3D Reconstruction using HDR-based SLAM

Chia-Hung Yeh[1, 2, *], Min-Hui Lin[1] and Wei-Chieh Lu[2]
[1]Department of Electrical Engineering, National Sun Yat-sun University, Kaohsiung, Taiwan
[*]E-mail: yeh@mail.ee.nsysu.edu.tw Tel: +886-02-77343545
E-mail: d063010005@student.nsysu.edu.tw
[2]Department of Electrical Engineering, National Taiwan Normal University, Taipei, Taiwan
E-mail: jeff09830222@gmail.com

***Abstract*— 3D reconstruction is the key technology to emerging technologies such as smart robotics, VR/AR/XR and autonomous driving. To enhance the robustness of our proposed 3D reconstruction system, the HDR-based SLAM is adopted in the camera pose estimation step to improve the qualitative result of geometric reconstruction. The proposed HDR-based SLAM uses the pre-calibrated inverse camera response function (CRF) to map a single RGB image into a radiance map. To exclude the influence of exposure time, normalized radiance maps independent of exposure time are used during tracking. Since ORB feature matching is the basic element of tracking and mapping in our system, the ORB descriptor patch is re-trained especially for normalized radiance maps. Experimental results have shown good performance of our system under challenging low-light environment, which helps expand the applicability of 3D reconstruction system.**

I. INTRODUCTION

Human discover and explore the globe through their sensory perceptions among which the vision is the most important one, while the computer receive most of the information with the aid of computer vision techniques in the real world. With the advance of artificial intelligence technology, the field of computer vision has become more popular than ever. On account of its wide range of applications, 3D scene reconstruction has been one of the most popular topics in computer vision over the past few years. Thanks to the launch of the consumer-grade depth sensor, the depth information of the objective could be obtained more efficiently and economically.

In 3D reconstruction, RGB-D camera is the most commonly used sensor for acquiring color and depth information due to its low-cost compared with other high expense of precise instruments. For tracking objectives in the scene, there are mainly two techniques. The first one is the simple frame-to-frame tracking, only the registration between the current frame and the previous frame are conducted based on either point-to-point or point-to-plane error matrix. However, serious error accumulation could occur over time. To diminish the problem, frame-to-model tracking has been widely used in recent reconstruction frameworks. Frame-to-model tracking establish a global model which latter frames are aligned with, thus reducing temporal error propagation. Newcombe et al. [1] use only depth images and course-to-fine iterative closest point (ICP) algorithm to fuse the depth information into the global model. Zhou et al. [2] find points of interest through density function and obtain globally consistent pose estimation for every frame in the scene to reduce alignment errors. Choi et al. [3] introduce global pose optimization on the basis of line processes which makes the reconstruction pipeline robust to erroneous alignment results. Dai et al. [4], instead of registering between neighboring frames, combine SIFT feature points with pose estimation framework to align current frame with keyframes. Our reconstruction pipeline details will be introduced in the following contents.

The High-dynamic-range (HDR) imaging is the technique to reproduce a wider range of brightness levels than the conventional one, which brings a deeper contrast to the screen, greater color intensity without being oversaturated, and more detail in low-light images [5]. For low-dynamic-range (LDR) imaging, a scene is captured using single exposure and the brightness levels are only 256 (8-bit unsigned char), resulting in overexposed bright regions or underexposed dark ones. In contrast, HDR imaging uses 32-bit float values per channel to better represent the luminance information similarly to the human visual system.

HDR images can be obtained using either hardware or software. The hardware one uses multiple devices or a device with specially designed CCD sensors, which is usually not for commercial purposes [6]–[7]. Alternately, the software one is more applicable, which uses common camera to obtain LDR images first and then transform them to HDR images by algorithms. The most common multi-exposure image fusion technique captures several images of the same scene with different exposure times, then merge them to generate a HDR image [8]–[10]. When the scene is dynamic or being captured hand-held, the misalignment issue and ghosting artefact need to be dealt with [11]–[12]. In addition, a HDR image can also be generated by a single LDR image using histogram-based methods [13]–[14] or deep learning [15]–[16].

In the field of computer vision, given that HDR imaging can preserve details in both extremely dark and light regions, it has great potential to facilitate various tasks, such as 3D reconstruction [17]–[18], visual simultaneous localization and mapping (visual SLAM) [19]–[20], object recognition [21] and image correction [22]. For 3D reconstruction, Meilland et al. [17] is the pioneer work focusing on real-time HDR texture mapping. In their visual SLAM system, gamma-based inverse CRF is used to transform RGB images into radiance domain and use them for tracking. Because the system relies on built-in auto exposure (AE), camera transformation and exposure time need to be estimated jointly. Li et al. [18] also relies on AE but decouples exposure compensation from tracking. By

using the normalized radiance maps that is independent of exposure time, the tracking becomes more robust. Recently, some researches focus on actively controlling the exposure time [19]–[20] to improve visual SLAM in HDR environments.

Unlike the previous works, which are based on dense-SLAM systems, we propose a feature-based HDR-SLAM, and incorporate it into the 3D reconstruction pipeline to improve the reconstructed results under low-light environments. The remainder of this paper is organized as follows. In Sec. II, we present the proposed 3D reconstruction pipeline. In Sec. III, the proposed HDR-based SLAM is elaborated. In Sec. IV, experimental results are demonstrated. Finally, Sec. V concludes this paper.

II. Proposed 3D Reconstruction Pipeline

As shown in Fig. 1, the proposed 3D reconstruction pipeline consists of the following three steps: (1) Use the commodity handheld RGB-D camera, such as ASUS Xtion, Microsoft Kinect or Intel RealSense to scan a scene or an object. In this step, 640×480 color and depth images are acquired and registered. (2) Use the HDR-based SLAM to estimate the camera trajectory. (3) Reconstruct the 3D surface mesh by fusing depth frames into the truncated signed distance function (TSDF) volume. (4) Map color images onto the geometric reconstruction. The details of these steps will be elaborated in the following subsections.

A. Camera Pose Estimation

Camera pose estimation plays an important role in 3D reconstruction because more accurate camera trajectory can generate better geometric model. Since Visual odometry (VO) incrementally estimate the current camera pose based on the previous motion, the measurement errors would accumulate over time and lead to serious odometry drift. Whereas Visual SLAM additionally builds a globally consistent map and uses loop closure detection to correct drift, so it can provide more accurate camera pose [23].

ORB-SLAM2 [24] is known as one of the renowned Visual SLAM systems, which is lightweight given the real-time performance on standard CPUs. By using strategies including loop closing, relocalization, map reuse and bundle adjustment, it can achieve state-of-the-art accuracy in a wide variety of environment. We choose ORB-SLAM2 in this step and further improve it using HDR images as the proposed HDR-based SLAM, which will be described in Section III.

B. Reconstruct The 3D Surface Mesh

To reconstruct the surface mesh, we first fuse each depth images into the TSDF volume [25] and then extract the mesh model. The TSDF volume is a 3D cube subdivided into a set of voxels. Each voxel in the volume contains a TSDF value and a weighting. The TSDF value stores the distances from the voxels to the observed surface, and the value is positive when in front of the surface, negative when behind, and nearing zero when at the surface. To obtain the fused volume, for each raw depth map, the data is integrated into the volume from the corresponding camera pose and the TSDF values are incrementally updated using a weighted average. Then, marching cube [26] is used to find the zero-crossings in the volume and generate the triangle mesh.

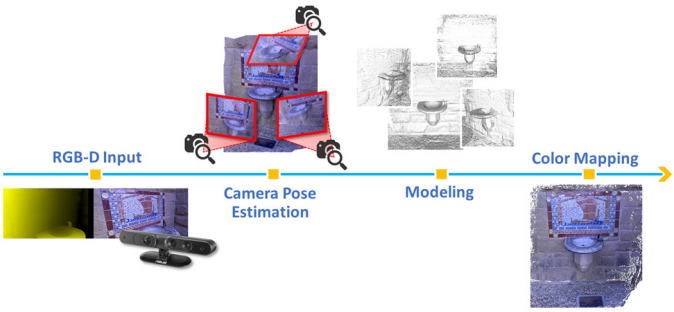


Fig. 1. The 3D reconstruction pipeline.

C. Color Mapping

The last step of 3D reconstruction is color mapping, which maps several color frames to the surface of a model to generate textured results. Here we use the comprehensive multi-view stereo texturing methods [27] to generate the color texture. Compared with volumetric blending used in many reconstruction systems [28], it can solve the blurring, ghosting and other visual artifacts and generate better results. Although the color mapping is the final part to complete the 3D reconstruction system, we will not show the textured models in Section IV since the texture would affect the qualitative evaluation of geometric reconstruction.

III. Proposed HDR-based SLAM

Compared to color images, HDR images in float format can present broader range of luminance present in real environment. The HDRFusion [18] shows that by integrating the radiance map into the SLAM system, both tracking and mapping can be improved. Therefore, to improve the ORB-SLAM2, two modifications are made: (1) Use normalized radiance maps as input instead of RGB-D images (2) Train the patch-descriptor especially for normalized radiance maps.

A. Generate the Radiance Map

When the depth sensor is held by hand to record a scene or an object in sequence, the common HDR imaging methods combining multiple images with different exposures is not applicable. Given that the camera response function (CRF) f can map the relationship between RGB pixel values to radiances, the inverse CRF f^{-1} is used to generate an HDR image from a single exposed LDR image. The CRF is defined as in [29]:

$$B = f(R + n_s(R) + n_c) + n_q, \tag{1}$$

where B is a pixel brightness value ranged from 0 to 255 and R is a radiance value. n_s is the noise dependent to radiance, n_c is the constant noise and n_q is the additional quantization noise, which can be ignored. Also, both the means of n_s and n_c are equal to zero, and their variances are defined as $Var(n_s) = R\sigma_s^2$ and $Var(n_c) = \sigma_c^2$ respectively.

The CRF of each camera is different, so it needs to pre-calibrated. The calibration setting is placing the depth sensor at fixed position, and capturing images with different exposures. Then, using the method described in [18], the CRF of our Asus Xtion sensor can be calculated.

B. Generate the Radiance Map

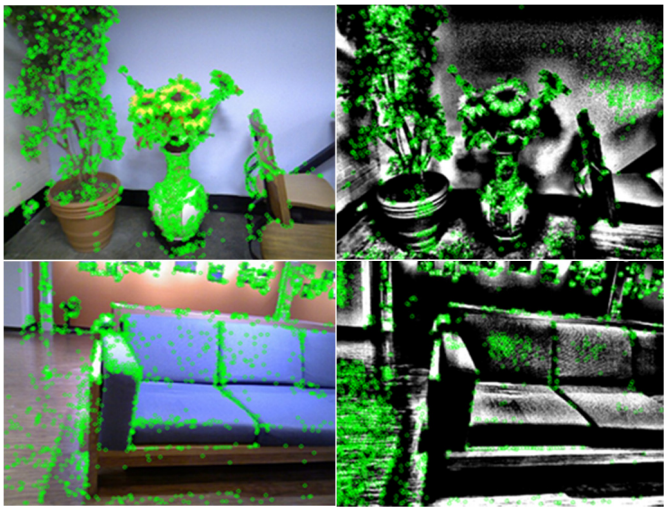
Radiance R measures how much luminance a sensor captured within exposure time Δt, which is formulated as $R = L\Delta t$. As in [18], the normalized radiance map is proposed as follows:

$$\overline{R_N}(u) = \frac{R_N(u) - E(R_N)}{\sqrt{Var(R_N)}} = \frac{R_N(u)\Delta t - E(L_N \Delta t)}{\sqrt{Var(L_N \Delta t)}} = \frac{L_N(u) - E(L_N)}{\sqrt{Var(L_N)}}, \quad (2)$$

where N is the 80×80 patch, u is a pixel location in the patch N, $\overline{R_N}(u)$ is the normalized value at pixel u, $E(R_N)$ is the mean radiance of the N, and $\sqrt{Var(R_N)}$ is the standard deviation of radiances in N. For example, a 640×480 radiance map would be divided into $(8 \times 6 = 48)$ numbers of 80×80 patches, then normalization by (2) is performed individually in each patch.

We can see that after normalization, $\overline{R_N}(u)$ is independent of exposure time Δt, and this property is proved to be useful when video flickering happens [18]. Depth sensors are usually equipped with default auto exposure to better capture images similar to the one being seem from human visual system. When the camera is moved from bright area to dark area, the exposure time is set longer gradually to make the image brighter.

However, when the camera is moved fast across the boundary of bright and dark area, the exposure time changes drastically, so that the captured sequence flickers. The issue is called video flickering, which would reduce the accuracy of camera tracking, or even fail to track. Because the normalized radiance map is invariant to exposure time, it can better represent the scene comparing to color image when video flickering happens. In addition, HDR images can present wider range of light conditions. Therefore, we use normalized radiance map as input of our proposed HDR-based SLAM system.



(a) (b)

Fig. 3. FAST feature detection on **(a)** color images; and **(b)** normalized radiance maps.

C. Train the patch-descriptor

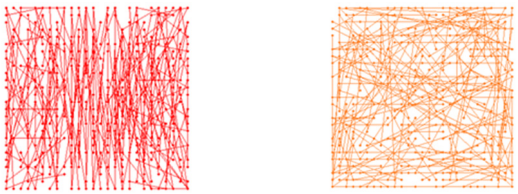
HDRFusion is based on direct tacking method, which directly optimizes the geometry by minimizing photometric errors using all information in the normalized radiance map. In comparison, the proposed HDR-based SLAM system is a featured-based system. First, the ORB features [30] are extracted from RGB images. Then, camera trajectories are calculated by optimizing the projection errors between corresponding feature points and sparse representation of map is built from selected features.

The feature matching process is divided into three steps: First, detect keypoints in an image. Second, use feature vectors to descript regions around keypoints. Last, find the corresponding features by comparing similarities between descriptors. ORB is a combination of oriented FAST (oFAST) and rotated BRIEF (rBRIEF), it uses oFAST for keypoint detection and a rBRIEF as the descriptor.

FAST is the corner detector implemented by comparing intensities of the centered pixel with its surrounding circular pixels [31]. To detect FAST features on normalized radiance map, we make some minor modifications, including changing some class declarations and data types in the source code of OpenCV library [32]. Fig. 3 shows the FAST feature extraction results of float-format normalized radiance map.

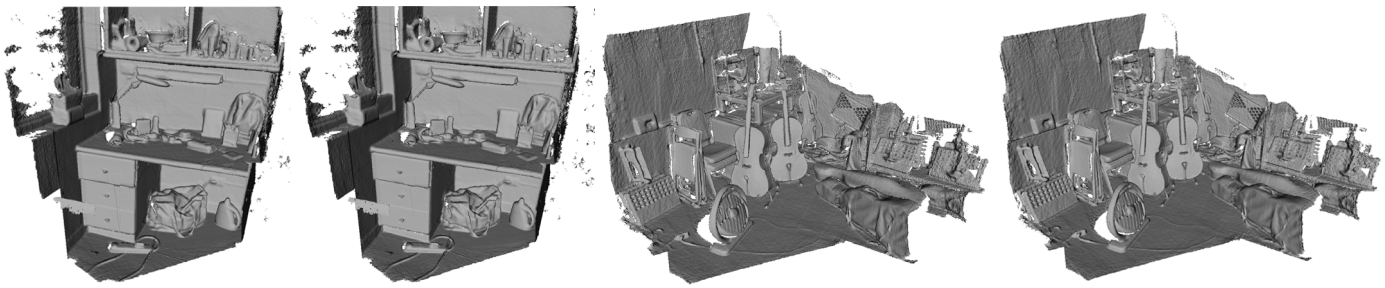
As a descriptor, rBRIEF encodes the information around a keypoint into binary strings. Then, the similarity of two descriptors is evaluated by calculating their hamming distance. If the distance is smaller than a given threshold, the two corresponding features are seen as highly-correlated and matched. To generate the binary descriptor, the patch which is centered at a keypoint and contains 256 pairs of points is introduced. For each pair, if the intensity value of the first point in the pair is larger than the second point, the descriptor value would be '1', otherwise be '0'. After that, we get a 256 binary string to describe the keypoint.

In rBRIEF, the patch is trained by 300k keypoints in the PASCAL 2006 dataset. The training process is designed to learn 256 pairs from about 200k possible pairs, and ensures that they have the following two properties, uncorrelation and high variance [30]. Uncorrelation means that the difference between each pair should be as large as possible, thus maximizing the amount of information 256 pairs carries. High variance makes



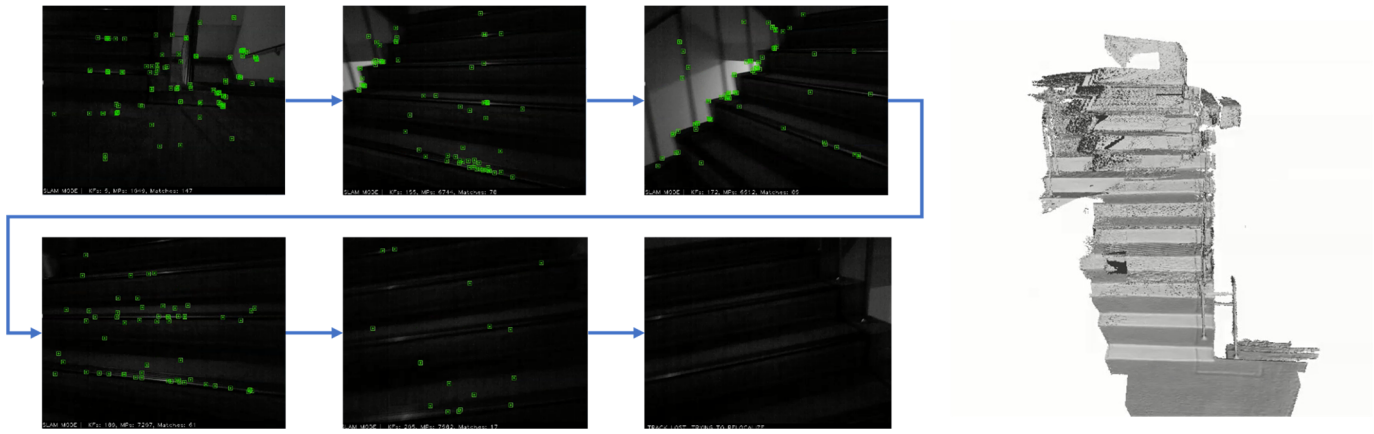
(a) (b)

Fig. 4. The descriptor patch trained by keypoints extracted from **(a)** color images; and **(b)** normalized radiance maps.



(a) (b) (c) (d)

Fig. 5. The geometric reconstruction results: **(a)** 'desk' using color images; **(b)** 'desk' using color images; **(c)** 'room' using normalized radiance maps; and **(d)** 'room' using normalized radiance maps.



(a) (b)

Fig. 6. For dataset 'stair': **(a)** illustration of losing tracking using color images; **(b)** reconstruction result with camera trajectory calculated from proposed HDR-based SLAM.

a feature to be more discriminative, so it can respond differently to different keypoints. Because the distribution of pairs in intensity image is supposed to be different from the one in normalized radiance map, we retrain the patch especially for normalized radiance map. First, the raw HDR images are collected online to generate normalized dataset. Then, about 200k FAST features are detected in these normalized radiance maps. Last, the learning process is re-implemented based greedy algorithm following instructions in [30]. The descriptor patch trained by 200k HDR-keypoints is shown in Fig. 4.

IV. Experimental Results

In our experiments, a small-scale dataset 'desk', a large-scale dataset 'room', and a low-light dataset 'stair' recorded by ourselves are tested. Because CRF is dependent on specific camera sensor, it is required to do the calibration first for each depth camera. The lack of calibrated CRF of public datasets is the reason why we are not able to use the well-known TUM RGB-D datasets [33] with ground-truth trajectories to evaluate the HDR-based SLAM. Therefore, we record the datasets by ourselves with calibrated Asus Xtion and do the qualitative evaluation on geometric reconstructions.

The experimental results are shown in Fig. 5. For 'desk' and 'room' dataset, the reconstructed results of normalized radiance map are as good as the ones from color images. As shown in Fig. 6, for the low-light dataset 'stair', the original ORB-SLAM2 loses tracking at the beginning (234^{th} frame) because of insufficient keypoints in RGB images. However, the proposed HDR-based SLAM is able to finish tracking total 2076 frames, and use the generated camera trajectory to complete the reconstruction.

V. Conclusions

The paper proposed a 3D reconstruction system which uses the HDR-based SLAM to calculate accurate trajectories in camera estimation step. We use normalized radiance map as input of HDR-based SLAM to generate more representative features comparing to features extracted from color images, and to eliminate the influence of changing exposure time. To design the feature matching process especially for normalized HDR inputs, the descriptor patch has been re-trained using 200k HDR-keypoints. The experimental results show that our proposed method can achieve good reconstruction performance of both small-scale and large-scale datasets, and can successfully achieve accurate camera tracking and geometric reconstruction under low-light environment.

ACKNOWLEDGMENT

This work was supported in part by Ministry of Science and Technology, Taiwan, under the Grants MOST 108-2218-E-003-002-, MOST 108-2218-E-110-002-, MOST 105-2221-E-110-094-MY3 and MOST 106-2221-E-110-083-MY2.

REFERENCES

[1] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges and A. Fitzgibbon, "KinectFusion: real-time dense surface mapping and tracking," *in Proceedings of IEEE International Symposium Mixed Augmented Reality (ISMAR)*, pp. 127-136, 2011.

[2] Q. Y. Zhou and V. Koltun, "Dense scene reconstruction with points of interest," *ACM Transactions on Graphics (ToG)*, vol. 32, no. 4, 2013.

[3] S. Choi, Q. Y. Zhou and V. Koltun, "Robust reconstruction of indoor scenes", *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5556-5565, 2015.

[4] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi and C. Theobalt, "Bundlefusion: real-time globally consistent 3d reconstruction using on-the-fly surface reintegration," *ACM Transactions on Graphics (ToG)*, vol. 36, no.4, 2017.

[5] G. Tiwari and P. Rani, "A review on high-dynamic-range imaging with its technique," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol.8, no. 9, pp. 93-100, 2015.

[6] A. El Gamal, "High dynamic range image sensors," *in Tutorial at International Solid-State Circuits Conference*, vol. 290, 2002.

[7] G. Wan, X. Li, G. Agranov, M. Levoy and M. Horowitz, "CMOS image sensors with multi-bucket pixels for computational photography," *IEEE Journal of Solid-State Circuits*, vol. 47, no.4, pp. 1031-1042, 2012.

[8] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," *in ACM SIGGRAPH 2008 classes*, p.31, 2008.

[9] T. Mertens, J. Kautz, and F. Van Reeth, "Exposure fusion: a simple and practical alternative to high dynamic range photography," *in Computer graphics forum*, vol. 28, no. 1, pp. 161-171, 2009.

[10] M. A. Robertson, S. Borman, and R. L. Stevenson, "Dynamic range improvement through multiple exposures," *in Proceedings of International Conference on Image Processing*, vol. 3, pp. 159-163, 2019.

[11] K, Ma, H. Li, H. Yong, Z. Wang, D. Meng and L. Zhang, " Robust multi-exposure image fusion: a structural patch decomposition approach, " *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2519-2532, 2017.

[12] M. Granados, K. I. Kim, J. Tompkin and C. Theobalt, "Automatic noise modeling for ghost-free HDR reconstruction," *ACM Transactions on Graphics*, vol. 32, no. 6, 2013.

[13] J. Im, J. Jeon, M. H. Hayes and J. Paik, "Single image-based ghost-free high dynamic range imaging using local histogram stretching and spatially-adaptive denoising," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 4, pp. 1478-1484, 2011.

[14] A. K. Johnson and C.V. Jiji, "Single shot high dynamic range imaging using histogram separation and exposure fusion," *in Proceedings of National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics*, Dec. 2015.

[15] G. Eilertsen, J. Kronander, G. Denes, R. K. Mantiuk and J. Unger, "HDR image reconstruction from a single exposure using deep CNNs," *ACM Transactions on Graphics*, vol. 36, no.6 , 2017.

[16] J. Cai, S. Gu, and L. Zhang, "Learning a deep single image contrast enhancer from multi-exposure images," *IEEE Transactions on Image Processing*, vol.27, no. 4, pp. 2049-2062, 2018

[17] M. Meilland, C. Barat, and A. Comport, "3D high dynamic range dense visual slam and its application to real-time object re-lighting, " *in Proceedings of IEEE International Symposium on Mixed and Augmented Reality*, pp. 143-152, 2013

[18] S. Li, A. Handa, Y. Zhang and A. Calway, "HDRFusion: HDR SLAM using a low-cost auto-exposure RGB-D sensor," *in Proceedings of International Conference on 3D Vision (3DV)*, pp. 314-322, 2016.

[19] C. Barat and A. I. Comport, "Active high dynamic range mapping for dense visual SLAM," *in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6514-6519, 2017.

[20] S. V. Alexandrov, J. Prankl, M. Zillich and M. Vincze, "High dynamic range SLAM with map-aware exposure time control," *in Proceedings of International Conference on 3D Vision (3DV),* pp. 48-56, 2017.

[21] T. M. Pinho, J. P. Coelho, J. Oliveira and J. Boaventura-Cunha, "Comparative analysis between LDR and HDR images for automatic fruit recognition and counting," *Journal of Sensors*, 2017.

[22] X. Yang, K. Xu, Y. Song, Q. Zhang, X. Wei and R. W. Lau, " Image correction via deep reciprocating HDR transformation," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1798-1807, 2018

[23] K. Yousif, A. Bab-Hadiashar and R. Hoseinnezhad, "An overview to visual odometry and visual SLAM: applications to mobile robotics, " *Intelligent Industrial Systems*, vol. 1, no. 4, pp. 289-311, 2015.

[24] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255-1262, 2017.

[25] B. Curless, B.and M. Levoy, "A volumetric method for building complex models from range images", 1996.

[26] W. E. Lorensen and H. E. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," *ACM Siggraph Computer Graphics*, vol. 21, no. 4, pp. 163-169, 1987.

[27] M. Waechter, N. Moehrle and M. Goesele, "Let there be color! – large-scale texturing of 3D reconstructions, " *in Proceedings of European Conference on Computer Vision*, pp. 836-850, 2014.

[28] Q. Y. Zhou and V. Koltun, "Color map optimization for 3D reconstruction with consumer depth cameras," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, 2014.

[29] C. Liu, W. T. Freeman, R. Szeliski and S. B. Kang, "Noise estimation from a single image," *in Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 901-908, 2006.

[30] E. Rublee, V. Rabaud, K. Konolige and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," *IEEE International Conference on Computer Vision*, 2011.

[31] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," *in Proceedings of European Conference on Computer Vision*, pp. 430-443, 2006.

[32] OpenCV, https://opencv.org/.

[33] J. Sturm, N. Engelhard, F. Endres, W. Burgard and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," *in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 573-580, 2012.