3D Human Tracking with Catadioptric Omnidirectional Camera

Fakhreddine Ababsa
Lispen Laboratory
Image Institute/Arts et Métiers
Chalon sur Saône France
Fakhreddine.ababsa@ensam.eu

Hicham Hadj-Abdelkader
Ibisc Laboratory
University of Evry
Evry France
hicham.hadjabdelkader@univ-evry.fr

Marouane Boui
Ibisc Laboratory
University of Evry
Evry France
Marouane.Boui@ibisc.univ-evry.fr

ABSTRACT

This paper deals with the problem of 3D human tracking in catadioptric images using particle-filtering framework. While traditional perspective images are well exploited, only a few methods have been developed for catadioptric vision, for the human detection or tracking problems. We propose to extend the 3D pose estimation in the case of perspective cameras to catadioptric sensors. In this paper, we develop an original likelihood functions based, on the one hand, on the geodetic distance in the spherical space SO^3 and, on the other hand, on the mapping between the human silhouette in the images and the projected 3D model. These likelihood functions combined with a particle filter, whose propagation model is adapted to the spherical space, allow accurate 3D human tracking in omnidirectional images. Both visual and quantitative analysis of the experimental results demonstrate the effectiveness of our approach

KEYWORDS

Human Tracking, Omnidirectional Camera, Particle filtering, Egomotion

1 Introduction

Omnidirectional cameras are commonly used in computer vision and robotics. Their main advantage is their wide field of view, allowing them to get an omnidirectional (360-degree) image with a single sensor. In this paper, we propose to perform a 3D human tracking using this kind of camera. The potential applications are numerous, including human behavior recognition, 3D human motion and human-machine interaction. There exist several techniques in the literature to localize in 3D a moving human with a visual sensor. A classical approach is to use several images captured by synchronized cameras [1]. However, deploying a multi-camera system in an uncontrolled environment remains very complicated, which limits the applicability of these methods. Furthermore, estimating 3D human pose from a single RGB image is a very difficult task. Over the years, the 3D human pose estimation problem using a monocular camera has received a lot of attention from the computer vision community. State of the art approaches can be classified in two main categories: model-based and Non-model-based methods. Methods without a model often use machine learning [2][3] to learn the mapping relationship between the human's appearance in images and their 3D posture in the workspace. These approaches are generally fast and accurate; however, they are limited by the need to use a large database for learning 3D poses. In addition, model-based approaches often use the geometry of the human body, which can be represented in different ways: articulated body, truncated cylinder, conical, etc. Constraints related to the mechanical structure of the human body and its kinematics make it possible to reduce the research space and thus to provide robust and accurate 3D human pose estimation. For example, in [4][5] the authors determine, in the current image, the 2D pose of the human using a "Flexible Mixtures of Parts" detector [6][7][8], then they use a regression technique to estimate the 3D pose in space. Moreover, cameras with large field of views (FOV) remains rarely used in the field of the 3D tracking, even if they have a real advantage by stretching the field of view to 360 azimuth. Several works have allowed the setting up of a precise model for the creation of images from omnidirectional sensors [9]. However, omnidirectional cameras remain mainly used to solve particular problems like visual servoing [10], navigation and motion estimation [11][12]. Only few works have studied the use of the omnidirectional camera for 3D tracking with catadioptric images. To our knowledge, the only works concerning 3D object tracking are of Caron et al.

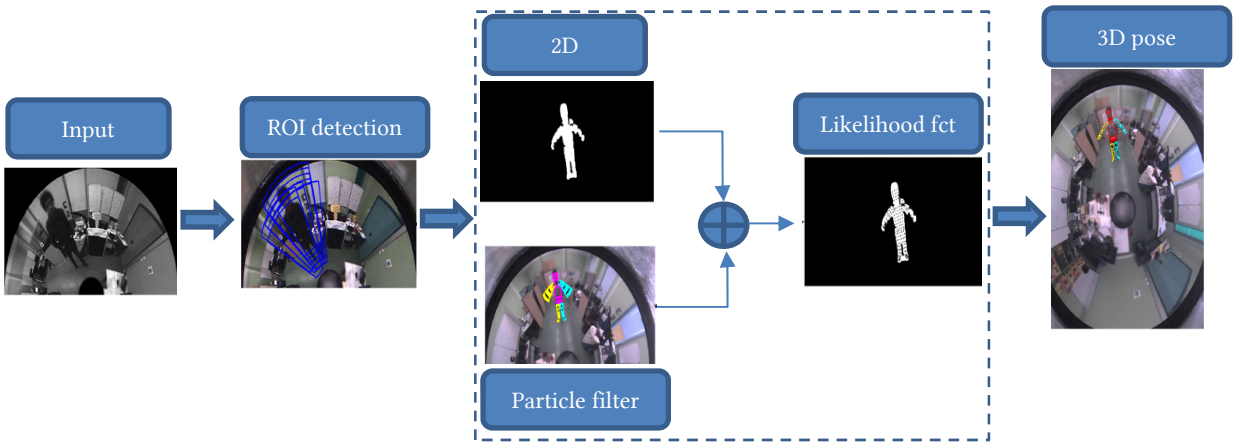


Figure 1: Data flow diagram of the proposed 3D tracking approach

[13], where several cameras are used to estimate the 3D object position, and the works of Yang et al. [14], which perform the 3D pose recognition without the pose estimation. In this research work, we make two main contributions: firstly, we develop an original likelihood functions based on geodesic distances in the SO3 space, to make more robust the human tacking in the omnidirectional images. Secondly, we developed a new database that contains omnidirectional images. Each image is associated with a 3D posture of the filmed person, captured using an external tracking system. 3D data are used as ground truth data to validate the developed approaches

2 3D Human tracking algorithm

Our 3D tracking approach is composed of several steps as illustrated in the figure 1. The first step concerns the detection of the region of interest (ROI) from the input image. To do this, we have implemented a human detection algorithm based on the HOG descriptors in omnidirectional images and using the gradient calculation in the Riemannian space. This step allows then the initialization of the tracking. From this initial pose, we can generate several positions thanks to the particular filter. Hence, each particle represents a posture of the 3D model respecting the possibilities of movement of the human body, and will be used to estimate the 3D pose in the next image.

2.1 3D Human model

In the literature, there exists several 3D articulated models to represent the human body. The number of degrees of freedom (dof) can vary from 82, as in [15] to 14 as in [16]. In most cases 32 dof are considered [17]. The choice of the number of degrees of freedom of the model is important, because it represents the number of parameters to be estimated for 3D tracking. Consequently, this number must be well chosen so that the tracking is accurate and real time. In our case, we chose a model with 34 degrees of freedom (figure 2). The upper and lower limbs are represented by truncated cylinders/cones.

This kind of representation is quite common in the literature [18][19] because it is easy to manipulate and to project on the images. The model is composed of 11 parts: pelvis, torso, head, arms, forearms, legs and thighs. The parameters of this model describe two complementary information about the body: the 3D pose and the shape. The shape is given by the length and width of the limbs, which in our case are supposed to be known. 30 parameters are used to define the model posture; they correspond to the position and global orientation of the pelvis and the relative articular angles between the neighboring limbs. Thus, the vector that gives a complete configuration of the kinematic model is $x = [x(1), x(2), \cdots, x(29), x(30)]$.

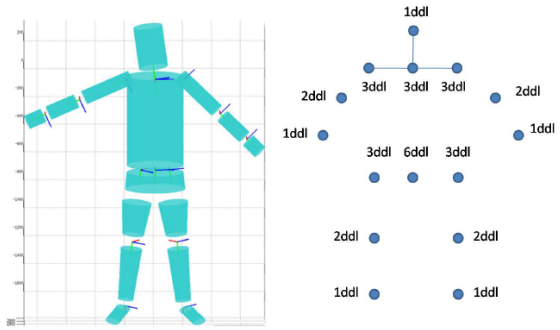


Figure 2: Degrees of freedom of the 3D model

2.2 Filtering

The 3D tracking problem can be modeled in a stochastic Bayesian framework [20] as an estimation problem of a conditional probability distribution (also called a posteriori) $p(x_t|y_{1:t})$. In our case, the state vector describes the 3D posture of the human body at time t ($t \in N$) and $y_{1:t} \equiv \{y_1, \cdots, y_t\}$ represent the observations extracted from the images. The distribution of the initial condition x_0 is assumed to be known and given by $p(x_0|y_0) = p(x_0)$. Such a process is considered to be a first-order Markov process because the state at the next time period is only reliant on the current state of the system. Hence, the dynamic equation can be given by $p(x_t|x_{1:t}) =$

$p(x_t|x_{t-1})$. Using the Bayes rule, the filter distribution can be calculated in two steps:

- Prediction step:

$$p(x_t|y_{1:t-1}) = \int p(x_t|x_{t-1}).p(x_{t-1}|y_{1:t-1}).dx_{t-1} \quad (1)$$

- Filtering step:

$$p(x_t|y_{1:t}) \propto p(y_t|x_t).p(x_t|y_{1:t-1}) \quad (2)$$

Several approaches can be used to resolve the filtering problem described above. The most popular is the Kalman filter [14]. In our case, we considered a refinement approach based on the simulated annealed particle filter (APF). This filter is based on Sequential Importance Resampling (SIR) algorithms [21][22] or CONDENSATION algorithm [23]. The APF filter was used for the first time for human tracking by Deutscher and Rei [24]. The main idea of the APF filter consists in iterating the state estimate several times in order to better localize the maximum of the likelihood function. For that, a set of steps (Layer) are iterated from layer M to layer 1. Thus, in the same image, the APF calculates the points associated with all the particles, it selects the particle with the highest weight, makes a finer sampling, and then re-estimates the weight of the new particles. The probability density at layer m + 1 is then represented by a set of N particles with their associated normalized weights: $S_{t,m+1} = \left\{x^i_{t,m+1}, \pi^i_{t,m+1}\right\}_{i=1}^{N}$. For the prediction step, a Gaussian diffusion model is considered and the Monte Carlo method used to resample the particles from the probability density in the previous layer m+1 as follows:

$$\left\{x^{(i)}_{t,m}\right\}_{i=1}^{N} \sim \sum_{j=1}^{N} \pi^{(j)}_{t,m+1}\, \mathcal{N}\left(x^{(j)}_{t,m+1}, \alpha^{M-m}C\right) \quad (3)$$

Where C is the covariance matrix and M the number of layers. The parameter α allows the covariance matrix to be gradually reduced during the successive iterations in order to drive the particles to the overall maximum of the likelihood function. The remaining particles that respect the body pose constraints (joint angle limits, no limb interpenetration) receive new normalized weights based on the "annealed" version of the likelihood function:

$$\pi^{(i)}_{t,m+1} = \frac{p\left(y_t\middle|x^{(i)}_{t,m}\right)^{\beta^m}}{\sum_{j=1}^{N} p\left(y_t\middle|x^{(j)}_{t,m}\right)^{\beta^m}}, i \in 1, \cdots, N \quad (4)$$

Where β^m is a parameter introduced to optimize the filter behavior so that about half of the particles are propagated to the next layer using the sampling equation (3). However, the choice of parameters α and β remains difficult. Heuristics can be used. In our case we took $\alpha = 0.4$ as recommended by [24].

2.3 Likelihood functions

In particle filter, the weights of the particles are proportional to the likelihood function. Thus, a high/low value of this function reflects whether the particle is in a region with a low/high posterior probability. In the context of 3D human tracking, the likelihood function must be able to measure the degree of similarity between the projection of the 3D human model and the image-segmented silhouette. In this work, we propose to combine three likelihood functions in order to make the tracking as robust as possible. Two functions are edge-based; they use gradient calculation in omnidirectional and spherical images, as well as geodesic distance to determine the distance between each pixel and the edge. The third likelihood function is silhouette-based and uses the projection of the 3D model in the spherical space.

2.3.1 Edge likelihood. to calculate the gradient in omnidirectional images we apply a differential operator on the Riemannian manifold [18][19]. Let $\mathcal{S}$ be a parametric surface on $\mathcal{R}^3$ with an induced Riemannian metric g^{ij} that encodes the geometrical properties of the manifold. The corresponding inverse Reimanian metric is defined as:

$$g^{ij} = \gamma \begin{pmatrix} -x^2(\xi-1)+\xi+1 & xy(\xi-1) \\ xy(\xi-1) & -y^2(\xi-1)+\xi+1 \end{pmatrix} \quad (5)$$

With

$$\gamma = \frac{(x^2+y^2+(1+\xi)^2)^2}{(1+\xi)\left(\xi+\xi^2+\sqrt{1-(x^2+y^2)(\xi^2-1)+2\xi+\xi^2}\right)^2} \quad (6)$$

The Riemannian metric can be seen as a weighing function of the classical gradient computed in the omnidirectional image:

$$\nabla f = g^{ij}\frac{\partial f}{\partial x^j} \quad (7)$$

For spherical images, the gradient is given by:

$$\nabla_{S^2} I_s(\theta,\phi) = \frac{\partial I_s(\theta,\phi)}{\partial \theta} e_\theta + \frac{1}{sin\theta}\frac{\partial I_s(\theta,\phi)}{\partial \phi} e_\phi \quad (8)$$

Where $I_s(\theta,\phi)$ is a spherical image, (θ,ϕ) are the longitude and colatitude angles respectively, and e_θ and e_ϕ are unit vectors. Once the gradient is calculated, the distance between the projections of the model in the spherical image and the contour can be obtained. In omnidirectional images unlike the perspective images, the distance between a pixel and its neighborhood depends on the position of the pixel in the image. Therefore, to compute the distance map we propose to use the geodesic distance. Let $\mathcal{P}$ be the projection that transforms an omnidirectional image $\mathcal{R}^2$ into a spherical image. The geodesic distance between two points in S^2, $x_1 = (\theta_1, \phi_1)$ and (θ_2, ϕ_2), is given by:

$$d_{S^2}(x_1,x_2) = arcos\left(\begin{bmatrix}\cos(\phi_1).\sin(\theta_1)\\ \sin(\phi_1).\cos(\theta_1)\\ \cos(\theta_1)\end{bmatrix}.\begin{bmatrix}\cos(\phi_2).\sin(\theta_2)\\ \sin(\phi_2).\cos(\theta_2)\\ \cos(\theta_2)\end{bmatrix}\right) \quad (9)$$

Thus, the edge distance map M^e_t can be calculated at time t. The likelihood function is then estimated by projecting the visible parts of the 3D human model into the edge map and calculating the mean squared error :

$$P^e(y_t|x_t) \propto \frac{1}{\xi^e_{x_t}(j)} \sum \left(1 - M^e_t\left(\xi^e_{x_t}(j)\right)\right)^2 \quad (10)$$

Where $\xi^e_{x_t}(j)$ represents the coordinates of the pixels corresponding to the projected 3D points along the different parts of the body, generated by the pose x_t.

2.3.2 Silhouette likelihood. Firstly, we used the unified spherical model [11] in order to project our 3D human model onto the unit sphere. Then, a Gaussian mixture model is implemented to estimate the scene background. The silhouette map M_t^s is generated by subtracting the estimated background at each time t. The likelihood function associated to this map can be written as follows:

$$P^s(y_t|x_t) \propto \frac{1}{\xi^s_{x_t}(j)} \sum \left(1 - M_t^s\left(\xi^s_{x_t}(j)\right)\right)^2 \qquad (11)$$

This function requires however that the 3D model be always projected inside the silhouette. In order to avoid this constraint. In order to avoid this constraint, we propose to extend the previous likelihood function in order to penalize regions that do not overlap. Let M^p be the binary silhouette map of the model projection. We then define three regions: R_t^1 the intersection of the maps M_t^p andM_t^s, R_t^2 the difference between the map M_t^s andR_t^1, R_t^3 the difference between the map M_t^p and R_t^1. The size of each region can be computed by summing all the pixels that compose it, as follows:

$$R_t^1 = \sum_i M_t^p(i) \,.\, M_t^s(i) \qquad (12)$$

$$R_t^2 = \sum_i M_t^s(i).\left(1 - M_t^p(i)\right) \qquad (13)$$

$$R_t^3 = \sum_i M_t^p(i).\left(1 - M_t^s(i)\right) \qquad (14)$$

Thus, the dual likelihood function will be defined as follows:

$$P^{sd}(y_t|x_t) \propto \frac{1}{2} \cdot \left(\frac{R_t^2}{R_t^1 + R_t^3} + \frac{R_t^3}{R_t^1 + R_t^3}\right) \qquad (15)$$

Finally, assuming that the likelihood functions are independent of each other conditionally to the pose x, we can merge them using the multiple probability formulation, which gives :

$$P(y_t|x_t) = \frac{1}{K|L|} \sum_{l \in L} \left(-log P^l(y_t|x_t)\right) \qquad (16)$$

Where y_t is the observation at time t and $L = \{e, s, sd\}$ is the set of the developed likelihood functions.

3 Experimental results

In this section, we present the performance of our 3D tracking algorithm applied on real data. The truth data is obtained through the SmartTrack tracking system of a Smarttrack system [25].

3.1 Performance criteria

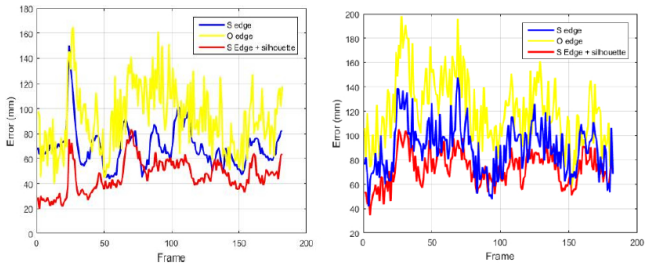
Various 2D and 3D evaluation methods have been proposed in the literature to evaluate human motion tracking and pose estimation. Several research studies propose to use the difference between the joint angles as an error measure [26] [27]. For our experiments, we use two comparison criteria. The first one concerns the 3D data; it uses the mean square error on the poses of the targets placed on the joints and extremities of the limbs. The 3D error is calculated as follows:

$$D_3(x, \hat{x}) = \frac{1}{N} \sum_{i=1}^{N} |m_i(x) - m_i(\hat{x})|^2 \qquad (13)$$

Where $m(x_i) \in \mathbb{R}^3$ is the position of the 3D target according to the pose x. Thus, the 3D error represents the distance in (mm) between our estimate and the ground truth data.
The second comparison criteria is based on the 2D error between the model projection and ground truth, measured directly in omnidirectional images. We used this criterion on the 2D ground truth video sequences.

3.2 Evaluation of likelihood functions

In this experiment, we tested the 3D tracking behavior according to the likelihood function used. We considered four likelihood functions (as defined in section 2.3): Spherical Gradient with Geodetic Distance (GG), Omnidirectional Gradient (OG), Dual Silhouette (DS), and a combination of DS and GG likelihood functions. The results obtained for sequence 1 and 2 are shown in the figure 3. We found that the (GG) function improves the results by about 11% compared to the (OG) approach. This is because the spherical image allows a better representation of omnidirectional images. In addition, the use of geodetic distances seems to give better results, and demonstrates that the distance calculated between the contour of the extracted person and the contour of the project model is better adapted and therefore more precise. This figure also shows that the combination of DS and GS likelihood functions gives best results



a) Sequence 1 b) Sequence 2

Figure 3: Performance of likelihood functions

4 Conclusion

In this paper, we described a new technique for the 3D body tracking using a catadioptric camera. The key feature of our approach is the development of several likelihood functions that take into account the geometry of omnidirectional images and spherical space. For that, we adapted the calculation of the gradient and used the geodesic distances, defined on the unit sphere, to generate the distance map for the gradient-based likelihood function. In addition, the 3D model projection and the silhouette extraction are performed in the spherical space in order to robustly construct the silhouette-based likelihood functions. We evaluated our method over several sequences. The obtained results are convincing and demonstrate the relevance of our tracking strategy. The further work includes : using a deformable model to improve the human detection; combining tracking results of an omnidirectional camera with other localisation sensors (like inertia sensors) for robust tracking; extending the work carried out by integrating a deep learning approach to improve the 3D tracking performance.

REFERENCES

[1] N. Dalal and B. Triggs, B. (2005). Histograms of oriented gradients for human detection. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1, pp. 886-893.

[2] M. Boui, H. Hadj-Abdelkader, F. Ababsa and E.H. Bouyakhf (2016). New approach for human detection in spherical images. IEEE International Conference on In Image Processing (ICIP 2016), pp. 604-608.

[3] A. Agarwal and B. Triggs. Recovering 3D human pose from monocular images. IEEE Trans. Pattern Anal. Mach. Intell.2006, vol. 28, pp. 44–58.

[4] G. Rogez, C. Orrite and J. Martinez-del Rincon. A spatiotemporal 2D-models framework for human pose recovery in monocular sequences (2008). Pattern Recognition, vol. 41, pp. 2926–2944.

[5] E. Simo-Serra, A. Quattoni, C. Torras and F. Moreno-Noguer (2013). A joint model for 2D and 3D pose estimation from a single image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3634-3641.

[6] Y. Yang and D. Ramanan (2011). Articulated pose estimation with flexible mixtures-of-parts. In IEEE Computer Vision and Pattern Recognition (CVPR), pp. 1385-1392.

[7] C. Geyer and K. Daniilidis (2000). A unifying theory for central panoramic systems and practical implications. In ECCV, pp. 445-461

[8] J.C. Bazin, C. Demonceaux, P. Vasseur and I.S. Kweon (2010). Motion estimation by decoupling rotation and translation in catadioptric vision. Journal of Computer Vision and Image Understanding, 114(2), 254-273.

[9] C. Mei, E. Sommerlade, G. Sibley, P. Newman and I. Reid (2011). Hidden view synthesis using realtime visual SLAM for simplifying video surveillance analysis. In IEEE International Conference on Robotics and Automation (ICRA), pp. 4240-4245.

[10] H. Hadj-Abdelkader, Y. Mezouar and P. Martinet (2009). Decoupled visual servoing based on the spherical projection of a set of points. In IEEE International Conference on Robotics and Automation (ICRA), pp. 1110-1115.

[11] K. K. Delibasis, S. V. Georgakopoulos, K. Kottari, V. P. Plagianakos and I. Maglogiannis (2016). Geodesically-corrected Zernike descriptors for pose recognition in omni-directional images. Integrated Computer-Aided Engineering, 23(2), 185-199.

[12] A. Elhayek, E. Aguiar, A. Jain, J. Tompson, L. Pishchulin, M. Andriluka and C. Theobalt (2016). MARCOnI-ConvNet-based MARker-less motion capture in outdoor and indoor scenes. IEEE transactions on pattern analysis and machine intelligence, 39(3), pp. 501-514.

[13] G. Caron, E.M. Mouaddib and E. Marchand (2012). 3D model based tracking for omnidirectional vision: A new spherical approach. Journal of Robotics and Autonomous Systems, 60(8), 1056-1068.

[14] Y. Yang and D. Ramanan (2013). Articulated human detection with flexible mixtures of parts. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(12), pp. 2878-2890.

[15] I. Kostrikov and J. Gall (2014). Depth Sweep Regression Forests for Estimating 3D Human Pose from Images. In BMVC, pp. 1-13.

[16] J. Gall, A. Yao, N. Razavi, L. Van Gool and V. Lempitsky (2011). Hough forests for object detection, tracking and action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(11), 2188-2202.

[17] M. Sanzari, V. Ntouskos and F. Pirri (2016). Bayesian image based 3D pose estimation. In European Conference on Computer Vision, pp. 566-582.

[18] H. Sidenbladh, M.J. Black and L. Sigal (2002). Implicit probabilistic models of human motion for synthesis and tracking. In European conference on computer vision, pp. 784-800.

[19] A. O. Balan, L. Sigal and M. J. Black (2005). A quantitative evaluation of video-based 3D person tracking. In proc. of 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, pp. 349-356.

[20] M. S. Arulampalam, S. Maskell, N. Gordon and T. Clapp (2002). A tutorial on particle fillters for online nonlinear/non-gaussian bayesian tracking. IEEE Transactions on signal processing, 50(2), pp. 174-188.

[21] C. Migniot and F. Ababsa (2013). 3D Human Tracking from Depth Cue in a Buying Behavior Analysis Context. International Conference on Computer Analysis of Images and Patterns (CAIP), pp. 482-489.

[22] C. Migniot and F. Ababsa (2013). 3D Human Tracking in a Top View Using Depth Information Recorded by the Xtion Pro-Live Camera. In International Symposium on Visual Computing (ISVC), pp. 603-612.

[23] M. Isard A. and Blake (1998). Condensation|conditional density propagation for visual tracking. International Journal of Computer Vision, 29(1), pp. 5-28.

[24] J. Deutscher, A. Blake and I. Reid, I. (2000). Articulated body motion capture by annealed particle filtering. In IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 126-133.

[25] https://ar-tracking.com/products/tracking-systems/smarttrack/

[26] H. Ning, W. Xu, Y. Gong and T. Huang (2008). Discriminative learning of visual words for 3d human pose estimation. In IEEE Computer Vision and Pattern Recognition (CVPR) 2008, pp. 1-8.

[27] R. Navaratnam, A. W. Fitzgibbon and R. Cipolla (2007). The joint manifold model for semi-supervised multi-valued regression. In IEEE International Conference on Computer Vision (ICCV), pp. 1-8.