Sport Players Detection and Tracking With a Mixed Network of Planar and Omnidirectional Cameras

Alexandre Alahi*,Yannick Boursier*,Laurent Jacques† and Pierre Vandergheynst*
*Institute of Electrical Engineering
Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
†Communications and Remote Sensing Laboratory
Université catholique de Louvain (UCL)
B-1348 Louvain-la-Neuve, Belgium

***Abstract*—A generic approach is presented to detect and track people with a network of fixed and omnidirectional cameras given severely degraded foreground silhouettes. The problem is formulated as a sparsity constrained inverse problem. A dictionary made of atoms representing the silhouettes of a person at a given location is used within the problem formulation. A re-weighted scheme is considered to better approximate the sparsity prior.**

Although the framework is generic to any scene, the focus of this paper is to evaluate the performance of the proposed approach on a basketball game. The main challenges come from the players' behavior, their similar appearance, and the mutual occlusions present in the views. In addition, the extracted foreground silhouettes are severely degraded due to the polished floor reflecting the players, and the strong shadow present in the scene. We present qualitative and quantitative results with the APIDIS dataset as part of the ICDSC sport challenge.[1]

***Index Terms*—Inverse problem, Sparsity, Sport player, Detection, Tracking.**

I. Introduction

Vision-based sport players detection and tracking has been of interest to the research community for the past decades, with previous work on soccer [1][2][3], hockey [4][5], or even volleyball [6]. Game or player analysis, sport summarization, and statistics gathering rely on an accurate detection and tracking process. In this work, basketball players are of interest. More precisely, this paper is dedicated to the challenge program consisting of detecting and tracking the basketball players within the APIDIS dataset[2].

The APIDIS dataset has the following challenges:

- Basketball players have abrupt changes of behavior, *i.e.* they run, jump, crouch, change suddenly their motion path, etc.
- Players on the same team have the same appearance.
- In some camera views, players greatly occlude each other.
- Some cameras have very similar viewpoints, affecting the resolution of the ambiguities arising with the occlusion problem.
- The reflection of the players on the floor and their strong shadows lead to severely degraded foreground silhouettes. Many false positives silhouettes are extracted with a standard background subtraction algorithm (*e.g.* the work of Stauffer and Grimson [7]).
- Players interact strongly with each other and their spatial distribution on the ground can be very dense and compact or spatially scattered.

A novel approach is proposed to cope with all these challenges. Given a set of severely degraded foreground silhouettes from a set of calibrated pseudo-synchronized cameras, all players are detected and tracked in each camera view as well as in the 3D world. The proposed algorithm is based on a sparse approximation of players location points on the ground floor. A dictionary made of atoms representing the presence of a player at a given location is used within an inverse problem formulation. The approach is generic to any camera geometry and sensing modality. The only features extracted from the cameras are the binary masks representing the pixels belonging to the foreground of the scene (referred to as the foreground silhouettes in the paper). Planar and omnidirectional cameras are naturally merged. The framework scales to any number of cameras. Although the proposed system is generic to any scene, this paper is dedicated to quantitatively measure the performance of our algorithm on a basketball game. No tuning for this specific basketball environment is performed. No texture or color information is used for the modeling of players, background, or ground floor. A basic background subtraction algorithm ([7]) is used leading to severely degraded features.

After briefly presenting related works, the problem is formulated in section III as an inverse problem with a sparsity prior. Basis Pursuit DeNoising (BPDN) is used with a ℓ_1 re-weighting scheme to solve the problem. The dictionary involved in the problem is described in the next section. In order to reduce the complexity of the problem, section V presents the steps used to reduce the dimensionality of the problem. Then, the tracking process is described in section VI. Quantitative and qualitative results are presented in section VII. The proposed algorithm is compared with the state of the art. The paper ends with concluding remarks.

[1]This work is submitted to the sport challenge track
[2]The dataset is publicly available at http://www.apidis.org/Dataset/

II. Related Work

Porikli in [8] presents a survey on object detection and tracking methods with a single fixed camera. Given a fixed camera, objects can be detected by modeling the background and tracking reduces to computing object correspondence across frames. Typically, the work of Stauffer and Grimson [7] can be used to extract the foreground pixels. Each pixel is modeled as a mixture of Gaussians with an on-line approximation for the update. Then, inter-frame tracking algorithms such as the one presented by Avidan in [9] can be used to track people across frames. He considers the tracking as a binary classification problem. However, those algorithms are less efficient to detect players in a basketball game, as players are not correctly segmented due to their mutual occlusions.

In order to deal with occlusions, the output of several cameras should be fused to detect the objects of interest. To be robust to variability in appearance between the views, coordinates of the detected objects in a common reference (*e.g.* ground plane) can be estimated. The unique 'world' coordinates, *i.e.* the coordinate of the object on the ground plane, is given by a planar homography. The planar homography is a 3×3 matrix transformation obtained by matching at least four points from two different coordinates. Most systems compute the homographies in an initial calibration step [10]. After projecting all detected objects into a common reference, Mueller *et al.* in [10] mark with same label the nearest object with the same size and center of gravity. Orwell *et al.* in [11] and Caspi *et al.* in [12] match objects by fusing the estimated trajectories obtained by each camera. However, such approaches do not take full advantage of the multi-view infrastructure, as each camera detects the objects independently without helping each other.

Khan and Shah in [13] present an approach to track people in crowded scene given multiple cameras. They pay attention to extract the feet region of the foreground people. Each point of the foreground likelihood (foreground silhouettes) from all views is mapped to the ground plane given a planar homography. Multiplying the mapped points segments the pixel corresponding to the feet of the people. Their approach can not be applied to an object viewed by one camera. In addition, a poor foreground segmentation - people detected with their shadow or missing foreground pixels - affects the performance of their system.

Fleuret *et al.* in [14] take advantage of the multi-view infrastructure to accurately track people across multiple cameras given degraded foreground silhouettes. They develop a mathematical framework to estimate the probabilities of occupancy of the ground plane at each time frame with a dynamic programming to track people over time. They approximate the occupancy probabilities as the marginals of a product law minimizing the Kullback-Leibler divergence from the true conditional posterior distribution (referred to as Fixed Point Probability Field algorithm). These probabilities are combined with a basic color and motion model. As in [13], their approach can not be applied to an object viewed by one camera. Their mathematical framework does not explicitly consider the sparsity of the desired solution. In addition, the computation cost of their algorithm depends on the number of ground plane points to be evaluated, leading to a limited area to be monitored. We will consider their work as the state of the art since they are able to handle challenging scenarios given noisy observation.

Reddy *et al.* in [15] use compressed sensing theory to track people in a multi-view setup. They use the sparsity of the observations, *i.e.* the foreground silhouettes extracted from the cameras. However, their sparsity constraint depends on the distance of the objects to the cameras. Objects close to the cameras will unfortunately generate large foreground silhouettes with poor sparsity. To accurately estimate the position of the objects on the ground plane multiple cameras are needed. No dictionary is used to model the presence of a person. Also, the complexity cost of their algorithm depends on the number of ground plane points, the grid size, to be evaluated.

In [16], we propose a framework to cope with the limitations of previous works. It scales to any number of cameras. A single camera can also be used. The proposed algorithm is based on a sparse approximation of the location points given an adaptive dictionary constructed on-line. The dictionary is made of atoms representing the foreground silhouettes of a player at a given location. The sparsity constraint in this work is not on the observations but on the desired solution. Even when a dense crowd of people is present in a scene, they occupy sparse locations on the ground plane.

In this paper, we quantitatively evaluate its performance given a basketball game. We compare our approach with the state of the art. In addition, omnidirectional cameras are also integrated to the system. The strength of the proposed approach is quantitatively and qualitatively presented in section VII.

III. Problem Formulation

The objective of this paper is to deduce the ground plane points (or *grid of occupancy*) occupied by the players present in the game given the foreground silhouettes provided by a set of C calibrated cameras. Mathematically, at a given time, each camera is the source of a binary silhouette image $y_i \in \{0,1\}^{M_i}$, where $M_i \in \mathbb{N}$ is the number of pixels of each camera indexed by $1 \leq c \leq C$. Stacking all these vectors gives the Multi-Silhouette Vector (MSV) $y = (y_1^T, \cdots, y_N^T)^T \in \{0,1\}^M$, with $M = \sum_i M_i$.

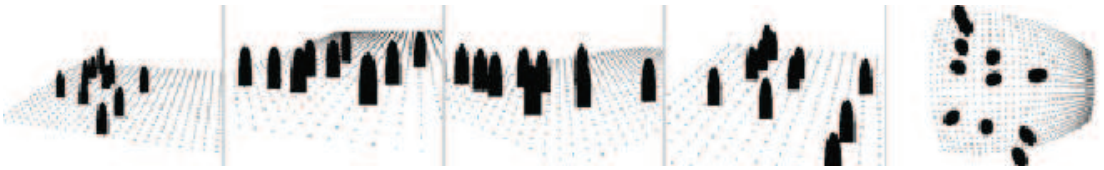
Let us discretize the observed ground in N subareas, so that the grid of occupancy of players on the ground is represented by the binary vector[3] $x \in \{0,1\}^N$. For simplicity, we assume that one observed player is exactly supported by one subarea of this grid.

Assuming that any player is represented by an invariant volume, it is clear that any configuration of x will correspond

[3]The grid is of course bidimensional but we represent it as a vector to simplify the notations.

(a) Camera Views

(b) Degraded foreground silhouettes y



(c) $\hat{y} = Q(Dx)$

Fig. 1. Illustration of the atoms modeling the given foreground silhouettes.

to a particular configuration of silhouettes in y. For instance, if x contains only one non-zero component, all y_i will contain only one silhouette (i.e. a connected area of non-zero pixels) with size and location related to the particular projective geometry combining the scene and the cameras.

Our inverse problem is thus to find x from y. The vector y is binary and does not contain any information about possible occlusion between persons. In addition, the background subtracting methods leading to the silhouette definition are severely degraded. We assume the grid x sparse, i.e. it has only few non-zero coefficients.

The generative (forward) model that associates to x a certain configuration of silhouette in y is the quantization of a linear operator. In other words, we obtain it from the one bit quantization of a *dictionary* $D \in \mathbb{R}^{M \times N}$ multiplied by x. The quantization operator is dealing with the occlusions of silhouettes. As explained in Section IV, each column (or atom) of D transforms a non-zero component of x to the corresponding silhouette generated by a player in the image plane of the cameras.

The recovery of the grid of occupancy from y should be solved by the following program:

$$\arg\min_{x} \|x\|_0 \quad \text{s.t.} \quad \|y - Q(Dx)\|_2 < \varepsilon \tag{1}$$

with ε is the desired residual error, and Q is the one bit quantization operator

The ℓ_0 norm is the appropriate measure of the sparsity since it counts the number of non-zero elements. However, minimizing ℓ_0 norm is prohibitive in high dimensions and is NP-hard. We approximate the ℓ_0 norm with a re-weighted ℓ_1 norm where the weights used for the next iteration are computed from the value of the current solution as introduced by Candes *et al.* in [17]:

$$x^{(l+1)} = \arg\min_{u \in R^N} \|W^{(l)} u\|_1 \text{ s.t. } \|y - Du\|_2 < \varepsilon, \tag{2}$$

$$W^{(l+1)} = \text{diag}\left(\frac{1}{|x_1^{(l+1)}|+\eta}, \cdots, \frac{1}{|x_N^{(l+1)}|+\eta}\right), \tag{3}$$

with $W^0 = \text{Id}$.

The parameter η is added to ensure stability and guarantees that a zero-valued component in x does not strictly prohibit a nonzero estimate at the next iteration ($\eta = 10^{-7}$).

Each weighted iteration of the re-weighted process is solved by the method of operator splitting and proximal methods described in [18], [19].

We removed the quantization operator in equation (2) to keep the problem linear and convex. The residual error is hence affected by the absence of the quantization operator and the approximation of the degraded foreground silhouettes.

IV. Dictionary

The dictionary D is made of atoms modeling the presence of a single player at a given location. Each atom of the dictionary represents the approximated silhouette generated by a single player in the image plane of each camera, i.e the MSV. The silhouettes are the atoms of the dictionary approximated with simple shapes (*e.g.* rectangular or elliptical shapes). Planar homographies computed at the calibration step are used to map 3D points in the world coordinates to the image plane of each planar camera (see [16]). The camera model presented in [20] is used to map points from the 3D world to the image plane of the omnidirectional cameras.

To cope with the various poses and shapes a person can generate in a camera view, a half-elliptical half-rectangular

shape is used to approximate the ideal silhouette of a person in a planar camera, and an elliptical shape is used for the omnidirectional cameras. Figure 1 illustrates an example of severely degraded foreground silhouettes (*e.g.* shaddows, player's reflection, missed regions) and the silhouettes used to model their presence in the set of planar and omnidirectional cameras.

The silhouettes associated to the c^{th} camera lead to the dictionary D_c. The product $D_c x$ corresponds to the synthetic image $\hat{y}_c$. Each column (atom) of the dictionary D_c is the image of a single silhouette reshaped as a vector (dictionary of silhouettes). There are as many columns as ground plane points. When several cameras are observing a scene, D is formed by stacking the D_c:

$$D = (D_1^T, D_2^T, ..., D_n^T)^T \tag{4}$$

where n is the number of cameras.

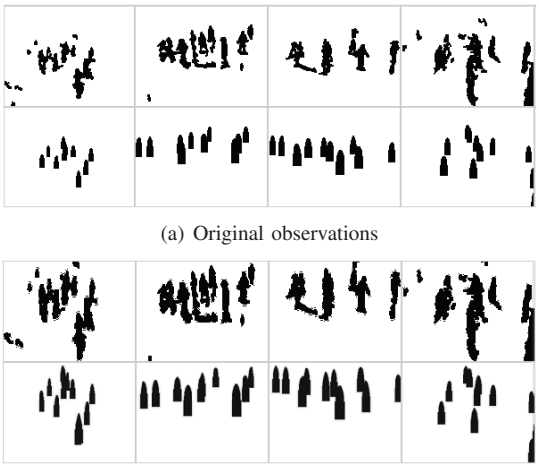
With such a model, there is no constraint on the number of cameras to use and the type of cameras, *i.e.* planar or omnidirectional.

V. Dimensionality Reduction

If many cameras are used, the dimensions of y and x become an issue in equation (2). They define the dimensionality of D which requires a large memory storage. In addition, the smaller the dimensions, the faster the algorithm converges towards an optimal solution.

A. Dimensionality reduction on the observations

The observations y have originally the same dimensions as the image resolution of all cameras. To reduce the computation cost, all images are first down scaled to a QVGA resolution (320×240). A background subtraction algorithm extract to foreground silhouettes on the QVGA resolution. Then the image plane of each camera view is cropped to the region where players can occur. Finally, all regions are normalized to the same size (107×80). Figure 2 illustrates such basic preprocessing step.



(a) Original observations

(b) Cropped and down scaled observations

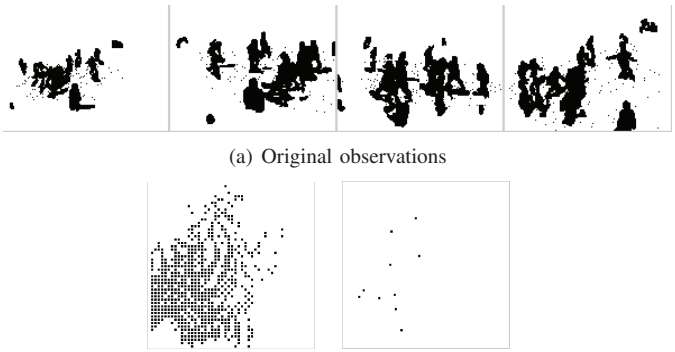
Fig. 2. Illustration of the dimensionality reduction on the observations

B. Dimensionality reduction in the search space

The complexity cost depends on the number N of ground plane points to locate as occupied or not. Previous works considered a fixed number of points regardless of the geometry of the scene and the sparsity of the people present in the scene. In addition, the ground plane points were sampled uniformly without considering the resolution of the observations. In this work, the ground plane points are sampled based on the resolution of the observations.

Two different ground plane points can correspond to the same pixel in the image plane of a camera. A translation of one pixel in the image plane can be equivalent to a translation of a few meters on the ground plane for far away regions, just as a translation of a few centimeters on the ground plane can correspond to a shift of several pixels for closer regions, depending on the resolution of the camera and the distance of the objects to the camera. Therefore, localization should be on a restricted number of ground plane points, called sample points.

The foreground pixels determines the sampling points to be used (see Figure 3). Each foreground pixel represents the potential feet location of the players. Each point is spaced by the 'just noticeable pixel step'. It represents the pixel accuracy to detect an object in the image plane. Given the calibration data, each point of each camera is mapped to a ground plane point forming x. In order to be certain not to miss a potential ground point, each foreground pixel is also considered as the upper limit (the head) of a player. Therefore, missing the feet region in the foreground likelihood will not affect the sampling process.



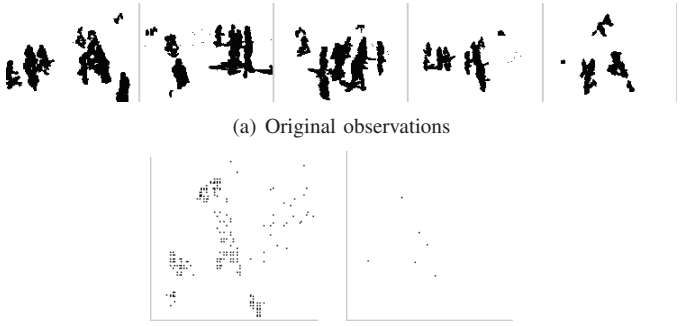
(a) Original observations

(b) Sampled points used to form x (top view)

(c) Players's exact locations (top view)

Fig. 3. Illustration of the sample points obtained with the given foreground silhouettes. Isolated points in the observations are sample points obtained from other cameras.

A further reduction in the obtained sampling points can be performed. We can keep only the sample points that are foreground pixels in all the observable cameras. Figure 4 presents the final ground plane points forming x with such constraint.



(a) Original observations

(b) Sampled points used to form x (top view)

(c) Players's exact locations (top view)

Fig. 4. Illustration of the sample points corresponding to the foreground pixels of all observable camera.

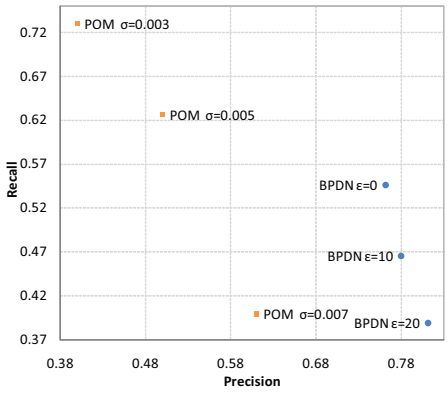


Fig. 5. Precision and recall rate for various ε given three cameras monitoring the scene (camera's id 2, 4, 5, and 7)

VI. Tracking

A. Prior from previous frame

The final sparse vector x found from previous frame can be used in the current frame as the initialization of the problem to minimize. Therefore, the algorithm could converge much faster to the new optimal solution. One can also reduce the search space, however, since new players can occur in a view, it is safer to keep the search space provided by the foreground silhouettes described in section V-B. In addition, if false detections occur in some frames, they will not affect the performance of future frames.

B. Player correspondence

Given the detected players from two consecutive frames, a correspondence between each player is needed across frames. Such correspondence is performed in the 3D world. The coordinates of the players on the ground are used to determine the correspondence. Their appearance can not be used since many players have similar appearance.

Any motion model can be applied to the person occupying a given point. The probability to match two points from two frames, can be computed by modeling the player behavior. The work of Antonini *et al.* in [21] can be used to model the player behavior. In this work, a simple motion model is used leaving for further work a more sophisticated modeling. A point is matched to the one with the shortest distance within a disk of radius equal to the maximum speed of a player.

VII. Performance Evaluation

A. Experiments

Since this paper is dedicated to the sport track of the ICDSC challenge[4], focus is on the APIDIS dataset. All videos are scaled to a QVGA resolution with approximately 25 fps. Performance over the left-half of the basketball court is measured since it is the side where the most number of cameras are monitoring the game (*i.e.* camera's id 1, 2, 4, 5, and 7).

[4]http://www.icdsc.org/challenge.html

The performance of the detection process is quantitatively evaluated by computing the precision (*i.e.* number of true positives divided by the sum of true positives and false positives) and recall (*i.e.* number of true positives divided by the sum of true positives and false negatives) measures. A true positive is when a person is correctly located on the ground plane.

The performance of the tracking process is measured by counting the number of missed tracked players.

The foreground silhouettes are extracted using the work of Stauffer and Grimson [7]. The outcome of the background subtraction algorithm is noisy. The silhouettes are severely degraded. Only part of the people are extracted, their shadow is considered, and random false positives are generated.

B. Impact of the residual error

The appropriate value for the residual error ε for the fidelity term in equation (2) is difficult to determine since the degradations on the foreground silhouettes and the number of occlusions are not predictable. Nevertheless, the performance of the algorithm depends on the chosen desired residual error in equation (2). Figure 5 presents the performance of the framework with four cameras for various ε. A high residual error expresses the severely degraded foreground silhouettes made of shadows, and missing points. The detection stage described by Fleuret *et al.* in [14] referred to as the Probability Occupancy Map (POM) is also evaluated given the same foreground silhouettes. We set the maximum of iterations to 1500 and vary their constant σ. Their approach is quite sensitive to noise, and detects many false positives. For an equivalent recall rate, our approach has a much better precision rate. Nevertheless, they reach a high recall rate with the price of a very low precision rate. In section VII-E, we present how we can obtain a recall rate as high as the highest rate reached with the POM approach with the difference that we also obtain a high precision rate. The better precision rate that we obtain can be justified as we explicitly consider the sparsity of our problem in the mathematical formulation.

C. Impact of tracking

The focus of this work is not on the tracking algorithm. However, with our simple motion model, players are correctly tracked across frame regardless their abrupt behavior and similar appearance. Videos illustrating the performance of our approach can be found at this address [22]. Note that the bottle neck of the approach is the detection stage. A missed tracked player is due to a missed detection. Therefore, the emphasis is on an accurate detection process.

D. Impact of multi-view

One of the advantages of our framework is that it scales to any number of cameras. Therefore, we compare the performance of the system with various number of cameras. We pick a camera and measure its performance when other cameras are monitoring the same scene in Figure 6. A true positive is when a player is correctly located in the particular view, *i.e.* when the overlap region between the proposed and the manually annotated rectangular bounding boxes represents at least 50% of the area of both bounding boxes. The performance of the single camera monitoring the scene is low if other cameras are not considered: $R = 0.57$ and $P = 0.62$. When another planar camera is used, both rates increase: $R = 0.64$ and $P = 0.64$. If an omnidirectional camera is added instead of a planar, the performance is considerably increased: $R = 0.76$ and $P = 0.72$. Figure 6 shows that adding planar cameras globally increase the performance. In addition, merging an omnidirectional camera with other planar cameras have the best performance. Nevertheless, it is interesting to note that if the omnidirectional camera is monitoring the scene alone, a poor detection is achieved due to the severely degraded foreground silhouette: $R = 0.47$ and $P = 0.55$. Most of the time, the people's shadow is much bigger than its silhouette affecting considerably the performance. In addition, in some areas, people are almost missed by the background subtraction algorithm since they occupy only few pixels (small surface). Finally, due to the small bounding box of the people, a small offset in the detection considerably affects the performance and leads to a missed person. Figure 8 presents the foreground silhouettes extracted and the detected people given various number of cameras.

E. Future Work

Although we propose a framework to efficiently deal with simple and noisy features (see Figure 7) in a well defined mathematical formulation, future work can use a less-degraded foreground extraction in order to improve the detection.

In addition, the proposed inverse problem can be solved with various strategies. We use BPDN to solve the problem while other techniques such as Lasso [23] formulation can also be used. The Lasso formulation bounds the desired solution instead of the fidelity term. Therefore, the error term to define becomes the maximum number of players to be detected, which is more predictable than the degradation of the foreground silhouettes. Given such formulation, we will present in a further work the gain in performance.

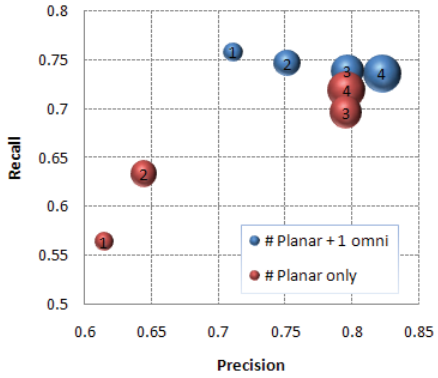


Fig. 6. Precision and recall rate for various number of cameras monitoring the scene. The number in each bubble represents the number of cameras used. First, the sequence of cameras id 5,7,2,4 (omni+planar) is used. Then, the sequence of cameras id 7,2,4,1 is used (planar only).

VIII. Conclusions

Formulating the localization of people as a sparse constrained inverse problem enables the detection and tracking of challenging scenarios. The strength of the proposed approach is quantitatively and qualitatively illustrated on a basketball game. Players are correctly detected and tracked given very noisy features. The approach is generic enough to be used with any calibrated camera. Planar and omnidirectional cameras are naturally merged. Any number of cameras can be used. The multi-view infrastructure is fully taken into consideration during the detection process. Furthermore, detected players are perfectly matched across cameras so that their reconstruction from all the views can be performed. Since the coordinates of the players are computed in the 3D world, each player can have a flag informing if a clear visualization is available in a view, *i.e.* other players are not occluding. Therefore, further processing such as recognizing its label can be performed.

References

[1] Y. Seo, S. Choi, H. Kim, and K.S. Hong, "Where are the ball and players? soccer game analysis with color based tracking and image mosaick," *Lecture Notes in Computer Science*, pp. 196–203, 1997.

[2] C.J. Needham, R.D. Boyle, University of Leeds, and School of Computer Studies, *Tracking multiple sports players through occlusion, congestion and scale*, University of Leeds, School of Computer Studies, 2001.

[3] J. Liu, X. Tong, W. Li, T. Wang, Y. Zhang, and H. Wang, "Automatic player detection, labeling and tracking in broadcast soccer video," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 103–113, 2009.

[4] Y. Cai, N. de Freitas, and J.J. Little, "Robust visual tracking for multiple targets," *Lecture Notes in Computer Science*, vol. 3954, pp. 107, 2006.

[5] W.L. Lu, K. Okuma, and J.J. Little, "Tracking and recognizing actions of multiple hockey players using the boosted particle filter," *Image and Vision Computing*, vol. 27, no. 1-2, pp. 189–205, 2009.

[6] T. Mauthner and H. Bischof, "A Robust Multiple Object Tracking for Sport Applications," in *Performance Evaluation for Computer Vision, 31st AAPR/OAGM Workshop*, 2007, pp. 81–89.

[7] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 246–252, 1999.

[8] F. Porikli, "Achieving real-time object detection and tracking under extreme conditions," *Journal of Real-Time Image Processing*, vol. 1, no. 1, pp. 33–40, 2006.
[9] S. Avidan, "Ensemble tracking," *Pattern Analysis and Machine Intelligence*, pp. 261–271, 2007.
[10] K. Mueller, A. Smolic, M. Droese, P. Voigt, and T. Wienand, "Multi-texture modeling of 3d traffic scenes," *icme*, vol. 2, pp. 657–660, 2003.
[11] J. Orwell, S. Massey, P. Remagnino, D. Greenhill, and G. A. Jones, "A multi-agent framework for visual surveillance," in *ICIAP '99: Proceedings of the 10th International Conference on Image Analysis and Processing*, Washington, DC, USA, 1999, p. 1104, IEEE Computer Society.
[12] Y. Caspi, D. Simakov, and M. Irani, "Feature-based sequence-to-sequence matching," *International Journal of Computer Vision*, vol. 68, no. 1, pp. 53–64, June 2006.
[13] S.M. Khan and M. Shah, "A multiview approach to tracking people in crowded scenes using a planar homography constraint," in *European Conference on Computer Vision 2006*, 2006, pp. IV: 133–146.
[14] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multi-camera people tracking with a probabilistic occupancy map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.
[15] D. Reddy, A.C. Sankaranarayanan, V. Cevher, and R. Chellappa, "Compressed sensing for multi-view tracking and 3-D voxel reconstruction," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, 2008, pp. 221–224.
[16] A. Alahi, Y. Boursier, L. Jacques, and P. Vandergheynst, "A Sparsity Constrained Inverse Problem to Locate People in a Network of Cameras," in *16th International Conference on Digital Signal Processing*, Aegean island of Santorini, Greece, 2009.
[17] EJ Candès, MB Wakin, and SP Boyd, "Enhancing sparsity by reweighting ℓ_1 ," Tech. Rep., Tech. Rep., California Institute of Technol., 2007 [Online]. Available: http://www. acm. caltech. edu/ emmanuel/publications. html.
[18] P.L. Combettes, "Solving monotone inclusions via compositions of nonexpansive averaged operators," *Optimization*, vol. 53, no. 5, pp. 475–504, 2004.
[19] M.J. Fadili and J.-L. Starck, "Monotone operator splitting for fast sparse solutions of inverse problems," *SIAM Journal on Imaging Sciences*, 2009, submitted.
[20] J. Kannala and S.S. Brandt, "A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 8, pp. 1335, 2006.
[21] G. Antonini, M. Bierlaire, and M. Weber, "Discrete choice models of pedestrian walking behavior," *Transportation Research Part B*, vol. 40, no. 8, pp. 667–687, 2006.
[22] "http://lts2www.epfl.ch/ alahi/data.htm," .
[23] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

ACKNOWLEDGMENT

The authors would like to thank the CVLAB [14] and M.J. Fadili [19] for their source codes. Y. B. is a post-doctoral researcher funded by the European APIDIS project. L. J. is a Postdoctoral Researcher of the Belgian National Science Foundation (F.R.S.-FNRS).

Fig. 7. Illustration of the degraded foreground silhouettes extracted.

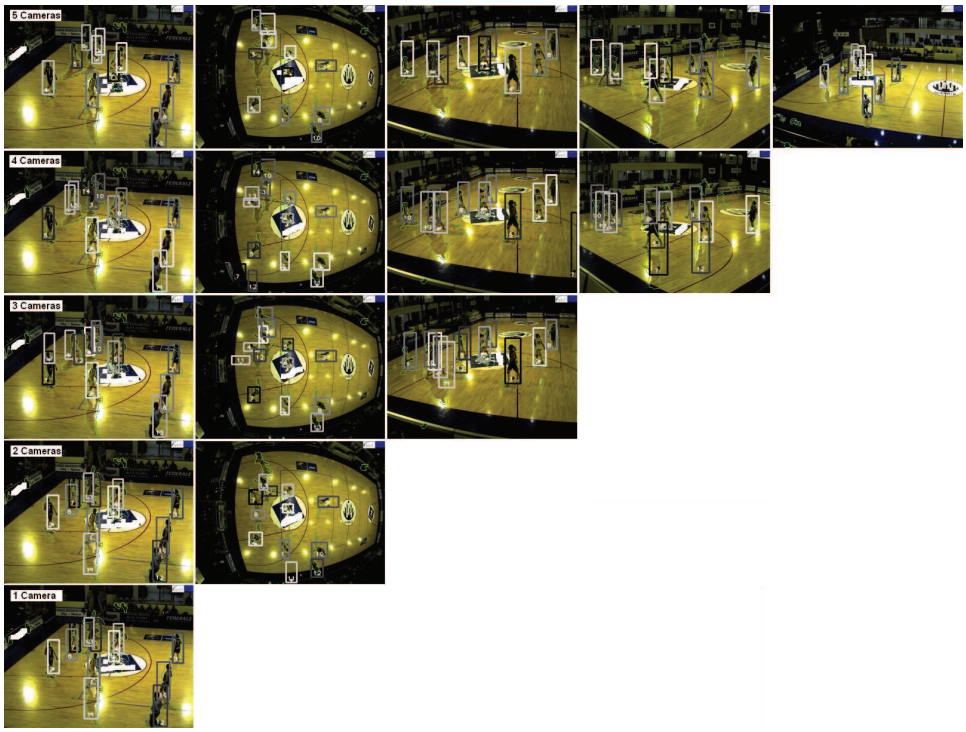


Fig. 8. Illustration of the detected players with various number of cameras.