Fourier Features For Person Detection in Depth Data

Viktor Seib(✉), Guido Schmidt, Michael Kusenbach, and Dietrich Paulus

Active Vision Group (AGAS), University of Koblenz-Landau,
Universitätsstr. 1, 56070 Koblenz, Germany
{vseib,guidoschmidt,mkusenbach,paulus}@uni-koblenz.de
http://agas.uni-koblenz.de

Abstract. A robust and reliable person detection is crucial for many applications. In the domain of service robots that we focus on, knowing the location of a person is an essential requirement for any meaningful human-robot interaction. In this work we present a people detection algorithm exploiting RGB-D data from Kinect-like cameras. Two features are obtained from the data representing the geometrical properties of a person. These features are transformed into the frequency domain using Discrete Fourier Transform (DFT) and used to train a Support Vector Machine (SVM) for classification. Additionally, we present a hand detection algorithm based on the extracted silhouette of a person. We evaluate the proposed method on real world data from the Cornell Activity Dataset and on a dataset created in our laboratory.

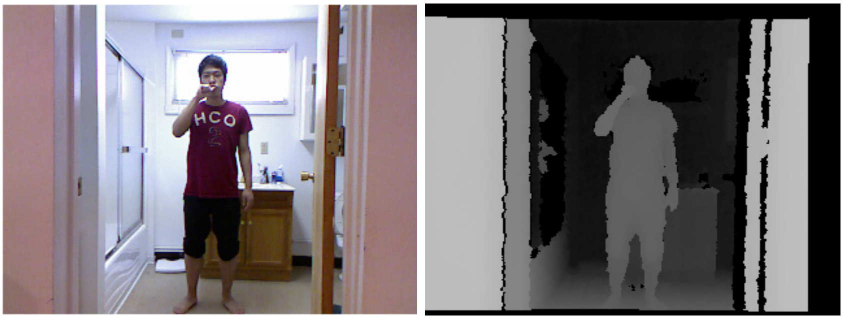
Keywords: People detection · Silhouette detection · Hand detection · Fourier features · Service robots

1 Introduction

The ability of service robots to properly react to the commands given by a user highly depends on the robot's capability to reliable detect the position of the interacting person. Apart from the position of the person itself, also the position of its hands is of large interest for a natural interaction. This comes from the customary practice of people to use gestures for interaction. Important gestures that a service robot needs to be aware of in its daily routine is pointing to a position or object of interest, raising a hand to call for attention or waving the hand to call the robot.

With the availability of affordable RGB-D cameras the problem of person detection can be addressed by the combination of RGB camera data, as well as geometrical information based on depth data (figure 1). We propose an approach that exploits these information by calculating 2 novel features. In the first step, a model-based search is performed on the input data to find possible person candidates. For each candidate, 2 features are computed: the *Frontal Feature* and the *Width Feature*. To obtain a low dimensional feature vector, the computed features are transformed to the frequency domain using Discrete Fourier Transform

G. Azzopardi and N. Petkov (Eds.): CAIP 2015, Part I, LNCS 9256, pp. 824–836, 2015.
DOI: 10.1007/978-3-319-23192-1_69



(a) (b)

Fig. 1. Example RGB image (a) and the corresponding depth image (b) taken from the Cornell Activity Dataset [14].

(DFT) and a Support Vector Machine (SVM) is trained on the obtained Fourier coefficients. The presented approach enables us to detect standing and sitting persons in different poses and orientations towards the camera. Additionally, for frontal facing persons hands are detected using skin color extraction from the face and distances between possible hand candidates on the previously detected person's silhouette. The presented approach is successfully applied on the service robot *Lisa* in our lab.

In the following section 2 we briefly review related work on people detection and especially people detection on depth data. The section 3 and section 4 describe our proposed approach in detail and introduce the proposed features. Finally, section 5 presents evaluation results obtained from the Cornell Activity Dataset [14] and a dataset acquired in our lab and discusses the obtained results. This paper is concluded by section 6 where a summary and an outlook to future work is presented.

2 Related Work

In the past years, several approaches for person detection and tracking using depth data were proposed. However, the actual devices that data stems from are changing. A few years ago, laser sensors or costly stereo camera systems were used. For instance, Bertozzi et al. [1] use a combination of an RGB stereo setup combined with infrared stereo cameras. They extract histograms of oriented gradients (HOG) for classification. The HOG feature is not only used for RGB data, as for example by Dalal and Triggs [5]. It was also adapted to depth data, resulting in the histograms of oriented depths (HOD) detector introduced by Spinello and Arras [13]. Also combinations of HOG and HOD are used as for instance demonstrated in [7].

Machine learning approaches are also exploited for shape matching e.g. the approach described by Xia et al. [11]. Xia et al. use chamfer distance matching to find candidate regions for people locations and examine these with a three

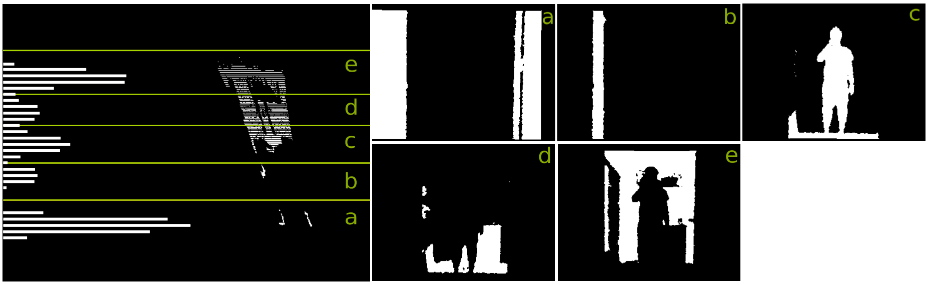


Fig. 2. The resulting depth histogram and the corresponding extracted depth slices are represented as binary images.

dimensional head model. Other approaches use distance transformations and extract a joint representation of detected people, as Lee et al. show in [10].

Paisitkriangkrai et al. [12] introduce spatial pooling on a set of covariance descriptors for feature extraction. Choi et al. [4] introduce a Monte Carlo based Particle Filter. The applied descriptor is based on HOG and on a shape vector is derived from the upper body part of a person candidate. This approach is capable of handling dynamic scenes.

Similar to the approach of Choi et al., our algorithm is also based on shape extraction of the shoulder region and the head. However, we use 2 features specifically designed to detect silhouettes from depth data. As proposed by Hordern and Kirchner [8], we also transform the features with DFT before using them for SVM training and classification.

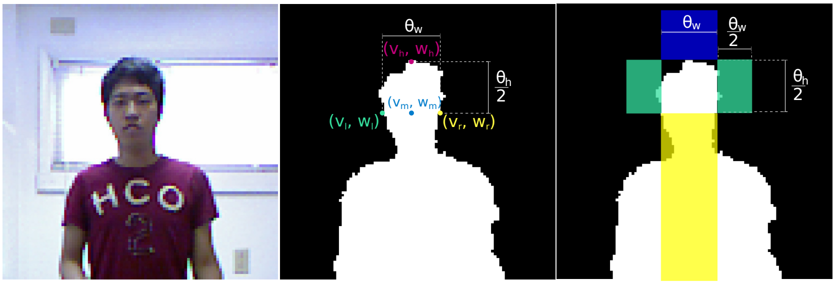
3 Person Detection

In this section we describe our algorithm for silhouette detection. In the following, we employ a right-handed coordinate system, where the x-axis points forward, the y-axis to the left and the z-axis to the top.

3.1 Candidate Detection

To find candidates for person detection, we first compute a point histogram along the depth axis (x-axis) and then perform a model based search. The input point cloud is divided into equally thick slices along the depth axis. In our experiments we use a thickness of $a = 0.1m$ for each slice. Each slice corresponds to a histogram bin, holding the number of points in that slice. The idea behind this histogram is that potential objects form local point clusters in the scene. To find these objects local minima inside the histogram are found. The objects are expected to be located between these extracted local minima.

In the following step, the point cloud is resliced at locations corresponding to minima in the histogram as shown in figure 2. For every depth slice, a binary image I is generated by only considering points located inside the corresponding



(a) (b) (c)

Fig. 3. The RGB data of a person's silhouette (a) and the corresponding binary image (b and c) is shown. The head model assumptions are illustrated in (b), whereas the occupancy of test regions is shown in (c).

slice. A model-based search is performed on the set of binary images $I_0 \dots I_n$ to find person candidates.

First, local maxima $I_i(v_h, w_h)$ on the vertical axis in each binary images I_i are extracted. These are possible head positions, which need to be validated against our model assumptions. Since the binary images were created from point clouds, a 3D point is known for every valid (i.e. "white") pixel $I_i(v, w)$. This permits us to define model parameters in metric space rather than in pixels. In the following, the function $\mathcal{P}(I(v, w))$ is used to obtain the corresponding 3D point $p = (x, y, z)^T$ from a pixel $I(v, w)$ in the binary image. Further, the inverse function $\mathcal{Q}(p)$ is used to obtain the pixel in the binary image corresponding to a 3D point p in the organized point cloud. The actual mapping of a color pixel to a depth pixel (and vice-versa) depends on the dataset (see section 5).

Our model assumption consists of a typical head width θ_w and height θ_h (given in a range of minimal and maximal valid sizes) and includes the approximated calculation of the possible head dimensions, as shown in figure 3. For a local maximum $I_i(v_h, w_h)$ to be selected as candidate, the following propositions must hold. First, there has to be a valid pixel $I(v_m, w_m)$ (middle point), defined as

$$\boldsymbol{I(v_m, w_m) = \mathcal{Q}(p_{v_m,w_m}) = \mathcal{Q}(p_{v_h,w_h} - (0, 0, \frac{\theta_h}{2})^T).} \tag{1}$$

in the binary image, corresponding to a point $\frac{\theta_h}{2}$ below the maximum. Equation 1 requires that at least the upper half of the head must be visible.

Two other points, the left and the right border point of the head, $I(v_l, w_l)$ and $I(v_r, w_r)$, have to be set in the binary image at the same height as the middle point $I(v_m, w_m)$, fulfilling the constraint

$$\theta_{h,min} < ||\mathcal{P}(I(v_l, w_l)) - \mathcal{P}(I(v_r, w_r))|| < \theta_{h,max}. \tag{2}$$

The found value $\theta_w = ||\mathcal{P}(I(v_l, w_l)) - \mathcal{P}(I(v_r, w_r))||$ is the hypothetical head width. As a result of processing every binary image according to this model-based search, a set of object hypotheses is obtained possessing human head-like

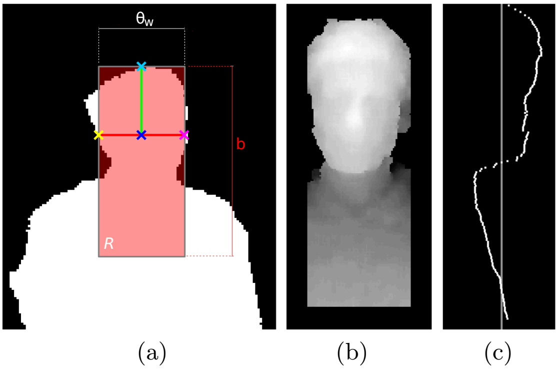


Fig. 4. The image section R used to extract the frontal feature is shown in (a). The contained part of the silhouette is shown in (b). Image (c) shows the frontal silhouette from the side. The horizontal line in (c) is the mean distance value in region R.

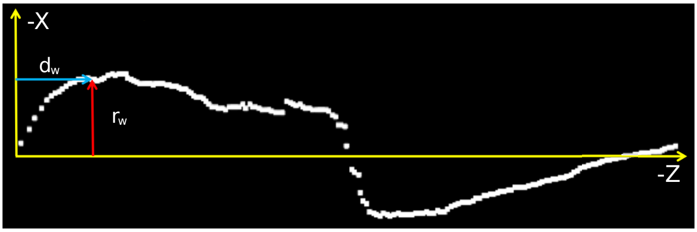


Fig. 5. The extracted values d_w and r_w for the frontal features are shown. This figure corresponds to figure 4 (c) rotated by 90° to the left. The axes are labeled according to the same coordinate system used for feature extraction. The resulting curve is the input data for the DFT.

characteristics at their highest position. Note that this model-based approximation of a head is computationally faster than fitting a circle around local maxima of the binary images.

This set of hypotheses is further refined by calculating the percentage of filled pixels in the binary image in four test regions around the detected hypothetical head (figure 3). The first region represents the area above the highest point of the head with a height of $\frac{\theta_h}{2}$ and the width θ_w. Further, 2 equal regions, one located on the left the other located to the right of the head, are examined. The fourth and last region is located below the head and extends to the bottom of the corresponding binary image. For every region, the percentage of filled pixels in the binary image is calculated. Ideally, the areas above, left and right of the head should be empty. On the other hand, the area below the head should be completely filled, representing a hypothetical torso and legs of a person. Because of possible occlusions by objects in front of the torso, pixels from depth slices in front of the detected candidate are also considered when determining the ratio of filled pixels in the bottom area. After applying these model-based rules, the search space for possible persons is reduced from the whole input point cloud to a set of person candidates.

3.2 Features

For further classification, we propose to extract 2 features for each candidate: the *Frontal Feature* and the *Width Feature*. Both features are well suited to differentiate between persons and objects and are described in the following.

Frontal Feature. The idea of our *Frontal Feature* is to encode the frontal depth contour of the face and upper body of a person. This feature is computed using the previously extracted binary image of the head and the torso and the corresponding point cloud. Therefore, the position, width and height of the head are known. We extract an image section R which is horizontally centered at the head. Vertically, R starts at $I_i(v_h, w_h)$ being the highest point, with the parameter b determining the sections height (figure 4). We obtained best results with $b = 0.5m$ in our experiments. The resulting region covers the candidate's head and upper body. For each pixel row w inside the binary image section R, the mean depth value r_w of all points in this row is computed as

$$r_w = \frac{\sum_{v=0}^{V} \mathcal{P}_x(R(v,w))}{V} \tag{3}$$

where V is the width of R in pixels, $R(v, w)$ is a pixel in R and $\mathcal{P}_x(R(v, w))$ is a function that returns the x-value (i.e. the value on the forward axis) of the point at pixel $R(v, w)$. This provides some stability against rotations and side movements of the head, as well as sensor noise. Further, we calculate the distance d_w of each row w to the highest point $I(v_h, w_h)$ as

$$d_w = \frac{\sum_{v=0}^{V} \mathcal{P}_z(R(v,w))}{V} - \mathcal{P}_z(I(v_h, w_h)) \tag{4}$$

where $\mathcal{P}_z$ is defined similar to $\mathcal{P}_x$, but returning the z-value (i.e. the vertical coordinate). Now we have obtained a mean depth value r_w and a vertical distance d_w to the top point for each row. The resulting frontal contour of a person is visualized in figure 4 for a person facing the camera. The computed values r_w and d_w from R are shown in figure 5. Depending on the person's orientation towards the camera, this feature provides different profiles.

Width Feature. This feature benefits from the fact that persons facing the sensor will have a broader shoulder section compared to their head width. Again we use $b = 0.5m$ for the height of the section of interest R, its width is set to the width of the binary image. We now extract the leftmost $\mathcal{P}(R(v_l, w_l))$ and the rightmost $\mathcal{P}(R(v_r, w_r))$ valid point in each row of the binary image, obtaining the row width b_w from

$$b_w = ||\mathcal{P}_y(I(v_l, w_l)) - \mathcal{P}_y(I(v_r, w_r))||, \tag{5}$$

where $\mathcal{P}_y$ returns the value on the horizontal axis. Further, we again use the vertical distance d_w of each row to the highest point of the head $I(v_h, w_h)$.

3.3 Fourier Feature Vectors

Both features can be seen as functions of the distance d_w from the top point of the candidate head. In case of the *Frontal Feature* function values are the average depth values r_w and in case of the *Width Feature* the width b_w of the candidate silhouette. These two features are computed for each candidate and transformed into the frequency domain using the Fourier transform. Since we have discrete functions (discretized by the lines of the binary images), we use the Discrete Fourier transform (DFT). The actual feature vector is composed of the resulting Fourier coefficients. We can adapt the length of the feature vector by omitting high frequency Fourier coefficients. While the computed features originally had a length of 50, after the Fourier transform we retain only the first 6 coefficients. Figure 7 depicts several sinusoids corresponding to the first Fourier coefficient. Unlike in the work of Hordern et al. [8], where the feature vector is divided by the constant component and therefore is scale invariant, we

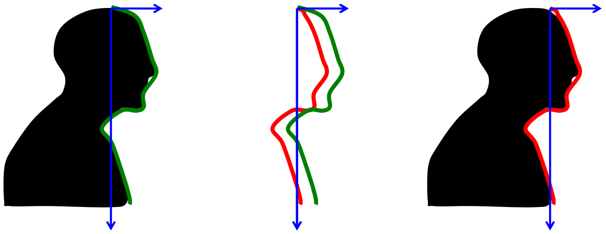
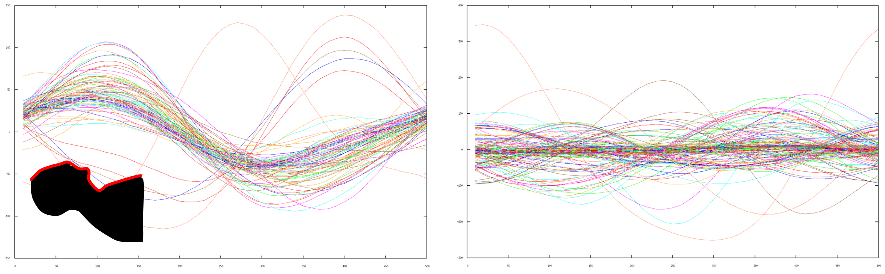


Fig. 6. Omitting the constant component of the Fourier Transform results in a distance independent feature vector sinusoid (shown in red), in contrast to the distance dependent sinusoid with the constant component (shown in green).



(a)

(b)

Fig. 7. The sinusoids corresponding to the first Fourier coefficient of several human silhouettes are depicted in (a). The outliers stem from partially visible silhouettes at image borders. Image (b) shows the first sinusoids of different objects.

omit the constant component from the feature vector without division. In our case this leads to the feature vector being invariant to the distance of the person to the sensor (figure 6). Our features are based on metric data and scale is an important factor in the process of people detection. We therefore do not want to have a scale invariant feature in order not to detect e.g. toys like dolls that have a similar silhouette, but different size as a human. The resulting feature vector is used to train a support vector machine (SVM) with a radial basis function kernel (RBF).

4 Hand Detection

Since we are interested in interacting with people, we focus on detecting hands of upraised and outstretched arms. These are signs of calling for attentions and thus should be recognized by a service robot.

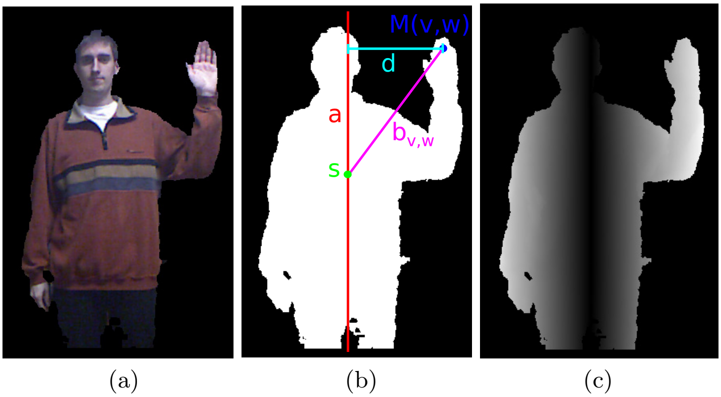


Fig. 8. Illustration of the geometric model to determine the distance between any silhouette point $\mathcal{P}(M(v,w))$ and the centroid s. Image (a) shows the RGB data of the silhouette, while (b) shows the binary image. In image (c) the distance matrix D is depicted.

Starting from the already known highest pixel $I(v_h, w_h)$ of a detected person, a region growing is performed on the binary image. We use the 4-neighborhood for region growing. For 2 neighboring pixels $I(v_a, w_a)$ and $I(v_b, w_b)$, with $I(v_a, w_a)$ being part of the region, $I(v_b, w_b)$ has to fulfill 2 criteria to be added to the region. First, its corresponding 3D point $\mathcal{P}(I(v_b, w_b))$ must not be farther away from the 3D point of a point in the region $\mathcal{P}(I(v_a, w_a))$ than the threshold value c_1:

$$||\mathcal{P}(I(v_a, w_a)) - \mathcal{P}(I(v_b, w_b))|| < c_1. \tag{6}$$

Additionally, it has to be within the range c_2 of the corresponding 3D point of the highest point of the head, $\mathcal{P}(I(v_h, w_h))$:

$$||\mathcal{P}(I(v_h, w_h)) - \mathcal{P}(I(v_b, w_b))|| < c_2. \tag{7}$$

In contrast to the initial binary image I, the resulting binary image M contains only the silhouette of the person (without possible background clutter). In the following, the hand is detected using two cues: by taking into account the distance of the hand from the body and exploiting skin color.

4.1 Distance Cue

To exploit the distance cue, the detected body in the silhouette binary image M is reduced to a single vertical line through the centroid of the silhouette. The centroid is calculated from all 3D points of the silhouette using the points p as

$$p = \begin{cases} \mathcal{P}(M(v,w)) & \text{if } M(v,w) = 1 \\ (0,0,0)^T & \text{else} \end{cases} . \tag{8}$$

The centroid s is then computed as the mean of all points

$$s = \frac{\sum_{v=0}^{V} \sum_{w=0}^{W} \mathcal{P}(M(v,w))}{\sum_{v=0}^{V} \sum_{w=0}^{W} M(v,w)} \tag{9}$$

where V and W represent the image width and image height, respectively. Note that in the denominator of equation 9 the number of the valid pixels in the silhouette binary image M is computed, since all pixels in M are either 1 or 0. We represent the silhouette's centroid line through the point s by a directional vector $\boldsymbol{a} = (0,0,1)^T$ parallel to the vertical axis of the coordinate reference frame. Now we can calculate a directional vector $\boldsymbol{b}_{v,w}$ pointing from an arbitrary point $\mathcal{P}(M(v,w))$ in the silhouette image to the centroid s as

$$\boldsymbol{b}_{v,w} = s - \mathcal{P}(M(v,w)). \tag{10}$$

As shown in figure 8 (b), the distance d of a given point $M(v,w)$ can then be calculated by the cross product

$$d_{vw} = ||\boldsymbol{a} \times \boldsymbol{b}_{v,w}|| \tag{11}$$

and a distance matrix D created according to

$$D(v,w) = \begin{cases} d_{vw} & \text{if } \mathcal{P}(M(v,w)) = 1 \\ 0 & \text{else} \end{cases} . \tag{12}$$

4.2 Skin Color

We use the skin color as the second cue for hand detection. The detected person's face color is used as a reference color. Please note that at this point the person's silhouette is already segmented. The skin color detection is thus not influenced by clutter in the environment.

To extract a representing skin color, we convert the RGB-data of the face region to HSV and then create a color histogram. The HSV color space has

Table 1. Dataset samples from the Cornell Activity Dataset [14] used to evaluate the people detection algorithm

Person 1				**Person 2**			
subset	**action**	**pose**	**# images**	**subset**	**action**	**pose**	**# images**
(A)	brushing teeth	standing	1351	(A)	still	standing	482
(B)	drinking water	standing	1746	(C)	drinking water	standing	508
(C)	wearing contact lenses	standing	440	(F)	opening pill container	standing	210
(E)	working on laptop	sitting	1265	(I)	cooking (stirring)	standing	442
(H)	talking on the phone	standing	1442	(L)	writing on white board	standing	609
Person 3				**Person 4**			
subset	**action**	**pose**	**# images**	**subset**	**action**	**pose**	**# images**
(A)	still	standing	374	(A)	still	standing	276
(D)	still	sitting	474	(F)	opening pill container	standing	215
(H)	opening pill container	standing	245	(G)	opening pill container	standing	346
(M)	walk around	standing	511	(J)	cooking (choping)	standing	390
(Q)	wearing contact lenses	standing	447	(K)	working on computer	sitting	411

been proved to be a reliable color space for skin and hand detection [9], [3], [6]. However, the exact choice of the color space is not critical, since we are not actually *detecting* skin color, but rather comparing the face color with possible hand locations on the segmented silhouette.

The generated histogram will show a clustering of the skin color. To clean the histogram from colors originating from the hair, eyebrows, eyes or lips, only the high-valued classes of the histogram are used. Starting with the highest valued class, further high-valued classes are added, until the number of pixels in the valid classes exceeds a predefined threshold. During detection, each pixel with a color inside the valid histogram classes will be regarded as skin color. The resulting colors and the corresponding pixels are stored in a separate binary image S. Both cues are combined to a hand confidence map M as shown in the following Equation:

$$M_{hand}(v,w) = \alpha \cdot D(v,w) + \beta \cdot \frac{S(v,w)}{max(S)} \tag{13}$$

with the 2 weights α and β and the normalization value $max(S)$ with the maximum histogram value. Note that no separate normalization for the distance value is needed, since it is already in metric space. For a validation of possible hands the maximum inside M_{hand} is computed. Regions in the neighborhood of the first maximum, as well as the maximum itself are excluded and the remaining values are used to compute a second maximum, resulting in both hands of a person.

5 Evaluation and Results

The people detection algorithm was evaluated with sample sets from the Cornell Activity Dataset [14]. The dataset was recorded with the Microsoft Kinect. The provided RGB and depth images are aligned, so no manual alignment is

Table 2. Person detection results with different features applied

Frontal Feature				
	Person 1	Person 2	Person 3	Person 4
Precision	0.679	0.958	0.606	0.887
Recall	0.999	0.904	0.940	0.960

Width Feature				
	Person 1	Person 2	Person 3	Person 4
Precision	0.684	0.887	0.468	0.973
Recall	0.976	0.904	0.996	0.955

Frontal & Width Feature				
	Person 1	Person 2	Person 3	Person 4
Precision	0.684	0.983	0.575	0.973
Recall	0.977	0.860	0.885	0.964

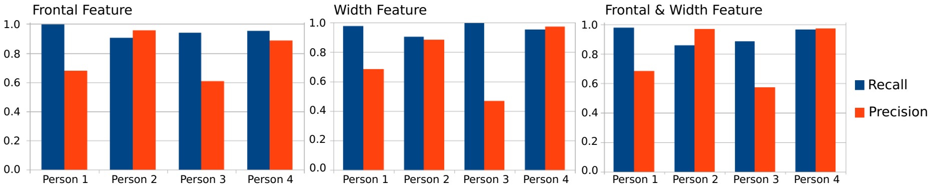


Fig. 9. Precision and recall results on the three different configurations

necessary. We used the CAD-60 subset, which originally consists of 60 RGB-D videos of 4 humans (2 male, 2 female) performing different home activities. For the evaluation, 5 subsets of each person from the dataset were used as indicated in table 1. Since the tested dataset was provided to evaluate action recognition, we had to manually annotate the different images with the correct locations of the persons. This dataset was used to evaluate only the people detection, because the dataset does not provide sufficient hand gestures. We used 60% of the images for training and the remaining 40% for testing. The evaluation was performed 3 times: using only the *Frontal Feature*, only the *Width Feature* and finally using both features. The results are reported in table 2 and figure 9.

As can be seen from table 2 the recall for the *Frontal Features* as well as for the *Width Feature* is in all cases above 90%. However, based on the results it can not be definitely decided which features is best. On the other hand, when both features are combined the recall decreases, which speaks in favor of only using one of the proposed features. In general, the precision is below the recall in all cases. This means that our algorithm rather oversees a person instead of detecting a person where there is none. This behavior is largely due the parameterization of the candidate detection step as described in section 3 and can be changed by adjusting the parameters. Still, certain poses could not be detected by our algorithm. Mainly, persons standing close to high objects or bending over objects have a low recognition rate. Further, problems occurred when the head was occluded.

To evaluate the hand detection, we used our own dataset with 457 test images. The sensor was calibrated using well established techniques [15], [2] to provide accurate RGB and depth image alignment to create the dataset. The hand detection performed with a recall of 73% and a precision of 79%. In this case, main problems arose due to complicated lighting situations, where skin colors differ between hand and face and therefore hands cannot be detected properly.

6 Conclusions and Outlook

In this work we proposed a system for people detection based on RGB-D data. Person candidates are extracted from the input point cloud. Two novel Fourier features are computed on the candidate silhouettes and used for classification. The experiments indicate that using only one of the features at a time leads to better results than using a combination of both features. When only one feature is used, recall is above 90% in all cases, while precision is lower, but still at a high value. The hand detection algorithms achieves a recall and precision above 70%. These promising results support our approach for silhouette and hand detection.

In its current state, our algorithm is challenged by people bending over, since the resulting pose does not support our model of a head being directly above the torso. Our future work will concentrate on further improving the algorithm and adjust our geometric human model. We will extend it to not only check the test areas below the head, but also in a diagonal direction to be able to detect bending people.

References

1. Bertozzi, M., Broggi, A., Del Rose, M., Felisa, M., Rakotomamonjy, A., Suard, F.: A pedestrian detector using histograms of oriented gradients and a support vector machine classifier. In: Intelligent Transportation Systems Conference, ITSC 2007, pp. 143–148. IEEE (2007)
2. Brown, L.G.: A survey of image registration techniques. ACM Computing Surveys (CSUR) **24**(4), 325–376 (1992)
3. Cerlinca, T.L., Pentiuc, S.G., Vatavu, R.D., Cerlinca, M.C.: Hand posture recognition for human-robot interaction. In: Proceedings of the 2007 Workshop on Multimodal Interfaces in Semantic Interaction, pp. 47–50. ACM (2007)
4. Choi, W., Pantofaru, C., Savarese, S.: Detecting and tracking people using an rgb-d camera via multiple detector fusion. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 1076–1083. IEEE (2011)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893 (June 2005)
6. Ghosh, S., Zheng, J., Chen, W., Zhang, J., Cai, Y.: Real-time 3d markerless multiple hand detection and tracking for human computer interaction applications. In: Proceedings of the 9th ACM SIGGRAPH Conference on Virtual-Reality Continuum and its Applications in Industry, pp. 323–330. ACM (2010)

7. González, D.I.R., Hayet, J.-B.: Fast Human Detection in RGB-D Images with Progressive SVM-Classification. In: Klette, R., Rivera, M., Satoh, S. (eds.) PSIVT 2013. LNCS, vol. 8333, pp. 337–348. Springer, Heidelberg (2014)
8. Hordern, D., Kirchner, N.: Robust and efficient people detection with 3-d range data using shape matching. In: Australasian Conference on Robotics and Automation (2010)
9. Kakumanu, P., Makrogiannis, S., Bourbakis, N.: A survey of skin-color modeling and detection methods. Pattern Recognition **40**(3), 1106–1122 (2007)
10. Lee, S.J., Nguyen, D.D., Jeon, J.W.: Design and Implementation of Depth Image Based Real-Time Human Detection. Journal of Semiconductor Technology and Science 14(2), 212–226 (2014)
11. Xia, L., Chen, C.-C., Aggarwal, J.K.: Human Detection Using Depth Information by Kinect. In: International Workshop on Human Activity Understanding from 3D Data in Conjunction with CVPR (HAU3D) (2011)
12. Paisitkriangkrai, S., Shen, C., van den Hengel, A.: Strengthening the Effectiveness of Pedestrian Detection with Spatially Pooled Features. CoRR, abs/1407.0786 (2014)
13. Spinello, L., Arras, K.O.: People detection in RGB-D data. In: IEEE/RSJ Int. Conf. on (2011)
14. Sung, J., Ponce, C., Selman, B., Saxena, A.: Human activity detection from rgbd images. plan, activity, and intent recognition, 64 (2011)
15. Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(11), 1330–1334 (2000)