Planar Features for Visual SLAM

Tobias Pietzsch

Technische Universität Dresden, International Center for Computational Logic,
01062 Dresden, Germany
Tobias.Pietzsch@inf.tu-dresden.de

Abstract. Among the recent trends in real-time visual SLAM, there has been a move towards the construction of structure-rich maps. By using landmarks more descriptive than point features, such as line or surface segments, larger parts of the scene can be represented in a compact form. This minimises redundancy and might allow applications such as object detection and path planning. In this paper, we propose a probabilistic map representation for planar surface segments. They are measured directly using the image intensities of individual pixels in the camera images. Preliminary experiments indicate that the motion of a camera can be tracked more accurately than with traditional point features, even when using only a single planar feature.

1 Introduction

In simultaneous localisation and mapping (SLAM), we are concerned with estimating the pose of a mobile robot and simultaneously building a map of the environment it is navigating [1]. The problem is formulated in a Bayesian framework where noisy measurements are integrated over time to create a probability distribution of the state of a dynamical system, consisting of landmark positions and the robot's pose. Visual SLAM tackles this problem with only a camera (monocular or stereo, typically hand-held) as a sensor.

Since the seminal work by Davison [2], the majority of existing systems for visual SLAM build sparse maps representing 3D locations of scene points and their associated uncertainties. This particular representation has been attractive because it allows real-time operation whilst providing sufficient information for reliable tracking of the camera pose. If we want to visual SLAM to move beyond tracking applications, we need maps which allow geometric reasoning. Planar structures allow a compact representation of large parts of the environment. For tasks such as scene interpretation, robot navigation and the prediction of visibility/occlusion of artificial objects in augmented reality, maps consisting of planar features [3, 4] seem more suitable than wire frame models [5, 6].

In this paper, we propose a map representation for planar surface segments within the framework of [2]. Those planes are measured directly using the image intensities of individual pixels in the camera images.

Gee et al.[3] presented a visual SLAM system in which planar structural components are detected and embedded within the map. Their goal is to compact

the representation of mapped points lying on a common plane. However these planes are not directly observable in the camera images, but are inferred from classical point feature measurements. Our approach has a closer relationship to the works by Molton et al. [4] and Jin et al. [7]. We also use pixel measurements directly to update estimates of plane parameters. Molton et al. [4] regard feature points as small locally planar patches and estimate their normal vectors. However, this is done outside of the SLAM filter, thus ignoring correlations between normal vectors and rest of the state. While being useful for improving the observability of point features, their approach does not scale to larger planar structures because these correlations cannot be neglected in this case. Jin et al. [7] are able to handle larger planes but rely on the linearity of the image gradient which limits their approach with respect to image motion and tolerable camera acceleration. We solve this issue by casting the measurement update in an iterative framework. Furthermore, our feature model employs an inverse-depth parameterisation which has been shown to be more suitable to linearisation than previous representations [8].

In the following section, we describe Davison's framework [2] for visual SLAM. In Sec. 3, we introduce our feature representation and measurement model. After discussing details of the measurement process in Sec. 4, we present experimental results for simulated and real scenarios in Sec. 5. Sec. 6 concludes the paper.

2 EKF-Based Visual SLAM

The task in visual SLAM is to infer the state of the system, i.e., the pose of the camera and a map of the environment from a sequence of camera images. A commonly used and successful approach is to address the problem in a probabilistic framework, using an Extended Kalman Filter (EKF) to recursively update an estimate of the joint camera and map state [2, 5, 3, 4].

The belief about the state $\mathbf{x}$ of the system is modeled as a multivariate Gaussian represented by its mean vector $\mu_{\mathbf{x}}$ and covariance matrix $\Sigma_{\mathbf{x}}$. The state vector can be divided into parts describing the state of the camera $\mathbf{x}_v$ and of map features $\mathbf{y}_i$.

$$\mu_{\mathbf{x}} = \begin{pmatrix} \mu_{\mathbf{x}_v} \\ \mu_{\mathbf{y}_1} \\ \vdots \\ \mu_{\mathbf{y}_n} \end{pmatrix} \quad \Sigma_{\mathbf{x}} = \begin{bmatrix} \Sigma_{\mathbf{x}_v\mathbf{x}_v} & \Sigma_{\mathbf{x}_v\mathbf{y}_1} & \cdots & \Sigma_{\mathbf{x}_v\mathbf{y}_n} \\ \Sigma_{\mathbf{y}_1\mathbf{x}_v} & \Sigma_{\mathbf{y}_1\mathbf{y}_1} & \cdots & \Sigma_{\mathbf{y}_1\mathbf{y}_n} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{\mathbf{y}_n\mathbf{x}_v} & \Sigma_{\mathbf{y}_n\mathbf{y}_1} & \cdots & \Sigma_{\mathbf{y}_n\mathbf{y}_n} \end{bmatrix} \tag{1}$$

The state estimate is updated sequentially using the predict-update cycle of the EKF. Whenever a new image is acquired by the camera, measurements of map features can be made and used to update the state estimate, resulting in a decrease of uncertainty in the update step. In the prediction step, a process model is used to project the estimate forward in time. The process model describes how the state evolves during the period of "temporal blindness" between images. Only the camera state $\mathbf{x}_v$ is affected by the process model, because we assume

an otherwise static scene. The camera is assumed to be moving with constant linear and angular velocity. The (unknown) accelerations that cause deviation from this assumption are modeled as noise. This results in an increase of camera state uncertainty in the prediction step. Similar to [2] we model the camera state as $\mathbf{x}_v = \begin{pmatrix} \mathbf{r} & \mathbf{q} & \mathbf{v} & \boldsymbol{\omega} \end{pmatrix}^\top$. Position and orientation of the camera with respect to a fixed world frame $\mathcal{W}$ are described by the 3D position vector $\mathbf{r}$ and the quaternion $\mathbf{q}$. Translational and angular velocity are described by $\mathbf{v}$ and $\boldsymbol{\omega}$.

The EKF update step integrates new information from measurements of map features into the state estimate. A generative measurement model

$$\mathbf{z} = \mathbf{h}(\mathbf{x}_v, \mathbf{y}_i) + \boldsymbol{\delta} \tag{2}$$

describes the measurement vector $\mathbf{z}$ as a function of the (true, unknown) state of the camera $\mathbf{x}_v$ and the feature $\mathbf{y}_i$. The result is affected by the zero-mean measurement noise vector $\boldsymbol{\delta}$. The current estimate of the camera and feature state can be used to *predict* the expected measurement. The difference between the predicted and actual measurement is then used in the EKF to update the state estimate.

In [2], a map feature $\mathbf{y}_i = \begin{pmatrix} x & y & z \end{pmatrix}^\top$ describes the 3D coordinates of a fixed point in the environment. A measurement of such a feature then consists of the 2D coordinates $\mathbf{z} = (u, v)^\top$ of the projection of this point into the current camera image. To be able to actually make measurements, new features are initialised on salient points in a camera image. With each new feature a small template patch of its surrounding pixels in this initial image is stored. In subsequent images, measurements of this feature are made by searching for the pixel with maximal correlation to this template. This search is carried out in a elliptic region determined by the predicted measurement and its uncertainty.

3 Feature Representation and Measurement Model

The template-matching approach to the measurement process works well as long as the feature is observed from a viewpoint reasonably similar to the one used for initialisation. To improve the viewpoint range from which a feature is observable, we can try to predict the change in feature appearance due to changed viewpoint prior to the correlation search. To do this, we assume that the feature is located on a locally planar scene surface. Given the surface normal and the prior estimate of the camera position the features template patch can be warped to give the expected appearance in the current camera image. Simply assuming a surface normal facing the camera in the initial image already introduces some tolerance to changing viewing distance and rotation about the camera axis. Molton et al. [4] go one step further by estimating the surface normal using multiple feature observations from different viewpoints. Both approaches are aimed at improving the stability of the template matching procedure by removing the effects of varying viewpoint.

However, changes in feature appearance also provide information which can be directly used to improve the state estimate. For instance if we observe that

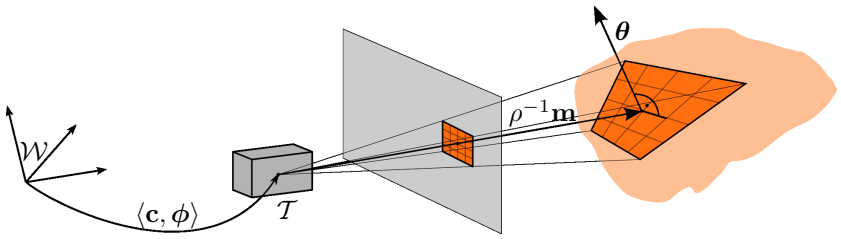


Fig. 1. The relative orientation of world frame $\mathcal{W}$ and template frame $\mathcal{T}$ is given by translation $\mathbf{c}$ and rotation $\boldsymbol{\phi}$. The unit vector $\mathbf{m}$ defines a ray to the feature center, ρ is the inverse depth along this ray. The plane normal is described by $\boldsymbol{\theta}$.

the scale of the observed image patch is larger than we expected, this tells us that the camera is closer to the feature than we predicted. Also, the amount of perspective distortion is directly related to the relative orientation between the camera and scene surface. To exploit such information, modeling a feature measurement as a single 2D coordinate as described above is not sufficient.

In the following, we describe a feature representation of planar segments in the scene. A measurement model of such a planar features is developed directly using the raw camera images, i.e., a measurement comprises a set of pixel intensity values. In our system, features represent planar surface segments in the environment. Such a segment is described in terms of its appearance in the image where it was initially observed, and the position of the camera when this initial image was taken, cf. Fig. 1.

For each feature, the initial camera image is stored as the template image $T : \mathbb{R}^2 \to \mathbb{R}$. We assume that the outline of the compact image area corresponding to the planar scene segment is known. Some (arbitrary) pixel within this area is chosen as the *feature center* projection. The unit vector $\mathbf{m}$ is the ray from the camera center through this pixel. Both T and $\mathbf{m}$ are fixed parameters, i.e., not part of the probabilistic state.

In the EKF state vector, the i-th map feature is parametrised as

$$\mathbf{y}_i = \left(\mathbf{c}\ \boldsymbol{\phi}\ \rho\ \boldsymbol{\theta}\right)^\top \quad . \tag{3}$$

The parameters $\mathbf{c}$, $\boldsymbol{\phi}$ describe the camera translation and rotation at the initial observation. This initial camera coordinate frame is referred to as the template frame $\mathcal{T}$. The parameter ρ is the inverse depth[1] measured along the ray defined by $\mathbf{m}$. The last component $\boldsymbol{\theta}$ is the normal vector of the feature plane, encoded in polar coordinates. Both $\boldsymbol{\theta}$ and $\mathbf{m}$ are measured with respect to the template frame $\mathcal{T}$.

Having defined the feature representation, we can now proceed to formulate a measurement model of the form (2). It is well-known from the computer vision

[1] Inverse depth parameterisation was chosen, because it can cope with distant features and its better suited to approximation by a Gaussian than depth [8].

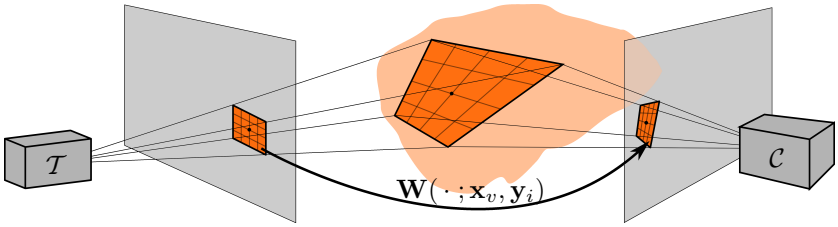


Fig. 2. The camera $\mathbf{x}_v$ and feature state $\mathbf{y}_i$ define a homography warp $\mathbf{W}$ between the template and current image. $\mathbf{W}$ can be used to predict intensities in the current image by looking up the intensities at the corresponding template pixels. The measurement consists of the observed intensities inside the feature outline in the current image.

literature that two images of a plane are related by a homography transformation [9]. Thus, pixel coordinates in the current camera image and the template image can be related through the homography induced by the feature plane. We will denote by $\mathbf{W}(\cdot\,;\mathbf{x}_v,\mathbf{y}_i)$ the homography warp function which maps points in the template image T to the corresponding points in the current image C, cf. Fig. 2. The homography is fully determined by the orientation of both the template and current camera coordinate frames, and the position and normal of the feature plane. Therefore, $\mathbf{W}$ is parameterised by the current state of the camera $\mathbf{x}_v$ and the feature state $\mathbf{y}_i$. The exact form of the warp function can be derived similarly to [4]. This is omitted here due to space limitations.

Using the warp function $\mathbf{W}(\cdot\,;\mu_{\mathbf{x}_v},\mu_{\mathbf{y}_i})$ parameterised on the prior state estimate, we can project the features outline into the current image to determine where we expect to observe the feature. Let $\mathbf{U} = \{\mathbf{u}_1,\ldots,\mathbf{u}_m\}$ be the set of pixel locations inside the expected projected outline. The measurement vector $\mathbf{z} = (z_1,\ldots,z_m)^\top$ consists of the intensities measured at these locations, i.e., $z_j = C(\mathbf{u}_j)$.

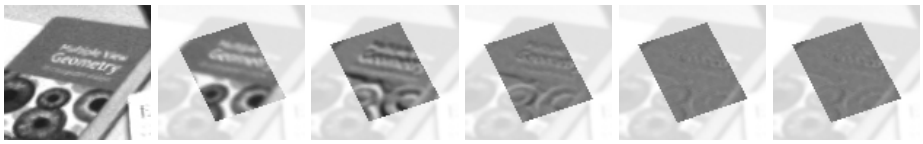
To formulate a generative model, we have to define the function $\mathbf{h}(\mathbf{x}_v,\mathbf{y}_i)$, i.e., we have to express what we expect to measure given a certain camera and feature state. The intensity we expect to measure at a given location $\mathbf{u}_j$ in the current image is the intensity at the corresponding location in the template image. This corresponding location is computed using the inverse of the warp function, $\mathbf{W}^{-1}$. Thus, the generative model for a single pixel is

$$z_j = h_j(\mathbf{x}_v,\mathbf{y}_i) + \delta_j = T(\mathbf{W}^{-1}(\mathbf{u}_j;\mathbf{x}_v,\mathbf{y}_i)) + \delta_j \tag{4}$$

where $\delta_j \sim \mathcal{N}(0,r_j)$ is zero-mean Gaussian noise with variance r_j, uncorrelated between pixel locations.

4 Measurement Process

The EKF update step involves a linearisation of the measurement model, i.e., (2) is replaced by its first-order Taylor expansion around the prior state estimate.



(a) (b) (c) (d) (e) (f)

Fig. 3. A real-world example of the measurement update iteration. (a) the current image. (b) the predicted feature template. (c)-(d) show the difference between the warped template and the current image. (c) before the first iteration, i.e., the difference between (a) and (b). (d) after the first iteration with the restricted measurement model. (e,f) after the second and third iteration which are performed using the full planar measurement model.

For the proposed measurement model, (4) must be linearised for every intensity measurement z_j. This involves linearisation of the template image function T around the template pixel position predicted as corresponding to $\mathbf{u}_j$. The intensity in the neighbourhood of a template pixel is approximated using the image gradients at that pixel. Image functions in general are highly non-linear. Therefore, the linearised model used in the EKF will provide good approximation only in a small region of state-space around the true state. To deal with this problem, we use an Iterated Extended Kalman Filter (IEKF). The basic idea of the IEKF is that because the posterior estimate is closer to the true state than the prior estimate, linearisation should rather be performed around the posterior estimate. The update step is repeated several times, where in each iteration linearisation is performed around the posterior estimate obtained in the previous iteration. During rapid camera accelerations the prior estimate can be quite far from the true state, rendering the initial linearisation of T useless. To guide the update process to the correct region of convergence the first iteration is done using a restricted model where the measurement consists only of the 2D image coordinates of the projection of the feature center. The measurement is obtained using point-feature-like correlation search for the feature template warped according to the prior estimate. This iterative process is illustrated in Fig. 3.

In our current implementation we use a stereo camera. Conceptually there is not a great difference between a monocular and a stereo setting. The additional intensity measurements obtained from the second camera eye contribute additional pixel measurements (4) with a different warp function $\mathbf{W}'$ which takes into account the baseline offset of the second eye from the camera center.

5 Experimental Results

We compared the performance of the proposed measurement model to that of a point-based model on a artificial stereo image sequence. The sequence was generated using the raytracer POV-Ray. The scene consists of a single large planar object. The camera parameters were chosen to resemble a Point Grey

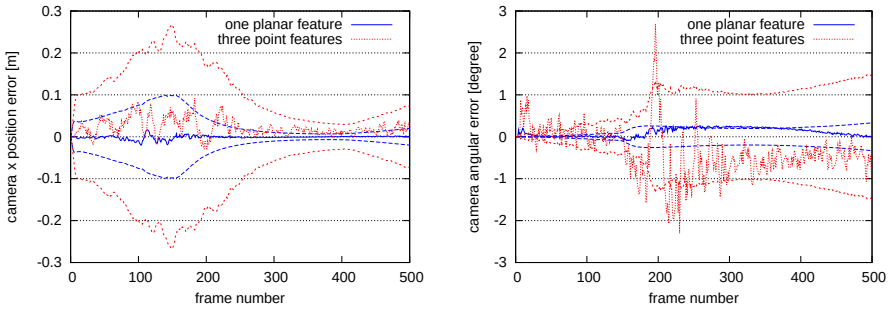


Fig. 4. Comparison of absolute errors in estimated camera orientations for the artificial image sequence. The position error in x direction is shown on the left. The total angular error is shown on the right. The dashed lines indicate the 3σ confidence bounds.

Bumblebee® stereo camera.[2] The camera is initially 1 m away from the planar object. It travels a path where the object is viewed from varying distance (0.4 to 1.7 m) and varying angles (0° to 70°).

To fully constrain the camera orientation estimate only one (sufficiently large) planar feature is needed. With the point-based model three features are needed. We use state-of-the-art point features with an inverse-depth representation and predictive template warping assuming camera-facing normal. All features where initialised by hand on salient image areas.

In Fig. 4 we compare the errors of the reconstructed camera trajectories with respect to ground truth. The trajectory obtained using the planar feature is more accurate and a lot smoother than the trajectory obtained using point features. Several factors contribute to this result. By using intensities at a large number of pixels[3] as measurements much more information from the images is used than by artificially reducing the measurement to a single 2D coordinate for each of the point features. Also, point feature measurements are made at integer pixel coordinates, although the approach could be extended to compute a sub-pixel interpolation of the correlation search maximum. Iterating the update process as described in Sec. 4 provides sub-pixel accurate registration of the feature template while at the same time keeping the same stability with respect to prior uncertainty as point features.

We have also tested the planar feature model with real image sequences, that were recorded with a Point Grey Bumblebee® stereo camera. Examples from one sequence are shown in Fig. 5. Again, only a single planar feature is used. In this case, the feature template is 41×41 pixels. The feature was selected by hand on a salient image area. The feature normal estimate is initialised with high uncertainty as pointing towards the camera. Convergence to the (visually judged)

[2] This type of camera was used for the real-world experiments.

[3] The feature used corresponds to a 71×71 pixel patch in the initial image, which means that ca. $10,000$ intensity measurements are contributed by one stereo image.

Fig. 5. Results for some pictures of a real image sequence using a single planar feature. The projected outline of the feature is shown in red. A coordinate frame attached to the reconstructed feature plane such is shown in black.

correct normal occurs after one frame. No ground truth was available for this sequence. To give an impression of the accuracy of the reconstruction the images in Fig. 5 are augmented with a coordinate frame attached to the reconstructed feature plane such that its Z axis coincides with the plane normal.

6 Conclusion

We presented a representation of plane segments as features in a probabilistic map and a method to directly measure such features in camera images. We have shown in experiments that the motion of a camera can be tracked more accurately than with traditional point features, even when using only a single planar feature. In order to build a fully functional SLAM system one important issue has yet to be addressed, namely the *detection* of planar scene structures which can be used as features. This is the focus of immediate future work.

References

1. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics. MIT Press (2005)
2. Davison, A.J.: Real-Time Simultaneous Localisation and Mapping with a Single Camera. In: International Conference on Computer Vision. (2003)
3. Gee, A.P., Chekhlov, D., Mayol, W., Calway, A.: Discovering Planes and Collapsing the State Space in Visual SLAM. In: BMVC. (2007)
4. Molton, N.D., Davison, A.J., Reid, I.D.: Locally Planar Patch Features for Real-Time Structure from Motion. In: BMVC. (2004)
5. Smith, P., Reid, I., Davison, A.: Real-Time Monocular SLAM with Straight Lines. In: BMVC. (2006)
6. Eade, E., Drummond, T.: Edge Landmarks in Monocular SLAM. In: BMVC. (2006)
7. Jin, H., Favaro, P., Soatto, S.: A semi-direct approach to structure from motion. The Visual Computer **19**(6) (2003) 377–394
8. Montiel, J., Civera, J., Davison, A.J.: Unified inverse depth parameterization for monocular SLAM. In: Robotics: Science and Systems (RSS). (2006)
9. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge Univerity Press (2000)