Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics (IEEE-SMC 2013), Manchester, GB, pp. 3573-3578, IEEE Computer Society CPS 2013

Fallen Person Detection for Mobile Robots using 3D Depth Data

Michael Volkhardt, Friederike Schneemann and Horst-Michael Gross
Ilmenau University of Technology
Neuroinformatics and Cognitive Robotics Lab
98684 Ilmenau, Germany
michael.volkhardt@tu-ilmenau.de

Abstract—**Falling down and not managing to get up again is one of the main concerns of elderly people living alone in their home. Robotic assistance for the elderly promises to have a great potential of detecting these critical situations and calling for help. This paper presents a feature-based method to detect fallen people on the ground by a mobile robot equipped with a Kinect sensor. Point clouds are segmented, layered and classified to detect fallen people, even under occlusions by parts of their body or furniture. Different features, originally from pedestrian and object detection in depth data, and different classifiers are evaluated. Evaluation was done using data of 12 people lying on the floor. Negative samples were collected from objects similar to persons, two tall dogs, and five real apartments of elderly people. The best feature-classifier combination is selected to built a robust system to detect fallen people.**

Index Terms—**Fallen person detection; Kinect; mobile robot; 3D depth data**

I. Introduction

About one third of elderly people aged over 65 fall at least once a year in their home [1]. Half of the people cannot manage to get up after the fall by own means. As lying on the floor for a long time can cause serious health risks, a reliable method to detect fallen people and to call for help is needed. Fast help after a fall can reduce the risk of death by 80% and the need for a hospital visit by 26% [2]. Hence, it is not surprising that fall detection is among the top requests to assistant technology in surveys among the elderly [3]. Current commercially available products provide only limited solutions, since they either must be worn by the user or require changes to the home of the user. Therefore, mobile companion robots as those developed within the CompanionAble project have a great potential in this field [4], [5].

This paper presents a method to detect fallen people in the depth data of a Kinect sensor mounted on a mobile robot. The mobility of the robot offers great benefits, like non-invasive plug-and-play solutions, handling different postures and occlusion by different viewpoints of the robot, and user feedback through the human-machine-dialog. Yet, methods on a mobile robot cannot rely on detecting the actual fall, since it might happen outside of the robot's field of view. Therefore, our method detects fallen people lying on the ground. Compared to visual methods, the 3D depth data offers new possibilities to increase the robustness of a detection. Ground plane estimation and object-background segmentation is much easier with given 3D data, and classifiers can focus on similar training samples as objects can be normalized in position and orientation. We evaluated different features originated from pedestrian and object detection in 3D depth data with different classifiers from the Machine Learning field of research. To increase robustness given occlusion, the point cloud is segmented into reasonable objects, which are aligned and vertically layered. We train a classifier on the layers of objects and detect fallen people if a certain number of layers is classified positively. The remainder of this paper is organized as follows: Section II summarizes related work in the research area, and Sec. III presents our method to detect fallen people. Afterwards, Sec. IV gives a description of experimental results, and Sec. V summarizes our contribution and gives an outlook on future work.

II. Related Work

Fall detection is well-covered in current research projects. Yet, recent approaches still suffer from several drawbacks, that prevent a breakthrough of consumer products. Methods that detect the actual fall either use worn sensors, external sensors [6], vision or audio [7]. Established worn sensors use acceleration towards the ground or the deviation of acceleration values from learned motion patterns [8]. However, these sensors are intrusive and can be forgotten to be worn by the elderly person, especially if he or she has cognitive impairments. 2D and 3D visual methods that track the user and detect the fall can be grouped into the analysis of changes in body shape [9], [10], position [11], motion velocity [12], and motion patterns [13] (see [14] for a survey of approaches published till 2009). They often require multiple static camera installations, have problems distinguishing a fall from lying or sitting down on furniture, or require training data of real fall sequences.

Another approach is the detection of fallen people lying on the ground, which can also be applied on a mobile robot. 2D methods [15], [16] use multiple models to deal with different view angles, multiple poses, and perspective shortening. Still, these methods are originated from up-right pose people

This work has received funding from the Federal State of Thuringia and the European Social Fund (OP 2007-2013) under grant agreement N501/2009 to the project SERROGA (project number 2011FGR0107).

detection and suffer from high complexity and low detection accuracy, because they cannot rely on the typical Ω-shape of people's head-shoulder contour or exploit the symmetry of people's body.

To the best of our knowledge, there are no 3D methods for mobile robots known that detect people lying on the ground. Therefore, we investigated methods from 3D people and object detection. Note that we focused on feature-based approaches and did not consider model-based approaches [17] or geodesic extrema [18], [19]. Feature-based approaches for people detection can be divided into histogram-based features [20]–[23] and geometrical and statistical features [24]. We expected that features which capture the surface of objects, like the Histograms of Local Surface Normals (HLNS) [23], are especially well suited since the local surface normals of fallen people should be irregular or cylindrical compared to the relatively regular, straight surfaces of artificial objects in home environments, like tables, walls, and chairs. The geometrical and statistical features of [24] should be well suited to detect fallen people since the statistical and geometric properties of a fallen person should be similar to the ones of up-right pose people. Features for object detection in 3D data are described in [25]–[27]. Again, we expected features which capture the curvature of objects, like the Fast Point Feature Histogram (FPFH) [25], to have the highest potential of detecting fallen people. Since fallen people are often occluded by parts of their body or furniture, partial occlusion is a main issue for the detection of fallen people. Therefore, a layer-based approach [23], [24] is well-suited to increase the robustness of the classification. Interest point detectors, like the SURE or NARF features [27], [28], promise to be another option to reach robustness towards partial occlusion. The SURE feature extracts feature descriptors which are similar to the FPFH at points that should be invariant against orientation and scale.

III. Fallen Person Detection

In this section, we introduce our method to detect fallen people. The approach consists of five phases as shown in Fig. 1: Preprocessing, Segmentation, Layering, Feature Extraction and Classification.

A. Preprocessing

The preprocessing phase uses the range image of a Kinect and converts it into a preprocessed point cloud.

1) Conversion: Using the intrinsic parameters of the Kinect, the range image is first converted into a 3D point cloud.

2) Downsampling: The number of points to be considered in subsequent processing stages is reduced by downsampling the point cloud using a voxelized grid approach with a grid-cell size of $3\,cm \times 3\,cm \times 3\,cm$.

3) Transformation: The conversion of the range image only considers intrinsic camera parameters. The transformation phase considers the mounting position of the camera and transforms the point cloud according to the extrinsic camera parameters. Especially the pitch angle of the Kinect is of particular importance, as the later ground plane estimation assumes the ground plane being parallel to the x-z-plane.

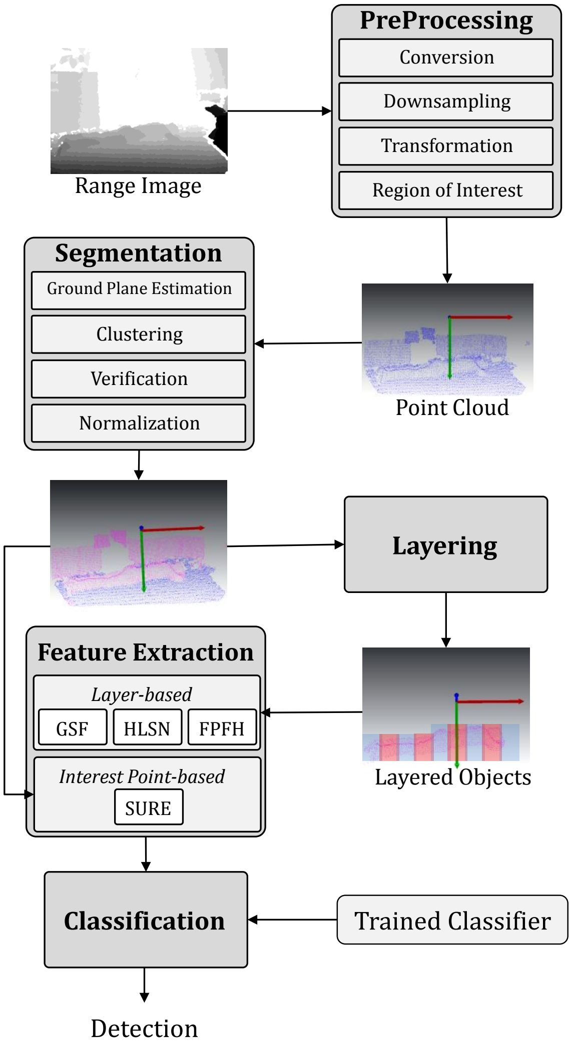

Fig. 1. The proposed approach for detecting fallen people consists of five phases (dark grey boxes).

4) Region of Interest (ROI): Fallen people occur close to the ground, hence only the lower region of the point cloud is of importance. Therefore, all points with a height over $60\,cm$ are removed from the point cloud by using a PassThrough-Filter.

B. Segmentation

The segmentation phase partitions the preprocessed point cloud into objects, which might represent a fallen person.

1) Ground Plane Estimation: The first step of segmenting the point cloud into its individual objects is the detection of points belonging to the floor. Therefore, the ground plane is detected using the RANSAC [29] algorithm. A plane with a normal parallel to the y-axes is used as model. All points supporting the model must have a maximum Euclidean distance to the model of $10\,cm$, and their surface normals must be parallel to the surface normal of the output plane, with a maximum angular deviation of $2°$.

2) Clustering: All points not belonging to the ground plane are subsequently segmented into a sequence of clusters

$\mathbf{C} = \{c_1, \ldots, c_M\}$. For segmentation, an Euclidean clustering technique is applied with a distance threshold of $3\,cm$.

3) Verification: Although the preprocessing restricts the point cloud already to the bounds of the ROI, the point cloud still contains objects which were originally higher than the ROI, as these objects are only cropped by the maximum height of the ROI. To eliminate these objects completely, we remove all segmented clusters from $\mathbf{C}$, whose height is similar to the height of the ROI, with a maximum deviation of $5\,cm$. All clusters remaining in $\mathbf{C}$ might represent a fallen person.

4) Normalization: For the following layering phase a normalization of the position and orientation of the remaining clusters is needed. Therefore, the centroid of each cluster is aligned to the coordinate origin. Afterwards, the cluster is rotated around the centroid by the angle between the x-axes and the cluster's main orientation, which is determined by the eigenvector related to the biggest eigenvalue of the cluster's covariance matrix.

C. Layering

As mentioned in Sec. II, a fallen person might be partially occluded. Therefore, a layer based approach is proposed. During the layering phase, each cluster c_j is partitioned into a sequence of adjacent layers $\mathbf{L}_j = \{l_{j,1}, \ldots, l_{j,K}\}$. To ensure that the layers of different people cover the corresponding body parts, even if the people are partly occluded, we are using a fixed layer size instead of a fixed layer number as [23] and [24] do. Otherwise the layer width and therefore the layer content would vary greatly depending on the degree of the partial occlusion of a person. To define the layer width, we divided the clusters of completely shown, lying people into eight layers. Averaging the layer width of several test data led to a layer width of $22.52\,cm$. To compensate the variance of the body height of different persons and the resulting variance of the layer content, we are using overlapping layers. An overlap of $2.5\,cm$ between two neighboring layers is used.

D. Feature Extraction

The segmentation phase generates a set of 3D clusters $\mathbf{C}$, where each cluster c_j consists of a set of several layers $\mathbf{L}_j$. During the feature extraction phase, for each layer $l_{j,k}$ a feature vector $f_{j,k}$ is computed. As mentioned in Sec. II, we expect features based on surface normals as well as geometrical and statistical features to be well suited for detecting fallen people. In order to determine the feature with the best performance, we evaluated the performance of the following four features.

1) Geometrical and statistical features (GSF): Our approach uses the nine geometrical and statistical features proposed in [24] to characterize the shape of each layer and to classify it in human or non-human (Tab. I).

2) Histogram of Local Surface Normals (HLSN): The HLSN uses a histogram of local surface normals plus additional 2D and 3D statistical features to describe the characteristics of an object. As in [23] we compute a separate histogram with seven bins for each normal axis (x, y and z) over the normals of all points in a layer and add the height and the depth of a layer to the feature vector.

TABLE I
GEOMETRICAL AND STATISTICAL FEATURES IN GSF

No.	Feature	No.	Feature
f_1	Number of Points	f_6	Kurtosis w.r.t. centroid
f_2	Sphericity	f_7	Avg. dev. from median
f_3	Flatness	f_8	Normalized residual planarity
f_4	Linearity	f_9	Number of points ratio
f_5	Std. dev. w.r.t. centroid		

3) Fast Point Feature Histogram (FPFH): The FPFH uses the orientation of the local surface normals to capture the geometry around a query point [25]. The relative differences between a point and its neighbors are captured in the FPFH by determining the differences between their associated normals. It uses a fixed coordinate frame at one of the points, which allows to express the difference between the normals as a set of three angular features. In our approach, we first compute one FPFH for each point of a layer to finally use the mean of all FPFHs as the descriptor of the layer. Yet, one could cluster all FPFHs and use the k centroids of the clusters as a descriptor. This approach would require a permutation of the centroids when classifying, which results in a higher complexity and therefore a higher computational effort.

4) Surface-Entropy (SURE): Besides the layering approach, the use of an interest point-based method is another option to reach robustness towards partial occlusions [27]. Therefore, we are evaluating the performance of the SURE feature. The SURE feature combines an interest point detector and a descriptor, both based on the orientation of surface normals. The interest point detector measures the variation in surface orientation from surface normals and detects local maxima. The extracted descriptor is similar to the described FPFH. As the usage of the interest point detector already leads to a limited number of feature vectors per object, the layering procedure is not used for this feature.

E. Classification

To assign a label (positive: human, negative: non-human) to each layer and to finally decide if an object represents a fallen person, a classification method is needed.

1) Classifier: To obtain the best feature-classifier-combination, we evaluated the performance of four popular machine learning techniques: Nearest Neighbor (NN), Random Forests (RF), Support Vector Machine (SVM) and AdaBoost (AB). The machine learners were further varied by different parameters (number of neighbors, number of trees, type of kernel, etc.), which led to an evaluation of 34 classifiers per feature.

2) Final Detection: While applying the proposed approach for detecting fallen people, we finally obtained a sequence of objects, where each object contains several classified layers. To decide if one of these objects represents a fallen person, the number of positively classified layers per object is analyzed. The evaluation shows, that the best results are obtained by deciding an object to be classified as fallen person, if a minimum of three layers are classified positively.

IV. Experimental Results

In order to evaluate the performance of each feature-classifier-combination and to determine the overall performance of the proposed approach, we carried out a comprehensive evaluation.

A. Data Set

A training and a testing data set were acquired by recording videos with the Kinect of a mobile robot. The data contains range images of positive and negative examples. As a manual annotation of each sample is a time consuming task, data which allowed an automatic annotation was collected. The videos with the negative examples did not contain any people. For the positive examples, we made sure that the person is lying in a predefined region of the range image and that this region did not contain any other objects. In doing so, we acquired data of lying people without any partial occlusion. Therefore, it is guaranteed, that every body part is contained equally often in the training set and that the classifier does not specialize on a certain body area. The videos of the positive training data contain nine different persons each in different poses. The positive test data mainly contains lying people which are partially occluded, as this scenario complies to the situation existing in reality. The videos of the positive test contain ten different persons in different poses. As the focus of this work is on evaluating the best feature-classifier-combination and not on evaluating the segmentation method, the positive data only contains lying people who can be segmented relatively easy from the background as other objects have a certain minimum distance to the person. For the negative data, videos recorded in five real apartments of elderly people, one common room of a retirement home as well as videos recorded in a lab containing objects similar to persons (e.g. a bunch of coats) and two tall dogs were acquired. As the size, shape, and the orientation of the surface normals of large (lying) dogs is similar to the one of lying persons, this data is especially challenging for those features which are based on surface normals. $\frac{2}{3}$ of the negative data is used to train the classifier, the remaining $\frac{1}{3}$ is used for testing. Fig. 2 shows some examples of the used data.

B. Performance Measurement

In the practical application of the proposed approach, a low false positive rate matters more than a low true positive rate. Since the robot is supposed to undertake control trips in the apartment during which it is looking for critical situations, like a fallen person on the floor, a high number of false alarms likely causes a shut down of the system by the user. In case of a fallen person, the robot perceives the person usually more than once during its control trip, which means the robot has more than one chance to detect the fallen person. As the $\mathcal{F}_{0.5}$-score [30] puts more emphasis on precision (PR) than recall (RC) and therefore the false positive detections are weighted higher than the true positive detections, it is used to measure and compare the performance of the different feature-classifier-combinations:

$$\mathcal{F}_{0.5} = \frac{(1 + 0.5^2) \cdot PR \cdot RC}{0.5^2 \cdot PR + RC}. \tag{1}$$

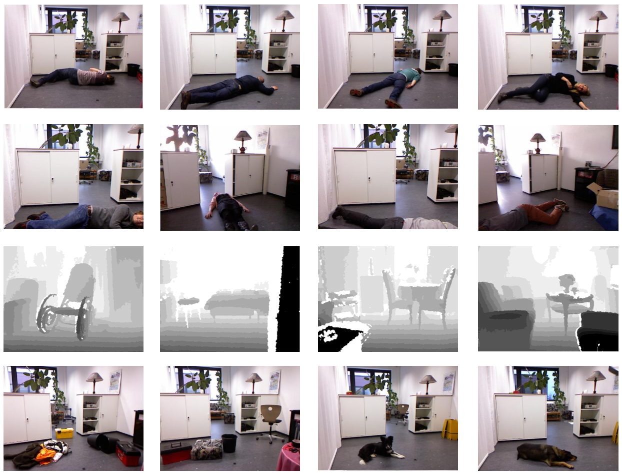

Fig. 2. First row: positive training data. Second row: positive test data. Third row: Negative data from the retirement home. Fourth row: Negative data captured in a living lab

C. Experimental Objectives

1) Object-unspecific Evaluation: In order to obtain the principal performance of each feature-classifier-combination, its object-unspecific performance on separating the two classes (human or non-human) was evaluated. Therefore, the feature vectors of all layers/interest points in the test data were classified, and the number of correctly and wrongly classified layers/interest points were accumulated without considering the object assignment of the layers/interest points.

2) Object-specific Evaluation: Subsequently, the object-specific performance for the best classifier of each feature was evaluated. For this purpose, we analyzed how many layers/interest points per object are assigned to which class. By varying the number of layers/interest points per object which need to be classified positively to classify the whole object as a fallen person, the best feature-classifier-combination with the best parameter setting can be determined. The object-specific detection performance is equal to the overall performance of the proposed approach.

3) Processing Time: For the practical implementation of the proposed approach, real-time capability is required. Therefore, it was determined how long each classifier needs to classify one test sample. Subsequently, the detection performance of the best and the fastest classifier using the best feature were compared.

D. Experimental Results

1) Object-unspecific Evaluation: The results of the object-unspecific evaluation show (see Fig. 3) that the classifier achieving the best detection performance is depending on the features combined. Only the AdaBoost classifier leads to poor performance with all four tested features. A reason for this might be that we used one-dimensional, brute-force

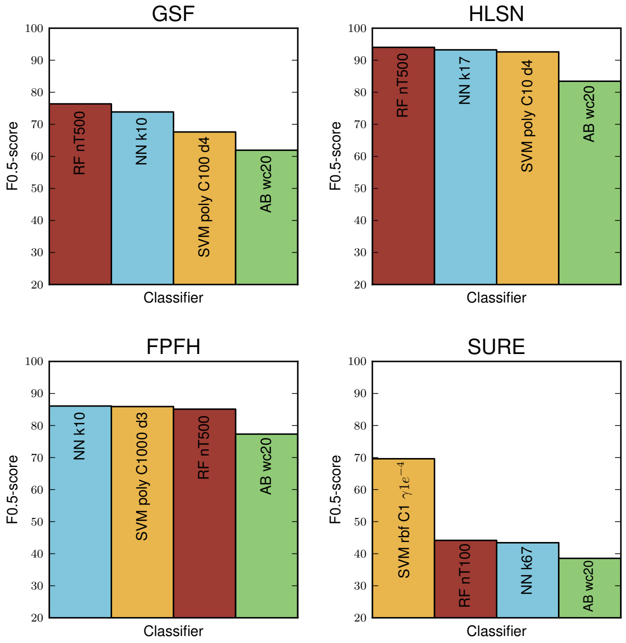

Fig. 3. Results of the object-unspecific Evaluation. The graphic shows the $\mathcal{F}_{0.5}$-score of the four evaluated features with four different classifiers and their best parameters – Nearest Neighbor (NN) with k-Neighbors, Random Forests (RF) with number of trees, Support Vector Machine (SVM) with kernel parameters and AdaBoost (AB) with number of weak classifiers (wc).

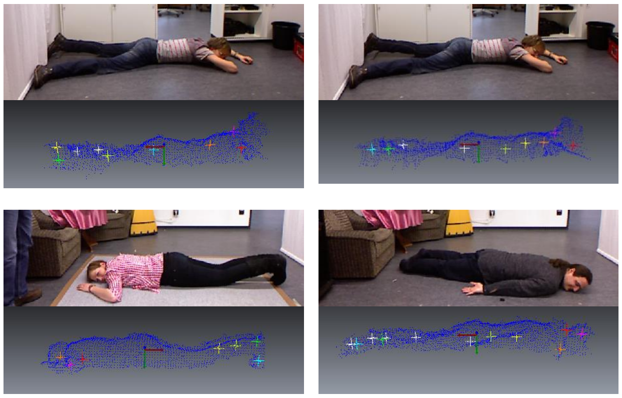

Fig. 4. The number and positions of the detected interest points are unstable on lying people

trained weak learners. Future work could use decision trees with more than 1 stump to increase performance. Focusing on the features, the results show that the HLSN outperforms all other features. Even in combination with the worst classifier (AdaBoost) the HLSN yields better or similar performance in comparison to the other features combined with their best classifier. In addition, the poor performance of the interest point-based SURE feature is conspicuous. As the SURE-descriptor is similar to the FPFH, the difference in performance can be substantiated by an instability of the detected interest points. As shown in Fig. 4, the number and positions of the detected interest points on different people or even on the same person in a slightly different pose is highly unstable.

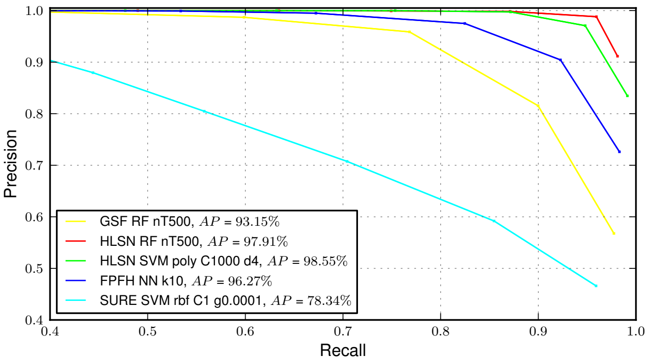

Fig. 5. Results of the object-specific Evaluation. The average precision (AP), determined by integrating the area under each curve is shown in the legend.

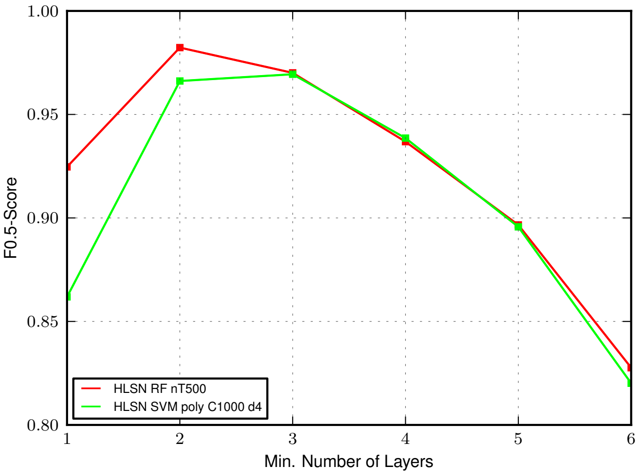

Fig. 6. HLSN: comparison between the classifier with the best detection performance (RF) and the fastest classifier (SVM).

2) Object-specific Evaluation: The results of the object-specific evaluation support the assumptions made on the basis of the object-unspecific evaluation. The HLSN again outperforms the other three features. Fig. 5 shows a Precision-Recall-curve of the results. The average precision (AP, see legend), determined by integrating the area under each curve, confirms the assessments made on base of the $\mathcal{F}_{0.5}$-score. The curve was created by varying the number of layers/interest points per object which need to be classified as positive to classify the whole object as a fallen person.

3) Processing Time: The best detection performance is achieved by the HLSN combined with a Random Forest classifier with 500 trees. But as 500 trees lead to a high degree of complexity, the time necessary to classify one object is $20.16\,ms$, which is too high for a robot operating in real-time. In contrast, a Support Vector Machine with a polynomial kernel function of degree $d = 4$ and a penalty parameter $C = 1000$ is the fastest classifier, taking $0.178\,ms$ to classify one object. As shown in Fig. 6, the difference in detection performance between the RF ($\mathcal{F}_{0.5} = 98.23\%$) and the SVM ($\mathcal{F}_{0.5} = 96.94\%$) is negligible compared to the difference in processing time. Therefore, the HLSN combined with the proposed SVM is considered to be the most suitable feature-classifier combination.

4) Minimum Number of Positive Layers: As shown in Fig. 6 the HLSN combined with the proposed SVM obtains the best results by deciding an object to be classified as fallen person, if a minimum of three layers are classified as positive.

5) Overall performance: The proposed feature-classifier-combination leads to a good overall performance. Due to its recall of $RC = 87.17\%$ and its precision of $PC = 99.74\%$, it achieves a $\mathcal{F}_{0.5}-$score of $\mathcal{F}_{0.5} = 96.94\%$ and an accuracy of $AC = 96.08\%$. The false positive rate is 0.1%, which is very low. The depth videos of the real apartments of elderly people are the most suitable representation of the situation existing during the practical assignment of the proposed approach. A further study on this data has shown, that none of these videos leads to a false positive detection due to the strict preprocessing and the determined feature-classifier-combination. Since recall is not perfect, a fallen person is not detected in every frame. Yet, while the robot drives through the apartment, there are more than enough positive classifications to integrate and finally detect a fallen person.

V. Conclusion

We presented a method to detect fallen people with a mobile robot using only 3D depth data. Our method segments objects in point clouds and layers them to deal with occlusion. Finally, a trained classifier is used to classify the layers of the objects, and fallen people can be detected by a certain number of positively classified layers. Evaluation showed, that the Histograms of Local Surface Normals in combination with a SVM classifier are well-suited to detect fallen people.

Currently, the Euclidean segmentation is the bottleneck of our approach. If people fall on or near furniture, the segmented object sometimes contains the user and parts of the furniture. In future work, the segmentation could be enhanced by combining it with RGB image data. Furthermore, our method uses a layer unspecific classifier which already proved to give good results. Yet, one could use a specific classifier for each layer which should improve accuracy of the method. However, since layers are likely occluded one would need to permute the layers or use an implicit shape model and let each layer vote for the object center [24]. However, both approaches would lead to higher computational requirements. Finally, the robot needs a suitable strategy to search for fallen people in the apartment if it has not recognized the user for a while [31].

References

[1] S. Lord, C. Sherringtion, and H. Menz, "Falls in Older People: Risk Factors and Strategies for Prevention," *Injury Prevention*, vol. 9, no. 1, pp. 93–94, 2003.

[2] N. Noury, P. Rumeau, A. Bourke, G. Laighin, and J. Lundy, "A proposal for the classification and evaluation of fall detectors," *IRBM*, vol. 29, no. 6, pp. 340–349, 2008.

[3] C. Huijnen, A. Badii, and H. van den Heuvel, "Maybe it becomes a buddy, but do not call it a robot - seamless cooperation between companion robotics and smart homes," in *Amb. Int.*, 2011, pp. 324–329.

[4] H.-M. Gross et al., "Further progress towards a home robot companion for people with mild cognitive impairment," in *Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics*. South Korea: IEEE, 2012, pp. 637–644.

[5] Ch. Schroeter et al., "Realization and user evaluation of a companion robot for people with mild cognitive impairments," in *Proc. IEEE Int. Conf. on Robotics and Automation*. IEEE, 2013, pp. 1145–1151.

[6] GmbH Future-Shape, "SensFloor - Hightech fuer mehr Lebensqualitaet," http://www.future-shape.de/en/technologies/23, 2010.

[7] M. Popescu and A. Mahnot, "Acoustic fall detection using one-class classifier," in *Annual Int. Conf of the IEEE Engineering in Medicine and Biology Society*, 2009, pp. 3505–3508.

[8] Philips Electronics, "Autoalert pendant," http://www.lifelinesys.com/content/lifeline-products/personal-help-buttons/auto-alert-pendant, 2012.

[9] M. Shoaib, R. Dragon, and J. Ostermann, "Context-aware visual analysis of elderly activity in a cluttered home environment," *EURASIP Journal on Advances in Signal Processing*, pp. 1–14, 2011.

[10] R. Planinc and M. Kampel, "Introducing the use of depth data for fall detection," *Personal and Ubiquitous Computing*, pp. 1–10, 2012.

[11] C. Rougier, E. Auvinet, and J. Rousseau, "Fall Detection from Depth Map Video Sequences," in *9th Int. Conf. on Smart Homes and Health Telematics*, 2011, pp. 121–128.

[12] G. Mastorakis and D. Makris, "Fall detection system using kintect's infrared sensor," *Real-Time Image Processing*, 2012.

[13] D. Anderson, J. Keller, M. Skubic, X. Chen, and Z. H., "Recognizing Falls from Silhouettes," in *Int. Conf. on Engineering in Medicine and Biology Society*, 2006, pp. 6388–6391.

[14] J. Willems, G. Debard, B. Bonroy, V. B., and T. Goedeme, "How to detect human fall in video? an overview," in *Positioning and context-aware international conference*, 2009, pp. 6388–6391.

[15] S. Wang, S. Zabir, and S. Leibe, "Lying Pose Recognition for Elderly Fall Detection," in *Proceedings of Robotics: Science and Systems*, 2011.

[16] Q. Lv, "A poselet-based approach for fall detection," in *Int. Symposium on IT in Medicine and Education*, 2011, pp. 209–212.

[17] L. Xia, "Human detection using depth information by kinect," in *Computer Society Conf. on Computer Vision and Pattern Recognition Workshops*, 2011, pp. 15–22.

[18] C. Plagemann, "Real-time identification and localization of body parts from depth images," in *IEEE ICRA*, 2010, pp. 3108–3113.

[19] L. Schwarz, "Estimating human 3d pose from time-of-flight images based on geodesic distances and optical flow," in *Int. Conf. and Workshop on Automatic Face & Gesture Recognition*, 2011, pp. 700–706.

[20] L. Spinello and K. Arras, "People detection in RGB-D data," in *IEEE/RSI IROS*, 2011, pp. 3838–3843.

[21] S. Wu, S. Yu, and W. Chen, "An attempt to pedestrian detection in depth images," in *3rd Chines Conf. on Intelligent Visual Surveillance*, 2011, pp. 97–100.

[22] S. Ikemura and H. Fujiyoshi, "Real-time human detection using relational depth similarity features," in *Proc. of the 10th Asian Conf. on Computer Vision*, 2010, pp. 25–38.

[23] F. Hegger, N. Hochgeschwender, K. Kraetzschmar, and P. Ploeger, "People detection in 3d point clouds using local surface normals," in *Proc. of the Robocup Symposium 2012*, 2012.

[24] L. Spinello, K. Arras, R. Triebel, and R. Siegwart, "A layered approach to people detection in 3d range data," in *Proc. of the 24th AAAI Conf. on Artificial Intelligence*, 2010.

[25] R. Rusu, "Semantic 3d object maps for everyday manipulation in human living environments," Ph.D. dissertation, Computer Science department, Technical University Munich, 2009.

[26] S. Tang, *Visual recognition using hybrid cameras*. Thesis, University of Missouri, USA, 2011.

[27] T. Fiolka, J. Stueckler, D. Klein, D. Schulz, and S. Behnke, "Sure: Surface entropy for distinctive 3d features," in *Spatial Cognition VIII*, ser. LNCS. Springer, 2012, vol. 7463, pp. 74–93.

[28] B. Steder, R. Rusu, K. Konolige, and W. Burgard, "Narf: 3d range image features for object recognition," in *Workshop on Defining and Solving Realistic Perception Problems in Personal Robotics at IROS*, 2010.

[29] M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography," *Commun. ACM*, vol. 24, pp. 381–395, 1981.

[30] C. J. V. Rijsbergen, *Information Retrieval*, 2nd ed. Newton, MA, USA: Butterworth-Heinemann, 1979.

[31] M. Volkhardt, S. Müller, C. Schröter, and H.-M. Gross, "Playing Hide and Seek with a Mobile Companion Robot," in *Proc. 11th IEEE-RAS Int. Conf. on Humanoid Robots*, 2011, pp. 40–46.