Pose Invariant People Detection in Point Clouds for Mobile Robots

Akif Hacinecipoglu, Erhan ilhan Konukseven and Ahmet Bugra Koku
Middle East Technical University, Ankara, Turkey
Email: akifhno@gmail.com, {konuk, kbugra}@metu.edu.tr

***Abstract*—To be able to navigate in socially complaint fashion and safely, people detection is a very important ability for robots deployed in our social environments. However, it is a challenging task since humans exhibit various poses in daily life as they bend, sit down, touch or interact with each other. A robust people detector should detect humans also in these arbitrary poses. In addition, mobile robots should be able to carry out detection in a real-time manner because our environment is highly dynamic. In this study we developed a fast head and people detector which can, pose invariantly, detect people. Method depends only on depth information of point clouds taken from RGB-D sensors. As a result, it is robust against sudden light and contrast changes. The algorithm runs relying only on CPU, which makes it applicable to mobile robots with low computational resources.**

***Index Terms*— head detection, robot vision, point clouds**

I. Introduction

Robots are becoming an integral part of our daily life. Social, advertisement, security, search and rescue robots share same environments with people. As a necessity, these robots should act in a natural and social manner in these contexts. Therefore, robots have to distinguish people from other objects to be able to behave accordingly. Due to human nature, people can be present in different poses (standing, bent over, laying, sitting, etc.) and can behave different from each other. Therefore, the developed detector should be as generic as possible. In addition, since people detection systems evolve from static systems to mobile robots, a good people detection method should be computationally effective and run in real-time. For decades, researchers developed different methods to solve this problem. With help of developing computation and sensor technologies, more satisfying results are obtained. However, recent methods have drawbacks in computational load, precision or application conditions, etc. Therefore, we aimed to develop a novel people detection method which can run on a computationally limited mobile robot with high performance, detecting people in arbitrary poses (not necessarily standing upright) and in changing light conditions. Our main contributions in this research are;

- Point cloud slicing method to gather human body parts in arbitrary poses,
- Iteratively using Principal Component Analysis (PCA) to extract pose invariant head region of human body,
- Implementing the algorithm in low resource intensive and real-time manner using depth information only

which result in high precision classifier with relatively low training samples.

This paper is organized as follows; First chapter is the introduction to the topic. Second chapter includes related work in the literature. Developed method is described in third part. Results are given and the study is concluded in last two sections.

II. Related Work

People detection has been an active research area in last two decades. Especially for people detection in outdoor scenes like pedestrian detection, most widely used method depends on Histogram of Oriented Gradients (HOG) in a 2D image [1]. Authors defined a fixed size window for detection. This window is further subdivided into grid of cells and orientations of gradients in each cell are computed and a 1D histogram is constructed. This histogram is used for training a linear Support Vector Machine (SVM). In classification stage, fixed-size windows are used in different scales over the image. The method performs better in upright and non-occluded people poses. However, the method fails to classify in changing light conditions and body postures. Since there is no depth information, small image patches resulting in similar HOG descriptors of human body lead to false positives. To use depth information, stereo camera pair is used in a research [2]. They propose a tracking-by-detection method which provides satisfactory results even in challenging scenes. However, their approach is highly resource intensive such that each frame is processed in about 30 seconds which is far from being real-time to be applicable.

With evolving technology in sensor hardware, incorporation of depth information into people detection methods became affordable. RGB-D sensors like Microsoft Kinect, Asus Xtion or Intel Realsense made depth perception available besides color information. Since these sensors perform best when there is no direct sunlight in the environment due to infrared light interference, they are implemented in indoor people detection applications, especially. Spinello and Arras [3]

Manuscript received July 8, 2019; revised April 14, 2020.

doi: 10.18178/ijmerr.9.5.709-715

used RGB-D data along HOG descriptors. They used a method similar to HOG but they incorporated depth information and developed a method called Histogram of Oriented Depths (HOD). They combined HOG and HOD descriptors to achieve best performance in different distance ranges. However, their method densely scans each frame for people and relies on GPU implementation for a real time performance. They do not provide a test result for articulated human bodies or other arbitrary poses but due to nature of the descriptors they use it can be concluded that their method will perform best for standing upright people with help of GPU implementation which may be a limitation for computational resources of mobile robots. HOD is also used in another study [4]. They segment the depth image into an initial number of regions and then they eliminate regions which are not consistent with some heuristics like width and height. Remaining regions are classified using HOD descriptors with SVMs.

There is a research on people detection for mobile robots with low height [5]. Since their robot has an RGB-D sensor at height level of human knee, they detect people using lower part of human body, namely, legs only. Since legs are not very distinctive feature of human body, they use a large training set, i.e. 26000 instance training dataset acquired in 15 different real world environments. Two new features for people detection are introduced by Liu et al. with RGB-D sensors [6]. Histogram of Height Difference (HOHD) and Joint Histogram of Color and Height (JHCH) are used to classify clusters as human. Before classification they gather human body plausible positions using a height map. Local height maxima are assumed as head crowns. However, this assumption yields false results when there is a person with his hand raised over his head. Therefore, this method may not be applicable for detecting people in arbitrary poses. In another research, top-view depth cameras are used to detect people [7]. To have an occlusion-free view, they position depth cameras overhead and they detect heads using hemi-ellipsoidal head model. They have satisfactory results for a stationary surveillance system positioned on a level above people but it is not applicable to mobile robots. Also it may fail in other poses than upright standing pose due to its dependency of hemi-ellipsoidal head shape.

A study similar to ours uses RGB-D sensor to detect people with satisfactory results in upright human poses [8]. Munaro and Menegatti downsize the point cloud with a voxel grid filter to make it easy to work with and to have constant point density along the depth. When clusters are separated with removal of ground plane, they label remaining clusters using Euclidean distances between points. To avoid over-segmentation problem (clustering parts of a body in different sets), they merge clusters that are close in ground plane coordinates. To solve under-segmentation problem (clustering more than one human into one set), they implement a head detection method using a height map. Local maxima in height map are regarded as heads of people in the scene which is not the case if a part of a body (like hands, arms, etc.) becomes higher than head. They apply a bounding box to a certain region around these detected head positions as candidate people locations. Finally, corresponding regions in RGB image are used for classification of these patches. HOG detector is used with SVM in 2D image to detect people. Again, problems with HOG detector raises here. Although this method detects people in 23 fps with reduced depth resolution of QQVGA (160x120 pixels) instead of VGA (640x480) resolution available in Kinect, it is apparent that it fails in non-upright human postures due to HOG descriptor and height map operation. If a person raises his hand above his head level, assumption of local maxima as head positions will not hold.

Zhang et al. employed similar method and generate depth contours, detect candidate head locations and then use a deep network to train and classify also using RGB information [9]. However, they are also assuming that maxima in depth contours will represent head tops. This assumption fails when one raises a hand or bends etc. Therefore, it is not applicable to arbitrary poses.

In our method we propose a novel approach that is able to detect people pose invariantly. People in direct contact with each other or partly occluded are also detected with our approach in high frame rates without any GPU implementation. This makes this method applicable in mobile robots operating in environments shared with people.

III. The Method

We first preprocess the point cloud with a voxel grid filter followed by a ground plane removal process. Remaining points are then clustered using Euclidean distance metric similar to [8]. Clusters are sliced to overcome segmentation problems and head sections are extracted. Extracting head section of human bodies in different postures in cluttered and dynamic environments is important since head provides 1-to-1 matching for a human and it is the most possible non-occluded part of human body. Finally, candidate head sections are classified.

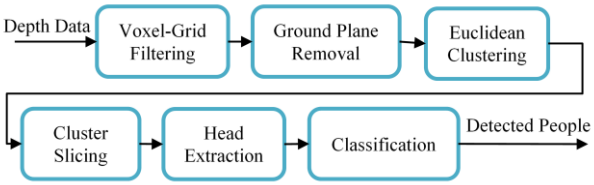

Figure 1. Detection pipeline of the proposed method.

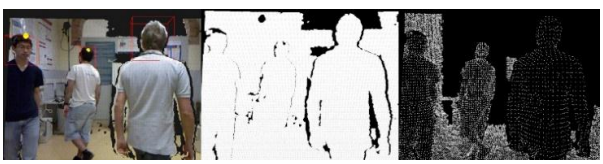

Figure 2. Downsampling a point cloud with voxel grid filtering.

Steps of the proposed method can be listed as; voxel-grid filtering, ground plane removal, Euclidean clustering, cluster slicing, head extraction and classification (Fig. 1 and Fig. 5). These steps are given in detail in upcoming

sections. The method is implemented with Robot Operating System (ROS) [10] and Point Cloud Library (PCL) [11]. This allowed us to develop the algorithm in a modular way which makes it easy to integrate into open source PCL codebase.

A. Voxel Grid Filtering

Point cloud data which is gathered from RGB-D sensor may have high resolution and density of points decreases with the increasing distance from the sensor. Voxel grid filter creates 3D voxels over the point cloud. Points in each voxel (i.e., a cube with a fixed size) are represented with their centroid. This reduces (downsamples) point cloud to a reasonable number of points which makes processing easier without losing important features (Fig. 2). Our default voxel size (i.e., edge length of cubic voxel) is 0.03 m. Since we try to extract head which has relatively low volume, this value provides us detailed point cloud for further processing and classification while keeping computational loads low. Another advantage of voxel grid filter is having nearly uniform density along the distance from the sensor. This enables treating number of points as object size.

B. Ground Plane Removal

Ground plane detection is important step for segmentation and processing performance. Detecting of initial ground plane relies on the assumption that the lowest three points on the point cloud belong to ground plane. The algorithm estimates and removes this plane from the point cloud provided by the voxel grid filter. We compute the plane coefficients with a RANSAC-based least square method and we remove all the inliers within a threshold (0.1 m) distance, due to sensor noise and shoes. Removal of ground plane helps us on clustering since people and objects on the ground will become unconnected. We should note that, our detection method does not rely on removal of ground plane. This process makes processing step faster since considerable amount of points belonging to ground plane are removed from the scene and it becomes faster to cluster different objects.

C. Euclidean Clustering

After removing the ground plane from the scene, we are left with people (if any), objects and walls. To narrow down the region of interest and gather point cloud for possible people locations, we cluster remaining points as much as possible to represent different objects. For this purpose, we use Euclidean distance metric.

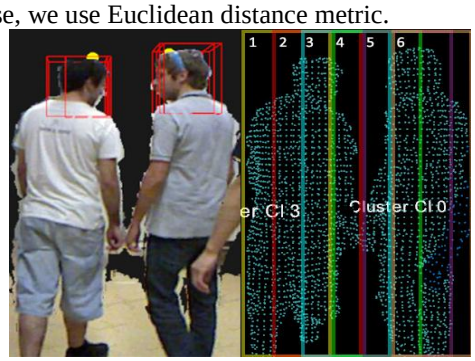

Figure 3. Two people are merged into one cluster due to contact.

We generate a search tree for efficient loop through every point in point cloud. Then we get a seed point and compare it with its neighbors. If Euclidean distance between these two points is below a certain threshold (0.06 m, twice the voxel size), we assume that these two points belong to same cluster. It may not be possible to cluster different objects to different clusters effectively especially if they are very close to each other, in contact or combined. Therefore, to be able to detect every people in the scene, we extract head sections of each person.

D. Cluster Slicing

After clustering, we may have multiple clusters for a single body (due to occlusions) or we may have multiple bodies in one cluster (due to physical contacts between multiple bodies). To solve these problems, several segmentation methods are proposed in the literature. Region growing segmentation is one of these methods [12] which merge regions which have similar smoothness properties. This method fails especially when viewing angle on RGB-D sensor results in missing points between body parts, or if there is a smooth transition of contacts between different humans and objects. Color based region growing [13] is similar to previous one but this method merges neighboring points if colors of them are similar. This method becomes hard to implement in people detection since there are different colored segments in human body (due to skin, hair and different colored clothes).

Since it is not possible to fully distinguish objects into different clusters, we process clusters in different manner. Instead of having the whole body cluster we concluded that we only need head and shoulder parts of a human body to be able to classify it as a person. Therefore, instead of distinguishing objects exactly at clustering step, we extract only possible head sections. We use a method of slicing point cloud vertically with consecutive zones intersecting in a pattern, i.e., next slice starts at a fixed distance before the end of the previous one (like a sliding window). With this approach it is guaranteed that at least one of the slices will contain full head section of a person even in nonstandard poses. For defining a single slice width, anthropometric data for human is used [14]. Mean shoulder and elbow-to-elbow breadths for adults are given as approximately 0.45 to 0.50 m. To regard additional width from cloths and to make sure that shoulder and full head portion of a body is included in a slice, we select slice width as 0.6 m and intersection width between adjacent slices as the 1/3 of the slice width, i.e., 0.2 m. To eliminate the possibility to slice the point cloud with head section to be divided into two sub-clusters, we use consecutive slicing with intersection. Slicing is carried out vertically, starting from left-most point towards right. An example slicing can be viewed in Fig. 3 where it is obvious that slices number 2 (red) and 5 (purple) contain full head portions.

This slicing implementation allows us to extract necessary head sections even there are under-segmented clusters (combined bodies), clusters of bending people, etc. This method also eliminates the main assumption of

many recent people detection methods like [8, 9], i.e., highest part of the body should be the head.

E. Head Extraction

When we have sub clusters which are candidates for possible human bodies, we need to extract head and shoulders section to be able to classify it with machine learning techniques in next step. For this purpose, we exploited principal components of 3-D point clouds. The main assumption is that, the eigenvector corresponding to the largest eigenvalue of point cloud of a human body always points from lower part of human body towards the head. Or in other words, largest variance should be along height of a person. Centroid of the point cloud $\vec{\bar{p}}$ is calculated first (Eqn. 1) where N is number of points and $\vec{p}_i$ is iterator for all points in the slice.

$$\vec{\bar{p}} = \frac{1}{N}\sum_{i=1}^{N}\vec{p}_i \,. \tag{1}$$

Covariance matrix C of the point cloud is obtained with (Eqn. 2) for further PCA application. In (Eqn. 3), $\vec{v}_j$ and $\vec{\lambda}_j$ represent eigenvectors and eigenvalues respectively where $j=1$ for the smallest eigenvalue and $j=3$ for the largest one.

$$C = \frac{1}{N}\sum_{i=1}^{N}\left(\vec{p}_i - \vec{\bar{p}}\right)\left(\vec{p}_i - \vec{\bar{p}}\right)^T \tag{2}$$

$$C\vec{v}_j = \vec{\lambda}_j\vec{v}_j \,,\ \ j \in \{1,2,3\} \tag{3}$$

To be able to detect people in various poses, in our approach we do not have the assumption of having people standing upright. Therefore, we need to align clusters while cropping them to head section. This novel approach enables us to have head sections of people even if they are bending or leaning in any direction by extracting head section from rest of the cluster with a fixed bounding box.

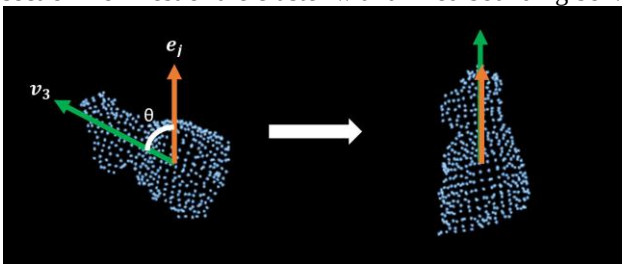

Figure 4. Alignment of the eigenvector with the vertical axis.

After having eigenvalues and corresponding eigenvectors of the cluster, we align the eigenvector, $\vec{v}_3$, corresponding to the largest eigenvalue with the vertical axis of the optical frame, i.e., we rotate the cluster around its centroid to align the direction of maximum variance with the vertical axis (Fig. 4). Rotation axis is $\vec{r}$ and rotation angle is θ (Eqn. 4-5). Here $\vec{e}_j$ is the unit vector along vertical (Fig. 4) and $\|\vec{v}_3\| = 1$.

$$\vec{r} = \vec{v}_3 \times \vec{e}_j \,. \tag{4}$$

$$\theta = \cos^{-1}\left(\vec{v}_3 \cdot \vec{e}_j\right). \tag{5}$$

After the rotation, first process aims to gather upper body part of the person in the slice. From [14], we know that average shoulder breadth is about 0.5 m and upper body height (i.e., stature-crotch height) is about 0.9 m. Therefore, we crop the slice with these dimensions consecutively. Cropping height is referenced from the centroid. At this point if the target slice contains a human, we assume that we have the upper body in cropped cluster. After every cropping operation, we need to apply PCA again and rotate the cropped cluster again to keep the head at the top part in upright position. Since we aim to extract head, we need to crop further to average head width which is 0.25 m from [14]. When we crop the cluster to width of 0.25 m, we also eliminate shoulders, arms and other parts of body in width. Cropped cluster is rotated again using PCA.

Last two cropping operations are in height only. We crop the height consecutively in two steps. In the former one, we crop the height starting from centroid to 0.45 m to keep the upper body aspect ratio constant. In the latter one, we crop the cluster to 0.35 m in height referenced from the top most point. As mentioned early, after each cropping operation we apply PCA and align the cluster. After these steps, resulting cluster has a size 0.25 x 0.35 meters in width and height axis respectively. These fixed dimensions for a possible head portion provides a fixed size test sample for classification process.

F. Classification

At the last step we classify candidate head sections. For 3D point cloud data of head section, we use Viewpoint Feature Histogram (VFH) descriptor [15]. VFH is an extended version of Fast Point Feature Histogram (FPFH) which is applicable to entire object cluster, faster than original FPFH [16] and incorporates viewpoint component using surface normals of the object.

Figure 5. Left to right: (a) Voxel-grid filtering, (b) Ground plane removal and clustering, (c) Head extraction, (d) Classification.

These properties of VFH makes it suitable for our application which is planned to require low computational resources while describing the 3D shape of the cluster satisfactorily. Support Vector Machines (SVM) is used as the classification method for this problem. Implementation of the method is carried out using LIBSVM [17]. To train the SVM, we extracted training samples from the processed clouds. In total 1891 training samples are gathered from an indoor test environment and they are labelled manually as positive (human) or negative. Training set contains 1121 negative and 770 positive samples. SVM is trained with these samples using Radial Basis Function (RBF) kernel and probability estimation option. Probability estimation feature of LIBSVM fits sigmoid function to outputs and maps them to probability scale [18]. Finally, classification model with probability estimation is obtained.

The resultant clusters from head extraction stage are classified with this SVM classifier. If the same head portion is present in two different slices of clusters due to intersecting slices, they are both classified as people. However, in such a case we merge two results into one if bounding boxes around two clusters which are classified as people overlap more than 50% in projected area to ground plane [19].

IV. Experiments

The proposed method detects people in indoor environment using depth data from RGB-D sensors. Since we introduce a novel method to detect people in various poses like bend, lean, sit, run, partly occluded, hands up etc., we have collected samples of different people wandering around in arbitrary poses using Microsoft Kinect v2 RGB-D sensor. Sensor is kept stationary in a cluttered indoor scene and frames are recorded during 57 seconds (Fig. 7). We used a part of the samples gathered from this setup as training samples for SVM. We evaluated detection ability in challenging scenes in these recordings. Additionally, to evaluate the classification and generalization performance of our method, we used state of art Kinect Tracking Precision (KTP) dataset which is distributed with ground truths labeled [8]. Since their method relies on the assumption that highest part of the body is head, KTP dataset does not include people with non-standard poses like bending, sitting, hands up etc.

We regarded a result as true positive if ground truth position of head and the detected bounding box are not apart more than 0.15 m i.e., nearly half width of a human head from [14] since there can be only one single head at this proximity. Experiments are conducted on a laptop computer with Intel i7 4700HQ 2.4 GHz processor and 8 GB RAM. Since we do not have a GPU implementation, only resource that we use is CPU.

A. Results

In KTP dataset Microsoft Kinect at 640x480 pixel resolution and at 30Hz frame rate is mounted on a mobile robot. They recorded four different cases for mobile robot movement; Still, Translation, Rotation and Arc. In each of these cases, people are moving in predefined numbers and paths. One person moves back and forth, three persons walk with random trajectories, two persons walk side by side in a linear path, one person runs and finally five persons gather in a group and walk away. Besides occlusions, there is no challenging arbitrary pose in this and other similar datasets like Freiburg RGB-D People Dataset [3]. However, it is still a good evaluation set for detection and generalization performance since it is recorded with older Kinect device with various people movements in it.

Results are evaluated in terms of precision, recall and frame rate. With 0.03 m voxel size, we achieved an average of 28 frames per second rate for KTP dataset and 22 for our own recordings. Difference is due to high resolution of Kinect v2 we used in contrast to Kinect v1 in KTP. Especially, voxel grid filtering and Euclidean clustering steps take longer in high resolution. Results are satisfactory for a mobile robot moving in a dynamic cluttered environment as one cycle approximately takes about 35 ms to complete. From Fig. 6 and Table 1, it can be seen that we have high precision even with high recall rate.

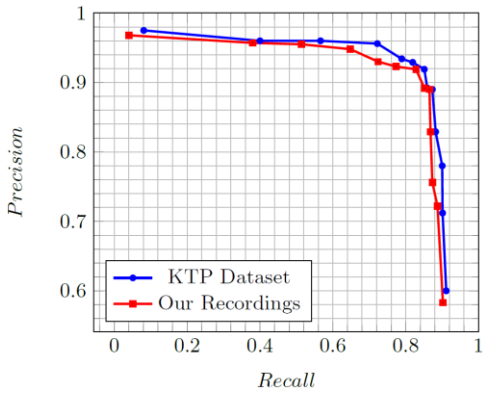

Figure 6. Precision and recall curves for our detection method.

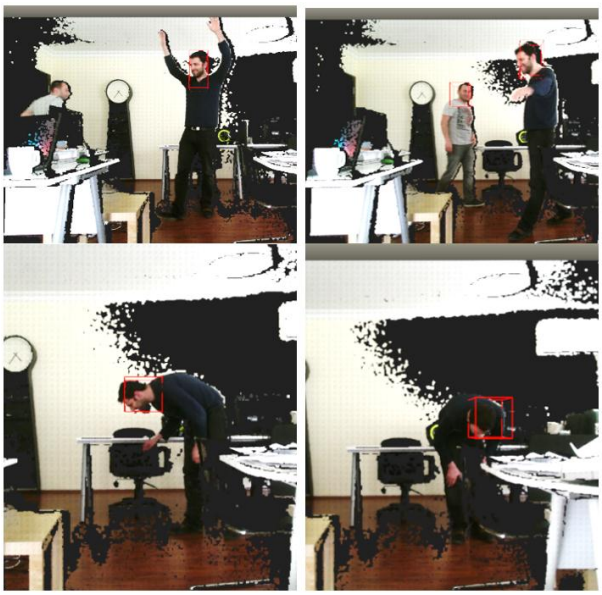

Figure 7. True positives with arms open in various directions (top), and while leaning in different orientations (bottom).

Equal Error Rate (EER), which is the point where precision equals to recall is 88% for KTP. In a similar method which relies only on depth information, authors achieved EER around 86% with their own dataset [3]. Additionally, results for our own arbitrary pose set (Fig. 7 and Table 1) show that our method detects people with high rate and accuracy even with arbitrary poses.

TABLE I. TEST RESULTS FOR DETECTION RATE AND EQUAL ERROR RATE

Dataset	Fps (Hz)	EER (%)
KTP Dataset [8]	28	87.6
Own Recordings	22	86.5

V. CONCLUSIONS

We introduced a novel method for detecting people in various poses using only depth data gathered using RGB-D sensors like Microsoft Kinect. With the cloud processing approach which we propose, effect of clustering/segmentation step is decreased on the results. Since head section of a human body is the most representative part for classification, we extract this part even in challenging scenes using the proposed method. We evaluated our method on a public dataset and on a collected dataset for various unconventional poses with high performance.

Method provides high detection and frame rate relying solely on CPU. This enables the proposed method to be implemented on mobile robots without acquiring high computational resources. Since the algorithm uses only depth information, light and contrast changes in indoor environment does not affect classification results directly. As future work, we may change body size parameters dynamically depending on the audience (adult or children) or the environment. Also we plan to release the implementation into PCL and ROS as a library which will make its further implementation easy. Since the literature lacks a dataset containing people in various non-standard poses gathered with RGBD sensors, we plan to prepare such a dataset for further benchmarking.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Akif Hacinecipoglu, Erhan Ilhan Konukseven and Ahmet Bugra Koku developed the method. Akif Hacinecipoglu designed and carried out experiments. All authors analyzed the data and wrote the paper. All authors had approved the final version.

REFERENCES

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. of CVPR '05: the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, - vol. 1, pp. 886-893, 2005.

[2] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "A mobile vision system for robust multi-person tracking," *IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 2008., pp. 1-8, 2008.

[3] L. Spinello and K. O. Arras, "People detection in RGB-D data," in *IEEE International Conference on Intelligent Robots and Systems*, pp. 3838-3843, 2011.

[4] B. Choi, C. Mericli, J. Biswas, and M. Veloso, "Fast human detection for indoor mobile robots using depth images," in *Proc. 2013 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1108-1113, 2013.

[5] A. P. Gritti, T. Oscar, J. Guzzi, G. A. Di Caro, C. Vincenzo, L. M. Gambardella, and A. Giusti, "Kinect-based people detection and tracking from small-footprint ground robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4096-4103, 2014.

[6] J. Liu, Y. Liu, G. Zhang, P. Zhu, and Y. Q. Chen, "Detecting and tracking people in real time with RGB-D camera," *Pattern Recognition Letters*, vol. 53, pp. 16-23, 2014.

[7] T. E. Tseng, A. S. Liu, P. H. Hsiao, C. M. Huang, and L. C. Fu, "Real-time people detection and tracking for indoor surveillance using multiple top-view depth cameras," in *Proc. IEEE International Conference on Intelligent Robots and Systems*, *Number IROS*, pp. 4077-4082, 2014.

[8] M. Munaro and E. Menegatti, "Fast RGB-D people tracking for service robots," *Autonomous Robots*, vol. 37, no. 3, pp. 227-242, 2014.

[9] G. Zhang, J. Liu, H. Li, Y. Q. Chen, and L. S. Davis, "Joint human detection and head pose estimation via multistream networks for rgb-d videos," *IEEE Signal Processing Letters*, vol. 24, pp. 1666-1670, 2017.

[10] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Mg, "ROS: an open-source Robot Operating System," in *IEEE International Conference on Robotics and Automation, Open-Source Software Workshop*, 2009.

[11] R. B. Rusu and S. Cousins, "3D is here: point cloud library," in *IEEE International Conference on Robotics and Automation*, 2011.

[12] T. Rabbani, F. A. V. D. Heuvel, and G. Vosselman, "Segmentation of point clouds using smoothness constraint," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences - Commission V Symposium 'Image Engineering and Vision Metrology'*, vol. 36, no. 5, pp. 248-253, 2006.

[13] Q. Zhan, L. Yubin, and Y. Xiao, "Color-based segmentation of point clouds," *Laser Scanning, IAPRS, XXXVIII*, pp. 248-252, 2009.

[14] M. H. Al-Haboubi, "Anthropometry for a mix of different populations," *Applied Ergonomics*, vol. 23, no. 3, pp.191-196, 1992.

[15] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3D recognition and pose using the viewpoint feature histogram," In *IEEE International Conference on Intelligent Robots and Systems*, pp. 2155-2162, 2010.

[16] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D registration," In *IEEE International Conference on Robotics and Automation*, pp. 3212-3217, 2009.

[17] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, pp. 1-39, 2013.

[18] J. C. Platt, "Probabilities for SV Machines," In A. J. Smola, P. J. Bartlett, B. Scholkopf, and D. Schuurmans, editors, Advances in Large-Margin Classifiers, pages 61-74. MIT Press, 2000.

[19] M. Everingham, L. V. Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303-338, 2010.

Akif Hacinecipoglu got his B.S. degree from Mechanical Engineering Department of Middle East Technical University at 2009, M.Sc. in Robotics again at Middle East Technical University at 2012. He received his Ph.D. degree in 2019 with research interests of autonomous mobile robots, robot vision and path planning.

Erhan ilhan Konukseven is a Professor at the Department of Mechanical Engineering, Middle East Technical University. He obtained Ph.D. degree in Mechanical Engineering from the Middle East Technical University in June 1997. During his Post-Doc studies he has focused on mobile robotics and sensor based motion planning at Mechanical Engineering Department, Carnegie Mellon University (CMU), Pittsburgh PA, USA. His research interests focus on robotics, robotic machining & deburring, virtual reality, haptic devices, sensor based motion planning and mobile robotics.

Ahmet Bugra Koku is an Assoc. Professor at the Department of Mechanical Engineering, Middle East Technical University. He obtained Ph.D. degree from the Vanderbilt University in 2003. His research interests are mechatronics, mobile robots, robot control architectures, robot learning and qualitative navigation.