Unsupervised Incremental Learning for Hand Shape and Pose Estimation

Pratik Kalshetti
Indian Institute of Technology Bombay
pratikm@cse.iitb.ac.in

Parag Chaudhuri
Indian Institute of Technology Bombay
paragc@cse.iitb.ac.in

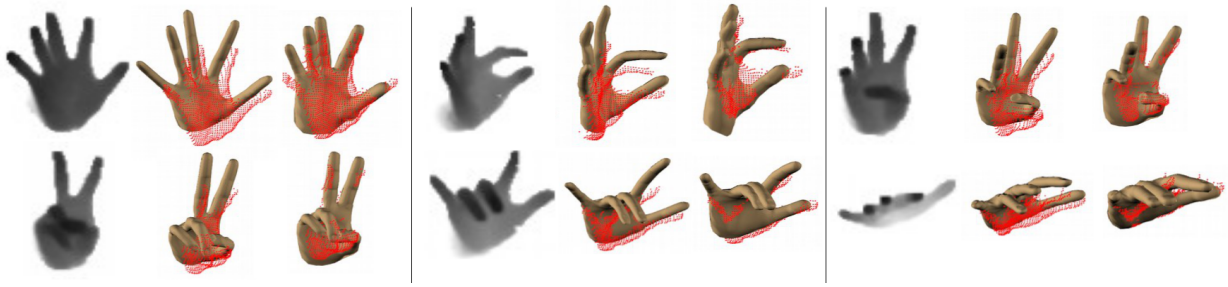

Figure 1: We show depth image frames acquired using a Kinect sensor from a real user. The *BaseNet* predictions and the improved *RefNet* predictions for hand shape and pose are shown next. The red point cloud is recovered from the depth image.

ABSTRACT

We present an unsupervised incremental learning method for refining hand shape and pose estimation. We propose a refiner network (*RefNet*) that can augment a state-of-the-art hand tracking system (*BaseNet*) by refining its estimations on unlabeled data. At each input depth frame, the estimations from the *BaseNet* are iteratively refined by *RefNet* using a model-fitting strategy. During this process, the *RefNet* adapts to the input data characteristics by incremental learning. We show that our method provides more accurate hand shape and pose estimates on both a standard dataset and real data.

CCS CONCEPTS

• **Computing methodologies** → **Shape inference**; *Online learning settings*; • **Human-centered computing** → Mixed / augmented reality.

KEYWORDS

incremental learning, model fitting, hand shape and pose estimation

ACM Reference Format:
Pratik Kalshetti and Parag Chaudhuri. 2019. Unsupervised Incremental Learning for Hand Shape and Pose Estimation. In *Proceedings of SIGGRAPH '19 Posters.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3306214.3338553

1 MOTIVATION

Accurate hand tracking is the key to achieve next generation human-computer interaction and virtual reality interfaces. Most existing approaches to hand tracking use neural networks for pose and shape estimation that require large supervised data for training but their accuracy is limited on unseen data. [Taylor et al. 2016] refines these estimates using an iterative model fitting stage for each input. However, it requires an offline model calibration phase. [Tkach et al. 2017] introduce the idea of online calibration but at the cost of substantial computational overhead and the method relies on accurate tracking being available during calibration. A refinement network was also used by [Oberweger et al. 2015], however it requires supervised data and a differentiable depth rendering. We introduce a refiner network that is capable of adapting to the input data characteristics without any supervision. Our method is capable of using traditional energy functions used by state-of-the-art hand tracking systems, in a neural network framework. We also devise a novel approach that utilizes a dense fitting loss without the need for a differentiable depth renderer.

SIGGRAPH '19 Posters, July 28 - August 01, 2019, Los Angeles, CA, USA

ACM ISBN 978-1-4503-6314-3/19/07.
https://doi.org/10.1145/3306214.3338553

2 METHOD

Given a depth image containing the hand, *BaseNet* predicts coarse shape $\mathbf{s}_0 \in \mathbb{R}^{n_s}$ and pose $\mathbf{p}_0 \in \mathbb{R}^{n_p}$ (n_s and n_p are number of shape and pose parameters). *BaseNet* can be any state-of-the-art hand tracking system that estimates hand shape and pose. We use a trained *BaseNet* as is without changing any of its components. These coarse estimates are iteratively refined by *RefNet* to produce $\mathbf{s}_n$ and $\mathbf{p}_n$ by minimizing an energy function E. During this refinement, the parameters of *RefNet* are also updated in an *unsupervised, incremental* manner, as shown in Figure 2. The qualitative improvement in results can be clearly seen in Figure 1.

2.1 Energy Function

The overall energy function $E = E_{fit} + \alpha E_{reg}$, involves a model-fitting term E_{fit} and a regularizer term E_{reg}. α is a scalar hyper-parameter, set to 0.1. The model fitting error, $E_{fit} = E_{m2d} + E_{d2m}$ captures the error between the sensor point cloud $\mathbf{d}$ (obtained from

input depth image) and the predicted mesh **m**. Here the first term ensures that the data points are explained by the model and the second term ensures that the model lies in the sensor visual-hull. These energy terms are detailed in [Tkach et al. 2017].

If the model-fitting energy terms are computed by rendering the model and using the depth buffer to sample points on it, the process is computationally expensive and is non-differentiable. In order to update the parameters of *RefNet* using E_{fit}, we require a differentiable function that can compute dense points on the mesh. We know that *BaseNet* estimates shapes by predicting shape parameters that are applied to a template hand mesh to get the final shape. We use a pre-computed area weighted surface sampling of the template hand mesh model and store them as barycentric coordinates. Since these samples are now represented as a convex combination of vertices of the mesh, the representation is differentiable. At runtime, we use the stored barycentric coordinates to compute sample point coordinates on the predicted hand mesh. Thus, we can use these model-fitting terms to train a neural network on unsupervised data. Only points facing the camera are considered for fitting.

E_{reg} ensures that the refinements do not substantially deviate from the predictions of *BaseNet*. This plays an important role to avoid catastrophic failure [Shmelkov et al. 2017] in the incremental learning of *RefNet*.

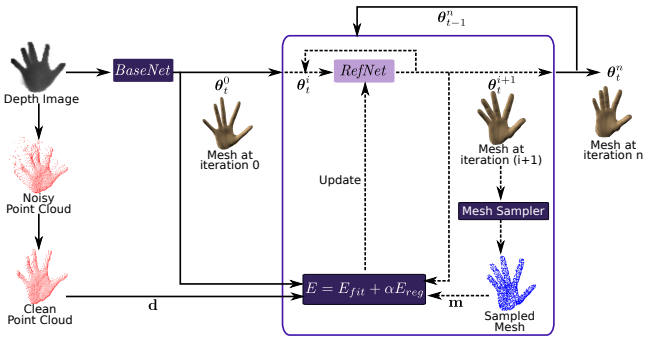

Figure 2: RefNet Pipeline

2.2 *RefNet* Architecture and Training

The architecture of *RefNet* consists of two dense layers with a ReLU non-linearity in between. This simple network can be trained efficiently on unsupervised data without affecting the performance of *BaseNet*. We use separate networks for refining pose and shape, with similar architecture. For each input frame, *RefNet* is iteratively trained using the energy function E for n iterations. At each iteration i, the input to *RefNet* are current frame's estimate $\boldsymbol{\theta}_t^i$ and previous frame's final refined estimate $\boldsymbol{\theta}_{t-1}^n$. It outputs the current frame's refined estimate $\boldsymbol{\theta}_t^{i+1}$, where $\boldsymbol{\theta} \in \{\mathbf{s}, \mathbf{p}\}$. Using $\boldsymbol{\theta}_{t-1}^n$ incorporates a measure of temporal smoothing into our predictions. This process is illustrated in Figure 2.

The parameters of *RefNet* are initialized using supervised shape and pose data. This pre-training is crucial for faster convergence of energy function on unsupervised data.

Table 1: Effect of iterations on Fitting Accuracy

Iterations	E_{m2d} (in mm)	E_{d2m} (in mm)
0	23.36	25.63
1	17.94	21.99
2	16.96	21.30
10	12.87	17.56
50	9.15	12.72
100	8.98	12.49

3 EVALUATION

We evaluate on unlabeled, real data acquired from a Kinect sensor. The model-fitting energy terms E_{m2d} and E_{d2m} are used as evaluation metric. In these tests *RefNet* was run for 100 iterations on the first frame, and then for a specified number of iterations on every subsequent frame. The initial 100 iterations stabilize the shape refinement. We study the effect of number of iterations of refinement per frame on the fitting accuracy in Table 1. The fitting accuracy improves as the number of iterations increases, at the cost of computation time. These values are averaged over a sequence of arbitrary poses, thus validating that there is no catastrophic failure in the incremental learning framework.

We also evaluate our approach on the NYU [Tompson et al. 2014] hand dataset. Mean E_{m2d} with *BaseNet* i.e., without refinement is 11.4*mm*. After refinement with 10 iterations per frame, it reduced to 5.5*mm*. Mean E_{d2m} similarly reduced from 17.6*mm* to 16.4*mm*. Mean E_{d2m} is higher than E_{m2d} because the depth images are noisy.

4 CONCLUSION

We demonstrate the first unsupervised incremental learning strategy for refining hand shape and pose estimates. We incorporate traditional energy functions in a neural network framework for refinement using unlabeled data. Our refinement strategy can be used to augment any state-of-the-art hand tracking system. It can also be useful in creating labeled data in the wild, which is extremely important for data driven hand tracking systems.

A limitation of our method is that the current prototype implementation takes 0.3 seconds for each iteration on the CPU. This is a bottleneck for large number of iterations to achieve better fits. We want to improve this with a more robust, optimized implementation.

REFERENCES

Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. 2015. Training a feedback loop for hand pose estimation. In *Proc. IEEE CVPR*. 3316–3324.

Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. 2017. Incremental learning of object detectors without catastrophic forgetting. In *Proc. IEEE CVPR*. 3400–3409.

Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Toby Sharp, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, et al. 2016. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. *ACM ToG* 35, 4 (2016), 143.

Anastasia Tkach, Andrea Tagliasacchi, Edoardo Remelli, Mark Pauly, and Andrew Fitzgibbon. 2017. Online generative model personalization for hand tracking. *ACM ToG* 36, 6 (2017), 243.

Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. 2014. Real-time continuous pose recovery of human hands using convolutional networks. *ACM ToG* 33 (August 2014), 169:1–169:10.