Upright and Stabilized Omnidirectional Depth Estimation for Wide-baseline Multi-camera Inertial Systems

Changhee Won[1,2], Hochang Seok[2] and Jongwoo Lim*[1,2]
[1] MultiplEYE Co., Ltd., Seoul, Korea.
[2] Department of Computer Science, Hanyang University, Seoul, Korea.
{chwon, hochangseok, jlim}@hanyang.ac.kr

Abstract

This paper presents an upright and stabilized omnidirectional depth estimation for an arbitrarily rotated wide-baseline multi-camera inertial system. By aligning the reference rig coordinate system with the gravity direction acquired from an inertial measurement unit, we sample depth hypotheses for omnidirectional stereo matching by sweeping global spheres whose equators are parallel to the ground plane. Then, unary features extracted from each input image by 2D convolutional neural networks (CNN) are warped onto the swept spheres, and the final omnidirectional depth map is output through cost computation by a 3D CNN-based hourglass module and a softargmax operation. This can eliminate wavy or unrecognizable visual artifacts in equirectangular depth maps which can cause failures in scene understanding. We show the capability of our upright and stabilized omnidirectional depth estimation through experiments on real data.

1. Introduction

Omnidirectional vision is becoming more popular for various application, *e.g.*, AR/VR, robotics, or automonous driving systems. In vision-based navigation systems for autonomous vehicles, it is necessary to estimate omnidirectional depths to detect and avoid surrounding obstacles. To this end, many omnidirectional vision systems and depth estimation methods have been proposed, for example, multiply mounted stereo cameras [15], wide field-of-view (FOV) fisheye lenses [2, 17, 16], and 360° catadioptric lenses or spherical panoramic cameras can be used for the omnidirectional depth estimation.

Recently, a wide-baseline omnidirectional multi-view stereo setup which uses only four fisheye cameras each of which has 220° FOV, and a spherical-sweeping method in which the input fisheye images are warped onto the pre-defined global spheres to match omnidirectional stereo correspondences are presented [17]. Further in [16], the spherical-sweeping method is adopted in deep neural networks, which consists of the unary feature extraction, spherical-sweeping, and cost volume computation blocks, and the proposed network named OmniMVS estimates an omnidirectional depth map in an end-to-end manner. Meanwhile, using the same camera systems, [12] have proposed a robust omnidirectional visual odometry (VO), ROVO, which utilizes omnidirectional visual features extracted from the proposed hybrid-projection images. Both OmniMVS [16] and ROVO [12] are extended and integrated into a localization and dense mapping system, OmniSLAM [18]—omnidirectional depth maps output by OmniMVS are integrated into the VO for better feature tracking, and estimated depth maps and rig poses are fused into a global 3D map. OmniSLAM [18] has shown the capability of its omnidirectional vision system, and favorable 3D perception performances through experiments on both indoor and outdoor rig setups.

In the process of the spherical-sweeping proposed in [17], the rig center is chosen for the origin of the spherical coordinate system, and the xz-plane is aligned close to the camera centers, *i.e.*, the equators of the global spheres are in the camera center plane, and the output omnidirectional depth map follows equirectangular projection (ERP) corresponding to the spherical coordinate system. In real-world environments, the camera rig can move in arbitrary direction (*e.g.*, mounted on a drone or hand-held by a person), and the result ERP depth maps or warped images are rotated and the structures and lines become wavy. These visual artifacts can lead to failures in visual scene understanding, *e.g.*, semantic segmentation or object detection. Also shakes of the rig cause large and consistent rotational motion in result videos, and the users or recognition systems can be confused.

In this paper, we present an upright and stabilized omnidirectional depth estimation from an arbitrarily rotated multi-camera inertial systems. We estimate the gravity di-

*Corresponding author.

rection using an inertial-measurement unit (IMU) in the initialization stage, and we align the rig poses from VO with the initialized gravity direction. With the gravity-aligned rig poses, we sweep global spheres so that the equators are parallel to the ground plane with uniformly sampled depth hypotheses to find omnidirectional stereo correspondences. Using deep neural networks, OmniMVS [18] extracts deep features from input images, and then warps onto the upright spheres to compute the matching cost volume by a 3D encoder-decoder architecture. Finally the upright stabilized omnidirectional depth map is acquired by a softargmax operation.

2. Related Work

Many algorithms have been proposed for the omnidirectional depth estimation [2, 11, 15]. Schönbein *et al.* [11] use rectified omnidirectional images from two horizontally mounted 360° FOV catadioptric cameras for disparity estimation. Gao and Shen [2] projects the ultra-wide FOV fisheye images from two vertically mounted cameras onto four directions to compute omnidirectional depth maps. Im *et al.* [4] use a short omnidirectional video for depth estimation of static scenes by temporally matching correspondences between adjacent frames. Recently SweepNet [17] and OmniMVS [16] propose a wide-baseline camera system with ultra-wide FOV lenses for omnidirectional depth estimation. SweepNet computes the matching costs between pairs of views, then the cost volume is processed by SGM [3]. On the other hand OmniMVS adopts an end-to-end deep neural network that considers all images at the same time. It can handle several hard cases of multiple true matches or textureless regions by learning the regularization of the cost volume using a 3D encoder-decoder network. Meanwhile, there have also been methods for camera rotation estimation from 360° images by extracting lines and vanishing points [1, 6], and deep neural networks [5].

3. Upright Stabilized Depth Estimation

In this section we briefly summarize the OmniMVS and ROVO algorithms, then describe the upright sweeping algorithm for upright and stabilized depth estimation.

3.1. Review of OmniMVS

We use the same wide-baseline multi-camera rig system and spherical sweeping with [17, 16, 18]. OmniMVS [16, 18] consists of unary feature extraction, spherical sweeping, and cost volume computation modules for dense omnidirectional depth estimation by multi-view stereo matching. The input images from the cameras are processed using the unary feature extraction network to build deep feature maps $\{F_i\}$, where i indicates the camera index. Similar to plane sweeping in conventional stereo, spherical sweeping generates a series of N concentric spheres with different radii to build the spherical feature maps for dense matching. Specifically, for each inverse depth hypothesis d, each ray in the $W \times H$ equirectangular image $\mathbf{p}(\theta, \phi) = (\cos(\phi)\cos(\theta), \sin(\phi), \cos(\phi)\sin(\theta))^\top$, where (θ, ϕ) is the spherical coordinate of $\mathbf{p}$, is projected to the corresponding sphere of radius $1/d$. The inverse radius d_n is swept from 0 to d_{N-1}, where $1/d_{N-1}$ is the minimum depth being considered. The 3D point $\mathbf{X}$ on the sphere is projected back to the input images to find the deep feature vectors for the ray, *i.e.*, for the i-th camera, the image pixel coordinate $\mathbf{x}_i$ is computed by the projection function Π_i with the intrinsic and extrinsic parameters; $\mathbf{x}_i = \Pi_i(\mathbf{X})$. Thus a point at $\langle\theta, \phi\rangle$ on the sphere of radius ρ is projected to $\Pi_i(\rho\,\mathbf{p}(\theta, \phi))$ in the i-th fisheye image.

The unary feature maps of $\frac{1}{2}W_I \times \frac{1}{2}H_I \times C$ with C channels are extracted from the input fisheye images of $W_I \times H_I$. The equirectangular spherical feature map for the n-th sphere is built by

$$S_{n,i}(\theta, \phi) = F_i(\frac{1}{2}\Pi_i(\mathbf{p}(\theta, \phi)/d_n)), \tag{1}$$

where F_i is the unary feature map of i-th camera and d_n is the n-th inverse depth. The result spherical feature maps are four $W \times H \times C \times (N/2)$ tensors (only every other spheres are used due to memory and speed issues), and they are concatenated to build the 4D omnidirectional feature data for matching.

Finally the 3D encoder-decoder architecture computes and regularizes the cost volume of $W \times H \times (N/2)$. The minimum index of inverse depths for each ray is chosen by the softargmax operation.

3.2. Review of ROVO

To estimate rig poses, we adopt ROVO [12] without depth map integration [18]. ROVO consists of four steps: hybrid projection, intra-view tracking and inter-view matching, robust pose estimation, and joint pose optimization.

Instead of using raw fisheye images or rectified images using a pinhole camera model, ROVO uses the hybrid projection model which combines planar and cylindric projection models. It can handle ultra-wide FOV imgaes with $> 180\,\mathrm{deg}$ with little radial distortion and also improves inter-camera feature matching as the appearances in two images are similar. The ORB features [9] are detected in each projected image, and are temporally tracked using KLT [7]. The inter-view feature matching between adjacent cameras is performed, where feature correspondences across views enable 3D triangulation of the features. From the 2D-3D feature correspondences, the rig pose is computed using robust multi-view P3P RANSAC [12] which uses the correspondences in all views, and the the result pose is optimized by minimizing the reproection errors of all inlier features in

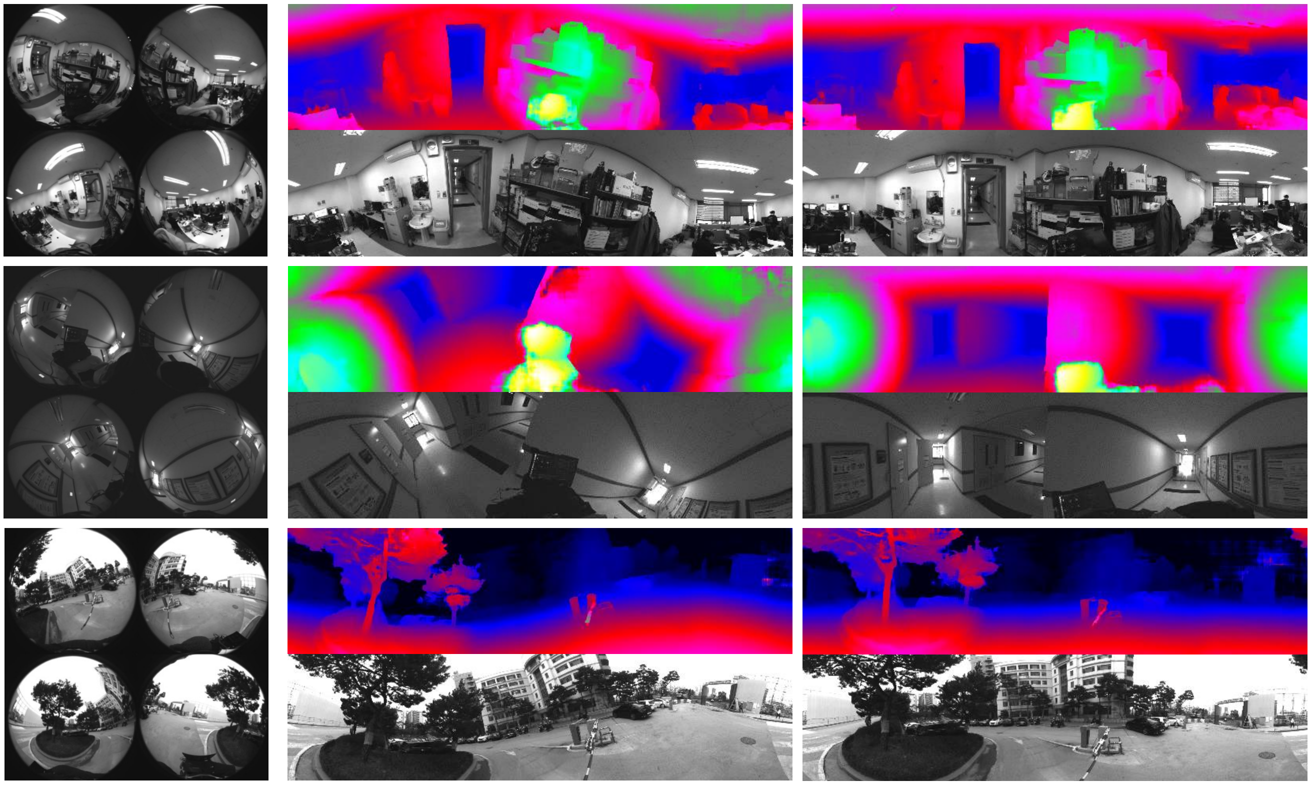

Figure 1: Qualitative results of upright adjusted omnidirectional depth estimation. From left: input images, before applying adjustment, and after. Color-coded inverse depth map and corresponding re-projected reference panorama images are shown.

all views. Finally, the estimated poses and reconstructed 3D feature points in the local window are simultaneously optimized using local bundle adjustment (LBA) [13].

3.3. Upright Spherical Sweeping

OmniMVS uses the default coordinate system whose xz-plane is aligned with the camera centers. This is a reasonable option as the epipolar lines between views are close to the horizontal axis of the images. However when the rig is moving freely in space, the horizon of the scene is not aligned with the horizontal axis of the estimated depth map, which causes the horizon to a wavy curve. In continuous capture, the camera motion appears in the result video, and sometimes it is preferable to have a stabilized output.

To this end we utilize an inertial measurement unit (IMU) to find the gravity direction, and the rig rotation estimated by ROVO [12, 18] to align the rig with the gravity. In the initialization stage, we assume that the rig is stationary, so that the gravitational acceleration $\mathbf{g} \in \mathbb{R}^{3\times1}$ is calculated by taking average of the accelerometer values accumulated for a short period of time ($<$ 50 ms), and the gravity direction $\bar{\mathbf{g}}$ is acquired by normalizing $\mathbf{g}$. Since the gravity is consistently observed from the accelerometer when the rig is stationary, the estimated direction is quite stable, and also the computational cost is negligible. We then transform the initialized gravity direction to the world coordinate system by using calibrated extrinsic between the IMU and the camera rig.

The rotation for upright adjustment has 2-DOF (the roll and the pitch) as the yaw is not related to the undesirable visual changes. Therefore, we compute the the rotation matrix $R \in \mathbb{R}^{3\times3}$ needed for the upright adjustment by aligning the up-vector of the rig pose with the gravity direction as:

$$\bar{\mathbf{g}} = R_i(-\mathbf{u}_i),$$

where $\mathbf{u}_i \in \mathbb{R}^{3\times1}$ is the up-vector of the estimated i-th rig pose. Following Eq. 1, the pixel value of the aligned n-th spherical image from i-th camera is determined as

$$\hat{S}_{n,i}(\theta,\phi) = F_i\big(\frac{1}{2}\Pi_i\big(R^\top(\mathbf{p}(\theta,\phi)/d_n)\big)\big), \qquad (2)$$

We warp the unary features extracted by the 2D CNN onto the aligned spheres, and the final omnidirectional depth map is acquired through the cost computation block and the softargmax operation [16, 18].

4. Experiments

We use a square-shaped rig (0.3 × 0.3 m) with four 220° fisheye cameras and one Xsens MTi-10 IMU sensor, and we

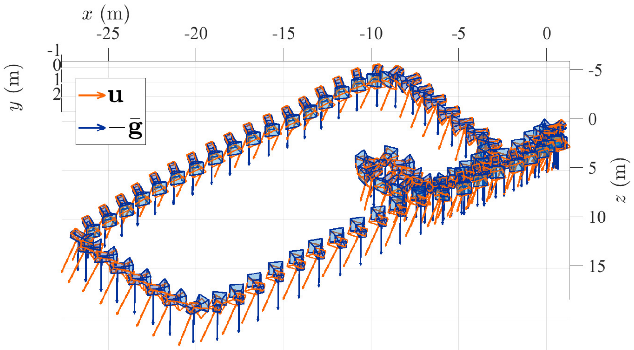

Figure 2: Gravity aligned visual odometer result. Orange denotes the original odometer estimated by ROVO [12, 18], and blue denotes the estimated gravity direction and the aligned odometer.

capture $4 \times (1600 \times 1532)$ gray images (20 Hz) and IMU measurements (200 Hz). We calibrate the intrinsic and extrinsic between cameras using a checkerboard [10, 14, 17]. Then, we also use the extrinsic between the camera and IMU which calibrated by the Kalibr [8]. For the depth and pose estimation, we use the OmniMVS and ROVO proposed in [18] without additional training and depth map integration. We set the size of output depth map to $W = 640$ and $H = 160$, the number of inverse depths $N = 192$, and ϕ from $-45°$ to $45°$. For ROVO, we set the number of features to 4×250, and the size of hybrid projection image to 640×480.

We show the qualitative results of upright adjusted omnidirectional depth map in Fig. 1. Indoor walls and floors as well as outdoor environments are well corrected, and objects in the adjusted depth maps and the reference images are more recognizable than without adjustment. Figure 2 also shows the odometer result on a sequence taken around a corridor of a building with the rig tilted to one side, and the rig poses estimated VO (orange) are well-aligned according to the gravity direction (blue).

5. Conclusions

In this paper we propose a upright and stabilized omnidirectional depth estimation algorithm. Based on the superior performance of OmniMVS and ROVO, the proposed algorithm keeps the estimated depth maps to be upright and stabilized by aligning the y axis with the gravity direction. The experimental results shows that the proposed algorithm can handle arbitrary rotational rig motions.

References

[1] Jean-Charles Bazin, Cédric Demonceaux, Pascal Vasseur, and Inso Kweon. Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment. *The International Journal of Robotics Research (IJRR)*, 31(1):63–81, 2012. 2

[2] Wenliang Gao and Shaojie Shen. Dual-fisheye omnidirectional stereo. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6715–6722, 2017. 1, 2

[3] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(2):328–341, 2008. 2

[4] Sunghoon Im, Hyowon Ha, François Rameau, Hae-Gon Jeon, Gyeongmin Choe, and In So Kweon. All-around depth from small motion with a spherical panoramic camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 156–172, 2016. 2

[5] Junho Jeon, Jinwoong Jung, and Seungyong Lee. Deep upright adjustment of 360 panoramas using multiple roll estimations. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 199–214, 2018. 2

[6] Jinwoong Jung, Beomseok Kim, Joon-Young Lee, Byungmoon Kim, and Seungyong Lee. Robust upright adjustment of 360 spherical panoramas. *The Visual Computer*, 33(6-8):737–747, 2017. 2

[7] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. 1981. 2

[8] Joern Rehder, Janosch Nikolic, Thomas Schneider, Timo Hinzmann, and Roland Siegwart. Extending kalibr: Calibrating the extrinsics of multiple imus and of individual axes. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4304–4311, 2016. 4

[9] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R Bradski. Orb: An efficient alternative to sift or surf. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, volume 11, page 2, 2011. 2

[10] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A flexible technique for accurate omnidirectional camera calibration and structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision Systems (ICVS)*, pages 45–45, 2006. 4

[11] Miriam Schönbein and Andreas Geiger. Omnidirectional 3d reconstruction in augmented manhattan worlds. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 716–723, 2014. 2

[12] Hochang Seok and Jongwoo Lim. Rovo: Robust omnidirectional visual odometry for wide-baseline wide-fov camera systems. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 6344–6350, 2019. 1, 2, 3, 4

[13] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *International Workshop on Vision Algorithms*, pages 298–372, 1999. 3

[14] Steffen Urban, Jens Leitloff, and Stefan Hinz. Improved wide-angle, fisheye and omnidirectional camera calibration. *ISPRS Journal of Photogrammetry and Remote Sensing*, 108:72–79, 2015. 4

[15] Yanchang Wang, Xiaojin Gong, Ying Lin, and Jilin Liu. Stereo calibration and rectification for omnidirectional multi-camera systems. *International Journal of Advanced Robotic Systems (IJARS)*, 9(4):143, 2012. 1, 2

[16] Changhee Won, Jongbin Ryu, and Jongwoo Lim. Omnimvs: End-to-end learning for omnidirectional stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 8987–8996, 2019. 1, 2, 3

[17] Changhee Won, Jongbin Ryu, and Jongwoo Lim. Sweepnet: Wide-baseline omnidirectional depth estimation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 6073–6079, 2019. 1, 2, 4

[18] Changhee Won, Hochang Seok, Zhaopeng Cui, Marc Pollefeys, and Jongwoo Lim. Omnislam: Omnidirectional localization and dense mapping for wide-baseline multi-camera systems. *arXiv preprint arXiv:2003.08056*, 2020. 1, 2, 3, 4