Tracking Articulated Bodies using Generalized Expectation Maximization*

A. Fossati
CVLab
EPFL, Switzerland
andrea.fossati@epfl.ch

E. Arnaud
Université Joseph Fourier
INRIA Rhone-Alpes, France
elise.arnaud@inrialpes.fr

R. Horaud
Perception Group
INRIA Rhone-Alpes, France
radu.horaud@inrialpes.fr

P. Fua
CVLab
EPFL, Switzerland
pascal.fua@epfl.ch

Abstract

A Generalized Expectation Maximization (GEM) algorithm is used to retrieve the pose of a person from a monocular video sequence shot with a moving camera. After embedding the set of possible poses in a low dimensional space using Principal Component Analysis, the configuration that gives the best match to the input image is held as estimate for the current frame. This match is computed iterating GEM to assign edge pixels to the correct body part and to find the body pose that maximizes the likelihood of the assignments.

1. Introduction

Tracking objects in 3D using as input a video sequence captured using a single camera has been known to be a very under-contstrained problem. This is especially valid if the target to be tracked is a human body. Persons usually perform fast motions, wear loose clothing and generate lots of self-occlusions and visual ambiguities. Other difficulties may be caused by cluttered backgrounds and poor image resolution. The problem is particularly acute when using a single video captured with a moving camera to recover the 3D motion and existing approaches remain fairly brittle. To cope with the under-constrained characteristic of the problem, incorporating motion models as prior into the algorithms has been shown to be a reasonable and effective assumption to obtain good results [8]. The models can be physics-based [2] or learned from training data [11, 16, 10, 1, 15, 13]. Furthermore, an efficient algorithm should be able (i) to obtain reliable image observations from the person of interest, these observations should not be corrupted by the moving background, and (ii) to fit a learned body configuration to these observations in a robust manner.

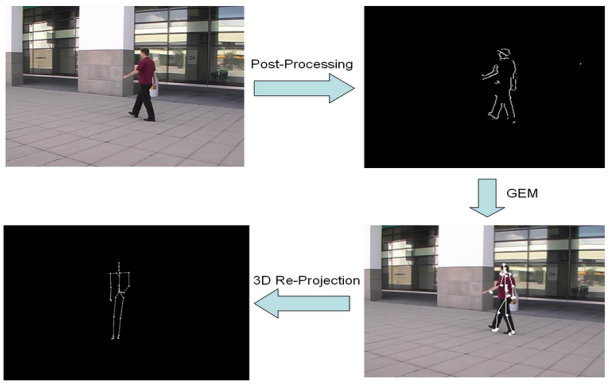

Figure 1. The full pipeline of the approach, from the input image to the 3D reprojection.

In this paper, we build on a recent work [6] that combines detection and tracking techniques to achieve 3D motion recovery of people seen from arbitrary viewpoints by a single camera. This algorithm helps us in having a first estimate of the body configuration for each image of the sequence, obtained by interpolating between detected key postures. Hence, the central idea of our work is to propose a robust framework to refine this first estimate that maximizes a novel image likelihood based on moving edge pixels. We first process the input sequence in order to obtain reliable edge information even if the camera is moving and thus no background subtraction is possible. We then use a Generalized Expectation Maximization (GEM) algorithm to iteratively assign the edge pixels to the correct body part and find the body configuration that maximizes the likelihood

*This work has been partially funded by the VISIONTRAIN RTN-CT-2004-005439 Marie Curie Action within the EC's Sixth Framework Programme. The text reflects only the authors' views and the Community is not liable for any use that may be made of the information contained therein.

of these assignments. This is done by fitting a mixture distribution to the set of observations. The proposed mixture is composed of a uniform law to cope with corrupted observations, and Gaussian laws, each one associated to one side of each body limb. Expectation-Maximization (EM) [4, 3] is a well established clustering technique and has been widely used in the Computer Vision community. It algorithm has been applied to the problem of articulated shape alignement with 2D image observations [12] or 3D data [5, 7]. GEM [4] is a variant of EM, that relaxes the maximization step into an optimization step.

In this paper, we propose to apply GEM to solve the problem of fitting the 2D projection of a 3D body cofiguration to a set of moving edge pixels. In our case the optimization is performed over parameters that define the 3D state of the person. We are therefore able to retrieve the full body pose and position in 3D even starting from single images. The exploration over the state space is constrained to a search over plausible configurations learned for a particular activity. This learning phase is performed using Principal Component Analysis, leading to embed the set of possible poses in a latent space of low dimensionality. The optimization is therefore performed over latent variables whose few dimensions keep the optimization problem tractable.

We demonstrate the effectiveness of the described approach on different sequences. The case of people walking along arbitrary trajectories is chosen. Persons who are not present in our motion database walk at different speeds and are seen from varying viewpoints, but are nonetheless accurately tracked in 3D. The results are also shown using a stick skeleton to demonstrate that the obtained results are fully 3D and can be reprojected to any viewpoint. The full pipeline of the approach is depicted in Fig. 1.

2. The Tracking Framework

The approach we have designed is structured as follows: first of all we obtain a reliable initial estimate of the 3D configuration of the person, using a key-pose detection technique together with the corresponding motion model, as suggested in [6]. Then we pre-process the video sequence we use as input in order to obtain a pretty clean edge image even if the camera is moving. Finally we use GEM to refine the initial pose estimation. This is done by matching the image edges to the edges obtained by projecting a 3D model of the person where limbs are considered as cylinders. We will explain in detail the 3 phases in the following subsections.

2.1. Pose Initialization

To obtain an initial estimate of the 3D pose of the person in each frame we adopt a techique presented in [6]. Basically it consists in detecting key-poses corresponding to a particular activity in few images of the video sequence. Since we consider walking in our case, the key-pose is chosen to be the particular pose when the legs are furthest apart, with the left leg in front of the right one. A rough estimate of the pose of the person in all the frames between two consecutive detections is then obtained: By interpolating the low-dimensional embedding of the activity through an appropriate motion model, an estimate value of the state $\mathbf{S} = \{\mathbf{P}, \mathbf{Q}\}$ of the person is retrieved. $\mathbf{P}$ is a 3-dimensional vector which represents the position and orientation of the body on a planar reference system coherent with the ground plane. $\mathbf{Q}$ is the set of the N joint angles in the body representation chain. In our experiments $N = 78$. In the case of the walking activity $\mathbf{Q}$ can be embedded in an n-dimensional PCA space. A body configuration is thus represented by the vector $\lambda = \{\mathbf{P}, \phi\}$ where ϕ is the latent variable of dimension n, with $n << N$, and a linear transformation relates ϕ to $\mathbf{Q}$. In practice, usually, $n = 3$.

2.2. Sequence Pre-Processing

To cope also with sequences shot by a moving camera, we decided to elaborate the input images in order to retrieve the edges corresponding to the moving objects. These are assumed to be the objects that move in the image at a different velocity than the background. This phase is composed of two main parts:

- *Motion Detection:* This step is taken from [9] and simply retrieves, using optical flow, which pixels in the image are used to estimate the global motion of the camera. It also retrieves which pixels are considered as outliers for this estimation, and these are the pixels on which we will focus our attention since they are the ones that move at a different velocity than the background.

- *Background matching:* To obtain a more robust estimate of the edges belonging to the foreground, we also adopt a homography-based technique. Assuming that the motion of the camera is not too fast and not too close to the scene, we can consider the background to be planar. We then can simply take a window of N frames centered around the current one I_t and match them to I_t using a standard approach based on robust estimation of homographies using keypoints. Then we extract the edges, using a Canny-style edge detector, from all the frames in the window. Finally we warp all the obtained edge images to match I_t, using the previously computed homographies. For all the pixels we will now have a set of N observations, which correspond to the same pixel being edge (1) or not (0) in the warped images. Now simply taking the median of these values for each pixel will tell us which edge pixels belong to the foreground (if the median is 0) and

which to the background (if the median is 1). At this point we have an estimate of the edge pixels belonging to the background at frame I_t, and simply subtracting this estimate from the edge extraction performed at I_t will give us an estimate of the edge pixels that belong to the foreground.

By making a simple intersection of the outputs of these two parts, for each input frame, we will obtain a pretty robust estimate of which pixels belong to the foreground and are at the same time part of some edges. All the parts of this pre-processing algorithm are summarized in Fig. 2. We will use the output of this procedure as input for the following phase. Note that this phase can easily be replaced by a standard background subtraction algorithm if the camera is not moving.

2.3. Pose refinement through GEM

2.3.1 Definitions

Before explaining how we plugged the GEM algorithm into our framework, some definitions are provided. The observations points $\mathbf{x} = \{x_1, \ldots, x_M\}$ are the points belonging to the contours obtained from the previous phase. Our goal is then to fit a body configuration to these observation points. To do so, we suppose that $\mathbf{x}$ is sampled from a 2D mixture distribution of K components (Gaussian laws) and an outlier component (uniform law). Each Gaussian is associated to one limb's side of the projected body pose. The parameters of the k^{th} Gaussian, i.e. its mean and covariance, are denoted as θ_k. Let us note that θ_k is a function of the the state $\mathbf{S}$ of the body, and therefore a function of λ. This parameterization is straightforward and is done as follows: From a given value of λ, the state $\mathbf{S}$, defined by the 3-dimensional body pose $\mathbf{P}$ and by the set of joint angles $\mathbf{Q}$, is used to generate a 3D representation of the human body. This representation has limbs which are considered as cylinders of different radius and length, dependending on the body part. Then this 3D model is projected onto the image and generates two segments for each cylinder, which should represent the 2 sides of the limb. Finally these segments are converted into Gaussian distributions, using the midpoint as representation of the mean and their length and a constant width to model an appropriate covariance matrix.

We then formalized the problem of fitting the projected body pose, now described as a Gaussian mixture, to the observed 2D cues as a classification task that could be carried out by the GEM algorithm. This problem boils down to the problem of finding an optimal value of λ such as the mixture components explain the image observation. The algorithm performs in 2 steps: First, each edge pixel is assigned to one of the components of the mixture. Let us note that a uniform component is added to the mixture to account for the corrupted observations. Second, the body configuration, i.e the mixture distribution, is fitted to the edge pixels by finding a new value of the parameter λ that decreases a distance function.

The assignement variables are denoted $\mathbf{z} = \{z_1, \ldots, z_M\}$. The event $z_m = k$, $m = 1, \ldots, M$, $k = 0, \ldots, K$ means that the observation x_m is generated from the k^{th} component of the mixture. The case $k = 0$ corresponds to the oulier case. By assuming conditional independence of the observations, we have:

$$p(\mathbf{x}|\mathbf{z}, \lambda) = \prod_{m=1}^{M} p(x_m|z_m, \lambda).$$

As explained before, the likelihood of an edge point being generated by the k^{th} limb's side is modeled as a Gaussian distribution of parameters $\theta_k(\lambda) = (\mu_k(\lambda), \Sigma_k(\lambda))$:

$$p(x_m|z_m = k, \lambda) = \mathcal{N}(x_m; \theta_k(\lambda)) \quad \text{if } (k \neq 0). \tag{1}$$

Similarly, we define the likelihood of an observation to belong to the outlier cluster as a uniform distribution:

$$p(x_m|z_m = 0, \lambda) = U[A] = \frac{1}{A}, \tag{2}$$

where A represent the observed data area i.e the image area. For simplicity, we assume that the assignement variables are independent. Their prior probabilities are denoted

$$p(z_m = k|\lambda) = p(z_m = k) = \pi_k \quad \forall k = 0, \ldots K$$

with

$$\sum_{k=0}^{K} \pi_k = 1$$

and therefore

$$\pi_k = \frac{1}{K+1}.$$

The components posterior probabilities are denoted as α_{mk}:

$$\alpha_{mk} = p(z_m = k|x_m, \lambda).$$

By applying Bayes' rule, we can obtain the following expression, where the observation likelihood are given by eq. (1-2):

$$\alpha_{mk} = \frac{\pi_k \, p(x_m|z_m = k, \lambda)}{\sum_{j=0}^{K} \pi_j \, p(x_m|z_m = j, \lambda)}.$$

For $k = 1, \ldots, K$, we have:

$$\alpha_{mk} = \frac{\pi_k |\Sigma_k(\lambda)|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\|x_m - \mu_k(\lambda)\|^2_{\Sigma_k(\lambda)}\right)}{\frac{2\pi\pi_0}{A} + \sum_{j=1}^{K} \pi_j |\Sigma_j(\lambda)|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\|x_m - \mu_j(\lambda)\|^2_{\Sigma_j(\lambda)}\right)}, \tag{3}$$

where the notation $\| \mathbf{a} - \mathbf{b}\|^2_{\Sigma} = (\mathbf{a} - \mathbf{b})^T \Sigma^{-1} (\mathbf{a} - \mathbf{b})$ accounts for for the Mahalanobis distance. For $k = 0$, we have:

$$\alpha_{m0} = 1 - \sum_{k=1}^{K} \alpha_{mk}. \tag{4}$$

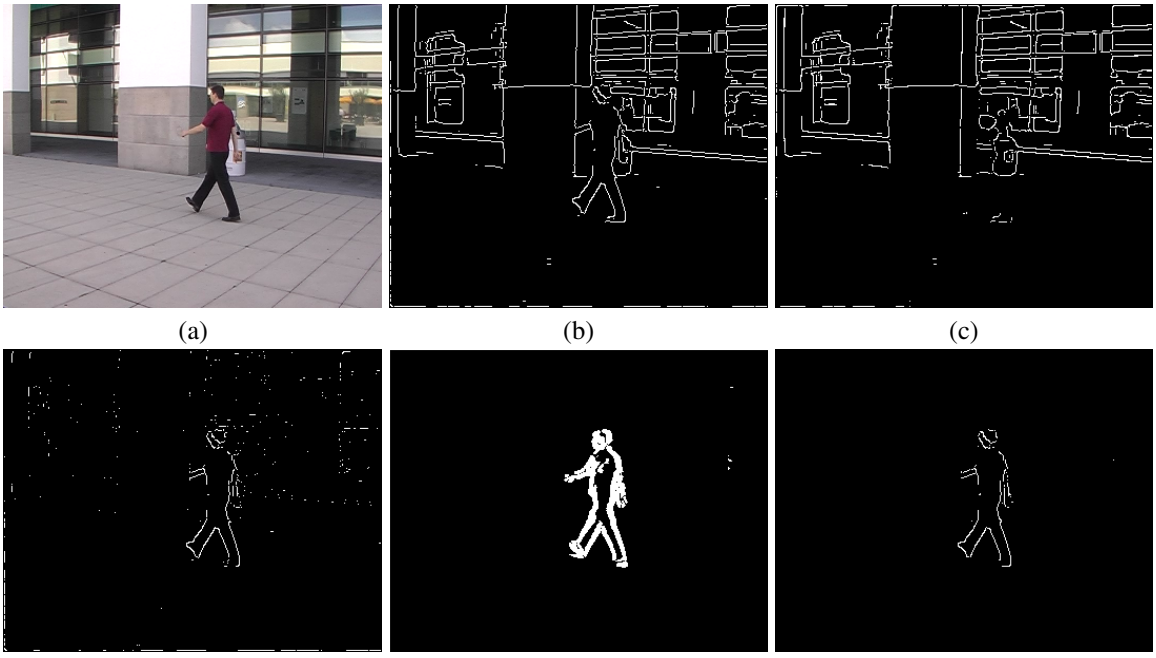

Figure 2. Summary of the pre-processing algorithm: (a) Input image. (b) Edges extracted from the input image. (c) Background edges reconstructed through homographies. (d) Subtraction between (b) and (c). (e) Ouliers retrieved by the camera motion estimation technique. (f) Final output of the algorithm, obtained as intersection between (d) and (e).

2.3.2 GEM framework

Given the probabilistic model defined above, the goal is to determine the value of λ whose associated mixture distribution better explains the observations $\mathbf{x}$. Treating assignements as the hidden variables, the GEM algorithm helps in achieving this goal by maximizing the joint probability $p(\mathbf{x}, \mathbf{z}|\lambda)$. This probabiliy can be written as:

$$\begin{aligned} p(\mathbf{x}, \mathbf{z}|\lambda) &= p(\mathbf{x}|\mathbf{z}, \lambda)\, p(\mathbf{z}, \lambda) \\ &= \prod_{m=1}^{M} p(x_m|z_m, \lambda)\, p(z_m|\lambda) \\ &= \prod_{m=1}^{M} \prod_{k=0}^{K} [\pi_k\, p(x_m|z_m = k, \lambda)]^{\delta_k(z_m)} \end{aligned} \quad (5)$$

The random variables $\delta_k(z_m)$ are defined as follows:

$$\delta_k(z_m) = \begin{cases} 1 & \text{if } z_m = k \\ 0 & \text{otherwise} \end{cases}$$

Starting with the initial value $\lambda^{(0)}$, the GEM algorithm proceeds iteratively and the iteration t consists in searching for the parameters λ that optimize the following expression:

$$Q(\lambda|\lambda^{(t)}) = E[\log p(\mathbf{x}, \mathbf{z}|\lambda)|\mathbf{x}, \lambda^{(t)}],$$

where $\lambda^{(t)}$ is the current estimate at iteration t. The expectation is calculed over all the possible assignements $\mathbf{z}$. Using eq (5), we have:

$$\log p(\mathbf{x}, \mathbf{z}|\lambda) = \sum_{m=1}^{M} \sum_{k=0}^{K} \log(\pi_k\, p(x_m|z_m = k, \lambda))\, \delta_k(z_m)).$$

Remarking that:

$$E[\delta_k(z_m)|\mathbf{x}, \lambda^{(t)}] = \sum_{k=0}^{K} \delta_k(z_m)\, p(z_m = k|\mathbf{x}, \lambda^{(t)}) = \alpha_{mk}^{(t)},$$

where $\alpha_{mk}^{(t)}$ are the posterior likelihood calculated using eq. (3-4) with $\lambda = \lambda^{(t)}$, we have:

$$Q(\lambda|\lambda^{(t)}) = \sum_{m=1}^{M} \sum_{k=0}^{K} \alpha_{mk}^{(t)}\ \log(\pi_k\, p(x_m|z_m = k, \lambda)).$$

Replacing the likelihoods by their expression given by eq. (1-2) leads to:

$$\begin{aligned} Q(\lambda|\lambda^t) \quad = \quad & \sum_{m=1}^{M} \sum_{k=1}^{K} \alpha_{mk}^{(t)} \left\{ -\frac{1}{2}\, \|x_m - \mu_k(\lambda)\|^2_{\Sigma_k(\lambda)} \right. \\ & \left. - \log\left(\pi_k\, (2\pi_k)^{-1}\, |\Sigma_k(\lambda)|^{-1/2}\right) \right\} \\ + \quad & \sum_{m=1}^{M} \alpha_{m0}^{(t)}\ \log(A\, \pi_0) \end{aligned} \quad (6)$$

We can now formulate the GEM algorithm as iterations of two steps at time t:

- **E-step** From the current value $\lambda^{(t)}$, this step simply requires the computation of the posterior probabilities $\alpha_{mk}^{(t)}$ using eq. (3-4). Each probability $\alpha_{mk}^{(t)}$ represents the likelihood of assigning observation point m to the k^{th} limb's side or to the outlier class.

- **M-step** Provided that $\alpha_{mk}^{(t)}$ are computed, now $Q(\lambda|\lambda^t)$ needs to be maximized over λ. Since the analytical computation would be higly non-linear, the generalized version of the EM algorithm is applied. This simply means that, instead of maximizing $Q(\lambda|\lambda^t)$, we simply find a state $\lambda^{(t+1)}$ that increases the value of $Q(\lambda|\lambda^t)$. In pratice, several λ_i are sampled around $\lambda^{(t)}$ until this condition is reached.

We iterate this procedure a certain number of times until an improvement in Q is obtained, and than retain the corresponding body pose calculated from $\lambda^{(final)}$ for the current frame.

3. Results

We now presents some results we obtained by applying the full framework explained above to different sequences. All the sequences have been captured in non-engineered outdoor scenes and the camera has been kept in motion throughout all of them. Unfortunately, due to the intrinsic strength of the approach, we are not able to provide quantitative results but just qualitative evaluations. In fact, no techniques to collect ground truth data in such difficult conditions are available at the moment.

In Fig. 3 and in Fig. 4 there are two persons in the scene but only one is tracked. The same procedure could have been applied to the other subject to obtain distinct tracking results for both of them. In Fig. 5 the walking subject is undergoing a slight viewpoint change but this does not have inlfuence on the tracking results. For this sequence we also provide the ouput of the pre-processing phase that we used as input to obtain the shown results. In all the three cases we obtain a good reprojection of the 3D model limbs onto the limbs of the tracked subject, and also the reconstructed 3D pose looks plausible.

4. Conclusion

In this paper we have presented an approach to retrieve the 3D pose of a person using single viewpoint sequences, shot with a moving camera in everyday life environments. To this end we first initialize the body pose with the help of a motion model and then refine it using a novel Generalized Expectation Maximization algorithm. This algorithms has the task of assigning the contour pixels, obtained from the input images after a few pre-processing steps, to the corresponding body part. It also correctly finds the outlier pixels assigning them to a special class. The pose that gives the best match between the image measurements and the body parts is finally kept as output.

This framework is promising and gives good results in the walking case. We now plan to extend it to track persons performing different activities. Another direction we will follow is to test the algorithm also on constrained environment sequences, for which ground truth data are available, such as the HumanEva dataset [14]. This will provide us useful quantitative evaluations.

References

[1] A. Agarwal and B. Triggs. Tracking articulated motion with piecewise learned dynamical models. In *ECCV*, 2004.

[2] M. Brubaker, D. Fleet, and A. Hertzmann. Physics-based person tracking using simplified lower-body dynamics. In *CVPR*, pages 1–8, 2007.

[3] G. Celeux, F. Forbes, and N. Peyrard. Em procedures using mean field-like approximations for markov model-based image segmentation, 2003.

[4] A. Dempster, N. Laird, and D. Rubin. Maximum Likelihood from Incomplete Data via the EM algorithm. In *Journal of the Royal Statistical Society, Series B*, 1977.

[5] G. Dewaele, F. Devernay, R. Horaud, and F. Forbes. The alignment between 3-d data and articulated shapes with bending surfaces. In *ECCV*, pages 578–591, 2006.

[6] A. Fossati, M. Dimitrijevic, V. Lepetit, and P. Fua. Bridging the Gap between Detection and Tracking for 3D Monocular Video-Based Motion Capture. In *CVPR*, 2007.

[7] C. Ménier, E. Boyer, and B. Raffin. 3d skeleton-based body pose recovery. In *Proceedings of 3DPVT*, 2006.

[8] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Comput. Vis. Image Underst.*, 104(2):90–126, 2006.

[9] J. Odobez and P. Bouthemy. Separation of moving regions from background in an image sequence acquired with a mobile camera. *Video Data Compression for Multimedia Computing*, pages 283–311, 1997.

[10] D. Ormoneit, H. Sidenbladh, M. Black, and T. Hastie. Learning and tracking cyclic human motion. In *NIPS*, 2001.

[11] B. Rosenhahn, T. Brox, and H. Seidel. Scaled motion dynamics for markerless motion capture. In *CVPR*, 2007.

[12] H. Rowley and J. Rehg. Analyzing articulated motion using expectation-maximization. In *CVPR*, 1997.

[13] H. Sidenbladh, M. J. Black, and L. Sigal. Implicit Probabilistic Models of Human Motion for Synthesis and Tracking. In *ECCV*, Copenhagen, Denmark, May 2002.

[14] L. Sigal and M. Black. Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical report, Brown University, 2006.

[15] L. Taycher, G. Shakhnarovich, D. Demirdjian, and T. Darrell. Conditional Random People: Tracking Humans with CRFs and Grid Filters. In *CVPR*, 2006.

[16] R. Urtasun, D. Fleet, and P. Fua. 3D People Tracking with Gaussian Process Dynamical Models. In *CVPR*, 2006.

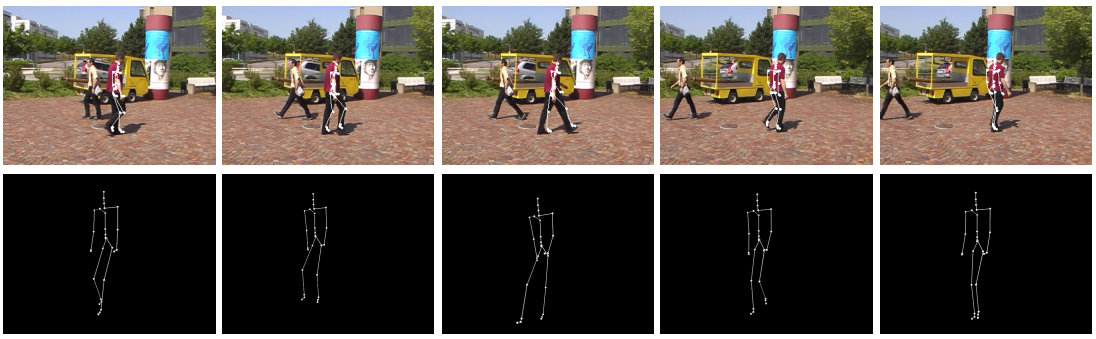

Figure 3. Pedestrian tracking and reprojected 3D model. **First row:** Frames from the input video. The recovered body pose has been reprojected on the input image. **Second row:** The 3D skeleton of the person is seen from a different viewpoint, to highlight the 3D nature of the results.

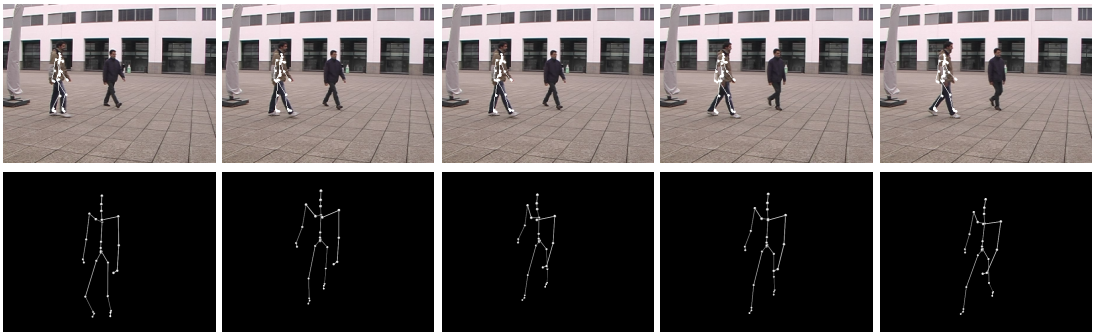

Figure 4. Pedestrian tracking and reprojected 3D model. **First row:** Frames from the input video. The recovered body pose has been reprojected on the input image. **Second row:** The 3D skeleton of the person is seen from a different viewpoint, to highlight the 3D nature of the results.

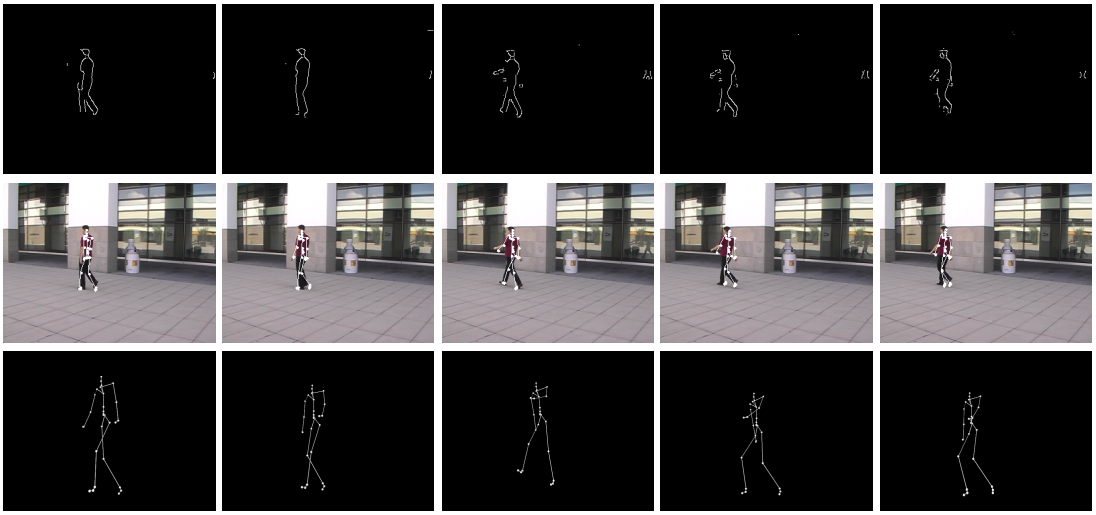

Figure 5. Post-processing output, pedestrian tracking and reprojected 3D model. **First row:** Output of the pre-processing phase, that is used as input for the following phases. **Second row:** Frames from the input video. The recovered body pose has been reprojected on the input image. **Third row:** The 3D skeleton of the person is seen from a different viewpoint, to highlight the 3D nature of the results.