REMODE (**RE**gularized **MO**nocular **D**epth **E**stimation)

Naam	Geïnstalleerd	Open Source	Configuratie	ROS benodigd	Up-To-Date	Licentie
REMODE	Nee	Ja	Monoculair	(Ja?)	Nee (latest commit Dec 5, 2015)	GPL v3

Demonstration of the approach:
REMODE

Github page:
REMODE - Github

System Overview

REMODE is used to solve the problem of estimating dense and accurate depth maps from a single moving camera. A probabilistic depth measurement is carried out in real time on a per-pixel basis and the computed uncertainty is used to reject erroneous estimations and provide live feedback on the reconstruction progress. It's a novel approach to depth map computation that combines Bayesian estimation and recent development on convex optimization for image processing. We call our approach REMODE (REgularized MOnocular Depth Estimation) and the CUDA-based implementation runs at 30Hz on a laptop computer.

Problems

We present a method to compute an accurate, three- dimensional reconstruction of the scene observed by a moving camera and provide, in real time, information about the progress and the reliability of the ongoing estimation process. This problem is highly relevant in robot perception, where cameras are valuable and widespread sensors. From a single moving camera, it is possible to collect appearance and range information about the observed three-dimensional scene. In a multi-view stereo setting, the uncertainty on the depth measurement depends on the noise affecting image formation, on the camera poses, and the scene structure. Knowing how these factors affect the measurement uncertainty, it is possible to achieve arbitrarily high levels of confidence by collecting measurements from different vantage points. The pose of the camera must be known and its accuracy influences the reconstruction quality. For a camera, information resides in the changing of the intensity gradient and this modality naturally fails in presence of low informative scenes that produce untextured images. It is therefore crucial to know how reliable each measurement is.

Contributions and Outline

A compact representation and a Bayesian depth estimation from multi- view stereo were proposed. We build on their results for per-pixel depth estimation and introduce an optimization step to enforce spatial regularity over the recovered depth map. We propose a regularization term based on the weighted Huber norm but, we use the depth uncertainty to drive the smoothing and exploit a convex formulation for which a highly parallelizable solution scheme has been recently introduced. The contributions are the following:

- A probabilistic depth map, in which the Bayesian scheme is integrated in a monocular SLAM algorithm to estimate per-pixel depths based on the live camera stream;
- A fast smoothing method that takes into account the measurement uncertainty to provide spatial regularity and mitigates the effect of noisy camera localization.

Monocular Dense Reconstruction

A. Considerations

The solution we propose to compute a dense reconstruction from a single moving camera is motivated by the following considerations.

a) A measure of uncertainty is needed in robotic perception: we are interested in accurately mapping the environment in order to allow robotic tasks, such as autonomous navigation and exploration, active perception or situation awareness in the case of human- operated systems. As a passive sensing modality, measurement uncertainty in monocular multi-view stereo is related to the camera motion and the amount of visual information present in the scene (e.g. texture). A probabilistic depth map handles measure uncertainty, thus, allowing efficient updating, optimal sensor placement, and fusion with different sensors.

b) A dense reconstruction is needed to interact: however, feature definitions change between sensing modalities and tasks; dense representations are, thus, required to actually solve the problem of registering data among largely different vantage points based on the three-dimensional structure. When the task involves physical interaction with the environment—as in obstacle avoidance, path planning and manipulation—the highest achievable level of detail is desirable in order to estimate the surfaces involved in the interaction.

c) Perception must be fast: depth estimation must be updated efficiently and the uncertainty in the estimation must improve according to the information conveyed by the image and the current camera pose. Depth estimation must take into account the uncertainty arising from the scene and the camera pose and the estimation must be carried out online and updated sequentially. Bayesian estimation offers a natural way to deal with measure uncertainty, to handle sequential measurement updates and to reject unreliable estimations in an online fashion.

B. Depthmap from Multi View Stereo

We formulate the depth computation as a Bayesian estimation problem. Each observation provides a depth measurement by triangulating from the reference view and the last acquired view. The depth of a pixel is described by a parametric model that is updated on the basis of the current observation. Finally, smoothness on the resulting depth map is enforced by minimizing a regularized energy functional.

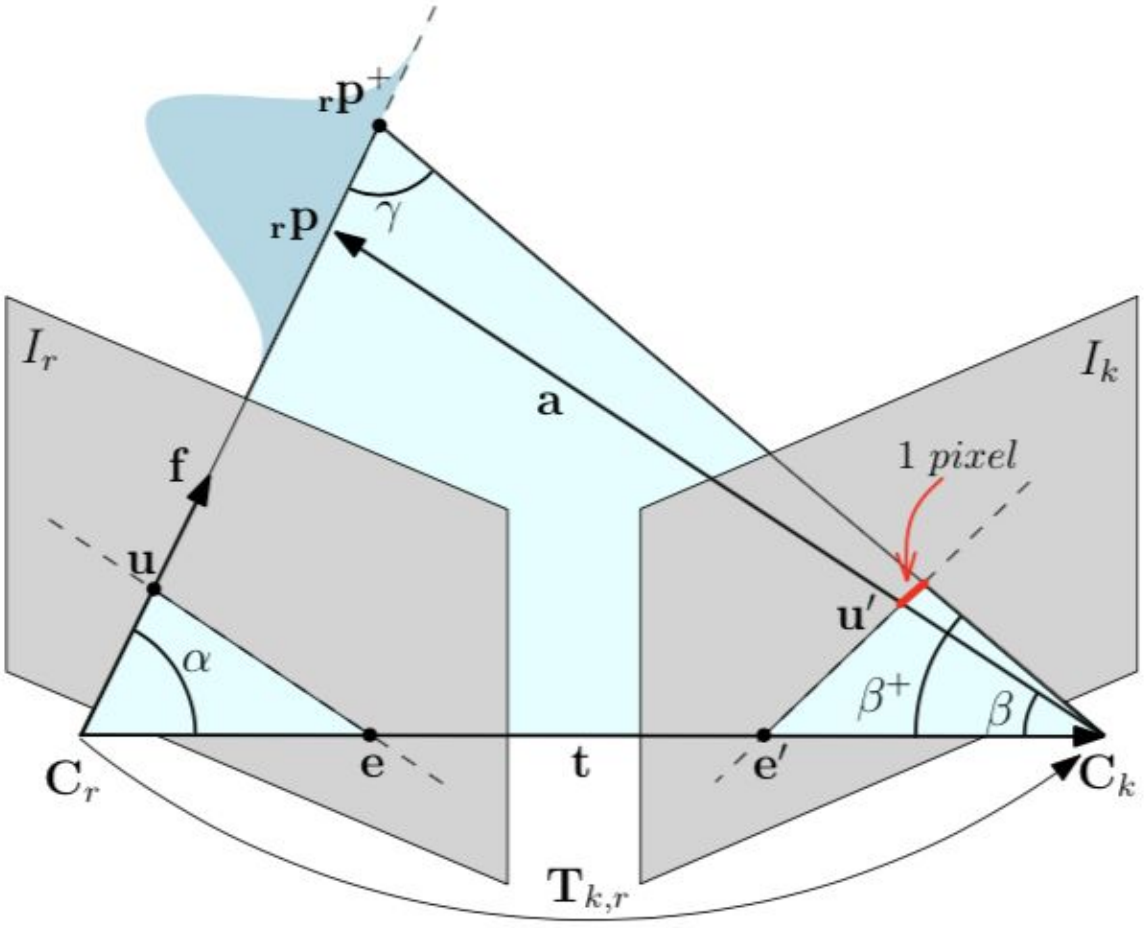

Fig. 2. Computation of the measurement uncertainty. The camera poses acquiring the views I_r and I_k are related by the transformation $\mathbf{T}_{k,r}$. The camera centres C_r, C_k and the current estimation of the scene point $_r\mathbf{p}$ lie on the epipolar plane.The variance corresponding to one pixel along the epipolar line passing through $\mathbf{e}'$ and $\mathbf{u}'$ is computed as $\tau_k^2 = \left(\|_r\mathbf{p}^+\| - \|_r\mathbf{p}\|\right)^2$.

Implementation

A. Camera pose estimation

At every time step , the pose of the camera in the depth map reference frame is computed by a visual odometry routine that is based on recent advancement on semi-direct methods for camera localization. The algorithm operates directly on the image intensity, eliminating the need for costly feature extraction and resulting in sub-pixel accuracy at high frame-rates. Our implementation is characterized by an average drift in pose of 0.0038 metres per second for an average depth of 1 metre and a computing time of 3.3 milliseconds per acquired image on the experimental platform. The visual odometry algorithm is run by the CPU, and its accuracy and efficiency support the simultaneous execution of the monocular reconstruction pipeline.

B. Measurement update

The parametric model is a compact representation, as it stores our confidence in the depth measurement corresponding to a pixel in only four parameters. When a reference frame is taken, the estimation for every pixel is initialized and updated with every subsequent view. We set the initial parameters. Upon the acquisition of the k-th view, the update is performed for every pixel of the reference view. We perform the update until the depth estimation converges or diverges. At this point, we can either consider the measurement reliable or discard it. We check the convergence and divergence conditions by looking at the variance of the depth posterior and the estimated inlier ratio.

The parameters control the estimation convergence and can be set according to the accuracy and robustness requirements for the application at hand. In order to deal with higher depth ranges, we base our implementation on the inverse depth and use the currently estimated variance to limit the search for correspondence on the epipolar line.

C. Measurement uncertainty

When triangulating matched points to estimate the depth from multiple views, frames taken from nearby vantage points are less affected by occlusions and allow high quality matches. On the other hand, a large baseline enables a more reliable depth estimation but with a higher chance to incur in occluded regions. Referring to Figure 2, let ${}_r p$ be the current estimation of the scene point corresponding to the pixel u in the image I_r. The variance on the position of ${}_r p$ is obtained by back-projecting a constant variance of one pixel in the image I_k. Let f be the camera focal length. The angle spanning one pixel can be added to β in order to compute γ and, thus, by applying the law of sines, recover the norm of ${}_r p_+$.