3D Upper-Body Pose Estimation and Classification for Detecting Unhealthy Sitting Postures at the Workplace

Christopher Pramerdorfer
cogvis & TU Wien
Wiedner Hauptstraße 17/3a
1040 Vienna, Austria
Email: pramerdorfer@cogvis.at

Martin Kampel
TU Wien
Favoritenstraße 9-11
1040 Vienna, Austria
Email: kampel@cvl.tuwien.ac.at

Johannes Heering
Fitbase
Brauhausstieg 21
22041 Hamburg, Germany
Email: heering@fitbase.de

Abstract—**Prolonged sitting in an unhealthy posture is a common cause of back pain and other health problems in office workers. People are often not aware that they are sitting in an unhealthy way, the problems this can cause in the long term, and how they should improve their posture. We present a system that is able to provide this information by analyzing people's postures over several days. The system is fully automatic and requires no worn devices. Instead, data from a depth sensor is used for periodic 3D upper-body pose estimation. This pose estimation is carried out by a convolutional neural network that was trained on synthetic depth data to overcome the lack of available real-world datasets. On this basis, each pose is assigned to one of several common classes of healthy and unhealthy sitting poses. This results in a large collection of body poses and classification results, which are used to generate a personalized posture report that includes suggestions for improving the sitting posture. We show experimentally that the system is able to estimate 3D poses and perform pose classification with high accuracy.**

***Keywords*–Workplace health promotion; Sitting posture estimation; Deep learning; Depth data analysis.**

I. Introduction

Approximately 75% of all employees in industrial countries have jobs that require working in a seated position [1]. In the DACH-region (Germany, Austria, and Switzerland) alone, this is the case for around 15 million people. Prolonged sitting is a common cause of pain and health problems in office workers [2][3]. People are often not aware that they are sitting in an unhealthy way, which problems this can cause in the long term, and how to improve their posture [4].

In this paper, we present *ergoscan*, a system we developed to address this issue. Ergoscan periodically measures the head and upper-body pose of people sitting in front of their computer at the workplace. The system requires no worn sensors or any form of user participation, which users might consider intrusive, and does not utilize image or video data to protect the privacy of monitored persons. Instead, ergoscan processes depth data, enabling 3D pose estimation and classification with high accuracy. This information is collected periodically over several days and sent to a server in an encrypted and anonymized form. All processing is carried out locally such that no other data have to leave the system.

An ergoscan system consists of a 3D sensor and an ARM-based single-board computer, which are integrated in a single casing that is mounted at the top of the user's computer monitor as illustrated in Figure 1. The system does not require accurate placement or alignment to facilitate installation; it obtains this information automatically via calibration based on visible planar surfaces such as walls. Installation takes less than a minute and requires no tools or adhesives.

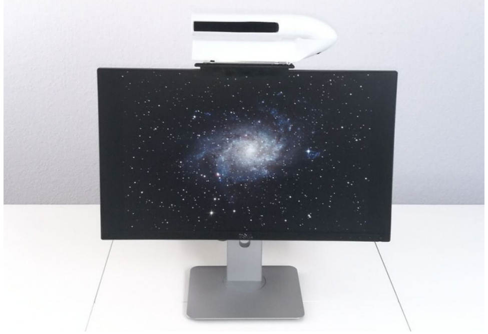

Figure 1. A ergoscan device mounted on top of a monitor.

Once installed, ergoscan periodically performs face detection using an efficient cascade detector [5] to determine whether a person is sitting in front of the screen. If a person is detected, a Convolutional Neural Network (CNN) with a novel architecture estimates their head and upper-body pose. CNNs achieve state-of-the-art performance in related tasks, such as pose estimation in color images, but require large datasets for training, which are not available in our specific problem domain [6][7]. As obtaining a suitable dataset would be a significant effort in this domain, we utilize synthetic depth data for training and show that this is an effective alternative.

The six keypoints located during pose estimation are nasion (intersection of the frontal bone and the two nasal bones of the human skull), chin center, front of the throat, manubrium, as well as the left and right shoulders. These keypoints were selected based on feedback by physiotherapists but the method is generic and can be adapted to any number of keypoints. Angles derived from the 3D coordinates of these keypoints are input to a random forest classifier [8], which assigns one of 15 classes of common healthy and unhealthy sitting postures that were defined together with experts. Two of these poses are visualized in Figure 2.

Systems remain at a particular workplace for up to one week. During this time, thousands of pose measurements and classifications are collected. Physiotherapists analyze this information in an aggregated form to identify unhealthy sitting poses that are commonly assumed. On this basis, the monitored person receives a personal posture report with descriptions and visualizations of these poses, as well as suggestions and links to video tutorials for improving their sitting posture in order to prevent long-term health problems. Figure 2 shows visualizations from an example report.

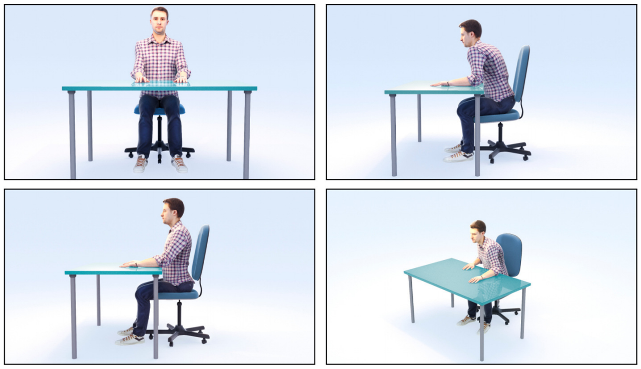

Figure 2. Example visualizations from a sitting posture report.

We assessed the performance of ergoscan on a dataset of 1500 samples, with each depicting one of 31 people assuming the 15 prototype postures. On this dataset, ergoscan estimates 3D keypoint coordinates with an average error of 2 to 5 cm depending on the keypoint, and is able to perform pose classification at an accuracy above 99%.

This paper is structured as follows. Section II summarizes related work on body pose estimation and synthetic dataset generation. Our pose estimation and classification methods are described in Sections III and IV, respectively. Section V describes the experiments and discusses the results, and Section VI concludes the paper.

II. Related Work

Human pose estimation in color images via CNNs is a popular research topic. Two seminal works in this field are [9] and [10]. Both use networks with fully-connected layers for regressing keypoint coordinates. The former work is the first to demonstrate that 3D poses can be recovered from color images, although this is possible only up to scale. More recent works such as [6][7][11] instead perform dense keypoint prediction with fully convolutional networks, which improves accuracy but is slower and requires more memory due to the the additional upsampling path. As these resources are limited on ergoscan devices, we opt for keypoint regression.

In contrast, there is lack of recent works that utilize depth data. A reason for this is that these sensors are not as widespread as cameras (and camera phones), and consequently a lack of large datasets. Pose estimation in depth data was a popular research topic following the release of the Kinect depth sensor in 2010 [12][13][14]. In contrast to these methods, which perform pose estimation via regression forests, we utilize CNNs for this task due to their higher performance. This was shown in [15], which presented a patch-based method for 3D pose estimation in depth data using a combination of a CNN and a recurrent neural network. Two more recent methods are [16] and [17]. The former utilizes a CNN for estimating the coefficients of a linear combination of prototype poses that result in the pose depicted in the input depth map. The latter both processes and predicts 3D volumes, arguing that regressing 3D poses directly from 2D depth maps hinders optimization during training. Our method processes 2D depth maps, which is computationally more efficient, and avoids such problems during training via two-stage keypoint regression.

Synthetic datasets are a promising means for enabling data-driven solutions in problem domains for which no comprehensive datasets are available. To our knowledge [12] was the first work to demonstrate the potential of this approach for 3D pose estimation in depth maps. We adopt this approach and train the CNN on synthetic depth maps, however we create both body poses and 3D models in software rather than employing actors and motion capturing, which is less labor-intensive and requires no special equipment. The most comprehensive public dataset that includes depth maps of people is SURREAL [18]. However, this dataset does not reflect our problem domain in terms of body poses and keypoints.

III. Pose Estimation

Our pose estimation method takes a depth map and a face bounding box as the input and outputs $K = 6$ 3D keypoints.

A. Preprocessing

Preprocessing entails converting the input to be compatible with the CNN. First, the face bounding box is extended by a factor of four in order to capture the head and upper body of the monitored person. The resulting depth map patch is extracted and resampled to a fixed size of 96×96 pixels. This follows normalization of the pixel values, which are given in mm, based on the operating conditions of the ergoscan system, namely assuming a maximum person distance of $d_{max} = 1500$ mm as well as considering that the sensor is unable to measure distances closer to $d_{min} = 400$ mm. To do so, pixels greater than d_{max} are set to 0, which effectively removes most background objects, and then all values are scaled linearly from $[d_{min}, d_{max}]$ to $[0, 1]$ to facilitate transfer learning. While the resulting *normalized distances* are no longer metric, relative distances are preserved and the original distances can be recovered via the inverse mapping.

B. Pose Estimation Network

Pose estimation is carried out by a CNN that first predicts image coordinates of all keypoints and, on this basis, the corresponding normalized distances. We found this approach to be more stable during training, as suggested in [17].

The network architecture is illustrated in Figure 3 and consists of three stages. The first stage performs feature extraction, from which the second stage regresses image coordinates. This follows a novel stage for distance prediction that integrates information from the previous stages and the input image. The outputs are the image coordinates predicted by the second stage and the corresponding distances from the third stage.

The feature extraction stage is a ResNet-18 [19] that was pre-trained for classification on ImageNet [20] and then fine-tuned for pose estimation in depth data. We chose this architecture due to its high performance and ability to run on the target hardware at the required speed. After pre-training, we replaced the classifier with a keypoint regressor that forms the second stage of the network, and performed fine-tuning.

This second stage regresses keypoint image coordinates using the features extracted in the previous stage. The layer

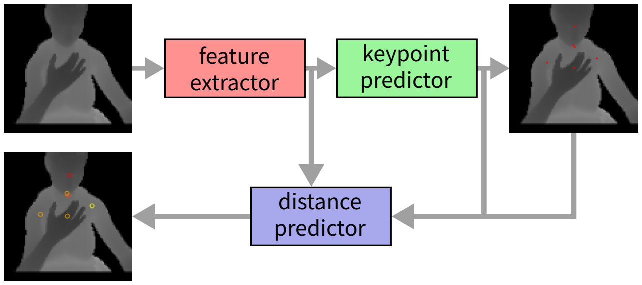

Figure 3. Overview of the stages and data flow through the network.

architecture is detailed in Table I and starts with a global average pooling layer to convert the feature tensors to vectors. This follows a linear layer with a ReLU activation and final linear layer with $2K$ neurons (K keypoints with two coordinates). Batch normalization and dropout layers are utilized to facilitate network optimization and for regularization, respectively.

TABLE I. IMAGE COORDINATE REGRESSION ARCHITECTURE.

Layer type
Global average pooling
Batch normalization
Dropout ($p = 0.25$)
Linear (512 neurons)
ReLU
Batch normalization
Dropout ($p = 0.25$)
Linear (12 neurons)

The first and second stages were trained together as follows. First, we trained only the second stage while utilizing the first (pre-trained) one as a static feature extractor. Once the validation loss had saturated, we fine-tuned both stages to allow the network to adapt to the depth data. This approach prevented the first stage from adapting in a detrimental way due to errors made initially by the untrained second stage. We minimized the Huber loss [21] between predicted and ground-truth image coordinates using stochastic gradient descent with a cyclic learning rate and momentum [22].

The distance regression stage predicts normalized distances for all keypoints. Our goal when designing this stage was to make all relevant information available to it, namely the keypoint coordinates predicted in the previous stage but also the corresponding normalized distances in the input depth map as well as the features extracted by the first stage. This is realized using a *distance lookup layer* that converts the predicted image coordinate vectors to integral coordinates and uses this information to access the corresponding normalized distances in the input depth map. If an image coordinate is out of bounds or if there is no distance information available, the layer assigns a normalized distance of -1 to signal the later stages that there are missing data. The layer returns the original keypoint coordinates as well as the corresponding distances, i.e. a $B \times 3K$ tensor with B being the minibatch size.

Table II summarizes the layer composition of the stage. Initially, there are two parallel branches. The branch that processes features starts identically to the keypoint prediction stage and outputs a $B \times F$ tensor. The other branch consists of the distance lookup layer. Dropout is omitted in this branch to preserve information. The outputs of both branches are then concatenated to a single $B \times F + 3K$ tensor. The remaining layers are consistent with the keypoint regression stage.

TABLE II. DISTANCE REGRESSION ARCHITECTURE.

Input-Features	-Keypoints	-Images
Global average pooling Dropout ($p = 0.25$)	Distance lookup	
Concatenation		
Batch normalization		
Linear (512 neurons)		
ReLU		
Batch normalization		
Dropout ($p = 0.25$)		
Linear (6 neurons)		

We added and trained this stage after the previous stages, which were not modified in this process. We again minimized the Huber loss but did not penalize errors for keypoints whose ground-truth image coordinates were outside the image.

C. Synthetic Training Set

The pose estimation network was trained solely on synthetic data. The key considerations during dataset design were realism and comprehensiveness in order to ensure that trained models would be able to generalize to actual sensor data. To this end, the goal was to capture a wide variety of realistic body types, poses, and office environments.

Pose animations were carried out using the *Blender* 3D modeling software with the *ManuelBastioniLAB* addon for person models and animations. These tools enable realistic and anatomically correct person animations in 3D. The amount and types of modeled poses were chosen based on studies on sitting postures in office environments as well as own analyses of office recordings. This was to ensure that the resulting set of poses would be both realistic and comprehensive. 5000 different poses were generated.

On this basis, 15000 different 3D person models were created. This highlights the potential of synthetic data – recruiting this many people for recording is infeasible for most companies and research institutes. Care was taken to ensure that the models capture a wide variety of realistic body shapes. For this purpose, character properties such as gender, age, height, weight, and body tone were varied, with each affecting body and face shapes in a realistic way. Each 3D model has hair and clothing for increased realism, and includes accurate ground-truth pose information. Figure 4 shows two examples.

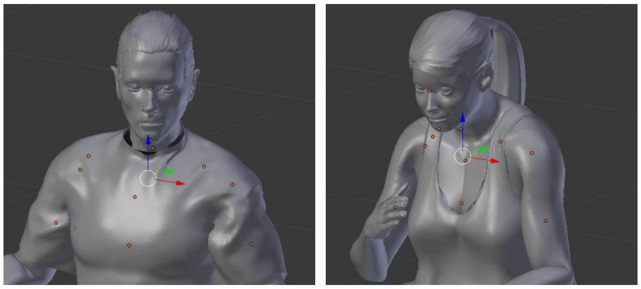

Figure 4. 3D person models for training purposes.

ISBN: 978-1-61208-759-7

Depth maps were rendered by a custom renderer that randomly selects a person, a desk and chair for the person, and a piece of background furniture from a pool of available 3D models, arranges these objects in the scene in a realistic way, places a virtual camera at a varying location and orientation, and then renders a depth map. The renderer also exports metadata such as camera parameters and image coordinates of keypoints. All 3D models of furniture were manually selected from the ShapeNet dataset [23]. The pool of background furniture comprised 1663 models of shelves, cupboards, and couches. The virtual camera had VGA resolution and a field of view of 60 degrees, similar to current off-the-shelf depth sensors. Figure 5 shows a rendered depth map.

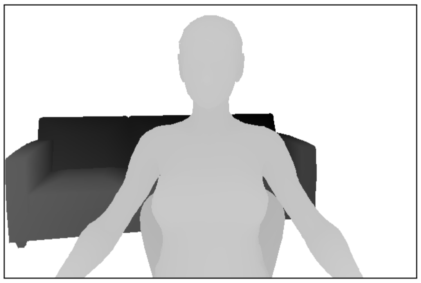

Figure 5. Visualisation of a rendered depth map (further objects appear darker). The desk is outside the field of view in this example.

For increased realism, sensor noise similar to that of the Kinect was simulated. Our method for simulating unsuccessful measurements (zero-pixels) was based on [24]. For each depth map, we computed a smoothed normal map and set depth map pixels whose normals were close to perpendicular to 0. We then applied additive Gaussian random noise with a standard deviation based on the measured distance, in approximation of the random noise of the sensor [25].

D. Postprocessing

To obtain a 3D pose from the CNN output, predicted normalized distances are first mapped to distances from the sensor. The predicted image coordinates are then converted to camera coordinates using the known sensor intrinsics and mapped distances, which in turn are mapped to world coordinates using the extrinsics estimated during system calibration.

IV. Pose Classification

Pose classification is based on angles rather than absolute world coordinates. This angle representation has the advantage of being invariant to the offset between monitored people and the sensor, which generally varies over time. It also increases robustness with respect to variations in person height as angle representations are invariant to uniform scaling.

Our angle representation of a given 3D pose is a collection of 20 informative 2D angles. We favor this 2D approach as we consider such angles – and derived classification rules – more intuitive than 3D angles. Each angle is calculated by computing the 3D vector between two specific keypoint coordinates, discarding a particular coordinate to obtain a 2D vector, and computing the angle between this vector and $(1, 0)$.

The rules for obtaining these angles, i.e. which 3D vectors to compute and which coordinates to discard, were found via feature selection on a training set consisting of 33000 3D poses estimated by several ergoscan devices. Each of these poses was assigned a ground-truth class label by experts. Feature selection was carried out by computing all 90 possible 2D angles for each of these poses, training a random forest on the resulting dataset, and determining the 20 most important angles according to the feature importances learned by the forest [8].

V. Experiments

We assessed the pose estimation and classification performance of ergoscan, and studied the performance impact of training on synthetic data.

A. Dataset

As there were no public datasets available that reflect our problem domain, we created such a dataset ourselves. For this purpose, 31 people were recruited, which were assuming the 15 prototype sitting poses under supervision. During this time, an ergoscan system computed 1500 pose estimates and classifications (100 per pose). Ground-truth keypoints and class labels for these samples were obtained using a professional motion capture system and manual labeling, respectively.

B. Pose Estimation Performance

We calculated the estimation error for a given pose estimate and keypoint as the Euclidean distance between the 3D keypoint coordinate measured by ergoscan and the corresponding ground-truth coordinate. We did so for each keypoint and sample, and report averages and standard deviations.

Figure 6 summarizes the results. The estimation errors are under 30 mm on average, with the exception of the shoulder keypoints. For the shoulders, the CNN often predicted keypoints that were too low. This was caused by missing data for the upper parts of the shoulders due to sensor limitations. The results are promising and confirm that it is feasible to train a CNN for pose estimation in depth data on synthetic data.

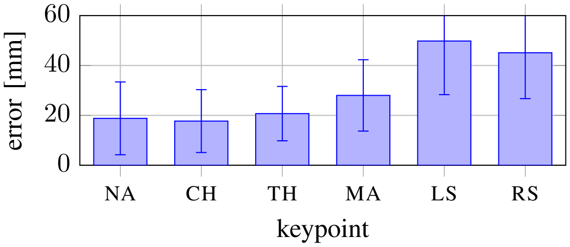

Figure 6. 3D keypoint estimation errors on the test dataset. NA: nasion, CH: chin, TH: throat, MA: manubrium, LS: left shoulder, RS: right shoulder.

C. Generalization Performance

We studied the decrease in performance incurred by training on synthetic data, which despite our efforts do not (and arguably cannot) perfectly match real sensor data. For this purpose, we split the synthetic dataset into a training set (52000 samples) and a test set (8000 samples), and retrained the network using the same hyperparameters. We then computed the per-keypoint estimation errors on the test set like before.

The results are shown in Figure 7. As expected, the estimation errors are significantly lower on synthetic data than on real data (Figure 6). As care was taken to render the synthetic data as realistic as possible, this indicates that a decrease in performance must be accepted in general when training on synthetic depth data.

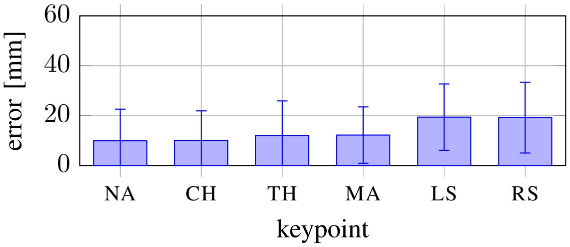

Figure 7. 3D keypoint estimation errors on the synthetic test set. NA: nasion, CH: chin, TH: throat, MA: manubrium, LS: left shoulder, RS: right shoulder.

D. Pose Classification Performance

Ergoscan misclassified only one of the 1500 samples in the test dataset. This confirms that ergoscan is able to classify poses with high accuracy and consequently that ergoscan can detect unhealthy sitting poses reliably.

VI. Conclusion and Future Work

We have presented ergoscan, a system for promoting a healthy posture in office workers by raising awareness. Ergoscan requires no user participation and monitors people's postures over several days to identify unhealthy postures that are frequently assumed. Posture monitoring is realized using a CNN for upper-body pose estimation with an architecture optimized for depth data analysis. On this basis, ergoscan automatically assigns each pose estimate to one of 15 common sitting poses. The results confirm that training CNNs on synthetic data can be a suitable approach if no comprehensive real datasets are available, and that ergoscan is able to perform pose estimation and classification reliably. We plan to investigate performance penalties due to synthetic training data in a more detailed and general way, and on this basis to develop improved sensor noise simulation methods. Another task planned for the future is providing realtime feedback to users via a website or smartphone app in addition to the reports.

Acknowledgment

This work was supported by the Austrian Research Promotion Agency (FFG-855696).

References

[1] S. Reinecke, R. Hazard, K. Coleman, and M. Pope, "A continuous passive lumbar motion device to relieve back pain in prolonged sitting," Advances in industrial ergonomics and safety IV, 2002, pp. 971–976.

[2] A. M. Lis, K. M. Black, H. Korn, and M. Nordin, "Association between sitting and occupational LBP," European Spine Journal, vol. 16, no. 2, 2007, pp. 283–298.

[3] B. Cagnie, L. Danneels, D. Van Tiggelen, V. De Loose, and D. Cambier, "Individual and work related risk factors for neck pain among office workers: a cross sectional study," European Spine Journal, vol. 16, no. 5, 2007, pp. 679–686.

[4] N. J. Delleman, C. M. Haslegrave, and D. B. Chaffin, Working Postures and Movements. CRC Press, 2004.

[5] S. Wu, M. Kan, Z. He, S. Shan, and X. Chen, "Funnel-Structured Cascade for Multi-View Face Detection with Alignment-Awareness," Neurocomputing, 2016.

[6] A. Newell, K. Yang, and J. Deng, "Stacked Hourglass Networks for Human Pose Estimation," in European Conference on Computer Vision. Springer, 2016, pp. 483–499.

[7] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," CoRR, vol. abs/1703.06870, 2017.

[8] L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, 2001, pp. 5–32.

[9] S. Li and A. B. Chan, "3D Human Pose Estimation from Monocular Images with Deep Convolutional Neural Network," in Asian Conference on Computer Vision, 2014, pp. 332–347.

[10] A. Toshev and C. Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks," in Conference on Computer Vision and Pattern Recognition, 2014, pp. 1653–1660.

[11] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep High-Resolution Representation Learning for Human Pose Estimation," in Conference on Computer Vision and Pattern Recognition, 2019, pp. 5693–5703.

[12] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-Time Human Pose Recognition in Parts from Single Depth Images," in Conference on Computer Vision and Pattern Recognition, 2011, pp. 1297–1304.

[13] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon, "Efficient regression of general-activity human poses from depth images," in International Conference on Computer Vision, 2011, pp. 415–422.

[14] H. Yub Jung, S. Lee, Y. Seok Heo, and I. Dong Yun, "Random Tree Walk toward Instantaneous 3D Human Pose Estimation," in Conference on Computer Vision and Pattern Recognition, 2015, pp. 2467–2474.

[15] A. Haque, B. Peng, Z. Luo, A. Alahi, S. Yeung, and L. Fei-Fei, "Towards Viewpoint Invariant 3D Human Pose Estimation," in European Conference on Computer Vision, 2016, pp. 160–177.

[16] M. J. Marín-Jiménez, F. J. R. Ramírez, R. Muñoz-Salinas, and R. Medina Carnicer, "3D human pose estimation from depth maps using a deep combination of poses," CoRR, vol. abs/1807.05389, 2018.

[17] G. Moon, J. Yong Chang, and K. Mu Lee, "V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map," in Conference on Computer Vision and Pattern Recognition, 2018, pp. 5079–5088.

[18] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid, "Learning from Synthetic Humans," in Conference on Computer Vision and Pattern Recognition, 2017.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," International Journal of Computer Vision, vol. 115, no. 3, 2015, pp. 211–252.

[21] P. J. Huber, "Robust estimation of a location parameter," in Breakthroughs in statistics. Springer, 1992, pp. 492–518.

[22] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1," CoRR, vol. abs/1803.09820, 2018.

[23] A. X. Chang, T. A. Funkhouser, L. J. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An Information-Rich 3D Model Repository," CoRR, vol. abs/1512.03012, 2015.

[24] C. Xu and L. Cheng, "Efficient Hand Pose Estimation from a Single Depth Image," in International Conference on Computer Vision, 2013, pp. 3456–3462.

[25] K. Khoshelham and S. O. Elberink, "Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications," Sensors, vol. 12, no. 2, 2012, pp. 1437–1454.