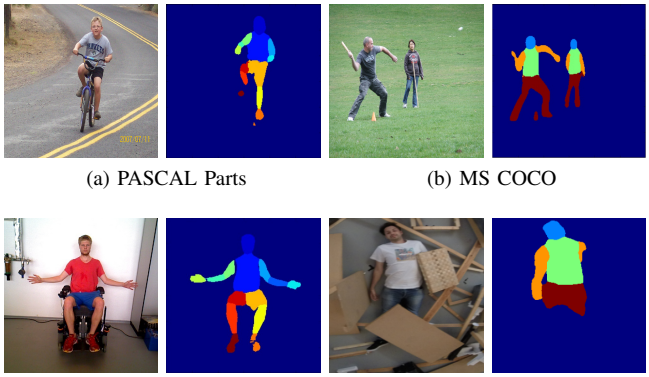
Deep Learning for Human Part Discovery in Images

Gabriel L. Oliveira, Abhinav Valada, Claas Bollen, Wolfram Burgard and Thomas Brox

***Abstract*— This paper addresses the problem of human body part segmentation in conventional RGB images, which has several applications in robotics, such as learning from demonstration and human-robot handovers. The proposed solution is based on Convolutional Neural Networks (CNNs). We present a network architecture that assigns each pixel to one of a predefined set of human body part classes, such as head, torso, arms, legs. After initializing weights with a very deep convolutional network for image classification, the network can be trained end-to-end and yields precise class predictions at the original input resolution. Our architecture particularly improves on over-fitting issues in the up-convolutional part of the network. Relying only on RGB rather than RGB-D images also allows us to apply the approach outdoors. The network achieves state-of-the-art performance on the PASCAL Parts dataset. Moreover, we introduce two new part segmentation datasets, the Freiburg sitting people dataset and the Freiburg people in disaster dataset. We also present results obtained with a ground robot and an unmanned aerial vehicle.**



(a) PASCAL Parts (b) MS COCO

(c) Freiburg Sitting People (d) Freiburg People in Disaster

Fig. 1: Input image (left) and the corresponding mask (right) predicted by our network on various standard datasets.

I. INTRODUCTION

Convolutional Neural Networks (CNNs) have recently achieved unprecedented results in multiple visual perception tasks, such as image classification [14], [24] and object detection [7], [8]. CNNs have the ability to learn effective hierarchical feature representations that characterize the typical variations observed in visual data, which makes them very well-suited for all visual classification tasks. Feature descriptors extracted from CNNs can be transferred also to related tasks. The features are generic and work well even with simple classifiers [25].

In this paper, we are not just interested in predicting a single class label per image, but in predicting a high-resolution semantic segmentation output, as shown in Fig. 1. Straightforward pixel-wise classification is suboptimal for two reasons: first, it runs in a dilemma between localization accuracy and using large receptive fields. Second, standard implementations of pixel-wise classification are inefficient computationally. Therefore, we build upon very recent work on so-called up-convolutional networks [4], [16]. In contrast to usual classification CNNs, which contract the high-resolution input to a low-resolution output, these networks can take an abstract, low-resolution input and predict a high-resolution output, such as a full-size image [4]. In Long et al. [16], an up-convolutional network was attached to a classification network, which resolves the above-mentioned dilemma: the contractive network part includes large receptive fields, while the up-convolutional part provides high localization accuracy. In this paper, we technically refine the architecture of Long *et al*. and apply it to human body part segmentation, where we focus especially on the usability in a robotics context. Apart from architectural changes, we identify data augmentation strategies that substantially increase performance.

For robotics, human body part segmentation can be a very valuable tool, especially when it can be applied both indoors and outdoors. For persons who cannot move their upper body, some of the most basic actions such as drinking water is rendered impossible without assistance. Robots could identify human body parts, such as hands, and interact with them to perform some of these tasks. Other applications such as learning from demonstration and human robot handovers can also benefit from accurate human part segmentation. For a learning-from-demonstration task, one could take advantage of the high level description of human parts. Each part could be used as an explicit mapping between the human and joints of the robot for learning control actions. Tasks such as human-robot handovers could also benefit. A robot that needs to hand a tool to its human counterpart must be able to detect where the hands are to perform the task.

Human body part segmentation has been considered a very challenging task in computer vision due to the wide variability of the body parts' appearance. There is large variation due to pose and viewpoint, self-occlusion, and clothing. Good results have been achieved in the past in conjunction with depth sensors [22]. We show that CNNs can handle this variation very well even with regular RGB cameras, which can be used also outdoors. The proposed network architecture yields correct body part labels and also localizes them precisely. We outperform the baseline by Long et al. [16] by a large

All authors are with the Department of Computer Science at the University of Freiburg, 79110 Freiburg, Germany. This work has partly been supported by the European Commission under ERC-StG-PE7-279401-VideoLearn, ERC-AG-PE7-267686-LIFENAV, and FP7-610603-EUROPA2.

margin on the standard PASCAL parts dataset.

To evaluate the approach directly in a robotics setting, we introduce two new datasets for human body part segmentation: Freiburg Sitting People and Freiburg People in Disaster. They provide high resolution data for experiments on ground and aerial robot segmentation applications.

The paper is organized as follows. We first discuss related work in Section II. In Section III, we present our methodology for human part segmentation including the proposed architecture. Experimental results are described in Section IV. Ongoing work and possible future research directions are discussed in Section V.

II. RELATED WORK

In the context of semantic segmentation, there are several approaches that encode segmentation relations using Conditional Random Fields (CRFs) [1], [18]–[20]. Plath *et al.* [20] present an approach that couples local image features with a CRF and an image classification approach to combine global image classification with local segmentation. Another branch of CRFs called Hierarchical Conditional Random Fields (HCRF) has been introduced by Boix *et al.* [1]. They propose a technique called harmony potential to overcome the problem of classical HCRFs, that they do not allow multiple classes to be assigned to a single region. Maire *et al.* [19] use an alternative people detection and segmentation approach, in which they merge the outputs of a top-down part detector in a generalized eigen problem, producing pixel groupings. Lucchi *et al.* [18] present an analysis of the importance of spatial and global constraints in CRFs when such features have already extracted information from the whole image.

For semantic segmentation, it is also popular to make use of pre or post-processing methods, such as superpixels [6], [11] and region proposals [9], [10]. Farabet *et al.* [6] classify superpixels using a CNN. Classification results are combined to obtain pixel-wise labeling. Gupta *et al.* [9] sample region proposals for detection and semantic segmentation. Hariharan *et al.* [10] introduce an approach that makes use of region proposals for detection and coarse segmentation. They use CNN features to describe the proposals and Support Vector Machines (SVM) to classify them. The results produced from the SVM are coarse masks and in order to improve it, a superpixel classification method is used to refine the initial coarse prediction. Their more recent hypercolumn representation makes use of an additional description of each pixel in the network [11]. A similar approach was applied to segmentation in a robotics context by Liu *et al.* [15]. All these approaches are based on CNN features, but due to the preprocessing, the task of semantic segmentation cannot be trained end-to-end, but requires some engineering for CNNs to be applicable.

In contrast, the so-called fully convolutional network (FCN) developed by Long *et al.* [16] allows training the network end-to-end for the semantic segmentation task. This more elegant approach also led to better performance and provides the state-of-the-art performance in semantic segmentation. The approach replaces the fully connected layers of a deep classification network, e.g. VGG [24], by convolution layers that produce coarse score maps. A successive up-convolutional network allows them to increase the resolution of these score maps. There have been some recent extensions of Long et al. [16]. Chen *et al.* [2] use a fully connected CRF to refine the segmentation maps obtained from [16]. Ronneberger *et al.* [21] applied the approach to cell segmentation in the biomedical context and proposed several technical improvements that allow training from few images and predicting higher resolution outputs. None of these approaches has been applied to human body part segmentation.

Literature includes several works on human or animal part segmentation and person keypoint prediction [12], [23], [26], [28], [29]. Zhang *et al.* [28] perform part detection based on region proposals that are classified using a CNN. The approach was demonstrated on a bird part segmentation dataset. Tompson *et al.* [26] developed an approach that learns an end-to-end human keypoint detector for pose estimation using a CNN. Zhang *et al.* [29] use poselets for part discovery and calculate features for each region using a CNN. Jain *et al.* [12] present a sliding window approach for part localization with CNNs. They employ a CNN at each position of the window to detect human body parts. This requires thousands of CNN evaluations and considering that the time for each evaluation is not negligible, the method yields long run times. Simon *et al.* [23] introduced a CNN approach called *part detector discovery*, which detects and localizes bird parts without training on the specific dataset. The method is based on analyzing the gradient maps of the network and finding the spatial regions related to the annotated parts.

III. METHODOLOGY

A. Problem Definition

Semantic segmentation associates to each pixel of an input image exactly one out of N_{cl} pre-defined class labels. In this paper, the class labels correspond to human body parts at two different granularity levels. In a coarser task, we consider four labels (head, torso, arms, legs). In the finer task, we have 14 labels and distinguish also between the left and right side of the person (head, torso, upper right arm, lower right arm, right hand, upper left arm, lower left arm, left hand, upper right leg, lower right leg, right foot, upper left leg, lower left leg and left foot.

We approach the problem with a CNN that is trained end-to-end to predict the class labels. Training minimizes the usual cross-entropy (softmax) loss. The softmax function converts a score a_K for class K into a posterior class probability $P_K \in [0, 1]$:

$$P_K = \frac{\exp(a_K)}{\sum_{l=1}^{N_{cl}} \exp(a_l)} \tag{1}$$

At test time, the softmax is replaced by the argmax function to yield a single class label per pixel.

B. Architecture

The architecture is based on the network from Long *et al.* [16], where we replaced their up-convolutional part

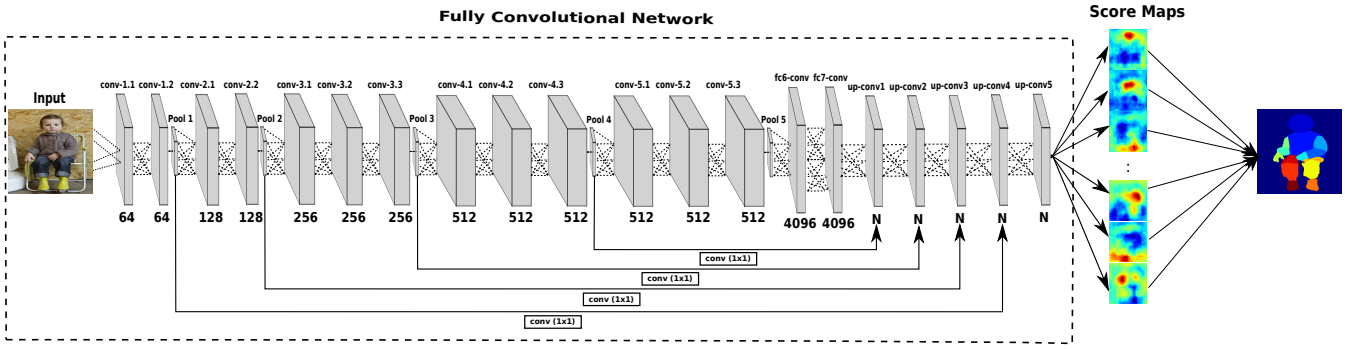


Fig. 2: Proposed architecture. Only convolutional, pooling, and up-convolutional layers are visualized. Up-convolutional layers have size $N = N_{cl}$. We call the network part up to fc7-conv the *contractive network part*, whereas the part after fc7-conv is called the *expansive network part*.

with our own refinement architecture. The whole network architecture is shown in Fig. 2. Like in Long *et al.*, the parameters of the contracting part of the network is initialized with the parameters of the VGG classification network [24].

The proposed refinement architecture is composed of multiple layers, where each layer combines the upsampled output of its previous layer with the pooled features of the corresponding layer of the contracting network part. The first provides the preliminary class scores at the coarse resolution, whereas the second contributes information for refining the resolution. The combination of both is detailed in Fig. 3. The coarse score map is fed into an up-convolutional layer, i.e., it is upsampled by a factor 2 via bilinear interpolation followed by a convolution. We use a ReLU activation function after each up-convolutional operation to better deal with the vanishing gradient problem. The feature map from the contracting network part is fed into a convolutional layer followed by dropout to improve the robustness to over-fitting. The effect of applying dropout to all refinement layers is analyzed in Section IV. Finally, the output of both streams are summed element-wise to yield the output of the refinement layer. This output is again the input for the next refinement layer. Each layer increase the resolution of the segmentation by a factor 2.

With this refinement architecture we manage to obtain a high quality output at the resolution of the input image. This is in contrast to Long *et al.* [16], who stopped their refinement after three layers, because they did not observe any improvement afterwards. A full description of the architecture is presented at Table I.

C. Feature Map Dropout

Another attribute of our proposed approach is to make dropout more robust. To this end, we implemented a new feature map dropout. We expand the random dropout to the entire feature map, which is based on the Spatial Dropout method [26]. One singular characteristic of human body part segmentation is strong spatial correlation, resulting in features that are likely correlated across the map. Hence, dropout must also be correlated. Feature map dropout performs a Bernoulli

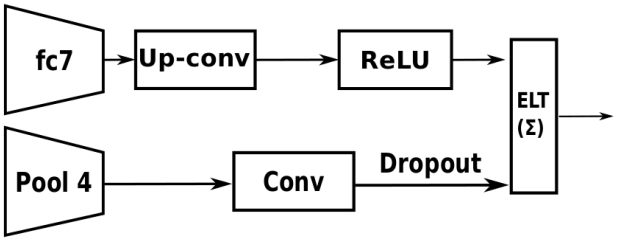


Fig. 3: Description of the first refinement layer. Successive refinement layers have the same architecture, but take different inputs. The upper stream takes the output from the contractive network (fc7) or from the previous refinement layer as input. It applies an up-convolution followed by a ReLU. The lower stream takes high-resolution features from the corresponding layer in the contractive network as input. It applies a convolution followed by dropout (only during training).

trial per output feature during training and propagates the dropout value across the entire feature map.

Given a network with L hidden layers and $l \in \{1, ..., L\}$. Let z^l be the vector of inputs into layer l and let y^l denote the vector of outputs from layer l. W^l and b^l are the weights and biases at layer l. $*$ denote the element-wise product and $r^{(l)}$ is the vector of independent Bernoulli random variables that has probability p of being 1. Feature Map Dropout is then expressed as

$$
\begin{aligned}
r_z^{(l)} &\sim \text{Bernoulli}(p) \\
\tilde{y}^{(l)} &= r^{(l)} * y^{(l)} \\
z_i^{(l+1)} &= w_i^{(l+1)} \tilde{y}^{(l)} + b_i^{(l+1)} \\
y_i^{(l+1)} &= f(z_i^{(l+1)})
\end{aligned}
$$

The variable $\tilde{y}^{(l)}$ is called thinned vector of outputs. This sets apart the feature map dropout from the standard dropout. The resulting thinner network $\tilde{y}^{(l)}$ has the entire feature maps zeroed. For instance, in a convolution layer of size $(1, 64, 20, 20)$, and a dropout of 0.5, approximately 32 of the 64 feature channels will be zeroed after the input passes the dropout layer.

name	kernel size	stride	pad	output size
data	-	-	-	$300 \times 300 \times 3$
conv1_1	3×3	1	100	$498 \times 498 \times 64$
conv1_2	3×3	1	1	$498 \times 498 \times 64$
pool1	2×2	2	0	$249 \times 249 \times 64$
conv2_1	3×3	1	1	$249 \times 249 \times 128$
conv2_2	3×3	1	1	$249 \times 249 \times 128$
pool2	2×2	2	0	$125 \times 125 \times 128$
conv3_1	3×3	1	1	$125 \times 125 \times 256$
conv3_2	3×3	1	1	$125 \times 125 \times 256$
conv3_3	3×3	1	1	$125 \times 125 \times 256$
pool3	2×2	2	0	$63 \times 63 \times 256$
conv4_1	3×3	1	1	$63 \times 63 \times 512$
conv4_2	3×3	1	1	$63 \times 63 \times 512$
conv4_3	3×3	1	1	$63 \times 63 \times 512$
pool4	2×2	2	0	$32 \times 32 \times 512$
conv5_1	3×3	1	1	$32 \times 32 \times 512$
conv5_2	3×3	1	1	$32 \times 32 \times 512$
conv5_3	3×3	1	1	$32 \times 32 \times 512$
pool5	2×2	2	0	$16 \times 16 \times 512$
fc6-conv	7×7	1	0	$10 \times 10 \times 4096$
fc7-conv	1×1	1	0	$10 \times 10 \times 4096$
Up-conv1	4×4	2	0	$22 \times 22 \times N_{cl}$
Up-conv2	4×4	2	0	$46 \times 46 \times N_{cl}$
Up-conv3	4×4	2	0	$94 \times 94 \times N_{cl}$
Up-conv4	4×4	2	0	$190 \times 190 \times N_{cl}$
Up-conv5	4×4	2	0	$382 \times 382 \times N_{cl}$
output	-	-	-	$300 \times 300 \times N_{cl}$

TABLE I: Our architecture in more detail. The Up-conv layers refer to each refinement step. For brevity reasons ReLUs, dropout and some layers from the up-convolution step are omitted from the table.

D. Data Augmentation

We augment the training data by randomly mirroring and cropping the images. Inspired by the data augmentation suggested in Dosovitskiy et al. [5], we additionally apply geometry and color transformations to increase the amount of training data and the robustness of our network to over-fitting. In particular, we implemented the following set of transformations:

- Scaling: Scale the image by a factor between 0.7 and 1.4;
- Rotation: Rotate the image by an angle of up to 30 degrees;
- Color: Add a value between -0.1 and 0.1 to the hue channel of the HSV representation.

Unlike in the setting of Dosovitskiy et al. [5], where rotation and scaling had the lowest impact among all transformations, these spatial augmentation strategies are very important for the task and data considered here, as shown in Section IV.

E. Network Training

Training is performed in a multi-stage process in order to save time. We initialize the contracting part of the network with the 16 layer version of the VGG architecture [24], which is the same as used by Long *et al.* [16]. The base network has small convolution filters (3×3) and 1 pixel stride. The network also has 5 max-pooling layers with 2×2 pixel windows with stride 2. We also considered training the whole network from scratch without initializing it with weights from the VGG network. However, this was inconvenient in terms of time, as the base network approximately takes four weeks to train on a multi-GPU cluster.

The overall network is then trained by backpropagation using Stochastic Gradient Descent (SGD) with momentum. Each minibatch consists of just one image. The learning rate and momentum are fixed to $1e^{-10}$ and 0.99, respectively. We train the refinement layer by layer, which takes two days per refinement layer. Thus, the overall training starting from the pre-trained VGG network took 10 days on a single GPU.

IV. EXPERIMENTS

We evaluated the performance of our network on the PASCAL Parts dataset, a new Freiburg Sitting People dataset, and a new Freiburg People in Disaster dataset. On all three datasets we report quantitative results and compare to results obtained with the state-of-the-art FCN baseline [16]. We fine-tuned the FCN for each dataset on the same training data that was used for training our network. Moreover, we conducted experiments in a direct robotics context with a ground robot and an unmanned aerial vehicle. The implementation was based on the publicly available Caffe [13] deep learning toolbox, and all experiments were carried out with a system containing an NVIDIA Titan X GPU.

A. PASCAL Parts dataset

The PASCAL Parts dataset [3] includes annotations for 20 PASCAL super classes and part annotations for each of them. We focused on the person subset of this dataset, which consists of 3539 images. The annotations even include eyes and ears, which may not seem relevant in a robotics context for now. Therefore we merged labels to two granularity levels, one with just 4 body parts and one with 14 body parts; see Section III-A. Our experiments were based on a fairly recent release, so there are not many works reporting part segmentation results. To the best of our knowledge, the only works reported so far are [17], [27], though none of them have reported results on the person category. Therefore, we present the first quantitative results on person part segmentation for the PASCAL Parts dataset.

As metrics, we chose pixel accuracy and intersection over union. Let n_{ij} be the number of pixels of class i predicted to belong to class j, where $t_i = \sum_j n_{ij}$ be the total number of pixels of class i. The pixel accuracy $Acc = \sum_i n_{ii} / \sum_i t_i$ takes into account also the prediction of background pixels. Background prediction is important to avoid false positives.

The downside of pixel accuracy as a sole measure, however, is the dominance of the background in the metric. More than three quarters of the images are background. Therefore, along with pixel accuracy, we also report the intersection over union (IOU), which is a popular metric for computer vision datasets. It is defined as $IOU = (1/N) \sum_i n_{ii} / (t_i + \sum_j n_{ji} - n_{ii})$. Unlike pixel accuracy, IOU does not take the background detection into account and solely measures the semantic

segmentation of the parts. However, it does penalize false positive pixel assignments.

1) Coarse body parts: We first predicted the coarse segmentation with four body part classes. We randomly divided the dataset into 70% training and 30% testing. Table II shows the results. There is a 5% percentage points improvement over the state of the art in both metrics.

TABLE II: Results on PASCAL dataset with 4 body parts. Table also includes the addition of dropout.

Method	Accuracy	IOU
FCN [16]	71.30	57.35
Ours - No dropout	74.60	61.20
Ours - With dropout	**76.58**	**63.03**

Additionally, we also perform experiments without the feature map dropout at the refinement part of the network. Table II presents our results for the network without feature map dropout at the expansive part of the network and when dropout is included. The addition of the dropout layer brings a considerable gain, in terms of better mean pixel accuracy and IOU. This result confirm that a spatial correlated dropout can benefit from the strong spatial correlation of human body parts. Based on the obtained results all the following experiments will report the proposed approach with the inclusion of the feature map dropout.

TABLE III: Results on the PASCAL dataset with 14 body parts.

Method	Accuracy	IOU
FCN [16]	75.60	53.12
Ours	**77.00**	**54.18**

2) Detailed body parts: When predicting all 14 body parts, we randomly divided the dataset into 80% training and 20% testing. Fig. 4 shows a set of results obtained by our network. The results are organized column-wise, where each column is an example and the rows correspond to input image, ground truth and results obtained using the FCN of Long *et al*. The last row constitutes the results using our network. The results of our approach are closer to the ground truth than the FCN baseline. Table III contains the corresponding quantitative numbers. We outperform the FCN baseline [16] by 1% percentage point in both metrics. The smaller improvement on the more complex task indicates that there was not enough training data to exploit the larger capacity of our network. For the experiments reported so far, we did not make use of any data augmentation. We shall see in the next section that the latter is important, especially for more complex tasks.

B. Effect of Data Augmentation

Apart from the usual mirroring and cropping, we applied two types of augmentations to our training data: spatial augmentations and color augmentation; see the detailed

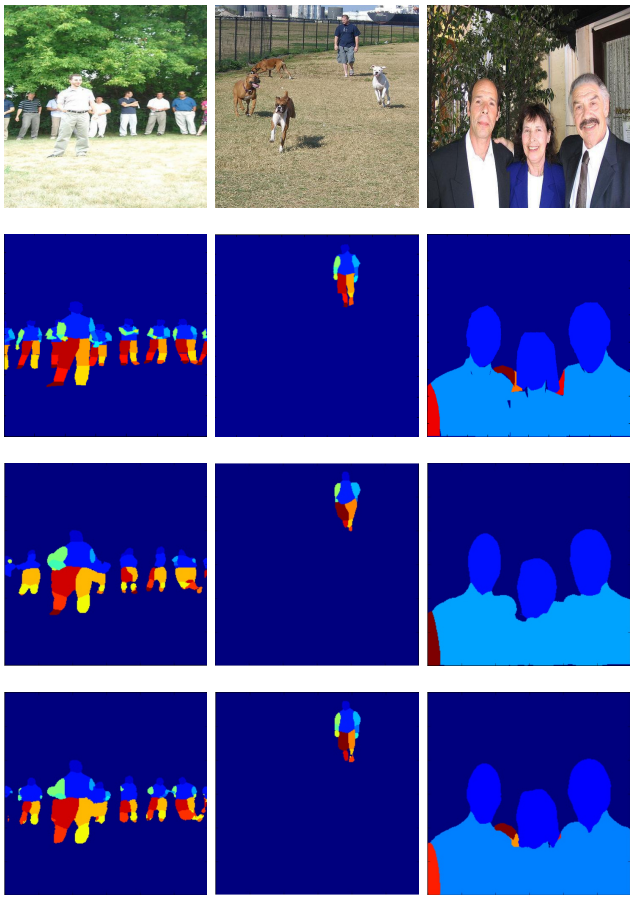


Fig. 4: Qualitative results on the PASCAL dataset (task with 14 body parts). **First row:** Input image. **Second row:** Ground truth. **Third row:** Result predicted with FCN [16]. **Fourth row:** Result predicted by our network. Our approach produces more accurate segmentation masks, not only for single person segmentation but also when there are multiple persons in the image.

description in Section III-D. Table IV shows the impact of these types of data augmentation on the result. Table **??** and Table VI summarize the IOU along with the pixel accuracy for the two granularity levels. Clearly, both types of data augmentation improved results significantly. These results emphasize the importance of a solid data augmentation technique when approaching relatively complex tasks with limited training set sizes. The relative improvement of data augmentation was bigger on the more difficult task with 14 classes, which can be attributed to the fact that, a more difficult task requires more training data.

TABLE IV: Augmentation results Accuracy and IOU on the PASCAL dataset with 4 body parts.

Method	Acc.	IOU				
		Head	Torso	Arms	Legs	All
FCN [16]	71.30	70.74	60.62	48.44	50.38	57.35
Ours	76.58	75.08	64.81	55.61	56.72	63.03
Ours (Spatial)	82.18	80.49	74.39	67.17	70.39	73.00
Ours (Spatial + Color)	**85.51**	**83.24**	**79.41**	**73.73**	**76.52**	**78.23**

TABLE V: Augmentation results (IOU) on the PASCAL dataset with 14 body parts.

Method	Head	Torso	L U arm	L LW arm	L hand	R U hand	R LW arm	R hand	R U leg	R LW leg	R foot	L U leg	L LW leg	L foot	Mean
FCN [16]	74.0	66.2	56.6	46.0	34.1	58.9	44.1	31.0	49.3	44.5	40.8	48.5	47.6	41.2	53.1
Ours (Spatial)	81.8	78.0	69.5	63.1	59.0	71.2	63.0	58.7	65.4	60.6	52.0	67.9	60.3	50.0	66.9
Ours (Spatial+Color)	**84.0**	**81.5**	**74.1**	**68.0**	**64.0**	**75.4**	**67.4**	**61.9**	**72.4**	**67.1**	**56.9**	**73.0**	**66.1**	**57.7**	**71.7**

R = right, L = left, U = upper, LW = lower.

TABLE VI: Augmentation summary on the PASCAL dataset with 14 body parts.

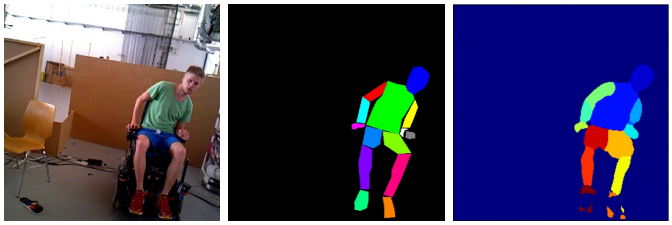
Method	Accuracy	IOU
FCN [16]	75.60	53.12
Ours	77.00	54.18
Ours (Spatial)	84.19	66.93
Ours (Spatial + Color)	**88.20**	**71.71**

TABLE VII: Results with and without training on the Freiburg people dataset.

Method	Accuracy	IOU
FCN [16]	59.69	43.17
Ours (Trained on PASCAL)	78.04	59.84
Ours (Training with 2 people, Testing with 4)	**81.78**	**64.10**

C. Freiburg Sitting People Part Segmentation Dataset

To evaluate the proposed part segmentation approach in a robotics task, we created a new dataset [1] that provides high resolution segmentation masks for people in sitting position, specifically people in wheelchairs. Fig. 5a, presents the input sample image and Fig. 5b its groundtruth, while Fig. 5c shows the segmentation prediction. The dataset has 200 images of six different people in multiple viewpoints and in a wide range of orientations. The ground truth annotation contains the 14 body part classes as used for the PASCAL parts dataset.



(a) Input Image (b) Groundtruth (c) Predicted Mask

Fig. 5: Results on the Freiburg Sitting People dataset.

Due to the unavailability of a large amount of data, we chose two different testing scenarios. First we trained our network on the PASCAL parts dataset, and used the full sitting people dataset for testing. Alternatively, we randomly chose two people from the dataset for training (along with the data from PASCAL parts) and the remaining four as the test set. Results are shown in Table VII. Obviously, the network generalized well to new datasets. The improvement over the FCN baseline was much larger than the difference between the network that had seen sitting people for training and the one that had not. Nonetheless, providing training data that is specific to the task helped improve the performance.

Another aspect of the network which is of great interest for robotic tasks is time performance. Our network can process a single frame in 229 ms, so providing more than 4 frames per second. Long et al. [16] provides inference times ranging from 150 to 175 ms. Our approach having a deeper refinement architecture presents a higher computational load. For an output smaller than the 300×300 used in our experiments, higher frames can be expected.

[1] http://www2.informatik.uni-freiburg.de/~oliveira/dataset.html

(a) Input Image (b) FCN [16] (c) Predicted Mask

Fig. 6: Segmentation results on the Microsoft COCO dataset. Our approach yields more details; see the arm detection and the line between torso and lower limbs. This is because the architecture yields the full resolution of the input image also for the output.

D. Microsoft COCO

Microsoft COCO constitutes a very large dataset for semantic segmentation. However, its focus is on whole objects and part annotations are not provided. Therefore, we cannot report quantitative results. Fig. 6 shows a sample result obtained on this dataset.

E. Real-World Robot Experiments

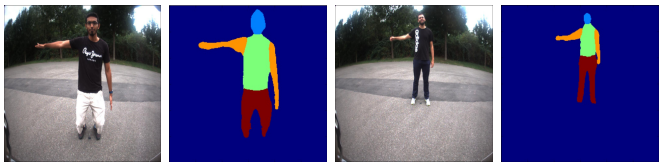
In this section we present experimental results performed using real robots. First, we performed experiments with a ground robot and measured how robust the technique is to scale changes. The ground robot used for the experiments is the Obelix robot, Figure 7a. Obelix is a robot designed for autonomous navigation in pedestrian environments and it is useful to mimic the human perspective for perception tasks.

In our experiments, we obtained data from a Bumblebee stereo camera. The main goal of the experiments performed with Obelix was to measure the response of the system to

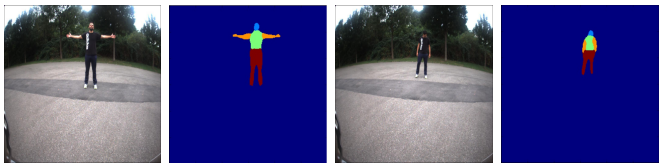
(a) Obelix ground robot.

(b) AR.DRONE 2.0 aerial platform.

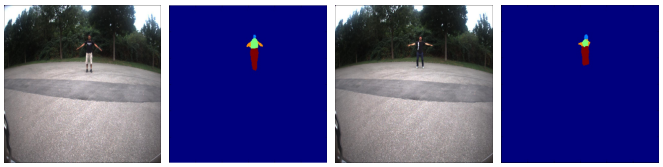
Fig. 7: Robotics platforms used in our tests.



(a) 1.0 meter (b) 2.0 meters



(c) 3.0 meters (d) 4.0 meters



(e) 5.0 meters (f) 6.0 meters

Fig. 8: Qualitative results of the range experiment with the Obelix robot. The lower resolution at one point does not allow detection of small body parts. However, the larger parts, such as the torso and head, are still detected correctly even at 6m distance.

different observation distances, as shown at Figure 8. For that, we segmented outdoor images of two different people on distances ranging from 0.8 m to 6.0 m, capturing images every 20 cm. Fig. 9 presents the results. There was no indication of a scale bias. The performance dropped proportionally with the decreasing resolution of the person in the image, which was expected, as there are fewer details of body parts visible at lower resolutions.

For a second experiment with a real robot we used an AR.DRONE 2.0 aerial platform, Figure 7b. Since the platform lacks a high definition downward facing camera, we mounted a GoPro HERO 4 to collect the Freiburg People in Disaster

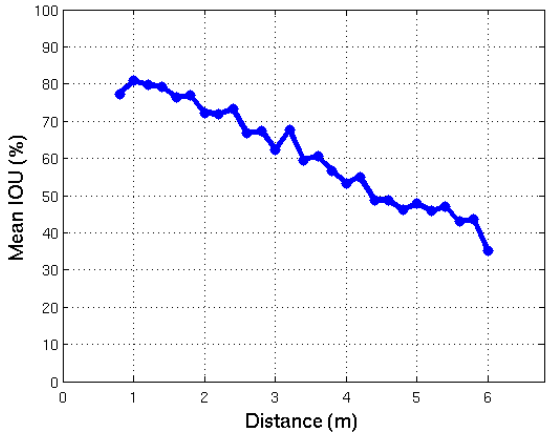
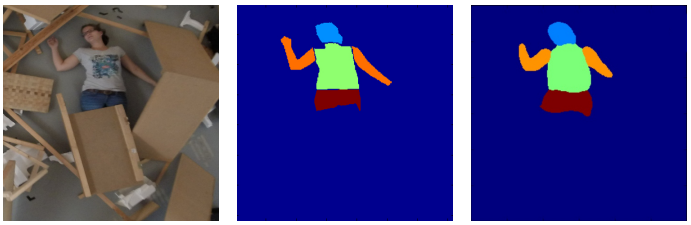


Fig. 9: Quantitative performance in terms of IOU with the Obelix robot taking pictures from 0.8 m to 6.0 m distance to the segmented person. The performance drops proportionally with the resolution of the person in the image.



(a) Input Image (b) Groundtruth (c) Segmentation mask

Fig. 10: Prediction of our network for the Freiburg People in Disaster Dataset.

dataset. The dataset consists of images and corresponding segmentation masks for a set of 4 people in an environment that mimics a disaster scenario, with clutter and heavy occlusion around. Figure 10 shows an example with the groundtruth and the results obtained using our approach.

As the dataset is rather small, we did not use it for training the network but just for testing. We used the network trained on the PASCAL dataset with 4 body parts. Table VIII presents the results for this dataset. Our approach performed 28% better than the state of the art FCN. While the FCN performance would be too weak for a robot to rely on, the results obtained with our network and data augmentation can already be useful in practice.

TABLE VIII: Results for Freiburg People in Disaster dataset.

Method	Head	Torso	Arms	Legs	IOU
FCN [16]	52.71	62.49	35.04	43.25	43.20
Ours	**80.56**	**79.45**	**63.93**	**64.91**	**71.99**

V. CONCLUSION AND FUTURE WORK

We presented a deep learning methodology for human part segmentation that uses refinements based on a stack of up-convolutional layers. It yielded semantically accurate results and well-localized segmentation maps. We identified augmentation strategies that substantially increased performance of

the network. We also demonstrated that adding feature map dropout to each refinement step boosts the overall system performance. In addition, we presented results on the PASCAL Parts, Microsoft COCO datasets and on two new robotics segmentation datasets: Freiburg Sitting People and Freiburg People in Disaster. Our approach advances the state of the art on all the above datasets. To the best of our knowledge our approach also is the first to tackle human part segmentation at this level of granularity (14 parts) with a single RGB camera.

Future work will include investigating the potential of our architecture for human keypoint prediction. A method that can accurately find body joints will have direct applications in human pose estimation and activity recognition. There are also many aspects of the method that we intend to refine, such as having multiple filters per class in the coarse refinement modules of the network. We also intend to work on simplifying the architecture for real-time part segmentation on smaller hardware. Another future line of research will be performing human part segmentation in videos while exploiting the temporal context.

REFERENCES

[1] Xavier Boix, Josep M. Gonfaus, Joost van de Weijer, Andrew D. Bagdanov, Joan Serrat, and Jordi Gonzalez. Harmony potentials. *IJCV*, pages 83–102, 2012.

[2] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *ICLR*, 2015.

[3] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*, 2014.

[4] Alexey Dosovitskiy, Jost Tobias Springenberg, and Thomas Brox. Learning to generate chairs with convolutional neural networks. In *CVPR*, 2015.

[5] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin A. Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *NIPS*, pages 766–774, 2014.

[6] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1915–1929, 2013.

[7] Ross Girshick. Fast R-CNN. *arXiv preprint arXiv:1504.08083*, 2015.

[8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[9] Saurabh Gupta, Ross B. Girshick, Pablo Andrés Arbeláez, and Jitendra Malik. Learning rich features from RGB-D images for object detection and segmentation. In *ECCV*, pages 345–360, 2014.

[10] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *ECCV*, 2014.

[11] Bharath Hariharan, Pablo Andrés Arbeláez, Ross B. Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015.

[12] Ajrun Jain, Jonathan Tompson, Mykhaylo Andriluka, Graham W. Taylor, and Christoph Bregler. Learning human pose estimation features with convolutional networks. In *ICLR*, 2014.

[13] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv:1408.5093*, 2014.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[15] Ming-Yu Liu, Shuoxin Lin, Srikumar Ramalingam, and Oncel Tuzel. Layered interpretation of street view images. In *Robotics: Science and Systems*, 2015.

[16] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CVPR*, nov 2015.

[17] Wenhao Lu, Xiaochen Lian, and Alan Yuille. Parsing semantic parts of cars using graphical models and segment appearance consistency. In *BMVC*, 2014.

[18] Aurelien Lucchi, Yunpeng Li, Xavier Boix, Kevin Smith, and Pascal Fua. Are spatial and global constraints really necessary for segmentation? In *ICCV*, pages 9–16, 2011.

[19] Michael Maire, Stella X. Yu, and Pietro Perona. Object detection and segmentation from joint embedding of parts and pixels. In *ICCV*, 2011.

[20] Nils Plath, Marc Toussaint, and Shinichi Nakajima. Multi-class image segmentation using conditional random fields and global classification. In *ICML*, pages 817–824, New York, NY, USA, 2009. ACM.

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[22] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from a single depth image. In *CVPR*, 2011.

[23] Marcel Simon, Erik Rodner, and Joachim Denzler. Part detector discovery in deep convolutional neural networks. In *ACCV*, volume 2, pages 162–177, 2014.

[24] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015.

[25] Niko Suenderhauf, Sareh Shirazi, Adam Jacobson, Feras Dayoub, Edward Pepperell, Ben Upcroft, and Michael Milford. Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free. In *Robotics: Science and Systems*, 2015.

[26] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. *CVPR*, 2015.

[27] Stavros Tsogkas, Iasonas Kokkinos, George Papandreou, and Andrea Vedaldi. Semantic part segmentation with deep learning. *CoRR*, abs/1505.02438, 2015.

[28] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. Part-based R-CNNs for fine-grained category detection. In *ECCV*, 2014.

[29] Ning Zhang, Manohar Paluri, Marc'Aurelio Ranzato, Trevor Darrell, and Lubomir D. Bourdev. PANDA: pose aligned networks for deep attribute modeling. In *CVPR*, pages 1637–1644, 2014.