https://doi.org/10.2352/ISSN.2470-1173.2021.18.3DIA-119

Generating a Hand Pose Data Set for Vision Based Manual Assembly Assistance Systems

Christian Jauch, Julia Denecke, Marco Huber; Fraunhofer Institute for Manufacturing Engineering and Automation IPA; Stuttgart, Germany

Abstract

Current assistance systems for manual assembly reduce the efficiency of the worker by being invasive in the workflow. To restore the efficiency and at the same time to maintain the benefits of assistance systems, real time hand pose estimation can be used. However, no suitable data set is available for training such application specific detectors. In the presented work, a data set is generated that allows the use of different work gloves and prepares the overlay of realistic hand textures. We use low cost data gloves for hand pose tracking and a RGBD camera to capture the data set with 30 data points per second. This low cost approach is presented in an application for the manual assembly scenario, although transfer of the method to other scenarios is possible.

Motivation

Production processes have been changing for years. Customers demand personalized products, which leads to an increase in product variety and smaller batch sizes of a product with the extreme case of a batch size of one. This increases setup time and lowers the cost efficiency of the production itself. To address this problem, automation is becoming more flexible, but the worker himself is one of the most flexible, if not the most flexible, tool in the production line. Therefore, manual assembly remains an important and time-consuming part of the production process, even though automation is on the rise. In addition to personalized products and the increased variety of products, manual assembly work is increasingly being performed by unskilled workers. Training these unskilled workers is also time-consuming and, above all, not cost-effective due to the additional effort required of the supervisor. Assistance systems for manual assembly can support the worker in overcoming these challenges, increase efficiency and at the same time relieve the worker. The assistance systems can guide the worker through the production process and help them remember small variations between similar products. This also helps the worker to learn new processes while being productive and at the same time takes the burden off the supervisor. This is an important part because hiring often occurs during periods of high demand, when extra work puts even more strain on workers. Assistance systems can also detect errors, increasing product quality and reducing the amount of rework required. Another benefit is automated documentation of the real process where required.

However, current vision-based assembly systems [1] [2] [3] interrupt the worker's workflow very strongly and the worker must interact with the assistance system in addition to his normal work. Interactions with the system do not add value and should therefore be reduced to a minimum. In addition, existing systems often rely on simple algorithms such as detecting contours in bounding boxes for a certain amount of time to detect whether a particular step has been performed. Another problem is the flexibility of the systems, which is why the acquisition of an assistance system often involves a major integration project. Changes in the process or adding new products are costly and time-consuming.

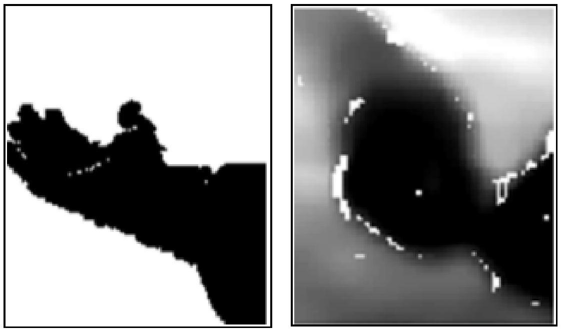


Figure 1. The left image shows a typical depth image from the data set, while the right image shows a typical depth image in the application scenario.

Hand pose estimation is not currently integrated in vision-based manual assembly assistance systems but it can be an enabling technology to solve the existing problems of current assistance systems. By tracking each process step based on hand pose estimation, the interaction between the assistance system and the worker can be kept to a minimum and the worker can remain in the workflow, while the assistance system only requires attention when it is actually assisting the worker in his value-adding task. However, hand pose estimation in manual assembly scenarios face its own challenges.

There are several methods and data sets for hand position estimation. However, the authors are not aware of any application-specific data set for hand position estimation in manual assembly scenarios. In [1], it is shown that current data sets differ from the real scenario and transfer is not practical for several reasons (see Fig. 1).

- Occlusion by objects (e.g., parts of the finished product or tools) and self-occlusion are strongly present. A few data sets already deal with this problem.
- Hands are very close to a surface and do not float in open space, which changes the depth data in particular so much that it is massively different from the data in the training set.
- Gloves are not included in the data sets, but are often worn

[1] ActiveAssist: https://www.boschrexroth.com/de/de/produkte/produktgruppen/montagetechnik/news/assistenzsystem-activeassist/index

[2] Schlaue Klaus: https://www.optimum-gmbh.de/produkte/der-schlaue-klaus

[3] Quality Assist: https://sarissa.de/qualityassist

in production. This could be remedied by using only depth data, which is impractical due to the previous point.

With these differences between the available training data and the real scenario, the authors were in [1] not able to use the trained models for hand pose estimation in this application scenario, so a new application-specific data set is needed. The required data set must be diverse and capture different process steps and work glove types. For completeness, it should include both depth data and color data. In addition, the amount of data needed is very large and the movements in production can be very fast, so the data should be recorded at a minimum of 30 data points per second.

This topic is important for the 3D imaging community because it allows them to quickly generate hand pose data sets for specific applications and to develop new applications based on hand poses themselves.

The paper is organized as follows. First, related work on hand position estimation, including available data sets, is discussed. Then, the hardware setup is described, including a comparison of different methods for acquiring hand position data. The following section discusses the calibration of the gloves to the subject's hand and the transformation between the different coordinate systems. The scope of the data set and the different data formats are then presented. Finally, a conclusion and an outlook are given.

Related Work

Hand pose estimation itself can be divided into different categories. There are generative approaches [2], [3], [4] and discriminative approaches [5], [6], [7], [8]. While generative approaches, or model-driven approaches, rely on knowledge about the structure of the hand, discriminative approaches are purely data-driven and attempt to learn the features using only data. Recently, hybrid approaches [9], [10], [11], [12] have shown very promising results. Here, knowledge about the structure of the hand and its limited degrees of freedom is integrated into the data-driven approach in different ways. This could be achieved, for example, by using Principal Component Analysis and dimensionality reduction [13] in the neural network. Convolutional Pose Machines [14], for example, uses different neural networks to determine the different joint positions and uses so-called Belief Maps to learn spatial models of the relationships between different joints.

One way to categorize the discriminative, data-driven and the hybrid approaches is the input and output data types of the estimators. Data-driven approaches initially relied on depth images as input images and output the 3D coordinates of the joints in camera coordinates, which yields a complete hand skeleton with usually 21 joint positions. DeepPrior++ [13] is a prominent example of this. However, using depth images can have some limitations, e.g., a good quality depth image is expensive or difficult to obtain (as in the application presented in this paper) and therefore limits the range of possible applications. With the use of RGB images as input data, the possible applications expand again. When using RGB, there is 2D hand pose estimation, which provides the position of the wrists in pixel coordinates, and 3D hand pose estimation, which learns the 3D hand pose from a single RGB image. An example of 2D hand pose estimation is SRHandNet [15], while an example of 3D hand pose estimation is presented in [16]. Recently, one can also find hand meshes as output data of a model [17]. Since the output of a mesh does not provide any benefit to the application scenario, the focus in this paper is on 3D hand pose estimation from RGB images.

Most methods rely on large data sets, so data sets with more than a million data points are no exception. Therefore, the acquisition of a data set is also a large effort and can be the first point of failure in hand pose estimation. The existing data sets can be divided into three categories. Depth image-based data sets, RGB image-based data sets, and RGBD image-based data sets that contain both an RGB and a depth image. Most data sets are based on real captured images, some consist of purely synthetically generated images or are a mixture of both. There are also different viewpoints, distinguished only between first- and third-person views. In the manual assembly application, however, a very specific third-person view exists; the top view of the workplace. Views from the front or side are not used in manual assembly assistance systems due to space conflicts with the part feeding system. The hand poses provided consist of varying numbers of joints, with 21 joints being the most common. Since meshes are not of interest for the task in this work, they are not considered, although some data sets provide them as well. Table 1 shows an overview of some of the larger data sets for hand pose estimation.

While the disadvantages of the depth image-based data sets for this application have already been reviewed, the RGB and RGBD image-based data sets have not been fully discussed. However, the first-person perspective data sets have already been covered. The RGB data sets *InterHand2.6M* and *CMU Panoptic HandDB* as well as the RGBD data set *RHD* do not contain object interactions and are therefore also not suitable for the application scenario. This results in two remaining data sets from the Table 1, both of which are fairly up-to-date: *FreiHAND* and *ContactPose*. *ContactPose* contains interactions with objects, but both the hand and the object are in open space and no interactions exist between the hands, which is unusual in manual assembly scenarios. The data set is closer to the application compared to other existing data sets, but not close enough so that an application-specific data set is still beneficial to close the gap between training data and real application. The same statement can be made about *FreiHAND*. In contrast to *ContactPose*, in *FreiHAND* there are more objects in the data set that resemble a manual assembly scenario, but the training data is still very unspecific with respect to manual assembly applications. Therefore, an application specific data set for hand pose estimation in manual assembly scenarios is missing.

Hardware Setup

It was discussed that an application-specific data set is required for hand pose estimation. However, there may be other applications where a separate data set could be beneficial. Therefore, the proposed method should be time efficient, transferable to other hand pose estimation use cases, and use comparatively inexpensive sensors. The method must be able to acquire both the RGB image and the depth image, and the acquisition of different types of work gloves is required for diversity. In addition, training a neural network for hand pose estimation requires a large amount of data and the movements in manual assembly are very fast. Therefore, recording at up to 30 frames per second should be possible.

Table 1: Data sets for hand pose estimation[4].

Data set Name	Type	Year	Real/Synth	No. Joints	Viewpoint	No. Frames	Source
NYU	Depth	2014	Real	36	3rd Person	72k	[18]
BigHands2.2M	Depth	2017	Real	21	3rd Person	2.2M	[19]
ICVL	Depth	2014	Real	16	3rd Person	331k	[20]
InterHand2.6M	RGB	2020	Real	21	3rd Person	2.6M	[21]
FreiHAND	RGB	2019	Real	21	3rd Person	130k	[22]
GANerated Hands	RGB	2018	Synth	21	1st Person	330k	[23]
CMU Panoptic HandDB	RGB	2017	Real+Synth	21	3rd Person	15k	[24]
ContactPose	RGBD	2020	Real	21	3rd Person	2.9M	[25]
Ego3DHands	RGBD	2020	Synth	21	1st Person	50k	[26]
RHD	RGBD	2017	Synth	21	3rd Person	41k	[27]
SynthHands	RGBD	2017	Synth	21	1st Person	64k	[28]

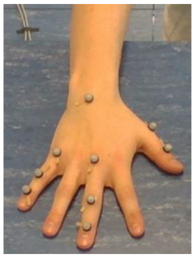


Figure 2. *Preparation of hands for data acquisition with the optical tracking system QualiSys.*

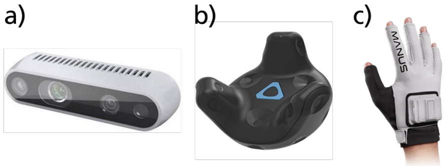


Figure 3. *The hardware used in this work from left to right: a) Realsense D435 Camera, b)VIVE trackers, c) ManusVR Prime II gloves.*

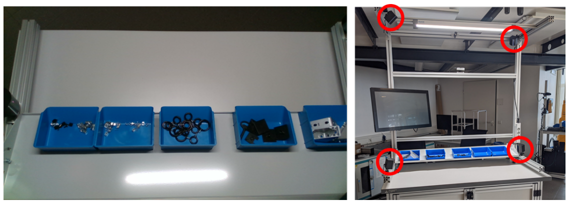


Figure 4. *Left image: Camera view on the workplace. Right image: Workplace with VIVE base stations.*

There are several ways to create a data set for manual hand pose estimation. However, manual labeling is very time consuming and therefore is not considered here. Optical trackers, gloves with magnetic sensors, and gloves with IMU sensors are instead potential means for automated labeling. Optical trackers have several disadvantages. The setup is very time consuming, as an optical marker must be attached to the hand for each joint. In addition, post-processing is demanding. First, the RGB images must be post-processed to remove the markers on the hands, as shown in Figure 2, and second, the data must also be post-processed. Every time the line of sight between a camera and one of the optical markers is broken, which is common in manual assembly scenarios, the system creates a new trajectory with a new ID, which then has to be manually connected to the existing trajectories. Therefore, optical trackers are not an optimal solution for the task at hand. However, magnetic and IMU-based gloves do not have these disadvantages. The magnetic sensors are more precise than IMUs, but are also susceptible to disturbances in the magnetic field that could be introduced by tools during manual assembly, for example. Since the disadvantage in precision is small and IMU-based gloves can still achieve sub-centimeter precision at remarkably lower cost, IMU-based gloves are used in this work. This setup allows us putting on work gloves over the data gloves during the data collection and is independent from occlusion and object interactions.

An Intel Realsense D435 (Fig. 3a) is used for the image data acquisition and placed above the workplace so that it looks down on the workplace from above, as shown in Fig. 4a. The camera captures both the RGB image and the 16-bit depth image with a resolution of 1280x720 pixels at 30 frames per second. For absolute positioning, VIVE trackers (Fig. 3b) are placed above the wrist. They can track their own positions in 3D space and require a line of sight with the VIVE base stations. To ensure permanent line of sight, the trackers are placed above the wrist in an area, which is not prone to occlusion. The hardware setup includes four different base stations (cf. red circles in Fig. 4b). The ManusVR Prime II gloves (Fig. 3c) are used for tracking hand rotations. They are IMU-based gloves that track the relative rotations of 21 joints and their lateral movements (e.g., finger spread). The joint rotations are accessible via an API as quaternions. To ensure diversity in the data set, work gloves and unicolor gloves are worn over the data gloves. The use of unicolor gloves allows for subsequent data manipulation with synthetically generated hand textures.

Calibration and Data Acquisition

The hand model is similar to other data sets and uses 21 hand joints as shown in Figure 5. The wrist joint is the source for all finger structures. Each finger structure, including the thumb, contains four joints. Starting from the wrist, these are the MetaCarpoPhalangeal joint (MCP), Proximal InterPhalangeal joint (PIP),

[4] https://github.com/xinghaochen/awesome-hand-pose-estimation

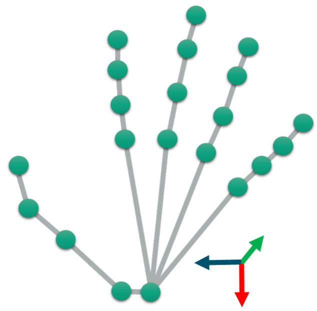


Figure 5. *A model of the 21 joints and the coordinate system, shown for the right hand. The x-axis (red) points towards the person, the y-axis (green) points upwards away from the top of the hand and the z-axis (blue) points to the left, so it is a right-sided system.*

Distal InterPhalangeal joint (DIP) and finally the tip of the finger. The thumb does not follow the medical definition as it has no DIP joint. For simplicity, our model differs from the medical definition. We place the MCP joint of the thumb next to the wrist, the medical MCP joint is the PIP joint, and the medical PIP joint is the DIP joint. Thus, the thumb also has 4 joints from MCP to TIP in our model. The wrist is placed in the extension of the middle finger, while the fake MCP joint is placed in the extension of the index finger.

The coordinate system is also shown in Figure 5. For each joint, the x-axis in red always points towards the person, while the y-axis in green points upwards away from the top of the hand. The z-axis in blue points to the left, resulting in a right-sided coordinate system.

The gloves provide the rotations for each of the joints in the coordinate system of the joint. For a complete hand model in the coordinates of the tracker system, the distances between the joints, the bone lengths, are still missing. This can be provided by generating a person-specific hand model.

The procedure consists of several steps. First, the hands are placed on a flat surface and an image is taken. Then, each joint is marked in the RGB image. This step is repeated n times to minimize errors. The result are n matrices $\mathbf{P}_i$ of size $j \times 2$ containing the pixel coordinates u, v for each joint j with range $0, \dots, 20$ in measurement i of n. The depth image is aligned to the RGB image and together with the intrinsic camera parameters, the coordinates of each joint can be calculated in camera coordinates resulting in n matrices $\mathbf{M}_i$ of size $j \times 3$ based on the pixel coordinates u, v with the function $f(u,v) = (x,y,z)$

$$\begin{aligned} z &= d, \\ x &= \frac{u - pp_x}{f_x} z, \\ y &= \frac{v - pp_y}{f_y} z, \end{aligned} \tag{1}$$

where d is the depth at coordinate (u,v) in the corresponding depth image, pp the principal point of the camera and f is the focal length of the camera.. The bone length d of bone b can now be calculated with the Euclidean distance

$$d_{b,i} = \sqrt{|m^2_{j,i} + m^2_{j+1,i}|}, \tag{2}$$

with j and $j+1$ selected such that they represent adjacent joints of the requested bone b according to the hand model. Now we can remove the outliers with a threshold approach, resulting in $\tilde{n}$ valid measurements and then calculate the length of each bone as the mean value

$$\tilde{d_b} = \frac{1}{\tilde{n}} \sum_{i=0}^{\tilde{n}} d_{bi}. \tag{3}$$

Together with the rotations of the individual joints and the tracker, we can calculate the forward kinematics in the tracker coordinates. However, the hand data is needed in the camera coordinate system thus, a proper tracker-camera calibration is necessary. The calibration from tracker to camera coordinates is done in a similar way than the calculation of the bone lengths and completes the step of describing the individual joints in camera coordinates. The process is separate from the generation of the hand model, since the hand model is person-specific, while the calibration of the camera coordinates is hardware-specific.

An image is acquired with both trackers visible. With the aligned depth image as well as the intrinsic camera parameters, the positions are available in both camera coordinates C and tracker coordinates T. Points are described as vector $\mathbf{p}$ and rotation matrices as matrix $\mathbf{R}$ with indexes describing the corresponding locations like T for the trackers or W for the wrist. The tracker coordinates are provided directly by the trackers. This step is repeated several times and results in two sets of corresponding points $\mathbf{M}_C$ and $\mathbf{M}_T$. The Umeyama implementation [29] in OpenCV computes the homogeneous transformation matrix $\mathbf{T}_{C,T}$, which contains the rotation matrix $\mathbf{R}_{C,T}$ and the translation vector $\mathbf{t}_{C,T}$ from the tracker coordinates to the camera coordinates.

The calculation of any joint is now possible in camera coordinates. The wrist W in camera coordinates $\mathbf{p}_W$ can be calculated as

$$\mathbf{p}_W = \mathbf{T}_{C,T}\mathbf{p}_T + \mathbf{R}_{C,T}\mathbf{R}_T\mathbf{b}_{T,W}. \tag{4}$$

$\mathbf{b}_{T,W}$ represents the corresponding bone from the tracker to the wrist according to the personal hand model. Based on this information the forwards kinematics can be calculated, for example the tip of the index finger $\mathbf{p}_{index,TIP}$ is calculated as

$$\mathbf{p}_{TIP} = \mathbf{p}_{DIP} + \mathbf{R}_{C,T}\mathbf{R}_T\mathbf{R}_W\mathbf{R}_{MCP}\mathbf{R}_{PIP}\mathbf{R}_{DIP}\mathbf{b}_{DIP,TIP}. \tag{5}$$

It is assumed that $\mathbf{p}_{DIP}$ is already calculated in an iterative manner comparably as in Eq. (5).

Data Set

With the hardware setup in place and the coordination transformation in place, the main parts of the data set generation system are complete. The system stores an RGB image and a 16-bit depth image. The RGB image is shown in Figure 6. As part of the data set, for every image there exist multiple text files containing data about the visible hands in the image.

The first text file contains the coordinates of each of the 21 hand joints, where each line represents one hand. The values are whitespace separated and contain the x, y and z value of each joint starting from the wrist and moving on with the thumb in order. The second text file contains the same information but in the pixel

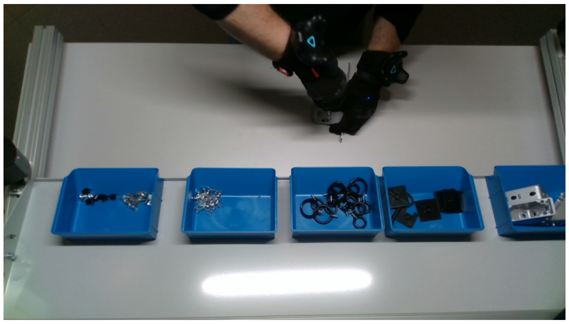


Figure 6. Sample datapoint of the data set. The worker is currently assembling a lock.

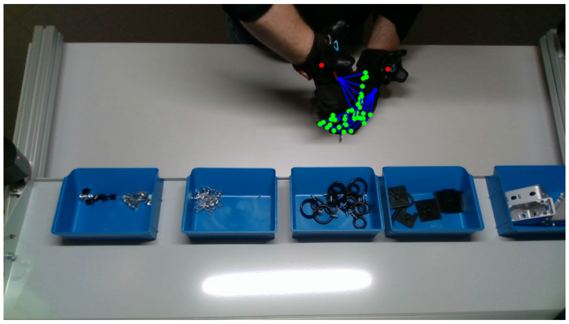


Figure 7. An image from the video of a recording. The hand data is drawn on the RGB image.

coordinates of the image. For this, the intrinsics are used with the inverted transformation from Equation (1).

Finally, by expanding the minimal and maximal pixel values in each dimension of each hand, a bounding box for each hand is defined. This bounding box can be used for training an object detector for hand localization in the image. The most prominent example currently is YOLOv3 [30], therefore the pixel values are translated into the corresponding format.

For each recording, a video is created where the pixel coordinates are drawn over the RGB images. An image of the video is shown in Figure 7.

Conclusion and Future Work

In the presented work, the need for an application-specific data set for hand pose estimation was motivated based on the current state of the art in manual assembly assistance systems. Different hand pose tracking methods were compared and IMU-based gloves were identified as a more suitable and cost-efficient solution compared to optical trackers, magnetic gloves and manual labeling. The hardware setup further comprises a low-cost RGBD sensor and trackers for absolute positioning of the hands. The calibration procedure involves matching the glove to the subject's hand as well as transforming the wrist positions from the trackers' coordinate system to the camera's coordinate system. As a result, the method is able to generate hand pose data in camera as well as pixel coordinates and hand localization data in the pixel coordinates. With the data provided by the method, it is possible to train a hand localizer and hand pose detection algorithm based on either RGB images, depth images or both.

The proposed method is comparatively inexpensive with hardware costs of about 5,000 EUR to record an application-specific data set for hand pose estimation. The method requires very little setup time and there is no post-processing of the hand data. For applications, where gloves are worn, the same is true for post-processing of the image data. If bare hands are required in the data set, the problem of manipulating the RGB images remains an unsolved problem. With the proposed method, the data set can be acquired at up to 30 data points per second which makes the data acquisition very fast. This is especially important for hand pose estimation data sets since a lot of data is required for good results. The presented method allows transfer to new applications apart from manual assembly and achieves high efficiency while maintaining diversity and low investment costs.

For future work, the resulting data set needs to be compared to existing methods and proven itself in real manual assembly scenarios. For this purpose, the authors need to train the application-specific data set with different hand pose estimation methods. Based on the results of the hand pose estimation, assistance systems for manual assembly can be improved as motivated in this work. In order to incorporate the possibility of capturing a bare hands data set with the proposed method, a method for post-processing or manipulating the RGB images to include bare hands is required. This could be achieved very easily by drawing available hand models with different textures over the current RGB image, based on the known hand pose. Another solution could be to use Auto-Encoders or GANs for this purpose, similar to Deep Fakes [31], [32].

References

[1] Martin Root and Christian Jauch, Challenges of designing hand recognition for a manual assembly assistance system, Proc. SPIE 11059, Multimodal Sensing: Technologies and Applications, (2019).

[2] Hui Liang, Junsong Yuan, Daniel Thalmann, and Zhengyou Zhang, Model-based hand pose estimation via spatial-temporal hand parsing and 3d fingertip localization, The Visual Computer Vol. 29, 2013, pp. 837–848.

[3] Iason Oikonomidis, Nikolaos Kyriazis and Antonis A. Argyros, Tracking the articulated motion of two strongly interacting hands, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 2012, pp. 1862-1869.

[4] David J. Tan et al., Fits Like a Glove: Rapid and Reliable Hand Shape Personalization, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 5610-5619.

[5] Liuhao Ge et al., 3D convolutional neural networks for efficient and robust hand pose estimation from single depth images, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 5679-5688.

[6] Chi Xu and Li Cheng, Efficient hand pose estimation from a single depth image, Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 3456– 3462.

[7] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit, "Hands deep in deep learning for hand pose estimation," Proceedings of Computer Vision Winter Workshop (CVWW), 2015, pp. 21-30.

[8] Danhang Tang, Tsz-Ho Yu and Tae-Kyun Kim, Real-Time Articulated Hand Pose Estimation Using Semi-supervised Transductive Regression Forests, Proceedings of the IEEE International Conference on Computer Vision (ICCV '13), 2013, pp. 3224–3231.

[9] Philip Krejov, Andrew Gilbert and Richard Bowden, Combining dis-

criminative and model based approaches for hand pose estimation, 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), Ljubljana, Slovenia 2015, pp. 1-7.

[10] Toby Sharp et al., Accurate, robust, and flexible real-time hand tracking, Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, 2015, pp. 3633–3642.

[11] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," ACM Transactions on Graphics, 2014, pp. 1-10.

[12] D. Tzionas et al., Capturing hands in action using discriminative salient points and physics simulation, International Journal of Computer Vision, 2016, pp. 172-193.

[13] Markus Oberweger and Vincent Lepetit, DeepPrior++: Improving Fast and Accurate 3D Hand Pose Estimation, IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 2017, pp. 585-594.

[14] Shih-En Wei et al., Convolutional pose machines, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 4724-4732.

[15] Yangang Wang, Baowen Zhang and Cong Peng, SRHandNet: Real-Time 2D Hand Pose Estimation With Simultaneous Region Localization, IEEE Transactions on Image Processing, vol. 29, 2020, pp. 2977-2986.

[16] Christian Zimmermann and Thomas Brox, Learning to Estimate 3D Hand Pose from Single RGB Images, IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 4913-4921.

[17] Breannan Smith et al., Constraining dense hand surface tracking with elasticity, ACM Trans. on Graphics 39, 2020.

[18] Jonathan Tompson et al., Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks, ACM Transactions on Graphics, 2014.

[19] Shanxin Yuan et al., Bighand2.2M benchmark: Hand pose dataset and state of the art analysis, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, pp. 2605-2613, 2017.

[20] Danhang Tang et al., Latent regression forest: Structured estimation of 3d articulated hand posture, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, pp. 3786-3793, 2014.

[21] Gyeongsik Moon et al., InterHand2.6M: A Dataset and Baseline for 3D Interacting Hand Pose Estimation from a Single RGB Image, European Conference on Computer Vision (ECCV), 2020.

[22] Christian Zimmermann et al., FreiHAND: A Dataset for Markerless Capture of Hand Pose and Shape from Single RGB Images, IEEE International Conference on Computer Vision (ICCV), pp. 813-822, 2019.

[23] Franziska Mueller et al., GANerated Hands for Real-Time 3D Hand Tracking from Monocular RGB, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, pp. 49-59, 2018.

[24] Tomas Simon et al., Hand Keypoint Detection in Single Images Using Multiview Bootstrapping, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, pp. 4645-4653, 2017.

[25] Samarth Bramhmbhatt et al., ContactPose: A Dataset of Grasps with Object Contact and Hand Pose, arXiv:2007.09545, 2020.

[26] Fanqing Lin, Connor Wilhelm and Tony Martinez, Two-hand Global 3D Pose Estimation Using Monocular RGB, Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2373-2381, 2021.

[27] Christian Zimmermann and Thomas Brox, Learning to Estimate 3D Hand Pose from Single RGB Images, arXiv:1705.01389, 2017.

[28] Franziska Mueller et al., Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1154-1163, 2017.

[29] Shinji Umeyama, Least-squares estimation of transformation parameters between two point patterns, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1991, pp. 376-380.

[30] Joseph Redmon and Ali Farhadi, YOLOv3: An Incremental Improvement, arXiv:1804.02767, 2018.

[31] Aliaksandr Siarohin et al., First Order Motion Model for Image Animation, Conference on Neural Information Processing Systems (NeurIPS), 2019.

[32] Tero Karras, Samuli Laine and Timo Aila, A Style-Based Generator Architecture for Generative Adversarial Networks, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, pp. 4396-4405, 2019.

Author Biography

Christian Jauch has been a research associate at the Fraunhofer Institute for Manufacturing Engineering and Automation IPA in Stuttgart, Germany since 2015 and studied technical cybernetics at the University of Stuttgart. He works in the Image and Signal Processing department and is part of the Scene Analysis group. His work focuses on industrial applications, more specifically on hand pose estimation and hand gesture recognition in manual assembly scenarios.

Julia Denecke studied computer science at the University of Stuttgart. Since 2007 she has been a research associate at the Fraunhofer Institute for Manufacturing Engineering and Automation in the department of Image Processing and Signal Analysis. 2013 she finished her PhD in the topic of volume data processing. Since 2016 she is the group leader of the scene analysis and focusses on 2D and 3D applications for dynamic detection in scene context.

Marco Huber received his diploma, Ph.D., and habilitation degrees in computer science from the Karlsruhe Institute of Technology (KIT), Germany, in 2006, 2009, and 2015, respectively. From June 2009 to May 2011, he was leading the research group "Variable Image Acquisition and Processing" of the Fraunhofer IOSB, Karlsruhe, Germany. Subsequently, he was Senior Researcher with AGT International, Darmstadt, Germany, until March 2015. From April 2015 to September 2018, he was responsible for product development and data science services of the Katana division at USU Software AG, Karlsruhe, Germany. At the same time he was adjunct professor of computer science with the KIT. Since October 2018 he is full professor with the University of Stuttgart. At the same time, he is director of the Center for Cyber Cognitive Intelligence (CCI) and of the Department for Image and Signal Processing with Fraunhofer IPA in Stuttgart, Germany. His research interests include machine learning, planning and decision making, image processing, data analytics, and robotics. He has authored or co-authored more than 100 publications in various high-ranking journals, books, and conferences, and holds two U.S. patents and one EU patent."

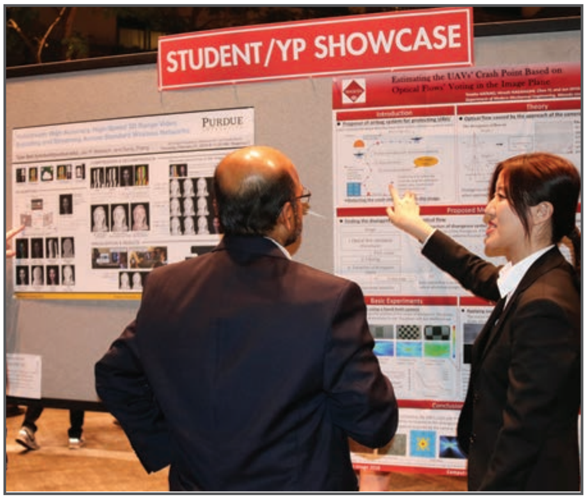
JOIN US AT THE NEXT EI!

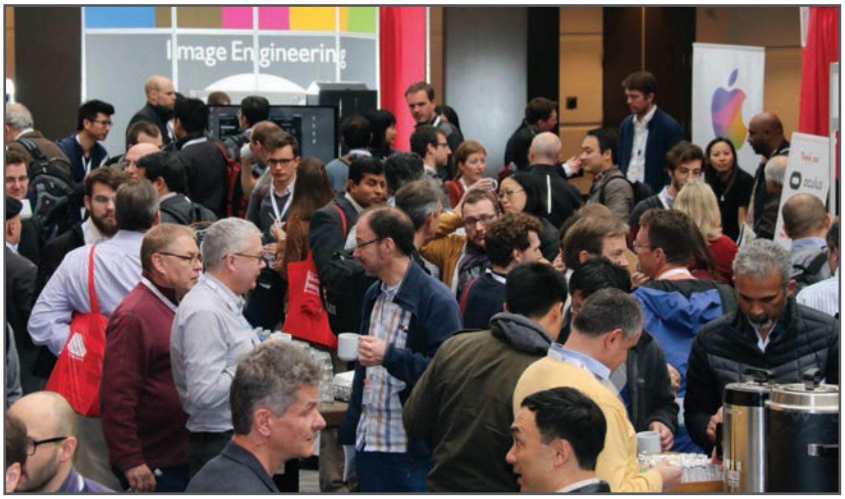
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





AR is a fundamental and transformative interface and platform.
20+ years from my initial working demo to today's AR displays
Optimistic that it won't take another 20 to reach ubiquitous consumer AR!

• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org