readings $\mathbf{C_c}$ are fed to a deep neural network that outputs hand poses (see Sec. 4). They can then be queried by an application,

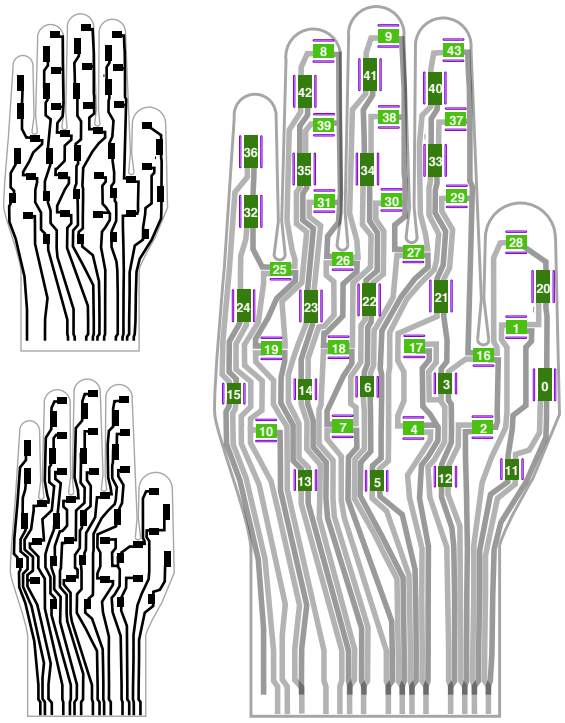

Fig. 4. Left: Patterns of the two conductive layers. Right: Wherever the two conductive layers overlap, local capacitors form (marked in green) and serve as local stretch sensors.

e.g., to render the hand in VR or perform collision detection with virtual objects for interaction. In our live experiments, the hand poses are filtered by a so-called 1€-Filter [Casiez et al. 2012].

Sensor layout. The sensor layout (Fig. 4) is manually designed by adding sensors in stages: (i) longer sensors directly correspond to the main joints of the fingers (21-24, 32-36, 40-42) and the thumb (0, 20); (ii) abduction sensors in-between the fingers (16, 25-27); (iii) perpendicular sensors on the fingers (8-9, 29-31, 37-39, 43) and the thumb (1, 28); (iv) a regular grid of both horizontal (2, 4, 7, 10, 17-19) and vertical (3, 5, 6, 11-15) sensors on the back of the hand. The subtle differentiation into horizontal and vertical sensors is the result of the ventilation cuts, explained in the next paragraph. Fig. 12 shows how each of these sensor categories helps to improve the reconstruction accuracy. Finally, the sensors are connected by leads in two layers, such that each pair of connected traces (from different layers) overlap at most once. We consider reduction of the lead lengths and avoidance of stretch absorption by the nearby cuts when determining the final sensor placement. For these reasons, e.g., the sensors over the knuckles (32-36, 40-42) are not centered, leaving some blank space.

Note that for good sensitivity with respect to finger abduction it is important that the sensors are pre-stretched when the glove is put onto the user's hand. It is thus crucial to fabricate the sensor array in the rest pose shown in Fig. 4, right. In particular, the fingers must be parallel without any gap in-between.

Cuts. Thin cuts (Fig. 4 right, in purple) with rounded ends are added via laser cutting on two sides of the rectangular sensors to enhance the wearing comfort by increasing ventilation. They also have a minor, yet positive, effect on the readings, since they lower stretch force resistance, thus making the sensors more sensitive to stretching parallel to the cuts. For example, sensors 21, 33 or 40, located over the joints of the index finger, are much less sensitive to volume changes of the finger, while sensors like 43, 37, 29 are mainly sensitive to volume or diameter changes of the finger (e.g., due to muscle bulging). In Fig. 4 (right) the sensors more sensitive to vertical stretch are colored in dark green, and the ones more sensitive to horizontal stretch in light green.

3.2 Fabrication

Our glove is made of a composite consisting of a silicone sensor sheet and an elastic textile, only requiring tools available in a modern fablab. It is fabricated in a two-stage approach, as outlined in Fig. 5: first, we fabricate the soft silicone sensor array (steps 1-8), covering the back side of the glove, and then we attach textile parts to the silicone sheet and close them up to form a soft and wearable glove (steps 9-12).

Stage I: Silicone. The hand-shaped silicone sensor array (see Sec. 3.1) consists of two conductive, patterned layers with a dielectric layer in-between and encapsulated by two shielding layers, shown schematically in Fig. 3. It is produced layer by layer using the following steps.

First, we cast an insulating base layer onto a glass plate, controlling the thickness by attaching tapes at the borders of the glass plate. Next, a conductive layer, made from Silbione RTV 4420 silicone [Sil 2019b] mixed with carbon black (conductive powder, [Ens 2019]), is cast directly onto the first layer. The laser cutter then removes, by repeatedly etching (5 times) the negative of the pattern shown in Fig. 4 (lower left), leaving the full base layer with the conductive traces on top. Then, a pure silicone dielectric layer is cast, followed by another conductive layer, which is also etched (Fig. 4, upper left). Finally, another insulating shielding layer is added.

Note that the conductive layers are produced with a thickness of 220 μm to allow for the needed leads of just 2 mm width (see Fig. 4). To keep the connection pads at the base of the sensor exposed, a thin tape is used to cover the pads before casting (for the last three layers) and removed before the curing in the oven. A detailed description of the silicone mixtures and additional information on the sensor-to-read-out-circuit interconnection are provided in Appendix B.

The laser cutter parameters in the etching step are Power=30, Speed=40, PPI=500 (Trotec Speedy 300 laser cutter). Using a higher power during the etching process would make the silicone sensor cross-linked with the base glass and hard to peel off in the end. After every full etching cycle, the sensor is cleaned from dust residue by carefully wiping it with a towel and isopropyl alcohol.

After every casting step the sensor is cured in the oven for 20 minutes at 90 °C. Before curing in the oven, sensors have to be left sitting for 15 minutes, to let the solvent evaporate. Otherwise, bubbles can form during curing due to the evaporation of the solvent from within, with the uppermost part of the layer already cured.

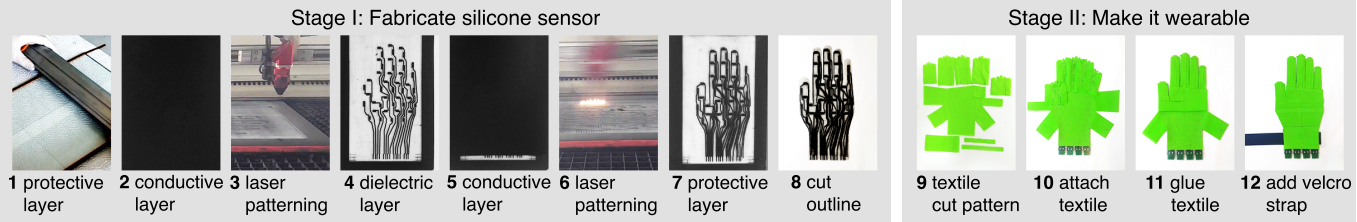

Fig. 5. The fabrication of a glove consists of two main stages: Fabricating the silicone sensor (1-8) and textile design for the glove (9-12). Note that the conductive layers (2,5) are a mixture of silicone and carbon black and therefore, mostly black.

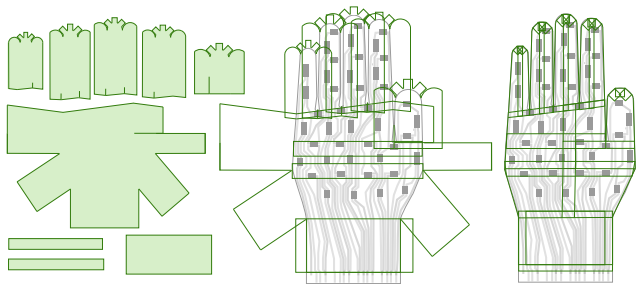

Fig. 6. Left: The textile cut pattern is made from a large palm part, one for each finger and 3 extra flaps for connections. Middle: Alignment of the cut flat pattern parts with the flat, hand-shaped silicone sensor. Right: Finished glove after closing up the flaps with textile glue.

Finally, the sensor is cut into a hand shape with the laser cutter. An accurate alignment of the etching and cutting steps in the laser cutter is crucial to avoid cuts in the sensors, as this could lead to short circuits between the conductive layers. The overall thickness of our sensor is 0.85 mm.

Stage II: Textile. The silicone sensor array is not wearable. There is no easy way to attach it firmly to the hand, and gluing two sheets of silicone together is a difficult (while not impossible) task. Attempting to put on or take off such a glove is very cumbersome due to large friction and tightness. We attempted to attach the sensor to a standard glove, but found that it is challenging to get a proper alignment with the major centers of articulation, and it is also difficult to do it robustly and with the needed repeatability.

Therefore, we propose a simpler and more effective solution, exploiting a laser cutter to cut a custom textile pattern (see Fig. 6). The textile parts can be attached to the silicone sensor while laying on a flat surface. First, a PET mask that covers the sensors and the cuts is placed on the sensor, then everything is covered with Sil-Poxy silicone adhesive ([Sil 2019a]), and finally, the mask is carefully removed, and the textile parts are placed and firmly attached.

In a second step, the different textile parts are closed up, using HT 2 textile glue ([HT2 2019]), and the seams are bonded with an electric iron. A highly elastic jersey textile (80% polyamid with 20% elastane) with a thickness of 0.35 mm is used. Finally, we attach a wrist strap with a velcro fastener to reinforce the tightness and ensure a repeatable alignment of sensor cells to joints.

3.3 Comparison to [Glauser et al. 2018]

While our sensor array is based on [Glauser et al. 2019]), their fabrication process cannot be directly applied to our setting. Our sensor is a composite material made of a silicone layer (based on [Glauser et al. 2019], see Appendix B) and an additional textile layer. The latter is crucial in making a functional glove, since pure silicone cannot be draped over complex geometries without the risk of immediate damage, especially due to the large friction with the human skin. In terms of the silicone sensor layer, there are two major differences: (1) the readout scheme is different (Sec. 3.1), allowing for a frame rate of 60 Hz (vs. 8 Hz), and (2) we seek to reconstruct a much more complex geometry that is articulated in more complex ways than the simple cylindrical shapes and single axis joints that shown in [Glauser et al. 2019]. The thin structures of the hand require high sensor density, but offer little surface area to place the sensor cells. To overcome this problem, we use much thinner leads (2 mm vs. 6 mm) and smaller local sensor cells (5 x 7.5 mm and 5 x 11.5 vs. 15 mm diameter). To keep the total resistance of the longest leads in a useful range, the conductive layers have to be five times thicker (220 μm instead of 45 μm). As a consequence, more etching cycles are required during fabrication, and the solvent must pre-evaporate to prevent bubbles from forming during curing in the oven.

4 DATA-DRIVEN HAND POSE ESTIMATION

Our composite capacitive glove contains more sensors (44) than the number of DoFs in the human hand model we consider (25), however, the stretch sensors' readings are not in a direct relationship with the joint rotation centers of the hand. Furthermore, sensor readings vary from person to person due to the different hand geometry. Designing a custom mapping from sensor readings to hand joint configurations manually is a highly complex task (e.g. [Kahlesz et al. 2004]), which requires experiments and manual work to adapt the model to the specific sensor layout. We propose instead a data-driven approach that can learn this highly non-linear mapping, across different sessions and users. While acquiring training data for hand pose estimation is generally difficult, gloves are a special case since they can be unobtrusive enough to be essentially invisible to a depth camera. Therefore, it is possible to capture training data efficiently using an off-the-shelf hand tracking system [Tkach et al. 2017].

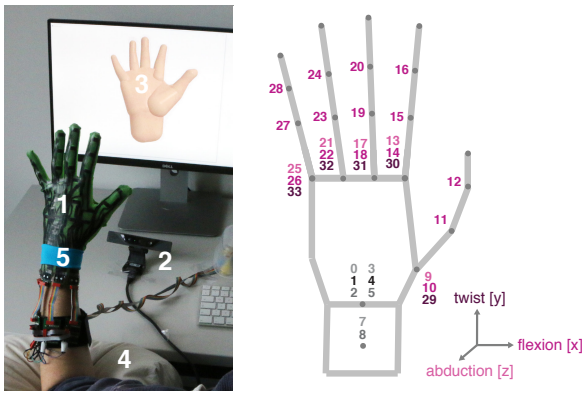

Fig. 7. Left: Our training data capturing setup. 1) Hand with glove; 2) RealSense SR300 depth camera; 3) computer running [Tkach et al. 2017]; 4) pillow for comfort; 5) blue segmentation wristband as required by [Tkach et al. 2017]. Right: The 34 hand pose parameters proposed by [Tkach et al. 2017]. Our glove only captures the 25 degrees of freedom in color. The DoFs in gray are the global translation and rotation, and the rotation of the wrist; global pose parameters cannot be captured using only stretch sensors.

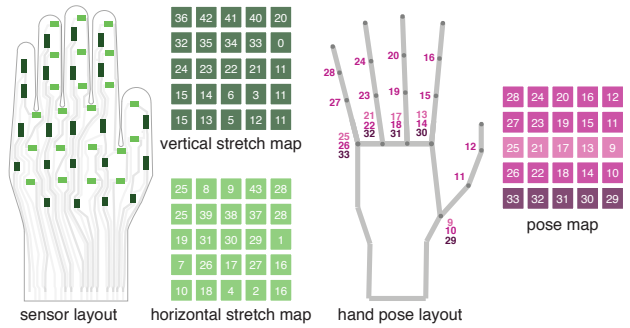

Fig. 8. The geometric correspondences between the input (sensor layout on the left) and output features (hand model on the right) should be considered. Both the input and the output can be naturally ordered in corresponding grid structures.

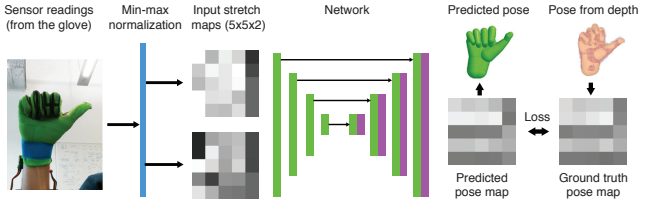

Fig. 9. From left to right: The sensor readout is normalized by the hand-specific min-max, arranged in two 5x5 stretch maps, and fed through a U-Net network that predicts a 5x5 pose map. The loss is the L_2-norm difference between the predicted pose map and the ground truth pose map derived from the hand poses captured by [Tkach et al. 2017].

Any standard neural network architecture could be used in our setting, including fully connected networks (FCN) and long short-term memory networks (LSTM). However, we observe that these standard approaches struggle to exploit the geometric layout of our data. By constructing an ad-hoc data layout and a network that implicitly encodes the sensor geometry and topology, we considerably improve the accuracy over standard baselines.

4.1 Data acquisition

Our setup for capturing training data is shown in Fig. 7 (left). For capturing the reference hand poses, we use an inexpensive Intel RealSense SR300 depth camera [Rea 2019]. Depth frames are fed to the (unmodified) algorithm of [Tkach et al. 2017], which requires a blue strip on the wrist for depth segmentation. We use their calibration method to compute the hand shape parameters per user. To capture meaningful training data, a good synchronization between the different data sources is crucial. To this end, we incorporate our code for communication with the glove sensor into the publicly available source code of [Tkach et al. 2017]. This allows for unified collection, evaluation and logging of both sensor and pose data.

4.2 Data representation and network

For N frames in the training data, the input $\mathbf{X} = \{x_i\}_{i \in N} \subset \mathbb{R}^{44}$ to our regression model is the readout from the 44 stretch sensors, while the target output $\mathbf{Y} = \{y_i\}_{i \in N} \subset \mathbb{R}^{25}$ are the 25 hand pose parameters as defined in [Tkach et al. 2017], covering the full pose DoFs of the hand (see Fig. 7).

Data representation. Our key observation is that the spatial correspondences between input and output features should be considered. A meaningful ordering and organization of features make the learning task easier. For example, a group of nearby sensors on the thumb (sensor cells 0, 1, 2, 11, 16, 20, 28 in Fig. 4, right) taken together should have a higher impact on the prediction of the thumb movement (parameters 9, 10, 11, 12, 29 in Fig. 8). Meanwhile, some high-level hand gestures, like the clenching of a fist, cause more uniform sensor actuation, which should be encoded globally and hence makes a priori definition of these inter-dependencies difficult. Training an FCN to learn such global-local information is theoretically possible, but practically it would require excessive amounts of model capacity, training data, and hyper-parameter tuning. We opt instead to directly build this geometric and topological prior into our network architecture to regularize the learning process and improve the reconstruction performance.

We use a fully convolutional neural network (CNN) and 2D grid representations as input and regression target. More specifically, we use 5×5 matrices to organize our input and output data. Fig. 8 shows how we map sensor cell readings and pose parameters to 2D grids, each capturing the spatial relationships. We use one matrix to organize the output, but a stack of two for the input, since each sensing location has two types of sensors measuring horizontal and vertical stretch (see Sec. 3). For example, both sensors 29 and 33 are located around the knuckle of the index finger, but each sensor captures different stretch directions.

2D network. We use the U-Net network architecture [Ronneberger et al. 2015] to transfer the organized sensor readout to hand pose parameters. The downsampling and upsampling structure of the network can encode the global information, while the symmetric

skip connections between the encoder and the decoder can preserve the local correspondences. Fig. 9 illustrates the structure of U-Net and how the network transforms the 2D sensor data to hand poses. We use L_2 loss for our regression task, $\mathcal{L}_{\text{reg}} = \sum_{i=1}^{25} \left\| \hat{y}_i - y_i \right\|_2$, where $\hat{y}$ is the prediction and y is the target pose parameter.

Experiments show that our model compares favorably to alternative network architectures. We provide a comparison against baselines in Sec. 5.4 and describe the experimental setup in detail in Appendix A.

4.3 Data processing

We improve our data quality by removing outliers and by using a min-max normalization method for calibration. That is, the input to our network is the processed and mapped sensor data, see Fig. 9.

Outlier removal. We remove frames that are likely to be outliers by detecting finger collisions, since they indicate unfeasible poses. We filter out frames where the collision energy defined in [Tkach et al. 2017] is above 80, indicating that the estimated pose is likely unnatural and wrong. This filter only removes about 2% of the data.

On-the-fly calibration. Ideally, the per-sensor reading magnitude should be normalized to become insensitive to the hand size. We observe that, once the glove is put on, the minimum and maximum magnitude of each sensor's readings is fixed, which can be used to normalize the sensor data. Therefore, we find that a per-sensor min-max calibration is a reasonable trade-off between cost and accuracy. The key is to find the min and max magnitude after the glove is put on. In practice, we propose a short calibration phase, where the user should freely explore different extreme poses, yielding the min and max values per sensor, which we then use to normalize the sensor data to the $[-1, 1]$ range. To make this process even more robust, we use a median filter (over 20 frames) while extracting the min and max values. This simple calibration method works surprisingly well in practice, due to the complexity and tightness of our soft glove, which provides a proper alignment.

5 RESULTS

We show how our glove and the symbiotic data-driven hand pose reconstruction method can capture accurate hand poses (Sec. 5.2) on a large dataset (Sec. 5.1). We compare our glove's performance on a pose sequence with two commercial state-of-the-art gloves (Sec. 5.3). Finally, we evaluate the proposed network architecture, contrasting it with alternative baselines (Sec. 5.4).

In the setting where a new glove user only needs to perform a minimal on-the-fly calibration (min-max normalization using a generic, pre-trained model), we achieve an overall mean error of only 7.6 degrees. In a comparison sequence, with a mean pose error of 6.8 our glove outperforms the ManusVR glove (mean error: 11.9) and the CyberGlove (mean error: 10.5). The proposed 2D network architecture can achieve a mean error of about 1 degree lower than the baseline fully-connected network.

5.1 Dataset

Our experiments are performed on a large data set[1] captured from 10 people (except where noted), including a wide range of hand sizes and shapes. The hand length varies from 17 to 20.5 cm, the width from 9 to 11 cm, and the aspect ratio of length to width from 1.6 to 2.1. For each person we capture five sessions using our data acquisition setup; each session lasts about 5 minutes. During three of the five sessions, the participant keeps the glove on continuously, while in-between the other two sessions the glove is taken off. We refer to these two regimes as *intra-session* and *inter-session*, respectively. To encourage the participants to explore the space of hand poses fully, we show a printed gallery of example poses during the recording sessions. During data acquisition and method development, one of our gloves was in use for over 25 hours (cumulative) — consistently capturing sensor data in high quality.

5.2 Evaluation on hand pose capture

We envision a standard scenario for our hand capture method, in which the proposed neural network is trained only once, preferably on a large data set containing samples from different hands. This way, a new user only needs to execute the on-the-fly calibration method for less than a minute before using the glove for interaction.

In our experiments, we refer to models that are trained using the data from all participants except leaving one participant out as test data as *generic models*. We also evaluate *personalized models*, which are trained on the data from one person only. This allows for even more accurate pose reconstruction and provides further insight into the capabilities of our glove. Table 1 summarizes results of all our models. For all experiments we used a medium sized glove (20×12.5 cm); despite the single size, it can handle a large variety of hands. The most significant error is produced by the smallest hand (last row of Table 1) – for a more accurate tracking, a smaller size glove would be required.

Personalized and Generic models. For the personalized model, we perform experiments on two types of data: using training and testing on intra-sessions only and using both types of sessions for training, tested on an inter-session. For the former, we use two sessions to predict the other one. For the latter, we use three intra-sessions and one inter-session to predict the other inter-session. The intra-session samples usually have better performance than inter-session ones. This is due to better alignment of the glove during a continuous session. The *Intra* and *Inter* columns in Table 1 show the mean angular reconstruction errors for ten different hands. On average, the mean error for the intra-sessions is 5.8 degrees versus 6.2 for the inter-sessions. The small error difference suggests that our soft glove provides consistent alignment across different sessions even when the glove is taken off in-between. See Fig. 10 for example frames from a real-time capture session.

A generic model is crucial for real-world applications aimed at a wide and diverse audience, since training a personalized model is time consuming (2 hours or more) and requires additional equipment (depth camera and GPU). We evaluate the approach in two variants: *with* and *without* using the calibration method described in section 4.3. In the case without calibration, a per sensor min-max values over all users are obtained from the training data and applied for normalization, both in training and in testing. Columns (3) and (4) in Table 1 show the effectiveness of our calibration method: the average pose reconstruction error is 7.6 degrees with calibration

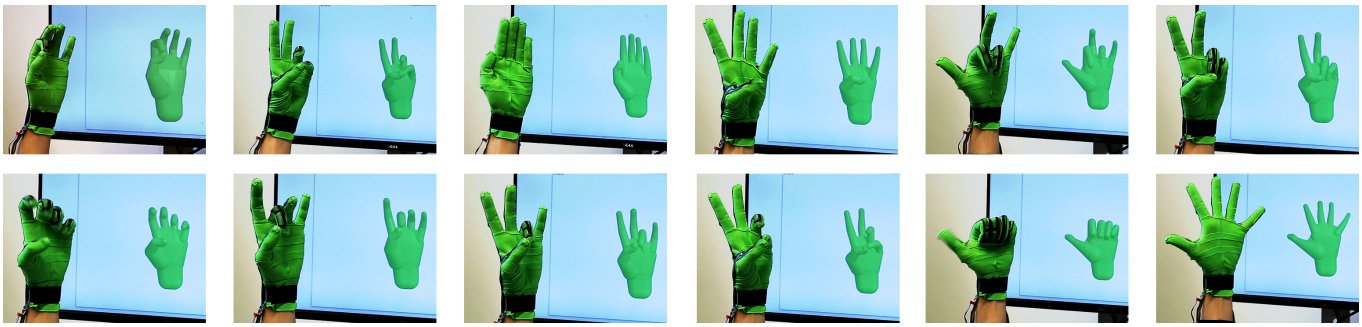

Fig. 10. A gallery of real-time session frames showing diverse poses predicted by a personalized model.

Table 1. Mean pose angle errors (in degrees) over sessions captured from people with different hand sizes and aspect ratios. The *Size* column lists bounding box volume in cm^3, and the second column gives the bounding box dimensions in cm. We report different scenarios: Personalized models, trained on sessions (1) with the exact same alignment like the test session, or (2) when the glove is taken off in-between; generic models trained on sessions from the other 9 participants (leave-one-out): (3) uses the per-feature min-max sensor data obtained from the training data, and (4) uses the personalized, on-the-fly min-max calibration. (5) Generic model fine-tuned with a short (5 minutes) session of personal training data. *External hardware* refers to the depth camera and GPU necessary for training data capture and processing.

Hands		Personalized		Generic		Fine-tuned
Size	L×W×H	(1) Intra	(2) Inter	(3) w/o Calib	(4) w Calib.	(5) Tuned
940	19×11×4.5	4.8	5.5	6.4	6.6	5.7
792	19×9.5×4	5.3	6.8	7.9	7.0	6.1
792	18×11×4	6.2	7.8	8.6	8.5	6.6
836	19×11×4	5.2	5.2	7.3	6.9	5.6
850	18×10.5×4.5	6.8	6.3	8.1	7.3	6.8
1025	20.5×10×5	5.1	5.9	8.5	7.5	6.4
840	20×10.5×4	5.2	5.9	8.8	8.0	7.2
680	17×10×4	5.6	5.7	8.6	8.2	6.7
800	20×10×4	6.1	6.5	9.0	7.3	6.0
612	17×9×4	7.5	6.6	10.1	9.1	8.1
Average error		5.8	6.2	8.3	7.6	6.5
Time investment		2 h	2.5 h	0	1 min	20 min
External hardware		Yes	Yes	No	No	Yes

versus 8.3 without. An angular reconstruction error of 7.6 is satisfactory for many applications (see Fig. 16 and the supplementary video for a visualization of different reconstruction errors). To further improve the reconstruction quality with minimal personalized data, we apply fine-tuning on then unseen data. That is, we load the network parameters from a pre-trained generic model and then use a small learning rate of 1×10^{-6} and batch size of 64 to further optimize all the network parameters, which helps in avoiding catastrophic forgetting. The results are reported in column (5) of Table 1; they are comparable in performance to a personalized model, but require a much lower investment of time.

Application scenarios. Our method supports five standard application scenarios, summarized in Table 1:

(1) An intra-session personalized model gives the best performance, but it requires to always keep the glove on.
(2) If a depth camera is available to the user, personal training data (20 minutes) can be captured with [Tkach et al. 2017] and used to train a personalized model for about 2 hours.
(3) If there is no time or ability (e.g., in a rehabilitation context) for training and calibration, our generic model can be used, combined with the per sensor min-max values extracted from the training set.
(4) By first exploring some hand-poses to gather personal min-max values on-the-fly and then using these values to normalize the sensor data, the accuracy of the generic model can be significantly improved, within less than a minute of calibration time.
(5) A trade-off alternative to scenarios (2) and (4) is to capture only 5 minutes of personal training data and fine-tune the generic model for about 15 minutes.

Options (3) and (4) require only the glove and a pre-trained model, while the others need a depth camera and a GPU to train or fine-tune the model. We believe that (4) is the most practical scenario, but applications requiring higher accuracy might benefit from a custom model (2) or (5). In practice, all of our models can capture hand poses reasonably well, and a visual comparison of models (2), (4) and (5) is shown in Fig. 11.

Number of sensors and training data. To illustrate the benefits of a dense sensor array, we run an ablation study on the number of sensor cells used, to simulate glove designs with fewer sensors (see Fig. 12). The results show that using more sensors leads to higher reconstruction accuracy, an 28% decrease in the mean error when going from 14 to 44 sensors.

Our training, validation, and test datasets for the personalized models contain 85K, 10K, and 15K samples, respectively. The numbers of samples for the non-personalized model are 800K, 90K, and 120K. To study the necessity of using such a large training data set, we gradually and randomly remove parts of our training data; the resulting reconstruction errors of the personalized model (2) and the generic model (4) are shown in Table 2. The drop in reconstruction accuracy demonstrates the benefit of having a large dataset, and

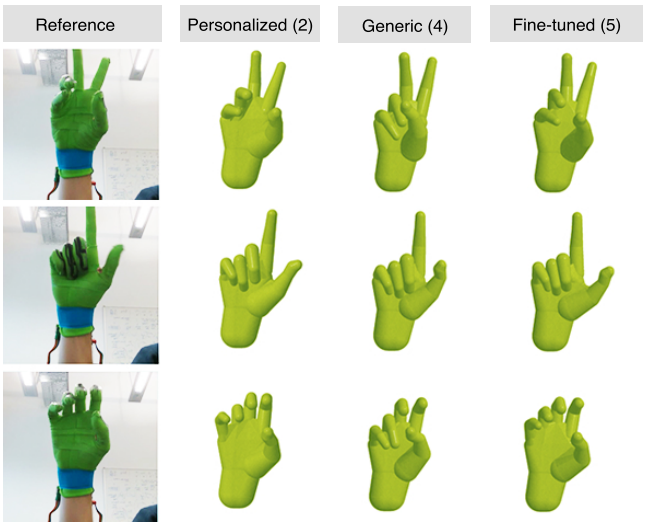

Fig. 11. Visual comparison of different models on three example frames. From left to right: ground truth pose, reconstruction of personalized (2), generic (4), and fine-tuned (5) models. While all models manage to capture these poses well, the personalized model (2) performs best. We carefully chose these frames to highlight the differences. Most poses have visually similar results for all models as shown in the accompanying video.

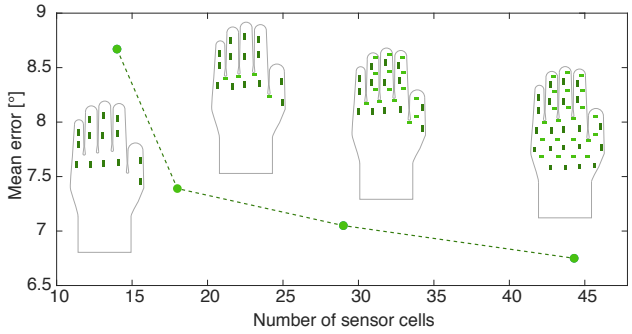

Fig. 12. As the number of sensors increases, the mean reconstruction error of a captured session decreases: from 8.67 with just 14 sensors covering the main joints to 6.75 for our full glove with 44 sensors.

Table 2. This table shows that the mean session error increases if less training data is available (or used). The number of samples in the training data is 85K for the personalized and 800K for the generic model.

Percentage	100%	75%	50%	25%	10%
Personalized (2)	5.50	6.32	6.47	6.57	7.14
Generic (4)	6.57	6.98	7.51	7.96	9.85

hence the importance of our unobtrusive glove that allows for a convenient data acquisition setup.

Generalization to a different glove. In all the experiments presented so far, we use a single glove prototype (*Glove I*), for both training data capture and testing. To evaluate the reproducibility of our fabrication procedure, we fabricate a second glove (*Glove II*) and assess how well a model trained on data from Glove I predicts poses (Fig. 13) using readings from Glove II. Table 3 summarizes the results, they are very encouraging, especially given that our current fabrication process includes some manual steps (see Sec. 3). We believe that an automated, industrialized version of our glove fabrication process could further improve the reproducibility of our composite glove.

Table 3. Generalization to a different glove: This table summarizes errors when evaluating model variants on a training session captured with Glove II. From left to right: Generic model trained on Glove I data only; Generic (Glove I) model fine-tuned with 5 minutes of data from Glove II; Personalized model trained on Glove II only.

Model Trained on	Generic w calib. Glove I	Fine-tuned Glove I & II	Personalized Glove II
Error	8.80	5.73	5.30

Fig. 13. Glove II can predict hand poses with reasonable accuracy using a model trained only on the data captured with Glove I.

Object interaction. In Fig. 1 and the supplemental video, we demonstrate our glove interacting with different objects. In general, touching or pressing onto capacitive sensor arrays influences the readings due to body capacitance or deformation of the local capacitors. But usual grabbing and holding of objects makes contact mostly occur on the inside of the hand or at the finger tips where no sensors are placed. Glauser et al. [2019] illustrate the effect of touching a capacitive sensor array in an experiment.

5.3 Comparison to state-of-the-art data gloves

We compare our glove to two state-of-the-art commercial glove products: a data glove by ManusVR [Man 2019] and the CyberGlove II [Cyb 2019] by CyberGlove Systems LLC. To the best of our knowledge, the ManusVR glove has ten bend sensors and 2 IMUs, while the CyberGlove II is equipped with 22 flex sensors. Before the evaluation, we calibrate the two state-of-the-art gloves with their proprietary software. Both routines ask the user to perform a given set of hand poses and only take a matter of minutes, comparable in time investment to our min-max sensor normalization, which we use for the comparison (generic model (4)). The gloves are queried through the provided SDKs to record pose data.

For each of the three gloves (ManusVR, CyberGlove and ours) the same sequence of 60 hand poses and a duration of about 3 minutes is recorded. Alongside, the hand pose angles are also captured by the depth tracking system [Tkach et al. 2017], which we use as

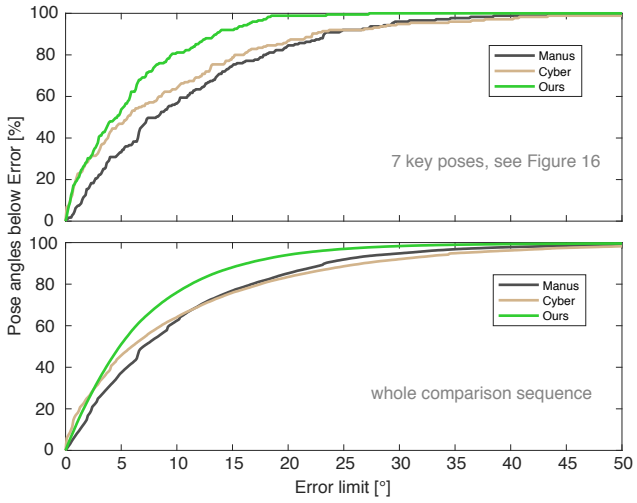

Fig. 14. Top: Cumulative error plot of the key poses, comparing different gloves. Over the key poses, our glove predicts 92% of the angles below an error of 15 degrees (CyberGlove: 79%, ManusVR: 75%). Bottom: Cumulative error plot over the whole comparison session.

ground truth. This choice might introduce a bias in the comparison due to the use of the same tracking system for training data acquisition. The angles received from the ManusVR, and the CyberGlove are mapped (following the description in the SDK) to the 25 degrees of freedom of [Tkach et al. 2017]. Some pose angles come with an offset, therefore all angles from the two state-of-the-art gloves are shifted, so that in the first frame of the recorded sequence they exactly match the ground truth. Over the whole sequence, the ManusVR glove has a mean error of 11.93 degrees, the CyberGlove 10.47 degrees and ours 6.76 degrees — this is 35% lower than the next best result. As the sequences are not exactly the same, in Fig. 16 we additionally show seven poses of the comparison sequence with the corresponding mean error over all degrees of freedom. Fig. 14 (top) shows a cumulative error plot comparing the percentage of angular degrees of freedom below a specified error threshold (on the x-axis) for the seven poses shown in Fig. 16. We observe that 92% of the angles have an error below 15 degrees for our glove, while for the CyberGlove it is 79% and the ManusVR glove 75%. The lower part of Fig. 14 shows a cumulative error plot for the entire comparison sequence.

5.4 Comparison of networks

We report results from experiments with two 1D baselines (FCN, LSTM) and three types of 2D network architectures: ResNet [He et al. 2016], U-Net [Ronneberger et al. 2015], and conditional generative adversarial network (CGAN) [Isola et al. 2017]. In Tables 4 and 5 we compare the five types networks on our personalized model (2) and generic model (4). In general, the 2D-based networks are faster to converge and lead to lower reconstruction error. The performance of FCN is not satisfactory, especially when the training set is not diverse, as in the case of the personalized model. LSTM yields smooth results with higher reconstruction accuracy than

Table 4. Comparison of different networks for the personalized model in terms of mean angle-error in degrees. (2). From left to right: five different network architectures. From top to bottom: varying amounts of network parameters. We adjust the sizes or numbers of layers for each network to meet the target number of parameters.

Network size	FCN	LSTM	ResNet	U-Net	CGAN
3M	6.63	5.81	6.06	5.63	5.59
13M	6.95	6.02	6.12	5.50	5.51
50M	7.10	6.38	6.20	5.55	5.47

Table 5. Comparison of different networks for the generic model in terms of mean angle-error in degrees. (4), trained on the leave-one-out dataset. From left to right: five different network architectures. From top to bottom: varying amounts of network parameters, similar to Table 4.

Network size	FCN	LSTM	ResNet	U-Net	CGAN
3M	7.64	7.68	7.28	6.81	7.09
13M	7.58	7.65	7.35	6.57	6.50
50M	7.98	7.76	7.18	6.65	6.61

FCN, but it tends to over-smooth some high frequency poses, like the touching of two fingers. Among the three 2D-based networks, ResNet already outperforms the FCN baseline considerably, but leaves room for improvement. Both U-Net and CGAN achieve high reconstruction accuracy. In our experiments, the predicted poses of U-Net are visually more stable than those predicted by CGAN. Thus the 13M U-Net is used for all other experiments. It yields the lowest error for both personalized and generic models (Tables 4 and 5). Experiments with networks with fewer than 3M parameters lead to an increased error. For comparison, we also trained an SVM on the data of Table 5, which results in a higher but still acceptable error of 7.8 degrees.

The models compared here cover a broad spectrum of modern machine learning techniques. An exploration of more advanced network architectures against our baselines, like a combination of LSTM and CNN, would be an interesting direction. Hence, we release all our training data[1].

6 CONCLUSION AND FUTURE WORK

In this paper, we focus on the core task of a data glove — capturing accurate hand poses. Furthermore, an optimal data glove should be comfortable to wear, real-time, low cost and easy to use. We achieve these goals via several technical contributions, including a glove-adapted stretch sensor layout and fabrication, a wearable composite of silicone and textile layers, an improved sensor readout scheme to achieve interactive frame rates, a structure-aware data representation and a minimal on-the-fly calibration method. Extensive experiments exploring different scenarios demonstrate the power of our data-driven model and the capabilities of the proposed stretch sensing glove.

To further improve the functionality and applicability of our glove, some essential features and many intriguing extensions are to be explored in the future.

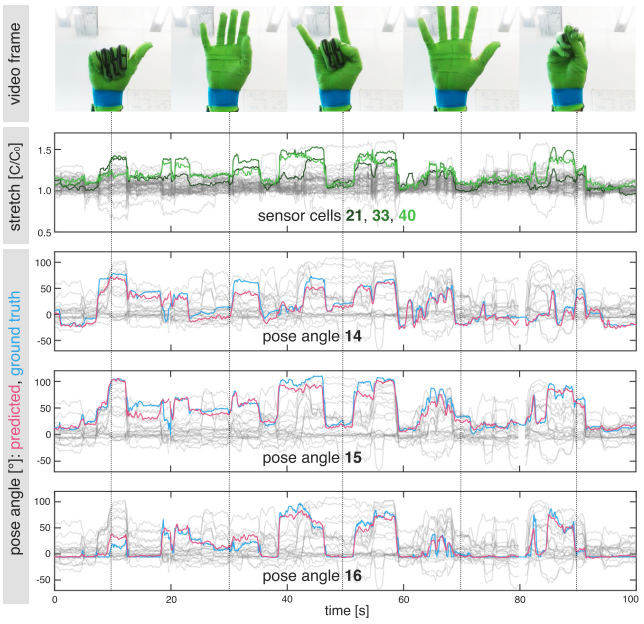

Fig. 15. Visualization of sensor readings and pose reconstruction over time (x-axis). From top to bottom: Five video frames (the dotted lines indicate the correspondences); plot of stretch sensor readings (y-axis), showing all 44 in light gray, with the readings of the three cells (21, 33, 40 in Fig. 4) over the knuckles on the index finger highlighted in green; finally, three plots of predicted and reference pose angles (y-axis), showing all 25 predicted angles in light gray, with the three flexion angles (14, 15, 16 in Fig. 8) of the index finger highlighted in red, and the reference angles captured by [Tkach et al. 2017] in blue.

Applicability. In the presented state, our glove does not come with a global translation and rotation tracking and is still cable-bound. Position and orientation tracking are essential for a "real-world" data glove. Removing the need for a cable (e.g., adding a battery and wireless data transmission) is a well-studied task. To obtain global translation and rotation information, the straightforward solution would be to use an off-the-shelf tracker (e.g., [Viv 2019]). Such a solution needs an extensive setup and still suffers from occlusion. Alternatively, an experimental setup of a sparse set of additional stretch sensors on the arm might allow tracking the hand position. For the wrist and the elbow, Glauser et al. [2019] have already demonstrated how stretch sensors could provide high-quality surface tracking. The efficient fabrication of such gloves at a larger scale requires further research and development. For the fabrication of the (flat) silicone sensors, a conveyor belt system combining the necessary production steps (casting, curing, laser cutting, and cleaning) is conceivable, while the textile glove part would probably need more fundamental adaptations to be better suited for further automation.

Noise and latency. Remaining prediction inaccuracies may be due to the following sources: noise and latency of the sensor readings, material hysteresis, training dataset size and overfitting. In the future, we will research which part contributes to the overall systematic error the most. Empirically, we believe adding more training data and reducing sensor noise are the most promising directions to minimize jitter. The overall latency as seen in the accompanying video ranges from 125 to 200 ms. About 45-90 ms of the latency is due to the oldest of the 180 readings. The inference time of the network model is about 5 ms. The remaining lag comes from un-optimized data communication, filtering, and rendering. For an example of sensor readings and pose predictions over time see Fig. 15.

Customization. So far we only fabricate medium (M) sized gloves, which are already able to handle a large variety of hands, as demonstrated in Table 1. However, we also observe that too small or too large hands can lead to a lower reconstruction accuracy. Therefore, likely two more sizes (an S and an L) are required. Per-person bespoke gloves and how they could further improve the accuracy is another promising direction for future research, especially since our fabrication pipeline trivially allows for adjustments of the size, shape and layout of the sensors. It is conceivable that even better sensor layouts could be found by an optimization based on a simulation, captured data, or a combination thereof.

Extensions. Employing a more involved motion tracking system like [Romero et al. 2017] to acquire training data would be more costly, but could also lead to improved accuracy. In many application scenarios (e.g., when used in combination with AR or VR headsets) cameras are already present — even though often with occlusion and out of field-of-view situations. Therefore, it would be interesting to explore how sensor readings from our glove can be fused with camera based pose predictions. A dense stretch sensor glove might be able to predict not only hand pose but also hand shape parameters. Our soft and thin glove is an ideal candidate to be worn below haptic devices [Hinchet et al. 2018] or soft hand exoskeletons [Polygerinos et al. 2015] that do not come with built-in hand pose capture sensors.

ACKNOWLEDGMENTS

We would like to thank Severin Klingler, Christian Schüller, Velko Vechev and Katja Wolff for insightful discussions, John Sivell for narrating the video and all the participants in the data set collection. This work was supported in part by the SNF grant 200021_162958, the NSF CAREER award IIS-1652515, the NSF grant OAC:1835712, GPU donations from the NVIDIA Corporation and a gift from Adobe.

REFERENCES

2018. Bando. https://www.youtube.com/watch?v=2XskNHtarj0.
2019. 5DT Glove. http://www.5dt.com/data-gloves/.
2019. CyberGlove. http://www.cyberglovesystems.com/.
2019. HT2 Textile Glue. https://consumer.guetermann.com/en/product-finder/textile-glue-ht2.
2019. HTC Vive Tracker. https://www.vive.com/eu/vive-tracker/.
2019. Imerys ENSACO 250G. http://www.imerys-graphite-and-carbon.com/wordpress/wp-app/uploads/2014/04/Polymer$_c$ompounds1.pdf.
2019. Intel Real Sense SR3000. https://software.intel.com/en-us/realsense/sr300.
2019. ManusVR glove. https://manus-vr.odoo.com/hardware.
2019a. Sil-PoxySilicone Adhesive. https://www.smooth-on.com/product-line/sil-poxy/.
2019b. Silbione RTV 4420. https://silicones.elkem.com/EN/our$_o$ffer/Product/90060082/90060081/SILBIONE-RTV-4420-B-U1.
2019. Stretchsense. https://www.youtube.com/watch?v=XJCqdpzxMME.
2019. Vicon. http://www.vicon.com.
2019. VPL Data Glove. https://www.britannica.com/technology/VPL-DataGlove.

Fig. 16. Key pose comparison of different gloves: ManusVR, CyberGlove and ours; we show a video frame of the pose, the pose as captured by the glove, and the mean angular error for this specific pose. Note how our glove has the lowest error for every pose but one (first row). To find similar poses, the lowest mean pose parameter difference is used.

Oluwaseun A. Araromi, Samuel Rosset, and Herbert R. Shea. 2015. High-Resolution, Large-Area Fabrication of Compliant Electrodes via Laser Ablation for Robust, Stretchable Dielectric Elastomer Actuators and Sensors. *ACS Applied Materials & Interfaces* 7, 32 (2015), 18046–18053.

Asli Atalay, Vanessa Sanchez, Ozgur Atalay, Daniel M. Vogt, Florian Haufe, Robert J. Wood, and Conor J. Walsh. 2017. Batch Fabrication of Customizable Silicone-Textile Composite Capacitive Strain Sensors for Human Motion Tracking. *Advanced Materials Technologies* 2, 9 (2017).

Luca Ballan, Aparna Taneja, Jürgen Gall, Luc Van Gool, and Marc Pollefeys. 2012. Motion capture of hands in action using discriminative salient points. In *Proc. ECCV*. Springer, 640–653.

Yujun Cai, Liuhao Ge, Jianfei Cai, and Junsong Yuan. 2018. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *Proc. ECCV*. Springer, Cham, 1–17.

Géry Casiez, Nicolas Roussel, and Daniel Vogel. 2012. Filter: A Simple Speed-based Low-pass Filter for Noisy Input in Interactive Systems. In *Proc. of the Conf. on Human Factors in Computing Systems*. 2527–2530.

Ke-Yu Chen, Shwetak N. Patel, and Sean Keller. 2016. Finexus: Tracking Precise Motions of Multiple Fingertips Using Magnetic Sensing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 1504–1514. https://doi.org/10.1145/2858036.2858125

Jean-Baptiste Chossat, Yiwei Tao, Vincent Duchaine, and Yong-Lae Park. 2015. Wearable soft artificial skin for hand motion detection with embedded microfluidic strain sensing. In *Proc. ICRA*. IEEE, 2568–2573.

Te-Shun Chou, Ashley Gadd, and Dave Knott. 2000. Hand-Eye: A Vision-Based Approach to Data Glove Calibration. In *Proc. Human Interface Technologies*.

Simone Ciotti, Edoardo Battaglia, Nicola Carbonaro, Antonio Bicchi, Alessandro Tognetti, and Matteo Bianchi. 2016. A Synergy-Based Optimally Designed Sensing Glove for Functional Grasp Recognition. *Sensors* 16 (06 2016), 811.

James Connolly, Joan Condell, Brendan O'Flynn, Javier Torres Sanchez, and Philip Gardiner. 2018. IMU Sensor-Based Electronic Goniometric Glove for Clinical Finger Movement Analysis. *IEEE Sensors Journal* 18, 3 (Feb 2018), 1273–1281.

Laura Dipietro, Angelo M. Sabatini, and Paolo Dario. 2008. A Survey of Glove-Based Systems and Their Applications. *Trans. Sys. Man Cyber Part C* 38, 4 (2008), 461–482.

Ali Erol, George Bebis, Mircea Nicolescu, Richard D Boyle, and Xander Twombly. 2007. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding* 108, 1 (2007), 52–73.

Bin Fang, Fuchun Sun, Huaping Liu, and Di Guo. 2017. Development of a Wearable Device for Motion Capturing Based on Magnetic and Inertial Measurement Units. *Scientific Programming* 2017 (01 2017), 1–11.

Max Fischer, Patrick van der Smagt, and Gerd Hirzinger. 1998. Learning techniques in a dataglove based telemanipulation system for the DLR hand. In *Proc. ICRA*, Vol. 2. IEEE, 1603–1608.

Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. 2017. 3D Convolutional Neural Networks for Efficient and Robust Hand Pose Estimation from Single Depth Images. In *Proc. CVPR*.

Reinhard Gentner and Joseph Classen. 2008. Development and evaluation of a low-cost sensor glove for assessment of human finger movements in neurophysiological settings. *Journal of neuroscience methods* 178 (12 2008), 138–47.

Oliver Glauser, Daniele Panozzo, Otmar Hilliges, and Olga Sorkine-Hornung. 2019. Deformation Capture via Soft and Stretchable Sensor Arrays. *ACM Transactions on Graphics* 38 (03 2019), 1–16.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proc. of the Int. Conf. on Artificial Intelligence and Statistics*. 249–256.

Weston Griffin, Ryan Findley, Michael Turner, and Mark Cutkosky. 2000. Calibration and mapping of a human hand for dexterous telemanipulation. In *ASME IMECE Symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems*.

Frank L Hammond, Yiğit Mengüç, and Robert J Wood. 2014. Toward a modular soft sensor-embedded glove for human hand motion and tactile pressure measurement. In *Proc. IROS*. IEEE, 4000–4007.

Shangchen Han, Beibei Liu, Robert Wang, Yuting Ye, Christopher D. Twigg, and Kenrick Kin. 2018. Online Optical Marker-based Hand Tracking with Deep Labels. *ACM Trans. Graph.* 37, 4 (July 2018), 166:1–166:10.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proc. CVPR*.

Ronan Hinchet, Velko Vechev, Herbert Shea, and Otmar Hilliges. 2018. DextrES: Wearable Haptic Feedback for Grasping in VR via a Thin Form-Factor Electrostatic Brake. In *Proc. UIST*.

Haiying Hu, Xiaohui Gao, Jiawei Li, Jie Wang, and Hong Liu. 2004. Calibrating human hand for teleoperating the HIT/DLR hand. In *Proc. ICRA*, Vol. 5. 4571–4576 Vol.5.

Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J Black, Otmar Hilliges, and Gerard Pons-Moll. 2018. Deep inertial poser: learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Trans. Graph.* 37, 6 (2018).

Umar Iqbal, Pavlo Molchanov, Thomas Breuel, Juergen Gall, and Jan Kautz. 2018. Hand Pose Estimation via Latent 2.5 D Heatmap Regression. *arXiv preprint arXiv:1804.09534* (2018).

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. 2017. Image-To-Image Translation With Conditional Adversarial Networks. In *Proc. CVPR*.

Lisa K Simone, Nappinnai Sundarrajan, Xun Luo, Yicheng Jia, and Derek Kamper. 2007. A low cost instrumented glove for extended monitoring and functional hand assessment. *Journal of neuroscience methods* 160 (04 2007), 335–48.

Ferenc Kahlesz, Gabriel Zachmann, and Reinhard Klein. 2004. 'Visual-fidelity'dataglove calibration. In *Computer Graphics International, 2004.* IEEE, 403–410.

G. Drew Kessler, Larry F. Hodges, and Neff Walker. 1995. Evaluation of the CyberGlove As a Whole-hand Input Device. *ACM Trans. Comput.-Hum. Interact.* 2, 4 (1995), 263–283.

David Kim, Otmar Hilliges, Shahram Izadi, Alex D. Butler, Jiawen Chen, Iason Oikonomidis, and Patrick Olivier. 2012. Digits: Freehand 3D Interactions Anywhere Using a Wrist-worn Gloveless Sensor. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 167–176. http://doi.acm.org/10.1145/2380116.2380139

Rebecca K Kramer, Carmel Majidi, Ranjana Sahai, and Robert J Wood. 2011. Soft curvature sensors for joint angle proprioception. In *Proc. IROS*. IEEE, 1919–1926.

Bor-Shing Lin, I-Jung Lee, Shu-Yu Yang, Yi-Chiang Lo, Junghsi Lee, and Jean-Lon Chen. 2018. Design of an Inertial-Sensor-Based Data Glove for Hand Function Evaluation. *Sensors* 18 (05 2018), 1545.

Federico Lorussi, Enzo Pasquale Scilingo, Mario Tesconi, Alessandro Tognetti, and Danilo De Rossi. 2005. Strain sensing fabric for hand posture and gesture monitoring. *IEEE transactions on information technology in biomedicine* 9, 3 (2005), 372–381.

Anil Menon, B. Barnes, R. Mills, Cynthia Bruyns, Xander Twombly, J. Smith, Kevin Montgomery, and Richard D. Boyle. 2003. Using Registration, Calibration, and Robotics to Build a More Accurate Virtual Reality Simulation for Astronaut Training and Telemedicine. In *WSCG*.

Hadrien O Michaud, Laurent Dejace, Séverine De Mulatier, and Stéphanie P Lacour. 2016. Design and functional evaluation of an epidermal strain sensing system for hand tracking. In *Proc. IROS*. 3186–3191.

Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. 2018. Ganerated hands for real-time 3d hand tracking from monocular RGB. In *Proc. CVPR*. 49–59.

Markus Oberweger and Vincent Lepetit. 2017. DeepPrior++: Improving Fast and Accurate 3D Hand Pose Estimation. In *International Conference on Computer Vision Workshops*.

Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. 2015. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807* (2015).

Ben O'Brien, Todd Gisby, and Iain A Anderson. 2014. Stretch sensors for human body motion. In *Proc. SPIE*, Vol. 9056. 905618.

Timothy F O'Connor, Matthew E Fach, Rachel Miller, Samuel E Root, Patrick P Mercier, and Darren J Lipomi. 2017. The Language of Glove: Wireless gesture decoder with low-power and stretchable hybrid electronics. *PloS one* 12, 7 (2017), e0179766.

Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. 2011a. Efficient model-based 3D tracking of hand articulations using Kinect. In *Proc. BMVC*.

Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. 2011b. Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *Proc. ICCV*.

Wookeun Park, Kyongkwan Ro, Suin Kim, and Joonbum Bae. 2017. A Soft Sensor-Based Three-Dimensional (3-D) Finger Motion Measurement System. *Sensors (Basel, Switzerland)* 17, 2 (02 2017), 420.

Panagiotis Polygerinos, Zheng Wang, Kevin C. Galloway, Robert J. Wood, and Conor J. Walsh. 2015. Soft robotic glove for combined assistance and at-home rehabilitation. *Robotics and Autonomous Systems* 73 (2015), 135 – 143.

Adnan Rashid and Osman Hasan. 2018. Wearable technologies for hand joints monitoring for rehabilitation: A survey. *Microelectronics Journal* (02 2018).

Javier Romero, Dimitrios Tzionas, and Michael J. Black. 2017. Embodied Hands: Modeling and Capturing Hands and Bodies Together. *ACM Trans. Graph.* 36, 6 (2017).

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Inte. Conf. on Medical image computing and computer-assisted intervention*. Springer, 234–241.

Hochung Ryu, Sangki Park, Jong-Jin Park, and Jihyun Bae. 2018. A knitted glove sensing system with compression strain for finger movements. *Smart Materials and Structures* 27, 5 (2018), 055016.

T. Scott Saponas, Desney S. Tan, Dan Morris, Ravin Balakrishnan, Jim Turner, and James A. Landay. 2009. Enabling Always-available Input with Muscle-computer Interfaces. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, New York, NY, USA, 167–176. https://doi.org/10.1145/1622176.1622208

Toby Sharp, Cem Keskin, Duncan Robertson, Jonathan Taylor, Jamie Shotton, David Kim, Christoph Rhemann, Ido Leichter, Alon Vinnikov, Yichen Wei, et al. 2015. Accurate, robust, and flexible real-time hand tracking. In *Proc. of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 3633–3642.

Zhong Shen, Juan Yi, Xiaodong Li, Mark Hin Pei Lo, Michael ZQ Chen, Yong Hu, and Zheng Wang. 2016. A soft stretchable bending sensor and data glove applications. *IEEE Int. Conf. on Real-time Computing and Robotics (RCAR)* 3, 1 (2016), 22.

Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. 2017. Hand keypoint detection in single images using multiview bootstrapping. In *Proc. CVPR*, Vol. 2.

Ayan Sinha, Chiho Choi, and Karthik Ramani. 2016. Deephand: Robust hand pose estimation by completing a matrix imputed with deep features. In *Proc. CVPR*.

Adrian Spurr, Jie Song, Seonwook Park, and Otmar Hilliges. 2018. Cross-modal Deep Variational Hand Pose Estimation. In *Proc. CVPR*.

Srinath Sridhar, Antti Oulasvirta, and Christian Theobalt. 2013. Interactive Markerless Articulated Hand Motion Tracking using RGB and Depth Data. In *Proc. ICCV*. 8.

Jan Steffen, Jonathan Maycock, Helge Ritter, Honghai Liu, and Daniel Schilberg. 2011. Robust Dataglove Mapping for Recording Human Hand Postures. In *Intelligent Robotics and Applications*.

Xiao Sun, Yichen Wei, Shuang Liang, Xiaoou Tang, and Jian Sun. 2015. Cascaded hand pose regression. In *Proc. CVPR*. 824–832.

Andrea Tagliasacchi, Matthias Schroeder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. 2015. Robust Articulated-ICP for Real-Time Hand Tracking. In *Proc. SGP*.

Danhang Tang, Hyung Jin Chang, Alykhan Tejani, and Tae-Kyun Kim. 2014. Latent Regression Forest: Structured Estimation of 3D Articulated Hand Posture. In *Proc. CVPR*.

Danhang Tang, Jonathan Taylor, Pushmeet Kohli, Cem Keskin, Tae-Kyun Kim, and Jamie Shotton. 2015. Opening the black box: Hierarchical sampling optimization for estimating human hand pose. In *Proc. ICCV*. 3325–3333.

Danhang Tang, Tsz-Ho Yu, and Tae-Kyun Kim. 2013. Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In *Proc. ICCV*. 3224–3231.

Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Toby Sharp, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, Arran Topalian, Erroll Wood, Sameh Khamis, Pushmeet Kohli, Shahram Izadi, Richard Banks, Andrew Fitzgibbon, and Jamie Shotton. 2016. Efficient and Precise Interactive Hand Tracking Through Joint, Continuous Optimization of Pose and Correspondences. *ACM Trans. Graph.* 35, 4 (2016), 143:1–143:12.

Jonathan Taylor, Vladimir Tankovich, Danhang Tang, Cem Keskin, David Kim, Philip Davidson, Adarsh Kowdle, and Shahram Izadi. 2017. Articulated Distance Fields for

Ultra-fast Tracking of Hands Interacting. *ACM Trans. Graph.* 36 (2017), 244:1–244:12.
Anastasia Tkach, Mark Pauly, and Andrea Tagliasacchi. 2016. Sphere-meshes for Real-time Hand Modeling and Tracking. *ACM Trans. Graph.* 35, 6 (2016), 222:1–222:11.
Anastasia Tkach, Andrea Tagliasacchi, Edoardo Remelli, Mark Pauly, and Andrew Fitzgibbon. 2017. Online Generative Model Personalization for Hand Tracking. *ACM Trans. Graph.* 36, 6 (2017).
Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. 2014. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Trans. Graph.* 33, 5 (2014), 169.
Hoang Truong, Shuo Zhang, Ufuk Muncuk, Phuc Nguyen, Nam Bui, Anh Nguyen, Qin Lv, Kaushik Chowdhury, Thang Dinh, and Tam Vu. 2018. CapBand: Battery-free Successive Capacitance Sensing Wristband for Hand Gesture Recognition. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems (SenSys '18)*. ACM, New York, NY, USA, 54–67. http://doi.acm.org/10.1145/3274783.3274854
Timo von Marcard, Bodo Rosenhahn, Michael J Black, and Gerard Pons-Moll. 2017. Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. *Comput. Graph. Forum* 36, 2, 349–360.
Chengde Wan, Thomas Probst, Luc Van Gool, and Angela Yao. 2017. Crossing Nets: Combining GANs and VAEs With a Shared Latent Space for Hand Pose Estimation. In *Proc. CVPR*.
Chengde Wan, Angela Yao, and Luc Van Gool. 2016. Hand pose estimation from local surface normals. In *European Conference on Computer Vision*. Springer, 554–569.
Robert Y. Wang and Jovan Popović. 2009. Real-time Hand-tracking with a Color Glove. *ACM Trans. Graph.* 28, 3 (2009), 63:1–63:8.
Yingying Wang and Michael Neff. 2013. Data-driven Glove Calibration for Hand Motion Capture. In *Proc. SCA*. 15–24.
Jiawei Zhang, Jianbo Jiao, Mingliang Chen, Liangqiong Qu, Xiaobin Xu, and Qingxiong Yang. 2016. 3D Hand Pose Tracking and Estimation Using Stereo Matching. *arXiv preprint arXiv:1610.07214* (2016).
Yang Zheng, Yu Peng, Gang Wang, Xinrong Liu, Xiaotong Dong, and Jue Wang. 2016. Development and evaluation of a sensor glove for hand function assessment and preliminary attempts at assessing hand coordination. *Measurement* 93 (06 2016).
Jilin Zhou, François Malric, and Shervin Shirmohammadi. 2010. A New Hand-Measurement Method to Simplify Calibration in CyberGlove-Based Virtual Rehabilitation. *IEEE Transactions on Instrumentation and Measurement* 59, 10 (Oct 2010), 2496–2504.
Christian Zimmermann and Thomas Brox. 2017. Learning to Estimate 3D Hand Pose from Single RGB Images. In *Proc. ICCV*. 4913–4921.

A NETWORK DETAILS

Our model is implemented in Pytorch and trained on an NVIDIA 1080Ti GPU. In the following, we describe the different network architectures for the size of 13M parameters. Networks of other sizes (3M and 50M) have the same structure but different numbers (or sizes) of layers.

For the FCN, we use the same network architecture as [Glauser et al. 2019], i.e., five fully connected layers: F44-F2048-F2048-F2048-F2048-F1024-F26. For the LSTM, we use two hidden layers, and each layer has 512 features in the hidden state. We use a window size of 5 and observe that the bigger the window size, the smoother the reconstruction, but also the lower the reconstruction accuracy. We use a standard SGD optimizer with a learning rate of 0.01 and batch size of 1024 for both the FCN and LSTM.

For ResNet, we use a 2-stride convolution and a 2-stride up-convolution for both the encoder and decoder networks and 12 residual blocks in-between. The architecture of U-Net is shown in Table 6. The generator G of CGAN has the same structure as U-Net, i.e., C64-C128- C256-C512-C512-C256-C128-C64, while the discriminator D has five convolution layers: C64-C128-C256-C512-C1.

We use the ADAM optimizer ($lr = 0.0002, \beta_1 = 0.5, \beta_2 = 0.999$) for training the CNN networks. Xavier [Glorot and Bengio 2010] is used for weights initialization. We use a batch size of 1024 and 256 for training the generic and personalized models, respectively. We choose the model that has the minimal error in the validation set for testing. In general, the training of personalized and generic models takes around 2 and 5 hours, respectively, except that the training time of LSTM is about three-fold. The inference time of a trained model is approximately 0.003 seconds.

Table 6. Network architecture of U-Net. N: the number of output channels, K: kernel size, S: stride size, P: padding size, BN: batch normalization.

Input → Output shape	Layer information
(5, 5, 2) → (5, 5, 64)	CONV-(N64, K5×5, S1, P2), ReLU
(5, 5, 64) → (3, 3, 128)	CONV-(N128, K3×3, S2, P1), BN, ReLU
(3, 3, 128) → (2, 2, 256)	CONV-(N256, K3×3, S2, P1), BN, ReLU
(2, 2, 256) → (1, 1, 512)	CONV-(N512, K4×4, S2, P1), BN, ReLU
(1, 1, 512) → (2, 2, 256)	CONV-(N512, K4×4, S2, P1), BN, ReLU
(2, 2, 256) → (3, 3, 128)	CONV-(N256, K3×3, S2, P1), BN, ReLU
(3, 3, 128) → (5, 5, 64)	CONV-(N128, K3×3, S2, P1), BN, ReLU
(5, 5, 64) → (5, 5, 1)	CONV-(N64, K5×5, S1, P2), Tanh

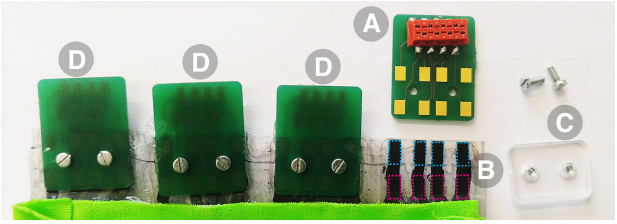

Fig. 17. (A) Connector board with eight pads and connector socket. (B) Exposed lead ends of the bottom (blue) and top (red) layer. (C) Acrylic counter-holder. (D) Three installed connector boards.

B FABRICATION DETAILS

B.1 Silicone mixtures

We employ the silicone mixtures suggested by [Glauser et al. 2019]. For the protective layer Silbione RTV 4420 [Sil 2019b] component A (weight ratio=1.0) and Toluol (1.0) are mixed and in a second step Silbione RTV 4420 (1.0) component B is added. For the conductive layer Silbione RTV 4420 component A (1.0) and Toluol (2.0) are mixed, then Silbione RTV 4420 (1.0) component B is added. Separately, Imerys Enasco 250 P [Ens 2019] conductive carbon black (0.2) is mixed with isopropyl alcohol (2.0). The isopropyl alcohol is added slowly while stirring. Finally, both compositions are combined and mixed for about 3 minutes. The dielectric layer is made from the same mixture as the protective layer.

B.2 Interconnections

To connect the individual leads of the fully soft silicone sensor to the read-out circuit (see Appendix B of [Glauser et al. 2019] for details) rigid printed circuit boards (PCB) are placed on the exposed sensor leads at the wrist end of the glove, supported by a PET foil and screwed into an acrylic counter-holder, see Fig. 17. The PET foil acts as intermediary from stretchable (silicone sensor), through flexible (PET), to fully rigid connector PCBs.