Depth and thermal information fusion for head tracking using particle filter in a fall detection context

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

AI Communications 0 (0) 1
IOS Press

Depth and thermal information fusion for head tracking using particle filter in a fall detection context

Imen Halima [a,b], Jean-Marc Laferté [a], Geoffroy Cormier [c], Alain-Jérôme Fougères [a,*] and Jean-Louis Dillenseger [b]

[a] *ECAM Rennes, Campus Ker Lann, Bruz, 35091 Rennes, France*
E-mails: imen.halima@ecam-rennes.fr, jean-marc.laferte@ecam-rennes.fr, alain-jerome.fougeres@ecam-rennes.fr
[b] *Inserm, LTSI - UMR 1099, University of Rennes, F-35000 Rennes, France*
E-mail: jean-louis.dillenseger@univ-rennes1.fr
[c] *Neotec Vision, 35740 Pacé, France*
E-mail: geoffroy.cormier@neotec-vision.com

Abstract. The security of elderly people living alone is a major issue. A system that detects anomalies can be useful for both individual and retirement homes. In this paper, we present an adaptive human tracking method built on particle filter, using depth and thermal information based on the velocity and the position of the head. The main contribution of this paper is the fusion of information to improve tracking. For each frame, there is a new combination of coefficients for each particle based on an adaptive weighting. Results show that the tracking method can deal with the cases of fast motion (fall), partial occultation and scale variation. To assess the impact of fusion on the tracking process, the robustness and accuracy of the method are tested on a variety of challenging scenarios with or without depth-thermal fusion.

Keywords: head tracking, sensor fusion, particle filter, thermal sensor, depth sensor

1. Introduction

According to the French institute of health education (INPES), 9,300 people die each year from falls. These falls occur mainly at home (78% of falls) and especially at night (60%) causing physical and psychological consequences. In accordance with the World Health Organization (WHO), falling is the second leading cause of accidental or unintentional injury deaths worldwide [1]. For these reasons, an automatic system that could prevent and detect falls and call emergency services can be useful even for retirement homes. Actually, many fall detection (FD) and fall prevention (FP) systems have been presented by researchers. These systems can be classified into three categories according to the type of the sensor used: wearable technologies, ambient technologies and a combination of wearable and ambient technologies. Wearable technologies encompass two different types of hardware: inertial sensors (e.g., tri-axial accelerometer) and locating systems (GPS). Ambient technologies include vision sensors (e.g., cameras), sound sensors (e.g., microphones), radar sensors (e.g., Doppler radar), infrared sensors and pressure sensors (e.g., floor sensors) or combinations of them [2].

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 [3] and to reduce the number of false alarms. Moreover, we would like to track the elderly person's activity in order to prevent falls.

In a previous work [4], a head tracking method using the fusion of low cost thermal and depth sensors for home environments whilst preserving privacy was proposed. The addition of thermal sensor improves the tracking with depth sensor. For example, thermal information adjusts depth detection by discriminating be-

*Corresponding author. E-mail: alain-jerome.fougeres@ecam-rennes.fr.

tween hot objects and cold objects moved after calculating the background image. The results demonstrate that fusion improves tracking, namely when segmentation was erroneous. However, it missed partial occluded falls, and was unable to track fast motions in real time which are interesting for fall detection. For these reasons, this paper examines the data fusion to improve fast motion tracking and partial occlusion using particle filter (PF) algorithm based on head position. Particle filtering is a sequential importance sampling method using a set of particles to estimate the posterior distribution of a Markovian process, given noisy observations. The key idea of PFs is to represent and maintain the posteriori density function by a set of random samples with associated weights and to compute the state estimate from those samples and their weights. For each depth-thermal image pair, the head position is first segmented in the depth image, and then matched with the thermal image using calibration information to predict the actual position according to the previous state. The fusion of thermal and depth information is used to update this predicted state.

This paper extends the depth-thermal tracking method based on particle filter, explained in [4], by including the velocity of the head in the state vector to improve fast motion. The method was tested on several sequences, with or without depth-thermal fusion: results show its robustness and accuracy and also demonstrate that adaptive measurements of each particle by using the velocity and the position of the head improve the fast motion, partial occlusion and scale variation.

The paper is organized as follows: Section 1 contains a general introduction of fall detection system. Section 2 gives an overview of the state-of-the-art vision fall detection systems. Section 3 describes the material used, the architecture of tracking algorithm, and proposed methodology to detect falls. Section 4 discusses the experimental set-up of our dataset, the results with or without depth-thermal fusion, as well as the performance evaluation and a detailed discussion. Section 5 provides conclusion and further research potential.

2. Related work

A fall is defined as an event which results in a person coming to rest inadvertently on the ground or floor or other lower level. Adults older than 65 years of age suffer the greatest number of fatal falls [1]. Several FD systems have been proposed to identify and classify human activities of daily living (ADL) and to reduce the risk of elderly falls, the response and the rescue time. Many studies focusing on FD survey were increased rapidly in the world. For example, Mubashir et al. [5] chose to classify the FD systems into three categories: wearable device based, ambience sensor based and camera (vision) based. However, Igual et al. [6] chose only two categories: context-aware systems and wearable devices. While falling detection context is promising, exciting challenges still occur. In this paper, we will study the most commonly cited works in the literature according to their advantages and their drawbacks such as cost, application, installation and privacy.

Over the last decade, the focus has been on context aware systems (vision systems especially), because the person is more independent and not constrained by the presence and the configuration of the device. Several methods use particle filters for object tracking and localization. In [7, 8] the authors describe the application of particle filters for tracking moving objects using background subtraction to track human silhouettes based on color images. In [9], Rougier et al. have used the head's velocity to detect the fall in visual videos by setting thresholds manually. In the same vein, [10] have used particle filter for head tracking based on colored histograms. In [11], L. Loza et. al have applied PF on thermal imagining. Mubashir et al. [5] have used head position to track the person's silhouette based on a Gaussian classifier. In [12] the silhouette was extracted from video to localize the person which is a common strategy in the literature. However, these methods provide false alarms because it is difficult to distinguish a fall from other similar actions, e.g. sitting down. Therefore, in [13], Auvinet et al. have added other cameras to analyze the shape of the person in 3D and avoid hidden falls. But elderly people dislike the use of visual cameras even with local processing. They prefer non-invasive devices which preserve their privacy according to a psychosocial study done by LAUREPS laboratory at University of Rennes 2.

In order to protect user privacy, 3D fall detection systems using depth sensors were used in a fall detection context. The aim of using a depth camera like Kinect is to analyze the human shape and extract 3D features for fall detection [14]. A recent work used head position detection, extracting from depth images [15] and the experimental results confirm the feasibility and the effectiveness of the approach for real world applications. In [16], 3D data are exploited to perform head detection for a fall detection framework.

Human silhouettes, obtained by a background subtraction, are detected and all possible head positions are searched on contour segments. But in fall detection, it could not recognize correctly for instance when the person bent his knee too much to slow down the fall.

To avoid this problem, some works have used other non-invasive sensors such as thermal sensors. For example, Hayashida et al. [17] integrate a thermal infrared array sensor to detect falls by computing the maximal thermal difference between the background and foreground pixels which is a technique used for static cameras. The current frame is subtracted from the model of the background scene and eventually, the difference, determines the moving objects. However, the configuration was sensitive to room temperature and brightness. In [18] authors proposed a system to recognize human activities, which include falls, by means of a single thermal infrared sensor. Several features based on temperature thresholds are proposed to be evaluated by a Support Vector Machine (SVM) in the classification. However, in [19, 20] authors have proposed another type of thermal sensor but relatively expensive. In [21], a very economical thermal imaging based input modality is proposed to detect falls using the optical flow of human movements tested on public datasets. These proposed methods achieved a good performance but included some confusion in distinguishing between falling and sitting.

The number of studies using analytical methods is still increasing but there is a new trend in fall detection which is the use of machine learning methods and the most popular algorithm in this context is deep learning. For instance, Quero et al. [22] detected falls from non-invasive thermal vision sensor (Heimann HTPA 32×31) using Convolutional Neural Networks (CNN). Wang et al. [23] proposed a fall detection system using a PCAnet to extract features from color images and then applied a SVM to detect falls. Nunez-Marcos et al. [24] proposed a similar approach but, instead of a PCAnet, they used a modified VGG16 architecture. These methods are promising but usually require a large dataset to train a classifier and are inclined to be influenced by the image quality.

In order to efficiently improve results, some papers combined sensors. Interesting examples are provided in [25] and [26], for example Kinect and accelerometers, or cameras with microphones plus accelerometers. In [27], human silhouette was extracted using RGB-D camera. Recently RGB-T systems attracted a lot of attention, e.g. Wu et al. [28] combined RGB and thermal data into one vector which, however, introduces redundant information and the use of color information cannot preserve user privacy.

In this paper, we propose a combination between depth and thermal sensors, FLIR sensor (80×60) and Kinect sensor (640×480) respectively. Our method aims to track the head position in a context of daily activities classification. We apply particle filter on fusion information, include the position and the velocity of the head to the state vector and modify the modality for each particle based on an adaptive weighting of each frame. In the interest of privacy, we chose not to use color imaging.

3. Material and methods

The proposed system aims to track the head position using two types of sensors with different resolutions which are mounted together. The head position can be tracked according to an analytical method applying on a segmented frame. With the calibration step done before starting processing, the unidirectional thermal-depth matching can be made throughout all sequences.

In this research, we chose head position as Region Of Interest (ROI) because it is non-deformable, the hottest, highest and least hidden part of the body which can easily be approximated as an ellipse with only few parameters. Head motion is also a significant marker for fall detection.

3.1. Cameras and dataset

The fall detection system is based on thermal sensor (FLIR lepton 2.5, Focal length: 5 mm, Thermal Horizontal Field of View T_{HFOV}: 51°, Thermal Vertical Field of View T_{VFOV}: 37.83°, Thermal Resolution $T_{X_{res}}$: 80 pixels and $T_{Y_{res}}$: 60 pixels) and a depth camera (Microsoft Kinect V1, Focal length: 6.1 mm, Depth Horizontal Field of View D_{HFOV}: 58°, Depth Vertical Field of View D_{VFOV}: 45°, Depth Resolution $D_{X_{res}}$: 640 pixels and $D_{Y_{res}}$: 480 pixels). Figure 1 shows the sample output images of Kinect and FLIR acquisition system. These sensors can capture 3D and thermal video data under various light conditions. The extracted dataset can be divided into two principal categories (ADL and abnormal activities) to experiment the proposed method. Totally 60 video sequences with 10000 frames are recorded from 5 different human subjects in three different places with or without presence of obstacles.

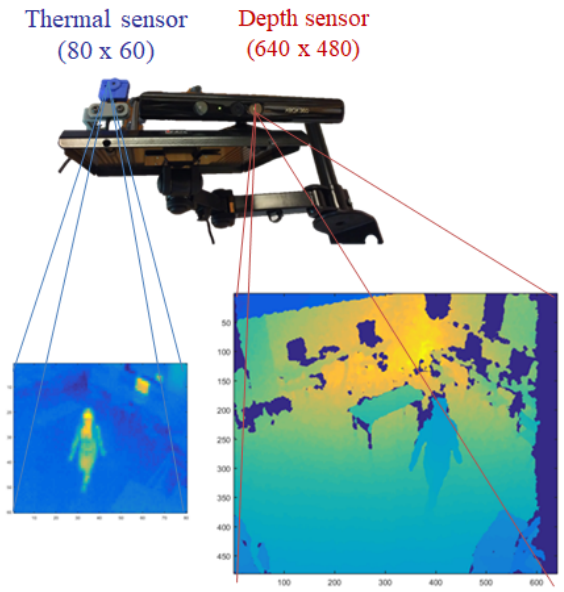

Fig. 1. Camera system

3.2. Framework

Figure 2 illustrates the framework of our proposed fall detection system. This proposed system can be divided into three principal stages: calibration, segmentation and tracking. The calibration step, which is done only one time, is executed after attaching the sensors to the ceiling to be able to match a depth pixel to its corresponding thermal pixel. The segmentation step, based on acquired images, serves to detect depth foreground image by subtraction of the background and extract head position. The tracking step is based on the head position segmented on the depth image and matched to the thermal position and improved by the particle filter.

3.3. Calibration

A calibration step is required to calculate the transformation parameters (extrinsic parameters). In the literature, a conventional black and white chessboard pattern is often used in many existing methods to calibrate two cameras or more. To obtain higher accurate calibration results, this pattern needs to be kept near the cameras. The orientation could sometimes result in limiting the number of poses [29]. Besides, this pattern cannot be seen by thermal sensors. For these reasons, we have decided to design a special pattern which contains several tubes of different heights mounted together on a board and different resistors fixed on each

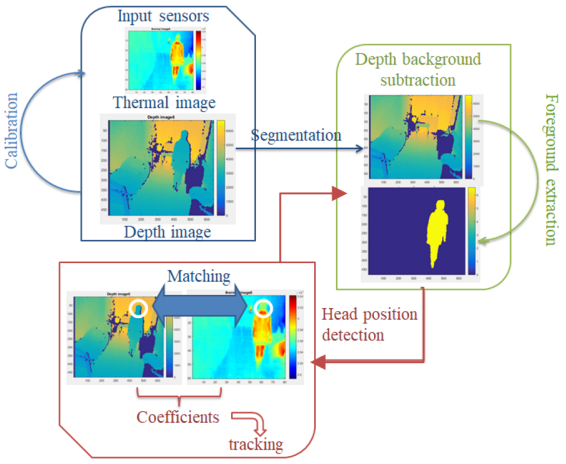

Fig. 2. Fall detection framework

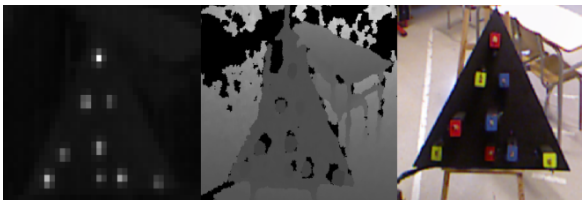

Fig. 3. Calibration pattern on thermal image, depth image and color image respectively from the left

tube. The idea of this pattern is simple. The tubes will be seen by the depth sensor and the heat emitted by the resistors will be seen by the thermal sensor. Calibration pattern is shown on depth, thermal and color images in Figure 3.

The calibration operation comprises modeling the image transformation process. This process transforms points from image coordinates to a common world coordinates system. The idea is to find the relation between the coordinates of a point in the depth image with the associated point in the thermal image taking into consideration the spatial coordinates of each point (Figure 4). The estimation of the relationship between these two coordinate systems needs three steps [30]:

(1) The estimation of the transformation of the depth image coordinates (u_d, v_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, Eq. (1):

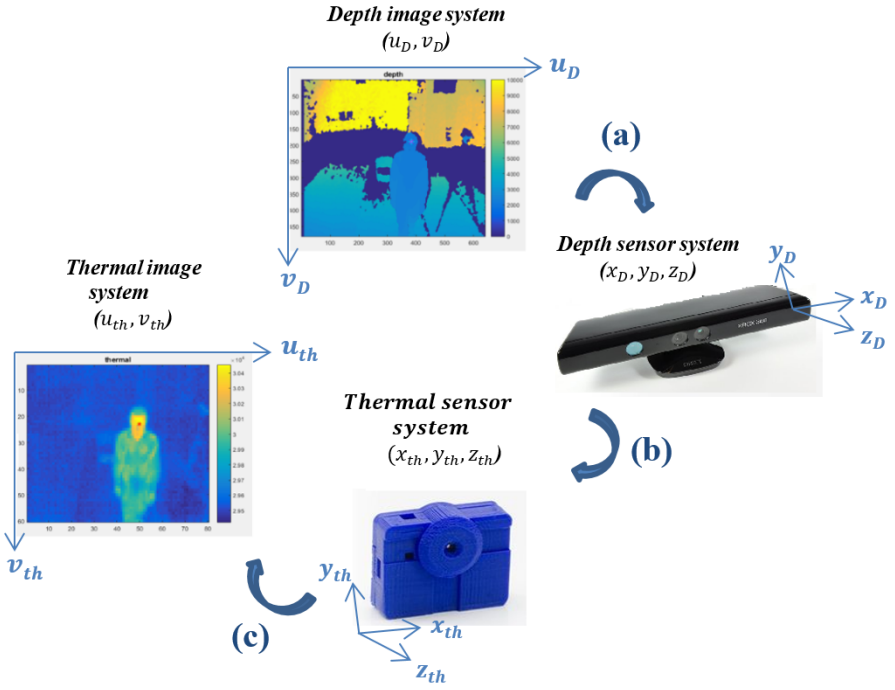

Fig. 4. Calibration system

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

(2) The transformation between the coordinate system (x_d, y_d, z_d) of the depth sensor to that (x_{th}, z_{th}, z_{th}) of the thermal sensor. It can be obtained from the extrinsic parameters, in our case a rotation matrix **R** and a translation matrix **T**, Eqs. (2, 3 and 4):

$$\mathbf{R} = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix} * \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \quad (2)$$

$$\mathbf{T} = \begin{pmatrix} d_x \\ d_y \\ d_z \end{pmatrix} \quad (3)$$

$$\begin{pmatrix} x_{th} \\ y_{th} \\ z_{th} \end{pmatrix} = \mathbf{T} + \mathbf{R} \begin{pmatrix} x_d \\ y_d \\ z_d \end{pmatrix} \quad (4)$$

where α, θ and β are the Roll, Pitch and Yaw angles [21].

(3) The transformation between the coordinate system (x_{th}, z_{th}, z_{th}) of the thermal sensor to the thermal image coordinates (u_{th}, v_{th}) (there are only 2 equations since the pixel value is the temperature which is not geometric information). This can be done analytically from the intrinsic parameters of the thermal camera, Eq. (5):

$$\begin{cases} u_{th} = \frac{T_{X_{Res}}}{2z_{th} \tan\left(\frac{T_{HFOV}}{2}\right)} x_{th} + \frac{T_{X_{Res}}}{2} \\ v_{th} = -\frac{T_{Y_{Res}}}{2z_{th} \tan\left(\frac{T_{VFOV}}{2}\right)} y_{th} + \frac{T_{Y_{Res}}}{2} \end{cases} \quad (5)$$

In our case, the intrinsic parameters are the values given by the constructor. So the purpose of the calibration is to estimate 3 parameters of rotation transformation and 3 parameters of translation transformation $(\alpha, \theta, \beta, d_x, d_y, d_z)$ respectively using a nonlinear optimization techniques (Levenberg-Marquardt [31]) and then to generate a one to one pixel correspondence from the depth to the thermal images (nonreciprocal correspondence).

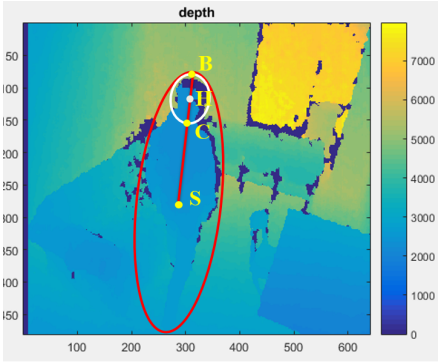

Fig. 5. Silhouette and head position

3.4. Segmentation step

In order to improve the segmentation robustness, we calculate a reference map based on the mean and standard deviation of the N first depth images without any moving objects. Then, we subtract the reference map from depth image to extract the depth foreground image. For a new frame, each pixel $p(i,j)$ is considered as foreground so long as it is above a threshold calculated by the variance of the reference map. Otherwise, the pixel is classified as background:

`if` $|p(i,j) - ref(i,j)| > 2\sigma(i,j)$
`then` $p(i,j) \in$ `Foreground`
`else` $p(i,j) \in$ `Background`

where $ref(i,j)$ is the mean pixel (i,j) from reference map and $\sigma(i,j)$ is the standard deviation of $p(i,j)$.

Next we have one or more areas detected as foreground: we compare these areas and we hold only the bigger one. Following the choice of the area, we approximate the body with an ellipse.

Finally, we model the head as a smaller ellipse with the same orientation of the silhouette ellipse. Human adult body proportions are brought about by differential growth of the body segments. From 25 years of age the head is only approximately one-sixth of the total body length [32]. Therefore, we fixed the center **SC** of the head ellipse **H** at the $1/6$ of half major axis **SB** from the upper part of major axis (see Figure 5), Eq. (6):

$$\mathbf{SC} = \frac{5}{6}\mathbf{SB} \tag{6}$$

where **S** is the silhouette ellipse center, **B** is the upper point of the major axis and **BC** is the head major axis knowing that the ratio between major axis and minor axis is set to 1.2 [7]. The sole use of the segmentation process is not a robust method to track the head position. False alarms can be trigged for any object moved after calculating the reference map. Thus, the use of a tracker is necessary.

3.5. Tracking process

The aim of the tracking is to estimate the position of the head during a sequence by considering the last movement of this ROI. Therefore, we chose a sequential Monte Carlo method which is Particle Filter (PF) method. At each frame t, the state vector x_t of PF is defined by the center **H**, the size **L** and the orientation θ of the head extracted by the segmentation step in depth image, Eq. (7).

$$x(S1)_t = (x_H, y_H, L, \theta)^T \tag{7}$$

PF method seeks to estimate the hidden state vector x_t from the previous state vector x_{t-1}, depth observation vectors $Z_t = \{z_1, \cdots, z_t\}$ and thermal observation vectors $H_t = \{h_1, \cdots, h_t\}$.

PF uses a sample of N particles $S_t = \{S_t^1, \cdots, S_t^N\}$ to approximate the conditional probability $p(x_t/Z_t, H_t)$. Each particle S_t^n can be seen as a hypothesis about x_t and is weighted by $\pi_t(n)$ which are normalized. Particles are resampled according to their weights and are updated according to new observations (coefficients) [33]. $new_{obsv}(n)$ is a linear combination of coefficients tested in different forms (details in section 3.6). Thus, the PF does not consider one state vector but N particles state vectors associated with different weights. At each frame, the estimation of the head position is based on two models, the first model called AM considers the weighted average of the N particles and the second model called MM considers the particle with maximum weight.

To improve the estimation of the head position especially in cases of fast motion, we added the velocity (v_{x_H}, v_{y_H}) on the state vector, Eq. (8).

$$x(S2)_t = (x_H, y_H, v_{x_H}, v_{y_H}, L, \theta)^T \tag{8}$$

Below, we briefly define the PF algorithm. For each frame, we resample a new sample of N particles named S_{t+1}^n, based on the previous state of each particle and the associated weights in order to prevent the problem of "particles degeneration" [34]. Next, we predict the

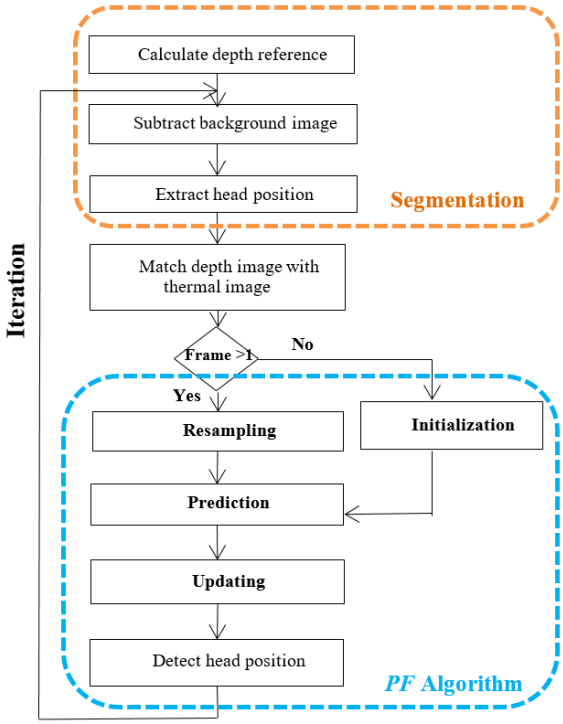

Fig. 6. Segmentation step and *PF* algorithm

actual state x_{t+1} according to the propagation of particles based on the prediction equation, Eq. (9):

$$S_{t+1}^n = AS_t^n + w_t \tag{9}$$

where A stands for 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 $new_{obsv}(n)$ (see figure 6).

Thus, the steps of iterative *PF* tracking algorithm are:

(1) *Initialization*: Generate a sample of N particles $S_1 = \{S_1^1, \cdots, S_1^N\}$ based on the probability of the state vector $p(x_1)$, and initialize the weight of each particle by $\pi_t(n) = 1/N$.
(2) *Resampling*: Resample 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 $\pi_t(n)$ at frame t according to observation vectors.
Then normalize the weight:

Table 1
Occluded particle conditionss

	Thermal image Flag(n) = 0	Thermal image Flag(n) = 1
Depth image Flag(n) = 0	G_T eliminated	Keep all coefficients
Depth image Flag(n) = 1	Resample particles	G_D and D_D eliminated

$$\pi_t(n) = \frac{\pi_t(n)}{\sum_{k=1}^{N} \pi_t(k)} \tag{10}$$

and return to step 2.

3.6. Depth-thermal fusion (first method)

Updating particle weights is a key point of *PF* and is specific for each application (see [35] for color information).

The weight of a particle is defined by Eq. (11):

$$\pi_t(n) = \frac{1}{\sqrt{2\pi}\sigma} \exp(new_{obsv}(n)/2\sigma^2) \tag{11}$$

where σ is theoretically the standard deviation of the coefficients combination. However, we have chosen a constant value of σ for computational cost reason.

We have tested several values and we returned the one that rendered the best result and $new_{obsv}(n)$ is a linear combination of three coefficients of thermal and depth observations: a depth distance coefficient D_D, a depth gradient coefficient G_D and a thermal gradient coefficient G_T, where:

- D_D is the distance between the center of the particle and the center of the segmented head in the depth image.
- G_D and G_T are the gradients along the particle ellipse in the depth image and the thermal image respectively, as inspired by [9].

When updating particle weight, we have observed that an occluded particle can decrease the performance of tracking. For example, it can influence the result of the *AM* model. To avoid this problem, we have added a flag to each particle at each frame (Flag (n) = 0 when the particle n is occluded) and we have eliminated the coefficient of this particle n on the update step following Table 1.

A particle is occluded for a sensor if it is out of the vision field of this sensor. This can occur especially after the prediction step.

Table 2
Importance factor IF values

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

In this work, we tested four models of coefficient combination to update the particle weights in Eq. (11).

The first model ($M1$) uses only 2 depth coefficients (D_D and G_D), Eq. (12):

$$new_{obsv1}^{(n)} = \alpha D_D^{(n)} + \beta G_D^{(n)} \tag{12}$$

where n is the particle index. There is no fusion here since we use only depth images.

The second model ($M2$) combines one depth coefficient (D_D) with one thermal coefficient (G_T), Eq. (13):

$$new_{obsv2}^{(n)} = \alpha D_D^{(n)} + \beta G_T^{(n)} \tag{13}$$

The third model ($M3$) combines the 3 coefficients, Eq. (14):

$$new_{obsv3}^{(n)} = \alpha D_D^{(n)} + \beta G_D^{(n)} + \gamma G_T^{(n)} \tag{14}$$

We call parameters $\alpha, \beta, \gamma \in [0,1]$ importance factors (IF). The distance coefficient is always considered because it is very discriminant.

We tested several combinations of static IF in order to estimate the impact of each coefficient (see Table 2).

3.7. Depth-thermal fusion (second method)

The use of static IF values is a general way to estimate coefficient impact because we fix a static value during the whole sequence. But at certain frames, thermal information can be more important than depth information and conversely. For instance, when the person is close to furniture that was moved after the reference map was calculated, the depth observations may not be relevant, because the silhouette can be merged with the furniture. Or if the person is close to a heater, the thermal observation cannot be efficient. Therefore, we decided to adjust the important factors dynamically and change the values at each frame according to the importance of each coefficient using these rules, Eqs. (15, 16 and 17).

$$\text{if } \max\left(\max_n G_D^{(n)}, \max_n D_D^{(n)}\right) < \max_n G_D^{(n)} \text{ then } \gamma = 1/2 \text{ and } \alpha = \beta = 1/4 \tag{15}$$

$$\text{if } \max\left(\max_n G_T^{(n)}, \max_n D_D^{(n)}\right) < \max_n G_T^{(n)} \text{ then } \beta = 1/2 \text{ and } \alpha = \gamma = 1/4 \tag{16}$$

$$\text{if } \max\left(\max_n G_T^{(n)}, \max_n G_D^{(n)}\right) < \max_n D_D^{(n)} \text{ then } \alpha = 1/2 \text{ and } \beta = \gamma = 1/4 \tag{17}$$

In subsequent sections of this paper, we have compared this model called ($M4$) with other models defined previously.

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 methods, (2) evaluate the performance of the fusion algorithm, (3) evaluate each IF model, and (4) compare IF values.

In all tests, we used the following values: $N = 1000$ 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 and the processing was performed using Matlab on Intel(R) Core(TM) i7-6700HQ CPU, 2.6 GHz.

The criteria for evaluation of our method utilizes two quantitative metrics (more details in [36]): the localization error (called *precision plot*) 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 (called *success plot*) which is the overlap of the ground truth area and the tracking area.

4.1. Fusion of information

In this section, we illustrate, as described in section 3.4 and 3.5, the results of head segmentation (Eq. (6)), depth version ($M1$ model) (Eq. (12)) and first fusion model $M2$ (Eq. (13)), the difference between AM and MM models and a comparison between the first fusion model $M2$ and second fusion model $M3$ (Eq. (14)).

Figure 7 shows a representative a normal ADL of our datasets used in the evaluation experiment. The first three images in Figure 7 represent the results of segmentation. These results show that segmentation is wrong because the size and the position of the head do not vary whereas the silhouette's position from the captor does. In this case, the problem is caused by the segmented silhouette which does not contain legs. The second test is based on the depth method mentioned before. Comparing these results, we can see that the depth version $M1$ is totally erroneous because this method used segmentation to calculate the distance coefficient. In other words, the depth sensor is useless on its own. The last three images show the results of the first fusion model $M2$ which is able to track the head more accurately as the person moves because it employs a combination of thermal and depth imaging.

The results of MM and AM models are shown on Figure 8.a) and 8.b) respectively. We can see that the AM model provides the closest pose to GT.

In order to evaluate coefficient combination, Figure 9 shows a comparison of two fusion models a) $M2$ model and b) $M3$ model. Visually the second fusion model provides the closest pose to GT.

To validate these results, we have evaluated these models according to the precision and success metrics. The evaluation results of quantitative measurements over a sequence show in Figure 10 that fusion of 3 coefficients provides the most accurate results. As expected, considering three coefficients together gives better results than using only two coefficients.

4.2. Comparison of static IF models

As mentioned in section 3.6, the third model ($M3$) combines the 3 coefficients, (Eq. (14)).

To evaluate the importance of each IF (α,β,γ) , we have performed different tests of IF values fixed on Table 2 on several sequences.

Figure 11 illustrates a comparison between these tests on a normal ADL frame. The visual results show the impact of IF in estimating the new head position. Confirming the IF impact during a sequence using the two quantitative measurements, Figure 12 shows a clear difference between the performance of $C4$ (Figure 11.d) compared to other tests.

4.3. Robustness of adding velocity on state vector

In this study, we present an improved version of an algorithm initially proposed in our earlier work [4]. In addition to size and orientation of the head ellipse, we have added the velocity to the state vector. Figure 13 illustrates a representative scene of fast motion. The first two images (Figure 13.a) show results of the algorithm without adding velocity on the state vector. Figure 13.b shows the impact of velocity especially in fast movement.

4.4. Robustness of adaptive weighting

As mentioned in section 3.7, the fourth model ($M4$) adjusts the important factors dynamically and changes the values at each frame according to the importance of each coefficient. To evaluate the impact of the adaptive combination, we have compared this model with the result of test $C4$ mentioned in section 4.2. Figure 14 illustrates a comparison between $C4$ and this model according to the success metrics. Figure 14 shows a clear difference between the performance of $M4$ compared to $C4$.

4.5. Summary

In this research, we started by testing the models of head estimation. The first model AM, which considers the weighted average of these N particles, provides the closest pose to GT as opposed to the second model MM which considers the particle with maximum weight. Then we compared segmentation and depth versions to the first and second fusion models ($M1$ and $M2$). We concluded that using both thermal and depth observations improved tracking results.

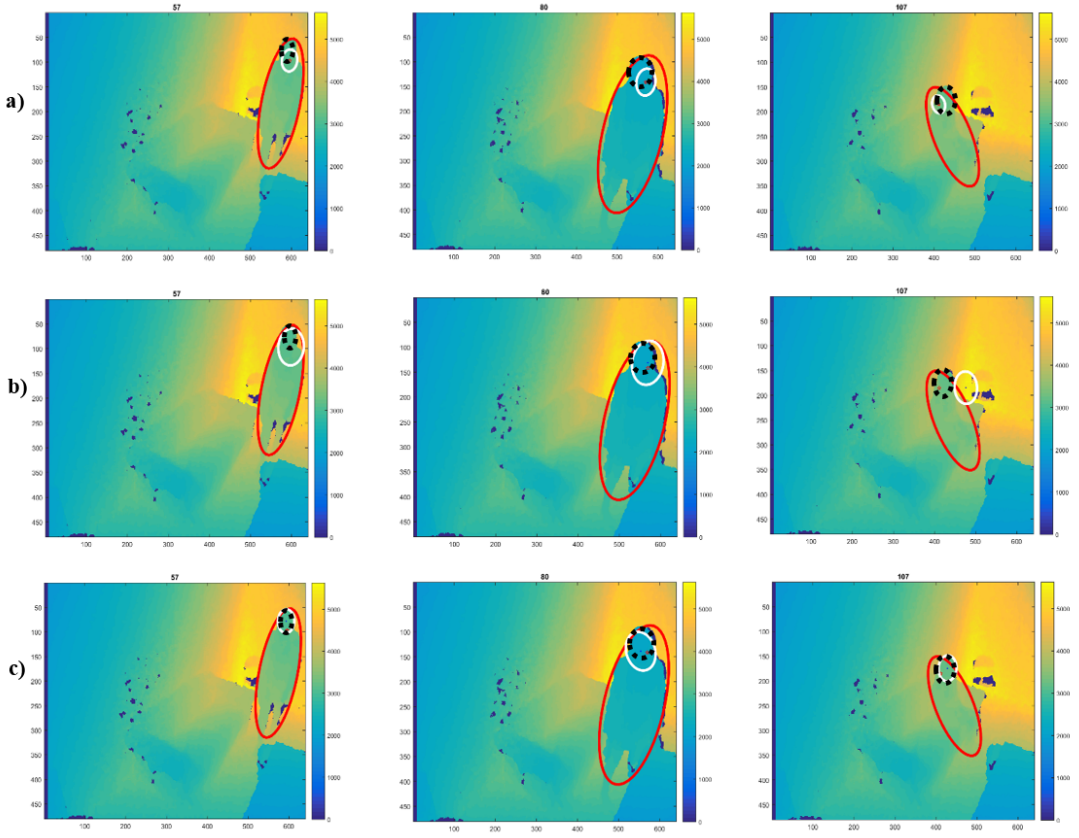

Fig. 7. Tracking results on different frames of depth sequence a) Segmentation only, b) Depth version ($M1$ model), c) First fusion model ($M2$). Tracking results are in white, silhouette ellipse is red and GT ellipse is black

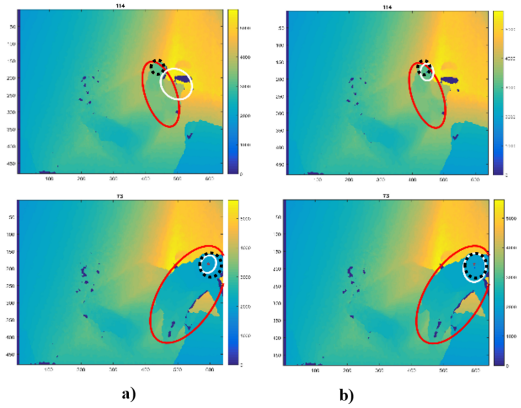

Fig. 8. Tracking results on two frames of sequence a) MM model and b) AM model. Tracking results are white, segmentation ellipse is red and GT ellipse is black

In order to estimate the impact of each observation, we assigned an importance factor IF to each coefficient and we compared 6 different tests of static IF.

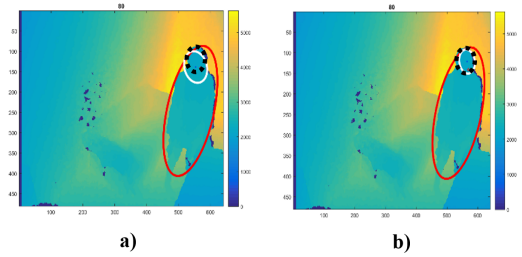

Fig. 9. Tracking comparison results between a) first fusion model ($M2$) and b) second fusion model ($M3$). Tracking results are white, silhouette ellipse is red and GT ellipse is black.

The results were clearly different between each test according to the environment at time t. For this reason, we modified the static IF to dynamic according to the coefficient value at each frame.

Finally, we added velocity to the state vector to improve the estimation of the head position especially in cases of fast motion.

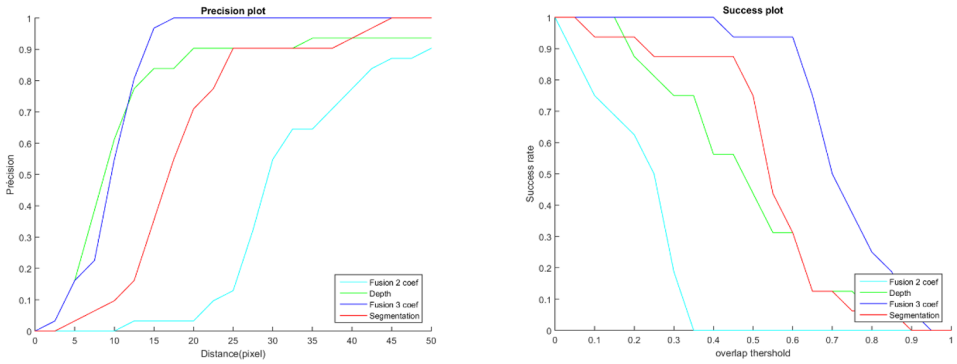

Fig. 10. Quantitative measurements over a sequence. Localization error (a) and the overlap score (b) using the segmentation (red), the depth version (blue) the $M2$ model (cyan) and the $M3$ model (blue)

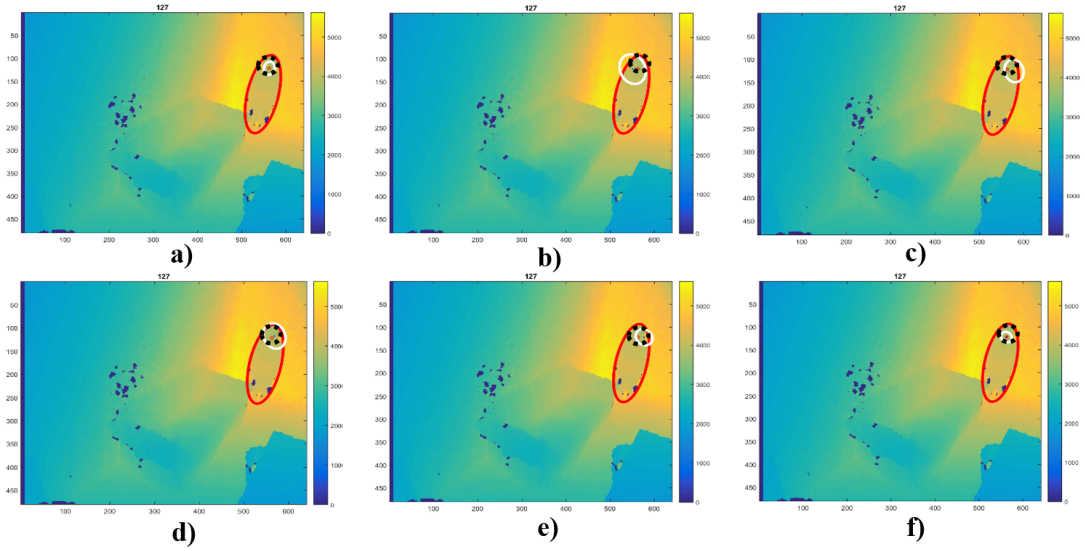

Fig. 11. Tracking results of 6 IF tests on the same frame a) $C1$ test, b) $C2$ test, c) $C3$ test, d) $C4$ test, e) $C5$ test and f) $C6$ test, tracking results are white, silhouette ellipse is red and GT ellipse is black.

5. Conclusion

In this paper, we have detailed a tracking approach based on a particle filter using depth and thermal information fusion to detect the position of the head of a person in an indoor environment. Position, velocity, orientation, and size of the ellipse enclosing the head are used to predict the new position of the head. Furthermore, adaptive weighting was applied on the measurements of each particle according to the strength of each coefficient to update the predicted position on each frame. Consequently, this method solves the updating problem we encountered in previous tracking works, caused by changes in the background. The proposed framework has been tested in several situations with different models and compared with other methods to establish the accuracy of the algorithm. Moreover, results have shown that our system gave the most accurate tracking results even in critical situations with very low resolution images.

Our aim is to refine the work presented here and better address the constraints of fall detection systems. Going forward, we plan to use deep learning (DL) methods due to their performances as mentioned in recent works [19, 20] to more accurately recognize human posture. We will start by 4 postures (standing up, sitting, lying on the ground and lying on a bed or sofa) in the context of fall detection and also fall prevention

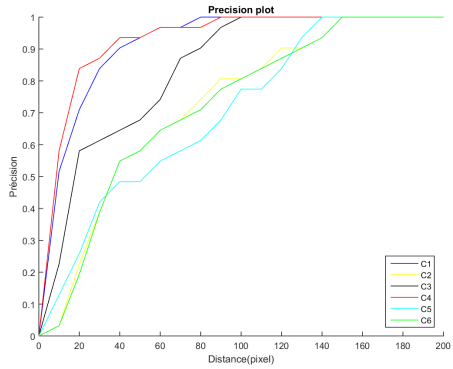

Fig. 12. *IF* quantitative measurement over a sequence. Localization error of $C1$ test (blue), $C2$ test (yellow), $C3$ test (black), $C4$ test (red), $C5$ test (cyan) and $C6$ test (blue).

by activity analysis. Before using DL, we will apply a preprocessing step on depth images to enhance their quality and avoid losing pertinent information (see Figure 15).

Acknowledgements

This work is funded under the PRuDENCE project (ANR-16-CE19-0015-02) which has been supported by the French National Research Agency. A sincere thank you to Tabitha Courbin for her diligent proofreading of this paper.

References

[1] World Health Organization , [homepage on the Internet consulted on 25 May 2019]. https://www.who.int/news-room/fact-sheets/detail/falls.

[2] Q. Zhang, L. Ren and W. Shi, HONEY: a multimodality fall detection and telecare system, *Telemedicine and e-Health* **19**(5) (2013), 415–429. doi:10.1089/tmj.2012.0109.

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

[4] I. Halima, J.-M. Laferté, G. Cormier, A.-J. Fougères and J.-L. Dillenseger, Sensors fusion for head tracking using Particle filter in a context of falls detection, in: *First International conference on signal processing & artificial intelligence (ASPAI' 2019)*, 2019, pp. 134–139.

[5] M. Mubashir, L. Shao and L. Seed, A survey on fall detection: Principles and approaches, *Neurocomputing* **100** (2013), 144–152. doi:10.1016/j.neucom.2011.09.037.

[6] R. Igual, C. Medrano and I. Plaza, Challenges, issues and trends in fall detection systems, *Biomedical engineering online* **12**(1) (2013), 66. doi:10.1186/1475-925X-12-66.

[7] M. Yu, S.M. Naqvi and J. Chambers, Fall detection in the elderly by head tracking, in: *Proc. 2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, 2009, pp. 357–360. doi:10.1109/ssp.2009.5278566.

[8] G. Debard, G. Baldewijns, T. Goedemé, T. Tuytelaars and B. Vanrumste, Camera-based fall detection using a particle filter, in: *proc. 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015, pp. 6947–6950. doi:10.1109/EMBC.2015.7319990.

[9] 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* **31**(3) (2013), 246–254. doi:10.1016/j.imavis.2012.11.003.

[10] N. Bouaynaya, W. Qu and D. Schonfeld, An online motion-based particle filter for head tracking applications, in: *Proc. IEEE Int. Conf on Acoustics, Speech, and Signal Processing, 2005 (ICASSP'05).*, Vol. 2, 2005, pp. 225–228. doi:10.1109/ICASSP.2005.1415382.

[11] A. Łoza, L. Mihaylova, D. Bull and N. Canagarajah, Structural similarity-based object tracking in multimodality surveillance videos, *Machine Vision and Applications* **20**(2) (2009), 71–83. doi:10.1007/s00138-007-0107-x.

[12] C. Rougier, J. Meunier, A. St-Arnaud and J. Rousseau, Robust video surveillance for fall detection based on human shape deformation, *IEEE Transactions on Circuits and Systems for Video Technology* **21**(5) (2011), 611–622. doi:10.1109/TCSVT.2011.2129370.

[13] E. Auvinet, F. Multon, A. Saint-Arnaud, J. Rousseau and J. Meunier, Fall detection with multiple cameras: An occlusion-resistant method based on 3-D silhouette vertical distribution, *IEEE Transactions on Information Technology in Biomedicine* **15**(2) (2010), 290–300. doi:10.1109/TITB.2010.2087385.

[14] S. Gasparrini, E. Cippitelli, S. Spinsante and E. Gambi, A depth-based fall detection system using a Kinect® sensor, *Sensors* **14**(2) (2014), 2756–2775. doi:10.3390/s140202756.

[15] A.T. Nghiem, E. Auvinet and J. Meunier, Head detection using Kinect camera and its application to fall detection, in: *Proc 11th Int Conf on Information Science, Signal Processing and their Ppplications (ISSPA)*, IEEE, 2012, pp. 164–169. doi:10.1109/isspa.2012.6310538.

[16] D. Ballotta, G. Borghi, R. Vezzani and R. Cucchiara, Fully convolutional network for head detection with depth images, in: *Proc. 24th Int. Conf. Pattern Recognition (ICPR)*, 2018, pp. 752–757. ISSN 1051-4651. doi:10.1109/ICPR.2018.8545332.

[17] A. Hayashida, V. Moshnyaga and K. Hashimoto, The use of thermal ir array sensor for indoor fall detection, in: *Proc. IEEE Int. Conf. Systems, Man and Cybernetics (SMC)*, 2017, pp. 594–599. doi:10.1109/SMC.2017.8122671.

[18] S. Mashiyama, J. Hong and T. Ohtsuki, Activity recognition using low resolution infrared array sensor, in: *Proc. IEEE Int. Conf. Communications (ICC)*, 2015, pp. 495–500. ISSN 1938-1883. doi:10.1109/ICC.2015.7248370.

[19] S. Wu, G. Zhang, M. Zhu, T. Jiang and F. Neri, Geometry based three-dimensional image processing method for electronic cluster eye, *Integrated Computer-Aided Engineering* **25**(3) (2018), 213–228. doi:10.3233/ICA-180564.

[20] S. Wu, G. Zhang, M.Z. Ferrante Neri, T. Jiang and K.-D. Kuhner, A multi-aperture optical flow estimation method for an ar-

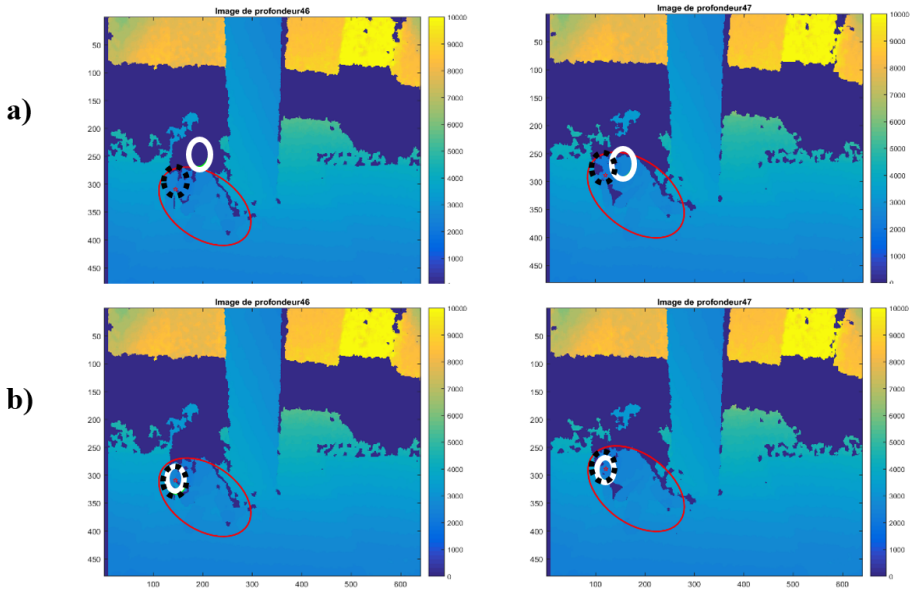

Fig. 13. Tracking comparison results between a) algorithm without velocity b) algorithm with velocity. Tracking results are white, silhouette ellipse is red and GT ellipse is black.

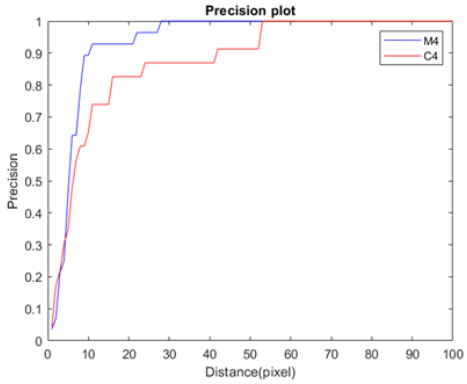

Fig. 14. Quantitative measurements over a sequence. Localization error of $C4$ (red) and $M4$ (purple)

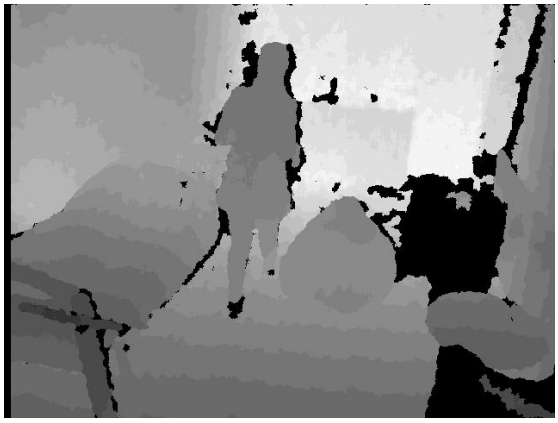

Fig. 15. Row depth image with large black regions (masked information due to poor quality)

tificial compound eye, *Integrated Computer-Aided Engineering* **26**(2) (2019), 139–157. doi:10.3233/ICA-180593.

[21] S. Vadivelu, S. Ganesan, O.V.R. Murthy and A. Dhall, Thermal Imaging Based Elderly Fall Detection, in: *Computer Vision - ACCV 2016 International Workshops*, Lecture Notes in Computer Science, Vol. 10118, 2016, pp. 541–553. doi:10.1007/978-3-319-54526-4_40.

[22] J.M. Quero, M. Burns, M.A. Razzaq, C.D. Nugent and M. Espinilla, Detection of Falls from Non-Invasive Thermal Vision Sensors Using Convolutional Neural Networks, in: *Proc. 12th Int. Conf.e on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018)*, Vol. 2, 2018, p. 1236. doi:10.3390/proceedings2191236.

[23] S. Wang, L. Chen, Z. Zhou, X. Sun and J. Dong, Human fall detection in surveillance video based on PCANet, *Multimedia Tools and Applications* **75**(19) (2016), 11603–11613. doi:10.1007/s11042-015-2698-y.

[24] A. Núñez-Marcos, G. Azkune and I. Arganda-Carreras, Vision-Based Fall Detection with Convolutional Neural Networks, *Wireless Communications and Mobile Computing* **2017** (2017). doi:10.1155/2017/9474806.

[25] G. Koshmak, A. Loutfi and M. Lindén, Challenges and Issues in Multisensor Fusion Approach for Fall Detection: Review Paper, *Journal of Sensors* **2016** (2016), 6931789:1–6931789:12. doi:10.1155/2016/6931789.

[26] K. Chaccour, R. Darazi, A.H. El Hassani and E. Andrès, From Fall Detection to Fall Prevention: A Generic Classification of Fall-Related Systems, *IEEE sensors journal* **17**(3) (2017), 812–822. doi:10.1109/JSEN.2016.2628099.

[27] S. Huang and Y. Pan, Learning-based Human Fall Detection using RGB-D cameras, in: *Proceedings of the 13. IAPR Int. Conf. on Machine Vision Applications, MVA 2013*, 2013, pp. 439–442.

[28] Y. Wu, E. Blasch, G. Chen, L. Bai and H. Ling, Multiple source data fusion via sparse representation for robust visual tracking, in: *Proc. 14th Int. Conf. Information Fusion*, 2011, pp. 1–8.

[29] P. Rathnayaka, S. Baek and S. Park, An Efficient Calibration Method for a Stereo Camera System with Heterogeneous Lenses Using an Embedded Checkerboard Pattern, *Journal of Sensors* **2017** (2017), 6742615:1–6742615:12. doi:10.1155/2017/6742615.

[30] R. Szeliski, *Computer Vision: Algorithms and Applications*, 1st edn, Springer-Verlag, Berlin, Heidelberg, 2010. ISBN 1848829345, 9781848829343.

[31] A.M. Mathai and R.S. Katiyar, A new algorithm for nonlinear least squares, *Journal of Mathematical Sciences* **81**(1) (1996), 2454–2463. doi:10.1007/BF02362352.

[32] B. Bogin and M.I. Varela-Silva, Leg length, body proportion, and health: a review with a note on beauty., *International Journal of Environmental Research and Public Health* **7** (2010), 1047–1075. doi:10.3390/ijerph7031047.

[33] M. Isard and A. Blake, CONDENSATION—Conditional Density Propagation for Visual Tracking, *International Journal of Computer Vision* **29**(1) (1998), 5–28. doi:10.1023/A:1008078328650.

[34] 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* **50**(2) (2002), 174–188. doi:10.1109/78.978374.

[35] K. Nummiaro, E. Koller-Meier and L.V. Gool, An adaptive color-based particle filter, *Image and Vision Computing* **21**(1) (2003), 99–110. doi:10.1016/S0262-8856(02)00129-4.

[36] Y. Wu, J. Lim and M. Yang, Online Object Tracking: A Benchmark, in: *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013, pp. 2411–2418. ISSN 1063-6919. doi:10.1109/CVPR.2013.312.