# Automated Walking Aid Detector Based on Indoor Video Recordings*

Steven Puttemans[1], Greet Baldewijns[2], Tom Croonenborghs[2], Bart Vanrumste[2] and Toon Goedemé[1]

***Abstract*— Due to the rapidly aging population, developing automated home care systems is a very important step in taking care of elderly people. This will enable us to automatically monitor the health of senior citizens in their own living environment and prevent problems before they happen. One of the challenging tasks is to actively monitor walking habits of elderlies, who alternate between the use of different walking aids, and to combine this with automated fall risk assessment systems. We propose a camera based system that uses object categorization techniques to robustly detect walking aids, like a walker, in order to improve the classification of the fall risk. By automatically integrating the application specific scenery knowledge like camera position and used walker type, we succeed in detecting walking aids within a single frame with an accuracy of 68% for trajectory A and 38% for trajectory B. Furthermore, compared to current state of the art detection systems, we use a rather limited set of training data to achieve this accuracy and thus create a system that is easily adaptable for other applications. Moreover, we applied spatial constraints between detections to optimize the object detection output and to limit the amount of false positive detections. Finally, we evaluate the output on a walking sequence base, leading up to a 92.3% correct classification rate of walking sequences. It can be noted that adapting this approach to other walking aids, like a walking cane, is quite straightforward and opens up the door for many future applications.**

## I. INTRODUCTION

As more and more older adults will require medical care in the coming years, the development of automated home-care systems has evolved into an important research field. Automated home-care systems aim to automatically monitor the health of senior citizens in their own living environment, enabling the detection of an initial decline in health and functional ability and providing an opportunity for early interventions. Fall prevention is one of the research fields in which automated home care systems can be an important asset. When approximately one in three older than 65 fall each year [10], [12] and 20-30% of those who fall sustain moderate to severe injuries, it is clear that fall prevention strategies should be put in place to reduce this number. When these automated systems detect an elevated fall risk, preventive measures can be taken to reduce this risk, e.g. installing an exercise program to enhance gait and mobility, adapting the medication regime and introducing walking aids such as walkers.

In order to continuously monitor the fall risk of a person in their home environment, the development of an automated fall risk assessment tool is needed. For this purpose a camera based system was installed in the homes of three senior citizens during periods varying from eight to twelve weeks, all three using walking aids to move around. The resulting video data was used to monitor predefined trajectories using the transfer times as an indicator of the fall risk, since they heavily relate to the general health of the person [8].

Previous studies [2] observed that the gait speed of a person differs when using walking aids like a walker or a cane. Transfer times measured with aids can therefore not be compared to transfer times measured without them. This underlines the necessity to automatically differentiate between the different video sequences and to determine whether a walking aid was used or not. Our system presented in this paper will focus on automatically detecting which walking aid is being used in the selected transfers based on the given video data, and more specifically focusing on the case of a walker. Examples of such a walker walking aid can be seen in Fig. 1. The output information of our software can then be used efficiently to differentiate between walker based sequences and non-walker based sequences (using a walking cane or no walking aid at all).

The applications of walking aid detection are not only limited to monitoring these transfer times. We can expand the applications to every situation where we want to have an objective measure of the indoor usage of any type of walking aid, like e.g. in an elderly home, where we could monitor how frequently people leave their room with a walking aid, by pointing the camera towards the entrance of the room.

Computer vision research provides a set of object detection algorithms called object categorization techniques, which focus on detecting object classes with a large intra

*This work is supported by the *IWT* via the *TOBCAT* project and by *iMinds* via the *FallRisk* project.

[1]Puttemans S. and Goedemé T. are with EAVISE, KU Leuven Department of Industrial Engineering Sciences. Both researchers are associated with ESAT/PSI-VISICS of KU Leuven. [steven.puttemans, toon.goedeme]@kuleuven.be

[2]Baldewijns G., Croonenborghs T. and Vanrumste B. are with AdViSe Technology Center, KU Leuven Department of Industrial Engineering Sciences. Baldewijns and Vanrumste are also with the iMinds Future Health Department and ESAT-SCD of KU Leuven. Croonenborghs is also with KU Leuven Department of Computer Science. [greet.baldewijns, tom.croonenborghs, bart.vanrumste]@kuleuven.be

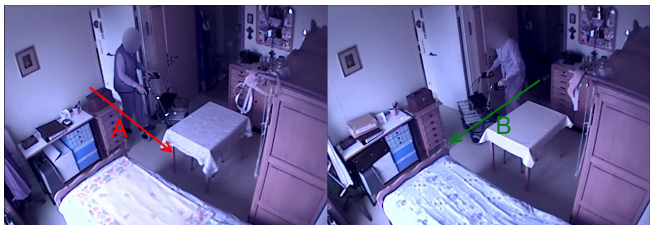



Fig. 1. Input frames with a walker present and both trajectories visualized.



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 13:41:05 UTC from IEEE Xplore. Restrictions apply.

class variance. We are aiming at using such techniques to automatically detect if a walker has been used in a certain indoor walking track. This benefits the automated analysis of transfer times a lot, removing the need to manually label each sequence and making the measurements more meaningful by automatically adding a label of the used walking aid. This switches the process from semi-automatic towards fully automated fall risk assessment. Similar steps can be applied to other walking aids like a walking cane to further differentiate the non-walker based sequences. As a result we are able to give caregivers an objective measure of the walker usage.

The remainder of the article is organized as follows. Section II will present related research. Then section III discusses how the data was collected and how the object model was trained using scene- and application-specific constraints. This is followed by section IV in which the complete proposed processing pipeline is evaluated. Section V elaborates on the obtained output of the algorithm on real life data. Finally, section VI draws some meaningful conclusions, while section VII discusses possible future adaptations to the current system.

## II. Related Work

Automatic detection of a walker in random video material is not an easy task. Many existing behavior measuring systems have inconvenience and obtrusiveness as a major downside [7]. Combined with that, many of the created approaches are tested in lab environment situations. These are not really representative for actual home situations, like in our case. The major advantages of our camera based approach are therefore the unobtrusiveness of the system, the fact that deviating walks are automatically discarded and that the algorithm works on a real life dataset.

In the computer vision literature, object categorization techniques like [3], [4], [6], [13] have been used exhaustively to robustly detect objects in very versatile situations going from outdoor pedestrian detection to microscopic organism analysis. A benefit of object categorization techniques is that they capture the variance of objects in color, shape, texture and size from large positive and negative object samples sets and model that variance into a single model. This results in a single model at detection time to perform robust detection of object class instances. Downsides of these techniques are the need of large quantities of training data and the long training time needed to obtain a single model. However, the research of Puttemans et al [11] has proved that even in situations where limited training data is available and where set-up and scene-specific constraints (like a constant illumination, a fixed camera position or a known background set-up) apply, that object categorization techniques manage to get very accurate detection results with a small set of training data.

The case of detecting walking aids for elderly people is an example of such an object categorization application in constrained scenery and is therefore an ideal case to prove the theorems suggested by [11]. First of all we have a fixed camera set-up, with multiple cameras observing the scene. Secondly this is combined with the fact that many elderly people have a fixed home set-up (location of bed, table, wardrobes, etc.) resulting in a known environment. This ensures that we have two advantages, namely a known background and a known object size range. By using this application specific knowledge, we succeed in building a robust walker detector. We also aim at using a limited training data set and a reduced training time in order to ensure that a specific set-up can be learned within a single day timespan.

Many existing industrial object detection applications rely heavily on uniform lighting conditions and a limited variation of the objects that need to be detected, leading to threshold-based segmentation of the input data. However, in our specific case, the variance of the object, a walker, is amongst others heavily due to lighting changes, day and night conditions and different viewpoints (fixed camera set-up but side and front view of walker in single sequence). This makes it nearly impossible to use segmentation approaches like [9]. On the contrary we have a known camera position, a rather known environment and a set of known trajectories that are followed by the elderly person. We show that these restrictions can effectively limit the amount of training data needed and increase the accuracy of our detector model to meet the desired standards.

## III. The setup

In this section we discuss how the dataset for training the walker detection model was acquired and filtered in order to reach usable sequences for object model training. We will also take a closer look at how scene-specific knowledge was used to further improve the detection result.

### A. The acquired dataset

The data for this paper was acquired using a multiple wall-mounted IP-camera set-up in the home environment of a participant recruited through convenience sampling. The participant is a seventy-five year old female living in a service flat, alternating between the use of a walker, a cane or no walking aid. She was monitored for a period of twelve weeks in which 444 walking sequences were automatically detected and timed using the research presented in [2].

In this research we only trained a detector for a walker. We decided to split up all walking sequences into direction specific parts, since the viewpoint changed too drastically during the complete walking sequences to be able to train a single object model. Also in large parts of the sequences the walking aid was completely occluded by the user, which doesn't yield usable training and validation data. We first defined trajectory A, as seen in Fig. 1, which is a forwards moving trajectory with a side view of the walker, followed by trajectory B, which is a reverse moving trajectory with a front view of the walker. Since the original video sequences had about double the amount of walks in trajectory A, compared to trajectory B, we kept the same ratio in training and test data between both trajectories.



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 13:41:05 UTC from IEEE Xplore. Restrictions apply.

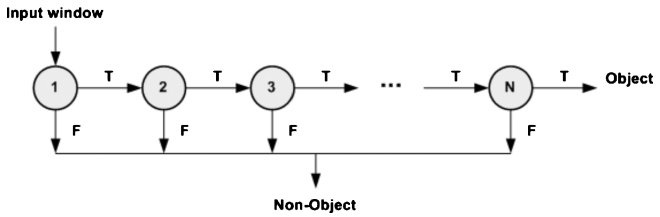



Fig. 2. An example of a cascade of weak classifiers with features calculated at each stage of weak classifiers and windows rejected as objects (T) or non-objects (F) by the early rejection principle. At each stage (1...N) multiple weak classifiers depending on single feature values are combined until the performance criteria is met.

From the pool of available and split video data, corresponding to both trajectories A and B, we randomly grabbed a training set of ten sequences of trajectory A and five sequences of trajectory B. Since trajectories could happen in both day and night conditions we assured that both conditions were available in the training set of both trajectories. The test set contains twelve randomly picked sequences for evaluating the detector of trajectory A and five sequences for evaluating the detector of trajectory B, again both containing day and night conditions and different from the selected training datasets. Since the original video sequences contained many frames without movement, only the parts with actual movement were kept and the other frames were discarded. A manual annotation of the walker location in all training frames was performed, storing the exact location of the walking aid appearances. This resulted in a positive training samples set of 695 walker annotations for trajectory A and 2200 walker annotations for trajectory B. The main difference in the amount of annotations is due to the duration of the sequence measured, the longer the sequence the more frames with walker appearances. As negative training data we used the provided sequence images, since they contain all the info which is needed, and set the pixel values of those regions to 255. This resulted in as many negative frames without walker aids as there are positive samples. However, these images are much larger than the model size and the positive training samples. Therefore we randomly sample, using the size of the model which is based on the average annotation size, 2000 negative training windows for trajectory A and 4000 negative samples for trajectory B. The ratio of positive versus negative samples is approximately $1:2$ and rounded to the nearest 1000 samples. This ratio is chosen from the experience of training multiple generic object detection models.

Since this application will be used as a reference for training similar walking aid setups, we like to keep a look at how long it takes to collect the training data. On average, once a person gets familiar with the annotation tool, he tends to provide around 500 annotations per hour.

### B. Our object detection approach

We used the approach suggested by Viola & Jones [13] as base technique for training the object model and performing the object detection. This technique focuses on generalizing specific properties of an object class into a single object model. It generalizes well over a large set of positive and negative examples in order to reach a generic object model that can be used to look for new object instances in test images. In order to learn this generic object model it uses a learning principle called AdaBoost combined with local binary pattern features (LBP) [14]. The approach combines a set of weak performing classifiers into a single good performing classifier based on a cascade structure approach as seen in Fig. 2. This ensures that the classification of negative samples improves compared to each weak classifier and thus the overall amount of false positive detections is drastically decreased. The choice of using LBP features [14] instead of the classical Haar wavelet features [13] is mainly due to the fact that they have a considerably faster training time. The main advantages of using a cascade of weak classifiers is the early rejection principle, where a small set of features is used to discard most object candidates, up to 75%. Only object candidates that pass these first weak classifiers will progress and will require more features to be calculated.

For both the object model of trajectory A and B, we choose a minimum hit rate of 0.995 and a maximum false alarm rate of 0.5, the default values assigned by the Viola and Jones framework [13]. This means that we need to correctly classify at least 99.5% of our positive training samples at each weak classifier stage while correctly classifying at least 50% of the used negative training samples. Both models can be trained within 4 hours of processing, still keeping it possible to get the whole setup up and running within a single day timespan.

## IV. Complete pipeline

In this section we will discuss the several building blocks (see Fig. 3) needed to supply a correct label for each video sequence, indicating if it is a walker or non-walker (walking cane or no walking aid) based sequence. The following subsections will discuss each part of this pipeline and the measures taken for increasing the accuracy of our algorithm.

### A. Using scene-specific information

This research focuses on detecting walking aids in constrained scenes like manually defined walking trajectories of

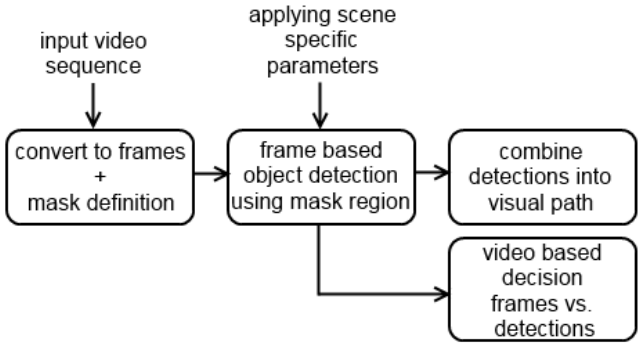



Fig. 3. The separate building blocks for performing the video sequence classifying into walker and non-walker sequences.



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 13:41:05 UTC from IEEE Xplore. Restrictions apply.

elderly people or specific regions that we want to monitor. Since we want to make a system that is as versatile as possible, we start by supplying the user with the ability to define a region of interest, under the form of a binary mask that can be created based on the users input. This mask contains the application specific regions of the input images in which a center point of an object instance can occur. Fig. 4 clearly shows how the user is asked to visually assign a mask to the recorded sequence. The mask allows us to simply ignore all windows that are not part of that mask with their centroid.

Since the exact position of the camera based capturing system is known, we can use this knowledge to apply some scene-specific constraints to the detection algorithm. Normally a complete image pyramid is built in order to perform multi scale detections [1]. However, the larger the input image, the larger the image pyramid and thus the larger the possible search space of object candidates. Using the knowledge of the fixed camera setup we reduce this search space drastically. Based on the provided training samples we can define an average width and height of object instances in the annotations. These dimensions are used to create a narrow scale range that prunes the image pyramid, leading to a huge reduction of the object candidates search space. The benefits are two-fold. The first benefit is that it reduces the time needed to process a single image drastically, since entire scale levels from which windows are sampled are removed. The second benefit is due to the removal of the scale layers, which also reduces the number of false positive detections for a given input image because there are less scale levels from which candidate windows are sampled. This means that the reduction of the image pyramid benefits both in accuracy and processing time.

### *B. A frame-by-frame detection of object instances*

The object categorization is applied on a frame-by-frame basis and has a huge downside that, due to the limited training data, it still yields false positive detections. Even when there is only a single walker object in the frame, this can still result in multiple detections in that frame. Applying a predefined mask region inside the larger images reduces already the amount of false positive detections, but they will never be gone completely, due to the limited training dataset. In order to avoid this problem we take into account a spatial relation between detections, by assuming only a small position shift between detections in consecutive frames. This assumption in spatial relation is proven by the fact that persons have a limited moving speed, which is certainty lower than the capturing speed of a standard camera (25 FPS). This means that the processed frames originate from a video sequence and thus that the detected walker positions should have a small spatial relation to each other.

The spatial relation can be found by looking for a connection between the obtained detections in the currently detection-triggering frame $F_T$ and the selected detection in the last processed frame $F_{T-1}$ as seen in (1). For each new detection, $D_1$ to $D_N$, of the current frame $F_T$, the Euclidean distance is calculated with the selected reference detection ($D_R$) from the last processed frame. Based on those scores, the detection with the smallest distance is kept for the current frame and stored as the new reference $D_R$ value.

$$D_R = \min_{i=1:N} [dist(D_i, D_R)] \tag{1}$$

At the first frame of a sequence, the average location of all triggered detections is used, since at that time no $D_R$ value exist. Due to the limited mask region that is selected the error introduced by this is rather limited and can thus be neglected. Fig. 5 illustrates this spatial relation between walker detections inside trajectory A and B and thus removing all detections that are further away. The complete calculation is repeated for each new frame where detections occur, else the frame is simply ignored in the pipeline. Applying this extra measure ensures another large decrease in false positive detections that occur inside a single frame. The combined trajectory is also visually stored for further analysis.

### *C. Detection on sequence basis*

The visual output is good for verifying the work of the object detector on a single sequence, but when someone needs to go through a large set of sequences, then this visual confirmation would be insufficient and an automated decision would be needed. In our approach we do this by training a decision classifier based on example sequences.

Due to the limited data we cannot train a classifier that will yield no false positive detections, so simply depending

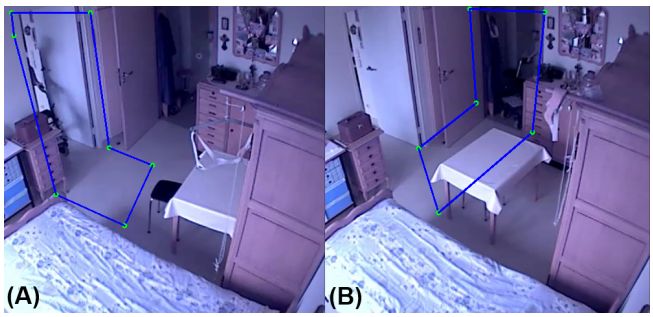



Fig. 4. Manual defined mask for (A) front model and (B) backwards model

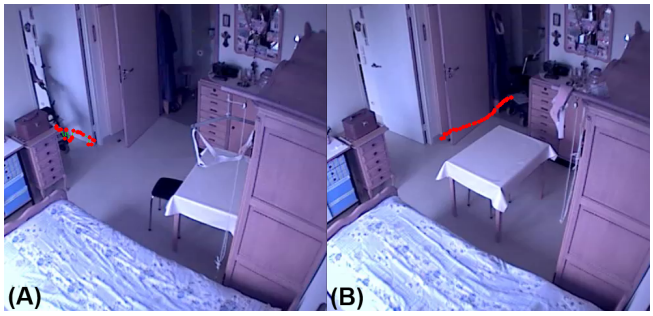



Fig. 5. Detection results for (A) trajectory A and (B) trajectory B trained object models.



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 13:41:05 UTC from IEEE Xplore. Restrictions apply.

on the fact that a detection occurred to classify a sequence is a bad idea. However, the amount of found detections is also highly dependent on the amount of frames that were processed from the initial sequence. We decided to calculate the ratio between the amount of walker detections over the complete sequence (#D) and the total number of frames (#F) inside a walking sequence. This ratio is called the walker certainty score ($C_{walker}$):

$$C_{walker} = \frac{\#D}{\#F} \tag{2}$$

We determined the optimal value of the walker certainty score by randomly selecting a subset of 20 validation sequences from the remaining video data, of which 10 walker based sequences (half trajectory A and half trajectory B) and 10 non-walker based sequence (again half trajectory A and half trajectory B), containing both day and night conditions and in the case of the non-walker sequences containing walking cane and no walking aid sequences. For this complete validation set we calculated the walker certainty score using the appropriate detection model and analyzed the data. We selected a threshold that was low enough to classify walker sequences with lower amount of good detections still as a walker sequence but ensuring that the few false positive detections that can occur in a non-walker sequences (e.g. by lighting changes or a walking cane that does trigger a detection) still get discarded and thus also get correctly classified. This yielded a walker certainty score threshold of 0.2, which works fine for both trained models, but which can be re-evaluated once a larger set of previously unused walker and non-walker sequences is available.

## V. Results

### A. Frame-by-frame detection results

Fig. 6 shows the precision-recall curves for both walker detection models for trajectory A and B. They show the maximal achieved accuracy of the trained models on the test set on a frame-by-frame basis. The curves were obtained by iteratively thresholding the detection certainty scores generated by the Viola & Jones framework [13] in small increasing steps. In order to quantify the accuracy in a single value, we calculated the area under the curve, where 100% equals the ideal object detector. We notice that the AUC value for trajectory A is better than for trajectory B. This can be explained by the fact that the training data for trajectory B was biased due to the walker occlusion from the dining table being in front of the trajectory.

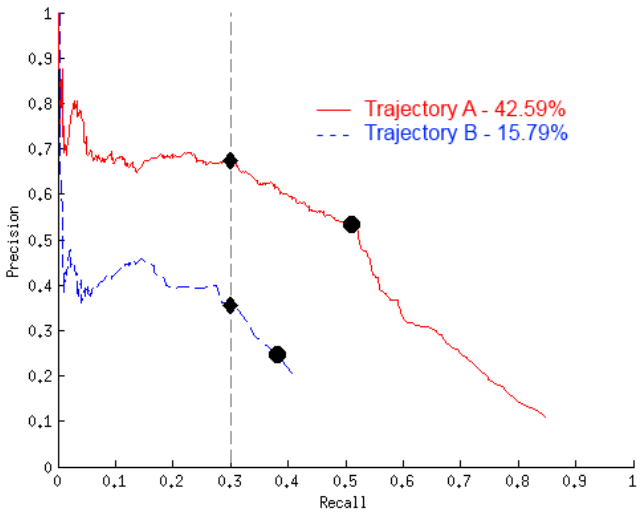



Fig. 6. Precision-recall curves for frame based object detection on both trajectory A and trajectory B. • denotes the best achieving configuration purely based on the model performance, while ♦ denotes the selected configuration for our specific setup.

In theory the best operating point (indicating the best detection threshold) is reached by selecting the curvature point that is closest to the top right corner, denoted by •. However, in our application we want a detection threshold that yields as less false positive detections as possible but still ensures that there are enough true positive detections (true positive detections with lower score also disappear when increasing the threshold) to retrieve a walker certainty score that is over 0.2. In our case this position was set at a recall level of 0.3, denoted by ♦. This was validated on the same validation set that was used to determine the optimal walker certainty score.

TABLE I

Confusion matrix of algorithm output using both trajectories A and B with a $C_{walker}$ = 0.2.

| | Classified as Walker | Classified as Non-Walker |
|---|---|---|
| **Walker Sequence** | 14 | 2 |
| **Cane Sequence** | 0 | 6 |
| **No Aid Sequence** | 0 | 4 |

### B. Sequence based detection results

All the above results are based on a frame-by-frame analysis, specific to the object categorization technique. In order to get a performance measure of our complete algorithm we grabbed the collected mixed test set of 26 walking sequences from both the frame-by-frame test set and the validation set of the walker certainty score threshold. For both the models of trajectory A and B, we used sequences containing walkers (16), a walking cane (6) or no walking aid (4) at all. The results of this walking sequence based classification can be seen in the confusion matrix in Table I.

We do acknowledge that still 2 walker sequences are classified incorrectly. After a small investigation into the matter it seems that their threshold was just below the set 0.2 walker certainty score. Further testing and validating over a larger set to find the most optimal threshold is required in this case.

### C. Training data versus accuracy

Since the goal is to supply a tool for caregivers that needs as little manual input as possible to obtain a result as accurate as possible, we roughly tested the relation between the amount of training data and the accuracy achieved by the frame-by-frame walker detector, since it is the core part of our pipeline. For this we took the model of trajectory A and



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 13:41:05 UTC from IEEE Xplore. Restrictions apply.

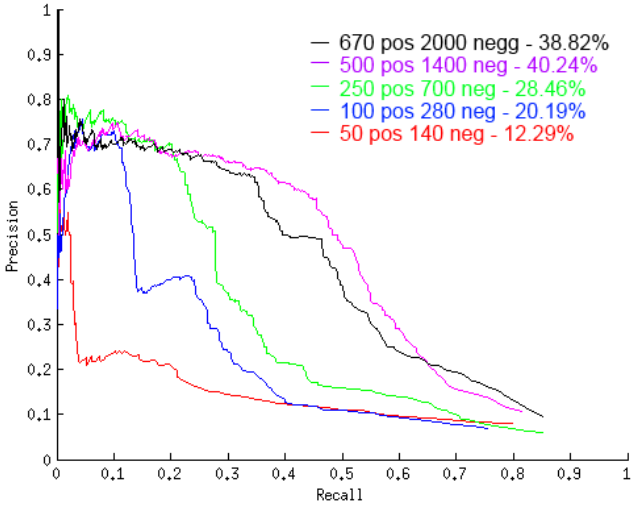



Fig. 7. PR curves under varying amounts of training data with the achieved object detection accuracy defined by the area under the curve (AUC).

varied the amount of positive and negative training data (see Fig. 7), while still keeping the pos:neg ratio at 1:2.8, which is the same ratio as for the training model.

We noticed that the first models show a steep increase in accuracy when adding more data, while reducing a bit when arriving at the final amount of data. For the complexity of the setup this test suggests that we would have achieved a slightly better result by using only 500 positive samples instead of all the 670 positive samples. This behavior is probably due to overfitting the training data.

## VI. Conclusion and discussion

The goal of this paper is automatically detecting walking aids in walking videos of elderly people with the goal of improving automated fall risk assessment. We applied a LBP feature based approach with an AdaBoost learner to robustly create two object detectors for walkers in two separate trajectory directions. Several improvements to the basic algorithm of Viola and Jones based on the scene and set-up specific knowledge were applied.

We proved that our approach can be used to automatically label video sequences for automated fall risk assessment. While the detection rates we achieved on frame-by-frame basis were only around 68% for trajectory A and 36% for trajectory B (both measured on a recall of 0.3), we achieved 92.3% accurate detection results on full trajectory video sequences by adding the spatial relation constraint and the walker certainty score thresholding. Secondly we showed that we created an easy to train system for a specific walker type, of course for a specific situation, but limiting the training time to just a couple of hours while still preserving decent accuracy. This ensures that it is easy to set-up a new system for another walker type in another location. We also proved that the amount of training data can be related to the achieved detection accuracy and the setup complexity. Since the performance difference between the best model and the used model for the approach is rather limited, we ignored the influence on the sequence based validation and did not repeat all tests with the best model.

For validating the complete pipeline with the case of using a walking cane, the available training set was too limited and the cane itself existed of only a few image pixels due to the low camera resolution, which does not work well with object categorization techniques.

## VII. Future Work

Future improvements could be to tackle the problem of applying this technique or similar approaches to walking canes. We are convinced that by simply increasing the video resolution, we could easily detect walking canes with our approach. Another improvement could be to make a more generic walker model that allows to detect multiple walker types by just one generic object class model. Of course we would need much more training data for that purpose and the limitation of a short training time should then be discarded.

Finally, we would also like to point out the benefits of further investigating the relation between training data and detection accuracy. Since this system should be easy to set-up by caregivers, looking for more ways to reduce the manual input sounds interesting. The ideal situation would be a one-shot-learning approach, as suggested by [5].

## VIII. Acknowledgment

The authors would like to thank the researchers of the *AMACS* project for the data collection as well as the persons who participated in the research by giving their permission to be monitored during several months.

## References

[1] E. H. Adelson, C. H. Anderson, et al. Pyramid methods in image processing. *RCA engineer*, 29(6):33–41, 1984.
[2] G. Baldewijns, G. Debard, et al. Semi-automated video-based in-home fall risk assement. In *AAATE 2013*, pages 59–64, 2013.
[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886–893. IEEE, 2005.
[4] P. Dollár, Z. Tu, et al. Integral channel features. In *BMVC*, volume 2, page 5, 2009.
[5] L. Fei-Fei, R. Fergus, et al. One-shot learning of object categories. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(4):594–611, 2006.
[6] P. Felzenszwalb, D. McAllester, et al. A discriminatively trained, multiscale, deformable part model. In *CVPR*, pages 1–8. IEEE, 2008.
[7] S. Hagler, D. Austin, et al. Unobtrusive and ubiquitous in-home monitoring: a methodology for continuous assessment of gait velocity. *Biomedical Engineering, IEEE Transactions on*, 57(4):813–820.
[8] S. A. B. R. W. Kressig. Laboratory review: the role of gait analysis in seniors mobility and fall prevention. 2010.
[9] C. M. Lee, K. E. Schroder, et al. Efficient image segmentation of walking hazards using ir illumination in wearable low vision aids. In *ISWC*, pages 0127–0127. IEEE, 2002.
[10] K. Milisen, E. Detroch, et al. Falls among community-dwelling elderly: a pilot study of prevalence, circumstances and consequences in flanders. *Tijdschr Gerontol Geriatr.*, 35(1):15–20, 2 2004.
[11] S. Puttemans and T. Goedemé. How to exploit scene constraints to improve object categorization algorithms for industrial applications? In *VISAPP*, volume 1, pages 827–830, 2013.
[12] M. E. Tinetti. Preventing falls in elderly persons. *New England Journal of Medicine*, 348(1):42–49, 2003.
[13] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, pages I–511, 2001.
[14] L. Zhang et al. Face detection based on multi-block lbp representation. In *Advances in Biometrics*, volume 4642 of *Lecture Notes in Computer Science*, pages 11–18. Springer Berlin Heidelberg, 2007.



Authorized licensed use limited to: KU Leuven Libraries. Downloaded on May 11,2020 at 13:41:05 UTC from IEEE Xplore. Restrictions apply.