2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Continuous Point Cloud Stitch based on Image Feature Matching Constraint and Score*

Fangchao Hu, Yinguo Li, Wei Huang and Zhen Tian

Abstract—**The purpose of environment perception for autonomous vehicle is to get the entire information around the ego-vehicle and understand the environment where the autonomous vehicle runs. In this paper, we attempt to rebuild a 3D environment for the autonomous vehicle using the vehicle mounted cameras. First, we generate the continuous 3D point clouds with the vehicle moving. Then we stitch the continuous point clouds by using the image feature matching constraint and scores. At last, we obtained an accuracy and efficiency point cloud of the environment. After the numerical experiments, the proposed stitching algorithm has been verified that the MSE is lower than the average of other algorithms. And the number of points decreases than the conventional algorithm.**

I. INTRODUCTION

An autonomous vehicle requires understanding of the environment to behave correctly, so it is necessary to obtain the entire scene around the autonomous vehicle. In most case, the cameras which are mounted on the autonomous vehicle are used to percept environment. The field angle of mounted camera is smaller than 145 degrees, the picture cannot contain entire scene when the camera take each shot. The way to get the wide-angle scene around the autonomous vehicle is to capture lots of pictures when the vehicle moving and then to stitch the pictures from the vehicle mounted camera. The algorithms of image stitching can be utilized to obtain the whole scene, like the panoramagram, to help the autonomous vehicle to plan a correct and safe path. The algorithms of images stitching have been widely studied and many meritorious methodologies have been proposed. To obtain the entire scene around autonomous vehicle, the mainstream vehicle manufactures deployed the image stitching to get a mosaicked image by minimizing seam. Most of the algorithms of 2D images stitching have achieved the performance that the images are stitched perfectly [1, 2]. However, in the application of environment perception of autonomous vehicle, the mosaicked 2D image cannot satisfied the requirement of perception. The distances between the objects and ego-vehicle are essential for control stage, thus stitched 3D point cloud which contained the location information of objects may satisfy the requirement. But stitching 3D point clouds is full of challenging problems which focus on the accuracy and instantaneity. In this paper, our main task is to improve the accuracy and speed of stitching 3D point clouds.

To obtain entire scene by using cameras, 3D point clouds stitching is an important step for environment perception. It not only rebuilds a 3D scene for autonomous vehicle to understand the environment, but also map the circumstance for autonomous vehicle to analyze the movement tendency of objects. The mainstream methods of 3D point cloud stitching are generally classified into two categories: the dense matching methods [3-5], feature matching methods [6-8]. Zheng, Hong, et al. [9] utilize both descriptor similarity and mutual spatial coherency of features existed in multiple frames to match these frames. They may lose the some of the matched features when they skip frames to speed up the stitch processing. Wang and Zhao, et al. [10] reduce the iteration times of ICP algorithm according to the weight which is corresponding to point in the point cloud. It will be trapped in to local optimum due to the reduced iteration times. Yang and Chang, et al. [11] divide the large site into different scenes, and then cluster the images into sets of individual scenes. The size of division is indeterminate, so the performance is fluctuant.

In this work, the pictures of scene are captured by the vehicle-mounted cameras. There are several problems when we reconstruct the 3D environment. First, the objects in this scene are large, relate to the objects in the indoor scene. Many existed algorithms for indoor scene cannot be deployed in this work. Second, the objects are easy to be confused with background. That make the features detection on the objects and scene reconstruction become difficult. Third, the processing time of stitching is long. Moreover, there are many points in the overlap area between two adjust point cloud, which are repeated. To address the balance of above-mentioned problems, a feature matching score-based method is presented to conquer the problems of mismatch and redundant using feature matching score. Figure 1 shows the workflow from images acquisition to 3D point cloud stitching.

The main contributions of this work are twofold. First, the proposed algorithm cut down the accumulate errors by deleting the point with low matching score in the overlap area.

* This paper is supported by Chongqing Science and Technology Commission (cstc2017zdcy-zdyfX0004), Ph.D.'s high-end talent program (Grant no. BYJS2016001), the China Scholarship Council (Grant no. 201608500103), Scientific and Technological Innovation of Chongqing (Grant No. CYS17233), Chongqing Municipal Education Commission (Grant No. KJ1500405).

Fangchao Hu is with the Chongqing University of Posts and Telecommunications, Chongqing, CQ 400065 P.R.China (corresponding author to provide phone: +86-15823981211; e-mail: fangchaohu1211@126.com).

Yinguo Li is with Chongqing University of Posts and Telecommunications, Chongqing, 400065 P.R.China. (e-mail: liyg@cqupt.edu.cn).

Wei Huang is with the Computer Science and Technology Department, Chongqing University of Posts and Telecommunications, Chongqing, CQ 400065 P.R.China (e-mail: cqhuang_w@163.com).

Zhen Tian is with the Computer Science and Technology Department, Chongqing University of Posts and Telecommunications, Chongqing, CQ 400065 P.R.China (e-mail: tianzmail@gmail.com).

Second, it reduces the number of points in stitched point cloud effectively and decreases the occupation of store space. That is useful for autonomous vehicle to plan the path and for ADAS (Advantage Driving Assistant System) to reconstruct the 3D environment rapidly.

II. Related Work

As shown in figure 1, the first step of this flow is images acquisition, then generate the point clouds using the matched feature points. And the last step is point clouds stitching between adjacent frames. Each step has many algorithms which have different characters, in the following subsections they will be explained in detail.

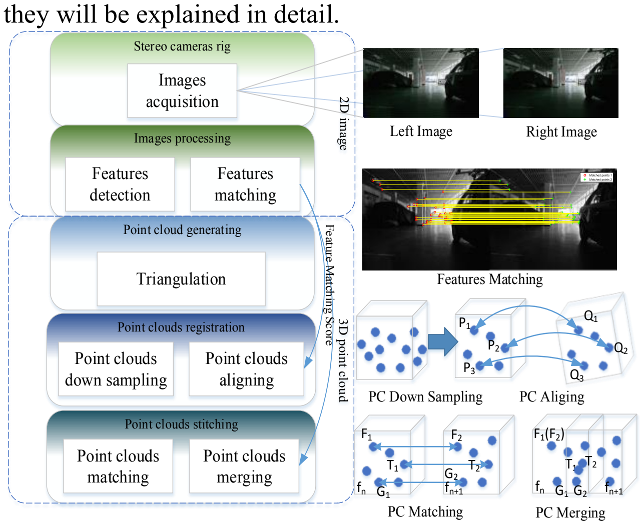

Fig. 1. Workflow from 2D image acquisition to 3D point clouds stitching.

A. Point cloud generation

In this work, the main task is stitching the continuous point clouds which describe the circumstance around autonomous vehicle. According to the different type of device, the mainstream methods of point cloud generation are classified into two categories [12]: the radar/lidar-based methods [13] and the triangularization-based methods [14]. In this work we chose the triangularization-based methods, because we plan to complete this 3D reconstruction task with a real-time scheme, the less points may help to reduce the computational burden.

Triangularization-based methods worked on the detected features of images. And feature detection of image is a proven technique, there are many algorithms of feature detection in different forms. The scale-invariant feature transform (SIFT) has been employed to find feature points in the original image and the Gaussian smoothed image [15]. Feature from Accelerated Segment Test (FAST) features was presented by Rosten [16], and they are acknowledged the rapid of feature extracting. Minimum eigenvalue algorithm [17] is utilized to detect corners, it can remove erroneous corners when the data sets are large. In this paper, we use the feature matching score to stitch the point cloud, so we compared many well-studied features to find the highest score in the following section. For real-time realization, FAST [16] or ORB (ORiented Brief) [18] algorithm can obtain the feature point of correspondence quickly between the images. After finding the matched points, the 3D point cloud will be produced by using triangularization. The triangularization is the geometric method to calculate the location of point in real world from more than two views of the same feature.

m_1, m_2 are the projection of M in the left and right image respectively. The location of M can be calculated by the formula as follow:

$$\rho_1 m_1 = K_1 R_1^T (M - C_1) \tag{1}$$

$$\rho_2 m_2 = K_2 R_2^T (M - C_2) \tag{2}$$

ρ_1, ρ_2 are the scale parameters, K_1, K_2, R_1, R_2 are the cameras parameters which can be calculated by calibrating. C_1, C_2 are the location of cameras in real world. The results are the essential data for 3D point cloud stitching.

B. Point cloud stitching

Point cloud is composed of millions of located points, many pieces of continuous point cloud could represent the entire scene. Before stitching the candidate point cloud, the point cloud need to be registered. Point cloud registration is the step that ensures the location of reference point cloud which is essential for point clouds to merge. After point cloud registering, the follow-up point cloud could be added to reference point cloud piecewise. Finally, the mosaicked point cloud is gained.

We investigated two algorithms for matching pairs of acquired 3D point clouds: ICP (Iteration Closest Points) [2], [4] and NDT (Normal Distributions Transform) [6]. ICP is the actual standard 3D registration algorithm which is used in the 3D stitching community, and NDT has appeared to be a compelling alternative in previous comparisons. The purpose of ICP is minimizing the equation (3)

$$E(R,t) = \sum_{i=1}^{N_m} \sum_{j=1}^{N_d} \omega_{i,j} \left\| m_i - (Rd_j + t) \right\|^2 \tag{3}$$

Where N_m and N_d are the number of points in the model set M and data set D, respectively. $\omega_{i,j}$ are the weights for a point match.

After the step of registration, the follow-up point cloud added to reference point cloud, the rigid transformation matrix need to be calculated. Rigid transformation is the most critical step for point cloud stitching, because in this step, the transformation may bring the error into the mosaicked point cloud. So, in order to bring less error, we need to optimize the algorithms which obtain the transformation matrix. After calculating the transformation matrix, the adjacent point cloud can be stitched in a uniform coordinate system. There are many methods to stitch the point clouds, but the traditional method is to add the current point cloud to the reference point cloud directly. It may not only increase the redundancy but also leading to the estimation error. In the step of stitching, we proposed a new framework to avoid the aforementioned problem.

III. Proposed Framework

In this paper, the demand of point cloud stitching is to obtain the multi-view information from the stereo cameras rig mounted on the moving vehicle. First of all, we capture the image pairs synchronously using the stereo cameras rig, then the SURF detector is used to detect the features of image pairs, SURF descriptors are utilized to match between the image pairs which capture synchronously in the same scene. The single view scene is reconstructed by the correspondent features within two images. With the vehicle moving we can get the multi-frame point clouds. The multi-view scenes are reconstructed by multi-frame point clouds. Due to the moving vehicle, the scene from camera view is changing, so in order to percept the changing environment, the point clouds stitching algorithm based on match score is presented. This score sways the density and accuracy of the stitched point clouds. That means if the former point cloud performs well during matching with the latter cloud, it will get a high score, if the former point cloud performs poorly during matching instead, and it will get a low score. Each of the candidate point clouds will get a score, then each score is normalized as the stitching weight while the point clouds stitching. With the features moving between the adjacent frames, we track the features to obtain transformation matrix like that in equation (3). We find the minimum reprojection error, as shown in equation (4).

$$T_{k,k-1} = \arg\min_T \sum_i \left\| u_i' - \pi(p_i) \right\|_\Sigma^2 \tag{4}$$

Wherein the $T_{k,k-1}$ is the transformation matrix from frame $k-1$ to k, u_i' is the feature point in frame k (actual value). The $\pi(p_i)$ is the reprojection point of p_i in frame k (theoretical value). We consider this constrain, we modified the objective function as follow to obtain optimized transformation matrix.

$$E(R,t) = \sum_{i=1}^{N_m}\sum_{j=1}^{N_d} \left\| m_i - (Rd_j + t) \right\|^2 + \sum_i \left\| u_i' - K \times [R \mid t] \times p_i \right\|_\Sigma^2 \tag{5}$$

Wherein the K is the intrinsic parameter of camera. R and t are the rotation matrix and translation vector between frame i and frame j, respectively. m_i is a point from model set, and d_j is a point from data set. The N_m, N_d is the number of model set and data set, respectively. The combined constrain not only reduce the mismatching but also improve the calculate speed, due to the limited feature pairs.

Features detecting and matching the image pairs are the essential issue of scores. We use the SURF feature as the gist to match between the image pair. Assuming that the numbers of features of each image pair are C_{n1}, C_{n2} $(n = 1...N)$, respectively, wherein n is the sequence number of frame. Then the number of matched features is C_m, the score of each image pair is gained as followed:

$$score = \frac{C_m}{\sum_n \min(C_{n1}, C_{n2})} (m, n = 1, 2, ..., N) \tag{6}$$

In other words, the scores equal to the sum of the minimum between C_{n1} and C_{n2} of each frame divided by the number of matched features C_m of corresponding frame. And then the scores will be normalized into the weight:

$$weight = \frac{score_n}{\sum_n (score_n + score_{n+1})} (n = 1, 2, ..., N-1) \tag{7}$$

The weight is gained by the two adjacent frames scores, theoretically, the overlap merely occurs at adjacent frames. Therefore, we should only compute the weight between two adjacent frames. In figure 2, the feature points n_1, n_1' are matched in the adjacent frames $image_1$, $image_1'$, the matching score is calculated according to the equation (6). Then the weight is gained according to the equation (7). o_1, o_1' are linked with red dashed line which indicate the mismatch points, between the adjacent frames $image_1$, $image_1'$. The matching score can be calculated by these matched feature points, and then the weights between the adjacent frames can be obtained.

The point clouds are stitched by the score-based algorithm, as algorithm 1 stated. The point clouds are multiplied by a weight which is produced by the scores of adjacent frames. It makes the mosaicked point clouds more precise, because in the existent algorithm the points which are in overlap region will be added up directly. But in the presented algorithm, the points will be multiplied by the weight, then the points in the overlap will be reduced correspondingly. It can guarantee the accuracy of matching.

Algorithm 1. Matching Score based Point cloud stitch algorithm

```
Input: Continuous Point Cloud P_n(0 < n < N)
Output: Stitched Point Cloud {C_m} (0 < m < M)
IF P_n(0 < n < N) ∈ moving point cloud
   IF P_n(0 < n < N) ∉ limit region
      The P_n was added to the point cloud C
      -> P_n + ∑{C_N}
    ELSE
      Calculate the weight according to the match score->
 weight_α
      Then the weight multiplied by P_n -> weight_α · P_n
      The weighted P_n was added to the point cloud C
      -> weight_α · P_n + ∑{C_N}
   IF END
ELSE
      Cast off this point
IF END
```

The weight of adjacent frames satisfied the following.

$$N_A + N_B = N_C \tag{8}$$

N_A, N_B are the number of points in the adjacent point clouds, N_C is the sum of points in the two adjacent point clouds.

$$weight_\alpha \cdot N_A + weight_\beta \cdot N_B = 1/2 N_C \tag{9}$$

Equation (8) into Equation (9), we can get.

$$weight_\alpha = \frac{(\frac{1}{2} - weight_\beta) N_B}{N_A} + \frac{1}{2} \tag{10}$$

From the equation (6) we can get.

$$weight_\alpha + weight_\beta = 1 \tag{11}$$

$weight_\alpha$, $weight_\beta$ are the weight calculated by the matching scores between the two adjacent point clouds. The points which are deleted in the overlap are taken away under algorithm 2, as shown in figure 3. The hollow points will be removed while stitching, which can make the mosaic point cloud more accurate and efficient.

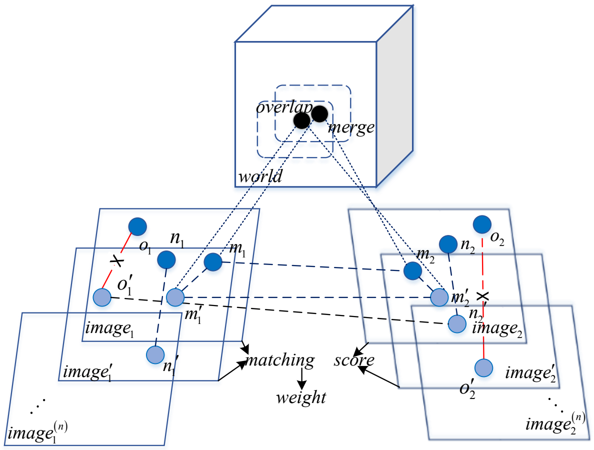

Fig. 2. Principle of the matching score.

For algorithm 2 the centroid point $P_f(x_f^c, y_f^c, z_f^c)$ of current frame f is given. The distance from each point to the centroid point is calculated d_f^n . When the d_f^n greater than the threshold ε , this point P_f^n whose distance is d_f^n will be removed. Correspondingly, the point P_{f-1}^{N-n} in frame $f-1$ whose distance order is $N-n$ will be removed. The value of threshold ε is determined according to the n^{th} nearest distance between the point P_f^n and centroid point P_f .

Algorithm 2. End to end reduction algorithm

Input: Stitched Point Cloud $\{C_m\}$ $(0 < m < M)$

Output: Refined Point Cloud $\{C_m{'}\}$ $(0 < m < M')$

Given centroid point of Current Frame $P_f(x_f^c, y_f^c, z_f^c)$

Calculate the distance between each point and centroid point $d_f^n (f = 1, 2, \ldots F, n = 1, 2, \ldots, N)$

For $f = 1$ **to** F {

For $n = 1$ **to** N {

$$d_f^n = \sqrt{(x_f^n - x_f^c)^2 + (y_f^n - y_f^c)^2 + (z_f^n - z_f^c)^2}$$

IF $d_f^n > \varepsilon$ {

$$P_f^n = 0 \,,\quad P_{f-1}^{N-n} = 0$$

}**END IF**

}**END FOR**

}**END FOR**

As shown in figure 3, the black point is the centroid point of the two adjacent point cloud in the overlap region. The orange and blue points belong to frame $f-1$ and f , respectively. The orange point with the distance d_{f-1}^1 is the nearest point from the centroid point in frame $f-1$. And the point with the distance d_f^1 is the nearest point from the centroid point in frame f . Similarly, the point with distance d_{f-1}^{n-1} and d_f^n are the farthest one in frame $f-1$ and f ,respectively. The orange points are removed from the nearest distance, and the blue points are removed from the farthest distance.

The redundancy points will be reduced compare to the conventional algorithm. And the number of points is controlled according to the scores, which are relevant to the performance of features matching.

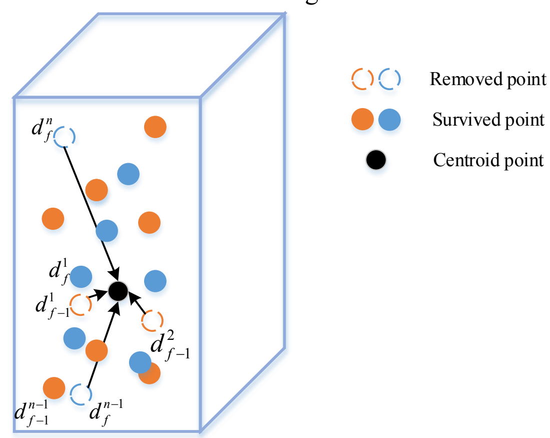

Fig. 3. Points reduction rule of end to end.

IV. Experiment

The image pairs are captured by the calibrated binocular camera which is built up by two monocular cameras (Basler ace-acA2500-60uc). The cameras are mounted on the vehicle, the optic center is 1.6m over the floor and the baseline is 1.2m. These images are acquired in the underground parking of the laboratory building in campus of Chongqing University of Posts and Telecommunications. The objects in these images are vehicles and pedestrians. The image size is 1920x1080. The image format is PNG. The number of the image pairs was 1385, and the size of the dataset was more than 2.9 GB. All methods in this paper were implemented using C++ and MATLAB 2016(a), they run on an Intel Core i7-4700 CPU@2.4GHz with 8G RAM.

TABLE I. SCORES OF DIFFERENT ALGORITHMS OF FEATURE DETECTION WITH IMAGE PAIR SEQUENCE

Feature detection	*Minimum Number of Features*	*Number of Matched*	*Score*	*Weight*
SURF	5611	**1332**	**0.2374**	**0.3048**
FAST	5379	660	0.1227	0.1576
MinEigen	**14046**	948	0.0675	0.0867
Harris	4413	540	0.1224	0.1572
BRISK	3048	109	0.0358	0.0460
MSER	4326	835	0.1930	0.2478

TABLE II. TIME COSTING IN DIFFERENT ALGORITHM OF STITCHING

Algorithm of point cloud stitch	*Time Cost of Computing (s)*
ICP	0.794970
L-M ICP	0.936874
NDT	1.200785
Present method	0.661914

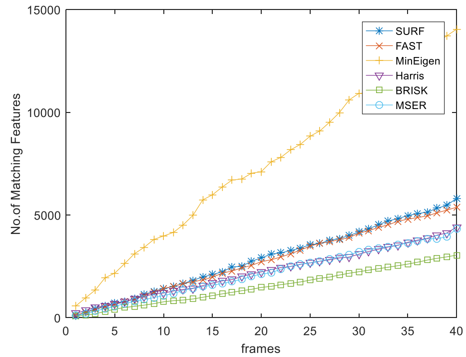

Fig. 4. Number of Matching Features with different methods.

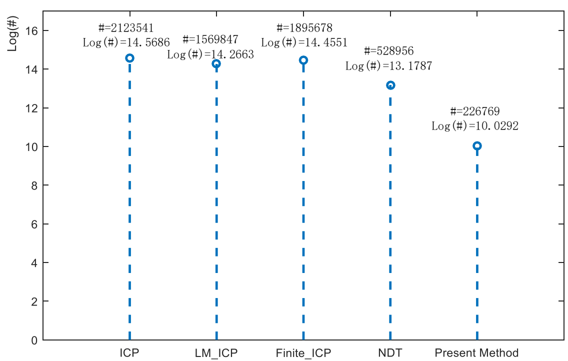

Fig. 5. Total number of stitched point cloud with different algorithm.

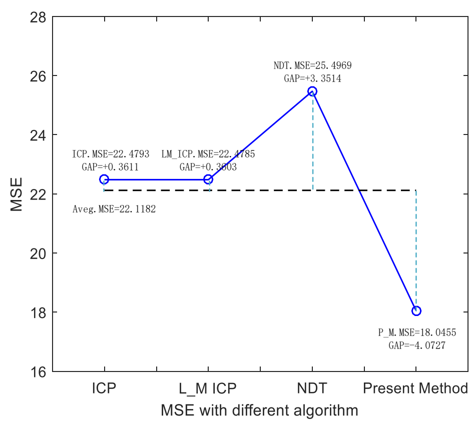

Fig. 6. MSE with different algorithm and gap between the Average MSE.

In order to get the accurate point cloud reconstruction, meanwhile to get higher score of matching, it required more number of matching features from image pairs. Several frequently-used methods of feature detection and matching are experimented to find the highest match score. As shown in Table 1, we deployed 6 type of feature detection algorithm to get the minimum number of features between one frames of image pair in second column. This number guarantee the range of matched features between corresponding image pairs in the third column. In the fourth column the scores are calculated according to the proposed algorithm. At last column the corresponding weight of each frame will be gained according to the match scores individually. The Table 1 indicates that in second column the bolded number are the highest number. In other word, among the six algorithms the MinEigen gets the maximum number of features in the same image pair. Similarly, MinEigen gets the maximum number of matched features. But SURF gets the maximum score and weight after calculation.

From Table 2, we can find that the time cost of ICP is low, but it is at the expense of bringing in mismatch. The proposed algorithm considered the problem of mismatch in overlap region. It not only makes the stitching of point cloud more accurate, but also lets the reconstructed scene more suitable for autonomous vehicle application. Besides, the time cost is improved compared to NDT and other algorithms. And the performance offset the deficiency of time cost.

The number of matching features with different feature detectors are shown in Figure 4. As we hope to get the feature points and matched feature points as many as possible, the Figure 4 indicate although the MinEigen produce maximum of feature points. SURF has a highest ratio in matched feature points. In other words, the scene can be represented in better perform with the same input images by using SURF. Therefore, we deployed SURF to detect image feature in this paper.

The number of points in the point cloud with different stitching algorithms are shown in Figure 5. In Figure 5, we find that the presented method can decrease the number of points 89.32% than traditional ICP. From Figure 5, we can find that the present method gets the minimum number of points after stitching than that of other methods. In other words, the propose algorithm use the least points to stitch the same point clouds to get a similar performance. The repetitive points are removed efficiently that makes the point cloud can be store and display more quickly in the subsequent procedure in autonomous vehicle technology.

In order to verify the accuracy of the proposed algorithm, we set a scene which we use the instrument to measure the rotation. And then we use the experimental camera rigs to acquire the image pairs. We use this data as the ground truth to compare the MSE (Mean Square Error) with different algorithms. The Figure 6 shows that the proposed algorithm has the lowest MSE, the MSE of other methods are approximate. The MSE of presented method is 23.13% lower than the average of the rest of three methods.

V. Conclusion

According to experimental data and figures, we find that the proposed method can stitch the point cloud without obvious seam visually. It achieved the goal of stitching sufficiently, which can reconstruct the scene using point cloud. Moreover, the proposed method has cut large number of unnecessary points in the overlap. It reduces the number of points in mosaic point cloud effectively, and decreases the occupation of store space. The accumulate errors have been cut down due to the deletion of the point cloud with low matching score.

In this paper, a new stitching algorithm of 3D point cloud are proposed for autonomous vehicle to reconstruct the 3D environment. The proposed matching score averted the mismatch of feature, and the reduction rule guaranteed the redundant points in overlap region of point cloud can be removed. It has achieved the goal of stitching sufficiently, which can reconstruct the scene using point cloud. Moreover, the proposed method has cut large number of unnecessary points in the overlap. It reduces the number of points in mosaic point cloud effectively, and decreases the occupation of store space. The accumulated errors have been cut down due to the deletion of the point cloud with low matching score. Furthermore, the algorithm can be used to stitch the LIDAR generated 3D point cloud of the outdoor scene and the Kinect generated 3D point cloud which are often occurred in indoor scene.

ACKNOWLEDGMENT

This paper is supported by National Natural Science Foundation (Grant no.51505054), Ph.D.'s high-end talent program (Grant no. BYJS2016001), the China Scholarship Council (Grant no. 201608500103), Scientific and Technological Innovation of Chongqing (Grant No. CYS17233), Chongqing Municipal Education Commission (Grant No. KJ1500405), Chongqing Science and Technology Commission (Grant No. cstc2015zdcy-ztzx60002, cstc2016jcyjA0538, cstc2015zdcy-ztzx30001).

REFERENCES

[1] Xiong, Yingen, and Kari Pulli. “Color Matching for High-Quality Panoramic Images on Mobile Phones.” *IEEE Transactions on Consumer Electronics* vol.56, no.4, 2010, pp. 2592–2600.

[2] Wang, Wei, and Michael K. Ng. “A Variational Method for Multiple-Image Blending.” *IEEE Transactions on Image Processing* vol.21, no.4, 2012, pp. 1809–1822.

[3] Ducrot A, Dumortier Y, Herlin I, et al. “Real-time quasi dense two-frames depth map for autonomous guided vehicles”. Intelligent Vehicles Symposium (IV), IEEE, 2011.pp. 497-503.

[4] Suhr, Jae Kyu, and Ho Gi Jung. "Noise-resilient road surface and free space estimation using dense stereo." *Intelligent Vehicles Symposium (IV),* IEEE, 2013, pp.461-466.

[5] Lefebvre, Sébastien, and Sébastien Ambellouis. "Vehicle detection and tracking using mean shift segmentation on semi-dense disparity maps." *Intelligent Vehicles Symposium (IV),* IEEE, 2012, pp. 855-860.

[6] Xu, Yuquan, et al. "3D point cloud map based vehicle localization using stereo camera." *Intelligent Vehicles Symposium (IV)*. IEEE, 2017. pp. 487-492.

[7] Cao, Mingwei, et al. "Robust bundle adjustment for large-scale structure from motion." *Multimedia Tools and Applications* 2017, pp. 1-25.

[8] He, Ying, et al. "An Iterative Closest Points Algorithm for Registration of 3D Laser Scanner Point Clouds with Geometric Features." *Sensors vol.*17 no.8, 2017, pp. 1862-1872.

[9] Zheng, Shuai, et al. "A multi-frame graph matching algorithm for low-bandwidth RGB-D SLAM." *Computer-Aided Design* vol.7, no.8, 2016, pp. 107-117.

[10] Wang, Xin, et al. "An iterative closest point approach for the registration of volumetric human retina image data obtained by optical coherence tomography." *Multimedia Tools and Applications* vol.76 no.5, 2017, pp. 6843-6857.

[11] Yang, Yueming, et al. "Efficient large-scale photometric reconstruction using Divide-Recon-Fuse 3D Structure from Motion." Advanced Video and Signal Based Surveillance (AVSS), *2016 13th IEEE International Conference on.* IEEE, 2016.pp.180-186

[12] Liang, Bin, and Lihong Zheng. "Specificity and Latent Correlation Learning for Action Recognition Using Synthetic Multi-View Data From Depth Maps." *IEEE Transactions on Image Processing*, vol.2 6 no.12, 2017, pp. 5560-5574.

[13] Roussel, Jean-Romain, et al. "Removing bias from LiDAR-based estimates of canopy height: Accounting for the effects of pulse density and footprint size." *Remote Sensing of Environment* vo.198, 2017, pp.1-16.

[14] Gabara, Grzegorz, and Piotr Sawicki. "Accuracy Study of Close Range 3D Object Reconstruction Based on Point Clouds." *Geodetic Congress (BGC Geomatics), 2017 Baltic. IEEE*, 2017.pp.25-29.

[15] Li, Yuanman, et al. "SIFT keypoint removal and injection via convex relaxation." *IEEE Transactions on Information Forensics and Security* vol.11, no.8, 2016, pp. 1722-1735.

[16] Li, Jianan, Tingfa Xu, and Kun Zhang. "Real-time feature-based video stabilization on FPGA." *IEEE Transactions on Circuits and Systems for Video Technology* vol.27 no.4, 2017, pp. 907-919.

[17] Yu, Liang, et al. "Acoustical source reconstruction from non-synchronous sequential measurements by Fast Iterative Shrinkage Thresholding Algorithm." *Journal of Sound and Vibration* , 2017. 351-367.

[18] Rublee, Ethan, et al. "ORB: An efficient alternative to SIFT or SURF." Computer Vision (ICCV), *2011 IEEE international conference on. IEEE, 2011*.pp.2564-2571.