# Regression-based 3D Hand Pose Estimation using Heatmaps

Chaitanya Bandi and Ulrike Thomas
*Robotics and Human-Machine-Interaction Lab, Chemnitz University of Technology, Reichenhainer str. 70, Chemnitz, Germany*

Keywords: Convolutional Neural Networks, Pose, Heatmaps, Regression.

Abstract: 3D hand pose estimation is a challenging problem in human-machine interaction applications. We introduce a simple and effective approach for 3D hand pose estimation in grasping scenarios taking advantage of a low-cost RGB-D camera. 3D hand pose estimation plays a major role in an environment where objects are handed over between the human and robot hand to avoid collisions and to collaborate in shared workspaces. We consider Convolutional Neural Networks (CNNs) to determine a solution to our challenge. The idea of cascaded CNNs is very appropriate for real-time applications. In the paper, we introduce an architecture for direct 3D normalized coordinates regression and a small-scale dataset for human-machine interaction applications. In a cascaded network, the first network minimizes the search space, then the second network is trained within the confined region to detect more accurate 2D heatmaps of joint's locations. Finally, 3D normalized joints are regressed directly on RGB images and depth maps can lift normalized coordinates to camera coordinates.

## 1 INTRODUCTION

In a Human-Machine Interaction (HMI) environment, 3D pose of the hand is significant. When handing over objects between machines like robots and humans, it is important to recognize and track the human hand in order to avoid collisions and collaboration. For grasping objects, in addition to tracking the hand it is also important to track the objects. The aim is to find free space on the surface of the object to grasp it. Thus, the most important information is the position of fingers. The robot should not grasp where humans hold the object as shown in Figure 1. 3D hand pose tracking has many such applications, including sign language, virtual reality, and gesture recognition but the building blocks of this paper have been motivated for robotics applications.

In the past few years, pose estimation has gained significant attention and has been improved, but in a human-robot collaboration scenario, high dexterity and self-occlusion of the human hand increase the complexity of 3D hand pose estimation. It still is a challenging issue. Due to high self-occlusion and dexterity, sensing equipment is preferred like markers, data-gloves or motion capture sensors to communicate with robots and to acquire the dataset. Most of the state-of-the-art techniques rely completely on depth information to estimate the 3D hand pose. The sensors like Kinect and Intel Realsense provide both RGB and depth information. With the information from low-cost RGB-D cameras, Convolutional Neural Networks (CNNs) have become the norm for pose estimation techniques.

In this paper, we introduce a complete pipeline for tracking the upper body pose and 3D hand pose in human-robot collaborated workspace. Due to integration flexibility, the Intel Realsense D435 camera will be considered for hand over applications in HMI environments as shown in Figure 1. Our goal is to estimate 3D human upper body poses and 3D hand poses in RGB images, given the respective aligned depth maps. Our approach consists of cascaded CNNs, post-processing to extend 2D heatmaps to 3D pose, and direct regression of the 3D pose from the network. The first network localizes the upper body and hand in the scene.

Figure 1: Human-Robot interaction.



Bandi, C. and Thomas, U.
Regression-based 3D Hand Pose Estimation using Heatmaps.
DOI: 10.5220/0008973206360643
In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020) - Volume 5: VISAPP*, pages 636-643
ISBN: 978-989-758-402-2; ISSN: 2184-4321

Localization drastically reduces the search space and further passed through the second cascaded network that estimates the heatmaps of keypoints in 2D images and then regresses the 3D normalized pose. Finally, the 3D normalized coordinates are transformed into camera coordinates or world coordinates. In addition to the unique pose network, we introduce a new small-scale multiview hand pose dataset (SSMH) for HMI applications.

## 2 RELATED WORKS

3D hand pose estimation is a very challenging task due to the high dexterity of the human hand (i.e., 21 degrees of freedom). We briefly review the state-of-the-art of 2D and 3D pose estimation methods that were successful over the past few years. The successful breakthrough in pose estimation was related to human pose estimation as in (Tompson et al., 2014) and (Toshev and Szegedy., 2014). The idea of belief maps was mentioned by (Wei et al., 2016) for 2D human pose estimation applying convolutional pose machines. Many papers were published based on this idea of heatmaps to improve the accuracy and to provide further extensions to 3D (Garcia-Hernando et al., 2018). (Tompson et al., 2014) presented the idea of hand pose recovery implementing CNNs in real-time inferring intermediate heatmap features to extract accurate 3D pose with the help of an inverse kinematic model on depth datasets. (Wan et al., 2017) proposed a deep 3D hand pose estimation approach that uses depth images. In this approach, depth feature maps from CNNs were divided into regions to form a tree-structured region ensemble network, these regions are passed through fully connected layers for 3D coordinate regression and for better accuracy.

Later, 3D hand pose estimation in single RGB images was proposed by (Zimmermann and Brox, 2017). A three-network structured approach consisting of HandSegNet for hand segmentation, PoseNet for 2D keypoints and lifting to 3D pose. (Zimmermann and Brox, 2017) created synthetic 2D and 3D hand pose datasets where the inference model cannot generalize well on real test data. Although the proposed network is efficient, it cannot be applicable in real-time. An extension to this is introduced in (Mueller et al., 2018). (Mueller et al., 2018) is one of the best state-of-the-art RGB only based 3D hand pose estimation architectures. One of the best state-of-the-art techniques for heatmap generation is mentioned in (Newell, Yang, and Deng, 2016). The authors use a stacked hourglass network to retain 2D heatmap information. (Wan et al., 2017) also worked on a similar idea of heatmaps on depth images for 2D and 3D, in addition to 3D directional vectors for 3D regression of hand pose. The further extensions to stacked hourglass network can be observed in (Zhou et al., 2017). The 2D CNNs are trained to infer heatmaps and the authors extend the network to output the 3D pose by adding a regression network. The unique pipeline presented in this paper is developed based on (Zhou et al., 2017).

To train the pose estimation architectures, the datasets are the key. There are many hand datasets available for research purposes, and most of them are captured for certain application. The most widely used open-source datasets are the NYU hand dataset (Tompson et al., 2014), ICVL (Tang et al., 2014), and the First-Person Hand Action Benchmark (Garcia-Hernando et al., 2018). The datasets consist of depth images with respective keypoints in 2D and 3D as the labeled dataset. The first-person action database is a large-scale dataset with depth maps, and it is annotated using motion capture sensors and kinematics. The available RGB datasets are the GANerated dataset (Mueller et al., 2018) and Large-Scale Multiview hand pose dataset (LSMH) (Gomez-Donoso, Orts-Escolano, and Cazorla, 2017). GANerated dataset is a hybrid dataset that builds a bridge between real and synthetic data for better generalization of trained network, but the data is egocentric. GANerated images are shown in Figure 2a and 2b. The LSMH pose dataset is captured with a leap motion sensor calibrated with 4 RGB cameras with distinct views, the dataset has many outliers. Few sample images can be seen in Figures 2c and 2d.

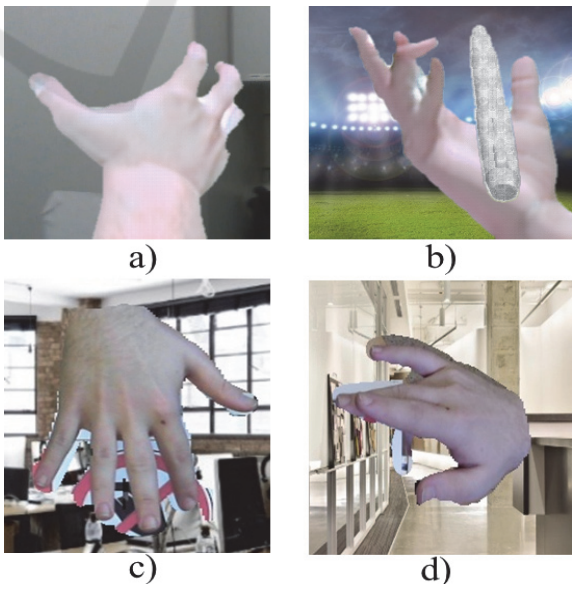


Figure 2: a) GANerated Hands with self-occlusion, b) with object occlusion, c) and d) LSMH dataset.

# 3 PROPOSED METHOD

The goal of the new efficient approach is to infer 3D joints of the human hand and the pose of the upper body. We tackle this problem using two-staged CNNs. Given a color image $I \in R^{NxMx3}$, the first network localizes the persons and their respective hands in the image. The localization reduces the search space for the 2D pose estimation network. Object localization has gained many research advancements since the introduction of CNNs. In this paper, we experiment with an architecture to estimate the 3D pose of the hand on the frame level. Cascaded CNNs are extremely helpful in multi-feature tracking applications (i.e., human, face and hand pose tracking). To work in a collaborated environment, human tracking and hand pose tracking is the key task to avoid collisions and to work as a team. In this approach, two-staged cascaded networks seem feasible for tracking hands and the upper body. The localized result from the first network can be connected to the distinct branches as described in Figure 3 to obtain 3D hand poses and/or body poses.

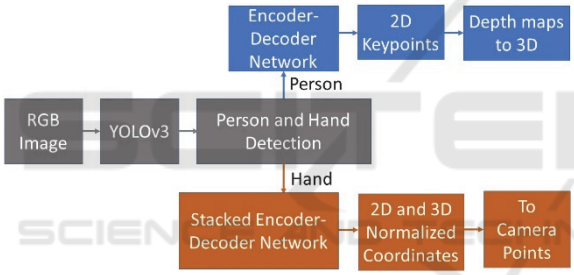


Figure 3: Proposed architecture, the gray blocks represent localization network, the blue blocks represent upper body pose estimation network, and the orange blocks represent the hand pose network.

## 3.1 Localization Network

In the localization network, object detection technique is used to obtain the region of interests (i.e., body and hands) as a bounding box and its score threshold. Since object detection is a highly researched topic, a significant number of architectures are built for object detection (e.g., ResNet (He et al., 2015), MobileNet (Howard et al., 2017), Darknet (Redmon et al. 2018). You only look once (YOLO) is one of the state-of-the-art real-time object detection system that uses the Darknet architecture. The more recent YOLOv3 (Redmon and Farhadi, 2018) architecture is fast and accurate compared to its predecessors and other available architectures. In this work, frame-based person and hand detection are trained by applying transfer learning on YOLOv3 darknet-53 architecture using the MPII Human Pose dataset (Andriluka et al., 2014). Leveraging the 2D keypoints extrema, bounding boxes were extracted from the MPII dataset. Since the dataset does not provide bounding box locations of the hands, hands were trained with a custom dataset and ego-hands dataset (Bambach et al., 2015). The network is retrained with these datasets.

## 3.2 Pose Network

Once the persons and hands are localized, the image containing hand is passed through the second network to detect heatmaps of 2D keypoints of respective hands in given RGB images. To estimate 3D coordinates, one could use the depth map to extend the 2D keypoints to 3D using calibration parameters. Another possibility is to train an encoder-decoder architecture to detect heatmaps and to directly regress 3D joints on RGB images. For training the network, the joint order must be preserved for visualization and further processing. Body joints and hand joints order used for training can be observed in Figure 4.

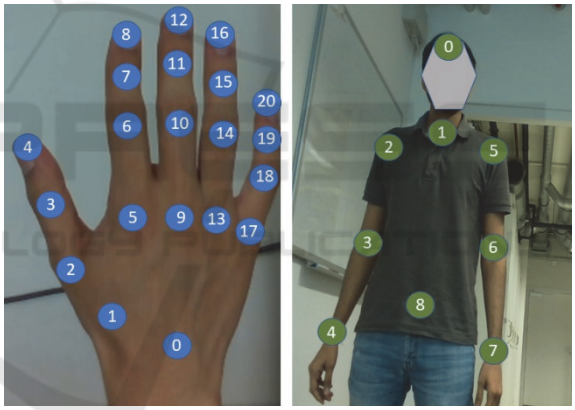

Figure 4: Hand and body joint annotation order.

### 3.2.1 SSMH Pose Dataset

In this work, the GANerated dataset and the LSMH pose dataset are considered for the experimentation with an RGB pipeline. Both datasets provide 2D and 3D joint locations of fingers and they have certain limitations. In addition to the available public datasets, we introduce a SSMH pose dataset. The dataset was captured using Intel Realsense D435 cameras in distinct settings. In the first setting, hand images and the respective aligned depth images are captured without self-occlusions. In the second set, two cameras are placed orthogonal to each other and calibrated together. The simultaneous frames with respective RGB and aligned depth information are captured.

Figure 5: top: Single camera setting without self-occlusions, bottom: Two camera calibrated orthogonal to each other with self-occlusions.

The transformations are used to convert keypoints from one camera to another. The keypoints are labeled manually. In two camera settings, the visible points in each image are labeled. Then the visible 2D keypoints are extended to 3D using the depth map and the intrinsic parameters. The images from dataset can be observed in Figure 5. In Figure 5, bottom, images are captured using two views and based on keypoint visibility all 21 joints are manually annotated. The advantage of this dataset is that both RGB images and depth images are available for experimentation unlike available dataset.

### 3.2.2 Stacked Encoder-decoder Architecture

One of the successful approaches to regress 3D joints are implemented on human pose estimation as described in (Zhou et al., 2017). In this work, we consider encoder-decoder architecture as the ratios of input and output images must be kept constant. The outputs of such architecture are 2D heatmaps of the joints. To create encoder-decoder architecture ResNet34 blocks are used. Instead of single encoder-decoder architecture, a stacked encoder-decoder network is designed for better generalization of data. Heatmaps of 21 joints are considered for training. To this network, we add a regressor block to train the 3D coordinates of normalized joints, the simple overview of the architecture is mentioned in Figure 6. In this architecture, images with three channels (i.e., RGB) are passed as input features. The input images of size 3x128x128 pixels and with convolution, features are extracted from 64x64x64 to as low as 512x4x4. Then the features are again upsampled using bilinear interpolation from 512x4x4 to 64x64x64.

The process is repeated for one more stage and the output heatmaps are processed and concatenated with intermediate features and passed through convolutional block. Finally, features are linearized to get 3D normalized joint coordinates. The complete feature representation architecture can be understood from Figure 7.

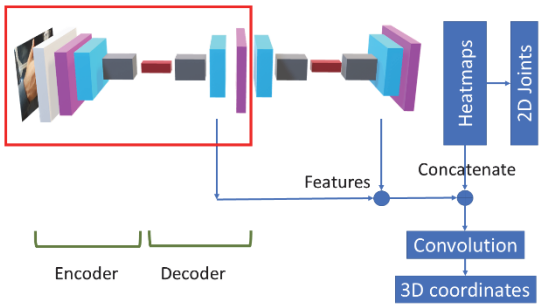


Figure 6: Stacked encoder-decoder architecture.

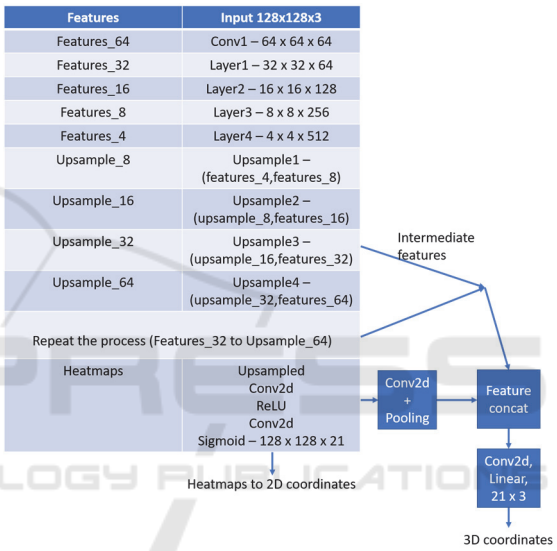


Figure 7: Feature representation of encoder-decoder architecture.

## 3.3 Pose Estimation

### 3.3.1 Upper Body Pose Estimation

From a localization network, a person and their respective hands are detected. The results of the localized network can be seen in Figure 8 left. The localized person is passed through single encoder-decoder architecture. As the 3D joints information of the MPII pose dataset is not available, 2D pose information is trained with the single model. The applied single encoder-decoder can be seen in Figure 6 (i.e., in red block). The images are scaled to similar size for batch normalization during training process. 20k images and their respective heatmaps are used for training.

Figure 8 right represents the detection of 2D keypoints of a person. Once 2D keypoints are

estimated, joints are extended to 3D using depth maps and camera calibration parameters.

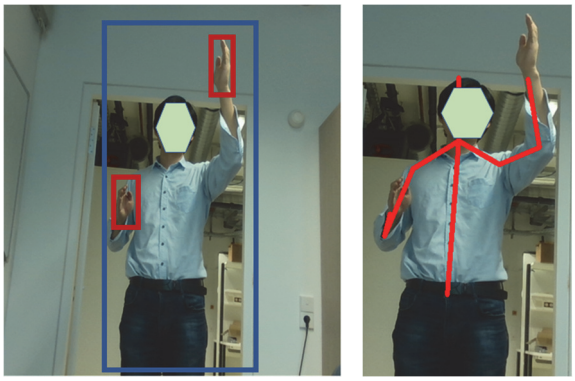

Figure 8: left: Bounding box, right: 2D Pose.

### 3.3.2 Hand Pose Estimation

The architecture mentioned in section 3.2.2 can be trained for 3D hand pose estimation. Since the LSMH pose dataset, the SSMH dataset introduced in this paper and the GANerated hands dataset contains 3D and 2D joint locations, we experiment with stacked encoder-decoder architecture for 3D pose regression. To train the network, images must be pre-processed (i.e., resizing the image and generating the heatmaps before passing through the training loop). Closely cropped hands from the localized network are first resized to 128x128 pixels and respective gaussian heatmaps of joints are generated with the size 21x128x128. The 3D joints are normalized with respect to the position of middle finger joint or joint number 9 in Figure 4 left for LSMH dataset and SSMH dataset. For GANerated dataset, 3D joints provided are normalized with respect to joint 9 and normalized distance from joint 0 to joint 9 is 1. This normalization assumes that the most constant values are joint 0 and joint 9.

Once the 2D heatmaps are estimated, the depth map of the respective RGB image with calibration parameters can be used to extend 2D joint positions to 3D coordinates. This technique can be implemented in an application where the average coverage area of fingers is adequate instead of accurate fingertips or in an application where self-occlusion is neglected. The spatial error due to conversion from 2D to 3D can be fixed with the help of the kinematic chain model.

### 3.3.3 Pose Estimation on LSMH Data, GANerated Data and SSMH Data

The LSMH pose dataset contains real images, the regressor architecture is trained with the dataset. Figure 9a contains images from the validation dataset. The 2D keypoint detections can be observed in Figure 9a. It might be seen that the detections have a slight error in 2D, but it is, in fact, an error in the dataset. The ground truth keypoints from leap motion sensor were not post-processed.

Later, the architecture is trained with GANerated hands dataset. This dataset contains over 300k synthetic images with and without objects. The GANerated dataset is highly egocentric and the hands have high self-occlusion. The output of the network with 2D keypoints can be seen in Figure 9b. We can observe that the detected 2D keypoints are almost error free even for high self-occlusion.

Finally, encoder-decoder architecture is trained with SSMH dataset introduced in this paper. The dataset consists of 5000 images with and without self-occlusions for training. Once the joints are normalized, coordinates are in lower dimension and they can converge faster to minima during the training process. 2D keypoint detection can be seen in Figure 9c. 3D hand pose estimation of all the dataset can be observed in Figure 10.

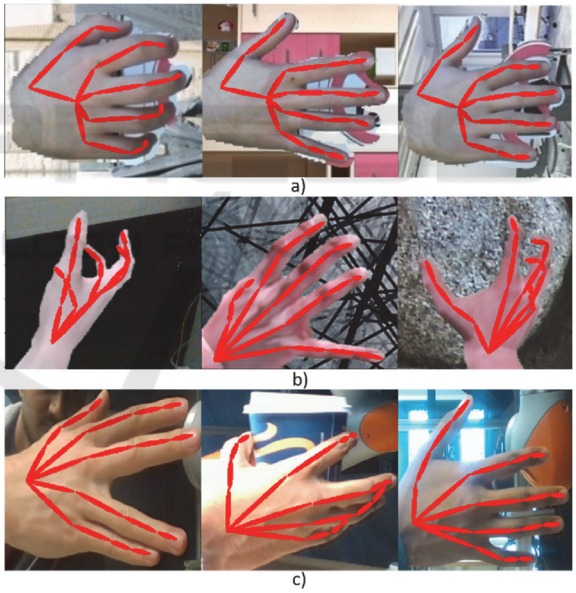


Figure 9: 2D detections, a) detection on LSMH dataset, b) detections on GANerated dataset, and c) detections on SSMH dataset.

## 4 EXPERIMENTS

### 4.1 Training Parameters

The proposed pipeline is implemented in Python using the Pytorch library. We opted to Pytorch as it is faster compared to Tensorflow and supports easy integration

with the Numpy library. For the localization network, approximately 22k images were considered for training. For estimating the upper body pose, the complete MPII dataset was trained for 1000 epochs with a batch size of 32 and 8 parallel workers on Nvidia 1080 Ti 12 GB memory.

From LSMH pose dataset, only 35k images with minimal error were considered for training with stacked encoder-decoder architecture. The weights for training the network were randomized and the complete architecture was trained from scratch. Since the GANerated hands dataset is synthetic, images contain a wide range of occlusions. Over 300k images were considered for training. Weights were randomized for SSMH dataset for training, only 5k labeled images were considered for training. The datasets were trained with multiple optimizers to test the accuracy. The optimizer used for this network is the RMSprop. Variable learning rates were implemented based on the number of epochs, varying from 0.005 to 0.00025. All datasets were trained for over 1000 epochs with 16 to 32 images per batch and approximately 8 to 12 parallel workers. Since the output is image coordinates or 3D coordinates, the Mean Squared Error (MSE) loss is the best fit for regression applications.

$$L_{MSE} = \frac{1}{n}\sum_{1}^{n}(\text{predicted} - \text{actual})^2 \quad (1)$$

MSE is calculated for both 2D keypoints and 3D keypoints. All the models were trained using Nvidia 1080 Ti 12 GB GPU and tested on Nvidia 1050 Ti 4 GB GPU memory.

## 4.2 Evaluation

We evaluated our approach with multiple experiments to achieve the best possible solution. As mentioned in section 3.2, we considered the bounding box approach to localize hands and persons in a single RGB image. The encoder-decoder architecture was developed based on the idea of hourglass but with ResNet architecture. The prior mentioned datasets are trained with stacked encoder-decoder architecture. The results can be observed in Figure 10.

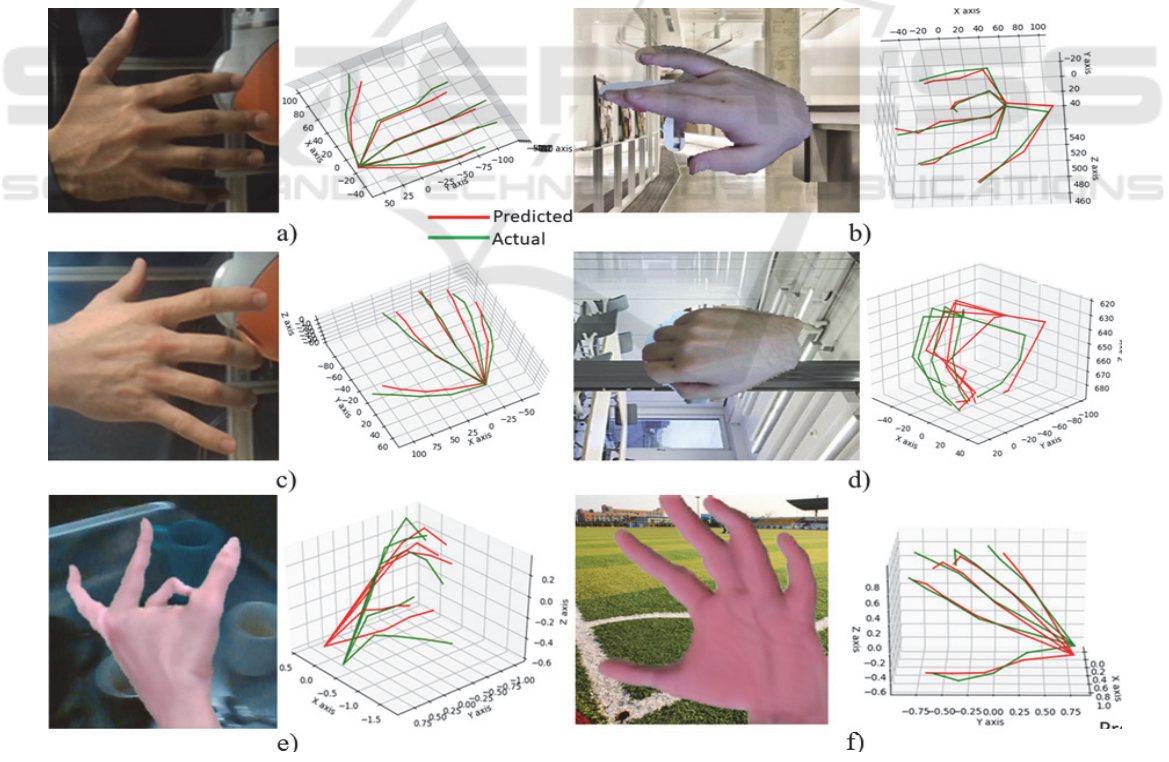


Figure 10: 3D hand pose, a) and c) are images from SSMH dataset, b) and d) are images from LSMH dataset, e) and f) are from GANerated dataset, a) Open hand without occlusions and respective 3D hand pose, c) 3D hand pose on closed hand with low self-occlusion. b) Self-occluded fingers and respective 3D hand pose of LSMH dataset, d) and f) high self-occluded detections, predicted values are plotted in red color and actual values are plotted in green color.

The performance of the LSMH dataset can be observed in Table 1. With encoder-decoder architecture, the performance of images is very acceptable. 3D detections on all datasets can be observed in Figure 10. Figure 10a does not contain any self-occlusion and the MSE was as low as 20 mm. In Figure 10b, there is a slight self-occlusion between three fingers and the MSE was around 27 mm. In Figure 10d, the fingers are completely closed, and the network failed to generalize in such situations. Figure 10e and f represents the GANerated hands 3D hand pose output. (Gomez-Donoso et al., 2018) utilized LSMH dataset for 2D applications and the improvement in MSE error can be seen in Table 1.

Table 1: Performance of the validation set.

| Large-scale Multiview hands | Mean 3D Pose error (mm) over 5000 images | Mean 2D Pose error (px) over 5000 images |
|---|---|---|
| (Gomez-Donoso et al., 2018) | - | 10 |
| Proposed method | 20-65 | 8.58 |

The performance of the GANerated Hands dataset can be observed in Figure 11. MSE of all joints is represented in the bar graph. We can clearly observe that the error is high with respect to visibility. Joint numbers 4, 8, 12, 16, 20 are the fingertip locations and there exists high error compared to other joints due to self-occlusion and/or object occlusion. Since the dataset consists of normalized 3D joint coordinates, comparison with (Mueller et al., 2018) was not possible. (Mueller et al., 2018) preferred a different metric to evaluate the performance of their architecture on GANerated dataset.

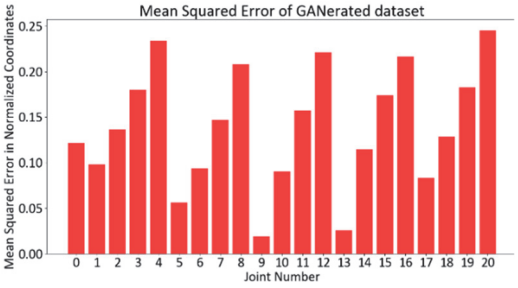


Figure 11: 3D coordinates MSE of GANerated dataset.

Similarly, SSMH dataset is evaluated using MSE. The dataset has keypoints with respect to camera coordinates and is normalized with respect to joint 10. The MSE error was estimated in millimeters. Figure 12 represents the error of 21 joints individually. From the Figure 12, we can observe that the pose estimation of fingertips has an error as high as 30 mm. Overall MSE achieved is as low as 19 mm. The MSE was estimated strictly on 500 images. Images with high self-occlusion achieved MSE over 60 mm. We work on adding more images to SSMH dataset. Once the dataset is refined, it will be released as open source for researchers and further information found in [1].

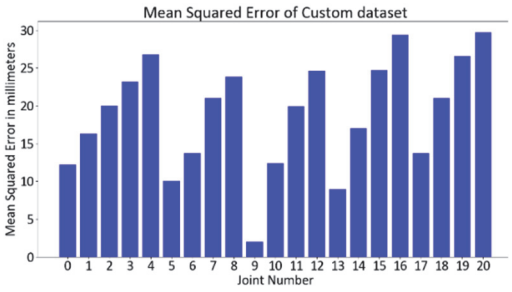


Figure 12: 3D coordinates MSE of our dataset.

There exist algorithms for 3D keypoint regression but most of the algorithms work with RGB image localization and regression on direct depth maps or pointcloud. The state-of-the-art depth-based methods like (Chen et al., 2018), and (Moon et al., 2018) achieved MSE less than 7 mm as in Figure 13. Figure 13 represents the mean error of hand joints between RGB based and RGB-D based methods. (Mueller et al., 2018) achieved a mean error as low as 50 mm with high occlusion datasets.

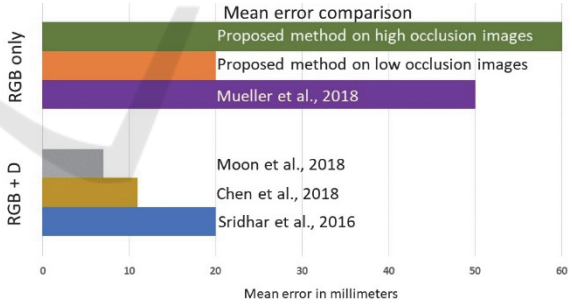


Figure 13: Mean error of hand joints in comparison to state-of-the-art RGB and RGB-D architectures.

We estimated the mean error for both low occlusion and high occlusion images and mean error is as low as 20 mm and as high as 60 mm respectively on SSMH dataset. We can observe that, the depth-based methods have low mean error compared to RGB only architectures. Nevertheless, RGB based methods have high potential to be improved further. In HMI applications highly occluded data is necessary and if

[1] https://orcid.org/0000-0001-7339-8425

the data is specifically captured in that workspace, then the performance of the network can be improved further.

## 5 CONCLUSIONS

We proposed a cascaded CNN pipeline for the upper body pose and the 3D hand pose estimation. Heatmaps and regression techniques are the norms for pose estimation in direct RGB images. We experimented with the stacked encoder-decoder architecture for heatmap based 2D detections and 3D direct regression. Two large-scale RGB datasets and a new SSMH custom dataset were considered for training and testing the performance of the proposed network. We observed that the network performs well under occlusions for all the datasets. We achieved the mean error as low as 20 mm for images containing minimal or no occlusions and mean error is over 60 mm for highly occluded images from SSMH dataset. To apply the proposed pipeline in real-time Human-Machine-Interaction applications, occlusion dataset must be extended and retrained. Further improvements like kinematic fitting and tracking could help in fingertip refinement.

## ACKNOWLEDGEMENTS

This research is supported by Saechsische AufbauBank (SAB – application no. 100378180).


## REFERENCES

Tompson, J., Stein, M., Lecun, Y., Perlin, K., 2014. Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks. ACM Transactions on Graphics, 33(5):1– 10.

Wei, S., Ramakrishna, V., Kanade, T., Sheikh, Y., 2016. Convolutional pose machines. In Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pages 4724–4732.

Toshev, T., Szegedy, C., 2014. Human pose estimation via deep neural networks. In Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pages 1653–1660.

Wan, C., Thomas, P., Van Gool, L., Yao, A., 2017. Dense 3D Regression for Hand Pose Estimation. arXiv:1711.08996v1 [cs.CV].

Garcia-Hernando, G., Yuan S., Baek, S., Kim T.K., 2018. First Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations. arXiv:1704.02463v2 [cs.CV].

Zimmermann, C., Brox, T., 2017. Learning to Estimate 3D Hand Pose from Single RGB Images. arXiv:1705.01389v3 [cs.CV].

Mueller, F., Bernard, F., Sotnychenko, O., Mehta, D., Sridhar, S., Casas, D., and Theobalt, C., 2018. GANerated Hands for Real-Time 3D Hand Tracking from Monocular RGB. CVPR 2018.

Gomez-Donoso F., Orts-Escolano, S., Cazorla, M., 2017. Large Scale Multiview 3D Hand Pose Dataset. arXiv:1707.03742v3.

Bambach, Sven and Lee, Stefan and Crandall, David, J., and Yu, Chen, 2015. Lending A Hand: Detecting Hands and Recognizing Activities in Complex Egocentric Interactions, The IEEE International Conference on Computer Vision (ICCV).

Newell, A., Yang, K., Deng, J., 2016. Stacked Hourglass Networks for Human Pose Estimation arXiv:1603.06937v2 [cs.CV].

Zhou, X., Huang, Q., Sun, X., Xue, X., Wei, Y., 2017. Towards 3D Human Pose Estimation in the Wild: a Weakly-supervised Approach, Shanghai Key Laboratory of Intelligent Information Processing School of Computer Science, Fudan University, The University of Texas at Austin, Microsoft Research arXiv:1704.02447v2 [cs.CV].

Tang, D., Chang, H.J., Tejani, A., Kim, T.K., 2014. Latent Regression Forest: Structural Estimation of 3D Articulated Hand Posture, Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, Ohio, USA.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep Residual Learning for Image Recognition, arXiv:1512.03385v1 [cs.CV]. Microsoft Research.

Howard, G.A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, Google Inc, arXiv:1704.04861v1 [cs.CV].

Redmon, J., Farhadi, A., 2018. YOLOv3: An Incremental Improvement, University of Washington, arXiv:1804.02767 [cs.CV].

Andriluka, M., Pishchulin, L., Gehler, P., Schiele, Bernt, 2014. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis, IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Chen, Xinghao, Wang, Guijin, Guo, Hengkai, Zhang, Cairong, 2018. Pose Guided Structured Region Ensemble Network for Cascaded Hand Pose Estimation. Neurocomputing Journal.

Moon, G., Chang, J.Y., Lee, K.M., 2018. V2V-Posenet: Voxel-To-Voxel Prediction Network for Accurate 3d Hand and Human Pose Estimation from a Single Depth Map, CVPR, arXiv:1711.07399[cs.CV].

Sridhar, S., Mueller, F., Zollhoefer, M., Casas, D., 2016. Real-time Joint Tracking of Hand Manipulating an Object from RGB-D Input. ECCV.