Sensors fusion for head tracking using Particle filter in a context of falls detection

Imen Halima, Jean-Marc Laferté, Geoffroy Cormier, Alain-Jerôme Fougère, Jean-Louis Dillenseger

SENSORS FUSION FOR HEAD TRACKING USING PARTICLE FILTER IN A CONTEXT OF FALLS DETECTION

I. Halima[1,2] *J.M. Laferté*[1] *G. Cormier*[3] *A.J. Fougère*[1] *J.L. Dillenseger*[2]

[1] ECAM Rennes, Campus Ker Lann, Bruz, 35091 Rennes, France
[2] INSERM, U1099, Rennes, F-35000, France ; Université de Rennes 1, LTSI, Rennes, F-35000, France
[3] Neotec Vision ltd, F-35740 Pacé, France

ABSTRACT

In the context of ageing societies, assessing risk factors and detecting falls for the elderly is becoming a crucial issue. In this paper, we propose an iterative head tracking method based on particle filtering using the fusion of low cost thermal and depth sensors for home environments whilst preserving privacy. The iteration process begins by segmenting the head in the depth image to calculate the depth coefficients and the thermal coefficients used for updating the particle weights. The method was tested on several sequences, with or without depth-thermal fusion: results show its robustness and accuracy, and also demonstrate that fusion improves tracking, namely when fast motion occurs (in case of a fall for instance) or when segmentation is erroneous.

Index Terms— head tracking, sensor fusion, particle filter, thermal sensor, depth sensor, fall detection.

1. INTRODUCTION

Due to the ageing population and the fact that falls are the second cause of accidental deaths worldwide (according to WHO[1], providing efficient fall detection systems of elderly is becoming crucial.

In previous work [1], we proposed a visual system solution based on the collaboration between a depth sensor and a low resolution thermal sensor to detect elderly falls. In the first study, we developed a fall detection system based only on the depth sensor. This system has been tested in 2 senior citizens' homes during 1 year but has produced too many false alarms caused mainly by occlusions and lack of shape tracking. For these reasons, we chose to add a thermal sensor. In this paper, we aim to develop a person tracking algorithm in order to improve the accuracy and the sensitivity of the system proposed in [1] and to reduce the number of false alarm.

This tracker will also be used to estimate the people's trajectories and to analyze their activities in order to prevent falls. This system is low cost, works day and night, can be easily installed in a room and moreover preserves privacy.

[1]World Health Organization (WHO)

In our framework, instead of tracking the whole body, we chose to solely track the head because it is non-deformable, the hottest, highest and least hidden part of the body which can easily be approximate as a ellipse with only few parameters. The motion head is also a significant marker for the fall detection.

Among tracking methods, particle filtering (PF)-based ones have proven to be very flexible and to more accurately model the dynamics of the object motion [2]. PF is often applied for tracking in color images. In this paper we present an adaptation of PF to track people using both depth and thermal information.

In general, when the tracker is based only on a signal feature, the result can be wrong due to the complexity of the environment and the process of tracking [3]. In order to improve the tracking algorithm, multi modal data can be used to increase the performance. The main contributions of our paper are the using of different observations from the 2 sensors and the form to integrate these measures which increases significantly the performance of the tracking method.

The rest of the paper is organized as follows. In section 2, we define calibration inter sensor and head segmentation steps. In section3, we illustrate particle filter algorithm on depth and thermal images. In section 4, we report the experimental results and provide some discussions. Finally, we summarize our paper and point out some further directions in section 5.

2. CALIBRATION INTER SENSORS AND HEAD SEGMENTATION METHOD

2.1. Depth and thermal calibration

Our system is based on a thermal sensor (FLIR lepton 2.5, Focal length: $5\,\mathrm{mm}$, Thermal Horizontal Field of View $THFOV$: 51°, Thermal Vertical Field of View T_{VFOV}: 37.83°, T_{Xres}: 80 pixels and T_{Yres}: 60 pixels) and a depth camera (Microsoft Kinect, Focal length: $6.1\,\mathrm{mm}$, Depth Horizontal Field of View D_{HFOV}: 58°, Depth Vertical Field of View D_{VFOV}: 45°, D_{Xres}: 640 and D_{Yres}: 480) which are aligned horizontally (see Figure 1).

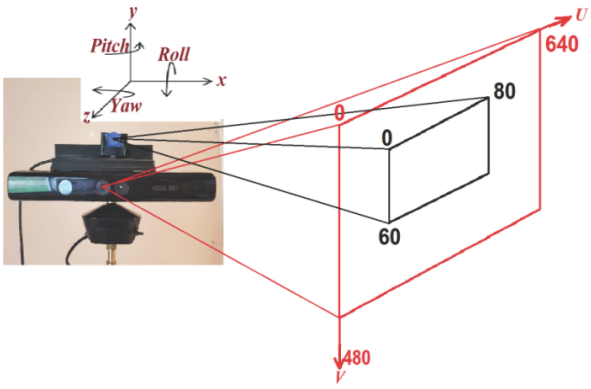

Fig. 1. Sensors coordinate systems

In order to match a point from depth image with a point in thermal image, a calibration step is required to calculate the transformation parameters (see Figure 2).

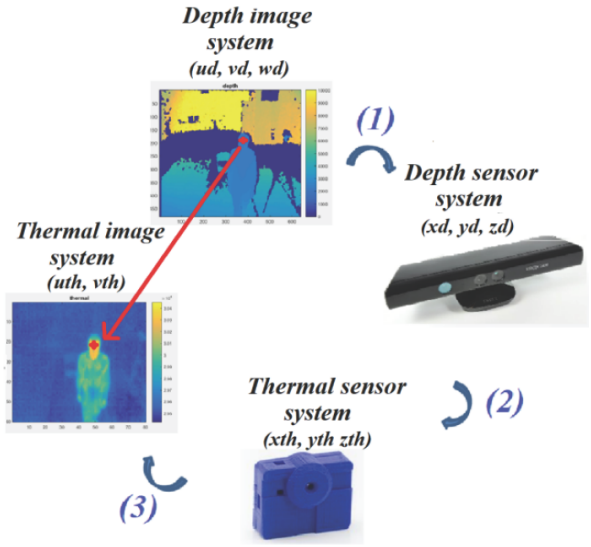

Fig. 2. Calibration system

As input we have depth and thermal images. u_i, v_i are the pixel coordinates in the images and w_d the depth information (i.e. the distance of the object to the depth optical center). The estimation of the relationship between these 2 coordinate systems needs 3 steps (see Figure 2): 1) The estimation of the transform of the depth image (u_d, v_d, w_d) to the coordinate system (x_d, y_d, z_d) of the depth sensor. This can be done analytically from the intrinsic parameters of the depth camera:

$$\begin{cases} x_d = \left(u_d - \frac{D_i}{2}\right) \frac{2w_d \tan\left(\frac{D_{HFOV}}{2}\right)}{D_{X_{Res}}} \\ y_d = \left(v_d - \frac{D_i}{2}\right) \frac{2w_d \tan\left(\frac{D_{VFOV}}{2}\right)}{D_i} \\ z_d = w_d \end{cases} \quad (1)$$

2) the transformation between the coordinate system (x_d, y_d, z_d) of the depth sensor to this (x_{th}, y_{th}, z_{th}), of the thermal sensor. It can be obtained from the extrinsic parameters, in our case a rotation and a translation matrix:

$$\begin{pmatrix} x_{th} \\ y_{th} \\ z_{th} \end{pmatrix} = T + R \begin{pmatrix} x_d \\ y_d \\ z_d \end{pmatrix} \quad (2)$$

$$R = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix} * \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \quad \text{and } T = \begin{pmatrix} d_x \\ d_y \\ d_z \end{pmatrix}$$

α, θ and β are the roll, pitch and yaw angles [1]. And 3) the transformation between the coordinate system (x_{th}, y_{th}, z_{th}) of the thermal sensor to the thermal image coordinates (u_{th}, v_{th}). This can be done analytically from the intrinsic parameters of the thermal camera:

$$\begin{cases} u_{th} = \frac{T_{X_{Res}}}{2z_{th} \tan\left(\frac{T_{HFOV}}{2}\right)} x_{th} + \frac{T_{X_{Res}}}{2} \\ v_{th} = -\frac{T_{Y_{Res}}}{2z_{th} \tan\left(\frac{T_{VFOV}}{2}\right)} y_{th} + \frac{T_{Y_{Res}}}{2} \end{cases} \quad (3)$$

In our case, the intrinsic parameters are the values given by the constructor. So the purpose of the calibration is to estimate 6 parameters of equation (2) and then to generate a one to one pixel correspondence of the depth and thermal images.

2.2. Segmentation

In order to track the head , we need to extract the head position from the image. After calibration, we extract this position according to segmentation stage using the following framework (Figure 3). The first step is to create a reference depth map by averaging the N first depth images without any moving objects. This defines the static background of the scene. From each frame at time $N + t$, we subtract the static background from this frame and detect so the moving objects we called as foreground. Then, we filter the foreground noise and we extract the silhouette and model it with an ellipse. Finally, the head size and pose are estimated from the silhouette ellipse. The head is modeled as a smaller ellipse with the same orientation as the silhouette ellipse but 3 time smaller. The center C of the head ellipse is set at 1/3 of the major axis from the upper part B (see Figure 4): $SC = 2/3SB$ with S the silhouette ellipse center.

We define the depth head position C from this ellipse as another ellipse located at the 1/3 of the upper part where $\vec{SC} = \frac{2}{3}\vec{SB}$ is the silhouette center and B is the upper point of silhouette. The head size and pose are extracted from the silhouette ellipse. They have the same orientation but the major axe of the head ellipse is 1/3 of silhouette major axis (see

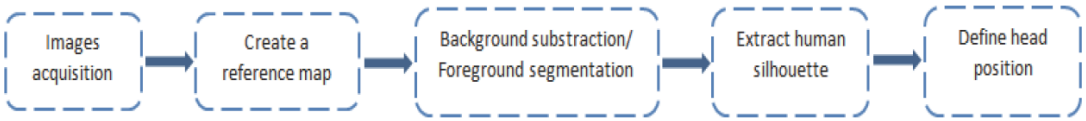

Fig. 3. Segmentation framework

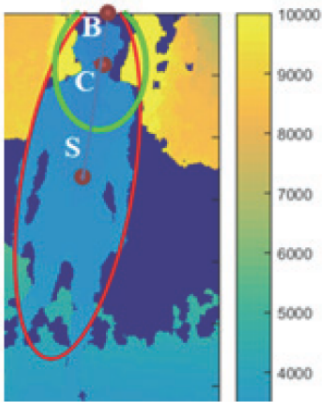

Fig. 4. Head position

Figure 4). In thermal image, we apply the matching to get the thermal head position.

3. PARTICLE FILTER ALGORITHM ON DEPTH AND THERMAL IMAGES

The PF tracking process is based on a hidden state vector x_t which is defined by the center, orientation and the size of the ellipse around the head in the depth image. x_t is predicted from x_{t-1} and the observation vectors $Z_t = \{z_1, \cdots, z_t\}$, and $H_t = \{h_1, \cdots, h_t\}$ obtained from depth and thermal sensors, respectively. PF uses a sample of N particles $S_t = \left\{S_t^1, \cdots, S_t^N\right\}$ to approximate the conditional probability $p(x_t/Z_t, H_t)$. Each particle S_t^n can be seen as a hypothesis aboutx_t (an ellipse model) and is weighted by $\pi_t(n)$ which are normalized. Particles are resampled according to their weights and are updated according to new observations new_{obsv} [3].

3.1. Tracking method

For each frame (time step t) t, we have x_t which represent the head. We sample this vector on N particles. Each particle S_t^n is defined by the same parameters of (x_t) and has a weight $\pi_t(n)$. Next, we predict x_{t+1} according to the propagation of particles based on:

$$S_t^n = AS_{t-1}^n + w_t \tag{4}$$

Where A is the transition model matrix and w_t is a Gaussian noise. Finally, we update the particle weight according to observation vectors where we combine depth and thermal information in new_{obsv} and resample particles to prevent the problem of particles degeneration (see figure 5) [4].

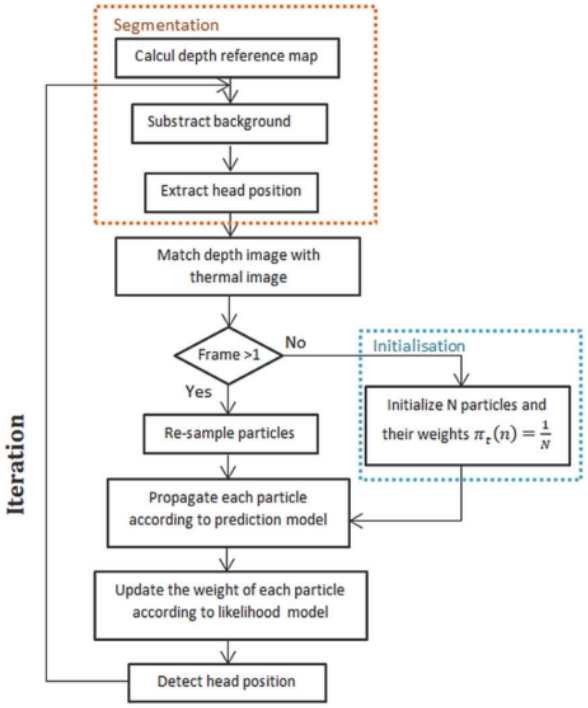

Fig. 5. Tracking method flowchart

Thus, the steps of iterative PF tracking algorithm are:

1. *Initialization*: Generate a sample of N particles $S_1 = \left\{S_1^1, \cdots, S_1^N\right\}$ based on the probability of the state vector $p(x_1)$, and initialize the weight of each particle by $1/N$.

2. *Resampling*: Re-sample particles to prevent the problem of particles degeneration, if frame > 1.

3. *Prediction*: Propagate particles according to prediction model to predict the state vector x_t.

4. *Updating*: Update the particle weight at frame t

$$\pi_t(n) = \frac{1}{\sqrt{2\pi}\sigma} \exp(new_{obsv}(n)/2\sigma^2) \tag{5}$$

where σ was fixed empirically.

Test	α	β	γ
C1	1/3	1/3	1/3
C2	1/4	1/2	1/4
C3	1/4	1/2	1/2
C4	1/2	1/4	1/4
C5	3/8	1/4	3/8
C6	3/8	3/8	1/4

Table 1. Importance factor IF values

Then normalize the weight:

$$\pi_t(n) = \pi_t(n) / \sum_{k=1}^{N} \pi_t(k) \tag{6}$$

and return to step 2.

3.2. Depth-thermal fusion

Updating particle weights is a key point of PF and is specific for each application (see [5] for color information).

In our framework, the weights are computed from the following three coefficients which combine depth and thermal information: C_d a distance coefficient, C_{g_d} a depth gradient coefficient and $C_{g_{th}}$ a thermal gradient coefficient. For each particle, C_d is the distance between the center of the particle and the center of the segmented head in depth image. C_{gd} and $C_{g_{th}}$ are based on the gradients along the head ellipse in the depth image and the thermal image respectively along the ellipse, as inspired by [6].

In this work, we considered two models to combine coefficients and use them in equation (5) for the updating step. Comparisons between these models are given in subsequent sections of this paper.

The first weighing model (M1) uses only 2 coefficients (C_d and $C_{g_{th}}$):

$$C_{fusion}(n) = \alpha C_d + (1-\alpha) C_{g_{th}} \tag{7}$$

α is an Importance Factors (IF) and n is the particle number.

The second weighing (M2) model combines all the 3 coefficients:

$$C_{fusion}(n) = \alpha C_d + \beta C_{g_d} + \gamma C_{g_{th}} \tag{8}$$

α and β are the IF of depth information and $\gamma = 1 - \alpha - \beta$ is the IF of thermal information. In order to estimate the impact of each coefficient in the particle filter, we tested several combination of static IF at each test (see table 1).

Particle weight is updated according to equation (5) by replacing new_{obsv} by the weighting model:

$$\pi_t(n) = \frac{1}{\sqrt{2\pi}\sigma} \exp(C_{fusion}(n)/2\sigma^2) \tag{9}$$

4. EXPERIMENTAL RESULTS

In this section, we demonstrate the performance of the proposed algorithm. We have performed several sequences of people moving in a room with co-calibrated static depth and thermal cameras which were fixed in the ceiling. We have tested our system with the following objectives: (1) compare our proposal work with segmentation only and depth tracking method, (2) evaluate the performance of the fusion algorithm, (3) evalute each weighing model, and (4) compare IF values.

In all tests, we used the following parameters: $N = 1000$ particles, $\sigma = 0.25$ and transition model matrix $A = I^4$. We fixed the acquisition frequency to 8 Hz.

The ground truth (GT) was established manually by setting an ellipse on each frame (white ellipse in Figure 6, 7) and the processing was performed using Matlab on Intel(R) Core(TM) i7-6700HQ CPU, 2.6 GHz.

We used 2 quantitative measurement to asses the accuracy of the specific versions: the localization error which is defined as the average Euclidean distance between the center locations of the tracked targets and the manually labeled ground truths, and the overlap score which is the overlap of the ground truth area and the tracking area [7].

Figure 6 illustrates the results of a) The head segmentation only, b) The depth version, c) The M1 model. Comparing these results, we can see that the head segmentation is totally wrong, the depth version estimates an area bigger than the head and the fusion model is able to track the head more accurately. These results shows that segmentation is erroneous in critical situations and depth sensor is usless on its own.

Figure 7 indicates the results of a) The M1 model and b) The M2 model. Visually M2 provide the closest pose to GT. In (Figure 8), the evaluation results show that the fusion of 3 coefficients provide the most accurate results. As expect, considering 3 coefficients together gives better results than using only two coefficients.

As illustrated on Figure 9, a comparator between the 6 tests performed with the second model (M2) that shows the impact of the coefficient IF to estimate the new head position.

When we evaluate these results using the 2 quantitative measurement, we observe in (Figure 10) a clear difference between the performance of the C4 test that assign more importance to distance coefficient and C2 that assigns more importance to depth gradient coefficient. For instance, when the person is walking in front of something, the depth gradient coefficient decreases the results and the distance coefficient gets the results better.

5. CONCLUSIONS

In this paper, we have presented a head algorithm based on particle filters that fuse the information of depth and thermal of a person in an indoor environment using the position, the orientation and the size of the ellipse around the head. The

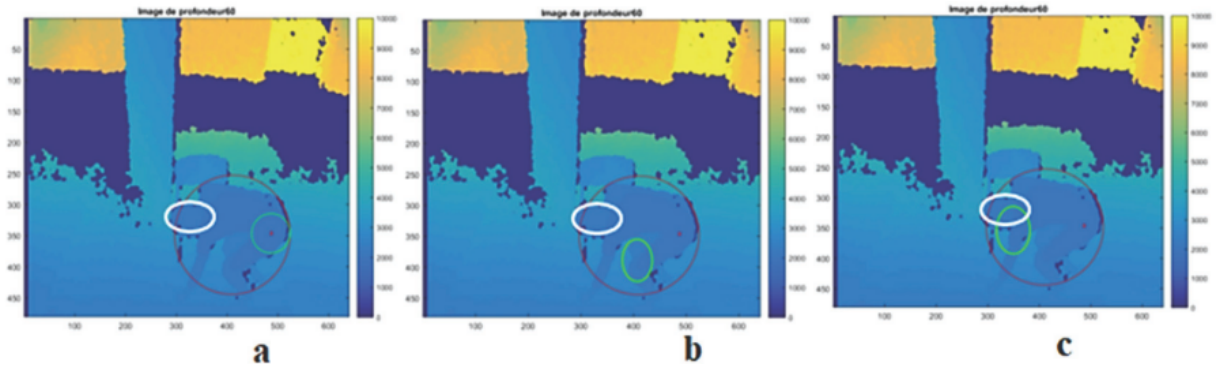

Fig. 6. Tracking results on one frame of a sequence: a) Segmentation only; b) Depth version; c) M1 model. Tracking results are in green, silhouette ellipse is red and GT ellipse is white.

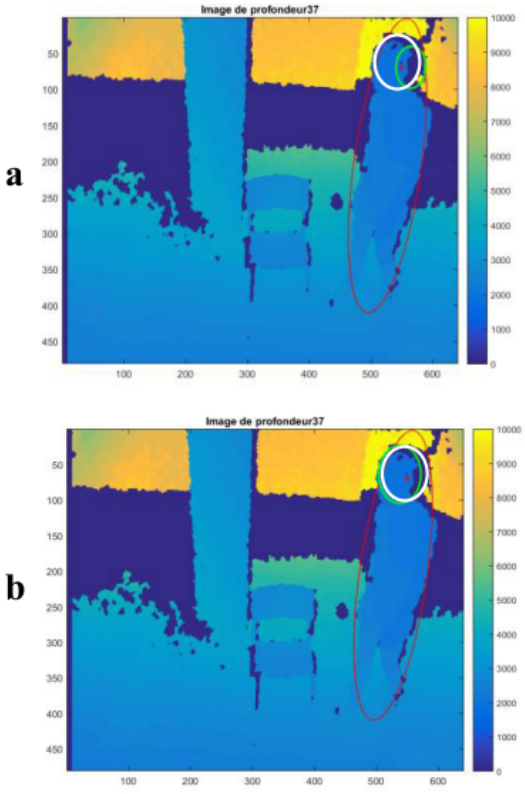

Fig. 7. Tracking comparison results between a) M1 model and b) M2 model. Tracking results are in green, silhouette ellipse is red and GT ellipse is white.

scene was acquired by two co-calibrated depth and thermal cameras. We have presented calibration system that can be implemented on any depth thermal system.

We have evaluted our proposal work in several situations with different models and compared it with other methods. Moreover, results have shown that our system gave the most accurate tracking results even in critical situations with very low resolution images. The combination of information pro-

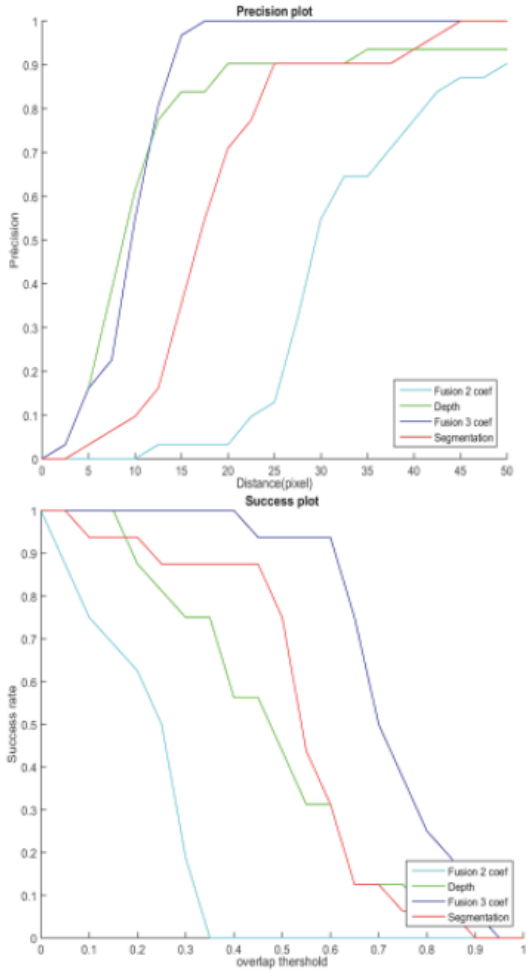

Fig. 8. Quantitative measurements over a sequence. Localization error (a) and the overlap score (b) using the segmentation (red), the depth version (green) the M1 model (cyan) and the M2 model (blue).

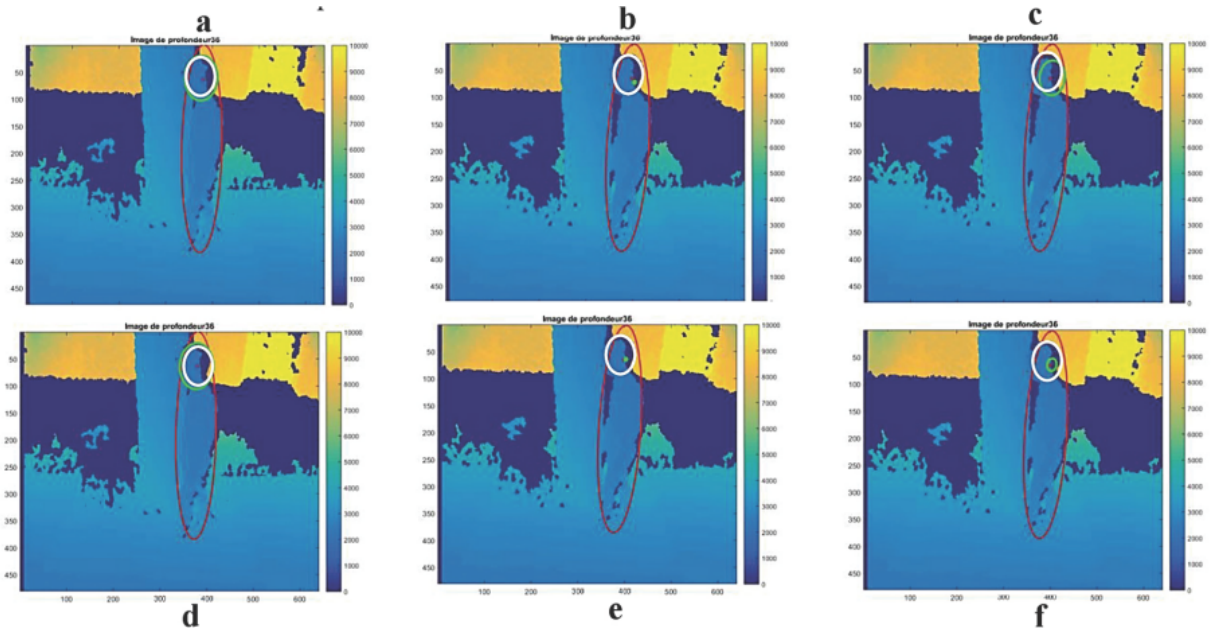

Fig. 9. Tracking results of 6 IF tests a) C1 test; b) C2 test; c) C3 test; d) C4 test; e) C5 test and f) C6 test. Tracking results are in green, silhouette ellipse is red and GT ellipse is white.

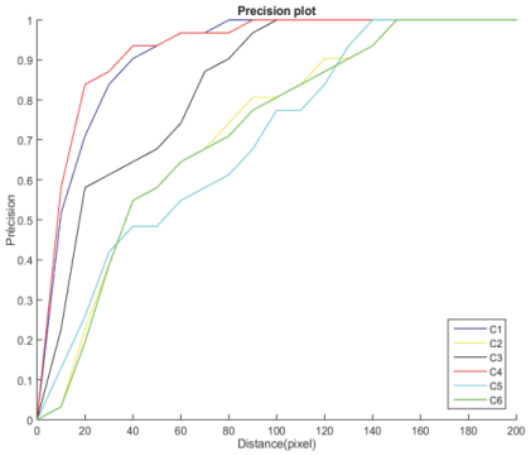

Fig. 10. Quantitative IF measurements over a sequence. Localization error of C1 test (blue), C2 test (yellow), C3 test (black), C4 test (red), C5 test (cyan) and C6 test (green).

vided by both cameras improve the tracking. For future work we plan: 1) to modify the constant importance factors α, β and γ to dynamic values according to the influence of each coefficient, 2) to make an automatic GT with Vicon Systems in living labs, 3) to compare the method with other robust methods like Deep Learning based methods and 4) to add the velocity in the state vector.

Acknowledgements

This work is funded under the PRuDENCE project (ANR-16-CE19-0015-02) which has been supported by the French National Research Agency.

6. REFERENCES

[1] G. Cormier, *Analyse statique et dynamique de cartes de profondeurs : application au suivi des personnes à risque sur leur lieu de vie*, Ph.D. thesis, Université de Rennes 1, 2015.

[2] E. Erdem, S. Dubuisson, and I. Bloch, "Visual tracking by fusing multiple cues with context-sensitive reliabilities," *Pattern Recognition*, vol. 45, no. 5, pp. 1948–1959, 2012.

[3] M. Isard and A. Blake, "Condensation-conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[4] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[5] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image and vision computing*, vol. 21, no. 1, pp. 99–110, 2003.

[6] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "3D head tracking for fall detection using a single calibrated camera," *Image and Vision Computing*, vol. 31, no. 3, pp. 246–254, 2013.

[7] Y. Wu, J. Lim, and M. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2013, pp. 2411–2418.