34th Annual International Conference of the IEEE EMBS
San Diego, California USA, 28 August - 1 September, 2012

Automated Fall Detection on Privacy-Enhanced Video

Alex Edgcomb and Frank Vahid*

***Abstract*—A privacy-enhanced video obscures the appearance of a person in the video. We consider four privacy enhancements: blurring of the person, silhouetting of the person, covering the person with a graphical box, and covering the person with a graphical oval. We demonstrate that an automated video-based fall detection algorithm can be as accurate on privacy-enhanced video as on raw video. The algorithm operated on video from a stationary in-home camera, using a foreground-background segmentation algorithm to extract a minimum bounding rectangle (MBR) around the motion in the video, and using time series shapelet analysis on the height and width of the rectangle to detect falls. We report accuracy applying fall detection on 23 scenarios depicted as raw video and privacy-enhanced videos involving a sole actor portraying normal activities and various falls. We found that fall detection on privacy-enhanced video, except for the common approach of blurring of the person, was competitive with raw video, and in particular that the graphical oval privacy enhancement yielded the same accuracy as raw video, namely 0.91 sensitivity and 0.92 specificity.**

***Keywords*— Fall detection, video privacy, assistive monitoring, smart homes, telehealth.**

I. Introduction

Falls are detectable by algorithms that process raw video from in-home cameras [5][8][12][14][15], but raw video raises privacy concerns, especially when stored on a local computer or streamed to a remote computer for processing. In assistive technology, privacy protection needs to be adjustable to a user's wishes while still yielding sufficient detection accuracy [2][3][6]. We explore how a fall detection algorithm performs on several forms of privacy-enhanced videos. The key point, summarized in Figure 1, is that the minimum bounding rectangles (MBR) around a video's foreground object are similar whether created from raw video or from privacy-enhanced video. A fall-detection algorithm based on MBRs may thus perform equally well on raw video and privacy-enhanced video.

Previously, Anderson [1] examined fall detection on videos privacy-enhanced by silhouetting. Our work considers four privacy-enhancements to provide user's with more customizability, which is essential to assistive technology adoption [3][6], comparing accuracy of a fall detection algorithm on those videos versus raw videos.

II. Fall detection on raw video

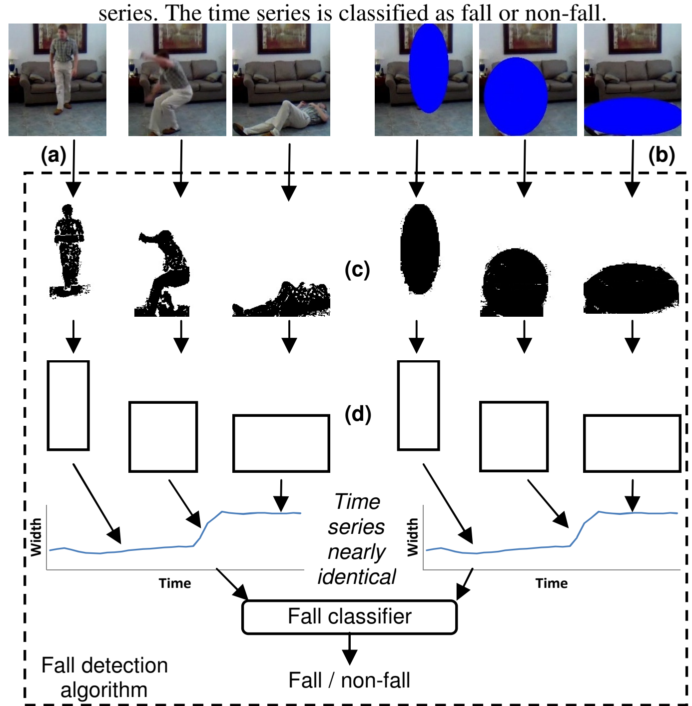
Raw video is unaltered video that shows as much of the camera's scene as possible. Many algorithms detect falls in raw video [5][8][12][14][15]. These algorithms typically use foreground-background segmentation to extract the foreground from a video frame, and then use features of the foreground to classify whether a fall has occurred.

A. Foregrounding via foreground-background segmentation

Foreground-background segmentation is the splitting of moving objects in a video's foreground from the video's static background. Typical moving objects in a home are people, especially in the home of a live-alone elderly person being monitored for falls. Many algorithms for accurate foreground-background segmentation exist [7][13]. Such algorithms typically detect non-changing portions of an image as representing the image background, and then subtract the background image from video frames, thus leaving only the moving objects.

We implemented a foreground-background segmentation algorithm using the OpenCV library in C++ [10]. The algorithm takes as input a background image (Figure 2(a)) and a video frame (Figure 2(b)), and computes the absolute difference in color between each pixel of the video frame and the respective pixel of the background image. If the absolute difference exceeds a threshold, then the respective pixel of the foreground image is colored black. Otherwise, the pixel is colored white. The algorithm outputs a

Figure 1: The fall detection algorithm is the same for (a) raw video as for each privacy-enhanced video, including (b) bounding-oval video. Foreground-background segmentation results in (c) foreground video. An (d) MBR is placed around the foreground video. A feature of the MBR, such as width, over time is a time series. The time series is classified as fall or non-fall.



*Alex Edgcomb and Frank Vahid are with the Department of Computer Science and Engineering, University of California, Riverside, CA, 92507. (e-mail: [aedgcomb/vahid]@cs.ucr.edu, homepage: [http://www.cs.ucr.edu/ [~aedgcomb/~vahid]). Frank is also with the Center of Embedded Computer Systems at UC Irvine.

Figure 2: Computing a foreground by foreground-background segmentation, whether operating on raw or privacy-enhanced video, takes as input (a) a background image and (b) a video frame, and outputs (c) a foreground.

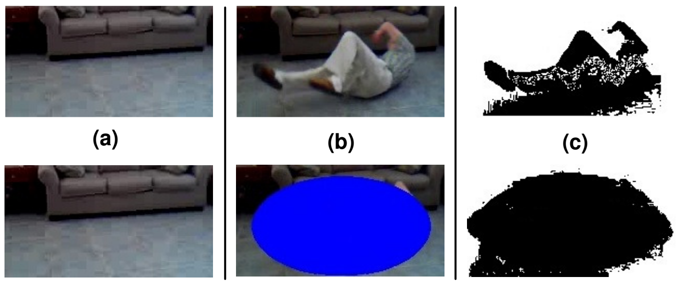
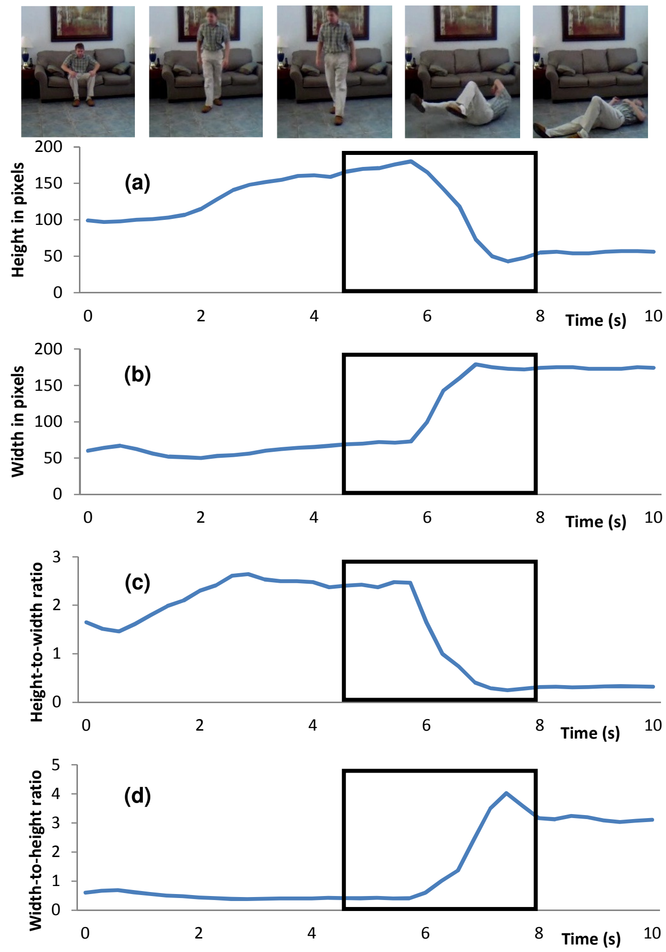


Figure 3: A fall sequence is shown along with four features based on the MBR's: (a) height in pixels, (b) width in pixels, (c) ratio of height-to-width, and (d) ratio of width-to-height. The highlighted region is a characteristic fall pattern for the respective feature.



foreground image (Figure 2(c)).

B. *Fall classification using foregrounds*

Many fall classification methods using foregrounds have been developed [1][5][8][12][14][15]. The general approach is to determine which features are good predictors of a fall, and then build a modeling system to determine if a fall has occurred.

Some approaches use a threshold for the modeling system. Miaou [8] uses the ratio of the foreground's height-to-width as a feature, and if that ratio exceeds a threshold, then a fall is said to have occurred. Williams [15] uses the ratio of the foreground's width-to-height as a feature, and assumes that a person is on the floor if a threshold is exceeded. If the person is on the floor for too long, then the system determines that a fall has occurred.

Other approaches use Hidden Markov Models (HMM) to generate a likelihood that a fall has occurred. Anderson [1] uses the ratio of the foreground's width-to-height along with the off-diagonal term from a covariance matrix as features. A threshold is set for each of the respective features, which indicates a fall has occurred. The threshold indications are the inputs to an HMM that determines whether a fall is likely to have occurred. Cucchiara [5] uses the foreground as input to a tracking algorithm that accounts for occlusions and outputs an MBR of the predicted person. The width-to-height ratio of the MBR is input to an HMM to generate a likelihood of a fall. Thome [14] builds an MBR around the foreground and computes the angle between the height and width of the MBR. A layered HMM uses the angle as the input feature for determining the probability that a fall has occurred.

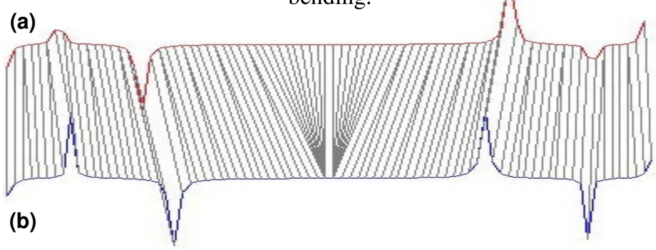
Rougier [12] applies a unique approach that uses the foreground to prune edges created from edge detection on the video frame. The resulting edges produce a shape of the moving object, which is assumed to be a person. The shape of the edges is compared to the shape of pre-determined fall shapes using Procruste Shape Analysis. The likelihood of a fall is based on the similarity of the shapes.

Our work considers four features based on the MBR around the foreground: height in pixels, width in pixels, ratio of height-to-width, and ratio of width-to-height. These features depict characteristic shapes when a fall occurs, as in Figure 3. For example, the height feature in Figure 3(a) is a steady high for a few seconds, then decreases for two seconds or less, then is a steady low.

A feature sampled at a constant rate is a time series, as a time series is a sequence of data points. Dynamic Time Warping (DTW) is a proven time series comparison method [11] that determines how much one time series must bend to become another time series. The comparison method is based on the shape of the two time series. For example, an observed height time series (Figure 4(a)) can be compared to a characteristic height time series of a fall (Figure 4(b)). If the two time series were similar, then DTW would return a small value. A threshold value can be used to classify whether the observed time series belongs to the same class as the characteristic time series.

We used a binary tree for classification. An observed time series begins at the root of the binary tree, continues to be classified as the observed time series traverses the tree, and receives a label at a leaf node. There are two types of nodes in the binary tree: non-leaf nodes and leaf nodes. Each non-leaf node contains a time series and a threshold value. A non-leaf node's time series is compared to the observed time series using DTW. If the returned DTW value is greater than the non-leaf node's threshold, then the non-leaf node's right

Figure 4: Dynamic time warping finds the minimum amount that (a) one time series must bend to become (b) another time series. The angled lines between the time series show the extent of the bending.



child is traversed. Otherwise, the left child is traversed. Each leaf node contains a label, in our case either "fall" or "non-fall". The observed time series is assigned the label of the leaf node reached.

We used logical-shapelet analysis software [9] to train such a binary tree. A shapelet is a subsection of a time series, such as the highlighted section of the height feature in Figure 3(a). The software takes as input labeled time series, in our case "fall" or "non-fall", and a minimum and maximum shapelet size. The software outputs a binary tree used for classification.

C. Feature comparison

We compared four features based on the MBR around the foreground: height in pixels, width in pixels, ratio of height-to-width, and ratio of ratio of width-to-height.

We recorded 23 raw videos (12 with falls and 11 without falls) of a sole male twenty-six year old actor, using a living room webcam capturing at a rate of 15 frames per second. Each video is approximately one minute long. The fall videos included stumbling and slipping on the floor, and loosing balance from the couch while reaching for a lamp. The non-fall videos included sweeping the floor, napping and watching television on the couch, and searching for a lost item.

For each video, we gathered the MBR from each frame of the video using our foreground-background segmentation algorithm. We converted each video's MBR data to a time series of each feature. We labeled each time series as either "fall" or "non-fall."

We used the leave-one-out method to evaluate each feature's fall detection accuracy. Accuracy is measured as two numbers: sensitivity and specificity. Sensitivity is the ratio of correct fall detections over actual falls, e.g., if 11 falls were correctly detected but there were 12 total falls, sensitivity is 11/12 = 0.92. Specificity is the ratio of correct non-fall reports over actual non-falls, e.g., if 10 non-falls were reported but there were 11 non-falls, specificity is 10/11 = 0.91. In particular, we trained a binary tree classifier with the logical-shapelet analysis software using leave-one-out of the videos, a minimum shapelet size of 5 seconds and maximum shapelet size of 15 seconds. Then, we tested the binary tree classifier with the remaining video. We performed the leave-one-out and testing for each video.

The average sensitivity and specificity for each feature are shown in Table 1. The width of MBR in pixels had the

Table 1: Fall detection accuracy of the leave-one-out method applied to each feature of the raw video's MBR. Higher is better.

Feature	Average sensitivity	Average specificity
Height of MBR in pixels	0.31	0.30
Width of MBR in pixels	0.91	0.92
Height-to-width ratio of MBR	0.44	0.50
Width-to-height ratio of MBR	0.64	0.67

largest average sensitivity of 0.91 and the largest average specificity of 0.92. These results are not unexpected as many of the falls happened to the left or right from the camera's perspective, as opposed to falling toward or away from the camera.

III. PRIVACY ENHANCEMENTS

Cameras typically output raw video, which we define:

- *Raw video*, shown in Figure 5(a), is normal video that shows the camera's scene as clearly as possible.

A privacy-enhanced video intentionally obscures the appearance of a person in the video to protect that person's privacy. Raw video is perceived to have less privacy than privacy-enhanced video [4][6][16]. We consider four privacy enhancements:

- *Blur video* (Figure 5(b)) smears the video, typically restricting the smearing to the region with movement.
- *Silhouette video* (Figure 5(c)) replaces the movement with an outline of the person filled with a solid color.
- *Bounding-oval video* (Figure 5(d)) covers the movement with a bounding oval around each person.
- *Bounding-box video* (Figure 5(e)) covers the movement with a bounding box around each person.

We built a tool to convert raw video to privacy-enhanced video. The raw video was processed with our foreground-background segmentation algorithm to extract a foreground and the MBR around the foreground. The blur video blurred the region of the raw video in which the MBR resides. The silhouette video overlayed the foreground onto the background image. The bounding-oval video covered the region of the MBR with a solid blue oval that has the same height and width of the MBR. The bounding-box video covered the region of the MBR with a solid blue rectangle

Figure 5: Picture of the same moment of the same video as (a) raw, (b) blur, (c) silhouette, (d) bounding-oval and (e) bounding-box.

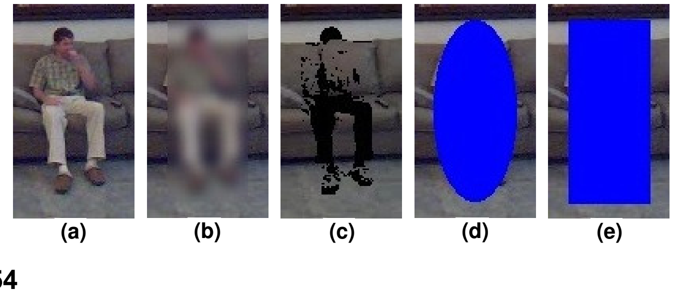


Table 2: Fall detection accuracy of the leave-one-out method using MBR width trained on raw video and tested on each video style. Higher is better.

Video style	Average sensitivity	Average specificity
Raw	0.91	0.92
Blur	1.00	0.67
Silhouette	0.91	0.75
Bounding-oval	0.91	0.92
Bounding-box	0.82	0.92

with the same height and width of the MBR.

IV. EXPERIMENT

We evaluated the sensitivity and specificity of our fall detection algorithm on privacy-enhanced video. We converted the 23 raw videos described in Section II.C into privacy-enhanced videos. For each raw video and privacy-enhanced video, we gathered the MBR from each frame of the video using our foreground-background segmentation algorithm. We converted each video's MBR data to a time series of width in pixels. We labeled each time series as either "fall" or "non-fall."

We trained a binary classifier in the logical-shapelet analysis software using the leave-one-out on the raw video's MBR width. Then, we tested the binary classifier with the remaining raw video's MBR width and the respective privacy-enhanced video's MBR width for each privacy enhancement. We performed the leave-one-out and testing for each video.

The average sensitivity and specificity are shown in Table 2. Among the privacy-enhanced videos, the bounding-oval had the highest average sensitivity of 0.91 and the highest average specificity of 0.92, which are identical results to the raw video. The bounding-box performed slightly worse in having misclassified one additional video over raw and bounding-oval. Blur and silhouette performed slightly worse still. The blur likely did worse because the color of the person and the color of the background are blurred together; therefore, the person is colored more like the background in blur video than raw video, which is harder for the foreground-background segmentation algorithm to handle. Blur may be improved by reducing the absolute difference threshold used to identify the foreground when subtracting a frame and the background image; however, this reduction may cause more noise in the foreground. The silhouette likely did worse because the black-colored silhouette would get confused for a shadow. Silhouette may be improved by making the color of the silhouette the same blue color as the bounding-oval and bounding-box.

V. CONCLUSION AND FUTURE WORK

A fall detection algorithm using a common approach involving features extracted from the MBR of foreground video achieves comparable sensitivity and specificity on both raw video and privacy-enhanced video. The bounding-oval privacy-enhanced video yielded the same sensitivity (0.91) and specificity (0.92) as raw video. Thus, the privacy enhancements described in this paper, except for blur video, may be applied prior to streaming the video to a local or remote computer that executes a fall detection algorithm, with small or no impact on fall detection accuracy.

In future work, we plan to compare our fall detection method to other methods using a larger video data set. We also plan to determine a user's privacy perception of raw video and each privacy-enhanced video, and determine a user's accuracy at manually detecting falls in raw video and each privacy-enhanced video.

REFERENCES

[1] Anderson, D., J.M. Keller, M. Skubic, X. Chen, and Z. He. Recognizing Falls from Silhouettes. Proceedings of the 28th IEEE EMBS Annual International Conference. New York City, USA, 2006.

[2] Beach, S., R. Schulz, K. Seelman, R. Cooper and E. Teodorski. Trade-Offs and Tipping Points in the Acceptance of Quality of Life Technologies: Results from a Survey of Manual and Power Wheelchair Users. International Symposium on Quality of Life Technology, 2011.

[3] Beach, S., R. Schulz, J. Downs, J. Mathews, B Barron and K. Seelman. Disability, Age, and Informational Privacy Attitudes in Quality of Life Technology Applications: Results from a National Web Survey. ACM Transactions on Accessible Computing. Volume 2 Issue 1, May 2009.

[4] Caine, K.E., A.D. Fisk and W.A. Rogers. Benefits and privacy concerns of a home equipped with a visual sensing system: A perspective from older adults. Proceedings of the Human Factors and Ergonomics Society, 50th Annual Meeting (pp. 180–184), 2006.

[5] Cucchiara, R., A. Prati and R. Vezzani. A Multi-Camera Vision System for Fall Detection and Alarm Generation. Expert Systems, Volume 24, Issue 5, pgs 334-345, November 2007.

[6] Demiris, G., M.J. Rantz, M.A. Aud, K. D. Marek, H.W. Tyrer and M. Skubic, A.A. Hussam. Older adults' attitudes towards and perceptions of 'smart home' technologies: a pilot study. Medical Informatics and The Internet in Medicine, 2004.

[7] Kim, K. T.h. Chalidabhongse, D. Harwood and L. Davis. Real-Time Foreground-Background Segmentation using Codebook Model. Real-Time Imaging, Volume 11, Issue 3, June 2005, pgs. 172-185.

[8] Miaou, S.-G., P.-H. Sung and C.-Y. Huang. A Customized Human Fall Detection System Using Omni-Camera Images and Personal Information. Proceedings of the 1st Distributed Diagnosis and Home Healthcare Conference, 2006.

[9] Mueen, A., E. Keogh and N. Young. Logical-shapelets: An Expressive Primitive for Time Series Classification. Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2011.

[10] OpenCV. http://opencv.willowgarage.com/wiki/. February 2012.

[11] Ratanamahatana, C. A. and E. Keogh. Three Myths about Dynamic Time Warping. In proceedings of SIAM International Conference on Data Mining, 2005.

[12] Rougier, C., J. Meunier, A. St-Arnaud and J. Rousseau. Procrustes Shape Analysis for Fall Detection. The Eighth International Workshop on Visual Surveillance, 2008.

[13] Stauffer C, Grimson WEL. Adaptive background mixture models for real-time tracking. IEEE International Conference on Computer Vision and Pattern Recognition 1999; 2 : pg. 246–52.

[14] Thome, N., S. Miguet and S. Ambellouis. A Real-Time, Multiview Fall Detection System: A LHMM-Based Approach. IEEE Transactions on Circuits and Systesm for Video Technology, Vol. 18, No. 11, November 2008.

[15] Williams, A., D. Ganesan and A. Hanson. Aging in Place: Fall Detection and Localization in a Distributed Smart Camera Network. Proceedings of the 15th International Conference on Multimedia, 2007.

[16] Zhao, Q. A. and J. T. Stasko. Evaluating Image Filtering Based Techniques in Media Space Applications. Proceedings of ACM conference on Computer Supported Cooperative Work, 1998.