Improved CNN-based Marker Labeling for Optical Hand Tracking

Janis Rosskamp[1], Rene Weller[1], Thorsten Kluss[2], Jaime L. Maldonado C.[2], and Gabriel Zachmann[1]

[1] University of Bremen, Computer Graphic and Virtual Reality, Germany
[2] University of Bremen, Cognitive Neuroinformatics, Germany

Abstract. Hand tracking is essential in many applications reaching from the creation of CGI movies to medical applications and even real-time, natural, physically-based grasping in VR. Optical marker-based tracking is often the method of choice because of its high accuracy, the support for large workspaces, good performance, and there is no wiring of the user required. However, the tracking algorithms may fail in case of hand poses where some of the markers are occluded. These cases require a subsequent reassignment of labels to reappearing markers. Currently, convolutional neural networks (CNN) show promising results for this re-labeling because they are relatively stable and real-time capable. In this paper, we present several methods to improve the accuracy of label predictions using CNNs. The main idea is to improve the input to the CNNs, which is derived from the output of the optical tracking system. To do so, we propose a method based on principal component analysis, a projection method that is perpendicular to the palm, and a multi-image approach. Our results show that our methods provide better label predictions than current state-of-the-art algorithms, and they can be even extended to other tracking applications.

Keywords: hand tracking · motion capturing · marker labeling.

1 Introduction

The human hand is the most versatile tool of the human body to interact with the surrounding world, e.g., by grasping or pointing at objects. In order to allow for this most natural interaction method in virtual environments, too, it is necessary to track the human hands and transfer their movements and poses into the VR world to control a virtual hand. A wide variety of different tracking methods have been developed and some commercial products, such as the Oculus Quest, already feature built-in markerless tracking. While tracking methods using single cameras are easy to set up, they can only track the user's hand with moderate accuracy. This is sufficient for basic interactions in virtual environments. Our motivation for developing a high-precision tracking pipeline is the investigation of natural grasping behavior of humans with its variety of complex manipulations and the capability to flexibly adapt to unexpected situations, in order to provide

models that enable a more natural grasping behavior in robotics. Thus, high precision tracking is a crucial precondition for the quality of the underlying datasets that will be generated with human participants.

In general, high precision hand tracking is essential in all scenarios where physically-based grasping is needed to enable dexterous manipulation of objects [21]. Here, hand deformation and friction is taken into account, and accurate hand poses are necessary for a realistic force estimation to guarantee the stability of grasps. High precision tracking is also necessary in other fields like immersive medical training or interactive virtual prototyping that require a precise recognition of hand poses, too.

The only technology available today that can generate the required accuracy is optical marker-based tracking using a multi-camera setup, such as *Optitrack* systems, which can deliver an accuracy in the sub-millimeter range with high frame rates. Typical optical tracking systems like Optitrack usually only track the positions of individual passive markers in 3D; connectivity information has to be computed from these marker positions algorithmically. Initially, *labels* can be assigned to the markers in order to relate them to their semantic positions, e.g., the tip of the thumb or the palm. The human hand, which has 27 degrees of freedom in a relatively small space, requires a dense marker set, especially for high-precision tracking. On the one hand, denser marker sets reduce tracking errors. On the other hand, due to the complex geometry and motion of the hand, markers are often occluded due to self-occlusion. In this case, reappearing markers must be *relabeled* to transfer the motion correctly to a virtual hand in the virtual environment. Especially for virtual reality applications, this relabeling process has to be performed in real-time and should be highly accurate. So, we cannot take subsequent motions into account, which is possible in mo-cap post-processing.

Han et al. [10] have formulated the labeling of dense marker sets as an image keypoint problem. Their basic idea was to solve it using a convolutional neural network (CNN). CNNs work very well for 2D images, but for unstructured 3D problems, CNNs are difficult to apply in real-time. Hence, they decided to project the 3D positions of the markers, delivered by the tracking system, onto a 2D plane and input this 2D image to the CNN. They report remarkable results with a real-time performance.

In this paper, we present several significant improvements over this state-of-the-art labeling prediction. Our main approach is to optimize the transformation of the 3D marker positions to the 2D planes. We propose three methods to optimize this crucial step. First, an obvious idea is to use several random directions instead of a single one and evaluate them in parallel. We combined this idea with an optimization algorithm to identify the best result. Second, we applied a principal component analysis (PCA) on the dense marker set to predict depth images with optimal spatial distributions. Our third method takes into account that typically the palm is easy to identify and unlikely to be occluded. Hence, we chose a projection perpendicular to the palm to increase the labeling prediction. In addition, with the additional knowledge about the projection direction in

case of the palm projection and the PCA, we hypothesized that a CNN trained specifically for this case could achieve better results than the original CNN that was trained for random projections. Consequently, we generated new CNNs for these cases.

We have tested our algorithms with synthetic data and in a real-world hand tracking application. Our results show significant improvements over the current state-of-the-art labeling prediction. Even more so, we were able to increase dramatically the number of tracking frames where *all* marker labels are predicted correctly. This is important measure of assessing labeling accuracy in cases where many markers were occluded and need to be relabeled when they reappear at the same time (e.g., in a motion from fist to open hand). Also, in case of small tracking volumes, the hand can temporarily leave that volume and must be relabeled completely when re-entering. The same can happen for complicated hand poses with a low number of cameras, where many self occlusions occur. In this paper, we have focused on hand tracking, but our methods can be easily generalized to other labeling problems as well.

2 Related Work

Optical motion capturing is used in many areas, for instance, in animation for games and movies, medical studies [6] or virtual assembly [22]. Labeling of passive markers on non-rigid structures is an active field of research. In general, it can be broadly categorized into two categories: tracking of sparse and dense markersets. Sparse markers are often used for large capture volumes and full-body tracking. Compared with dense markers where 19 markers are used for hand tracking, labeling is easier, but small nuances in movement are hard to track with 13 or fewer markers [1].

Meyer et al. [13] used key poses for labeling and a least-squares method to track marker positions to recover from occlusions. Schubert et al. [18] relaxed the requirement for an initial pose and allows nearly arbitrary poses. Aristidou and Lasenby [2] predict positions of occluded markers using a Variable Turn Model within an unscented Kalman filter without assuming any skeleton model. In Alexanderson et al. [1], Gaussian mixture models are used to track sparse markersets in large capture volumes. They do not require any key poses, and the system is stable even when the user leaves and enters the capture volume. After initializing labels, reappearing markers are relabeled in real-time using inverse kinematics (IK) in Maycock et al. [12]. They predict positions of non-critical occluded markers during run time. Ghorbani et al. [8] use permutation learning to automatically label markers without manual initialization for full body tracking. Han et al. [10] use a CNN to label dense markersets by creating depth images from 3D marker positions.

There exist many hand tracking methods besides passive optical marker tracking. Pavllo et al. [16] are using active markers and IMUs to predict motion even when occlusions happen. While this method is accurate, it requires a heavy glove with cables. Sensor-based tracking using stretch-sensing [9] or bend-

sensing gloves [23] is easy to set up, without occlusions and usable in nearly every environment. However, they usually require cables or batteries and the average error of joint angles is 6-8 degrees, even after a user-specific calibration. Recently IMU-based gloves were developed for medical evaluations [5, 11]. Hand tracking with single RGB cameras [14, 19, 20] or depth cameras [3, 7] are the most obtainable methods. In [15] even tightly interacting hands can be tracked. However, they are not suitable for applications that need high precision tracking.

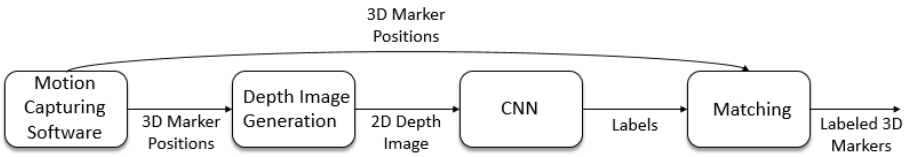

Fig. 1. Overview of our labeling pipeline (adapted from [10]).

3 Our Approach

Optical motion capturing software uses multiple cameras, often in the infrared spectrum, to track a set of reflective markers and computes their 3D positional data. Tracked data is accurate but often suffers from occlusion of markers. In hand tracking, a typical example of occlusions are markers that are too close to the body; hence, cameras cannot see them. When markers become visible again, correct labels must be assigned to every reappearing marker. For instance, the markers on the tip of the index finger must be differentiated from the tip marker on the middle finger but also from the index fingers' carpus marker to avoid wrongly detected hand poses. For the online labeling of markers, [10] proposed a method based on convolutional neural networks for relabeling. We will give a short recap of this method because our approach is inspired by it.

3.1 Recap: CNN-based Marker Labeling

The main idea presented by [10] to solve the relabeling problem is to use CNNs. An outline of the labeling pipeline is shown in Figure 1. In practice, 3D CNNs are too slow for real-time applications [17]; hence, the authors decided to use a network with 2D convolutions. To do that, they transform the 3D marker positions delivered by the tracking software into a 2D image using orthographic projection. The direction of the projection axis is found by creating a random point and determining the direction to the weighted center of the point cloud from this position. Values along the projection axis can be understood as depth values and are normalized in the range $[0.1, 1.0]$. By splatting the markers on

the image, their relative depth is preserved. The resulted image (Figure 2 b)) is used as input for the actual CNN, which predicts a vector of 3D marker positions. The vector elements' order is fixed and corresponds to a label; for instance, the fourth vector element corresponds to the marker on the thumb tip. To assign these labels to the original markers, matching of the predicted 3D positions to the markers' real 3D positions is done by solving a minimum weight bipartite matching problem. The most important step to influence the quality of the resulting label is the generation of the 2D image from the 3D points, i.e., mainly the choice of the projection direction. In [10], the authors simply created ten 2D images using random projection axes (RPA) to label a single frame of markers positions. From these ten images, the one with the highest spatial spread is selected and fed into the CNN. In the following, we propose three methods for image generation which all improve the labeling results compared to RPA.

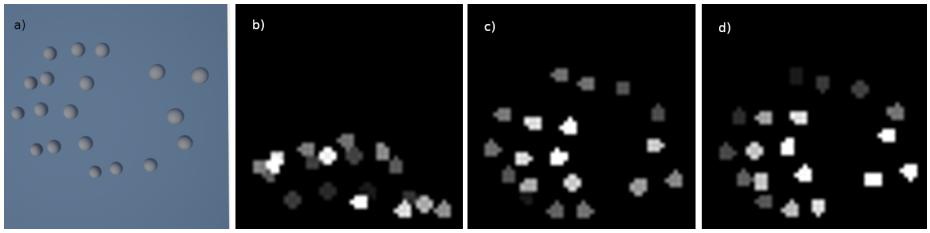

Fig. 2. Image a) shows the point cloud data of the tracked markers for a flat hand pose (small joint angles). To label the 3D positional data, a depth image is created from a). In b), the depth image was created from a random projection direction. Image c) uses a projection axis generated with the PCA method. Image d) was created using a projection axis perpendicular to the palm (PalmP). The depth information is visualized as pixel intensities.

3.2 Our Projection Methods

As mentioned above, the main idea for the improvement of the labeling quality is the choice of an optimized projection direction. We propose three methods that we will detail in the following.

Multi Images - Minimal Cost The RPA method creates ten images from the point cloud but feds only one of them for labeling into the CNN. A straight forward idea for the improvement would be to feed them all into the CNN and choose the best result from the *output* of the network. This consideration is the basis of our *multi images - minimal cost method (Multi)*. Obviously, this requires an appropriate rating function. Moreover, the method can be generalized to create n depth images and choose an optimum number of projection directions based on results from experiments which we will discuss in Section 4. Finally,

we decided to generate the directions for the n images not completely randomly but distribute them uniformly on a sphere around the point cloud's center. All images pass through the network, and we get n vectors of 3D marker data. We use minimum weight bipartite matching to solve the minimum-cost flow problem for all output vectors. In the minimum-cost flow problem, the edges between the initial and predicted 3D marker positions represent distances. Subsequently, selecting the solution with the lowest C corresponds to a matching where the euclidean distance between initial and predicted markers is minimal:

$$\min(C^{(1)}, \ldots, C^{(n)}) \quad \text{with} \quad C^{(i)} = \sum_{j=1} \left\| y_j^{(i)} - x_{M(j)}^{(i)} \right\|_2 . \tag{1}$$

Consequently, we select the set of labels with the lowest matching cost C to get the best fitting solution.

Principal Component Analysis Instead of selecting the highest spatial spread out of a random set of images, we can also calculate a projection plane that yields an image with a high spatial spread for the given marker positions. A traditional method to find such a projection axis is to compute a principal component analysis (PCA) for the 3D points. More precisely, using PCA, we can find the three principal axes of our point cloud. The first two principal axes are pointing in the direction of the highest variance, also called the spatial spread. Consequently, we have chosen the projection direction as the last principal axis. In detail: Our projection vector $\vec{v}$ is given by the eigenvector $\vec{v}_i, i = 1...3$ which corresponds to the smallest eigenvalue λ of the covariance matrix of the point cloud:

$$\vec{p} = \vec{v}_i \quad \text{with} \quad \lambda_i = \min(\lambda_1, \lambda_2, \lambda_3). \tag{2}$$

Figure 2 c) shows a depth image created by the PCA method.

Palm Prediction While the previously proposed methods can be generalized easily to arbitrary labeling problems, our final method is based on the domain knowledge that we are actually tracking a human hand. The idea is to find a projection axis that will lead to similar images independent of the actual hand poses. We decided to define a projection perpendicular to the palm. To get the palm's orientation in our marker point cloud, we require that the markers attached on the back of the hand are identified. Our glove has three rigidly attached markers on the back of the hand (see Figure 7). We can then easily determine these markers under the assumption that the distance between them does not change. Figure 2 d) show an image created with our *palm prediction method (PalmP)*.

3.3 CNN Training

In principle, we could simply reuse the original CNN network proposed by [10]; all our methods are compatible. However, this network is trained with random

projection directions and hence would deliver the best results with this kind of input. In the case of our PCA and especially the Palm Prediction projections, that partly consider domain knowledge, the specifically trained neural network could provide better results which is also supported by our experiments (see Section 4). Hence, we decided to train the network with specifically generated input images for these particular methods.

4 Results

We have evaluated the performance as well as the quality of our CNN-based labeling methods. We have them with both, synthetic data but also in a real VR hand tracking environment. All our experiments were performed on a Linux-PC running Ubuntu 20.04 with an Intel Core i7 3.5GHz, 16 GByte of main memory, and an NVIDIA GTX 1080 Ti GPU with Tensorflow 1.13.1.

4.1 CNN Architecture and Training

In order to guarantee a fair comparison of our methods with the current state-of-the-art labeling method, we decided to use the same network architecture and training data as proposed by [10]. They used a VGG-style neural network with several 3×3 convolutional layers followed by a fully connected layer. As input for the CNN, they used depth images of size 52×52. There is also a training set of 168691 frames of labeled hand configurations available (please note, in [10], there were 170330 frames used), which were synthetically generated from real hand motion of five different users. In [10], the network was trained using random depth images. To increase the labeling accuracy of PCA and PalmP we decided to additionally use networks trained with depth images generated from the PCA and PalmP methods. This increased the accuracy by more than 40 percent points compared to the network trained with random sampling. To avoid overfitting, we split the data into a training and validation set for PCA and PalmP networks, which reduces the training set to 137357 frames. In the following, we evaluate all methods using this synthetic validation data set.

4.2 Synthetic Data

We first evaluated the labeling performance of our networks on the synthetic data set provided by [10]. The results are summarized in the left plot of Figure 3. In comparison to the original RPA projection, our labeling methods improve the number of correctly labeled markers in all cases. For instance, PCA increases the labeling accuracy to 90%, which is an improvement of 14 percent points. It can be directly applied on the point cloud data without the need to know any marker labels beforehand. When labels for the markers for the palm are known, we can apply the PalmP method, which improves the accuracy up to 97%. The accuracy of the Multi method with 20 images is 95%, which is slightly lower than the PalmP results. Similar to PCA, no marker labels or geometric information

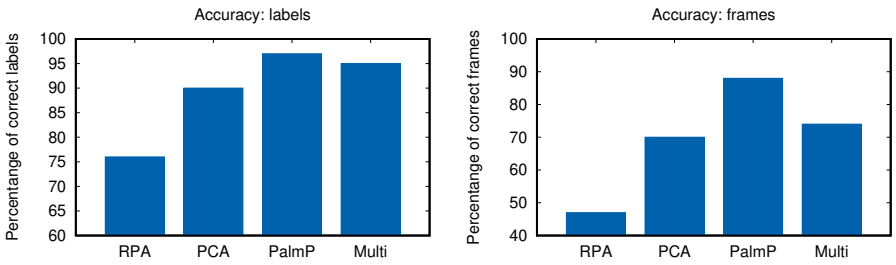

Fig. 3. Using the synthetic validation set, the labeling accuracy of all methods is compared. The left image shows the percentage of correctly labeled markers. The right image shows the percentage of frames, where every marker has the correct label. All our methods improve the current state-of-the-art (RPA). For the Multi method 20 images were used.

is needed. Instead, we pass as many images as possible to the CNN and select the output with the lowest matching cost as our solution. This leads to higher run times, which are investigated in Section 4.4.

Interestingly, our implementation of the orig. RPA method was not able to reproduce the results from [10], where an accuracy of 85-99% was reported. To minimize the chance of implementation errors, we implemented a number of tests for our code and tried both the original released pre-trained network and our own trained network. Even with these discrepancies between the results, it is clear that our projection methods improved the results.

We further evaluate the capacity of the networks to label *all* markers in a frame correctly. This is important in cases where many markers are occluded simultaneously, and all must be relabeled from scratch. As an example, consider the case where only two markers were mislabeled. We would still have a labeling accuracy of 89%, but labels of the two markers are mixed up and hand pose reconstruction would fail. The results are shown in the right of Figure 3. Our methods label between 70% (PCA) and 88% (PalmP) of all frames entirely correctly. This is an improvement of up to 41 percent points compared to the original RPA method.

PalmP In the PalmP method, we use a projection perpendicular to the palm. However, we could also choose other markers to define a coordinate system for the projection, it is not obvious that the projection perpendicular to the palm is best. An example of an alternative would be an axis perpendicular to the plane containing two markers from the back of the hand and one from the thumb tip. To find the optimal projection, we have trained the CNN using other projections and computed the labeling accuracy. Figure 4 shows the results. Indeed, we achieve the best accuracy for the palm projection. We get slightly lower results for a plane containing two palm markers and one at a fingertip. A plane constructed from three fingertip markers produces the worst results. The projection axis using the

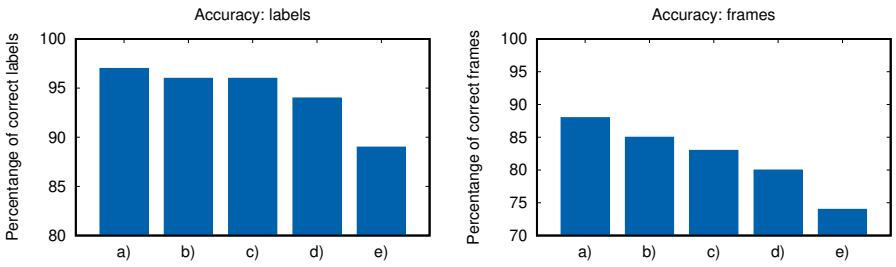

Fig. 4. The labeling accuracy for different image planes for the PalmP method is shown. In a) the three markers on the back of the hand are used. For b)-d) a combination of a marker on the fingertip and two markers on the back of the hand is used: b) is the index finger tip, c) is the thumb tip, and d) the pinky tip. In e) the tips of the thumb, index and pinky finger are used for the image plane.

palm not only produces the best results, but the markers are also the easiest to identify in the point cloud if we use a marker setup, as shown in Fig. 7. Here, the three markers are rigidly attached, so the distance between them remains constant.

Multi The Multi method uses multiple CNN predictions to label a single frame of motion capture data. In this section, we investigate the dependency of the labeling accuracy and the number of predictions. Obviously, the accuracy increases with an increasing number of predictions. Figure 5 shows the labeling accuracy in relation to the number of CNN passes. For a small number of CNN calls, accuracy improves fast, and, if we use five instead of only two calls, our results improve by 6 percent points. On the other hand, the results only change by less than 2 percent points if we use 40 instead of 30 calls. Using 20 CNN passes, we obtain an accuracy of 95% and can still run our labeling step in real-time (see Section 4.4).

4.3 Real Data

We also evaluate the labeling performance of our methods on 1927 frames of real motion capture data. The data was created using a setup of six Optitrack cameras and the glove shown in Figure 7. We decided to use a relatively small number of cameras in order to increase the number of occlusion cases and, thus, stress our hand tracking. Markers on our glove are placed between joints on top of the phalanges to decrease slipping if joints are rotated. An additional inverse kinematics step is necessary to compute the joint angles. Moreover, inverse kinematic is used to check if the labels predicted by the CNN are correct. We use a standard damped least-squares inverse kinematics method [4]. The labeling accuracy for all methods is shown in Figure 6. Similar to the validation with

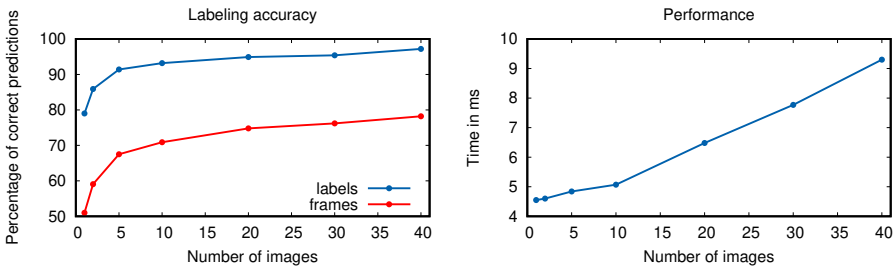

Fig. 5. These plots show the influence of the number of CNN calls on the labeling accuracy (left) and the run-time (right) is investigated for the Multi method. In the left plots, the blue curve denotes the percentage of all correctly labeled markers. The red curve shows the percentage of correct frames, where all markers were assigned a correct label.

synthetic data, we observe that PCA, PalmP, and Multi outperform the standard random projections method. The labeling performance on real data is very similar to the results from synthetic data.

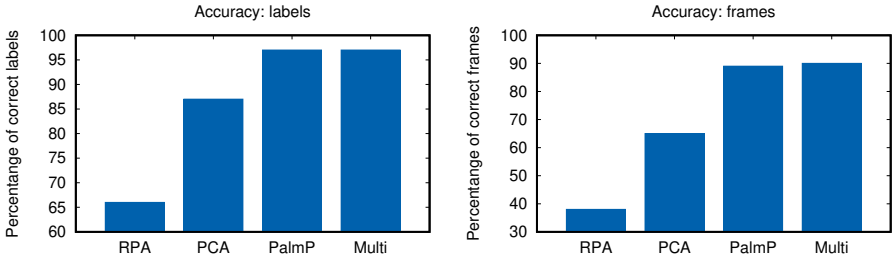

Fig. 6. Using real motion capture data, the labeling accuracy of all methods is compared. The left image shows the percentage of correctly labeled markers. The right image shows the percentage of frames, where every marker has the correct label. For the Multi method 20 images were used.

4.4 Performance

A complete labeling step with PCA or PalmP takes approximately 4.5ms, where image generation and marker matching take around 0.05ms, and the CNN prediction approximately 4.45ms. The Multi method uses multiple CNN calls. The right image of Figure 5 shows the performance of the Multi method with respect to the number of CNN passes. For 40 calls, the prediction requires around 9.3ms. This is only twice the time used for single image labeling and can be explained by batching in the prediction step. Often a dedicated computer for tracking is

Fig. 7. A glove attached with 19 markers for tracking with optical motion capture systems. The three markers on the back of the hand have a fixed distance and can be interpreted as a rigid body.

used, and the joint angles are streamed to the VR application. Hence, the Multi method runs in real-time with 60Hz, even if 20 images are used for labeling.

5 Conclusions and Future Works

We have presented three new methods for CNN-based marker labeling for optical hand tracking. The approach was to transform 3D marker positions into 2D depth images that can be input into a convolutional neural network. The goal was to maximize the spread of the marker positions in the image to achieve best labeling accuracy. To do that, we proposed methods based on PCA, a multi image approach, and a method that also considers domain knowledge for the case of hand tracking (PalmP). Moreover, we have trained two CNNs for the PCA and the PalmP method to further improve the quality of the tracking. Our results show that the PCA method increases the accuracy to 90%, which is an improvement of 14 percent points compared to the state-of-the-art method. It can be applied to every marker set and does not increase runtime. If we have prior knowledge of some marker labels, in our case, the back of the hand, labeling accuracy improves to up to 97% with our PalmP method. If no information about the marker set is available, our multi-projection method achieves similar results to the PalmP method, depending on the number of projections. It also allows for an easy trade-off between performance and accuracy.

Our work also offers interesting avenues for future works: for instance, we want to investigate the labeling accuracy of our methods on non-hand markersets. Transferring the PCA or Multi method to other markersets is straight forward. Using the PalmP method requires a proper projection direction based on domain knowledge for optimal results. Additionally, we want to investigate simultaneous marker labeling of two interacting hands. At the moment, the hands are separated using clustering and then labeled individually using the CNN. An open challenge is the prediction of 3D marker positions of occluded markers during online tracking.

Acknowledgment

The research reported in this paper has been (partially) supported by the German Research Foundation DFG, as part of Collaborative Research Center (Sonderforschungsbereich) 1320 "EASE - Everyday Activity Science and Engineering", University of Bremen (http://www.ease-crc.org/). The research was conducted in subproject H01 <Acquiring activity models by situating people in virtual environments>.

References

1. Alexanderson, S., OSullivan, C., Beskow, J.: Real-time labeling of non-rigid motion capture marker sets. Comput. Graph. **69**(C), 59–67 (Dec 2017). https://doi.org/10.1016/j.cag.2017.10.001, https://doi.org/10.1016/j.cag.2017.10.001
2. Aristidou, A., Lasenby, J.: Real-time marker prediction and CoR estimation in optical motion capture. Vis Comput **29**(1), 7–26 (Jan 2013). https://doi.org/10.1007/s00371-011-0671-y, https://doi.org/10.1007/s00371-011-0671-y
3. Baek, S., Kim, K.I., Kim, T.K.: Augmented Skeleton Space Transfer for Depth-Based Hand Pose Estimation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8330–8339. IEEE, Salt Lake City, UT, USA (Jun 2018). https://doi.org/10.1109/CVPR.2018.00869, https://ieeexplore.ieee.org/document/8578967/
4. Buss, S.R.: Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods p. 19
5. Connolly, J., Condell, J., O'Flynn, B., Sanchez, J.T., Gardiner, P.: IMU Sensor-Based Electronic Goniometric Glove for Clinical Finger Movement Analysis. IEEE Sensors Journal **18**(3), 1273–1281 (Feb 2018). https://doi.org/10.1109/JSEN.2017.2776262, conference Name: IEEE Sensors Journal
6. Galna, B., Barry, G., Jackson, D., Mhiripiri, D., Olivier, P., Rochester, L.: Accuracy of the Microsoft Kinect sensor for measuring movement in people with Parkinson's disease. Gait Posture **39**(4), 1062–1068 (Apr 2014). https://doi.org/10.1016/j.gaitpost.2014.01.008
7. Ge, L., Cai, Y., Weng, J., Yuan, J.: Hand PointNet: 3D Hand Pose Estimation Using Point Sets. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8417–8426 (Jun 2018). https://doi.org/10.1109/CVPR.2018.00878, iSSN: 2575-7075
8. Ghorbani, S., Etemad, A., Troje, N.F.: Auto-labelling of Markers in Optical Motion Capture by Permutation Learning. In: Gavrilova, M., Chang, J., Thalmann, N.M., Hitzer, E., Ishikawa, H. (eds.) Advances in Computer Graphics. pp. 167–178. Springer International Publishing, Cham (2019)
9. Glauser, O., Wu, S., Panozzo, D., Hilliges, O., Sorkine-Hornung, O.: Interactive hand pose estimation using a stretch-sensing soft glove. ACM Trans. Graph. **38**(4), 1–15 (Jul 2019). https://doi.org/10.1145/3306346.3322957, http://dl.acm.org/citation.cfm?doid=3306346.3322957

10. Han, S., Liu, B., Wang, R., Ye, Y., Twigg, C.D., Kin, K.: Online optical marker-based hand tracking with deep labels. ACM Trans. Graph. **37**(4), 1–10 (Jul 2018). https://doi.org/10.1145/3197517.3201399, http://dl.acm.org/citation.cfm?doid=3197517.3201399
11. Lin, B.S., Lee, I.J., Yang, S.Y., Lo, Y.C., Lee, J., Chen, J.L.: Design of an Inertial-Sensor-Based Data Glove for Hand Function Evaluation. Sensors (Basel) **18**(5) (May 2018). https://doi.org/10.3390/s18051545, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5982580/
12. Maycock, J., Rohlig, T., Schroder, M., Botsch, M., Ritter, H.: Fully automatic optical motion tracking using an inverse kinematics approach. In: 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids). pp. 461–466 (Nov 2015). https://doi.org/10.1109/HUMANOIDS.2015.7363590
13. Meyer, J., Kuderer, M., Müller, J., Burgard, W.: Online marker labeling for fully automatic skeleton tracking in optical motion capture. In: 2014 IEEE International Conference on Robotics and Automation (ICRA). pp. 5652–5657 (May 2014). https://doi.org/10.1109/ICRA.2014.6907690, iSSN: 1050-4729
14. 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: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 49–59. IEEE, Salt Lake City, UT (Jun 2018). https://doi.org/10.1109/CVPR.2018.00013, https://ieeexplore.ieee.org/document/8578111/
15. Mueller, F., Davis, M., Bernard, F., Sotnychenko, O., Verschoor, M., Otaduy, M.A., Casas, D., Theobalt, C.: Real-time pose and shape reconstruction of two interacting hands with a single depth camera. ACM Trans. Graph. **38**(4), 1–13 (Jul 2019). https://doi.org/10.1145/3306346.3322958, http://dl.acm.org/citation.cfm?doid=3306346.3322958
16. Pavllo, D., Porssut, T., Herbelin, B., Boulic, R.: Real-time finger tracking using active motion capture: a neural network approach robust to occlusions. In: Proceedings of the 11th Annual International Conference on Motion, Interaction, and Games. pp. 1–10. MIG '18, Association for Computing Machinery, New York, NY, USA (Nov 2018). https://doi.org/10.1145/3274247.3274501, https://doi.org/10.1145/3274247.3274501
17. Riegler, G., Ulusoy, A.O., Geiger, A.: OctNet: Learning Deep 3D Representations at High Resolutions. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6620–6629. IEEE, Honolulu, HI (Jul 2017). https://doi.org/10.1109/CVPR.2017.701, http://ieeexplore.ieee.org/document/8100184/
18. Schubert, T., Gkogkidis, A., Ball, T., Burgard, W.: Automatic initialization for skeleton tracking in optical motion capture. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 734–739 (May 2015). https://doi.org/10.1109/ICRA.2015.7139260, iSSN: 1050-4729
19. Simon, T., Joo, H., Matthews, I., Sheikh, Y.: Hand Keypoint Detection in Single Images Using Multiview Bootstrapping. pp. 4645–4653 (Jul 2017). https://doi.org/10.1109/CVPR.2017.494
20. Spurr, A., Song, J., Park, S., Hilliges, O.: Cross-Modal Deep Variational Hand Pose Estimation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 89–98. IEEE, Salt Lake City, UT (Jun 2018). https://doi.org/10.1109/CVPR.2018.00017, https://ieeexplore.ieee.org/document/8578115/

21. Verschoor, M., Lobo, D., Otaduy, M.: Soft Hand Simulation for Smooth and Robust Natural Interaction. pp. 183–190 (Mar 2018). https://doi.org/10.1109/VR.2018.8447555
22. Vélaz, Y., Lozano-Rodero, A., Suescun, A., Gutiérrez, T.: Natural and hybrid bimanual interaction for virtual assembly tasks. Virtual Reality **18**(3), 161–171 (Sep 2014). https://doi.org/10.1007/s10055-013-0240-y, http://link.springer.com/10.1007/s10055-013-0240-y
23. Wang, Y., Neff, M.: Data-driven glove calibration for hand motion capture. In: Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation - SCA '13. p. 15. ACM Press, Anaheim, California (2013). https://doi.org/10.1145/2485895.2485901, http://dl.acm.org/citation.cfm?doid=2485895.2485901