# Experimental Evaluation of Keypoints Detector and Descriptor Algorithms for Indoors Person Localization

Velislava Grishina Spasova

*Abstract* - **This paper presents an experimental evaluation of the accuracy and runtime of several combinations of algorithms for keypoints detection and description. The algorithms are used to localize a person in a room as the first step of a computer vision based fall detection system, part of an Ambient assisted living (AAL) solution.**

*Keywords* – **Keypoints Detector, Keypoints Descriptor, Person Localization, Ambient Assisted Living, Computer Vision**

## I. INTRODUCTION

The gentrification of developed societies is the main reason for the increasing attention from the research community to Ambient Assisted Living (AAL) which is a relatively new and rapidly expanding area of research. The main objective in front of AAL is to use the most recent advances in technology in order to intelligently assist the elderly and people with disabilities and to help them to live a longer, better and independent life [1].

Falls in the elderly are a major health risk and as such automatic fall detection is among the primary tasks in front of AAL systems. There are various approaches to fall detection – wearable sensors based, ambient sensors based and computer-vision based fall detection [2]. Computer vision based fall detection is getting more focus in recent years but due to the complicated nature of computer vision and the wide variety of used approaches and algorithms, there isn't a single and best solution [3,4].

This paper briefly presents our approach to fall detection and further illustrates the results from an experimental evaluation of a group of computer vision keypoints detector and descriptor algorithms used in the first step of the proposed solution.

The rest of the paper is organized as follows: Section II presents our approach to fall detection; Section III gives a brief overview of the evaluated algorithms; Section IV presents the results from the experiments; finally, Section V concludes the paper.

## II. MULTIMODAL FALL DETECTION

As previously mentioned fall detection is both critical in terms of required accuracy and complex in terms of choices of possible approaches and their implementations. As such,

V.Spasova is with the Department of Electronics and Electronics Technologies, Faculty of Electronic Engineering and Technologies, Technical University - Sofia, 8 Kliment Ohridski blvd., 1000 Sofia, Bulgaria, e-mail: vgs@tu-plovdiv.bg

there isn't a single best approach – all solutions have positive and negative aspects.

Our solution is based on the idea that we could merge two distinct research directions such as wearable sensors based fall detection and computer vision based fall detection into a single multimodal approach.

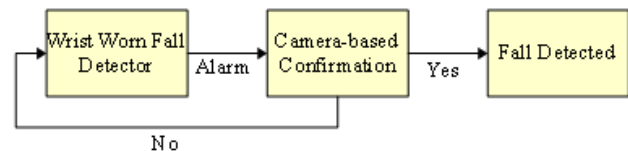Our system is composed of two modules which are presented at Figure 1.



Fig. 1. Architecture of fall detector

The wearable fall detector has an integrated 3 axes accelerometer which is used to detect falls monitoring the total sum vector of the acceleration to the ground of the three axes. This system has high sensitivity (all falls are detected) but low specificity (a lot of false positives). Experimental results from this module have been published in another paper [5].

The second module is computer vision based and is used in order to verify the thrustworthiness of a fall alarm generated by the wearable module. Its organization is presented at Figure 2.
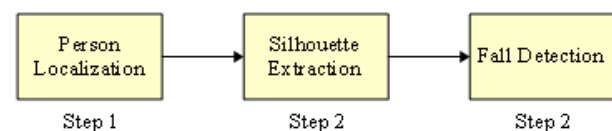


Fig. 2. Camera-based fall detection

The camera captures pictures of every 10 degrees of its field of view (180 degres) at regular intervals throughout the day. These images are the background images which are stored by the module. The first step upon arrival of a hypothetical fall alarm from the wearable module is to localize the person in the room. This is done as the current picture (with the person) is matched against the set of background images. The second step is to extract the silhouette of the person from the pair of images (current image + its corresponding background image). This silhouette is fed into the third step to detect whether the person has been fallen.

In this paper we present evaluation of several algorithms which are used to match the images in the first step of this module.

## III. Evaluated Algorithms

Keypoints, also named features, are used to characterize interesting points in an image. Most often these interesting points are corners. Keypoints are often used in object detection – i.e. it is given an image of the object and the algorithms should detect it in a set of other images.

In literature there are papers presenting experimental analyses of keypoints detectors and descriptors [6]. However, all these papers focus on the traditional case of object detection. In this paper we are using the keypoints to detect the **background** of the image instead of an object in the image.

### A. Algorithms

All algorithms' implementations are provided through the OpenCV library [7]. Five keypoints detectors and six keypoints descriptors have been tested. All combinations of detectors and descriptors are tested for two matchers – BruteForce matcher and approximate nearest neighbors search based matcher which uses the FLANN (Fast Library for Approximate Nearest Neighbors) library [8].

The tested algorithms are:

- SIFT (Scale-Invariant Feature Transform) keypoints detector and descriptor [9];
- SURF (Speeded-Up Robust Features) keypoints detector and descriptor [10];
- FAST (Features from Accelerated Segment Test) keypoints detector [11];
- BRISK (Binary Robust Invariant Scalable Features) keypoints detector and descriptor [12];
- BRIEF (Binary Robust Independent Elementary Features) keypoints descriptor [13];
- ORB (Oriented FAST and Rotated BRIEF) keypoints detector and descriptor [14];
- FREAK (Fast REtinA Keypoint) keypoints descriptor [15].

### B. Evaluation Methodology

All combinations of detectors, descriptors and matchers have been tested with the exception of the combination of SIFT+ORB for both matchers because due to a bug in the OpenCV implementation it is impossible to combine a SIFT detector with an ORB descriptor (the opposite is a valid combination).

Four images with a person on each of them are matched against all background images (19 background images taken every 10 degrees from 0 degrees to 180 degrees) for all the combinations resulting in 76 runs for each combination. Each match between two images gives as a result a number of matched keypoints called "inliers". The correct correspondence, i.e. the background image that corresponds to the background of the image with person, should have the maximum number of inliers. Also, the total runtime for each image match has been recorded and the average runtime for all runs of a given combination has been calculated. The results of the experiments are summarized in the next section.

## IV. Results

After all the runs for all the combinations, the results have been averaged and have been ranked with accuracy mark which reflects how accurate is the correspondence between the current image and its true background image. The accuracy goes from 1 to 5, 1 corresponding to very bad accuracy, 5 meaning very good accuracy. Some statistics for the good (meaning with good accuracy) combinations are presented in Table 1. The other combinations don't have good accuracy and have been omitted here due to a lack of space. All the tests have been run on platform Intel Atom, CPU D410, 1.67 GHz, 1.49 GB RAM under Ubuntu 10.04.

TABLE 1. ALGORITHMS WITH GOOD ACCURACY

| Detector | Descriptor | Accuracy | BF [sec] | FLANN, [sec] |
|---|---|---|---|---|
| FAST (2231) | BRIEF | 5 | 153.24 | 725.87 |
|  | BRISK | 4 | 327.45 | 722.2 |
|  | ORB | 4 | 149.85 | 689.07 |
|  | SIFT | 4 | 754.19 | 112.12 |
| SURF (1002) | BRIEF | 4 | 28.18 | 144.02 |
|  | BRISK | 4 | 43.15 | 208 |
|  | ORB | 4 | 27.93 | 114.05 |
|  | SIFT | 4 | 154.72 | 71.86 |
|  | SURF | 4 | 70.72 | 30.11 |

As it could be seen from the table, the best combination in terms of accuracy is FAST + BRIEF. However, it is rather slow at more than 2 min 30 sec for BruteForce matcher and more than 12 min for FLANN based matcher. This is mostly due to the very big number of keypoints, detected by FAST (the number in parenthesis in the first column of the table) which slows down the match process. As this step is part of a bigger algorithm which is running in real time, some balance between accuracy and runtime should be achieved. Bearing the real time limitation, it seems that the combinations SURB + BRIEF + BruteForce, SURF + ORB + BruteForce, and SURF + SURF + FLANN matcher are more promising for our project.

The plot "degrees to number of inliers" for the combination FAST + BRIEF + BruteForce for Person1.jpg is presented at Figure 3. It could be seen that the peak of the number of inliers at 90 degrees corresponds to the correct background for the image Person1.jpg.

Figure 4 presents the visualization of the matched keypoints for the same combination for 90 degrees (correct correspondence) and Figure 5 – for 60 degrees (incorrect correspondence).
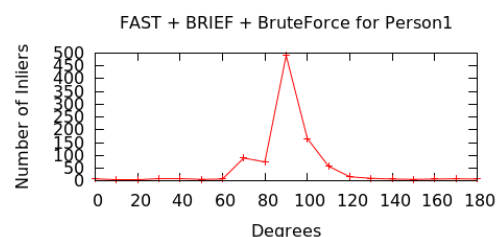


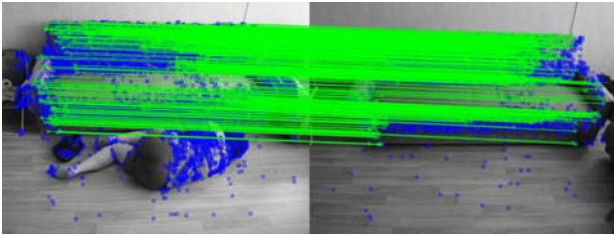Fig. 3. FAST + BRIEF + BruteForce accuracy

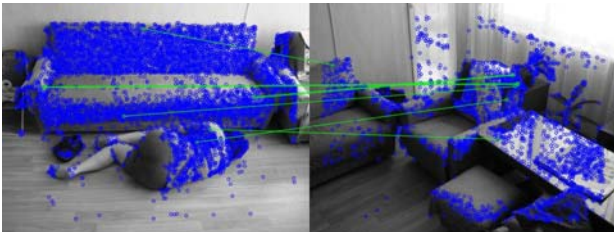Fig. 4. FAST + BRIEF + BruteForce for correct correspondence (90 degrees)



Fig. 5. FAST + BRIEF + BruteForce for incorrect correspondence (60 degrees)

## V. CONCLUSION AND FUTURE WORK

In this paper we have presented the experimental evaluation of several combinations of keypoints detector, descriptor and matcher algorithms which are to be used in a computer vision module for fall detection, part of an AAL system. The algorithms will be used to localize the user of the system in the room, i.e. to match an image with a person to the right background image, which is the opposite of the typical use of this kind of algorithms - object recognition.

The algorithms have been evaluated according to the accuracy of the match between the image with person and the correct background image, and according to their runtime. The experiments have proven that certain combinations of algorithms such as FAST + BRIEF + BruteForce or SURF + SURF + FLANN are viable solutions to the aforementioned task.

Future work will be concentrated on optimizing the runtime for the good combinations in order to be used in real time systems such as fall detection. Also, more experiments are needed to test how the accuracy and runtime change with change of the size of the images and with the addition of noise to the images.

## VI. ACKNOWLEDGEMENT

## REFERENCES

[1] D. Cook, J.C. Augusto, V.R. Jakkula. Ambient Intelligence: Technologies, Applications, and Opportunities, Journal of Pervasive and Mobile Computing, August 2009, Vol. 5, Issue 4, pp 277-298.
[2] M. Mubashir, L. Shao, L. Seed. A survey on fall detection: Principles and approaches, Journal of Neurocomputing, 2013, Vol. 100, pp 144-152.
[3] F. Cardinaux, D. Bhowmik, C. Abhayaratne, M.S. Hawley. Video based technology for ambient assisted living: A review of the literature, Journal of Ambient Intelligence and Smart Environments , 2011, Vol. 3, pp 253-269.
[4] D. Weinland, R. Ronfard, E. Boyer. A Survey of Vision-Based Methods for Action Representation, Segmentation and Recognition, Journal of Computer Vision and Image Understanding, February 2011, Vol. 115, Issue 2, pp 224-241.
[5] I Iliev, S Tabakov, I Dotsinsky. Two steps approach for falls detection in the elderly, Annual Journal of Electronics, Vol. 5, Issue 2, 46-48, 2011.
[6] O. Miksik, K. Mikolajczyk. Evaluation of local detectors and descriptors for fast feature matching, Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), pp 2681 – 2684, 2012.
[7] http://www.opencv.org
[8] http://www.cs.ubc.ca/research/flann/
[9] D.G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, 2004, Vol. 60, Issue 2, pp 91-110.
[10] H. Bay, T. Tuytelaars, L. van Gool. SURF - Speeded Up Robust Features, Proceedings of the 9th European Conference on Computer Vision (ECCV2006), pp 404-417, 2006.
[11] E. Rosten, T. Drummond. Machine learning for high-speed corner detection, Proceedings of the 9th European Conference on Computer Vision (ECCV2006), pp 430-443, 2006.
[12] S. Leutenegger, M. Chli, R.Y. Seigwart. BRISK - Binary Robust Invariant Scalable Keypoints", Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV2011), pp 2548-2555, 2011.
[13] M. Calonder, V. Lepetit, C. Strecha, P. Fua. BRIEF - Binary Robust Independent Elementary Features, Proceedings of the 11th European Conference on Computer Vision (ECCV2010), pp 778-792, 2010.
[14] E. Rublee, V. Rabaud, K. Konolige, G. Bradski. ORB - An Efficient Alternative to SIFT or SURF, Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV2011), pp 2564-2571, 2011
[15] A. Alahi, R. Ortiz, P. Vandergheynst. FREAK - Fast Retina Keypoint, Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp 510-517, 2012.