Computing View-normalized Body Parts Trajectories

Frédéric Jean   Robert Bergevin
Computer Vision and Systems Laboratory
Laval University
Quebec, Canada
{fjean, bergevin}@gel.ulaval.ca

Alexandra Branzan Albu
Computer Vision Laboratory
University of Victoria
Victoria, Canada
aalbu@ece.uvic.ca

Abstract

This paper proposes an approach to compute view-normalized body part trajectories of pedestrians from monocular video sequences. The proposed approach first extracts the 2D trajectories of both feet and of the head from tracked silhouettes. On that basis, it segments the walking trajectory into piecewise linear segments. Finally, a normalization process is applied to head and feet trajectories over each obtained straight walking segment. View normalization makes head and feet trajectories appear as if seen from a fronto-parallel viewpoint. The latter is assumed to be optimal for gait modeling and recognition purposes. The proposed approach is fully automatic as it requires neither manual initialization nor camera calibration.

1. Introduction

A well-known medical study has shown that human gait is a complex motion that may be decomposed into twenty-four components [11]. It is believed that the complexity of the interactions between the various components encodes relevant information about the identity of the moving person. Recent progress in computer-based analysis of gait has confirmed its potential as a biometric feature. First and foremost, gait analysis allows for person identification at a distance, which is difficult or even impossible with other biometric techniques such as retinal scanning, fingerprints, or face recognition.

Gait-based person identification represents a key element in the design of robust visual surveillance systems. However, recently proposed computer-based surveillance systems [2] have focused mostly on gait detection [3] and low-level tracking of human subjects using basic appearance models [6]. More sophisticated 3D part-based models may also be obtained but they require manual interventions for initialization [5]. Besides, stereo data is required for methods using 3D temporal motion models [13]. Finally, 3D

models are computationally expensive and therefore difficult to use in real-time surveillance.

In many realistic settings, for instance when pedestrians are observed in extended premises via a network of loosely-coupled nodes [1], a more efficient and automatic modeling approach is required. Therefore, a 2D model-based method is proposed here, as it is more easily integrated in such a real-time visual surveillance system.

A key surveillance issue addressed by the proposed approach is the varying angle between the camera optical axis and the walking trajectory direction of an observed pedestrian. This phenomenon introduces a variation of the human motion captured with an uncalibrated camera. In practice, many gait modeling approaches are either only applicable to fronto-parallel viewpoints [16, 14], or at least view-dependent [15]. The method proposed in [12] compensates angular errors due to motion projection for monocular sequences acquired from different viewpoints. That method assumes the trajectory angle to be known a priori, which can hardly apply in a general surveillance context. Synthesis of a canonical side view from an arbitrary view is performed in [9] via two methods, namely perspective projection and optical flow-based structure-from-motion. However, the synthesis of a side view is only feasible from a limited number of initial views. The method in [8] involves a scaling process, for each known view, on silhouette parameters such as height and distance between head and pelvis. Therefore, applicability of those methods to monitoring unconstrained human walk is limited.

The approach proposed in this paper aims at computing body part trajectories that are invariant to walking trajectory and viewpoint. The trajectory of a body part (foot, head, hand) is defined as a sequence of the successive 2D positions it takes in the frames of a video sequence. On a frame-by-frame basis, each body part is represented by one point respectively. Body part trajectories are assumed to contain sufficient information about the gait of a person for view-invariant modeling and recognition. The walking trajectory, which is the path followed by a person on the floor, is not
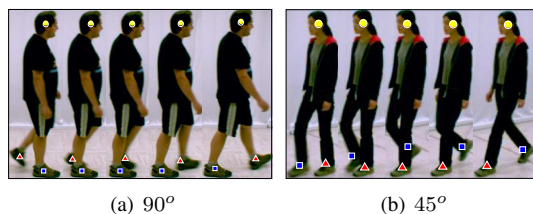
(a) 90° (b) 45°

Figure 1. Body part tracking

assumed to be a single straight line. Instead, it is assumed to be a polyline that is, a sequence of straight-line segments of variable orientations and lengths. View normalization consists in making body part trajectories appear as if seen from the same fronto-parallel viewpoint for all straight-line walking segments of the video sequence. The proposed approach to view normalization features automatic initialization, no camera calibration, as well as a low computational complexity.

The approach is detailed in Section 2. Experimental results are presented in Section 3. Finally, Section 4 draws conclusions and outlines the main directions of future work.

2. Proposed Approach

The proposed approach requires preprocessing. This is described next. Then, an overview of the approach is presented. A detailed discussion of the corresponding algorithms completes the section.

2.1. Preprocessing

The algorithms introduced in [7] produce the original head and feet trajectories that is, body part positions at each frame of the input video sequence. Temporal information helps achieve body part correspondence between frames. It does so while handling sporadical foot occlusion resulting from observing the pedestrian from a given angle. Fig. 1 is an excerpt of head and feet positions obtained for two pedestrians observed from different viewpoints. Squares and triangles represent feet positions while disks represent head positions. Correspondence is achieved since a symbol associated with a given foot remains on it despite intervening occlusion.

The proposed approach needs head and feet trajectories in order to compute the normalizing transformation. However, once the transformation is computed, it may be applied to other body part trajectories. In this paper, results are presented only for the normalized trajectories of the head and the feet.
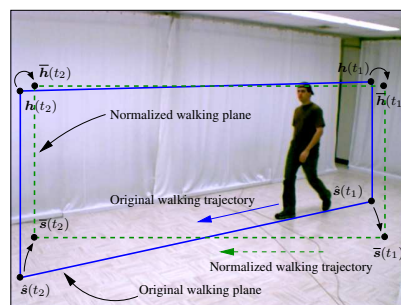

Figure 2. View normalization

2.2. Overview

Fig. 2 presents a situation where a person follows a straight walking trajectory. As the camera is fixed, consecutive frames of the sequence have similar backgrounds but varying foreground. In this case, the person is to occupy different positions with different postures while walking from the right-hand side at time t_1 to the left-hand side at time t_2. The walking trajectory angle with respect to the camera optical axis is about $45°$. The "original walking plane" is formed by joining corresponding positions along the walking trajectory and head trajectory (the method used to estimate the walking trajectory is described below). This plane indicates how the pedestrian is positioned with respect to the camera over a given time slice. Here, he is far from the camera at right and closer to the camera at left. Hence, the plane edges are not parallel to the image frame.

A "normalized plane" has edges parallel to the image frame. The original plane is normalized when the walking trajectory is fronto-parallel that is, when the straight walking trajectory is perpendicular to the optical axis of the camera. In other cases, such as the one illustrated in Fig. 2, an original plane may be normalized using an homography-based transformation. The homography matrix may be computed once a correspondence is established between the four corners (top and bottom positions) of the original plane ($\boldsymbol{h}(t_1)$, $\hat{\boldsymbol{s}}(t_1)$, $\boldsymbol{h}(t_2)$ and $\hat{\boldsymbol{s}}(t_2)$) and four corners of the normalized plane ($\overline{\boldsymbol{h}}(t_1)$, $\overline{\boldsymbol{s}}(t_1)$, $\overline{\boldsymbol{h}}(t_2)$ and $\overline{\boldsymbol{s}}(t_2)$) at time t_1 and t_2. The computed homography may be applied to the coordinates of the body parts between time t_1 and t_2 in order to transform them from the original plane to the normalized plane. Normalized body part trajectories appear as obtained from a fronto-parallel viewpoint. An important assumption behind the normalization is that the motion of each foot occurs in a plane parallel and close to the original walking plane. This assumption holds better when the distance to the camera is large compared to the size of the pedestrian.

In realistic surveillance situations, pedestrians cannot be assumed to always follow a single straight line. Besides,

their walking trajectory cannot be known in advance. The proposed approach deals with both difficulties by first estimating the walking trajectory using the original feet trajectories. The estimated walking trajectory is then "spatiotemporally" decomposed into piecewise linear segments. Next, original and normalized plane parameters are computed for each of those segments. Finally, body part trajectories of each segment are normalized using the computed homography between the corresponding planes.

2.3. Estimating the Walking Trajectory

The walking trajectory is estimated using feet trajectories. One a frame-by-frame basis, the walking trajectory consists in one point defined by its 2D coordinates in the image plane. As shown in Fig. 3, the estimated walking trajectory appears as a series of segments separated by local discontinuities caused by temporary feet self-occlusion. The main issue in estimating the walking trajectory is the selection of the representative point. It has been found that the vertical projection $\hat{s}$ of the silhouette mass center $\boldsymbol{s} = [s_x , s_y]^T$ on the line joining the feet positions on the floor is a reliable global estimate of the walking trajectory. The x-coordinate of $\hat{s}$ is identical to s_x, while the y-coordinate is computed as follows. The position of each foot on the floor ($\hat{\boldsymbol{f}}^1$ and $\hat{\boldsymbol{f}}^2$) is obtained by vertically projecting its mass center ($\boldsymbol{f}^1$ and $\boldsymbol{f}^2$) to the lowest silhouette pixel at that x position: $\hat{\boldsymbol{f}}^1 = \left[f_x^1 , \mathcal{Y}(f_x^1) \right]^T$, $\hat{\boldsymbol{f}}^2 = \left[f_x^2 , \mathcal{Y}(f_x^2) \right]^T$. In these equations, $\mathcal{Y}(f_x^1)$ and $\mathcal{Y}(f_x^2)$ represent the lowest y pixel positions on the silhouette at the x position of the mass center of foot 1 and 2, respectively. The projected silhouette mass center is then computed as :

$$\hat{\boldsymbol{s}} = \left[s_x , \; \frac{\hat{f}_y^2 - \hat{f}_y^1}{\hat{f}_x^2 - \hat{f}_x^1}(s_x - \hat{f}_x^1) + \hat{f}_y^1 \right]^T \tag{1}$$

The projected silhouette mass center provides a good sample on the walking trajectory as long as both feet are touching the floor. When one foot is moving, it could still be close to the floor. In that case, the resulting error is acceptable. However, when one foot is occluded, the moving foot is usually farther from the floor and a large error could be introduced. The tracking algorithm computing the original body part trajectories detects feet occlusions. It is thus possible to project the mass center of the silhouette only when both feet are visible and close to the floor. As a result, piecewise-continuous walking trajectory samples are obtained as shown in Fig. 3. The segmentation of the estimated walking trajectory into linear trajectory segments is the next step of the algorithm.
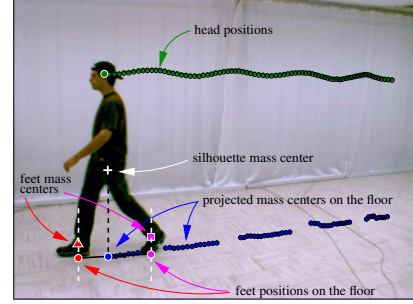

Figure 3. A typical walking trajectory

2.4. Segmenting the Walking Trajectory

The trajectory segmentation algorithm is to be explained using results for a sequence with a single sudden walking direction change (Fig. 4). The original head and feet trajectories obtained by the tracking algorithm are shown in Fig. 4(a). Fig. 4(b) presents the piecewise-continuous walking trajectory samples obtained by projecting the silhouette's mass center on the floor. The idea of the segmentation algorithm is to fit a straight-line segment to each continuous group of samples and estimate junction points linking consecutive segments (see Fig. 4(c)). The obtained samples are typically imperfect as a result of simplyfing assumptions. Hence, junction points cannot be obtained by simply computing line intersections. The computed junction points are next considered as samples on a curve to be approximated by a polyline (open polygon). The number of straight-line segments in the polyline should match the number of straight walking segments in the pedestrian's trajectory. That is, significant corners have to be identified along the junction-sampled curve.

A group $\mathcal{G}^k$ of samples covers an interval $[t_b^k, t_e^k]$, where t_b^k is the first sample after an occlusion and t_e^k is the last sample before the next occlusion. $N_{\mathcal{G}}$ is the number of groups and $k = 1 \ldots N_{\mathcal{G}}$. In Fig. 4(b), there are seven such groups of samples. Most cases where a pedestrian changes direction imply a feet occlusion event that is, one foot is temporarily occluded by the rest of the body. Hence, it is assumed that direction changes occur between groups of continuous samples. A junction point $\boldsymbol{j}^l$ is computed between each consecutive group of samples $\mathcal{G}^l$ and $\mathcal{G}^{l-1}$. There are $N_l = N_{\mathcal{G}} + 1$ junction points, with special cases $\boldsymbol{j}^1 = \hat{\boldsymbol{s}}(t_b^1)$, the first projected mass center of the first group, and $\boldsymbol{j}^{N_l} = \hat{\boldsymbol{s}}(t_e^{N_l-1})$, the last projected mass center of the last group. Intermediate junction points $\boldsymbol{j}^l$, $l = 2, \ldots, N_l - 1$ are computed as follows (see Fig. 5):

$$j_x^l = \left(\hat{s}_x(t_b^l) + \hat{s}_x(t_e^{l-1}) \right) / 2 \tag{2}$$

where j_x^l is the x coordinate of the junction point, $\hat{s}_x(t_b^l)$ is the first point of the group $\mathcal{G}^l$ and $\hat{s}_x(t_e^{l-1})$ is the last
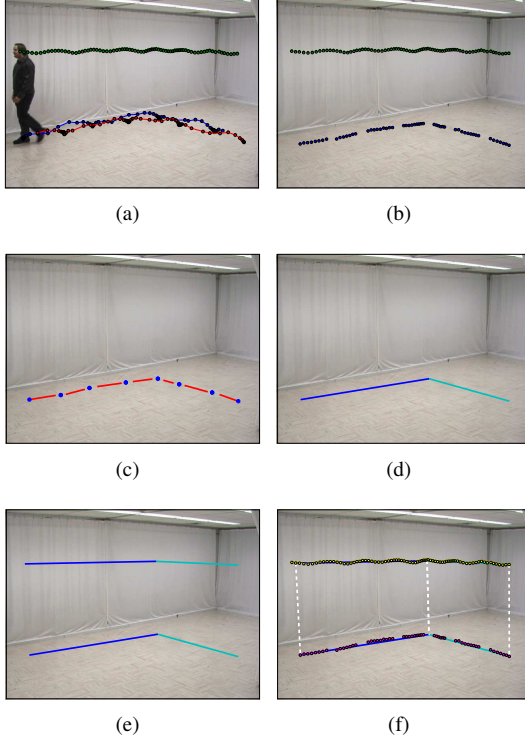
Figure 4. Walking trajectory segmentation

Figure 5. Computing junction points

point of the group $\mathcal{G}^{l-1}$. The x coordinate is simply the middle x position between these two x coordinates. For the computation of the y coordinate, the fitted lines $\mathcal{L}^{l-1}$ and $\mathcal{L}^l$ are first used to extrapolate missing positions of projected mass center due to the feet occlusion event. The number of missing samples between the two groups is computed as:

$$\Delta t = t_b^l - t_e^{l-1} \tag{3}$$

To extrapolate the two lines, missed samples are first split between the two groups :

$$\Delta t^l = \lfloor (\Delta t - 1)/2 \rfloor, \quad \Delta t^{l-1} = \lceil (\Delta t - 1)/2 \rceil. \tag{4}$$

The horizontal distance $\Delta x = \left| \hat{s}_x(t_b^l) - \hat{s}_x(t_e^{l-1}) \right|$ between point $\hat{s}(t_e^{l-1})$ and point $\hat{s}(t_b^l)$ is then split between the two groups according to the number of missing samples associated with each one:

$$\Delta x^l = \left(\Delta x \Delta t^l \right) / \Delta t \tag{5}$$

and

$$\Delta x^{l-1} = \left(\Delta x \Delta t^{l-1} \right) / \Delta t. \tag{6}$$

The junction y coordinate is then computed as :

$$j_y^l = \frac{\mathcal{L}_{\hat{s}}^l \left[\hat{s}_x(t_b^l - \Delta t^l) \right] + \mathcal{L}_{\hat{s}}^{l-1} \left[\hat{s}_x(t_e^{l-1} + \Delta t^{l-1}) \right]}{2} \tag{7}$$

where

$$\hat{s}_x(t_b^l - \Delta t^l) = \hat{s}_x(t_b^l) + a\Delta x^l, \tag{8}$$
$$\hat{s}_x(t_e^{l-1} + \Delta t^{l-1}) = \hat{s}_x(t_e^{l-1}) - a\Delta x^{l-1}, \tag{9}$$

and a is the sign of $\left\{ \hat{s}_x(t_e^{l-1}) - \hat{s}_x(t_b^l) \right\}$. $\mathcal{L}_{\hat{s}}^l [x]$ represents the y coordinate at x on line $\mathcal{L}_{\hat{s}}^l$. The y coordinate of the junction point is therefore at mid-distance from the two points extrapolated from the lines fitted to the groups $\mathcal{G}^l$ and $\mathcal{G}^{l-1}$. The x coordinate of the junction is the average of last sample of $\mathcal{G}^{l-1}$ and first sample of $\mathcal{G}^l$. Six computed intermediate junction points are displayed in Fig. 4(c).

A classical iterative polyline fitting algorithm [4] is used next with ordered junction points acting as consecutive curve samples. The main steps of the algorithms are as follows:

1. $\mathcal{I} \leftarrow \{1\}$

2. $b = 1, e = N_l$

3. Draw a line linking junction points $\boldsymbol{j}^b$ and $\boldsymbol{j}^e$.

4. Compute the distances from junction points $\boldsymbol{j}^{b+1}$ to $\boldsymbol{j}^{e-1}$ to the line.

5. Determine the junction point $\boldsymbol{j}^d$ ($b < d < e$) whose distance to the line is maximal.

6. If that distance is above a predefined threshold T_d then $\mathcal{I} \leftarrow \mathcal{I} \cup \{d\}$ and repeat step 3 twice for $b = b, e = d$ and $b = d, e = e$.

7. $\mathcal{I} \leftarrow \mathcal{I} \cup \{N_l\}$

Different results may be obtained according to the value selected for T_d. In our experiments, a single default value was identified and applied to all sequences. A set $\mathcal{I}$:

COMPUTER
SOCIETY

$\{i_1, i_2, \ldots, i_{N_\mathcal{I}}\}$ of $N_\mathcal{I}$ junction indices is obtained that correspond to positions where the walking trajectory is to be segmented. The walking trajectory is approximated by a polyline whose corners are the retained junction points. The latter are denoted as $\boldsymbol{j}_r^m = \boldsymbol{j}^{i_m}$, $m = 1, \ldots, N_m$ for $N_m = N_\mathcal{I}$. A frame number is associated with each retained junction point: $t_r^m = t_b^{i_m} - \Delta t^{i_m}$ for $m = 2, \ldots, N_m - 1$, and the special case $t_r^1 = t_b^1$ and $t_r^{N_m} = t_e^{N_l - 1}$. In Fig. 4(d), the default threshold value produced four retained junction points whose indices are 1, 4, and 8.

Once the walking trajectory is segmented, the same frame indices are used to segment the head trajectory. Defining $\boldsymbol{h}(t)$ as the head position at time t, a straight-line segment $\mathcal{L}_h^m$ is fitted to the corresponding groups of head points. The obtained segments are then used to compute junction points $\boldsymbol{q}^m$ for the head trajectory :

$$q_x^m = \{h_x(t_r^m) + h_x(t_r^m - 1)\}/2, \qquad (10)$$

$$q_y^m = \left\{\mathcal{L}_h^m\left[h_x(t_r^m)\right] + \mathcal{L}_h^{m-1}\left[h_x(t_r^m - 1)\right]\right\}/2, \quad (11)$$

where $\mathcal{L}_h^m[x]$ represents the y coordinate at the x position on the line $\mathcal{L}_h^m$. This is computed for $m = 2, \ldots, N_m - 1$ only, with special cases $\boldsymbol{q}^1 = \boldsymbol{h}(t_r^1)$ and $\boldsymbol{q}^{N_m} = \boldsymbol{h}(t_r^{N_m})$. As for the projected mass center, these junction points have a x coordinate at mid-distance from $h_x(t_r^m)$ and $h_x(t_r^m - 1)$, and a y coordinate at mid-distance from the two extrapolated y coordinates. The extrapolated y coordinates produce accurate junction points, assuming that the head trajectory is sinusodal (the original y coordinates $h_y(t_r^m)$ and $h_y(t_r^m - 1)$ are not used). Finally, consecutive head junction points are linked to form a polyline approximating the head trajectory. Fig. 4(e) presents the resulting approximated head trajectory. One can see in Fig. 4(f) that the approximated trajectories fit well the original head and walking trajectories. Corresponding junction points are linked with dashed lines to show the two estimated original walking planes.

2.5. Computing Plane Parameters

An original walking plane is defined by two head junction points and two walking trajectory junction points. The planes are denoted Π^p for $p = 1, \ldots, N_p$, $N_p = N_m - 1$, and the four corners are defined as $\boldsymbol{\pi}_{\mathrm{BB}}^p = \boldsymbol{j}_r^p$ (beginning bottom position), $\boldsymbol{\pi}_{\mathrm{EB}}^p = \boldsymbol{j}_r^{p+1}$ (ending bottom position), $\boldsymbol{\pi}_{\mathrm{BT}}^p = \boldsymbol{q}^p$ (beginning top position) and $\boldsymbol{\pi}_{\mathrm{ET}}^p = \boldsymbol{q}^{p+1}$ (ending top position).

Once the four corners of the planes are known, planes are ready to be normalized. Fig. 6 shows the first normalization step of a plane Π^p transformed into a plane $\hat{\Pi}^p$. Defining $d^p = \|\boldsymbol{\pi}_{\mathrm{EB}}^p - \boldsymbol{\pi}_{\mathrm{BB}}^p\|$ the length of the line segment in the approximated walking trajectory, the corners of the new plane are defined as :

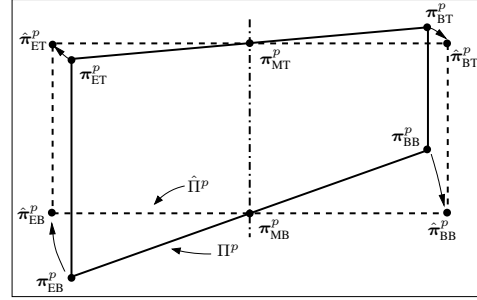**Figure 6. Plane normalization**

$$\hat{\boldsymbol{\pi}}_{\mathrm{BB}}^p = \left[\pi_{\mathrm{MB},\,x}^p + uc^p d^p/2,\ \pi_{\mathrm{MB},\,y}^p\right]^{\mathrm{T}} \qquad (12)$$

$$\hat{\boldsymbol{\pi}}_{\mathrm{EB}}^p = \left[\pi_{\mathrm{MB},\,x}^p - uc^p d^p/2,\ \pi_{\mathrm{MB},\,y}^p\right]^{\mathrm{T}} \qquad (13)$$

$$\hat{\boldsymbol{\pi}}_{\mathrm{BT}}^p = \left[\pi_{\mathrm{MT},\,x}^p + uc^p d^p/2,\ \pi_{\mathrm{MT},\,y}^p\right]^{\mathrm{T}} \qquad (14)$$

$$\hat{\boldsymbol{\pi}}_{\mathrm{ET}}^p = \left[\pi_{\mathrm{MT},\,x}^p - uc^p d^p/2,\ \pi_{\mathrm{MT},\,y}^p\right]^{\mathrm{T}} \qquad (15)$$

where

$$\boldsymbol{\pi}_{\mathrm{MB}}^p = (\boldsymbol{\pi}_{\mathrm{EB}}^p + \boldsymbol{\pi}_{\mathrm{BB}}^p)/2\,,\quad \boldsymbol{\pi}_{\mathrm{MT}}^p = (\boldsymbol{\pi}_{\mathrm{ET}}^p + \boldsymbol{\pi}_{\mathrm{BT}}^p)/2 \quad (16)$$

are the middle points of the top and bottom lines, respectively. Parameter c^p is the sign of $(\pi_{\mathrm{EB},\,x}^p - \pi_{\mathrm{BB},\,x}^p)$ and u is a parameter taking value 1 or -1 defined by the user which indicates the direction preference of the body parts trajectories progression. Using $u = 1$, the progression will be from left to right, while for $u = -1$ the progression will be from right to left. Depending on the relative position of components $\pi_{\mathrm{EB},\,x}^p$ and $\pi_{\mathrm{BB},\,x}^p$, the relative x positions of the plane corners will be switched if $\pi_{\mathrm{EB},\,x}^p < \pi_{\mathrm{BB},\,x}^p$ and $u = 1$, or if $\pi_{\mathrm{EB},\,x}^p > \pi_{\mathrm{BB},\,x}^p$ and $u = -1$. This switching is necessary for body part trajectories normalization since it makes normalized trajectories appear as if the person had walked along a single direction, even when the walking trajectory includes a change in x direction. As shown in Fig. 6, a new plane is formed by transforming the bottom and top borders of the original walking plane so they become parallel to the image frame. The bottom edge keeps its length while the top edge also has the same length. As a result, side edges become parallel to the image borders too.

Scaling and shifting is applied to the obtained normalized planes in order to be able to normalize body part trajectories. Normalized planes must all have the same height, have a width propotional to the elapsed time, and be connected in order to avoid discontinuities in normalized body part trajectories. The initial height H^p and the width W^p of each normalized plane $\hat{\Pi}^p$ are computed as follows:

$$H^p = \|\hat{\boldsymbol{\pi}}_{\text{BT}}^p - \hat{\boldsymbol{\pi}}_{\text{BB}}^p\| \ , \ \ W^p = \|\hat{\boldsymbol{\pi}}_{\text{EB}}^p - \hat{\boldsymbol{\pi}}_{\text{BB}}^p\|. \qquad (17)$$

A ratio R^p is then computed for each normalized plane. It indicates the relationship between the plane's width and its number of frames. This ratio is used to scale the width:

$$R^p = \frac{W^p}{(t_r^{p+1} - 1) - t_r^p} = \frac{W^p}{\Delta t^p} \qquad (18)$$

where t_r^p represents the time associated with the junction point $\boldsymbol{j}_r^p$. The height and width at which all planes are scaled is not important. Scaling uses a fixed beginning bottom corner $\hat{\boldsymbol{\pi}}_{\text{BB}}^p$. It was chosen to scale the height to H_{median} and use the maximum ratio of all normalized planes R_{max} :

$$H_{\text{median}} \ = \ \underset{p=1,\ldots,N_p}{\text{median}} H^p, \qquad (19)$$

$$R_{\text{max}} \ = \ \underset{p=1,\ldots,N_p}{\max} R^p. \qquad (20)$$

The obtained width of a normalized plane is:

$$W_{\text{scaled}}^p = R_{\text{max}} \Delta t^p. \qquad (21)$$

Setting the width using the same R_{max} ratio for all planes implies that the walking velocity is constant across all planes. The positions of the corners are finally computed using the new height and width ($\hat{\boldsymbol{\pi}}_{\text{BB}}^p$ remains at the same position):

$$\hat{\boldsymbol{\pi}}_{\text{EB}}^p \ = \ \left[\hat{\boldsymbol{\pi}}_{\text{BB},\,x}^p + uW_{\text{scaled}}^p \,, \ \hat{\boldsymbol{\pi}}_{\text{BB},\,y}^p \right]^{\text{T}} \qquad (22)$$

$$\hat{\boldsymbol{\pi}}_{\text{BT}}^p \ = \ \left[\hat{\boldsymbol{\pi}}_{\text{BB},\,x}^p \,, \ \hat{\boldsymbol{\pi}}_{\text{BB},\,y}^p + H_{\text{median}} \right]^{\text{T}} \qquad (23)$$

$$\hat{\boldsymbol{\pi}}_{\text{ET}}^p \ = \ \left[\hat{\boldsymbol{\pi}}_{\text{BB},\,x}^p + uW_{\text{scaled}}^p \,, \ \hat{\boldsymbol{\pi}}_{\text{BB},\,y}^p + H_{\text{median}} \right]^{\text{T}} \qquad (24)$$

The last plane normalization step is the shifting of each plane such that the beginning corners of plane $\hat{\Pi}^{p+1}$ are at the same position as the ending corners of plane $\hat{\Pi}^p$. If $\boldsymbol{z}^p = \hat{\boldsymbol{\pi}}_{\text{BB}}^p - \overline{\boldsymbol{\pi}}_{\text{EB}}^{p-1}$ represents the amount of shift between the beginning bottom corner $\hat{\boldsymbol{\pi}}_{\text{BB}}^p$ and the new ending corner $\overline{\boldsymbol{\pi}}_{\text{EB}}^{p-1}$, then the new corners of the normalized plane $\overline{\overline{\Pi}}^p$ are:

$$\overline{\boldsymbol{\pi}}_{\mathcal{C}}^p = \hat{\boldsymbol{\pi}}_{\mathcal{C}}^p - \boldsymbol{z}^p \qquad (25)$$

for $\mathcal{C} : \{\text{BB}, \text{EB}, \text{BT}, \text{ET}\}$ and $p = 2, \ldots, N_p$. For the special case $p = 1$, $\overline{\boldsymbol{\pi}}_{\mathcal{C}}^1 = \hat{\boldsymbol{\pi}}_{\mathcal{C}}^1$.

2.6. Normalizing Body Part Trajectories

Once the normalized planes are obtained, it is possible to compute a homography matrix $\boldsymbol{E}^p$ by constructing an 8-equation linear system using the correspondences between

Subject no.	Head	Foot 1	Foot 2
1, original, 90 & 60 deg.	470.8	459.3	736.4
1, normalized, 90 & 60 deg.	5.0	12.2	7.3
2, original, 90 & 60 deg.	384.7	469.9	697.7
2, normalized, 90 & 60 deg.	7.4	5.6	10.8
3, original 90 & 45 deg.	1116.2	918.0	1745.8
3, normalized 90 & 45 deg.	7.6	27.9	20.6

Table 1. MSE alignment values

the corners of the original and normalized walking planes Π^p and $\overline{\overline{\Pi}}^p$. The linear equation systems are solved using the Gauss-Jordan method.

The homography matrix associated with each normalized plane $\overline{\overline{\Pi}}^p$ may be applied to body part trajectories in order to retrieve their normalized trajectories. If the position at time t of a body part $\mathcal{R}$ is defined as $\boldsymbol{b}^{\mathcal{R}}(t)$, then the normalized trajectory $\overline{\boldsymbol{b}}^{\mathcal{R}}(t)$ is computed as

$$\left[\alpha \overline{b}_x^{\mathcal{R}}(t), \ \alpha \overline{b}_y^{\mathcal{R}}(t), \ \alpha \right]^{\text{T}} = \boldsymbol{E}^p \left[b_x^{\mathcal{R}}(t), \ b_y^{\mathcal{R}}(t), \ 1 \right]^{\text{T}} \quad (26)$$

for $t_r^p \leq t < t_r^{p+1}$ and $p = 1, \ldots, N_p$ (α is a scale factor). The homography matrix $\boldsymbol{E}^p$ of the normalized plane $\overline{\overline{\Pi}}^p$ is only used to normalize trajectory positions for $t_r^p \leq t < t_r^{p+1}$, since that time interval corresponds to the original walking plane Π^p.

3. Results

View normalization was applied to head and feet trajectories obtained from a tracking algorithm [7]. Video sequences were acquired in an indoor environment. A first set of twelve video sequences was obtained where three different subjects walk along a single straight line. Each subject was observed by four time-synchronized cameras (30 fps, 640×480 resolution) for validation purposes. Each sequence was processed independently. The optical axis of the four cameras intersected the walking trajectories at 90°, 75°, 60°, and 45°, respectively. A second set of three video sequences (15 fps) was obtained where a single subject makes both smooth and sudden changes in his walking trajectory.

The normalized body part trajectories of the time-synchronized sequences should match the trajectories from the fronto-parallel view. For each of the three subjects, the fronto-parallel (90°) sequence along with a sequence from another view were normalized. For the fronto-parallel view, the normalization process made only small changes to the trajectories. Normalized trajectories from the two views of the same subject were scaled vertically and horizontally so that they would have the same normalized planes height and R ratio. In this case, it is possible to do the vertical scaling
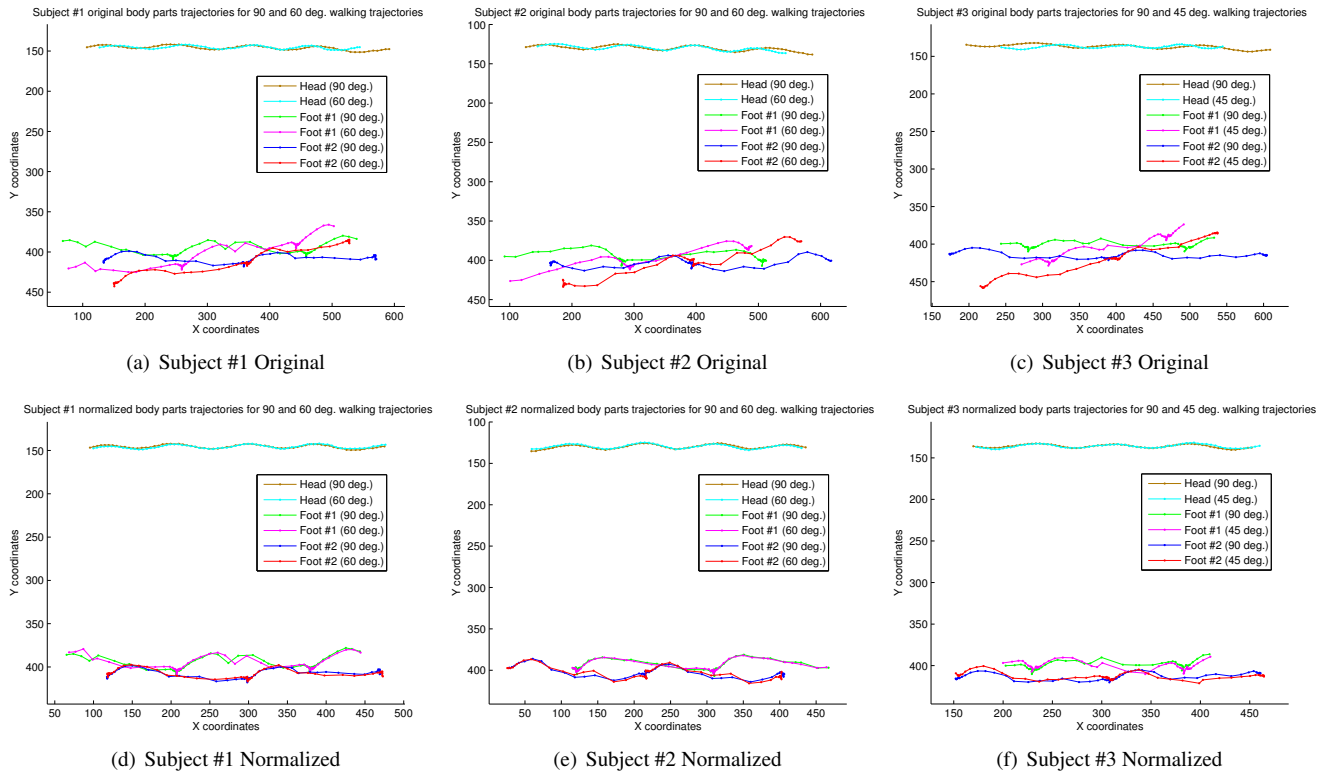
Figure 7. Body part trajectories comparison

by assuming the same height since it is known that the trajectories came from the same person.

In Fig. 7, time-synchronized original and normalized trajectories are spatially aligned and compared. In Fig. 7(a), original trajectories at 90° (fronto-parallel view) and 60° for subject no. 1 show signifiant differences. In Fig. 7(d), normalized trajectories are well aligned. Results for other subjects and angle pairs are shown in Fig. 7(b)-7(e), and Fig. 7(c)-7(f).

A quantitative validation measure was computed during the alignment process. Aligment was computed independently for each trajectory, before and after normalization. Alignment of two trajectories consists in computing the 2D translation minimizing the means square error (MSE) between positions at corresponding frames. As this is a quite involved computation, an approximation was computed instead by selecting the best discrete aligment of positions at corresponding frames. Table 1 shows the MSE of the aligned trajectories in Fig. 7. MSE alignment values for normalized trajectories are much lower than values for original trajectories. Values obtained for normalized feet trajectories of subject no. 3 are higher than those for subjects no. 1 and 2. This is partly due to feet occlusion lasting longer in that view (45°) compared to other views, which leads to least precise position of the occluded foot in the tracking

algorithm.

Fig. 8 shows the result of view normalization on the three sequences with changes in walking direction. The normalized trajectories look like trajectories obtained from a fronto-parallel viewpoint (side view), which is assumed optimal for gait modeling and recognition [10]. In Fig. 8(f), normalized feet and head trajectories are slightly deformed around the junction between the third and fourth planes. This is due to both the slowdown and the sudden change in walking direction. As a result, the feet locally violate the basic planar motion assumption.

4. Conclusion

The proposed approach looks promising since it has direct applications to gait-based modeling and identification, which performs significantly better from a side view. As proved by experimental validation, the normalized trajectories of head and feet from different views are well aligned with real side-view trajectories. Ongoing work focuses on testing the proposed approach on trajectories of additional body parts (hands, knees etc.) involved in human walk.

(a) Sequence #1 Original   (b) Sequence #2 Original   (c) Sequence #3 Original

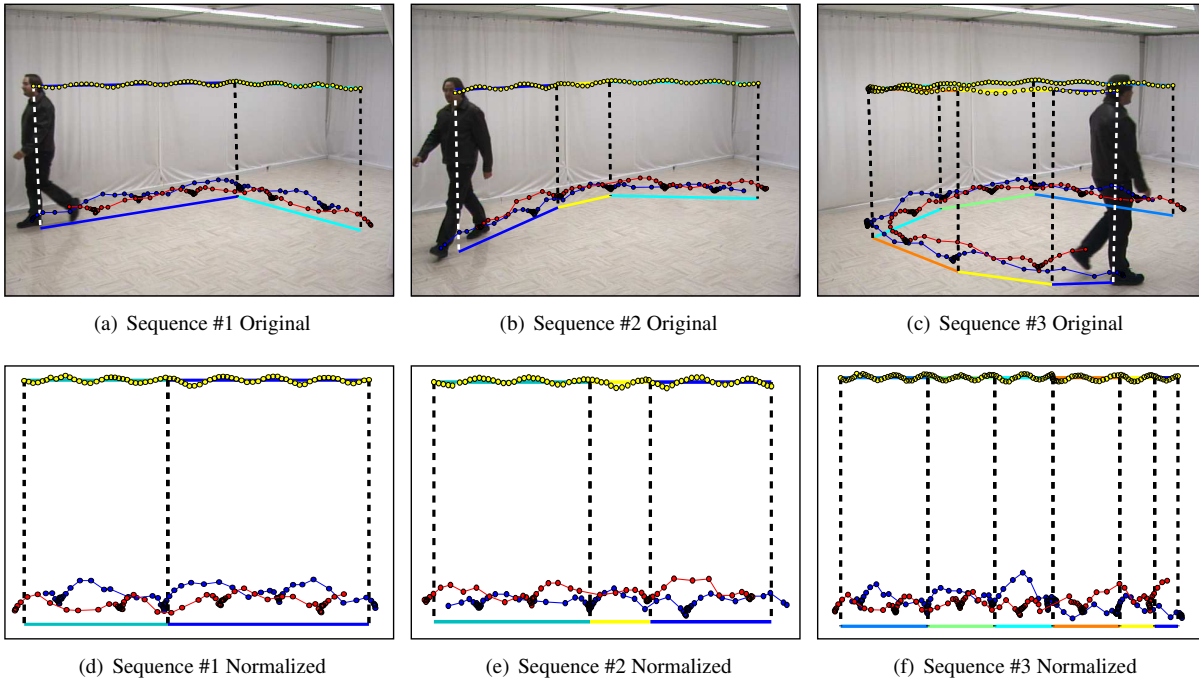(d) Sequence #1 Normalized   (e) Sequence #2 Normalized   (f) Sequence #3 Normalized

Figure 8. Trajectories normalization

References

[1] A. Branzan-Albu, D. Laurendeau, S. Comtois, D. Ouellet, P. Hébert, A. Zaccarin, M. Parizeau, R. Bergevin, X. Maldague, R. Drouin, S. Drouin, N. Martel-Brisson, F. Jean, H. Torresan, L. Gagnon, and F. Laliberté. Monnet: Monitoring pedestrians with a network of lossely-coupled cameras. In *Proc. of ICPR*, Hong Kong, China, August 20-24 2006.

[2] R. T. Collins, A. J. Lipton, and T. Kanade. Introduction to the Special Section on Video Surveillance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):745–746, 2000.

[3] R. Cutler and L. S. Davis. Robust Real-time Periodic Motion Detection, Analysis, and Applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):781–796, 2000.

[4] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. New York: John Wiley & Sons, 1973.

[5] J. Han and B. Bhanu. Performance Prediction for Individual Recognition by Gait. *Pattern Recognition Letters*, (26):615–624, 2005.

[6] I. Haritaoglu, D. Harwood, and L. S. Davis. W4 : Real-Time Surveillance of People and Their Activities. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):809 – 830, 2000.

[7] F. Jean, R. Bergevin, and A. B. Albu. Body tracking in human walk from monocular video sequences. In *Proc. of IEEE Conf. on Computer and Robot Vision*, pages 144–151, Victoria, BC, May 9-11 2005.

[8] A. Y. Johnson and A. F. Bobick. A Multi-view Method for Gait Recognition Using Static Body Parameters. In *Proc. of Int. Conf. on Audio- and Video-Based Biometric Person Authentication*, pages 301–311, Halmstad, Sweden, 2001.

[9] A. Kale, A. K. R. Chowdhury, and R. Chellappa. Towards a View Invariant Gait Recognition Algorithm. In *Proc. of IEEE Conf. on Advanced Video and Signal Based Surveillance*, pages 143–150, Los Alamitos, CA, USA, 2003.

[10] A. Kale, A. Sundaresan, A. Roy-Chowdhury, and R. Chellappa. *Gait-Based Human Identification From A Monocular Video Sequence, Handbook on Pattern Recognition and Computer Vision*. World Scientific Publishing Company Pvt. Ltd., 3rd edition, 2005.

[11] M. P. Murray. Gait as a Total Pattern of Movement. *American Journal of Physical Medicine*, 46:290–332, 1967.

[12] N. M. Spencer and J. N. Carter. Viewpoint Invariance in Automatic Gait Recognition. In *BMVA Symp. Advancing Biometric Techniques*, pages 1–6, 2002.

[13] R. Urtasun and P. Fua. 3D Tracking for Gait Characterization and Recognition. In *Proc. of IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pages 17–22, Seoul, Korea, 2004.

[14] J. H. Yoo and M. S. Nixon. Markerless Human Gait Analysis via Image Sequences. In *Proc. of Int. Society of Biomechanics XIXth Congress*, 2003.

[15] S. Yu, L. Wang, W. Hu, and T. Tan. Gait Analysis for Human Identification in Frequency Domain. In *Proc. of Int. Conf. on Image and Graphics*, pages 282–285, Hong Kong, China, 2004.

[16] R. Zhang, C. Vogler, and D. Metaxas. Human Gait Recognition. In *Proc. of Conf. on CVPR Workshops*, volume 1, pages 18–25, Washington, DC, USA, 2004.

IEEE
COMPUTER
SOCIETY