3D Hand Tracking by Employing Probabilistic Principal Component Analysis to Model Action Priors

Emmanouil Oulof Porfyrakis[1,2], Alexandros Makris[1], and Antonis Argyros[1,2]

[1] Institute of Computer Science, FORTH, Greece
[2] University of Crete, Greece
{porfyrak, amakris, argyros}@ics.forth.gr

Abstract. This paper addresses the problem of 3D hand pose estimation by modeling specific hand actions using probabilistic Principal Component Analysis. For each of the considered actions, a parametric subspace is learned based on a dataset of sample action executions. The developed method tracks the 3D hand pose either in the case of unconstrained hand motion or in the case that the hand is engaged in some of the modelled actions. The tracker uses gradient descent optimization to fit a 3D hand model to the available observations. An online criterion is used to automatically switch between tracking the hand in the unconstrained case and tracking it in the case of learned action subspaces. To train and evaluate the proposed method, we captured a new dataset that contains sample executions of 5 different grasp-like hand actions and hand/object interactions. We tested the proposed method both quantitatively and qualitatively. For the quantitative evaluation we relied on our dataset to create synthetic sequences from which we artificially removed observations to simulate occlusions. The obtained results show that the proposed method improves 3D hand pose estimation over existing approaches, especially in the presence of occlusions, where the employed action models assist the accurate recovery of the 3D hand pose despite the missing observations.

1 Introduction

The problem of effectively inferring the 3D pose of human parts and understanding human actions is a challenging topic in computer vision. In real life, human hands support important functions by executing complex tasks such as object manipulation and sign-based human-to-human communication. By developing technical systems that are able to observe and understand the configuration of human hands we can support applications such as sign language recognition, interactive games or virtual reality environments, robotic arm tele-operation and many others. Such applications typically require high accuracy and robustness. To meet these requirements, many challenges must be addressed such as occlusions, uncontrolled environments and fast hand motions. We focus on the

problem of 3D hand pose estimation and gesture recognition based on modeling and exploiting action priors. More specifically, the goal is to exploit prior knowledge on the hand actions to estimate the hand pose and the performed gesture. Taking into account the high dimensionality of hand models, we use Probabilistic Principal Component Analysis [23], a linear dimensionality reduction technique, combined with gradient based optimization. The input to our approach is RGB image sequences. Prior knowledge in the form of kinematic constraints (average size of an articulated structure, degrees of freedom for each articulation), or motion dynamics (physical laws ruling the object movements, assumptions on grasp movements) may provide rich information and facilitates the solution of the aforementioned problem. In our case, prior knowledge is based on the modeling of a set of predefined actions. The main assumption is that the finger motions are correlated given a particular hand action such as an object grasp. In other words, we assume that a grasp that concerns a particular object type will be performed similarly regardless of the subject that performs it.

1.1 Related Work

A large number of methods have been proposed for solving the 3D hand pose estimation and gesture recognition problems using markerless RGB-D or RGB observations. Several works employ prior information on the hand motion to facilitate and speed-up pose estimation, and/or to deal with missing observations.

Model based approaches use 3D hand models and local optimization to estimate the hand pose. Several optimization algorithms have been proposed, such as Particle Swarm Optimization [18,13], hierarchical particle filters [11], or the quasi-Newton method [4]. Methods that estimate the shape of the hand in addition to the pose by using deformable hand models have also appeared [10,21].

Discriminative approaches attempt to regress the pose directly from observations. Hybrid methods use a discriminative component to extract high level features which are then fed into a generative component. Over the last years, Convolutional Neural Network (CNN) based approaches dominate this category. One direction is to estimate 2D keypoints which are then lifted to 3D [2,16]. The downside of passing through a 2D representation is the presence of projection ambiguities which can be overcome by employing suitable priors. Approaches that rely on RGB-D provide good accuracy and avoid the projection related ambiguities [20,17]. Another approach is to directly estimate the 3D pose from RGB images [12].

For gesture recognition, recent methods rely mostly on CNNs. Liang [9] proposed a multi-view framework for recognizing hand gestures using point clouds captured by a depth sensor. They used CNNs as feature extractors followed by an SVM classifier to classify hand gestures. In [15] they utilized a CNN and stacked a denoising auto-encoder for recognizing 24 hand gestures of the American Sign Language. In [8], they used two CNN architectures, one lightweight CNN architecture for detecting hand gestures and a deep CNN for classifying them.

Several approaches use prior motion information and dimensionality reduction to facilitate hand pose estimation. In [7,5], they employ Principal Component Analysis (PCA) to learn a lower dimensional space that describes compactly and effectively the human hand articulation, thus reducing the computational effort needed for hand poses estimation. In [14], they use PCA to learn subspace models from cyclic motions. Nonlinear dimensionality reduction techniques have also been used, as, for example, in [6] where ST-Isomap is used. However, Isomap and LLE do not provide mapping between the latent space and the data space. Gaussian Process Dynamical Model (GPDM), a nonlinear reduction method, had been applied [19] for 3D human body tracking. Urtasun et al. [24] use a form of probabilistic dimensionality reduction with a GPDM to formulate the tracking as a nonlinear least-squares optimization problem. Tian et al. [22] use Gaussian Process Latent Variable Models (GPLVM) for 2D pose estimation.

Our contribution: This work aims at exploiting prior knowledge about particular hand motions to reduce the dimensionality of the hand pose estimation problem, which (a) speeds-up the tracking and (b) provides robustness to noise and missing observations. The first contribution is the coupling of the state of the art hybrid approach of [16] with probabilistic PCA dimensionality reduction. An additional contribution is the compilation of a dataset comprised of several actions (mostly grasping) executed by multiple actors. The dataset has been used for the training of our method and will become publicly available.

2 Method Description

2.1 Hand Model

The hand model we use (see Fig. 1) is comprised of a kinematic skeleton and a 3D mesh that represents the geometry of its surface. It has 26 degrees of freedom (DOFs), 6 for global position and rotation and 20 for finger articulation. Specifically, the kinematics of each finger is modeled using four parameters, two for the base joint of the finger and one for each of the two remaining joints.

2.2 Action PPCA Training

The Probabilistic Principal Components Analysis (PPCA) requires a dataset of example executions (RGB sequences) of a set of actions. The hand pose in each frame of the dataset is annotated. As described in Sec. 2.1, the hand pose is comprised of a global translation and rotation and the hand joints articulation. The action modeling concerns only the articulation part, so in the following we stripped the global transform DOFs from the hand state. Given the articulation pose sequences, a small number of key poses specific for each motion is identified. Subsequently, the motions are time wrapped so that the key poses are temporally aligned. Furthermore, the number of poses of each sequence is reduced to a predetermined value N. The state of a pose is denoted by t_n where the index n is the action phase with $n \in [1..N]$.

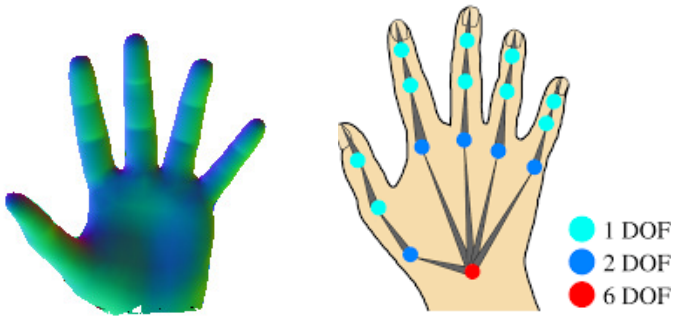

Fig. 1. Hand model: Left: hand shape/geometry, right: hand kinematics.

We used the Expectation-Maximization (EM) algorithm for training a PPCA model for each action. As input, the algorithm takes the state trajectories of a set of sample action executions. Each trajectory state t results from the concatenation of the N hand pose states t_n. The output of EM is the estimated weight matrix W and the variance of noise σ^2. Using these we can convert full dimensional states t_n to reduced dimensional states x_n and vice versa using:

$$x_n = Y^{-1}W^T(t_i - \Theta), \tag{1}$$

$$t_n = x_n * W + \Theta, \tag{2}$$

where Θ is the training states mean and $Y = \sigma^2 I + W^T W$.

2.3 PPCA Hand Tracking

The input to the proposed tracking algorithm is an RGB image and the M action PPCA models. From the image we extract the 2D hand joint locations. These locations are used in an optimization algorithm (see following paragraph) to estimate the hand pose. The optimization is performed $M + 1$ times i.e. on the full dimensional space, and on each of the M modeled sub-spaces. Each optimization provides a candidate solution and the best solution is selected using a method described later in this section. All the steps of the proposed method are summarized in Algorithm 1.

Optimization Algorithm: 3D hand pose estimation is treated as an optimization problem, as in [16]. The input to the optimizer is a set of 2D hand keypoints which are localized in the input image using OpenPose [3]. Typically, the optimization is performed on the full hand state. We also follow this approach to track free hand motion. However, for the pre-modeled actions, we exploit dimensionality reduction to perform optimization on a lower dimensional space.

Given a hand pose and its forward kinematics function, we compute the positions of the joint keypoints $m_i = (u_i, v_i)$, $i \in [1, I]$, $I = 18$, on the image plane.

Algorithm 1 Hand pose estimation.

Initialization: t_{f_0}
for <each frame RGB_f> **do**
  $[t_t^0, S_t^0] = solver(RGB_f, t_{f-1}^0)$ # Sec. 2.3
  for <every model m> **do**
    $[x_t^m, S_f^m] = solver(RGB_f, x_{f-1}^m)$ # Sec. 2.3
    $t_t^m = x_{f-1}^m * W + \Theta_n^m$
  end for
  $m_{sel} = select\_model([x_f^m, S_f^m]_{m=0}^M)$ # Sec. 2.3
  Solution: $t_f^{m_{sel}}$
end for

Let $o_i = (u_i, v_i)$, $i \in [1, I]$, represent the detected 2D joints (using OpenPose) and f_i be a binary flag taking the value of 1 if the i-th keypoint is actually detected and 0 otherwise. For a given pose, the total discrepancy $S(\mathbf{x}, o)$ between the observed and the model joints is given by:

$$S(\mathbf{x}, o) = \sum_{i=1}^{I} f_i \|m_i - o_i\|. \tag{3}$$

The 3D pose x^* that is most compatible with the available observations can be estimated by minimizing the objective function of Equation 3:

$$\mathbf{x}^* = \arg\min_x \{S(\mathbf{x}, o)\}. \tag{4}$$

This is achieved using the Levenberg-Marquardt optimizer that minimizes this objective function after the automatic differentiation of the residuals. In our implementation, optimization has been performed by employing the Ceres Solver [1].

Model Selection: The selection of the appropriate low dimensional model to be used is performed automatically and on-line. The selection relies on the optimization score but for stability we propose a model locking mechanism based on the action phase and the model likelihood.

For each frame we perform the optimization procedure using all the available models (including the full dimensional model aiming at recovering free hand motion). The optimization score S_f^m for each model m approximates the degree of fit of each model to the observations. We select the model with the minimum score value: $m_{sel} = \arg\min_m \{S_f^m\}_{m=0}^M$. The pose estimation of this model t_f^m is thus the output of the algorithm for the frame f.

The selection procedure based solely on the optimization score is unstable. This is mainly due to the fact that the optimization algorithm relies only on the visible keypoints whose number fluctuates during tracking. To achieve model selection that is robust to the score fluctuations, we use a model locking approach. By this approach, we lock to a specific model if two criteria are met: (i) the model likelihood L^m is above a threshold value and (ii) the action phase n is

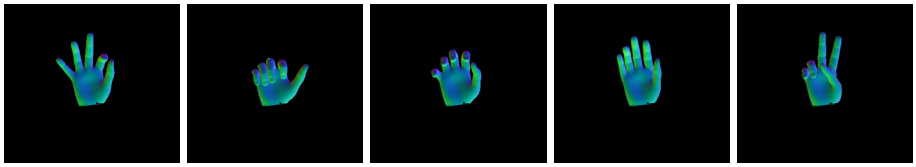

Fig. 2. Grasp actions that have been used in the developed dataset. (a) Pincer grasp, (b) palm grasp, (c) spherical grasp, (d) parallel extension grasp, (e) ring pinch grasp.

above a threshold value. Essentially, these two criteria ensure that if a particular action is detected with a high likelihood and the action execution has advanced considerably, the algorithm will lock the selection to that action model until action completion. The model likelihood is given by:

$$L^m(x_n) = \exp - \frac{((x_m - \Theta_n^m)C^{m-1}(x_m - \Theta_{n)}^m)^T)}{2}, \tag{5}$$

where Θ_n^m is mean value for model m and C^m is the covariance matrix.

3 Experiments

Dataset: For the purposes of this work, we created a new dataset for training and testing the proposed method. The dataset contains 5 grasping actions performed by 6 subjects, 2 females and 4 males. Every subject repeated each action 6 times. The instructions that had been given to all subjects was the verbal description of the actions they had to perform. This gave the opportunity to have action executions with considerable variability. Characteristic snapshots of the specified set of actions are shown in Fig. 2. For every action in the dataset the hand starts from a neutral (open) configuration.

To enable the quantitative evaluation of the proposed method, we used the real world dataset to create a synthetic one. To do so, we tracked the hands in the real dataset to obtain 3D hand poses that we considered as ground truth. We then used the known camera parameters to project the 3D joint locations extracted from the aforementioned ground truth poses back to the image. We provide these 2D image locations as input to the method. To simulate occlusions, we selectively removed some of the 2D keypoints from the input that is provides to the evaluated methods.

Evaluated Methods: We implemented and evaluated the following methods:

- **LEV**: Levenberg-Marquardt optimization without dimensionality reduction.
- **APLEV**: Proposed optimization, exploiting the dimensionality reduction with automatic selection of the modeled actions.
- **SPLEV**: Proposed optimization exploiting the dimensionality reduction assuming knowledge (from the ground truth) of the performed actions.

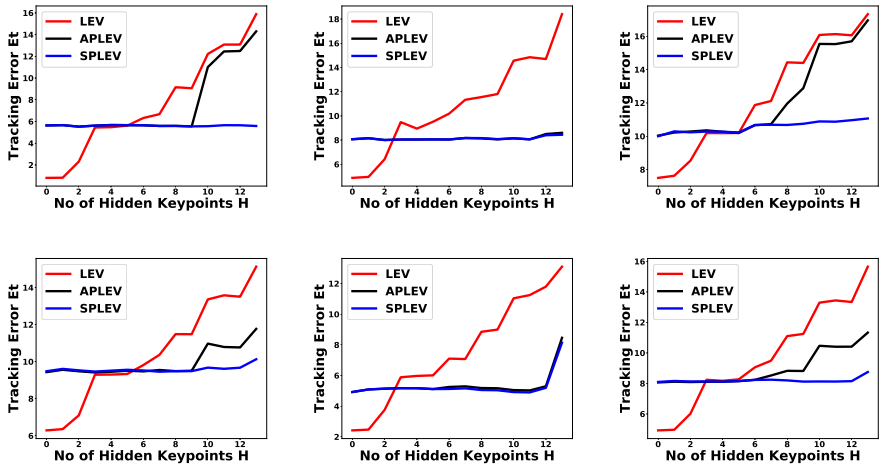

Fig. 3. The tracking error E_t as a function of the occlusion ratio. Each plot concerns sequences of a particular action: (a) pincer grasp, (b) palm grasp, (c) spherical grasp, (d) parallel extension grasp, (e) ring pinch grasp, (f) average over all grasps.

Table 1. Average hand action classification accuracy A_c as a function of the number H of hidden keypoints.

H	**0**	**3**	**6**	**9**	**12**
Pincer grasp	0.99	0.98	0.95	0.64	0.46
Palmer grasp	0.99	0.97	0.93	0.89	0.80
Spherical grasp	0.78	0.71	0.50	0.38	0.37
Parallel extension grasp	0.79	0.71	0.63	0.57	0.41
Ring pinch grasp	0.86	0.79	0.77	0.69	0.54
Average	**0.86**	**0.84**	**0.75**	**0.64**	**0.52**

For the low dimensional sub-spaces of **APLEV** and **SPLEV** methods we used 10 dimensions, 7 for global position and rotation, 2 for the articulation, and 1 for the action phase. Moreover, the likelihood threshold for the experiments are 0.55 and the phase threshold 0.31.

Quantitative Evaluation: We evaluated the methods quantitatively using the synthetic dataset described in Sec. 3. We measured the tracking error E_t which is defined as the average 3D distance between the estimated 3D joint locations and their corresponding ground truth values. We also measured the action classification accuracy A_c which is the percentage of of frames that were classified to the correct action class.

In order to assess the ability of the methods to deal with occlusions, the tracking error was measured for different occlusion ratios. As mentioned in Sec. 3, to simulate occlusions we ignore a number H of 2D keypoints. In Fig. 3, we compare the performance of the methods for an occlusion percentage range from

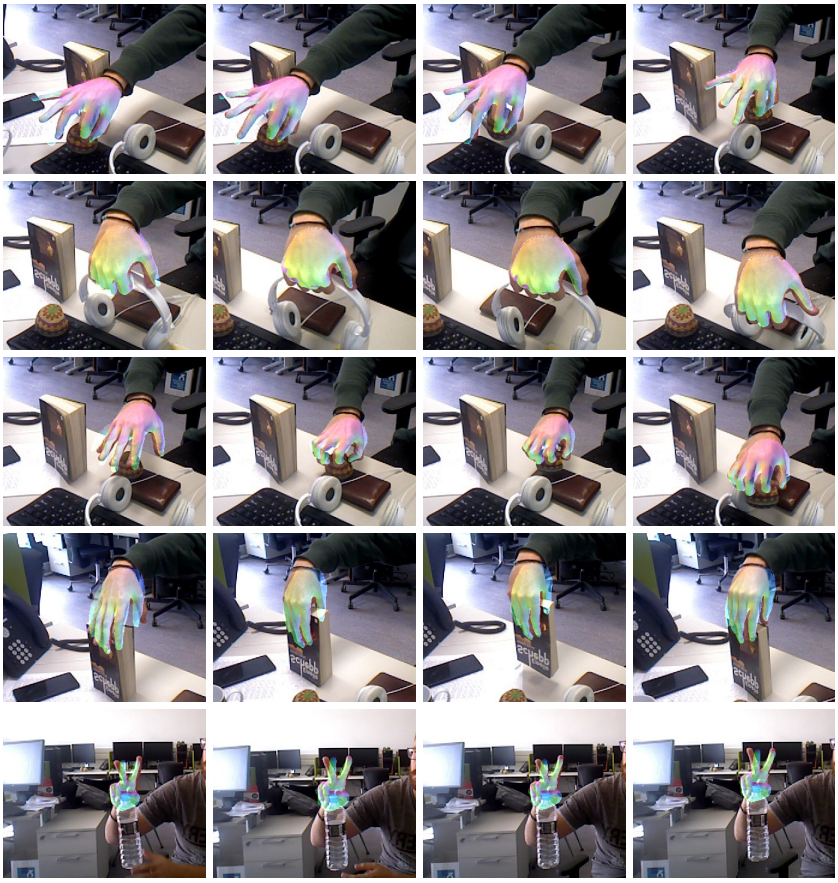

Fig. 4. Qualitative Results for grasping objects using **APLEV**. Every row represents a motion model in different phases.

0% to 60% which corresponds to $H = 0$ up to $H = 12$ of hidden 2D keypoints. The results show that the error of the proposed methods is smaller compared to the baseline method **LEV** if 4 or more keypoints are hidden. For the majority of the modeled actions, automatic model selection performs well and therefore the error of **APLEV** is on par with that of **SPLEV**. In two of the action classes, model selection does not perform so well, so **APLEV** has inferior performance to **SPLEV**. Nevertheless, it still compares favourably to the performance of the **LEV** baseline method.

The primary goal of the proposed **APLEV** method is to leverage prior knowledge about the performed actions in order to perform better tracking. To do so, it classifies each frame either into one of the modeled actions or as free hand motion. In Table 1 we present the action classification accuracy A_c results for different occlusion ratios. We observe that the classification accuracy remains

high even in the presence of considerable occlusions. At the same time, classification as a function of occlusion vary considerably among different actions.

Qualitative Evaluation: For the qualitative evaluation we used a real world dataset where a hand performed object manipulation of various objects such as books, paper, bottle, small balls and pens. The obtained videos have a length between 250 and 600 frames and each of them contained at least 2 actions. As it can be verified in Fig. 4, the proposed approach captures the configuration of the hand correctly, despite the considerable occlusions between the hand and the manipulated object. More qualitative results are available as a youtube video[3].

4 Summary

We presented a method for markerless, model-based tracking of human 3D hand pose using dimensionality reduction based on action priors. We developed a dataset that contained instances of 5 action models and performed Probabilistic Principal Component Analysis to model them. The obtained quantitative and qualitative results demonstrate that the proposed approach manages to track the 3D pose of a hand robustly, even in the presence of considerable occlusions due to hand-object interactions. We intend to increase the grasp type action models so as to to have a more complete relevant dataset. We also plan to incorporate object detection methods and enrich our method by exploiting fingertip/object contact points as location priors. Another future research direction is the exploitation of the proposed approach for 3D human body tracking.

Acknowledgements

This work was partially supported by the EU H2020 project Co4Robots (Grant No 731869).

References

1. Agarwal, S., Mierle, K., et al.: Ceres solver (2012)
2. Ballan, L., Taneja, A., Gall, J., Van Gool, L., Pollefeys, M.: Motion capture of hands in action using discriminative salient points. In: European Conference on Computer Vision. pp. 640–653. Springer (2012)
3. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7291–7299 (2017)
4. de La Gorce, M., Fleet, D.J., Paragios, N.: Model-Based 3D Hand Pose Estimation from Monocular Video. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(9), 1793–1805 (sep 2011)
5. Douvantzis, P., Oikonomidis, I., Kyriazis, N., Argyros, A.: Dimensionality reduction for efficient single frame hand pose estimation. In: International Conference on Computer Vision Systems. pp. 143–152. Springer (2013)

[3] https://youtu.be/L09qeohuJ9k.

6. Jenkins, O.C., Matarić, M.J.: A spatio-temporal extension to isomap nonlinear dimension reduction. In: Proceedings of the twenty-first international conference on Machine learning. p. 56. ACM (2004)
7. Kato, M., Chen, Y.W., Xu, G.: Articulated hand tracking by pca-ica approach. In: 7th International Conference on Automatic Face and Gesture Recognition (FGR06). pp. 329–334. IEEE (2006)
8. Köpüklü, O., Gunduz, A., Kose, N., Rigoll, G.: Real-time hand gesture detection and classification using convolutional neural networks. arXiv preprint arXiv:1901.10323 (2019)
9. Liang, C., Song, Y., Zhang, Y.: Hand gesture recognition using view projection from point cloud. In: 2016 IEEE International Conference on Image Processing (ICIP). pp. 4413–4417. IEEE (2016)
10. Makris, A., Argyros, A.: Model-based 3D Hand Tracking with on-line Shape Adaptation. pp. 77.1–77.12. British Machine Vision Association (2015)
11. Makris, A., Kyriazis, N., Argyros, A.A.: Hierarchical particle filtering for 3D hand tracking. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 8–17. IEEE (jun 2015)
12. Mueller, F., Bernard, F., Sotnychenko, O., Mehta, D., Sridhar, S., Casas, D., Theobalt, C.: Ganerated hands for real-time 3d hand tracking from monocular rgb. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
13. Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Efficient model-based 3d tracking of hand articulations using kinect. In: BmVC. vol. 1, p. 3 (2011)
14. Ormoneit, D., Sidenbladh, H., Black, M.J., Hastie, T.: Learning and tracking cyclic human motion. In: Advances in Neural Information Processing Systems. pp. 894–900 (2001)
15. Oyedotun, O.K., Khashman, A.: Deep learning in vision-based static hand gesture recognition. Neural Computing and Applications **28**(12), 3941–3951 (2017)
16. Panteleris, P., Oikonomidis, I., Argyros, A.A.: Using a single rgb frame for real time 3d hand pose estimation in the wild. In: IEEE Winter Conference on Applications of Computer Vision (WACV 2018), also available at Arxiv. pp. 436–445. IEEE, lake Tahoe, NV, USA (March 2018)
17. Poier, G., Schinagl, D., Bischof, H.: Learning Pose Specific Representations by Predicting Different Views (apr 2018)
18. Qian, C., Sun, X., Wei, Y., Tang, X., Sun, J.: Realtime and Robust Hand Tracking from Depth. pp. 1106–1113 (jun 2014)
19. Raskin, L., , .: Dimensionality Reduction for 3D Articulated Body Tracking and Human Action Analysis. Technion-Israel Institute of Technology, Faculty of Computer Science (2010)
20. Sun, X., Wei, Y., Liang, S., Tang, X., Sun, J.: Cascaded hand pose regression. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)
21. Tan, D.J., Cashman, T., Taylor, J., Fitzgibbon, A., Tarlow, D., Khamis, S., Izadi, S., Shotton, J.: Fits Like a Glove: Rapid and Reliable Hand Shape Personalization. Microsoft Research (jun 2016)
22. Tian, T.P., Li, R., Sclaroff, S.: Tracking human body pose on a learned smooth space. Tech. rep., Boston University Computer Science Department (2005)
23. Tipping, M.E., Bishop, C.M.: Mixtures of Probabilistic Principal Component Analyzers. Neural Computation **11**(2), 443–482 (feb 1999)
24. Urtasun, R., Fua, P.: 3d human body tracking using deterministic temporal motion models. In: European conference on computer vision. pp. 92–106. Springer (2004)