Robust Marker-Based Tracking for Measuring Crowd Dynamics

Wolfgang Mehner[1(✉)], Maik Boltes[2], Markus Mathias[1], and Bastian Leibe[1]

[1] Visual Computing Institute, Computer Vision Group,
RWTH Aachen University, Aachen, Germany
{mehner,mathias,leibe}@vision.rwth-aachen.de
[2] Forschungszentrum Jülich GmbH, Jülich, Germany
m.boltes@fz-juelich.de

Abstract. We present a system to conduct laboratory experiments with thousands of pedestrians. Each participant is equipped with an individual marker to enable us to perform precise tracking and identification. We propose a novel rotation invariant marker design which guarantees a minimal Hamming distance between all used codes. This increases the robustness of pedestrian identification. We present an algorithm to detect these markers, and to track them through a camera network. With our system we are able to capture the movement of the participants in great detail, resulting in precise trajectories for thousands of pedestrians. The acquired data is of great interest in the field of pedestrian dynamics. It can also potentially help to improve multi-target tracking approaches, by allowing better insights into the behaviour of crowds.

Keywords: Vision system application · Multi-target tracking · ID-markers

1 Introduction

The field of *pedestrian dynamics* deals with the analysis of collective behaviour of crowds of people, *e.g.*, investigating and modelling human behaviour during the evacuation of large public buildings. In order to assess basic properties of this behaviour, for example to determine the capacity of emergency exits, laboratory experiments are invaluable [16]. Only with such experiments one can do parameter studies, for example to investigate the influence of the width of an emergency exit on its throughput. A whole community [5,20] performs research on this subject and uses data from laboratory experiments for the estimation and validation of pedestrian models. They generate statistics of crowd motion, observe behavioural patterns, and collect empirical data for influencing norms and policy making.

When conducting such experiments with hundreds or thousands of participants not many technologies are available [3]. Motion capturing does not work for the targeted density and the number of participants. Other technologies, such

L. Nalpantidis et al. (Eds.): ICVS 2015, LNCS 9163, pp. 445–455, 2015.
DOI: 10.1007/978-3-319-20904-3_40

as RFID chips [17], do not offer sufficient precision, cannot track the required number of persons, or would be too cost intensive. Therefore, we propose a system based on video and use visual markers to detect and track participants. This system generates trajectories which can be analysed in great detail later on.

When conducting experiments, each participant wears a hat with a marker printed on it (Fig. 1). Using our proposed markers, each participant can be uniquely identified. We are then able to analyse motions of individuals across different runs of the experiments, and link it to other information, such as their gender and age.

In our application the main focus lies on precise measurements of the position and head orientation of all participants. To that end the frame rate during recording should at least be 16 fps, to enable the capturing of sudden motions. The tracking should not introduce any smoothing and not make assumptions about the underlying behaviour, as our application requires precise measurements. To achieve these goals, we create a special laboratory environment. This contrasts to classical tracking applications, where models have to be used to enable tracking and the reacquiring of targets after occlusions.

Our main contributions are: (1) We present a system to conduct measurements during laboratory experiments with thousands of pedestrians. (2) We propose a design for markers which guarantees a minimal Hamming distance between the used codes, even in the presence of rotated versions of the markers. (3) We present a pipeline to detect and precisely localize the resulting marker trajectories in 3D. We show the systems' applicability by using it on large scale experiments. It has been used to analyse about 200 experiments. Every experiment has been recorded by 12 to 24 cameras. Individual runs have been performed with up to 1000 participants. Trajectories produced by our system are already used to conduct research in pedestrian dynamics.

The remainder of the paper is organized as follows. After summarizing related work in Sect. 2, Sect. 3 explains the system set-up. Section 4 introduces the rotation invariant markers. In Sect. 5 we present the processing pipeline and evaluate it in Sect. 6.

2 Related Work

In the pedestrian dynamics community, various experiments have been evaluated using video processing methods. Boltes et al. used structured markers to track pedestrians during laboratory experiments [2], but without the possibility to identify individual persons. Colour is used by Daamen et al. to mark different classes of participants in their experiments [7]. Recently, ID-markers have been used by Stuart et al. [19] and Bukáček et al. [4]. However, their markers are larger than heads, making them unsuitable for high densities. Motion capturing is employed by Lemercier et al. [11,13] for experimenting with pedestrians walking in a line. In this case, motion capturing seem ideal. The view of the pedestrians' shoulder, which are also equipped with reflective markers, will not be obfuscated by pedestrians walking next to them.

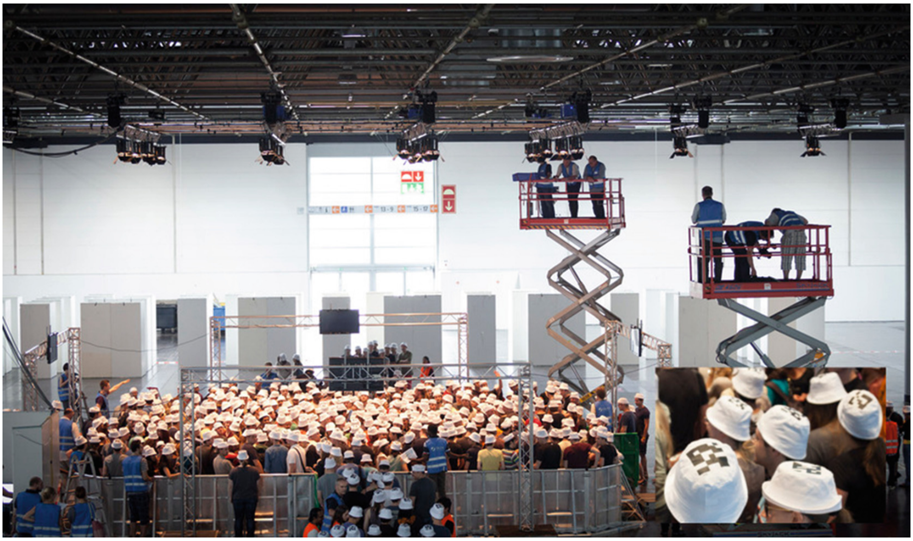


Fig. 1. An experiment in progress. The cameras are fixed to the same metal frame as the spotlights, 7.5 m above the floor. (image credit: Marc Strunz)

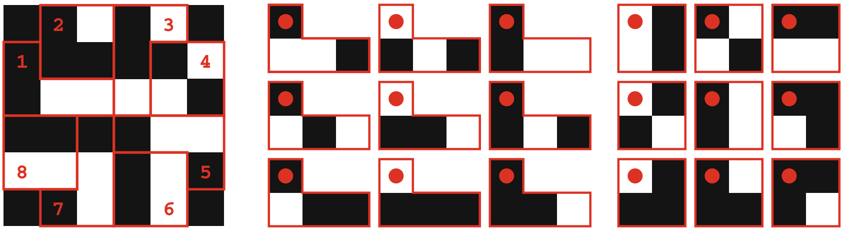
ID-markers have been the subject of research in the field of augmented reality. The software ARToolkitPlus [8] provides markers similar in grid size to those we use, but with an additional border. Our markers are better optimized for the given use case, i.e. they are better suited for smaller resolutions. Also the well-known QR Codes [1] require a minimum size of 11 by 11 marker bits, and therefore suffer the same problem of being difficult to read in low-resolution images.

Multi-target tracking and the analysis of human crowds are an active topic in computer vision research. For a comprehensive summary see [21]. Even though our tracking problem is more restricted, we target a different application with a special focus on high precision.

3 System Set-Up

The intended use of the system for conducting laboratory experiments yields a special set of requirements. The focus is on precise measurements. All participants should be tracked continuously, so occlusions have to be avoided. Furthermore, we rather want to report no measurement than a wrong one, since this is easier to correct in an interactive fashion. In conclusion, we focus on high precision in terms of the detection rate, precise positioning, and correct identification.

In the proposed set-up we use overhead views for the cameras, both to avoid occlusions and to facilitate the marker read out. We like to keep the viewing angles small, such that larger pedestrians can not occlude smaller ones, even at the borders of the field of view. Furthermore, small opening angles reduce the



(a) Positions of the letters on the marker.

(b) Encoding of the nine alphabet letters as L-shaped letters.

(c) Encoding as square letters.

Fig. 2. Marker Layout. (The red digits and the red dots mark the anchor points of each letter (Color figure online).)

image distortion. A large number of cameras increases the resolution to read out the markers for identification of the participants. We use a camera grid of six by four cameras (with a resolution of 1280 x 1024), mounted 7.5 m above the floor, which covers a little over 10 m by 10 m at ground level. The markers are of size 8.5 cm by 8.5 cm, resulting in an edge length of the marker bits of 1.4 cm.

Other than the grid cameras, one additional camera is placed in between two of the grid cameras to allow measuring the heights of the participants. Lastly, a fish-eye camera is placed in the middle of the grid, overlooking the entire experiment.

3.1 Camera Calibration

The internal calibration of each camera is computed beforehand [22]. The grid is calibrated using visual correspondences. A group of them is produced by laying out markers in the measurement area, which instantly gives identifiable feature points in the image spaces of the cameras. Additionally, out of plane correspondences are marked by hand. The calibration is then calculated via bundle adjustment [10] over the point correspondences.

4 Marker Design

The markers are designed to fulfil the constraints imposed by the experimental set-up (Sect. 3). Their maximum size is limited by several factors: the head size, the image resolution and the need for robust detection. A marker should not exceed the person's head, or have sharp edges, since that would be unsafe at the targeted density of up to six persons per square meter. Furthermore, the markers have to be readable from a distance of 6 m, given our camera resolution of 1280 x 1024.

The markers are encoded using an error-correcting code [6], which is able to detect errors caused by a wrong binarization of the markers. In coding theory, it is expected that unreliable channels distort the data, causing so-called transmission errors. Error-correcting codes encode messages (in our case the IDs of

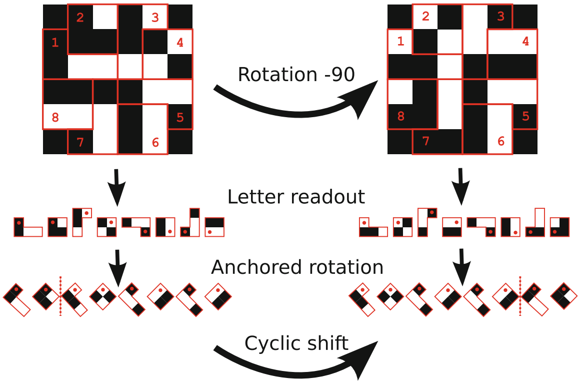


Fig. 3. Marker read out procedure and cyclic shifts. Each letter is indicated with a red rim, the number corresponds to the position of the letter in the codeword. After reading out each letters, the position of the number (now indicated with a red dot) corresponds to an anchor point. An anchored rotation aligns squares with a red dot. A rotation by 90 degrees corresponds to a cyclic shift of the codeword by two letters. (Color figure online).

the markers) with codewords. A minimal Hamming distance between each valid codeword guarantees the ability to detect or even correct transmission errors.

Rotation invariance can be resolved in various ways, *e.g.* by encoding rotation information into the marker design or by mapping different rotations of the marker to valid codewords.

We follow the second approach by ensuring that a rotation of the marker results in a valid codeword. To achieve this, we base our marker on a *cyclic* Reed-Solomon code [15]. Cyclic means that a cyclic shift of each valid codeword is still a valid codeword, e.g. if $(c_1, c_2, c_3, \ldots, c_8)$ is a valid codeword, so is $(c_2, c_3, \ldots, c_8, c_1)$.

Our marker layout is designed in a way that the readout of a rotated marker is just a cyclic shift of the readout of an unrotated marker. By design, all marker rotations are also valid codewords and fulfil the minimal Hamming distance.

Our Reed-Solomon code has messages of length $k = 5$ and codewords of length $n = 8$ over an alphabet with $q = 9$ letters. Figure 2a shows an example of a marker codeword consisting of 8 letters. Each letter is encoded with four bits (see Figs. 2b and 2c), the bits can be arranged L-shaped or squared. As can be seen, the letters may appear in different rotations. The positions of the numbers and red dots in Fig. 2 correspond to anchor points and define the rotation for each letter, *e.g.* in Fig. 2a letter 6 and letter 8 are the same.

By using only these 9 letters, we discard 7 $(= 2^4 - 9)$ combinations for potential letters. This helps us to impose additional structure on the marker, e.g. by not using letters which are completely white or black, and allowing only patterns with at least one black bit on the border of the marker.

With these parameters we are able to encode $q^k = 59049$ different messages. After identifying four different cyclic shifts of codewords, the number is reduced to $q^k/4 \approx 14762$. Some codewords are the same after a cyclic shift of 2 or 4 (or a rotation of 90° or 180°), which would not allow us to detect the full 360° rotation. After removing these markers, we can produce 14580 different markers.

The minimal distance of two codewords is $n - k + 1 = 4$, which allows us to detect two transmission errors, and correct one. Here, a transmission error means that one of the eight letters which are encoded on the marker is binarized erroneously.

The encoding of the alphabet with the nine different bit patters, combined with the four black corner-bits, gives rise to desirable visual properties. At least twelve bits are always black, eight are always white, and equally distributed over the marker. Furthermore, at least four bits on each border are set to black. Therefore, our marker does not require an explicit border [8], as it has a sufficient amount of edge pixels along the border.

Figure 3 shows that reading out a marker and a rotated version of the same marker results in a cyclic shift of the codeword by two positions.

5 Marker Detection and Tracking

After recording the experiments, the analysis is done off-line. The markers are detected, read out (Sect. 5.1), and tracked (Sect. 5.2) in the images of the grid cameras. Then the generated trajectories are projected into the world coordinate system using pre-determined heights (Sect. 5.3). Afterwards the trajectories from the different cameras are combined, or stitched, which gives us the path of each participant through the whole set-up (Sect. 5.3).

5.1 Marker Detection

We first find regions-of-interest (ROI), image patches with potential markers. The goal of ROI detection is a hight recall, in order not to lose any markers in this stage.

To detect ROIs, we run Harris corner detection and look for parts of the image with many high-scoring corner points. Given the structure of the marker, we expect many corner point at their location. For each pixel we sum up all the corner responses in a window given by the marker size. This responses space is smoothed and non-maximum suppression is run to produce the ROIs.

The marker detection algorithm runs on these patches. We find lines using Hough transform. From the found lines, marker candidates are generated, which are then binarized and decoded to confirm that an actual marker has been found.

We first run Hough transform, where each pixels votes for a line, weighted by its gradient magnitude. The usual line parametrization (α, d) is used, with the angle $\alpha \in [0, 2\pi)$ and the distance to the origin $d \in [0, D_{max}]$. Then the voting space is collapsed and all votes for each angle accumulated. This allows us to compute the main orientation of the potential marker. We have to take into account that several angles belong to the same main orientation, separated by offsets $\Delta_{off} = \{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$:

$$\alpha_{main} = \operatorname{argmax}_{\alpha} \sum_{d=0}^{D_{max}} \sum_{\delta \in \Delta_{off}} vote(\alpha + \delta, d) \tag{1}$$

Using the main orientation we find lines by using non-maximum suppression on all the votes which have an angle α equal to the main orientation. For this step we fix the angles, and search for line candidates $C_{lines,i}$ along the main angles $\alpha_1 = \alpha_{main}$ and $\alpha_2 = \alpha_{main} + \frac{\pi}{2}$:

$$\begin{aligned} C_{lines,1} &= \{(\alpha_1, d) \mid vote(\alpha_1, d - \epsilon) < vote(\alpha_1, d) > vote(\alpha_1, d + \epsilon)\} \\ C_{lines,2} &= \{(\alpha_2, d) \mid vote(\alpha_2, d - \epsilon) < vote(\alpha_2, d) > vote(\alpha_2, d + \epsilon)\} \end{aligned} \tag{2}$$

This gives us a set of lines from which to sample the 7 by 7 grid lines of the potential marker.

Each pair of lines from the candidate set $C_{lines,i}$ votes for the size of marker bits s_i along the directions given by α_i. All pairs of line candidates (α_i, d_a) , $(\alpha_i, d_b) \in C_{lines,i}$ vote for marker bit sizes $|d_a - d_b|$. Using the lines and sizes, voting for the middle lines d_i of the marker can be performed next, using a similar idea as in the previous step.

We obtain the central point (m_x, m_y) by intersecting the lines given by (α_i, d_i). This yields candidate detections parametrized by $(m_x, m_y, \alpha_1, \alpha_2, s_1, s_2)$. These parameters describe an affine transformation, but the representation is better suited for our problem. We then rescore the candidates using the corner responses inside the area of the marker. Finally, the markers are binarized and read out.

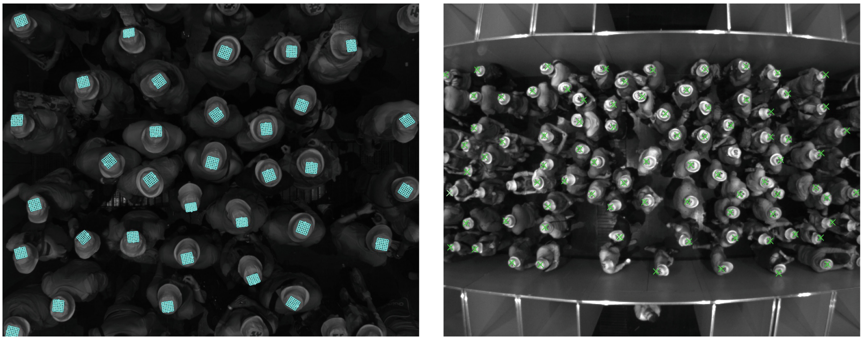
5.2 Trajectory Generation

The tracking of the marker trajectories is in essence single-object tracking, since we can track one ID at a time. As we assume no severe measurement noise, we use the detections and IDs in every frame. Missing detections, which are located between frames with detections, are interpolated using optical flow, which is a procedure adapted from [18].

We use a variation of Lucas-Kanade image registration, which computes the rotation of the image patch as well. This is already outlined in their original paper [14]. We use a special instance of this problem, and register image patches F and G by minimizing:

$$E(t, \alpha) = \sum_{x \in R} \left(G \left(\begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} x + t \right) - F(x) \right)^2 \tag{3}$$

This way, we can interpolate over a gap (up to a maximum of 20 frames) in the trajectory in forward and backward direction, and compare the resulting tracks to improve the quality of the tracking.



(a) A grid camera. (b) Overview camera.

Fig. 4. Camera views with annotated markers.

5.3 3D Localization

To perform 3D localization, we also require the height of each participant. For this we use an additional camera in the grid. This camera shares its entire field of view with two other cameras, so we can cast rays and compute the heights of the pedestrians from the intersections of rays of corresponding marker detections. We use the middle of the 6 by 6 marker grid to define a precise image location.

After detecting and tracking the markers in the grid cameras, the desired 3D information in world coordinates has to be reconstructed. Since this is not directly possible from monocular cameras, we use the marker ID to obtain the height h of each participant. Then, for each tracked marker position, we cast rays and intersect them with a plane which is parallel to the ground plane at height h.

Only then we combine the different views of the grid cameras. All tracklets, the parts of the trajectories as seen in a single view, are stitched together in 3D space. This is done separately for each ID. If one person is seen and detected in the intersection area of two cameras, we take the detection which is closer to the centre of the image plane. Since all the cameras are of the same type, and mounted at the same height above ground, distances in the image plane are directly comparable.

6 Evaluation

In the following, the different steps of the processing pipeline are evaluated.

6.1 Marker Detection

The marker detection and tracking are evaluated in the view of a grid camera. We evaluate the different stages ROI generation, marker detection, and tracking

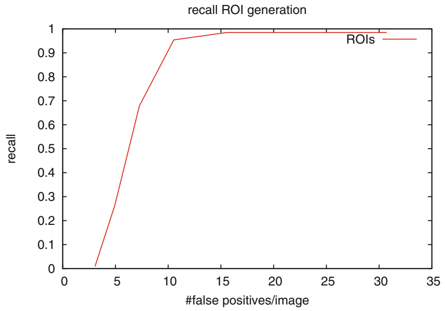


Fig. 5. Performance of the ROI generation.

Table 1. Performance of the marker detection and tracking, using different thresholds for the position.

Step	Recall	Precision	FP / image
ROI	0.99	0.60	15.56
Detection (5px)	0.78	0.93	1.27
Detection (10px)	0.78	0.94	1.24
Tracking (5px)	0.89	0.94	1.27
Tracking (10px)	0.95	0.95	1.12

in image space. As ground-truth data we use an annotated sequence of 600 frames, which contains both a completely empty scene and various densities of pedestrians (see Fig. 4a).

A region-of-interest is considered correct (true positive), if an actual marker falls inside its outline. Only one ROI is excepted per ground-truth detection. The score of a ROI is computed using the scores of the Harris corner points in its image patch. The recall of the ROI generation can be seen in Fig. 5. While the number of false positives per image is substantial, the precision is still satisfactory, reaching 0.60 at a recall of 0.99 (see Table 1), meaning that more than half of the ROIs correspond to actual markers.

Table 1 shows the performance of the different steps of the pipeline. The goal up to this stage is still a high recall, to lose as few actual trajectories as possible before stitching. A marker position, both detected and tracked, is considered correct if it is within 5 or 10 pixels of the actual position. This corresponds to a positioning error of approximately 1.5 cm or 3 cm in our set-up. The tracking is able to increase the recall while keeping the precision, by filling in gaps and rejecting isolated detections.

6.2 Tracking and Stitching

To evaluate the tracking trough the entire set-up, we use the multiple object tracking accuracy (MOTA) metric [12]. As ground-truth data, we use trajectories

Table 2. Performance after stitching. "Mostly Hit" are trajectories for which at least 80 % of the positions have been found.

Experiment	MOTA	Traj	Mostly Hit	Mostly Tracked	Mostly Lost
Corridor	0.87	544	490	374	22
Crossing 90	0.86	599	569	534	8
Crossing 120a	0.87	710	639	587	16
Crossing 120b	0.89	783	688	604	22

which have been manually annotated in the view of the overview camera (see Fig. 4b). Several runs of experiments are used for this purpose, each with a different geometry. We evaluate using the IDs of the trajectories, so there can be no mismatches. Instead, all potential mismatches count as false positives. Furthermore, we report the number of mostly tracked and mostly lost trajectories. "Mostly tracked" means that more than 80 % of the ground-truth trajectory are tracked in one continuous segment. We also report the number of "mostly hit" trajectories, which is the number of trajectories for which more than 80 % of the positions of the ground-truth trajectory are found. The results are listed in Table 2. The "mostly hit" trajectories are fragmented, but are still useful in our case. Since all the data has to be cleaned up manually to be suitable for further research, fragments which can be stitched together still reduce the workload for that task.

7 Conclusion

In this paper, we introduced a new design for ID-markers, which guarantees a minimal distance in the presence of rotated versions of the marker. We presented a pipeline to detect and track these markers. We used them in a system to evaluate large-scale experiments with thousands of pedestrians. This system allows us to produce precisely localized marker trajectories in 3D. In the future, we plan to improve multi-target tracking systems by extracting insights of the behaviour of crowds from the generated data, which could yield better statistical models for tracking.

Acknowledgements. This study was performed within the project BaSiGo (Bausteine für die Sicherheit von Großveranstaltungen, Safety and Security Modules for Large Public Events) funded by the Federal Ministry of Education and Research (BMBF) Program on "Research for Civil Security – Protecting and Saving Human Life". Markus Mathias and Bastian Leibe are funded by the ERC Starting Grant Project CV-SUPER (ERC-2012-StG-307432).

References

1. Qr code - official website. www.qrcode.com
2. Boltes, M., Zhang, J., Seyfried, A., Steffen, B.: T-junction: experiments, trajectory collection, and analysis. In: ICCV Workshops, pp. 158–165 (2011)
3. Boltes, M.: Automatische Erfassung präziser Trajektorien in Personenströmen hoher Dichte. Ph.D. thesis, Forschungszentrum Jülich (2015)
4. Bukáček, M., Hrabák, P., Krbálek, M.: Experimental study of phase transition in pedestrian flow. Transp. Res. Procedia **2**, 105–113 (2014)
5. Chraibi, M., Boltes, M., Schadschneider, A., Seyfried, A. (eds.): Traffic and Granular Flow ’13. Springer, Switzerland (2015)
6. Clark Jr., G.C., Cain, J.B.: Error-Correction Coding for Digital Communications. Springer, New York (1981)
7. Daamen, W., Hoogendoorn, S.: Capacity of doors during evacuation conditions. Procedia Eng. **3**, 53–66 (2010)
8. Daniel, W., Dieter, S.: Artoolkitplus for pose tracking on mobile devices. In: Proceedings of 12th Computer Vision Winter Workshop (2007)
9. Fiala, M.: ARTag, an improved marker system based on artoolkit. Technical report, Institute for Information Technology, National Research Council Canada (2004)
10. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2003)
11. Jelić, A., Appert-Rolland, C., Lemercier, S., Pettré, J.: Properties of pedestrians walking in line: fundamental diagrams. Phys. Rev. E **85**(3), 036111 (2012)
12. Keni, B., Rainer, S.: Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. **2008** (2008)
13. Lemercier, S., Moreau, M., Moussaïd, M., Theraulaz, G., Donikian, S., Pettré, J.: Reconstructing motion capture data for human crowd study. In: Allbeck, J.M., Faloutsos, P. (eds.) MIG 2011. LNCS, vol. 7060, pp. 365–376. Springer, Heidelberg (2011)
14. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI, pp. 674–679 (1981)
15. Reed, I.S., Solomon, G.: Polynomial codes over certain finite fields. J. Soc. Ind. Appl. Math. **8**(2), 300–304 (1960)
16. Schadschneider, A., Klingsch, W., Klüpfel, H., Kretz, T., Rogsch, C., Seyfried, A.: Evacuation dynamics: empirical results, modeling and applications. In: Meyers, R.A. (ed.) Encyclopedia of Complexity and System Science, pp. 3142–3176. Springer, New York (2009)
17. Secoando, F., Plagemann, C., Jiménez, A.R., Burgard, W.: Improving rfid-based indoor positioning accuracy using gaussian processes. In: 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN) (2010)
18. Shi, J., Tomasi, C.: Good features to track. In: CVPR, pp. 593–600 (1994)
19. Stuart, D., Christensen, K., Chen, A., Kim, Y., Chen, Y.: Utilizing augmented reality technology for crowd pedestrian analysis involving individuals with disabilities. In: Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (2013)
20. Weidmann, U., Kirsch, U., Schreckenberg, M. (eds.): Pedestrian and Evacuation Dynamics 2012. Springer, Switzerland (2014)
21. Zhan, B., Monekosso, D.N., Remagnino, P., Velastin, S.A., Xu, L.Q.: Crowd analysis: a survey. Mach. Vis. Appl. **19**(5–6), 345–357 (2008)
22. Zhang, Z.: A flexible new technique for camera calibration. PAMI **22**(11), 1330–1334 (2000)