Learning to Infer the Depth Map of a Hand from its Color Image

1st Vassilis C. Nicodemou
Comp. Science Dept.
University of Crete
Heraklion, Greece
nikodim@ics.forth.gr

2nd Iason Oikonomidis
Inst. of Comp. Science
FORTH
Heraklion, Greece
oikonom@ics.forth.gr

3rd Georgios Tzimiropoulos
School of Computer Science
University of Nottingham
Nottingham, UK
yorgos.tzimiropoulos@nottingham.ac.uk

4th Antonis Argyros
Inst. of Comp. Science
FORTH
Heraklion, Greece
argyros@ics.forth.gr

***Abstract*—We present the first direct approach targeted explicitly on human hands that infers depth from monocular RGB images. We achieve this with a Convolutional Neural Network (CNN) that employs a stacked hourglass model as its main building block. Intermediate supervision is used in several outputs of the proposed architecture in a staged approach. To aid the process of training and inference, hand segmentation masks are also estimated in such intermediate supervision steps, and used to guide the subsequent depth estimation process. In order to train and evaluate the proposed method we compile and make publicly available HandRGBD, a new dataset of 20,601 views of hands, each consisting of an RGB image and an aligned depth map. Based on HandRGBD, we explore variants of the proposed approach in an ablative study and determine the most accurate one. The results of an extensive experimental evaluation demonstrate that hand depth estimation from a single RGB frame can be achieved with an accuracy of 22mm, which is comparable to the accuracy achieved by contemporary low-cost depth cameras. Such a 3D reconstruction of hands based on RGB information is valuable as a final result on its own right, but also as an input to several other hand analysis and perception algorithms that require depth input. In this context, the proposed approach bridges the gap between RGB and RGBD, by making all existing RGBD-based methods applicable to RGB input.**

***Index Terms*—hand depth estimation, depth from RGB, CNN, 3D hand pose.**

I. Introduction

The task of observing and understanding human activities is of great interest to the field of computer vision. Among other approaches, human activity can be studied by observing and monitoring the state of the human body, either in 2D or in 3D. Particular emphasis is given to the human hands as the interpretation of their behavior is key to understanding the interaction of humans with their environment. Several efforts have been devoted to this direction and important milestones have been achieved [1], [2]. However, despite the significant progress, a general solution to these problems is still lacking.

This work deals with the problem of estimating the depth map of a hand observed from a regular color camera. Depth information is lost during color image formation and is important for the analysis of hands. We propose a method that accepts as input a conventional RGB image of a hand (Fig. 1, top, left) and produces the depth map of the observed hand (Fig. 1, top, right). Analyzing the rest of the scene (non-hand regions) is out of the scope of this work and the proposed method only marks them as background.

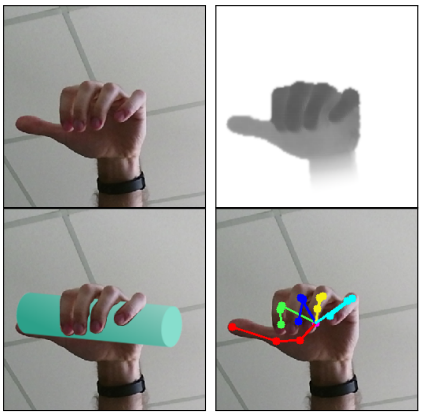

Fig. 1. Given an RGB image of a human hand (top-left) our goal is to produce a depth map of the hand region (top-right). This can support applications such as AR/VR (bottom-left) or 3D hand pose estimation (bottom-right). Note that applications such as AR/VR can exploit 3D hand pose estimation, but many can be supported using only hand depth information.

Solving the problem of hand depth estimation is both interesting and useful. When observing a scene using regular images, it is very appealing to be able to recover the suppressed depth information without stereo 3D reconstruction or structure from motion. Moreover, the recovery of this information may have significant impact on the solution of several practical problems. As an example, the hand depth information can be used to capture and understand hand movement within the 3D space, facilitating tasks such as 3D hand shape and pose estimation, hand-object interaction monitoring, etc, with immediate implications to areas such interaction [3], medical rehabilitation, computer games, and more. For several of the above applications, it suffices to have a 3D reconstruction of a hand, without necessarily solving the 3D hand pose estimation problem. For example, the 3D reconstruction of the hand

The research work was supported by the Hellenic Foundation for Research and Innovation (HFRI) under the HFRI PhD Fellowship grant (Fellowship Number: 803). This work was partially supported by the EU project Co4Robots (H2020-ICT-2016-1-731869). Also co-financed by the European Union and Greek national funds through the Operational Pro-gram Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH – CREATE - INNOVATE (project code:T1EDK-01299 - HealthSign)

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

suffices to support the realistic blending of a real hand with a virtual object (Fig. 1, bottom, left). 3D hand pose would be exploitable, but not required. At the same time, if the 3D reconstruction of a hand can be achieved from a single RGB frame, the inferred depth information can be fed to a depth-based 3D hand pose estimation method (Fig. 1, bottom, right).

II. Related Work

Depth from color for general scenes: The general problem of extracting depth information from color images is very interesting [4]–[8], and is still a research topic under investigation [9]–[11], remaining unsolved in its full generality. The recent success of machine learning, including (most notably) deep neural networks has led to new methods that achieve increasingly better performance and more accurate results [12]–[21]. A significant category of methods deals with the problem of estimating the 3D surface of a deformable object [22], [23]. However, these methods are designed to tackle deformations of paper and cloth, making the deformation model unnecessarily complicated for the case of human hands.

Depth from color for human body parts: An older work related to our approach was proposed by Fanello et al. [24]. In that work, the depth of human body parts was recovered using slightly modified color cameras, based on near-infrared illumination of the scene. A more recent line of methods tackle the problem of depth estimation for specific parts of the human body. The first of these methods [25] estimates the face structure from a single color image. To do so, it uses volumetric information to train a neural network that was based on a stacked hourglass architecture. More works estimating the 3D structure of human faces shortly followed [26], [27]. In parallel to the works on human face, a similar architecture was proposed, targeting the human body [28]. In that work, training data are derived from accurate models of pose, shape and appearance of the human body. None of these approaches achieves a direct estimation of the depth model of the observed scene. Instead, intermediate steps with higher-level information are used, such as the estimation of the landmark positions of facial features (for faces), or 2D/3D body joints (for the human body). To the best of our knowledge, currently there is no method for estimating depth information from an RGB image of a hand.

Use of depth for 3D hand pose estimation: Most of the recent works in the area [2], [29]–[37] assume the availability of scene depth information, capitalizing on the advent of inexpensive, high-quality depth sensors. Much more recently, a new trend is currently forming [38]–[42] that tackles the problem assuming only monocular RGB input. The performance (estimation accuracy and speed) of the older, depth-based approaches is better than that of the more recent RGB-based ones. This is to be expected given the richer nature of the depth map as input information.

Our contribution: An important goal of this work is to close the gap between depth-based approaches and the newer trend of works based on RGB input. Until today, the only available reliable option for extracting depth information for hands directly, is to resort to depth sensors. Although there exist methods that deal with inferring depth from general scenes [20], [21], initial experiments showed that those methods do not perform accurately on hands while they require considerable resources. Specifically, our proposed network recovers the depth of hands with almost half error, while it requires one third of the network parameters compared to the general-purpose methods. The proposed method estimates hand depth information of comparable accuracy given only regular color images. This constitutes a significant complexity simplification and cost reduction of the sensing process. At the same time, several robust, depth-based hand perception methods become applicable to regular RGB input. In summary, the major contributions of this work are:

- The first method that estimates directly depth information from monocular color views of hands. This is achieved with an accuracy that is comparable to the accuracy of a low cost depth sensor.
- The *HandRGBD* dataset [1] of $20,601$ high resolution RGB hand images that are aligned with their depth maps and involve a variety of hand shapes as well as illumination and background conditions.

III. Hand Depth Estimation from RGB Input

At the core of the proposed approach, a deep neural network undertakes the task of estimating the geometry of a hand observed in a single RGB image. A stacked hourglass model [43] is inspired from parts of the architecture in [25] and is used as our main building block. The resulting network accepts as input a regular RGB image and outputs the estimated hand depth map. The output of the network is a map of relative depths for all hand pixels of the input image. The absolute depth of the hand is a separate problem [5]. We tackle it through an extension of our method, that infers the absolute depth by learning the intrinsics of the input given only the bounding box of the hand (see Section IV-E). Intermediate supervision is used in several intermediate levels of the proposed architecture in a staged approach. To aid the process of training and inference, hand segmentation masks are also estimated in such an intermediate supervision step, and used as guidance for the subsequent depth estimation.

A. Ground Truth Annotation

The training data are assumed to be pairs of aligned RGB and depth hand images. The viewpoint of each image pair is assumed to be identical, i.e., each RGB pixel essentially corresponds to the pixel at the same position in the depth map, as if the two streams were captured from the same center of projection. This kind of data can be obtained using common RGBD sensors like Microsoft Kinect2. Most commercially available RGBD cameras have different sensors for each modality, however the viewpoints are very close, and the availability of depth data enables the alignment of

[1] https://www.ics.forth.gr/hand_rgb2depth

the two streams. To achieve this, an accurate intrinsic and extrinsic calibration of the two sensors is required. Given this, a reprojection of the depth map to the RGB image yields the correspondences between the two images.

Given this capturing process, the training data for a learning approach is already at a usable state and no further manual annotation is required. The only processing that is still required is the segmentation of the RGB and depth channels into foreground (hands) and background (non-hands), as well as the normalization of the depth range into relative depths. To facilitate this, it is assumed that the hand is the object closest to the camera, thus, foreground/background segmentation is easy to perform on the depth map. Specifically, the minimum value D_{min} in the depth map $D(i,j)$ is estimated, corresponding to the distance of the hands' point that is closest to the camera. The indices i and j run on the horizontal and vertical image dimensions. All pixels with depth value within a predefined threshold t to this minimum depth D_{min} are considered as the foreground H. The value $H(i,j)$ of the boolean foreground mask H at point (i,j) is defined as:

$$H(i,j) = D(i,j) < (D_{min} + t). \tag{1}$$

Working with depth maps in millimeters, it suffices to set $t = 300mm$, a maximum estimation of the possible depth difference within a hand. Since the RGB and depth images correspond pixel-to-pixel, the resulting binary segmentation H is valid for the RGB image, too.

Let us denote with $D[H]$ the depth map D masked with the foreground mask H. $D[H]$ is used to compute the average depth value $\overline{D[H]}$ of hand points. $\overline{D[H]}$ is then subtracted from D and a fixed scaling is applied to the depths, bringing the depth values in the range $[-1, 1]$. Specifically, the relative depth map D_T that is used for training is

$$D_T(i,j) = c \cdot \left(D(i,j) - \overline{D[H]} \right), \tag{2}$$

where c is a value that scales the depths in the range $[-1, 1]$. For all cases, it suffices to set c as the inverse of the maximum depth difference of an observed hand. When working with depth values in mm it suffices to use $c = 1/200$. Finally, the background pixels are set to 1, the largest value in the target range, that is essentially used to denote background areas.

B. Stacked Hourglass Architecture

The proposed network is based on the approach of stacked hourglass modules [43], [44]. Intermediate supervision is also applied to the output of each hourglass module, which is a commonly adopted strategy [43]. The resulting architecture is illustrated in Fig. 2. In the following description, the intermediate parts of the network are called stages.

The main building block of the proposed architecture is the hourglass network of [43] built using the residual block of [44][2]. Fig. 3 and 4 illustrate the residual block and the network used. A hourglass module (see Fig. 4) accepts as input

[2]Implementation available online at https://github.com/1adrianb/face-alignment

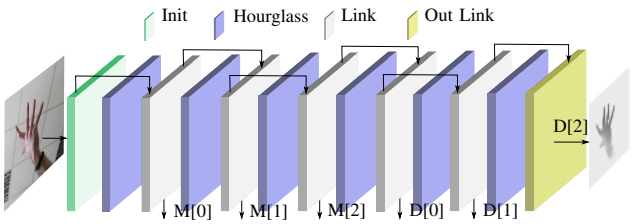

Fig. 2. Stacked Hourglass Architecture: The proposed architecture with the intermediate supervision types. The input is preprocessed by some initialization layers ("Init", light green) that include a convolutional layer and two residual blocks (Fig. 3) and compute a feature map to be passed to the first hourglass (see Fig. 4) module. Its output is passed to layers that apply some additional convolution layers ("Link", gray) before passing it to the next hourglass module. The Link module also outputs a map to be intermediately guided. Skip connections are used parallel to each hourglass module. The first three outputs of the network target segmentation masks and the remaining three target depth maps with the latter being the final output of the network.

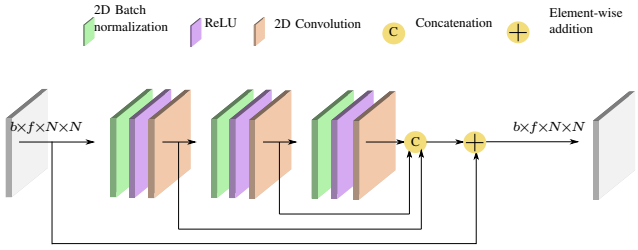

Fig. 3. Residual block: The building block of the proposed neural network for hand depth estimation. The input is assumed to be a feature map of spatial dimension $N \times N$. In the figure, the feature count is f. The batch size b is also shown in the tensor dimensions. The output of the block is usually fed to more than one layers, e.g., to serve a skip connection, as shown here.

a set of feature maps. The residual block of [44] proceeds by applying three successive sets of convolution, batch normalization and ReLU non-linearity operations, using also skip connections, similar to the DenseNet architecture [45]. This is shown in Fig. 3. After these operations, a down-sampling is performed, halving the input dimension. Parallel to this branch with halved spatial dimension, a skip connection runs through another residual block. In total, four repeated residual blocks and resolution halving are applied, and four long-skip connections run in parallel, each at a different spatial resolution. After the last subsampling and application of a residual block, the reverse process is followed, doubling the spatial dimension by upsampling and applying new residual block operations. After each upsampling, the long-skip connection of the appropriate spatial dimension is added to the current feature map. After four upsampling operations, the original input spatial and feature dimension is again reached, forming the complete hourglass module.

For the proposed network, we stack 6 such hourglass modules, having a total of 6 stages of intermediate supervision. A convolution operation is applied to the input image to compute a feature map of appropriate dimension to be the input of the first hourglass module. The reverse process is followed

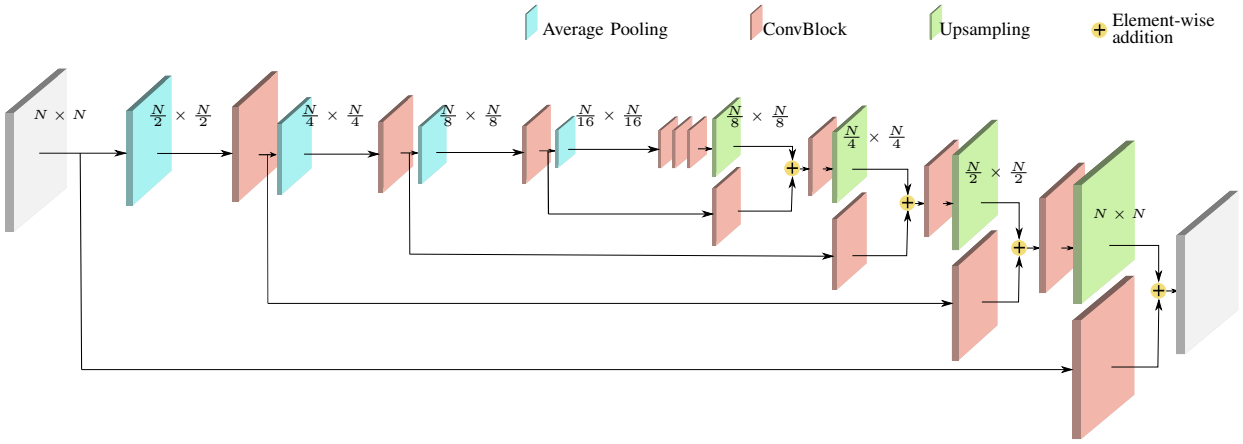

Fig. 4. The hourglass building block that is used in the proposed network. Each input resolution shown in the figure is actually $b \times f \times sr$, where sr are the corresponding spatial dimensions illustrated. Its main building block is the residual block, illustrated in detail in Fig. 3. The main idea is to successively lower the spatial input resolution for a total of four input halving steps. After this, the reverse procedure is followed to reach again the input resolution.

at the end of the network, and at the end of each hourglass module for intermediate supervision. Specifically, a single 1×1 convolution is applied, yielding a single-channel feature map. Each such output is trained against the foreground mask in the first stages, while the later stages are trained against the depth target. We set equal effort for estimating the mask and the depth, thus giving 3 stages for each such estimation.

C. Loss Function

Each stage has its own target output, and thus, its own loss. The global loss function of the network is the sum of the individual losses. For each stage, regardless of the type of intermediate supervision, its loss is obtained by comparing the two images, the predicted and the target image. We define the loss of each stage as the Mean Square Error (MSE) of the target and the output. The final loss function L is:

$$L(m, d, \tilde{m}, \tilde{d}) = \sum_{k=1}^{S_D} \frac{(\tilde{d}_k - d)^2}{||N||} + \sum_{l=1}^{S_M} \frac{(\tilde{m}_l - m)^2}{||N||}, \qquad (3)$$

where d and m are the target depth and mask, each having N pixels, S_D and S_M are the total number of depth and mask stages and $\tilde{d}_k$, $\tilde{m}_l$ are the estimated depths and masks for the kth and lth stage for $k = 1 \dots S_D$ and $l = 1 \dots S_M$.

D. Data Augmentation

During training, data augmentation aims at enriching the diversity of the training set and at increasing the generalization capability of the trained network. In our case, the input is regular RGB images, and common augmentation practices apply. Specifically, we apply (a) random horizontal flip (so that we don't have to capture both hands from a subject) (b) random rotation, (c) random crop and (d) random color jittering (to capture the widest possible range of skin tones and different illumination cases). The geometric transformations are applied to both the RGB and the depth maps, ensuring pixel-to-pixel correspondence. Finally, all data are resized to fit the network's input and output dimensions.

IV. Experimental Evaluation

In some first experiments, we attempted to reconstruct the depth of a hand based on general-purpose depth estimation techniques such as the works of [20], [21]. Specifically, when based on their pre-trained models, the above methods yield an error E of 47.59mm and 35.24mm, respectively, on the test set of *HandRGBD* dataset. We show that on the same dataset, the proposed method reduces this error by almost 50%. Another category of experiments on the *HandRGBD* dataset assess the adopted design choices and define meta-parameters of the proposed approach in an ablation study. We also evaluate our approach on the publicly available Stereo Hand Tracking (SHT) dataset [46]. Finally, we assess the potential of the proposed method to support other methods that perform depth-based 3D hand pose estimation.

A. Hand-related Datasets

Works related to hand appearance and shape modeling as well as pose estimation, require datasets appropriately annotated with ground truth for the purposes of objective, quantitative comparison of competitive approaches, and also - whenever applicable - for training. Therefore, numerous datasets have been proposed so far in the relevant literature.Input modalities such as monocular RGB, stereo, multiview, and depth are covered. Also, scenarios including egocentric viewpoint, hand-object interaction, and hand-hand interaction are available.

The training and evaluation tasks of the problem we are addressing in this work call for a dataset that includes aligned RGB and depth observations of hands. The RGB input should be unaltered, since the goal is to apply our method to regular color input. Some datasets [47], [48] warp the RGB image to the depth map, introducing big black holes in the images that defeat this goal. Another dataset [49] segments the hand in the image and masks the background with a black color. Given that one of our goals is also to learn this segmentation, the

TABLE I
DATASETS ON HUMAN HANDS. FOR THE PURPOSES OF THIS WORK, ALIGNED PAIRS OF RGB AND DEPTH DATA ARE REQUIRED.

Dataset	RGB	Depth	Alignment
Gomez [51]	✓	-	-
Simon [52]	✓	-	-
Bambach [53]	✓	-	-
Dreuw [54]	✓	-	-
Yuen [55]	✓	-	-
Tang [33]	-	✓	-
Sun [56]	-	✓	-
Yuan [57]	-	✓	-
Xu [58]	-	✓	-
Tompson [47]	Warped	✓	-
Tkatch [48]	Warped	✓	-
Rogez [59]	✓	✓	-
Sridhar [32]	✓	✓	-
Zhang [46]	✓	✓	-
Kanhangad [49]	No BG	✓	✓
Zimmermann [38]	✓	✓	✓
Tzionas [50]	✓	✓	✓

dataset becomes unusable. Two additional requirements are the presence of multiple actors and close-up views of the depicted hand(s), so that details on the variation of hand shapes across humans and under articulation are adequately captured.

Table I presents the most relevant datasets to our work. Columns list some of the requirements mentioned above (availability of RGB and depth data and of their alignment). Evidently, only the datasets by Zimmerman and Brox [38], called "Rendered Handpose Dataset" (RHD) and Tzionas et al. [50], called "Hands in Action" have aligned RGB and depth data. Unfortunately, the RHD dataset [38] is synthetic, and, although it has a large variation on hand sizes, shapes and appearances, it is of rather low resolution (320×240) and contains distant views of a hand. The Hands in Action dataset [50] contains real world data, and the depth is captured by a structured light sensor. The actor diversity is small and the view is not closeup, in images of resolution 640×480.

The *HandRGBD* Dataset: Despite the existence of several hand datasets, none of them fully covers the requirements of this work. Consequently, we resorted to creating *HandRGBD*, our own dataset of aligned RGB and hand depth images.

As the capturing device, we employed a Kinect V2 [60] because of its high quality color camera, and the Time of Flight depth sensor. Among the available options, this provided the best combination of image and depth resolution and quality. The native SDK does not provide an alignment of the depth data to the RGB image, only the opposite, resulting in black holes in the RGB image. Therefore, we used the library libfreenect2 [61][3] that supports this functionality, simultaneously scaling and aligning the depth on the color image.

HandRGBD consists of $20,601$ images along with their depth maps. These come from 47 sequences, each consisting of approximately 450 frames. In total, 17 subjects (13 male, 4 female) contributed to the dataset. The depicted hands are in closeup view, in distances ranging from $40cm$ to $100cm$ from the sensor. Some of the captured images contain two hands that interact (strongly, in some cases). All subjects were recorded more than once and in a variety of illumination conditions. Special care was taken to capture the hands in front of different background scenes, facilitating the generalization of foreground/background segmentation. The subjects were instructed to keep their hand(s) roughly in the center of the camera field of view, but some images were also captured with hands close to the image edges. The subjects were also instructed to perform free hand motions, exploring as much as possible the hand articulation space.

The SHT Dataset: Given that the SHT dataset by Zhang et al. [46] is annotated with 3D hand poses, we employed it, mainly for the quantitative evaluation of the proposed approach on the task of supporting 3D hand pose estimation. Since this dataset does not have aligned RGB and depth streams, we manually mapped the depth map on the pixels of the RGB images using the provided calibration.

B. Training Details

We implemented the proposed approach using the PyTorch framework [62]. The Adam optimizer was used to train it for 100 epochs, with a learning rate value of 10^{-3}, weight decay of 10^{-5} and a learning rate scheduler with $\gamma = 0.5$ applied every 30 epochs. For training, we employed an Nvidia GTX 1080 Ti GPU. On that machine, each epoch took about 825 seconds. For all the experiments, the input size to the network was a 256×256 RGB image, and the output a 64×64 depth map. For these resolutions, the inference time for a single image ranges from 7 to 31ms, depending on the number of stages (see also Table II).

We split *HandRGBD* into a training set of $19,104$ samples and a test set of $1,497$ samples, from sequences that are not included in the training set. Specifically, we left aside 3 sequences of our dataset for testing, with 1 female and 2 male subjects. This choice was made because of the ratio between the recorded sequences of female and male subjects. Moreover, each of the three test sequences have backgrounds that appear only in these three sequences. Following the same reasoning, for the experiments on the SHT dataset, 11 out of 12 sequences ($16,500$ samples) were used for training and the remaining sequence ($1,500$ samples) was used for testing.

As already mentioned, data augmentation was used in order to increase the generalization potential of the network. Each training sample was randomly flipped horizontally with probability 0.5. Also, a random rotation in the range of $[-90°, 90°]$ was applied. For the random cropping, a bounding box of size 0.8 of the original size was selected. Finally, a random intensity value in the range of $[-20, 20]$ for each color channel was added for color jittering.

C. Evaluation Metrics

Depth estimation accuracy: For each hand pixel we consider the absolute difference of ground truth and estimated depth. The first error metric E (in mm) is the average of all these differences for all *actual hand pixels* and all frames of a test set.

[3]Available online at https://github.com/OpenKinect/libfreenect2.

TABLE II
ABLATIVE STUDY FOR THE PROPOSED HAND DEPTH ESTIMATION.

Architecture	Error E (mm)	*IoU*	Runtime
0 Mask, 1 Depth Stage	39.75	0.62	$7.07ms$
1 Mask, 2 Depth Stages	33.16	0.65	$16.35ms$
1 Mask, 3 Depth Stages	29.04	0.70	$20.82ms$
1 Mask, 4 Depth Stages	28.05	0.73	$25.73ms$
1 Mask, 5 Depth Stages	28.83	0.72	$30.93ms$
2 Mask, 4 Depth Stages	25.00	0.73	$31.21ms$
3 Mask, 3 Depth Stages	**24.64**	**0.81**	$31.01ms$
4 Mask, 2 Depth Stages	34.42	0.68	$31.51ms$

A second error metric is the percentage $F(e)$ of hand pixels in the test set for which the absolute difference between ground truth and estimated depth is less that a threshold e.

Hand/background segmentation: The proposed method also produces a segmentation of the hand regions from the background. To assess this, we compute the *IoU* (Intersection over Union) criterion for this classification.

D. Ablative Study

We evaluate different architectural choices (Section III-B) based on a subset of *HandRGBD*. Specifically, variants of the proposed method were trained on $4,500$ images (subset of the main, training dataset) and tested on 500 separate images (one of the dataset's test sequences).

An important hyper-parameter of the proposed network is the number of intermediate supervision stages that target the mask segmentation. Experimenting with different training strategies for the proposed network, it became apparent that the hand segmentation mask is an important cue for the task at hand. In a preliminary experiment, the ground truth segmentation mask was provided as a fourth channel concatenated along with the RGB image to the network. This experiment lowered significantly the depth estimation error, indicating that the segmentation mask is indeed useful. It is therefore important to use this cue as an intermediate supervision target, since it aids the task of the network.

In a network with a fixed number of hourglass modules, some of the first hourglass module outputs target segmentation masks and the rest target depths. We performed an experiment to determine the optimal number of stages for each of the two tasks, experimenting also with the total number of hourglass modules. The results of these experiments are summarized in Table II. The results of highest accuracy are highlighted with bold font. From this information we can conclude that, in fact, the segmentation cue is equally important to the depth map itself. The best performing network with six stages was trained with the first three stages targeted as segmentation mask and the rest targeting depth.

E. Hand Depth Estimation Accuracy

Relative depth estimation accuracy: We explored the performance of the best performing variant (line 7 in Table II) when trained on the full training set of *HandRGBD* (Section IV-B). The depth estimation error was $E = 22.88$mm and the estimated *IoU* was equal to 0.84. When the same variant is trained on the training set of SHT and tested on the corresponding test set, we obtain $E = 16.40$mm and *IoU*$= 0.88$. Fig. 5 shows the metric $F(e)$ for both experiments. Finally, when this variant is trained in the union of the two training sets and tested in the union of the test sets, we obtain a depth estimation error E of 19.14mm and *IoU* of 0.87.

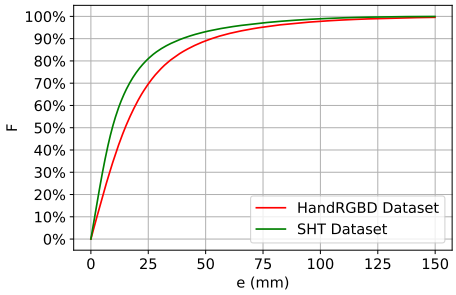

Fig. 5. The error metric $F(e)$ for the depth accuracy estimation experiment on the *HandRGBD* and SHT test sets.

Absolute depth estimation accuracy: A slight modification of the proposed network enables the estimation of an absolute depth map of a hand. The last feature vector of the proposed architecture is used to estimate (a) the relative depth map using a 1×1 convolution as already described, and (b) a single absolute depth value, using as additional input the bounding box of the observed hand within the input frame. Specifically, three stages of convolution and max-pooling are applied, and then two fully connected layers are used to estimate a single value, the median depth, using the quantity $\overline{D[H]}$ of Eq.2 as ground truth. In total, the additional parameters required for this branch are less than 5×10^5, a small percentage of the 3.55×10^7 parameters for the proposed setup using 3 mask and 3 depth stages. Using this modification to the proposed architecture, we achieve an average absolute depth error E of 28.27mm on the test set of SHT with *IoU* equal to 0.86.

F. Supporting 3D Hand Pose Estimation

We assessed the quality of the depth estimated by the proposed method by evaluating the extend at which it can support depth-based 3D hand pose estimation. To do so, we employed the test set part of the SHT dataset on which we applied the Pose-REN 3D hand pose estimation method of Chen et al. [63]. We chose this method because it is a recently proposed approach that achieves close to state-of-the-art hand pose estimation and uses depth information. We applied Pose-REN in two different experimental conditions: **C1**, on the actual depth information of the testset as this was measured by the Intel Real Sense F200 sensor, and **C2**, on the depth that has been estimated by our method. By doing so, we can assess the potential of our method to provide depth maps that are usable by higher-level hand perception methods. We quantified the 3D hand pose estimation error by measuring the average distance of the estimated hand joints from their ground truth locations. We did that for the case of (a) relative depth estimation and (b) absolute depth estimation. For (a), the estimated hand is

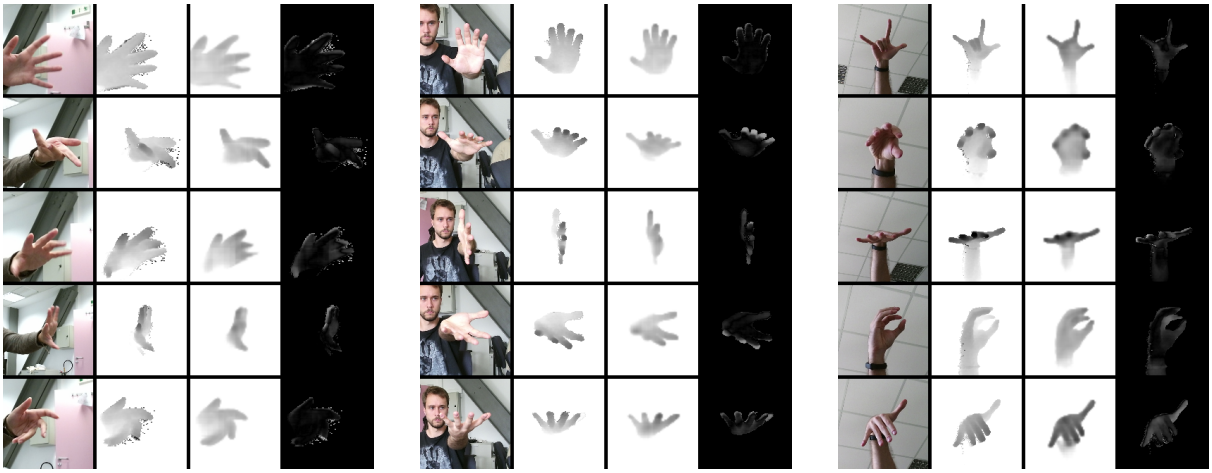

Fig. 6. Indicative depth estimation results on the 3 test sequences of *HandRGBD*. For each sequence (4 columns per sequence) we show the RGB input (1st column), ground truth depth (2nd column), estimated depth (3rd column) and difference between ground truth and estimated depth (4th column).

TABLE III
3D HAND POSE ESTIMATION ERROR (IN mm) OF Pose-REN [63] ON THE SHT DATASET FOR RELATIVE AND ABSOLUTE DEPTHS. **C1**: GROUND TRUTH DEPTHS, **C2**: DEPTHS ESTIMATED BY OUR METHOD.

	Relative depth		**Absolute depth**	
	C1	**C2**	**C1**	**C2**
3D hand pose error	47.87	49.70	51.67	58.26

assumed to be located at its ground truth position. For (b), the accuracy of the estimated 3D hand pose is affected by errors in the estimation of the absolute depth.

Table III summarizes the obtained results. As expected, 3D hand pose estimation is more accurate in the case of relative depths compared to the case of absolute depths. However, the discrepancy between the conditions **C1** (ground truth depth) and **C2** (depth estimated by our method) is smaller. Thus, Pose-REN is only $2-7$mm more accurate when applied to real depth data, compared to when it is applied to the depth data estimated by our method. This is a strong indication that the proposed method can support 3D hand pose estimation.

G. Qualitative Results

Fig. 6 shows representative depth estimation results on three sequences of the test set of *HandRGBD*. For each sequence, we show the input RGB image, the ground truth depth map, the estimated one, and their color-coded difference. It can be verified that the depth maps estimated by our method are very close to the ones measured by the depth sensor.

More experimental results, including representative depth estimation results on the SHT dataset together with their corresponding hand pose estimations, are available in the supplementary material[4] accompanying this paper.

V. CONCLUSION

We presented the first method that has been specifically designed to estimate the depth map of a human hand based on a single RGB frame. The proposed method consists of a specially designed convolutional neural network that has been trained and evaluated on *HandRGBD*, a new dataset of aligned RGB and depth hand images. Extensive experiments evaluated design choices of the proposed method, verified its depth estimation accuracy and provided evidence on the potential of the method to support existing depth-based hand pose estimation methods. The obtained results demonstrate that for the specific context of hands observation, the proposed method constitutes an important step towards turning a conventional RGB camera to an RGBD one. Furthermore, the experimental evaluation of the proposed approach shows that the task-specific intermediate supervision using the hand mask visual cue is more beneficial for the training process than directly using the target depth map for intermediate training. Future plans include the refinement of the results using higher accuracy data acquired by a laser-based depth sensor, as well as testing our method for other depth reconstruction tasks.

[4]https://youtu.be/q0sw8dZ3LlU

REFERENCES

[1] N. Sarafianos, B. Boteanu, B. Ionescu, and I. A. Kakadiaris, "3D Human pose estimation : A review of the literature and analysis of covariates," *CVIU*, vol. 152, pp. 1–20, 2016.

[2] S. Yuan, G. Garcia-Hernando, B. Stenger, G. Moon, J. Y. Chang, K. M. Lee, P. Molchanov, J. Kautz, S. Honari, L. Ge, J. Yuan, X. Chen, G. Wang, F. Yang, K. Akiyama, Y. Wu, Q. Wan, M. Madadi, S. Escalera, S. Li, D. Lee, I. Oikonomidis, A. Argyros, and T.-K. Kim, "Depth-Based 3D Hand Pose Estimation: From Current Achievements to Future Goals," in *CVPR 2018*, 2018. [Online]. Available: http://arxiv.org/abs/1712.03917

[3] O. Hilliges, D. Kim, S. Izadi, M. Weiss, and A. Wilson, "HoloDesk: Direct 3D Interactions with a Situated See-Through Display," *CHI'12*, p. 2421, 2012.

[4] H. G. Barrow and J. M. Tenenbaum, "Interpreting Line Drawings as Three-Dimensional Surfaces," *Artificial Intelligence*, vol. 17, no. 3, 1981.

[5] A. Torralba and A. Oliva, "Depth estimation from image structure," *IEEE Trans. on PAMI*, vol. 24, no. 9, pp. 1226–1238, 2002.

[6] A. Saxena, S. H. Chung, and A. Y. Ng, "Learning Depth from Single Monocular Images," in *NIPS*, 2006.

[7] A. Saxena, M. Sun, and A. Y. Ng, "Make3D : Depth Perception from a Single Still Image," *Aaai*, pp. 1571–1576, 2008.

[8] B. Liu, S. Gould, and D. Koller, "Single image depth estimation from predicted semantic labels," *CVPR*, pp. 1253–1260, 2010.

[9] Z. Chen, V. Badrinarayanan, G. Drozdov, and A. Rabinovich, "Estimating Depth from RGB and Sparse Sensing," *arXiv preprint arXiv:1804.02771*, pp. 1–20, 2018.

[10] A. Pumarola, A. Agudo, L. Porzi, A. Sanfeliu, V. Lepetit, and F. Moreno-Noguer, "Geometry-Aware Network for Non-Rigid Shape Prediction from a Single View," *CVPR*, 2018.

[11] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised Learning of Depth and Ego-Motion from Monocular Video Using 3D Geometric Constraints," in *CVPR 2018*, 2018, pp. 5667–5675.

[12] Y.-H. Lin, W.-H. Cheng, H. Miao, T.-H. Ku, and Y.-H. Hsieh, "Single image depth estimation from image descriptors," in *ICASSP*, 2012.

[13] D. Eigen, C. Puhrsch, and R. Fergus, "Depth Map Prediction from a Single Image using a Multi-Scale Deep Network," *NIPS*, pp. 1–9, 2014.

[14] F. Liu, C. Shen, and G. Lin, "Deep Convolutional Neural Fields for Depth Estimation from a Single Image," *CVPR*, pp. 1–13, 2015.

[15] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," *ICCV*, vol. 2015 Inter, pp. 2650–2658, 2015.

[16] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," *3DV'16*, pp. 239–248, 2016.

[17] R. Garg, B. G. Vijay Kumar, G. Carneiro, and I. Reid, "Unsupervised CNN for single view depth estimation: Geometry to the rescue," *LNCS*, vol. 9912 LNCS, pp. 740–756, 2016.

[18] J. Li, R. Klein, and A. Yao, "A Two-Streamed Network for Estimating Fine-Scaled Depth Maps from Single RGB Images," in *ICCV*, 2017.

[19] L. He, G. Wang, and Z. Hu, "Learning Depth from Single Images with Deep Neural Network Embedding Focal Length," *IEEE Transactions on Image Processing*, no. April, 2018.

[20] S. Niklaus, L. Mai, J. Yang, and F. Liu, "3d ken burns effect from a single image," *ACM Transactions on Graphics*, vol. 38, no. 6, pp. 184:1–184:15, 2019.

[21] J. H. Lee, M.-K. Han, D. W. Ko, and I. H. Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," *arXiv preprint arXiv:1907.10326*, 2019.

[22] A. Varol, M. Salzmann, E. Tola, and P. Fua, "Template-Free Monocular Reconstruction of Deformable Surfaces," in *ICCV*, 2009.

[23] M. Salzmann and P. Fua, "Linear Local Models for Monocular Reconstruction of Deformable Surfaces," *IEEE Trans. on PAMI*, vol. 33, no. 5, pp. 931–944, 2011.

[24] S. R. Fanello, C. Keskin, S. Izadi, P. Kohli, D. Kim, D. Sweeney, A. Criminisi, J. Shotton, S. B. Kang, and T. Paek, "Learning to be a depth camera for close-range human capture and interaction," *ACM ToG*, vol. 33, no. 4, p. 86, 2014.

[25] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos, "Large Pose 3D Face Reconstruction from a Single Image via Direct Volumetric CNN Regression," *ICCV*, vol. 2017-Octob, pp. 1031–1039, 2017.

[26] A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt, "MoFA : Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction," in *ICCV*, 2017, pp. 1274–1283.

[27] A. Tewari, M. Zollhöfer, P. Garrido, F. Bernard, H. Kim, P. Pérez, and C. Theobalt, "Self-supervised Multi-level Face Model Learning for Monocular Reconstruction at over 250 Hz," in *CVPR 2018*, 2018.

[28] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid, "BodyNet: Volumetric Inference of 3D Human Body Shapes," in *ECCV*, 2018, pp. 1–27.

[29] I. Oikonomidis, N. Kyriazis, and A. Argyros, "Efficient model-based 3D tracking of hand articulations using Kinect," in *BMVC*, Dundee, UK, 2011, pp. 101.1–101.11.

[30] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun, "3D hand pose estimation and classification using depth sensors," in *Signal Processing and Communications Applications Conference (SIU)*, 2012.

[31] D. Tang, T.-H. Yu, and T.-K. Kim, "Real-Time Articulated Hand Pose Estimation Using Semi-supervised Transductive Regression Forests," in *ICCV*, 2013, pp. 3224–3231.

[32] S. Sridhar, A. Oulasvirta, and C. Theobalt, "Interactive markerless articulated hand motion tracking using RGB and depth data," in *ICCV*, 2013, pp. 2456–2463.

[33] D. Tang, H. Jin Chang, A. Tejani, and T.-K. Kim, "Latent regression forest: Structured estimation of 3d articulated hand posture," in *CVPR*, 2014, pp. 3786–3793.

[34] M. Oberweger, P. Wohlhart, and V. Lepetit, "Training a feedback loop for hand pose estimation," *ICCV*, vol. 2015 Inter, pp. 3316–3324, 2015.

[35] A. Tagliasacchi, M. Schröder, A. Tkach, S. Bouaziz, M. Botsch, and M. Pauly, "Robust Articulated-ICP for Real-Time Hand Tracking," in *Computer Graphics Forum*, 2015.

[36] C. Wan, A. Yao, and L. Van Gool, "Direction matters: hand pose estimation from local surface normals," in *ECCV*, 2016.

[37] G. Moon, J. Y. Chang, and K. M. Lee, "V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map," *CVPR 2018*, pp. 29–31, 2018.

[38] C. Zimmermann and T. Brox, "Learning to Estimate 3D Hand Pose from Single RGB Images," *ICCV*, 2017.

[39] P. Panteleris, I. Oikonomidis, and A. Argyros, "Using a single RGB frame for real time 3D hand pose estimation in the wild," in *WACV*, 2018.

[40] A. Spurr, J. Song, S. Park, O. Hilliges, and E. Zurich, "Cross-modal Deep Variational Hand Pose Estimation," in *CVPR 2018*, 2018.

[41] Y. Cai, L. Ge, J. Cai, and J. Yuan, "Weakly-supervised 3D Hand Pose Estimation from Monocular RGB Images," in *ECCV*, 2018, pp. 1–17.

[42] U. Iqbal, P. Molchanov, T. Breuel, J. Gall, and J. Kautz, "Hand Pose Estimation via Latent 2.5D Heatmap Regression," in *ECCV*, 2018.

[43] A. Newell, K. Yang, and J. Deng, "Stacked Hourglass Networks for Human Pose Estimation," in *ECCV*, 2016.

[44] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks)," in *ICCV*. IEEE, 2017, pp. 1021–1030.

[45] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *CVPR*, vol. 2017-Janua, 2017.

[46] J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu, and Q. Yang, "3D Hand Pose Tracking and Estimation Using Stereo Matching," *arXiv:1610.07214*, 2016.

[47] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-Time Continuous Pose Recovery of Human Hands Using Convolutional Networks," *SIGGRAPH*, vol. 33, no. 5, pp. 1–10, 2014.

[48] A. Tkach, M. Pauly, and A. Tagliasacchi, "Sphere-meshes for real-time hand modeling and tracking," *ACM ToG*, vol. 35, no. 6, pp. 1–11, 2016.

[49] V. Kanhangad, A. Kumar, and D. Zhang, "Contactless and pose invariant biometric identification using hand surface," *IEEE TIP*, vol. 20, no. 5, pp. 1415–1424, 2011.

[50] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall, "Capturing Hands in Action Using Discriminative Salient Points and Physics Simulation," *IJCV*, vol. 118, no. 2, pp. 172–193, 2016.

[51] F. Gomez-donoso, S. Orts-escolano, and M. Cazorla, "Large-scale Multiview 3D Hand Pose Dataset," pp. 1–23.

[52] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," *CVPR*, vol. 2017-Janua, pp. 4645–4653, 2017. [Online]. Available: http://arxiv.org/abs/1704.07809

[53] S. Bambach, S. Lee, D. J. Crandall, and C. Yu, "Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions," *ICCV*, vol. 2015 Inter, pp. 1949–1957, 2015.

[54] P. Dreuw, T. Deselaers, D. Keysers, and H. Ney, "Modeling image variability in appearance-based gesture recognition," *Proc. of the ECCV 2006 3rd Workshop on Statistical Methods in Multi-Image and Video Processing (SMVP), 12 May, Graz, Austria*, pp. 7–18, 2006.

[55] K. Yuen, "VIVA Hand Tracking Challenge," 2015. [Online]. Available: http://cvrr.ucsd.edu/vivachallenge/index.php/hands/hand-tracking/

[56] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun, "Cascaded hand pose regression," in *CVPR*, 2015, pp. 824–832.

[57] S. Yuan and T.-k. Kim, "BigHand2 2M Benchmark: Hand Pose Dataset and State of the Art Analysis," in *CVPR*, 2017, pp. 15–20.

[58] C. Xu and L. Cheng, "Efficient Hand Pose Estimation from a Single Depth Image," in *ICCV*, 2013.

[59] G. Rogez, M. Khademi, J. S. Supancic, J. Montiel, and D. Ramanan, "3D Hand Pose Detection in Egocentric RGB-D Images," in *ECCVW*, 2014.

[60] M. C. Redmond, "Kinect for XBox One."

[61] L. X. et al, "libfreenect2: Release 0.2," Apr. 2016.

[62] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.

[63] X. Chen, G. Wang, H. Guo, and C. Zhang, "Pose guided structured region ensemble network for cascaded hand pose estimation," *Neurocomputing*, 2018.