Continuous Localization and Mapping of a Pan Tilt Zoom Camera for Wide Area Tracking

Giuseppe Lisanti\textsuperscript{a}, Iacopo Masi\textsuperscript{a}, Federico Pernici\textsuperscript{a}, Alberto Del Bimbo\textsuperscript{a}

\textsuperscript{a}Media Integration and Communication Center, University of Florence

Abstract

Pan-tilt-zoom (PTZ) cameras are powerful to support object identification and recognition in far-field scenes. Real-time detection and tracking of targets with these cameras is nevertheless complicated by the fact that the geometrical relationship between the camera view and the 3D observed scene is time-varying and, over long periods of operation, real world scenes exhibit changes either due to objects or context variations. In this paper, we present a solution that provides continuous estimation of the camera pose and mapping from scene landmarks, which is robust to rapid camera motion, changes of the environment due to illumination or objects, and scales beyond thousands of landmarks. The estimation of camera pose permits to compute the relationship between the positions of the targets in the 3D world plane and their corresponding positions in the 2D image, and to perform real-time tracking of multiple targets in world coordinates. Since camera motion is compensated, an absolute reference coordinate system can be used and targets can be tracked in the world plane with high and stable degree of accuracy also at large distances and any zooming level. We provide experimental evidence that the solution proposed is capable of supporting effective real-time tracking of multiple targets with PTZ cameras.

Keywords: Rotating and Zooming Camera, PTZ Sensor, Localization and Mapping, Real Time Multiple Target Tracking, Single View Geometry.

1. Introduction

Pan-tilt-zoom (PTZ) cameras are powerful to support object identification and recognition in far-field scenes. They are equipped with adjustable optical zoom lenses that can be manually or automatically controlled to permit both wide area coverage and close-up views at high resolution. This capability is particularly useful in surveillance applications to permit tracking of multiple targets in high resolution and zooming in...
on biometric details of parts of the body in order to resolve ambiguities and understand target behaviors.

However, their practical use in automated video surveillance is complicated by the fact that with this type of camera, real time detection and tracking of targets is challenging. On one hand, due to panning, tilting and zooming, the geometrical relationship between the camera view and the 3D observed scene is time-varying. So, in order to provide precise localization and tracking of moving targets, at each time instant this relationship must be estimated and camera motion compensated. This is a complex and time consuming operation. Values from the PTZ camera motor encoders can be exploited but these measurements are too inaccurate to be used effectively [1].

On the other hand, due to the mode of operation we cannot assume any fixed background for target detection. Besides, over long periods of operation, real world scenes also exhibit changes either due to objects or to context variations (Fig. 1). According to this, well known detection methods based on background subtraction cannot be used, and some adaptive representation of the scene under observation is also necessary.

Moreover, since we must expect that the imaged targets undergo frequent scale changes, classical target detectors that use specialized classifiers [2, 3, 4] are inadequate, since they would require too much computational effort to permit real-time operation.

In the following, we present a solution that provides continuous adaptive calibration of a PTZ camera and enables effective real-time tracking of multiple targets, so to fully exploit the superior capabilities of PTZ cameras for surveillance applications in large areas. In an offline stage, we collect scene landmarks from a finite number of keyframes taken from different viewpoints. At run time, at each time instant, we estimate the homographic transformation between the camera view and the 3D world plane by matching the keypoints in the current view with those extracted from the keyframes in the offline stage. As a result, camera motion is compensated and the relationship between target positions in the 3D world plane and positions in the 2D image is continuously maintained. Changes in the scene that have occurred over time due to illumination or objects are accounted for with an adaptive representation of the scene under observation that models and updates the uncertainty in landmark localization. Tracking of targets is performed in real world coordinates with a high and stable degree of accuracy, as well as with continuity in the presence of occlusions when targets exit the scene for a few frames.

Figure 1: Real world scene with changes due to objects and illumination variations. Planar mosaics from a PTZ camera.
2. Related work

In the following, we review research papers that are relevant to the scope of this work and have connections with the method proposed. In particular, we review solutions for self-calibration and target tracking with moving and PTZ cameras.

PTZ camera self-calibration

Self-calibration of moving cameras has been the subject of several research papers. Both offline and online methods have been proposed, some of which have specifically considered PTZ cameras.

A paper by Hartley [5] first demonstrated the possibility of performing offline camera self-calibration based on only image content for moving cameras undergoing pure rotation. This method was later extended in many different ways (see [6]) and has also been assumed as the central building block of mosaicing [7, 8]. More recently, Sinha and Pollefeys [9] used the same method to perform offline self-calibration of PTZ devices in a camera network. For each camera, they used feature matching and bundle adjustment to compute the approximate values of the intrinsic parameters of a set of view images taken at different pan and tilt angles and the lowest zooming level. Afterwards, other view images at increasing zooming levels are acquired and processed in the same way to estimate the intrinsic camera parameters for the full range of zooming. The mosaics of the camera views are hence matched each other, and the extrinsic camera parameters are estimated for each camera.

Solutions to online estimation of camera pose for moving and PTZ cameras were proposed respectively in [10, 11] and [12, 13]. In [10], Klein and Murray presented a method for real time pose estimation of a moving camera, where they applied online bundle adjustment to the five nearest keyframes sampled every ten frames of the sequence. Unfortunately, this approach cannot be used with PTZ cameras when applied to monitor large areas, since it is likely to produce over-fitting in the estimation of the camera parameters. In [11], re-localization of the camera was performed by using a randomized lists classifier to find the correspondences between the features in the current image and pre-calculated features from all the possible views of the scene, and hence applying RANSAC to obtain the correct camera pose. However, their experiments were performed in indoor environments and the scene under observation was restricted to a relatively small area. Civera et al. [12], proposed a method to perform real-time sequential mosaicing of a scene with a rotating camera. They used Extended Kalman Filter (EKF) and SLAM to estimate the location and orientation of the camera. They only considered the case of camera rotations, and did not account for any zooming operations. The landmarks extracted from the scene were included in the filter state. Due to this, the solution cannot scale with the number of features, and is likely to show very poor accuracy as the number of features grows beyond a few hundred. Lovegrove et al. [13] claimed to provide full PTZ camera self-calibration but did not demonstrate self-calibration with variable focal length. As an alternative to using EKF sequential filtering, these authors suggested to use keyframes to achieve scalable performance. The camera parameters between consecutive images were obtained by simple image alignment.
The main limitations of these approaches are that they all assume that the scene is almost stationary that changes are only due to camera motion and there is no zooming operation. These assumptions are clearly inappropriate for tracking with PTZ cameras in real applications.

A few authors have attempted to solve the so-called hand-eye calibration problem [14] using the information provided by the built-in motor actuators of PTZ cameras. The authors of [1] defined a PTZ camera model that explicitly accounts for the drift of camera calibration over many hours of operation by reflecting how focal length and lens distortion vary as a function of zoom scale. A few images were used for initialization, followed by a nonlinear optimization. Results have shown a better calibration accuracy with respect to [9], especially at high zoom scales. However, as reported by the authors, the system fails when a large component in the scene has been moved or when the background is changing slowly. A similar hand-eye calibration strategy was also applied in [15], but accounted for pan and tilt camera movements only.

Tracking with PTZ cameras

General tracking methods cannot perform well on sequences from PTZ cameras. This is mainly due to the fact that with moving sensors targets undergo large and abrupt scale changes that cannot be handled unless the method exploits some prior knowledge or calibration on the scene under observation. Moreover, if camera motion is not compensated, the motion of targets cannot be distinguished from the motion of the camera. Due to this and because of the difficulty of granting at the same time precise online camera self-calibration and real-time tracking, most of the solutions proposed for PTZ camera tracking that have appeared so far were limited to either unrealistic simplified contexts of application or restricted special domains. Several authors assumed scenarios with a single moving target [16, 17]. In [16], Hayman et al. used the affine transform applied to lines and points on a fixed scene background to adapt the PTZ camera focal length so to compensate the changes of size of targets. In [17], Tordoff et al. adjusted the PTZ camera focal length by considering the relative depths of new and old camera poses, estimated according to geometrical constraints.

Other methods were targeted to a specific domain of application [18, 19, 20, 21] or exploited context-specific fiducial markers to obtain an absolute reference and compute the time-varying relationship between the target positions in the 2D image and the positions in the 3D world plane [19, 20, 21]. In [18], hockey players were tracked in a PTZ camera sequence using a detector specialized for hockey players trained with Adaboost and particle filtering based on the detector’s confidence. The changes in scale of the targets was managed with simple heuristics using windows slightly larger/smaller than the current target size. This solution was improved in [19] by exploiting motion compensation of the PTZ camera. The authors used the a-priori known circular shape of the hockey rink and playfield lines to locate the reference points needed to estimate the world-to-image homography and compute camera motion compensation. Beyond the fact that these solutions are domain-specific, fiducial markers are likely to be occluded and impair the quality of tracking.

A PTZ camera tracking solution based on motion compensation was proposed in [22] by Kumar et al.. They introduced a layered representation in which spatial and temporal constraints on shape, motion, and layer appearance are modeled and jointly
estimated. However, adaptation to the changes of the size of the moving targets was simply modeled by allowing the variance to change according to the target shape. This approach is therefore likely to fail in the presence of abrupt scale changes.

2.1. Contributions

In this paper we present a novel solution that, under reasonable assumptions holding in many real contexts, allows continuous adaptive, real-time self calibration of a PTZ camera, so to permit tracking of multiple targets in wide areas with a single camera, even in the presence of changes of the scene appearance and for long periods of operation. We provide several contributions:

- We develop a single frame Bayes-optimal method to jointly estimate PTZ camera pose (including focal length) and scene landmark positions with no synchronization with the camera actuators. This differs from classical SLAM, as in [12], since landmarks only refer to the nearest keyframe instead of the full scene. This results in a principled approach which is robust to rapid and unpredictable camera motion and scales beyond thousands of landmarks.

- We provide an adaptive representation of the scene under observation. Landmarks are continuously updated in the scene map to account for the changes of the environment due to illumination changes and objects entering/exiting or changing position in the scene. This permits continuous camera calibration over hours of activity as discussed in [23].

- From the estimation of camera pose we can compute the relationship between target positions in the 3D world plane and positions in the 2D image and infer the expected imaged height of a target at any image location. This improves in both precision/recall performance of the detector as well as execution time by searching for targets exactly where they are expected to be found.

- Since camera motion is compensated, an absolute reference system can be used and tracking can be performed in the world plane, instead of the image plane. This allows real-time tracking with high and stable degree of accuracy also at large distances and zooming levels.

Experimental results are presented that validate the method in comparison with other solutions and demonstrate that it improves with respect to the state of the art in tracking with PTZ cameras.

Some of the ideas for calibration and tracking were also used under simplified assumptions in [24]. In this paper, differently from the solution subsequently described, targets were detected manually in the first frame of the sequence and the scene was assumed not to change through time. The method proposed was not robust to rapid camera motion and could not maintain camera calibration over hours of activity.
3. Camera Pose Estimation and Mapping

In the following, we introduce the scene model and define the variables used. Then we discuss the offline stage, where a scene map is obtained from the scene landmarks of the keyframes, and the runtime operation, where we perform continuous camera pose estimation and updating of the scene map. Target localization in 3D world coordinates and multiple target tracking are hence discussed in Sect. 4.

3.1. Scene model

We consider an operating scenario where a single PTZ camera is allowed to rotate and zoom around its nodal point, observing persons that move over a planar scene. The following entities are defined as random variables:

- The **camera pose** $c$. Camera pose is time varying, i.e. $c(t)$. Following [25, 8], we have defined the camera pose only in terms of the (time varying) pan and tilt angles ($\psi$ and $\phi$, respectively), and focal length $f$ of the camera. In fact, the principal point is a poorly conditioned parameter, and more precise calibration is obtained if it is assumed to be constant when pan, tilt and focal length are allowed to vary.

- The **scene landmarks** $u$. These landmarks account for salient points of the scene background and are initially detected in keyframe images in the offline stage using SURF [26]. Keyframes are sampled at fixed intervals of pan, tilt and focal length. The SURF descriptor is maintained associated to each landmark. During online camera operation, since the scene background will modify due to content or illumination variations, new landmarks will be added while others will be discarded. According to this, we should assume $u(t) = [x(t), y(t)]$, during runtime camera operation.

- The **view map** $m$ and **scene map** $M$. A view map is created for each keyframe that collects the scene landmarks (i.e. $m = \{u_i\}$). The scene map is obtained as the union of all the view maps and collects all the scene landmarks that have been detected in the entire scene at different pan, tilt and focal lengths values (i.e. $M = \{m_k\}$). Since the scene landmarks change through time, these maps will change accordingly. Due to this, we assume $m(t)$ and $M(t)$, during runtime camera operation.

- The **target state** $s$. The target state is represented in 3D world coordinates and includes both the position and speed of the target. It is assumed that targets move on a planar surface, i.e. $Z = 0$, so that $s = [X, Y, \dot{X}, \dot{Y}]$.

- The **landmark observations** $v$. These landmarks account for the salient points that are detected at the current frame. They can either belong to the scene background or to targets. The SURF descriptors of the landmark observations $v = [x, y]$ are matched with the descriptors associated to the scene landmarks $u$ registered in the scene map, in order to estimate the camera pose and derive the correct transformation between the current view and the 3D scene.
Figure 2: Target localization in world coordinates from a PTZ camera, main relationships and elements: the current frame and the landmark observations extracted; the view maps including the scene landmarks; the initial scene map obtained from the union of the view maps; the 3D scene.

- The target observations in the current frame, \( p \). This is a location in the current frame that is likely to correspond to the location of a target. At each time instant \( t \) there is a non-linear function \( g \) relating the position of the target in world coordinates to the location \( p = [x, y] \) of the target in the image. Its estimation depends on the camera pose \( c \) and the scene map \( M \) at time \( t \).

Camera localization and mapping requires inference of the joint probability of the camera pose \( c(t) \) and scene landmark locations in the map \( M(t) \), given the landmark observations \( v \) until time \( t \) and the initial scene map:

\[
p(c(t), M(t)|v(0 : t), M(0)).
\]

In order to make the problem scalable with respect to the number of landmarks, Eq. (1) is approximated by decoupling camera pose estimation from map updating:

\[
\frac{p(c(t)|v(t), M(t − 1)) p(M(t)|v(t), c(t), M(t − 1))}{p(c(t)|v(t), M(t − 1)) p(M(t)|v(t), c(t), M(t − 1))}
\]

We use this model to derive a relationship between the target position in the 2D image and its position in the 3D world plane. Fig. 2 provides an overview of the relationships between the main entities used to perform runtime target localization in the 3D world plane.

3.2. Scene Map Initialization

Scene map initialization is done in an offline stage. We perform a uniform sampling of pan and tilt angles and focal length and take a keyframe at each sample so to have
a complete representation of the scene under observation. We also register the coarse values of the pan, tilt and focal length, as provided by the camera actuators. For each keyframe we extract SURF keypoints \[26\], and create a view map \(m\) that collects all the scene landmarks detected in it, and the camera parameters estimated. The scene map \(M\) is hence obtained as the union of the view maps \(m\).

According to [8], we estimate the optimal values of the external camera parameter matrix \(R_k\) and the internal parameter matrix \(K_k\) at the \(k\)-th keyframe, by applying offline bundle adjustment to the sampled keyframes. Differently from [10], where bundle adjustment is performed online applied to a small subset of the frame sequence, this solution exploits the complete scene representation and avoids over-fitting camera parameters, that is a particularly critical phenomenon when the PTZ camera is used in large areas. Evidence of this fact is clearly visible in Fig.3.

Given a reference keyframe and the corresponding view map \(m_r\), the homography that maps each \(m_k\) to \(m_r\) can be estimated as in the usual way for planar mosaicing:

\[
H_{r,k} = K_r R_k R^{-1}_k K^{-1}_k
\]  

Figure 3: Use of bundle adjustment for the estimation of the camera focal length in a sample sequence of a PTZ camera monitoring a large area with right panning and progressive zooming-in. The focal length is estimated at the last frame of the sequence (evidenced with a small square box on the scene mosaic). Left: focal length estimated by online bundle adjustment taking 1 frame every 10 of the sequence (741.174 pixels). Right: focal length estimated by offline bundle adjustment (2097.5 pixels). The true focal length of the last frame of the sequence is 2085 pixels.

### 3.3. Continuous Homography Estimation

When the PTZ camera is in operation, at each time \(t\) we use the absolute pan tilt and zoom positional values provided by the camera actuators in order to retrieve the keyframe closest to the current view. These values do not have the required accuracy for the estimation of the camera pose. They are not synchronized with the video stream and non-repeatable controls, small changes in camera pose during operations or lack of stability of the lens system at high zoom typically affect the precision of the measures. Nevertheless, the view map \(m_{k,*}\) with the closest values of pan, tilt and focal length actuators, is likely to contain most of the scene landmarks that are also visible in the current view.

Following this step, the descriptors of landmark observations \(v\) detected in the current view are matched against the descriptors of scene landmarks \(u\) in \(m_{k,*}\), according to the distance ratio criterion of [27]. The homography \(H(t)\) from the current view to \(m_{k,*}(t)\) is hence estimated with RANSAC. The homography \(H_{r}(t)\) between the current
and the reference view is calculated as:

\[ H_r(t) = H_{rk}, \cdot H(t). \]  \hspace{1cm} (4)

It includes the information on camera pose with respect to the reference view.

3.4. Scene Map Updating

Changes of the visual environment due to illumination or objects entering, leaving or changing position in the scene, will modify the original map as time progresses. They will determine drifting of camera pose estimation and consequently will affect tracking performance. However, while some of these elements will permanently change the scene content, others will only determine temporary changes. In order to have a continuous updated map of the scene content and not account for temporary elements, we introduce a landmark birth-death process, and perform a recursive single frame Bayes-optimal estimation of landmark locations at each time instant.

![Figure 4: Landmark birth process. Of the new landmarks (yellow circled) only that within the dashed boundary (red bordered image) is added to the scene map (yellow bordered image). Probability of being added to the map is evaluated as the ratio between the solid and dashed bounding boxes.](image)

**Landmark birth-death process**

In our approach, moving objects are considered distractors for the original scene map and are discarded as outliers in the computation of \( H(t) \). However, such landmarks are considered as feasible candidates modeling for novel scene structure and are therefore included into \( m_k, \) as new landmarks, and their locations initialized with \( H(t) \) with respect to the keyframe reference coordinate system (see Fig. 4). For each candidate landmark, the probability of being added to the map is evaluated as the ratio between the area of the enclosing bounding box of matched landmarks and the area of the extended bounding box considering the candidate landmark. Landmark termination is achieved by counting the number of times that a landmark that is potentially visible from the current view is not matched.

This procedure grounds on the fact that the transformation between two near frames undergoing pan tilt zoom motion can be locally approximated by a similarity transformation. Under such conditions the asymptotic stability of the procedure is still guaranteed by the Multiplicative Ergodic Theorem, as demonstrated for the general case.
in [23], so that no drifting in the homography estimation is assured. In particular, drifting is not likely to occur if dynamic initialization of the new scene landmarks is made by extending the rectangular region containing the landmarks already mapped [23]. Some empirical evidence of this fact is provided in Sec. 5.6.

**Landmark uncertainty modeling and updating**

A precise localization of landmarks is obtained by applying the Extended Kalman filtering to the observation model and considering all the possible sources of error that might affect landmark observations.

According to the process described above, at each time \( t \) only the view map \( m_{k,\star} \) is updated. Therefore, the map updating factor in Eq. (2) can be rewritten as:

\[
p(m_{k,\star}(t)|v(t), c(t), m_{k,\star}(t-1))
\]

By applying Bayes theorem to Eq. (5), and assuming that landmark observations \( v \) that match the scene landmarks in \( m_{k,\star}(t) \) are independent of each other, given the scene landmark locations and camera pose, i.e.:

\[
p(v(t)|c(t), m_{k,\star}(t)) = \prod_i p(v_i(t)|c(t), u_i(t)),
\]

it results:

\[
p(m_{k,\star}(t)|v(t), c(t), m_{k,\star}(t-1)) = \prod_i p(v_i(t)|c(t), u_i(t))p(u_i(t)|u_i(t-1))
\]

where \( p(u_i(t)|u_i(t-1)) \) is the prior pdf of the \( i \)-th scene landmark at time \( t \) given its state at time \( t - 1 \).

Under the assumptions that: scene landmarks \( u_i(t) \) have a Gaussian pdf, the Direct Linear Transform (DLT) is used to compute \( H(t) \) and landmark localizations error have a Gaussian distribution, the observation model \( p(v_i(t)|c(t), u_i(t)) \) corresponds to a linear measurement function of the form:

\[
v_i(t) = H_i(t)u_i(t) + \lambda_i(t)
\]

where \( H_i(t) \) is the \( 2 \times 2 \) matrix obtained by linearizing the homography \( H(t) \) at the matched landmark observation \( v_i(t) \) and \( \lambda_i(t) \) is an additive Gaussian noise term with covariance \( \Lambda_i(t) \).

The covariance of the observation model \( \Lambda_i(t) \), can be defined to include all the sources of error that may affect landmark observations, namely: the landmark transfer error (arising from landmark spatial distribution and the DLT method), the landmark uncertainty in the map and the keypoint detection error (originated by the detector). In homogeneous coordinates, \( \Lambda_i(t) \) can be expressed as:

\[
\Lambda_i(t) = B_i(t) \Sigma_i(t) B_i(t)^T + H(t)^{-1} P_i(t) H(t)^{-\top} + \Lambda'_i,
\]

where:
• $\Sigma_i(t)$ is the $9 \times 9$ homography covariance matrix that has closed-form expression according to [29] and $B_i(t)$ is the $3 \times 9$ block matrix of landmark observations (in homogeneous coordinates).

• $P_i(t)$ is the covariance (in homogeneous coordinates) of the estimated landmark position on the nearest view map through $h(t)$.

• $\Lambda'_i$ is the keypoint detection error.

The Bayes optimal updating of Eq. (7) can be obtained in closed form through multiple applications of the Extended Kalman Filter to each landmark.

The effect of the uncertainty of the homography $H_i(t)$ in Eq. (9) for the $i$-th landmark position estimation is propagated through the Kalman gain, computed as:

$$K_i(t) = P_i(t|t-1)H_i(t)^{-1} \left[ H_i(t)^{-1}P_i(t|t-1)H_i(t)^{-\top} + \Lambda'_i(t) \right]^{-1}$$.  

(10)

Figure 5: The transformation from the 2D mosaic plane (Left) to the 3D world plane (Right). The vanishing points and the vanishing lines are used for the computation of matrix $H_p$. A pair of corresponding points in the mosaic and world plane is shown.

4. Application to Multiple Target Tracking

In this section, we demonstrate how camera pose estimation can be exploited to perform effective target detection and tracking in 3D world coordinates with PTZ cameras.

The reference plane of the mosaic (i.e. the image plane of the reference keyframe) is related to the 3D world plane according to a stationary homography:

$$H_W = H_sH_p$$,

(11)

where $H_p$ is the rectifying homography obtained by exploiting the single view geometry between the planar mosaic and the scene plane$^1$ and $H_s$ is a transformation from pixels in the mosaic plane to 3D world coordinates. The transformation $H_p$ is obtained from the projections of the vanishing points$^3$. The transformation $H_s$ is estimated from

$^1$In the case of a PTZ sensor, the homography between each keyframe and the reference keyframe is the infinite homography $H_\infty$ that puts in relation vanishing lines and vanishing points between the images.
the projection of two points at a known distance $L$ in the world plane onto two points in the mosaic plane (Fig. 5).

The function $g$ mapping the position of a generic target in the world plane onto its position $p$ in the current frame can be represented through the time varying homography matrix $G(t)$, in homogeneous coordinates, (see Fig. 5):

$$G(t) = \left( H_W H_r(t) \right)^{-1} = \left( H_s H_r \cdot H(t) \right)^{-1}. \quad (12)$$

4.1. Context-based Target Detection

Camera pose $c(t)$ and the homography $G(t)$ calculated at each time instant can be exploited to perform efficient and effective detection of moving targets under reasonable assumptions.

**Image Slicing using Geometric Constraints**

Assuming that targets remain nearly vertical in the scene, the position $h$ of the head of the target can be estimated from the feet position $p$ according to the homology relationship:

$$h = \mathcal{W}p \quad (13)$$

$\mathcal{W}$ being defined as:

$$\mathcal{W} = I + (\mu - 1) \frac{v_\infty \cdot l_\infty^T}{v_\infty^T \cdot l_\infty^T}, \quad (14)$$

where $I$ represents the identity matrix, $l_\infty$ is the world plane vanishing line, $v_\infty$ is the vanishing point of the world normal plane direction, and $\mu$ is the cross-ratio. The vanishing point $v_\infty$ is computed as $v_\infty = KK^T \cdot l_\infty$, with $l_\infty = G \cdot [0, 0, 1]^T$ and $K$ can be derived from $H(t)$ as in [24] (the dependency on $t$ has been omitted for the sake of clarity in all the expressions above).

Using this information, for each frame, we consider horizontal slices at different vertical positions, such that their height is calculated according to Eq. (13) ±10\% (Fig. 6). The HoG template of [3] is hence applied to each slice appropriately rescaled. Variations of scale of ±10\% have insignificant influence on the recall of the detector [31] as shown in Fig. 7(a). This solution allows detection of targets at a single scale with constant rate for each slice, so resulting in computational savings.
Figure 7: (a) Recall performance with image re-scaling using the fixed scale HOG template of [3] on the MICC UNIFI PTZ dataset, with SVM confidence values of 0 and -1. Variations of ±10% in scale don’t affect the recall performance substantially. (b) Target detections using probabilities of Eq. (15): blue regions indicate zones where scene background is highly probable and red regions where targets are more likely to be found.

Use of Contextual Information

Since in our case background regions are clearly identified by the presence of matches between scene landmarks and landmark observations, we have exploited this fact and used the probability of false target presence in a window $W$ to discriminate between targets and background regions with human-like patterns:

$$p(\text{target} = \text{false} \mid W) = \frac{1}{W} \sum_{i=1}^{m} K\left(\frac{||w - v_i||}{\sigma_{v_i}}\right) dw$$  \hspace{1cm} (15)

where $W$ is the target detection window, $w = [x, y] \in W$ are the locations in the detection window, $v_i$ are the landmark observations matched in $W$, $\sigma_{v_i}$ is the scale of $v_i$, and $K(\cdot)$ is a Gaussian kernel.

This improves the precision of the detector and reduces the presence of false positives with respect to previous approaches [3] where it is assumed that false positive detections are distributed uniformly in the image [32]. Fig. 7(b) shows an example of detection using Eq. (15).

4.2. Multiple Target Tracking in World Coordinates

The relationship of Eq. (12) permits target detections in the image plane be in correspondence with their real positions in the world plane at each time instant $t$. Similarly, the target predictions in the world plane can be put in correspondence with the target observations in the image. Tracking can therefore be performed in an absolute reference coordinate system in world coordinates with precise separation of the motion of the targets. In order to perform multiple target tracking we follow a two-stage association process using both appearance and motion information.

Greedy Update of Target Track Templates

Active target tracks in the world plane are represented with the color spatiogram of the template of the last target detected. New detections update the track representations according to a greedy, threshold-based approach, using the likelihood function:

$$\gamma_{ij} = \gamma_{ij}^a \cdot \gamma_{ij}^m,$$  \hspace{1cm} (16)
where for each target detected \( i \) and active track \( j \) are considered both the Mahalanobis-Bhattacharyya distance \( \gamma_{ij}^a \) between the color spatiograms of the target detected and the active track, and the Mahalanobis distance \( \gamma_{ij}^m \) between the position of the target detected and the predicted position for the track calculated in the image plane. As a new detection is associated to an active track, the track template is updated with the template of the detected target. If a new detection is not associated to any active target track, a new track is initiated. If an existing active track has not been linked with any new detections, it is terminated.

**Soft Association of Observations to Target Tracks**

For each active target track we use the Extended Kalman Filter to estimate the new position of the target in the world plane. The observation model for each target is defined as:

\[
p(t) = g(s(t)) + \zeta(t),
\]

where \( s \) is the target state, represented in 3D world coordinates, and \( g : \mathbb{R}^4 \rightarrow \mathbb{R}^2 \) is a measurement function from the world space to the image space defined as:

\[
g(s(t)) = \begin{bmatrix} G(t) & O_{2 \times 2} \end{bmatrix} s(t),
\]

with \( G(t) \) being the linearization of Eq. (12) and \( O_{2 \times 2} \) the \( 2 \times 2 \) zero matrix. \( \zeta(t) \) is a Gaussian noise term with zero mean and diagonal covariance that models the target localization error in the current frame.

Assuming constant velocity, the motion model in the 3D world plane is defined as:

\[
p(s(t)|s(t-1)) = N(s(t); As(t-1), Q),
\]

where \( A \) is the \( 4 \times 4 \) constant velocity transition matrix and \( Q \) is the \( 4 \times 4 \) process noise matrix. The predicted target positions in the image plane are directly obtained from the target positions predicted in the world plane according to the transformation of Eq. (12).

In order to achieve better discrimination between targets that are very close each other, \( n \) points are randomly sampled in the neighbourhood of the predicted position and the rectangular template of height given by Eq. (13) is extracted at each sample. The three templates with the most similar color spatiograms are used to calculate the probability of association between the \( k \)-th observation and the \( j \)-th active track. Cheap-JPDAF [34] is used in order to provide an efficient and effective data association:

\[
\beta_{kj} = \frac{\gamma_{kj}}{\sum_k \gamma_{kj} + \sum_j \gamma_{kj} - \gamma_{kj} + \kappa},
\]

where \( k \) is the index of the selected samples, \( \kappa \) is a parameter that models the probability that a target observation is generated by some spurious element and \( \gamma_{kj} \) is computed as in Eq. (16). These probabilities are hence used as weights to compute the innovation \( \nu_j \) of each active track and update the covariance. Innovation is calculated as the weighted combination of the innovations \( \nu_{kj} \):

\[
\nu_j = \sum_k \beta_{kj} \nu_{kj}.
\]
Updating of the target covariance $P_j$ is obtained as:

$$P_j(t|t) = (1 - \sum_k \beta_{kj})P_j(t|t - 1) + (\sum_k \beta_{kj})\bar{P}_j(t) + \tilde{P}_j(t)$$ (22)

where the components respectively account for the uncertainty $P_j(t|t - 1)$ derived from the association between the predicted target position and the sampled location, the uncertainty $\bar{P}_j(t)$ propagated by Kalman Filter state update and the uncertainty $\tilde{P}_j(t)$ that models erroneous associations.

5. Experimental results

The method described above permits continuous real-time self calibration of a PTZ camera and effective real-time tracking in world coordinates of multiple targets with no fiducial markers. It supports very high and constant precision of target localization even at large distances and any zooming level and operates under assumptions that are verified in most real world contexts. In the following, we present a comparative analysis showing that the state of the art tracking methods do not offer solutions for PTZ cameras with similar characteristics and performance.

A comparative analysis of tracking solutions for PTZ cameras is complicated by several facts. On one hand, the UBC Hockey sequence [18] is the only publicly available dataset recorded from a PTZ camera. It is very short and includes frames of a hockey game. So longer and more complex PTZ camera sequences are needed in order to provide meaningful performance assessments. According to this, we have created a new dataset, the MICC UNIFI PTZ dataset [35] including longer PTZ sequences, with several different critical conditions and calibration data associated to each frame. On the other hand, among the few methods that have reported tracking performance figures on the UBC Hockey sequence, the method of [18] uses context-specific fiducial markers and the methods in [36] and [37] do not make their code publicly available. So they cannot be assessed on other datasets. Tracking methods that are claimed to have general application can eventually be applied to PTZ camera sequences. Among the most recent and best performing methods we could select only three of them, namely [38], [39] and [40], as their authors were available to a comparative verification.

Considering these facts, we have assessed our method on both the UBC Hockey sequence, and the MICC UNIFI PTZ dataset. On the UBC Hockey sequence we have compared the performance of our method against the performance reported by [18], [36] and [37]. We also ran the author implementations of [38] and [39] and reported the performance measured. On the MICC UNIFI PTZ dataset we compared our method against [40], [38] and [39].

5.1. Sequences tested

The UBC Hockey sequence includes 101 frames of a hockey game. Targets have small size and move erratically and have frequent occlusions. All the targets remain in
the scene during the sequence. The scene is observed from a far distance with a large initial tilt angle with respect to the ground plane. The PTZ camera is steered to follow the scene while continuously zooming-in with little pan and tilt.

The MICC UNIFI PTZ dataset contains four PTZ sequences with a total of 3,662 frames and 9,685 annotations of labeled targets. In more detail:

- Seq.#1 “Long” was taken outdoor using a Sony SNC-RZ30 camera. The sequence has low target density, but presents tracks of length up to 70 meters. The PTZ camera is steered to follow targets while continuously zooming-in;

- Seq.#2 “Focus” was recorded outdoor from a Sony SNC-RZ30 camera. This sequence has medium target density and covers approximately the same distance as Seq.#1. The camera zooms on a target for a few frames, then it returns to the group of targets. Sunlight results in strong variations in target appearance;

- Seq.#3 “Dense” was taken outdoor from a Sony SNC-RZ30 camera. This sequence has the highest density of targets. Targets have small size and move erratically, entering and leaving the field of view, and have frequent occlusions. They have similar appearance in most cases;

- Seq.#4 “Rapid Motion” was recorded indoor from an Axis Q6032-E camera. Many targets move simultaneously in this scene while the camera performs fast patrolling and zoom-in operations. Targets partially occlude each other frequently and also leave the field of view for a period before re-entering.

For each sequence, the map of the area under observation was obtained from keyframes taken at intervals of 10 and 20 degrees of tilt and pan angles respectively, and at different levels of zooming. Settings and viewing conditions for each sequence are reported in Table 1.

Table 1: Overview of the MICC UNIFI PTZ dataset.

| MICC UNIFI PTZ | #Frames | Resolution | #Keyframes | #Zoom Levels | Density | Place/Size | Scene Texture | Illumination |
|---------------|---------|------------|------------|-------------|---------|------------|---------------|--------------|
| Seq.#1 “Long” | 782     | 128 x 240  | 269        | 4           | Low     | Outdoor/Wide | Weak          | Natural      |
| Seq.#2 “Focus”| 530     | 368 x 276  | 144        | 3           | Medium  | Outdoor/Wide | Weak Highlights|
| Seq.#3 “Dense”| 1750    | 320 x 240  | 190        | 4           | High    | Outdoor/Wide | Weak          | Natural      |
| Seq.#4 “Rapid Motion” | 600     | 320 x 240  | 140        | 1           | High    | Indoor/Small | Strong        | Artificial   |

5.2. Characteristics of the methods compared

The authors in [18] use a particle filter to perform tracking in the image plane and a specialized detector trained with Adaboost to detect the hockey players in the sequence.

The methods in [36] and [37] are general tracking methods in the image plane that have reported performance figures on the UBC Hockey sequence. Both methods perform target tracking with a particle filter based on detector confidence. The method reported in [36] has been used with the detectors of [3] and [41]. In [37], the authors used Felzenszwalb’s part-based detector [2]. Tracking requires learning of a weighting parameter from part of the sequence, and complex hierarchical data association is applied to track multiple targets.
The method by Yang et al. [40] uses the detector of [42] and learns a discriminative part-based appearance model for each target that is continuously updated. Tracking is performed in the image plane by creating tracklets from the association of the detector responses and combining tracklets according to MHT [43].

In the method by Pirsiivash et al. [38], detections are performed according to [2]. Target tracking in the image plane is performed by computing the shortest path in a path graph. A greedy algorithm that performs non-maximum suppression of the detector responses is applied to boost tracking performance.

The method by Choi et al. [39] performs multiple target tracking in the world plane with a moving sensor. Target detections are extracted using [2] and target tracking is performed using the MCMC algorithm [44]. The camera pose estimation only accounts for slight variations of the focal length from frame to frame. This method requires that camera pose with respect to the ground plane is available for the first frame. Then camera calibration in the following frames is obtained by sequentially tracking the features with KLT [45]. Camera parameters are hence used to obtain the transformation that relates the target position in the 2D image to its corresponding location on the ground plane.

5.3. Parameter Settings and Metrics

Tests were performed under the following settings:

Camera Pose Estimation and Map Updating Parameters

The interest points were detected and represented using SURF. The RANSAC threshold was set to 3 pixels. The new landmarks observed were tracked for 20 frames and then added to the map. The scene landmark lifetime with no inlier matching was set to 40 frames.

Detection Parameters

Detection is performed using a HoG-based person detector [3]. The detector was forced to optimize recall. A detection is signaled if the confidence is higher than $-1.0$. New target tracks are initialized for $p(\text{target} = \text{false} \mid W) < 0.05$.

Tracking Parameters

For the outdoor sequences of the MICC-UNIFI PTZ dataset we set the threshold of Mahalanobis-Bhattacharyya distance between the color spatiograms to 0.65 and used a 8-bin spatiogram quantization. For the indoor sequence we used a 0.6 threshold and 16 bin. 100 samples were taken in the neighbourhood of the predicted target position. The lifetime of a track before being terminated was set to 80 frames. Detections are assigned to tracks if the probability of association of Eq. (20) is higher than 0.7.

Performance metrics

The accuracy for target detection was evaluated using Recall/FPPI curves and the precision was estimated according to the Multi-Object Detection Precision (MODP) metric, as the average VOC score (the intersection over the union of bounding boxes of ground truth and detection) over all the true positives.
The performance for multiple target tracking was evaluated according to the CLEAR MOT metrics \cite{46}. The MOTA accuracy index is calculated as:

\[
\text{MOTA} = 1 - \frac{\sum_t (\text{TFN}_t + \text{TFP}_t + \text{ID}_{SW}t)}{\sum_t n_t}
\]  

(23)

where \(\text{TFN}_t\) is the number of tracking false negatives, \(\text{TFP}_t\) is the number of tracking false positives, \(\text{ID}_{SW}t\) is the number of identity switches and \(n_t\) represents the true number of targets, in the frame at time \(t\). The precision index MOTP is defined as:

\[
\text{MOTP} = \frac{\sum_{i,t} \text{VOC}_{i,t}}{\sum_t \text{TTP}_t}
\]

(24)

where \(\text{VOC}_{i,t}\) is the VOC score calculated for the \(i\)-th target and \(\text{TTP}_t\) is the number of tracking true positives, at time \(t\).

| Seq. | Res. | SVM+HoG | LSVM+HoG | Our Detector |
|------|------|---------|----------|--------------|
| Seq.#1 | Low  | 61.73   | 63.63    | 67.35        |
|      | High | 69.09   | 65.86    | 69.19        |
| Seq.#2 | Low  | 66.89   | 62.24    | 69.18        |
|      | High | 64.40   | 64.04    | 65.68        |
| Seq.#3 | Low  | 68.19   | 73.87    | 74.95        |
|      | High | 65.50   | 74.54    | 82.17        |
| Seq.#4 | Low  | 71.26   | 76.39    | 79.68        |
|      | High | 71.85   | 75.95    | 73.58        |

Table 2: MODP% of people detection methods evaluated over the MICC UNIFI PTZ dataset at low and high resolution for each sequence.
5.4. Target Detection

Fig. 8 displays examples of target detection on sample frames with our method as expounded in Sect. 4.1, in comparison with detections obtained with the direct application of the Dalal-Triggs’ detector (SVM+HoG) and Felzenszwalb’s part-based detector (LSVM+HoG).

Fig. 9 compares the Recall/FPPI plots evaluated at the original low resolution (top row) and at two-times magnified resolution (bottom row). The average MODP scores are shown in Table 2.

It is possible to notice the improvement of detection performance due to the exploitation of camera pose information. The only exception is in Seq.#4 where a target has several partial occlusions. In this case, a better performance is obtained with the LSVM-HoG detector that permits to detect body-parts.

5.5. Multiple-Target Tracking

MICC-UNIFI PTZ Dataset

Figs. 10, 11, 12, 13 show the target trajectories obtained from our tracking for the four sequences of the MICC UNIFI PTZ dataset. In Table 3 we report the CLEAR MOT metrics computed on these sequences in comparison with the methods in [40, 38, 39]. For a more complete investigation we also report the tracking false positives (TFP%) and false negatives (TFN%) rates, the number of identity switches due to exchange of target identities (ID_SW) and the number of trajectory fragmentations (TR_FR) (typically when a target exits the camera field of view for several frames either due to camera zooming in or random movements of a target with respect to the camera).
(a) Qualitative results on Seq.#1 “Long”.

(b) Qualitative results on Seq.#2 “Focus”.

Figure 10: Tracking with our method on Seq.#1 (Top) and Seq.#2 (Bottom) MICC UNIFI PTZ dataset. (Left) Sample frames with tracked targets. (Right) Target trajectories.

Figure 11: Tracking with our method on Seq.#3 MICC UNIFI PTZ dataset (frames from 1 to 900). (Top) Sample frames with tracked targets. (Bottom) Target trajectories. In this sequence, targets move quite randomly with frequent occlusions between targets and make sudden turns and changes of directions.

Figure 12: Tracking with our method on Seq.#4 MICC UNIFI PTZ dataset. (Left) sample frames with tracked targets. (Right) Target trajectories. In this sequence, targets move quite randomly with frequent occlusions between targets.
| Seq.#1 | Our approach | MOTA% | MOTP% | TFN% | TFP% | ID_SW | TR_FR |
|--------|---------------|-------|-------|------|------|-------|-------|
|        | Pirsiavash [38] | 84.7  | 66.7  | 5.42 | 0.92 | 0     | 21    |
|        | Choi [39]    | 56.0  | 63.4  | 41.1 | 0.94 | 1     | 46    |
|        | Yang [40]    | 66.7  | 58.2  | 7.4  | 25.2 | 0     | 18    |

| Seq.#2 | Our approach | MOTA% | MOTP% | TFN% | TFP% | ID_SW | TR_FR |
|--------|---------------|-------|-------|------|------|-------|-------|
|        | Pirsiavash [38] | 74.8  | 67.3  | 22.1 | 0.85 | 0     | 33    |
|        | Choi [39]    | 50.5  | 60.0  | 47.1 | 0.69 | 1     | 31    |
|        | Yang [40]    | 74.7  | 54.4  | 3.5  | 20.7 | 2     | 18    |

| Seq.#3 | Our approach | MOTA% | MOTP% | TFN% | TFP% | ID_SW | TR_FR |
|--------|---------------|-------|-------|------|------|-------|-------|
|        | Pirsiavash [38] | 63.3  | 66.8  | 34.6 | 0.42 | 0     | 55    |
|        | Choi [39]    | 21.6  | 65.4  | 75.6 | 1.34 | 2     | 51    |
|        | Yang [40]    | 51.0  | 65.6  | 4.52 | 41.4 | 2     | 122   |

| Seq.#4 | Our approach | MOTA% | MOTP% | TFN% | TFP% | ID_SW | TR_FR |
|--------|---------------|-------|-------|------|------|-------|-------|
|        | Pirsiavash [38] | 51.1  | 74.1  | 48.2 | 0.63 | 0     | 20    |
|        | Choi [39]    | 48.2  | 71.3  | 48.1 | 1.92 | 0     | 32    |
|        | Yang [40]    | 45.6  | 66.7  | 50.1 | 2.85 | 1     | 24    |

The experiments show that our method has the highest accuracy than the other methods with all the sequences. It scores very good accuracy (MOTA) on Seq.#1 and Seq.#2 and has good accuracy also in the more complex scenes of Seq.#3 and Seq.#4.

Our method has the highest precision (MOTP) on Seq.#3 and almost the same precision as [38] on Seq.#1 and Seq.#2. In Seq.#4 The higher precisions of [38], and [39] should be ascribed to the use of the Felzenszwalb’s detector [2]. In fact, in this sequence, the camera movements determine frequent partial occlusions of the targets that cause misses with the full body HoG detection of our method.

Our method scores the lowest trajectory fragmentation score, showing that it can support continuous tracking much better than the others. Tracking in world coordinates permits recovery from critical cases such as when a target exits the field of view for a few frames with no identity switches (see f.e. frames #479 and #498 of Seq.#2 in Fig. 10(b)). Trajectory fragmentations occur only in those cases where targets exit the field of view for long periods as in Seq.#3. In these cases they originate a new track when they reenter in the scene.

With our method, tracking false negatives occur when targets have small size (see f.e. frame #475 in Seq.#3 in Fig. 11) or are very close to the camera (see f.e. frame #34 of Fig. 12). The TFN score is much lower than the methods by [38, 39] in all the sequences. Lower TFN rates of [40] in Seq.#2 and in Seq.#3 are counterbalanced by very high TFP rates in the same sequences. Tracking false positives are likely to occur when the tracker drifts on the background or when a target exits the field of view for a few frames (see frame #254 of Seq.#4 in Fig. 12 f.e.). The methods of [38, 39] have a lower TFP rate only apparently. Indeed in these methods, differently from our case, when the detector misses the target for several frames, the tracker does not try to recover the track, but terminates the current track and initiates a new one. As a
result, an unreasonably high number of trajectory fragmentations is generated. All the methods show a very low number of identity switches.

**UBC Hockey Sequence**

The performance figures obtained with the *UBC Hockey* sequence are reported in Table 4, in comparison with the other methods.

We used the detector of Okuma et al. [18] in our method to detect targets. The camera calibration data, needed to build the scene map $\mathbf{M}(0)$, were taken directly from the sequence, sampling a keyframe every ten frames. It is apparent that our method largely outperforms the methods of [36] and [18] in both precision and accuracy. A lower precision is instead observed with respect to [37]. This is determined by a special unique situation (see frame #38 of Fig. 13), where a target track remains stuck to the scene background for five frames, so generating several false positives. The method of [39] did not provide significant results mainly because that method requires that the scene horizon is in the initial image of the sequence, which is a condition that is not satisfied in the *UBC Hockey* sequence.

![Figure 13: Tracking with our method on the UBC Hockey sequence (Left): sample frames with tracked targets. (Right) Target trajectories.](image)

| Method          | MOTA% | MOTP% | TFN% | TFP% | IDSW |
|-----------------|-------|-------|------|------|------|
| Our approach    | 91.6  | 61.3  | 6.3  | 2.0  | 1    |
| Breitenstein [36]| 76.5  | 57.0  | 22.3 | 1.2  | 0    |
| Okuma [13]      | 67.8  | 51.0  | 31.3 | 0.0  | 11   |
| Yan [37]        | 91.7  | 71.6  | **1.76** | 6.49 | 0    |
| Pirsiavash [38] | 16.4  | 74.4  | 82.9 | 0.19 | 5    |
| Choi [39]       | Tracks targets occasionally | | | | |
| Yang [40]       | *     |     |     |     |      |

Table 4: Tracking performance on UBC Hockey dataset. *Code not made available by authors.

### 5.6. Long term experiments

The effectiveness of the mechanism of map updating described in Sec. 3.4 with regard to the estimation of the homography $\mathbf{H}(t)$ is shown in Fig. 14. Fig. 14(a) shows the birth-death of keypoints for one (randomly chosen) of the keyframes of the scene.
map over 20 minutes of observation (12000 frames sampled at 10 frames per second). Scene landmarks with ID $\in [0..2000]$ are those observed in the phase of creation of the scene map. Landmarks with ID $\geq 2000$ are those observed during the time of operation. Lifetime of each landmark is reported in the $y$–axis. It can be observed that only a few of the original scene landmarks have survived at the end of the observation period, and new landmarks that are observed progressively replace the original ones in the map. Fig. 14(b) shows the average variance of keypoint localization using the estimated homography, computed over a sample keypoint (ID 3201 randomly chosen among those that survive until the end of the observation period). It can be observed that the new landmarks are sufficiently stable to guarantee a correct estimation of the homography and subpixel accuracy in keypoint localization.

![Figure 14: (a) Lifetime of landmarks observed in a sample keyframe (a region of the scene under observation). (b) Homography estimation accuracy measured as variance in the estimated position for a sample keypoint (ID 3201).](image)

In Fig. 15 we show a few frames of real world PTZ operation in a parking lot over 10 hours. As can be observed, both illumination and scene conditions have significant changes through time. Despite of the presence of moving objects and scene variations, the solution proposed estimates the camera pose at each instant and estimates the 3D world plane correctly. Fig. 15 top shows the current camera view, with the 3D ground plane estimated, superimposed over the image. The scene landmarks and the estimated homography are shown in Fig. 15 middle together with the effects of scene map updating. Fig. 15 bottom shows the corresponding original views recovered from the values of the camera actuators.

5.7. Operational Constraints and Computational requirements

The solutions in [38], [40] and [37], although having good performance figures in terms of accuracy and precision, nevertheless all require that target detections be available beforehand. So they cannot be employed for real-time tracking with PTZ cameras in real applications. In their experiments on the UBC Hockey sequence, the authors of [35] have reported a variable performance between 0.4 and 2 fps on a Dual-core@2.13GHz. On the same dataset, the method in [18] has reported 1fps on a Dual-core@2.66 GHz. The authors of [39] have reported that their method is capable to operate at 5 fps, not accounting for detection. Therefore, we can presume that the total rate of the complete system is less than 2 fps, that is too slow for effective tracking.

Our method has been verified to perform real-time multiple target tracking in sequences from PTZ cameras at 12 fps on Intel Xeon Dual Quad-Core at 2.8GHz and 4GB of memory, with no GPU processing. The current implementation of the method...
exploits multiple cores and was developed in C/C++. Frame grabbing, camera calibration and context analysis are calculated in one thread and sent to the other threads where detection and tracking are performed. Extraction of target measurements was implemented using the Intel Threading Building Block library.

6. Conclusions

In this paper, we have presented an effective solution for real-time multiple target tracking from a single PTZ camera observing a planar scene. The solution integrates a complex unified framework for on-line camera calibration, context-based target detection, tracking in world coordinates and multi-stage data association. It maintains a continuous relationship between the target observations in the image plane and the corresponding positions in the world plane, as estimated with online continuous calibration of the camera during operation. This permits improvement of the performance of the detector and allows more effective tracking of multiple targets in world coordinates with camera motion compensation. As a result, the solution proposed allows tracking of multiple targets with PTZ cameras in real-time with high and stable degree of accuracy, also at large distances and any zooming level. It achieves the state of the art performance of tracking with these cameras.

References

[1] Z. Wu, R. Radke, Keeping a pan-tilt-zoom camera calibrated, IEEE Transactions on Pattern Analysis and Machine Intelligence 99 (2012) 1.

[2] P. Felzenszwalb, R. Girshick, D. McAllester, Cascade object detection with deformable part models, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010.
[3] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005.

[4] P. Dollár, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: A benchmark, in: CVPR, 2009.

[5] R. Hartley, Self-calibration from multiple views with a rotating camera, In Proceedings European Conference on Computer Vision.

[6] L. de Agapito, E. Hayman, I. D. Reid., Self-calibration of rotating and zooming cameras., International Journal of Computer Vision 45 (2).

[7] H.-Y. Shum, R. Szeliski, Panoramic image mosaics, Tech. rep. (1997).

[8] M. Brown, D. Lowe, Recognising panoramas, in: Proc. of the International Conference on Computer Vision, 2003.

[9] S. Sinha, M. Pollefeys., Towards calibrating a pan-tilt-zoom cameras network, P. Sturm, T. Svoboda, and S. Teller, editors, OMNIVIS.

[10] G. Klein, D. Murray, Parallel tracking and mapping for small AR workspaces, in: Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007.

[11] B. Williams, G. Klein, I. Reid, Real-time SLAM relocalisation, in: Proceedings of the IEEE International Conference on Computer Vision, 2007.

[12] J. Civera, A. J. Davison, J. A. Magallon, J. M. M. Montiel, Drift-free real-time sequential mosaicing, International Journal of Computer Vision 81 (2) (2009) 128–137.

[13] S. Lovegrove, A. J. Davison, Real-time spherical mosaicing using whole image alignment, in: Proceedings of European Conference on Computer Vision, 2010.

[14] R. Y. Tsai, R. K. Lenz, A new technique for fully autonomous and efficient 3d robotics hand-eye calibration, in: Proceedings of the 4th international symposium on Robotics Research, 1988, pp. 287–297.

[15] D. Song, K. Goldberg, A minimum variance calibration algorithm for pan-tilt robotic cameras in natural environments, in: in IEEE International Conference on Robotics and Automation, 2006.

[16] E. Hayman, T. Thorhallsson, D. W. Murray., Zoom-invariant tracking using points and lines in affine views - an application of the affine multifocal tensors, in: Proc. of the International Conference on Computer Vision, 1999.

[17] B. Tordoff, D. Murray, Reactive control of zoom while fixating using perspective and affine cameras, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (1) (2004) 98–112.
[18] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little, D. G. Lowe, A boosted particle filter: Multitarget detection and tracking., in: Proceedings of the European Conference on Computer Vision, 2004.

[19] N. d. F. Yizheng Cai, J. Little., Robust visual tracking for multiple targets., in: Proceedings of the European Conference on Computer Vision, 2006.

[20] Y. Seo, S. Choi, H. Kim, K.-S. Hong, Where are the ball and players? soccer game analysis with color based tracking and image mosaic, in: Proc. of the International Conference on Image Analysis and Processing, 1997.

[21] L. Barceló, X. Binefa, J. R. Kender, Robust methods and representations for soccer player tracking and collision resolution, in: Proceedings of the International Conference on Image and Video Retrieval, 2005.

[22] H. Tao, H. S. Sawhney, R. Kumar, Object tracking with bayesian estimation of dynamic layer representations, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (1) (2002) 75–89.

[23] A. Del Bimbo, G. Lisanti, I. Masi, F. Pernici, Device-tagged feature-based localization and mapping of wide areas with a ptz camera, in: Proc. of CVPRW on Socially Intelligent Surveillance and Monitoring, 2010.

[24] A. Del Bimbo, G. Lisanti, F. Pernici, Scale invariant 3D multi-person tracking using a base set of bundle adjusted visual landmarks, in: Proceedings of ICCV International Workshop on Visual Surveillance, 2009.

[25] E. H. L. De Agapito, I. Reid, Self-calibration of rotating and zooming cameras, no. OUEL 0225/00, 2000.

[26] H. Bay, T. Tuytelaars, L. V. Gool, SURF: Speeded Up Robust Features., in: Proceedings of the European conference on Computer vision, 2006.

[27] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal on Computer Vision 60 (2) (2004) 91–110.

[28] F. Pernici, A. D. Bimbo, Object tracking by oversampling local features, IEEE Transactions on Pattern Analysis and Machine Intelligence 99 (2013) 1.

[29] A. Criminisi, I. Reid, A. Zisserman, A plane measuring device, Image and Vision Computing (1999) 625–634.

[30] D. Liebowitz, A. Zisserman, Metric rectification for perspective images of planes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1998.

[31] O. Tuzel, F. Porikli, P. Meer, Pedestrian detection via classification on riemannian manifolds, Pattern Analysis and Machine Intelligence, IEEE Transactions on 30 (10) (2008) 1713 –1727.
[32] B. Benfold, I. Reid, Stable multi-target tracking in real-time surveillance video, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2011.

[33] S. T. Birchfield, S. Rangarajan, Spatiograms versus histograms for region–based tracking.

[34] R. J. Fitzgerald, Pack biases and coalescence with probabilistic data association., IEEE Transactions on Aerospace and Electronic Systems AES-21.

[35] http://www.micc.unifi.it/vim/datasets/mtt/.

[36] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Online multi-person tracking-by-detection from a single, uncalibrated camera, IEEE Transactions on Pattern Analysis and Machine Intelligence (99).

[37] X. Yan, X. Wu, I. Kakadiaris, S. Shah, To track or to detect? an ensemble framework for optimal selection, in: ECCV, 2012.

[38] H. Pirsiavash, D. Ramanan, C. C. Fowlkes, Globally-optimal greedy algorithms for tracking a variable number of objects, in: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2011.

[39] W. Choi, C. Pantofaru, S. Savarese, A general framework for tracking multiple people from a moving camera, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI).

[40] B. Yang, R. Nevatia, Online learned discriminative part-based appearance models for multi-human tracking, in: ECCV, 2012.

[41] B. Leibe, A. Leonardis, B. Schiele., Robust object detection with interleaved categorization and segmentation, International Journal of Computer Vision 77 (1-3) (2008) 259–289.

[42] C. Huang, R. Nevatia, High performance object detection by collaborative learning of joint ranking of granules features, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 41 –48.

[43] D. Reid, An algorithm for tracking multiple targets, IEEE Transactions on Automatic Control (1979) 843 – 854.

[44] M. Isard, A. Blake, Condensation conditional density propagation for visual tracking, International Journal of Computer Vision 29 (1998) 5–28.

[45] C. Tomasi, T. Kanade, Detection and tracking of point features, Tech. rep., International Journal of Computer Vision (1991).

[46] K. Bernardin, R. Stiefelhagen, Evaluating multiple object tracking performance: the CLEAR MOT metrics, Journal on Image and Video Processing 2008.