Optical Localization of Very Distant Targets in Multi Camera System

Jan Bednářik*

Abstract
This paper presents a system for automatic optical localization of distant moving targets using multiple pan-tilt cameras. The cameras were precisely calibrated and stationed using custom designed calibration targets and methodology. The detection of the target is performed manually, while the automatic visual tracker combines the background/foreground modeling and motion model in the particle filer framework. The estimation of the 3D location is based on the N-view triangulation. A basic setup consisting of two camera units was tested against static targets and a moving terrestrial target, and the location estimation precision was compared to the theoretical model. The modularity and portability of the system allows fast deployment in a wide range of scenarios including perimeter monitoring or early threat detection in defense systems, as well as air traffic control in public space.

Keywords: multi-camera localization — visual object tracking — 2D motion prediction — particle filter based tracking — stationing and rectification — articulated model of PT unit — 3D localization using triangulation — physical simulation using Gazebo — robotic system design using ROS

Supplementary Material: Demonstration Video

*jan.bednarik@hotmail.cz, Faculty of Information Technology, Brno University of Technology

1. Introduction

An autonomous localization of arbitrary moving targets is an essential system component in multiple domains, such as air traffic control, robotic workspaces or surveillance and defense systems. If the sensory data measured by the target are available, it is straightforward to derive its location (by means of the GPS, radio multilateration, etc.). There are scenarios, however, were the target is unable (malfunctioning aircraft) or reluctant (UAV intruder) to expose its location. Then the localization estimation system is left with its own observations.

Radars, the most widely used devices for localizing distant targets, suffer from being unportable, energy-intensive and expensive. Furthermore, it might be desirable that the tracked object not find out that it is being tracked, which is the condition an actively radiating system cannot achieve.

This paper introduces a semi-autonomous passive multi-camera system for tracking and localizing the distant objects, which is based merely on ordinary RGB cameras — Optical Localization System (OLS). The system is designed to suit mobility and temporary deployment because each camera station weights no more than twenty kilograms and the whole system is inexpensive by comparison to radars as well.
2. Related Work

The choice of how the targets are represented determines the domain of approaches used for visual detection and/or tracking. In general, two main representations are used [1]: a shape model which encompasses e.g. points [2], contours [3,4] or articulated models [5,6], and an appearance model which is represented by a template [7] or active appearance model [8].

Moving object detection Depending on the object model, the detection might be performed either by detecting keypoints and matching them against the pretrained model [9,10,11], or by dividing the image into individual patches in which the object is searched for. For each patch, the template matching is performed [12,7] or feature set is extracted; consequently, the model presence probability is evaluated using the generative or the discriminative classifier [13,14]. Since the exhaustive search within the whole image is computationally expensive, the cascade classifiers are applied [15,16]. Alternatively, the moving object can be detected in the image regions yielding the highest response of frame differencing [17,18].

Object tracking There are multiple approaches to visual tracking. Keypoint tracking represents one of the most common ones [2,19]. Kernel approaches are based on a weighted kernel used to derive smooth distance function which can be optimized in the means of target position using traditional gradient based methods such as gradient descent [20], or even multiple collaborative kernels might be used [21,22]. Other approaches rely on tracking-by-detection concept which heavily utilizes the detection principles in combination with motion-aware approaches to localize the object [23,24]. To reinforce the tracker robustness, the motion models are often used, Kalman filter and particle filter being the most popular ones [25,7].

Multi-view optical localization Multi-camera localization is mostly used in the domain of robotics, where the intelligent space consisting of several cameras with a priory known and fixed intrinsics and extrinsics is utilized [26]. The centralized system uses either mere visual information or enhances the localization with the help of robots’ sensory data [27,28,29,30]. Bound to the predefined space and using fixed cameras, those systems do not need to deal with the imprecise estimates of a current camera pose.

3. System Overview

The main building block of the OLS is a camera station (CS), a standalone unit consisting of hardware modules necessary for capturing the images, manipulating the pose of the camera and estimating its own geographical coordinates. There are two type of CSs. The overview station is designed to be controlled manually by the human operator and is equipped with the zooming lens that allows achieving both a wider scanning range and a more detailed view of the farther objects. The tracking station consists of the fixed lens and takes part in the autonomous tracking of the moving objects.

The OLS is designed to work with an arbitrary number of CSs. Due to the geometric limitations of the multi-view systems, which affect the localization precision, the tracking stations should be positioned so as to form approximately regular polygon with long enough bases (see Section 4, see Figure 1).

The camera station itself consists of a surveying tripod, a P&T unit1 Flir PTU-D46-70, a camera Prosilica GT 1290C (RGB, 1280 × 960 px, 33.3 FPS), an inclinometer and a GPS sensor. A camera unit is modeled as a kinematic chain consisting of six joints and five links corresponding to the distance between separate parts of the tripod and the manipulator (see Figure 2). The transformation between the GROUND and ORIGIN reflects the positioning and heading of the given manipulator within the local coordinate frame.

The kinematic chain is designed as composition of transformation matrices where a single joint can be located by applying the Euclidean transformation on the position of the joint which it is depending on:

\[
M_{\text{next}} = M_{\text{previous}}T_{\text{next}}R_{\text{next}}R_{\text{next}}^{-1},
\]

where \(M\) is the transformation matrix of the given joint, \(T\) is the translation between successive joints and \(R_a\) is the rotation around axis \(a\).

4. Localization Precision

The precision of estimating the target position is subject to systematic error (miscalibration of the instru-
ments) and random error (wrong measurements and disturbances in the environment) [31]. Atmospheric turbulence, refractive index fluctuations and uncertainty of the visual tracker are the main causes of the random error which is analysed in section 4.1. The systematic error was alleviated and/or measured using the custom designed stationing and rectification process.

### 4.1 Random Error Analysis

Stereoscopic systems are affected by a phenomenon of diminishing accuracy of depth measurement with increasing distance of the target from the cameras [32]. The depth measurement resolution for canonical stereo setup is $R = \frac{f^2 Z^2}{b - pr Z}$, where $f$ is the focal length, $b$ is the base length, $r$ is the horizontal size of one pixel and $Z$ is the target distance. By substituting $r$ by $pr$, where $p$ is the random error represented by integer number of pixels we obtain the position estimation error function

$$E = \frac{pr^2}{f b - pr Z}.$$

The OLS does not conform to the canonical stereo setup (all cameras can rotate freely), so the dependence of the error on the target distance is no longer quadratic (considering the setup of two cameras where only one of them makes error): $E = B \tan(\arctan(\frac{D}{b}) + \arctan(\frac{f}{r})) - D$. The cameras setup as well as the error shown as the function of the base length and target distance is depicted in Figure 3.

A more realistic scenario, where each camera makes a random error $p$, is depicted in Figure 4. A significant advantage of using multiple cameras is demonstrated — geometrical limitations of the two-camera setup make it impossible to precisely evaluate the position of the target placed close to the line collinear with the baseline. In the multi-camera setup, on the other hand, the subset of two cameras forming the baseline $B_i$ is used for each position of the target, following the rule:

$$i = \arg\max_i \vec{t}_i \vec{n}_i,$$

where $\vec{t}_i$ is the direction of the line segment linking the center of the baseline and the target and $\vec{n}_i$ is the normal vector of the baseline $B_i$.

### 4.2 Stationing and Rectification

The stationing procedure alleviates two types of systematic error: wrong heading estimation and undesirable tilt of the camera station. Since the accuracy of the commercial magnetometers is too low (hundreds to thousands of milliradians), the precise heading must be estimated visually by observing the distinctive landmarks. To achieve the horizontality of the station a digital inclinometer can be used.

The imprecision of the camera-manipulator attachment causes slight undesirable rotation of the camera coordinate frame. Three horizontally leveled rectification targets are used to alleviate and/or measure all rotation angles (around X, Y and Z axis): 5):

**Rotation around optical axis** The target contains parallel horizontal lines and the camera displays the blend of the original and vertically mirrored streams. The aim is to rotate the camera manually so that the
lines would appear aligned.

**Rotation around azimuthal axis** The target contains parallel horizontal lines and a pair of crosses whose distance equals the distance between ELE and CAMERA joint (see Figure 2). The aim is to measure the distance between the right cross and the intersection of the optical axis with the target, which translates to an error angle in the azimuth.

**Default elevation angle** Two targets which contain black and white lines representing a ruler are positioned in a row. The aim is to adjust the tilt of the camera so that the optical axis would intersect the same mark on both targets and the resulting elevation angle could be measured.

## 5. Object Tracking

The detection itself has been performed manually so far in the man-in-the-loop manner, while the autonomous tracking uses the implementation of the visual tracker combining the background subtraction, motion model and object model in the particle filter framework [7]. This approach can even cope with the moving cameras and thus is suitable for the OLS. The operation of the tracker is described below.

The target is represented as a rectangular template (consisting of gray-scale intensity values), which is normalized to the size $24 \times 24$ pixels. The advantage of the template representation is that it contains both spatial and appearance information. The template is created only once during the initialization, and thus the tracker could fail if the target changed its appearance significantly during the course of tracking. However, for very distant targets, no or merely small change is expected.

The Bootstrap particle filter (BPF) — the variant of a particle filter following the sequential importance sampling approach [33] — is used to generate and evaluate candidate positions of the target. Each particle (i.e. the state of the system) is represented as $\vec{x}_n = (x, y, v_x, v_y, h, w)$, where $(x,y)$ represents the 2D position of the target, $(v_x, v_y)$ represents the estimated speed of the target and $(h, w)$ represents the bounding box size.

The perturbations in the observed position of the target caused by the moving camera are alleviated using the motion model which is applied in the prediction step of the BPF:

$$pos_{n+1} = pos_n + vel_n + \gamma_{pos} \sim \mathcal{N}(\mu, \sigma),$$

$$vel_{n+1} = vel_n + \gamma_{vel} \sim \mathcal{N}(\mu, \sigma),$$

$$bb_{n+1} = bb_n + \gamma_{bb} \sim \mathcal{N}(\mu, \sigma),$$

where scalar $pos_n$ is the x or y position, scalar $vel_n$ is the x or y velocity, scalar $bb_n$ is the w or h size of the bounding box in time n, and $\gamma$ is the noise drawn from the Gaussian distribution $\mathcal{N}(\mu, \sigma)$, where scalars $\mu$ and $\sigma$ parameters are set empirically for each parameter.

In the update step, each particle is assigned a new weight $w$ using the objective function reflecting the similarity of the template and the candidate patch:

$$w = \sum_{(x,y) \in I} e^{\min(M_t^{(x,y)} \cdot M_c^{(x,y)})(1 - |l_t^{(x,y)} - l_c^{(x,y)}|)^2},$$

where $M_t$ and $M_c$ are the foreground masks (FM) of the template and the current candidate respectively, $I$ is the image, $t,c$ subscripts denote template and candidate patch respectively, and $(x,y)$ superscript denotes indexing 2D array (an image). The FMs are estimated by subtraction of the two images where the bounding boxes denoting the position of the target do not overlap (the FM $M_t$ is estimated only once). The resulting estimate of the target position is chosen using the Maximum a posteriori approach.

In order to enable the motion of the camera, the transformation between each pair of adjacent frames is estimated by detecting and tracking the keypoints using KLT tracker [2] and then estimating the homography using the RANSAC algorithm [34].

The homography might not be found, which often occurs if the airborne target with the uniform background of the sky is tracked or if the manipulator moves the camera too harshly. To deal with such cases in OLS, the tracker was adjusted so that in a prediction step of the BPF a small subset of particles would be forced to take the image positions yielding the highest response of adjacent frame differencing which is expected to contain the moving target of interest.
where $\vec{R}$ is the rotation matrix and $C$ is the translation vector representing the orientation and position of the camera frame. The 3D point $\vec{X}$ projects to the 2D image point $\vec{x}$ via $\vec{x} = \vec{P} \vec{X}$. If only the projection $\vec{x}$ is observed, the 3D line mapping to $\vec{x}$ can be computed using back-projection:

$$X(\lambda) = P^+ \vec{x} + \lambda C,$$

where $P^+$ is the pseudo-inverse of $P (P^+ = P^T (PP^T)^{-1})$.

In OLS, the intrinsics were estimated for each camera during calibration and extrinsics are known at each time due to the sensory data streamed from the manipulators. However, the rays backprojected from each camera might not intersect in the 3D space due to both systematic and random errors (see Figure 6). The estimation of the 3D position of the target consists of the following steps. First, back-projection is used to find the vectors $\vec{u}$ and $\vec{v}$ which form the planes $M_1 M_2 U$ and $M_1 M_2 V$ with the angle $\alpha$ between them. Both vectors are then rotated around the axis $\vec{m}$ so that they lie in the same plane $M_1 M_2 W$. $\vec{u}' = R(\beta_1) \vec{u}$, $\vec{v}' = R(\beta_2) \vec{v}$. The rotation angles might be of the same value $\beta_1 = \beta_2 = \alpha / 2$; however, to achieve higher precision the angles might be weighted by the trackers’ beliefs: $\beta_1 = \alpha - \beta_2$. Finally, the intersection $\vec{W}$ of the vectors $\vec{u}'$ and $\vec{v}'$ is found.

If multiple camera units are used, 3D location can be estimated as the weighted centroid of the estimates computed by each pair of the camera units forming the baseline $b_i$ (see Algorithm 1). The weights correspond to the angle between the baseline and the line intersecting the (estimated) position of the target and the baseline center, since this angle significantly affects the precision (see Section 4).

Since the 3D position estimation might be completely wrong occasionally, the position estimates are smoothed by the moving average computed over $h$ consecutive estimates ($h$ was empirically set to 10).

**Algorithm 1:** Estimation of the 3D position from n-views

**Input:** Set of bases $B = b_1, b_2, ..., b_N$.

**Output:** 3D position estimate $T$.

/* 3D location estimate disregarding weights */
1 foreach $b_i \in B$
2 $\vec{t}_i = \text{Estimate3DPosFrom2Views}(b_i)$
3 end
4 $\vec{T}' = \frac{1}{N} \sum_{i=1}^{N} \vec{t}_i$

/* Weighted estimation of the 3D location. $bc_i$ represents center of the baseline $b_i$ */
5 foreach $b_i \in B$
6 $w_i = -\frac{e^{\beta_i (\vec{T}' - \vec{b}_i)}}{\sum_{j=1}^{N} e^{\beta_j (\vec{T}' - \vec{b}_j)}}$
7 end
8 $\vec{T} = \sum_{i=1}^{N} w_i \vec{b}_i$

7. Implementation and Experimental Results

The implementation is built on a robotic framework ROS$^2$ and a physical simulator Gazebo$^3$. ROS was chosen for its wide support of hardware components and a seamless way to implement multi-process distributed system. The whole system was modeled and simulated in Gazebo (see Figure 7), which facilitated hardware-in-a-loop testing of the manipulators [35].

The system was tested in the real-world environment in the basic two-camera setup. The CSs were precisely positioned using the differential GPS sensor (achieving accuracy of ca 0.01 m) so that the base would be exactly 30 m long. The local heading was estimated by aiming the units on each other. The system

---

$^2$Robot Operating System: [http://www.ros.org](http://www.ros.org)

was tested against both static and dynamic targets, and in both cases only horizontal position was considered.

As for the static targets, nine landmarks with a priori known UTM coordinates (obtained from the cadastral map) and one target carrying an ordinary mobile GPS sensor were chosen (see Figure 8). The localization error, given as the Euclidean distance between the ground truth and the estimated locations, was compared with the estimated error (see Table 1). Note that both measured and estimated error follow the same trend (see Figure 9); however, the measured error is higher mainly due to the insufficient precision of calibration, stationing and rectification.

### Table 1

<table>
<thead>
<tr>
<th>object</th>
<th>dist. [m]</th>
<th>α [rad]</th>
<th>est. Δ [m]</th>
<th>Δ [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>pillar1</td>
<td>91,92</td>
<td>0,38</td>
<td>0,20</td>
<td>4,41</td>
</tr>
<tr>
<td>pillar2</td>
<td>199,14</td>
<td>0,46</td>
<td>0,95</td>
<td>5,51</td>
</tr>
<tr>
<td>pillar3</td>
<td>285,01</td>
<td>0,48</td>
<td>1,93</td>
<td>11,73</td>
</tr>
<tr>
<td>pillar4</td>
<td>386,81</td>
<td>0,41</td>
<td>3,39</td>
<td>17,16</td>
</tr>
<tr>
<td>tree1</td>
<td>433,88</td>
<td>0,34</td>
<td>4,13</td>
<td>17,20</td>
</tr>
<tr>
<td>person</td>
<td>479,96</td>
<td>0,10</td>
<td>4,65</td>
<td>23,90</td>
</tr>
<tr>
<td>hide</td>
<td>526,86</td>
<td>0,77</td>
<td>7,57</td>
<td>22,77</td>
</tr>
<tr>
<td>tree2</td>
<td>634,46</td>
<td>0,35</td>
<td>8,56</td>
<td>28,33</td>
</tr>
<tr>
<td>mast</td>
<td>1379,67</td>
<td>0,33</td>
<td>41,24</td>
<td>34,21</td>
</tr>
</tbody>
</table>

The system was tested against one dynamic terrestrial target equipped with a mobile GPS sensor (a walking person). The target was tracked for 120 s and the estimated positions were captured and compared to the ground truth path (see Figure 10). On average the system achieved the precision of 6.25 m. Note that the position estimates oscillate around the ground truth trajectory, which is caused by the random error made by both trackers; the error, however, keeps in the specific range and reaches maximum of 13.35 m. The mean error is higher as compared to the estimated error (see Section 4), which is again caused by the systematic error (imprecise calibration, rectification and stationing).

### 8. Conclusion

This paper introduced a novel system capable of autonomous tracking and localization of distant moving targets using multiple cameras. The paper proposes precision analysis which aims on finding and alleviating the most prominent sources of error, as well as the methodology to calibrate and station all camera units.

The system utilizes a visual tracker based on the Bootstrap particle filter framework combining both visual and motion model of the target and positionable camera. The localization of the target uses the principle of triangulation, where both the belief of the tracker and the geometrical limitations given by the angle between the base and the target are incorporated into the final weighted estimate.

The system was tested in real world conditions against static and dynamic targets whose position was
known either from the cadastral map or captured by the GPS sensor. The localization precision follows the trend of diminishing accuracy of depth measurement and reaches slightly higher error than the theoretical model, namely due to the insufficiently precise calibration, rectification and stationing. This, however, can be improved by using more reliable hardware components and by performing the rectification procedure more thoroughly.

Though still in early development, the OLS system has great potential for being widely used as a passive, modular and highly portable substitute for the recently widely used radars for the applications ranging from automatic traffic control to national defense systems protecting the sensitive perimeters.

In the near future, the OLS system will be extended by the 3D environment reconstruction subsystem which should make the tracker predict occlusion and estimate more accurately the motion of the tracked target. Furthermore, more thorough tests will be carried out in order to spot the sources of error and reinforce the overall precision.

Acknowledgements

This work is supported by RCE systems s.r.o. which provided the development space, necessary hardware sources, and consultations. I thank my Master’s thesis supervisor, prof. Ing. Adam Herout, Ph.D., for guidance, and my adviser, doc. Ing. Vladimír Čech, CSc., for providing his insight into the project.

References


