# Detection of Gait Characteristics for Scene Registration in Video Surveillance System

László Havasi, *Student Member, IEEE*, Zoltán Szlávik, *Member, IEEE*, and Tamás Szirányi, *Senior Member, IEEE*

***Abstract*—This paper presents a robust walk-detection algorithm, based on our symmetry approach which can be used to extract gait characteristics from video-image sequences. To obtain a useful descriptor of a walking person, we temporally track the symmetries of a person's legs. Our method is suitable for use in indoor or outdoor surveillance scenes. Determining the leading leg of the walking subject is important, and the presented method can identify this from two successive walk steps (one walk cycle). We tested the accuracy of the presented walk-detection method in a possible application: Image registration methods are presented which are applicable to multicamera systems viewing human subjects in motion.**

***Index Terms*—Image registration, motion analysis, object detection.**

## I. Introduction

THE process of extracting and tracking of human figures in image sequences is a key issue for video surveillance and video-indexing applications. The need for automated person identification systems strongly motivates this interest. The process can be broken down into the following steps: detection [1], tracking, classification [2], and identification [3], [4] of human movement or gait. There are several approaches for each of these subproblems. A useful and popular approach is based on silhouette analysis [5] with spatiotemporal representation, where the goal is to achieve an invariant representation of the detected object. In [4], symmetries of the silhouette are utilized as a gait parameter for person identification. Other methods focus on the legs [6] and periodicity of human movements [1], [7]. This short summary demonstrates the wide variety of approaches and features that can be used for object detection/identification.

The main aim of our paper is to present a method for the detection of human walking in videos and for the extraction of gait features. Our feature extraction method, which is based on method proposed in [8], utilizes extended third-level symmetries of the edge map to detect and track structural changes of moving objects in video sequences. In this paper we present a novel method that can extract gait information such as the walk period (begin and end of walk pattern) and identification of the "leading" leg during a walk cycle [9]. This method is based on detecting the moving leg pairs and developing the symmetry based approach in [8] for more robust cases: independence of noise and the varying frame rate [10]. We apply an invariant and effective data representation in the Eigenwalk space, based on spline interpolation and a dimension-reduction technique. Here, we present a more established pattern classification method (in contrast to [8]) based on the continuous interpolation of the symmetry patterns. A more robust classification is carried out via support vector machine (SVM) with Gaussian kernel function.

An interesting application of the extracted gait features is their use for the analysis of the observed scene [11]. Registration between partially overlapping wide baseline views of the same scene is an important task in a number of applications involving multicamera systems [12], [13]. For testing the efficiency and the robustness of the afore-mentioned feature extraction method, it has been applied for camera registration (the idea was introduced in [18]). The features we used (concurrent walk steps and leading-leg identity) seem to be beneficial to provide data (matching points) for the estimation of transformation between two different camera views of the same scene. Registration between nonoverlapping views is still a challenge and it is only solved in special cases [14]–[16]. It will be shown experimentally that, in the case of motion along a line, our feature detection method provide usable information for registration of nonoverlapping views. The cameras in this test are pointed in opposite directions and they are mounted on different sides of the same wall.

The assumptions we use in the paper are:

- the camera is in arbitrary static position;
- the image motion can be from more than one person;
- the image capture rate is at least 10 fps, but its stability is not supposed;
- the height of each "target" person is at least 100 pixels;
- leg opening is visible in most cases.

## II. Symmetry Pattern Extraction

The basis of our algorithms is the ability to detect human movements. We present a simple motion pattern generation and extraction method.

This task is a binary classification problem: the periodicity of human walking, together with the characteristic human shape of the target, provides key features which enable us to distinguish pedestrians' motion patterns from the motion patterns of other objects. Our approach uses the motion information contained in the video sequences so that the extracted motion patterns consist

Manuscript received January 19, 2006; revised August 22, 2006. This work was supported by the NoE MUSCLE project of the EU. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ercan E. Kuruoglu.

L. Havasi is with the Peter Pázmány Catholic University, Budapest, Hungary (e-mail: havasi@digitus.itk.ppke.hu).

T. Szirányi and Z. Szlávik are with the Analogical and Neural Computing Laboratory, Hungarian Academy of Sciences, Budapest, Hungary (e-mail: sziranyi@sztaki.hu; szlavik@sztaki.hu).

Digital Object Identifier 10.1109/TIP.2006.888339

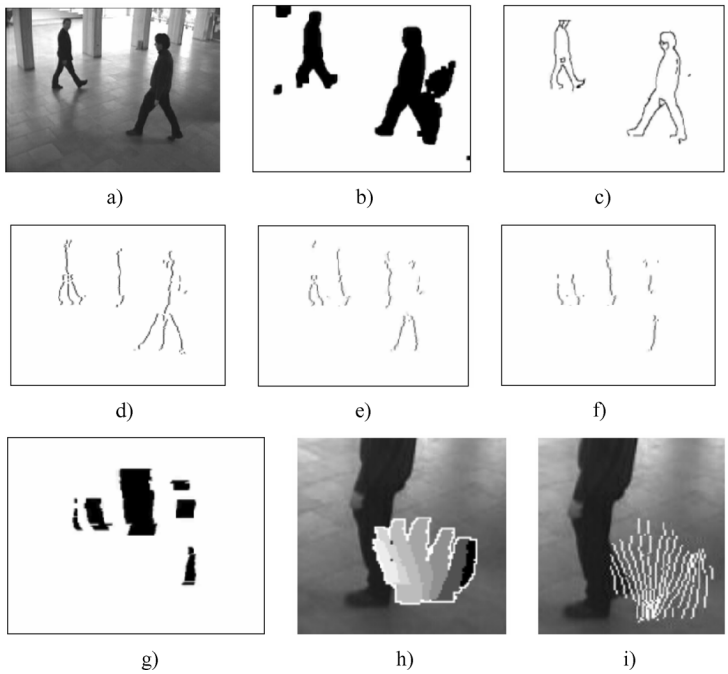



Fig. 1. Overview of feature extraction steps [8]. (a) Image from input sequence. (b) Result of change detection. (c) Filtered Canny edge map. (d) First-level symmetries. (e) Second-level symmetries. (f) Third-level symmetries (L3S). (g) Reconstructed masks from symmetries. (h) Tracking, showing coherent masks in the sequence (of seven representative frames). (i) Symmetry pattern (of the total 25 frames).

of information about the spatiotemporal changes of a moving object.

The main steps of the algorithm are as follows.

- *Background subtraction, change detection:* An elementary method to reduce the computation cost of methods using motion information derived from static-position cameras is background subtraction [11]; that is, remove all but what are important artifacts [see Fig. 1(b)].
- *Edge map detection and symmetry computation (first level):* Our symmetry detection method [8] is based on the use of morphological operators to simulate spreading waves from the edges. In that pedestrian detection approach, only horizontal morphological operators are used to extract the symmetries.
- *Extension of symmetry computation up to three levels (L3S):* As illustrated in Fig. 1(e) and (f), the symmetry concept can be extended by iterative operations. Higher order symmetries are used to describe local structure reflecting the overall complexity of an object. L3S is a representative feature of objects having two coherent parts with two parallel edges.
- *Temporal tracking using reconstructed masks:* In general, the image may contain a number of symmetry samples which have arisen from errors in change detection or from the complexity of the background. We have implemented an effective method for tracking the extracted symmetry fragments [8]. The first L3S appears when the legs are opening and the last is detected just before the legs are closed; so, a symmetry pattern of a walking person's step corresponds to the movement of the legs from opening to closing.

Samples of the image processing steps are shown in Fig. 1, which illustrates the results of the algorithmic procedures up to the stage of symmetry pattern extraction from the reconstructed masks.

## III. Detection of Walk Patterns and Gait Feature Extraction

The extended symmetry feature gives a specific pattern when it is tracked through the frames of 1–2 walking steps. We show that we can definitely differentiate between the symmetry pattern of walking legs and that of other parts, e.g., arms and head.

### A. Representation and Resampling

The extracted symmetry patterns are represented with the upper and lower end points (two each) of the L3S in each frame. Thus, there are four 3-D (space and time) coordinates, which correspond approximately to the "end points" of the two legs. Temporally, these patterns depend both on the frame rate and the walking speed, so a pattern usually contains data from 5–30 frames [see Fig. 2(b)]. All the symmetries composed of four or fewer frames are filtered out and not classified because they are usually produced by noisy backgrounds. Before any further analysis, the data is normalized with respect to time for presenting an invariant description of the motion; we perform this task with Bezier spline interpolation [19].

This technique has the advantage that it performs two tasks: 1) data is resampled in a defined time interval with a fixed-point count; 2) noise iltering is performed on the trajectories, which results in a smoother symmetry pattern. The noise cleaning is critical because, in real scenes, these patterns are often damaged [see Fig. 2(b)]. The Bezier spline is a good choice because the

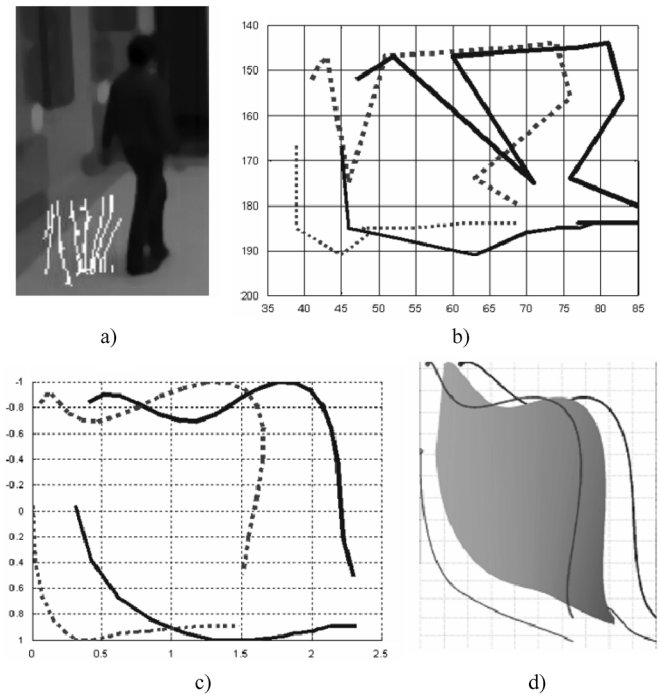



Fig. 2. (a) Original symmetry pattern and the trajectories of nine frames. The four curves in (b) (trajectories) are the upper and lower—both left and right—end points of the symmetry sample expanded with its radius. Interpolated trajectories of 100 points by using Bezier splines in (c) and the numerically integrated surface of the pattern in (d). The surface is formed from the interpolated upper and lower end points of symmetries which represents the height of the visible area of leg opening.

effect of base points is global; so, the presence of some damaged points, coming from erroneous symmetry extraction and unstable video frame rates, does not cause significant change in the whole trajectory.

This time-extended data representation permits the integrated analysis of data obtained from several cameras where the frame rates are different and unstable [10] (e.g., network cameras); the extracted features must be resampled with a continuous time division. The result of Bezier spline interpolation of data can be seen in Fig. 2(c). The current implementation generates 100 interpolated points of both coordinates of every end point. These points are termed with the following vectors (each has dimension of 100) $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$. Where $(x_1, y_1)$ is the upper left, $(x_2, y_2)$ is the lower left, $(x_3, y_3)$ is the upper right, and $(x_4, y_4)$ is the lower right endpoint positions of the symmetry mask, respectively.

### *B. Dimension Reduction*

The interpolated 3-D (XYT) points are rearranged into a row vector with a dimension of 800 (because it is the concatenation of the eight vectors of coordinates) $\tilde{x} = [x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4]$.

The linearity of the time coordinate makes a smooth time division (time is linearly related to the successive samples thanks to the resampling). Consequently, we can omit this coordinate; it has no discriminative information content. After we center the patterns for both the $x$ and $y$, both coordinates are normalized using the same constant chosen such that $\max(\mathrm{y}) = 1$ and

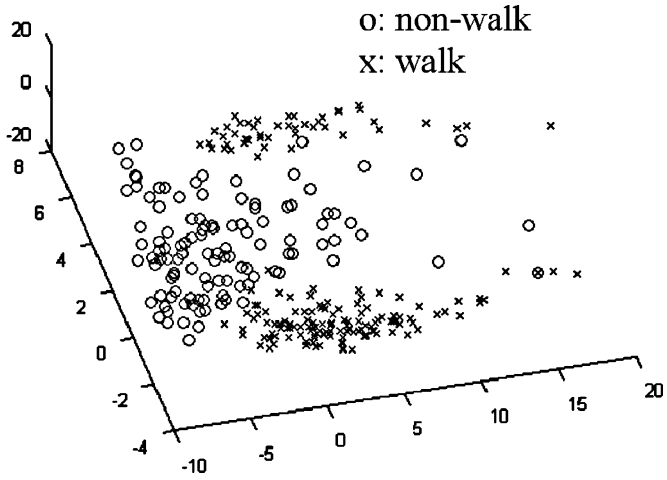



Fig. 3. "Walk" and "nonwalk" patterns in the eigenspace.

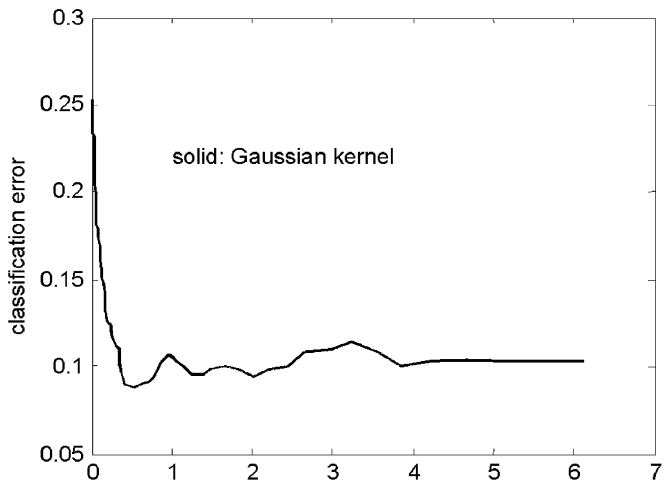



Fig. 4. Relation between the kernel parameter and the classification error rate for the Gaussian kernel.

$\min(\mathrm{y}) = -1$; we do this because we have found that the $y$ size of the patterns varies less than does the $x$ size. We do not normalize with individual coefficients for x and y, since in that case the information content of the ratio of $x$ and $y$ values would be lost.

A well-known technique for dimension reduction is the PCA method [20]. We considered the space spanned by the three most significant eigenvectors of the covariance matrix of the interpolated data set that account for 93% of the variation in the input space: We call this the Eigenwalk space. The associated eigenvectors form the eigenspace transformation matrix.

This drastically reduced number of dimensions greatly assists in increasing the classification speed, which is an important factor in real-time applications. Fig. 3 demonstrates the results using the test set of labeled "walk" and "nonwalk" symmetry patterns.

### *C. Classification of Symmetry Patterns*

Level three symmetries can also appear in other parts of the image, not only between the legs, and the tracking method also collects all of these related symmetries. Walk patterns lie on

a) b) c) d) e)

Fig. 5. (a) Image showing the location of the derived symmetry pattern (marked with white border; "x" marks a feature point). (b), (c) Illustrations of our definition of "leading leg"; the "standing" or leading leg is the right leg in (b), and the left leg in (c) (legs highlighted manually). (d), (e) The detected patterns for the same steps as shown in (b) and (c); the 2-D direction is bottom left to upper right (case 2 in Table I).

a nonlinearly shaped manifold in the eigenspace (see Fig. 3). The classification process is carried out via nonlinear method, namely the SVM [21] with a radial-basis kernel function $k(\bar{x}, \bar{x}_i) = \exp(-\|\bar{x} - \bar{x}_i\|^2/2\sigma^2)$.

The training data set, assembled from indoor video sequences, contained 750 "walk" and 14200 "nonwalk" patterns in the eigenspace. The parameter $(\sigma)$ was determined in the interval $[0.1 \ldots 6.0]$; from this, an optimal value is given by 0.4 where the valid classification (see Fig. 4) rate is 93.8% on the training set with 217 support vectors. Our main goal was to reliably detect human movements, but, at the same time, with a false-positive (5.2%) detection rate as small as possible (the false-negative rate was 1.0%).

### D. Leading Leg Identification

According to our terminology, the leading leg is the "standing" leg, which, at that instant, carries the person's weight [see Fig. 5(b) and (c)]. In this section, we present a method to determine, from one detected walk cycle (two consecutive steps), whether the leading leg is the right or the left leg, by estimating 2-D direction of walk and the "ratio" of consecutive walk patterns. The 2-D motion vector on the image plane, and the walker's gait period, can be extracted directly from the detected patterns: We estimate the motion vector by fitting a regression line to the last half trajectory of the lower two points of the pattern.

The nonrigid human body during a walking cycle has a useful property, which assists us in recognizing the leading leg. Depending on the 3-D walk direction, and on which is currently the leading leg, one leg or the other practically obscures the visible area between the legs.

During one cycle, the left leg and right leg in turn are in the leading position. The afore-mentioned method can detect one step. To connect two successive steps as one walk cycle, we calculate the 2-D displacement vector of a detected step and then search for another step (walk pattern) in the estimated 2-D position and at a time point after a forecasted walk period.

During a walk cycle [two consecutive steps; see Fig. 5(d) and (e)], the ratio of the visible leg-opening areas, together with the 2-D direction on the image plane, can be used to identify which is the leading leg. The visible leg opening area is approximated by the surface defined by symmetries between the legs from consecutive frames [see Fig. 2(d)]. To measure the area between the legs, we used a numerical integral (1) of the surface defined by the interpolated patterns described in Section III. The area of surface was approximated by dividing it into triangles and summing areas of triangles. The areas are computed by using the well-known vector product

TABLE I
SURFACE DEPENDENCIES ON 2-D WALK DIRECTION AND LEADING LEG

| Case | 2D Dir | Leading Leg | Ratio |
|---|---|---|---|
| 1 | ↗ | Right | >1 |
| 2 | ↗ | Left | <1 |
| 3 | → | Right | ≈1 |
| 4 | → | Left | ≈1 |
| 5 | ↘ | Right | <<1 |
| 6 | ↘ | Left | >>1 |
| 7 | ↖ | Right | <1 |
| 8 | ↖ | Left | >1 |
| 9 | ← | Right | ≈1 |
| 10 | ← | Left | ≈1 |
| 11 | ↙ | Right | >>1 |
| 12 | ↙ | Left | <<1 |

$$\text{area} \approx \frac{1}{2}\sum_{i=1}^{n-1}\begin{pmatrix} \|(\vec{r}_U(i)-\vec{r}_L(i))\times(\vec{r}_U(i+1)-\vec{r}_L(i))\| + \\ \|(\vec{r}_U(i+1)-\vec{r}_L(i))\times(\vec{r}_L(i+1)-\vec{r}_L(i))\| \end{pmatrix} \tag{1}$$

where $n = 100$, the number of interpolated points, and $r_U$ and $r_L$ are the upper and lower midpoints of the interpolated patterns

$$\vec{r}_U(i) = \left[\frac{x_1(i)+x_3(i)}{2}; \frac{y_1(i)+y_3(i)}{2}\right]$$
$$\vec{r}_L(i) = \left[\frac{x_2(i)+x_4(i)}{2}; \frac{y_2(i)+y_4(i)}{2}\right] \tag{2}$$

where the running index $i$ is along the trajectories of the symmetry pattern.

Table I summarizes the relationship between the leading leg and the ratio of surfaces from two successive patterns. A limitation of the described method is that it cannot identify the leading leg when the motion is parallel to the camera plane, since, in such cases, the areas are nearly equal (cases 3, 4, 9, and 10 in Table I).

Fig. 6. Detection of symmetry patterns in various outdoor videos.

Fig. 7. Detection of symmetry patterns in indoor video.

## IV. Application of Walk Detection for Camera Registration

An important application of walk detection is the estimation of relative camera positions from the detected walking. We show that this specific type of feature can be used well for generating matching points. If the observed motions are on the ground plane then the relation between images of the same scene can be modeled as a homography [22]. This view registration also provides us with a checking methodology for the accuracy of gait features on the single views.

### A. Two Overlapping Views

The problem can be summarized as follows: given a set of points $x_i$ in a view and a corresponding set of points $x_i'$ in another view, we need to compute the projective transformation (2-D homography) that takes each element $x_i$ to $x_i'$ (vectors are in homogenous form). The problem is to compute a $3 \times 3$ matrix, $H$ (point map), such that $x_i^{'} = Hx_i$. This computation can be accomplished in several ways; details can be found in [22]. To solve the problem, we need at least four point correspondences.

To detect such corresponding points, we use our walk detection and leading-leg identification methods. Both methods provide information, which is useful in matching points between the two views: detected walk patterns must be concurrent in both views, and, likewise, the leading leg must be the same. In both views, the central lower points of the detected walk patterns are the feature points [e.g., the one marked with a black "x" in Fig. 5(a)]. This fact results in some outliers in the detected points. For the estimation of the transformation $H$ that maps points of one camera scene onto the other, and for rejection of outliers from the set of candidate point pairs, we have implemented both the simple direct linear transformation (DLT) method, and its extension using the random sample consensus (RANSAC) algorithm.

### B. Nonoverlapping Views

In our surveillance system, there is a nonoverlapping camera configuration where the persons walk from the view of "entrance" camera to the view of "main hall" camera. The motion from one view to the other is rectilinear in this configuration and the following computation is utilizing this property.

The computation differs from the overlapping case because corresponding points could not be detected. An alternative way to determine the matrix H is the use of line correspondences instead of point correspondences [22]. This approach needs the assumption that the motion is along a straight line and these line fragments may be detected in both views. The line sets built from two successive walk steps (a walk cycle) which define two points on the ground plane; thus, the parameters of the line across these points can be calculated directly.

## V. Experimental Results

### A. Detection and Feature Extraction

The detailed implementation issues of symmetry extraction and tracking are described in [8]. Careful implementation of the method with the new filtering and detection step resulted in 10–15 ms processing speed for a symmetry pattern, on a state of the art desktop PC. The number of extracted symmetries affects the speed of filtering damaged points (interpolation). Hence, the processing speed depends on cameras frame rate.

During the test, we used the frame sequences as captured, recognizing walking patterns real time. We have tested our methods using test inputs from both indoor and outdoor videos, where the following factors were varied: camera viewpoint, number of "targets," and image-capture rate. These videos contained 420 steps and 150 walk cycles in the indoor scenes, and 350 steps and 110 cycles in outdoor environments. Figs. 6 and 7 show sample results of symmetry pattern extraction in various videos.

As it can be seen from the results, the algorithm performs well in the case of very different lighting conditions, image quality,

Fig. 8. Detection of symmetry pattern in the case of poor silhouette extraction [11], [17]. Upper row: The extracted foreground image mask in representative frames of a walk cycle. Lower row: Corresponding frames with the detected symmetry pattern in the last frame.

TABLE II
EXPERIMENTAL RESULTS ON DETECTION OF WALK PATTERN

| Method | Data set | Detection Rate | False-Positive | False-Negative |
|---|---|---|---|---|
| KFDA [8] (Gaussian kernel) | Training | 89.2% | 8.2% | 2.6% |
| SVM (Gaussian kernel) | Training | 93.8% | 5.2% | 1% |
| KFDA [8] | Indoor test | 75.7% | 15.3% | 9% |
| SVM | Indoor test | 89.5% | 8.9% | 1.6% |
| SVM | Outdoor test | 78.1% | 14.1% | 7.8% |

background, and video frame rate (10, 15, and 25 FPS). The ability of walk extraction in the case of an inaccurate foreground image mask is demonstrated in Fig. 8, where the objects are not separated.

There are several obvious limitations of the tracking algorithm. The method could not detect "near-frontal" human movements (motion directly toward camera, or nearly so). Also, when the leading leg covers the rear leg, the symmetries do not appear [8]. We obtained a detection rate of 78.1% for outdoor and 89.5% for indoor videos, in cases where the leg motion (and the leg opening) was visible (detailed results in Table II).

The leading leg identification worked well, with 99% correct identification in cases where the walk cycle was detected correctly. This method fails in two special cases where the walker is approaching "close" (Table I, cases 5–6 and 11–12), *viz.*: 1) top left to bottom right, and right leg leading (cases 5 and 11) or 2) top right to bottom left, and left leg leading (cases 6 and 12). This walk-detection procedure has been implemented in the camera system of university campus.

### B. Registration of Camera Views

We evaluated the registration algorithm by using surveillance cameras placed in a public area located in the university building. The angle between the view axes of the two overlapping cameras employed was nearly 90° (hence, to detect corresponding points using standard optical methods would be difficult). In the nonoverlapping layout, the two cameras are placed oppositely to each other.

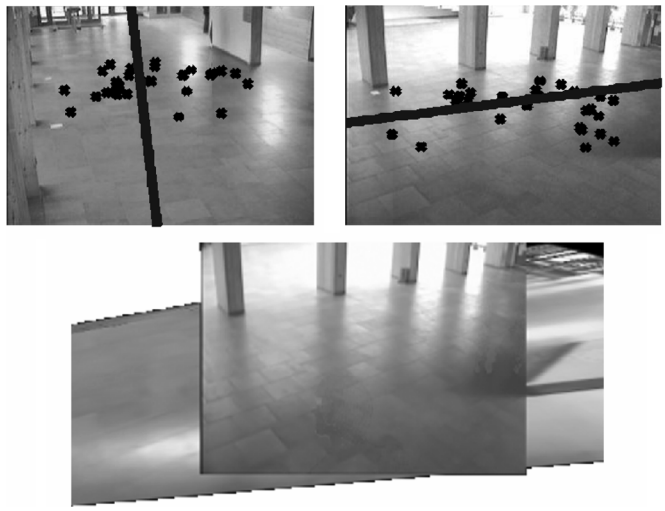


Fig. 9. Transformation from the first camera view (left in the upper row) to the second (right in the upper row): Detected corresponding points and a synthetic line trajectory are demonstrated in the upper images and alignment of views in the lower image.

TABLE III
EXPERIMENTAL RESULTS ON DATA FROM "ENTRANCE" CAMERAS (RANSAC DISTANCE THRESHOLD IS $T = 0.01$)

| Case | Input | Points | Correct points | Detected outliers | Avg. error in pixel |
|---|---|---|---|---|---|
| METHOD: DLT | | | | | |
| 1 | S2 | 8 | 8 | - | 6.4 |
| 2 | S1+S2 | 54 | 39 | - | 250.2 |
| METHOD: RANSAC+DLT | | | | | |
| 3 | S2 | 8 | 8 | 4 | 12.5 |
| 4 | S1+S2 | 54 | 39 | 25 | 7.8 |
| 5 | S1 | 46 | 31 | 28 | 6.2 |

*1) Overlapping Views:* In our series of tests, the successfully detected and classified walk patterns were 241 for the first camera, and 220 for the second camera [see Fig. 9(a) and (b)]. In our system, the cameras are approximately synchronized, but

Fig. 10. Images of "Main hall" and "Entrance" cameras with control lines on the ground (marked with two long paper tapes) for verification in upper row. Result of alignment of nonoverlapping views with the highlighted control lines in the lower image.

there is a small temporal drift between the walk patterns generated by each camera; hence, we define a permitted time window for events, which are classed as "concurrent." This time window for concurrent checking was five frames. After such checking, there remained 46 concurrent corresponding points (S1 dataset) and eight with the leading leg verified (S2 dataset). We found 15 invalid points in the S1 dataset. Table III summarizes the results of the simple DLT and the $\mathrm{RANSAC} + \mathrm{DLT}$ methods applied to several combinations of the S1 and S2 datasets (cases 1 to 5). We assessed the accuracy of the computed transformations (rightmost column) using manually selected control points.

Because of the near-orthogonal orientation of the two cameras used for the tests, the algorithm can rarely detect two successive walks for leading-leg identification, and, therefore, there are only a few points in the S2 dataset. Nevertheless, in case 1, all the points in S2 are correct points; and the simple DLT method can compute a good transformation. The DLT method fails when there are outliers (as for the $\mathrm{S1} + \mathrm{S2}$ dataset), and, in this case (case 2), the position error is extremely high. In cases 4 and 5, the RANSAC algorithm has managed to reject the outliers from S1, and the DLT method then computes an appropriate transformation. In case 3, $\mathrm{RANSAC} + \mathrm{DLT}$ fails to give good accuracy because there are only a few points in the S2 dataset.

In the indoor test sequences, the height of people was 115 pixels and their width was 40 pixels in average. The camera view registration is a good test bed for the evaluation of the proposed feature extraction method. The not so high average error of alignment (cc. 5% relative error to the height of object) with respect to the object size proves the usability of the localized features.

*2) Nonoverlapping Views:* In the last experiment, we aligned images of cameras with nonoverlapping field of view. The images of the "Main hall" and the "Entrance" cameras are shown in Fig. 10(a) and (b). It can be seen that the field of views of the cameras are not overlapped because of the wall between "Main hall" and "Entrance" areas, but they virtually are. The estimation of the homography is based on line correspondences and not on point correspondences as in previous experiments. Two successive walk steps were detected and a line was calculated through them.

The major assumption in this experiment is that people are moving along straight lines from "Main hall" to "Entrance," and vice versa. Every line from one view was paired with every line in the other view, and the RANSAC algorithm was used for the estimation of the model and rejection of outliers.

The results of aligned images are shown in Fig. 10(c). The results are based on 235 detected walks; 42 walk cycles (two walks form a line fragment as mentioned above) and nine inliers left after the RANSAC have been performed. The average deviation of the gradient of the real "paper tape" lines was $12.5^\circ$.

## VI. Conclusion

In this paper, we have introduced a robust pedestrian detection and gait feature extraction method. We are able to achieve a reliable detection rate using an invariant and effective data representation in the Eigenwalk space, based on spline interpolation and a dimension-reduction technique. A novel method for leading-leg identification has been presented; this is a possible gait characteristic for walker registration between multiple cameras capturing different views of the same target. An important goal was to use this feature for the purpose of multiple-camera registration.

A camera-registration method has been presented which uses walk parameters as features to identify corresponding points. The features we used (concurrent walk steps, leading leg identity, and 2-D motion vector) seem potentially to provide good data for the estimation of homography between two different camera views of the same scene and an occurring configuration of nonoverlapping views. The registration method has been verified on an actual indoor camera surveillance system and was able to provide real-time feature (walk) detection. This efficient

camera registration proves the accuracy of the localization of our gait features.

## REFERENCES


[1] R. Cutler and T. Ellis, "Robust real-time periodic motion detection, analysis and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 781–796, Aug. 2000.

[2] H. Murase and R. Sakai, "Moving object recognition in eigenspace representation: Gait analysis and lip reading," *Pattern Recogn. Lett.*, vol. 17, pp. 155–162, Feb. 1996.

[3] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette analysis-based gait recognition for human identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1505–1518, Dec. 2003.

[4] J. Hayfron-Acquah, M. Nixon, and J. Carter, "Automatic gait recognition by symmetry analysis," *Pattern Recogn. Lett.*, vol. 24, pp. 2175–2183, Sept. 2003.

[5] M. Soriano, A. Araullo, and C. Saloma, "Curve spreads: A biometric from front-view gait video," *Pattern Recogn. Lett.*, vol. 25, pp. 1595–1602, Oct. 2004.

[6] C. Curio, J. Edelbrunner, T. Kalinke, C. Tzomakas, and W. von Seelen, "Walking pedestrian recognition," *IEEE Trans. Int. Transport Syst.*, pp. 155–163, Sep. 2000.

[7] C. BenAbdelkader, R. Cutler, H. Nanda, and L. Davis, "Eigengait: Motion-based recognition of people using image self-similarity," in *Proc. Int. Conf. Audio- and Video-Based Biometric Person Authentication*, 2001, pp. 284–294.

[8] L. Havasi, Z. Szlávik, and T. Szirányi, "Higher order symmetry for non-linear classification of human walk detection," *Pattern Recogn. Lett.*, vol. 27, pp. 822–829, May 2006.

[9] L. Havasi, Z. Szlávik, and T. Szirányi, "EigenWalks: Walk detection and biometrics from symmetry patterns," in *Proc. IEEE Int. Conf. Image Processing*, 2005, pp. 289–292.

[10] S. Chaudhuri and D. R. Taur, "High-resolution slow-motion sequencing: How to generate a slow-motion sequence from a bit stream," *IEEE Signal Process. Mag.*, vol. 22, no. 3, pp. 16–24, Mar. 2005.

[11] Cs. Benedek, L. Havasi, Z. Szlávik, and T. Szirányi, "Motion-based flexible camera registration," in *Proc. IEEE AVSS*, 2005, pp. 439–444.

[12] Z. Szlávik, L. Havasi, and T. Szirányi, "Estimation of common groundplane based on co-motion statistics," in *Proc. ICIAR*, 2004, pp. 347–353.

[13] L. Lee, R. Romano, and G. Stein, "Monitoring activities from multiple video streams: Establishing a common coordinate frame," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 758–767, Aug. 2000.

[14] J. Kang, I. Cohen, and G. Medioni, "Persistent objects tracking across multiple non overlapping cameras," in *Proc. WACV/MOTION*, 2005, vol. 2, pp. 112–119.

[15] A. Rahimi, B. Dunagan, and T. Darrell, "Tracking people with a sparse network of bearing sensors," in *Proc. ECCV*, 2004, pp. 507–518.

[16] Y. Caspi and M. Irani, "Alignment of non-overlapping sequences," *Int. J. Comput. Vis.*, vol. 48, pp. 39–51, 2002.

[17] C. Stauffer, W. Eric, and L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[18] L. Havasi, Z. Szlávik, and T. Szirányi, "Use of human motion biometrics for multiple-view registration," in *Proc. IEEE Int. Conf. Image Processing*, 2005, pp. 35–44.

[19] C. de Boor, *A Practical Guide to Splines*. New York: Springer Verlag, 1978.

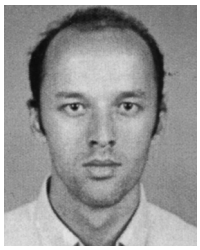
[20] P. Huang, C. Harris, and M. Nixon, "Human gait recognition in canonical space using temporal templates," in *Proc. IEE Vision Image and Signal Processing Conf.*, 1999, pp. 93–100.

[21] K.-L. Müller, S. Mika, G. Ratsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, May 2001.

[22] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.


**László Havasi** (S'01) received the M.Sc. degree in electrical engineering in 2002. He is currently pursuing the Ph.D. degree at the Péter Pázmány Catholic University, Budapest, Hungary.

He is currently a Research Fellow at the Computer and Automation Research Institute, Hungarian Academy of Sciences. His current research interests include robust motion detection and scene geometry exploration.



**Zoltán Szlávik** (M'03) received the B.Sc. and M.Sc. degrees in applied mathematics from the State University of Uzhgorod, Hungary, in 1999 and 2000, respectively.

He is currently a Research Fellow at the Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest. His current research interests include scene geometry exploration, camera systems, and visual event detection.

**Tamás Szirányi** (SM'91) received the Ph.D. degree in electronics and computer engineering in 1991 and the D.Sci. degree in image processing in 2001 from the Hungarian Academy of Sciences, Budapest.

He was appointed to a Full Professor position in 2001at Veszprém University, Hungary, and, in 2004, at the Péter Pázmány Catholic University, Budapest. He is currently a Scientific Advisor at the Computer and Automation Research Institute, Hungarian Academy of Sciences. His research activities include texture and motion segmentation, surveillance systems for panoramic and multiple camera systems, measuring and testing the image quality, digital film restoration, Markov random fields and stochastic optimization, image rendering, and coding.

Dr. Szirányi was the founder and past president (1997 to 2002) of the Hungarian Image Processing and Pattern Recognition Society. He is an Associate Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING. He was honored with the Master Professor award in 2001.