Robust Scene Reconstruction by Combing RGB-D Scans and Photographs

Anonymous CVPR submission

Paper ID ****

Abstract

1. Introduction

Photo-realistic geometry reconstruction systems typically fall into two categories, namely, those that are based on RGB-D and LIDAR sensors and those that are RGB image streams. Although these two category of methods have enjoyed tremendous progress during the past decades, they still exhibit significant limitations.

RGB-D based reconstruction systems tend to drift due to limited field-of-view, and they do not capture all the material. The color information is usually not very good.

RGB based reconstruction systems have low-range of field, and tend to produce reconstructions that are less drifting. However, it does not produce dense point clouds, and correspondences can be wrong.

We propose to combine both streams, by solving a joint optimization problem. The key idea is to establish high-quality correspondences between both streams.

2. Related Works

One paper is this [2].

Survey on shape reconstruction [3]

Joint analysis of Image and Shapes [1]

Need to talk about the comparison to circular scanners used by MagicLeap.

3. Problem Statement and Approach Overview

Problem Statement.
Approach Overview.

4. Approach

In this section, we introduce the technical details of our approach.

4.1. Joint Geometry and Texture Reconstruction

Our reconstruction aligns the reconstruction with the input RGB-D scans and the RGB images for reconstruction. Specifcally, we represent Consider a point cloud P with position, normal and semantic descriptors. Our goal is to optimize a rigid transform T_i for each depth scan S_i and a camera pose P_i for each image I_j to minimize the alignment error. To make it robust, we introduce a robust norm to handle noise in the geometric alignment.

$$\min_{\{T_i\},\{C_j\}} \quad \sum_{i=1}^{n_S} d^2(T_i(S_i), P) + \lambda \sum_{j=1}^{n_I} d^2(C_j(I_j), P) \quad (1)$$

Representation of latent surface. We represent a latent surface as a point cloud $\mathcal{P} = \{\boldsymbol{p}_i, 1 \leq i \leq N\}$, where each point $\boldsymbol{p}_i$ is associated with a optimized latent descriptor $\boldsymbol{d}_i$.

Alignment of the latent surface and the input scans

$$\sum_{j=1}^{n_i} \left(\left((R_i \boldsymbol{p}_{ij} + \boldsymbol{t}_i - \boldsymbol{q}_{k_{ij}})^T \boldsymbol{n}_{k_{ij}} \right)^2 + \mu \| \boldsymbol{d}_{ij} - \boldsymbol{f}_{k_{ij}} \|^2 \right) \quad (2)$$

Alignment of the latent surface and the RGB images

$$\sum_{j=1}^{n_i} \left(\| C_i(\boldsymbol{q}_{k_{ij}}) - \boldsymbol{p}_{ij} \|^2 + \mu \| \boldsymbol{d}_{ij} - \boldsymbol{f}_{k_{ij}} \|^2 \right) \quad (3)$$

4.2. Invariant Descriptor Learning

Our key idea is to learn dense pixel-wise descriptors to align RGB and RGB-D images for dense registration.

4.3. Project Plan

- Study how to perform image-to-shape alignment to predict camera poses, assuming that we have dense image descriptors.
- Study how to use multiple images to improve the quality of the shape.
- Integrate image-based reconstruction and RGB-D based reconstruction.
- Study how to learn invariant descriptors.

5. Experimental Evaluation

5.1. Experimental Setup

5.2. Analysis of Results

5.3. Ablation Study

6. Conclusions and Future Work

7. Idea/algorithm

Supposed we have a group of 3d point cloud and a group of images, due to the poor condition of 3d scans, 3d reconstruction with them has some degree of distortion. So we would like to use the image to align the 3d reconstruction so that we can offset this distortion.

7.1. Basic idea

Supposed we have two point clouds,$P1$ and $P2$, and a set of pictures $C = \{c_1, c_2, ..., c_n\}$,of which the positions have already marked, we want to find transpositions T_1 T_2 for two point clouds to minimize distance between two point clouds and distance between point clouds and pictures.

$$\min_{\{T\}} f_d(T_1(P_1), T_2(P_2)) + \sum_i^{n_C} g_d(T_2(P_2), c_i) + \sum_i^{n_C} g_d(T_1(P_1), c_i) \tag{4}$$

The loss function of two point clouds follows the loss function in ICP algorithm.

$$f_d(Q_1, Q_2) = \sum_{(i,j)\in\Omega} \|q_{1,i} - q_{2,j}\|_2^2 \tag{5}$$

where $q_{1,i} \in Q_1$, $q_{2,j} \in Q_2$,Ω is a nearest-neighbor matching index from Q_2 to Q_1.

As for the point-cloud-to-image loss, we would like to reflect the 3d point to the image and calculate the distance between the 3d points and the 2d points.

$$g_d(Q, C) = \sum_i^{n_{fc}} (\|A_C \begin{bmatrix} q_{\Lambda(C,i)} \\ 1 \end{bmatrix} - c_i\|_2^2) \tag{6}$$

where n_{fc} is the number of the feature in image C, $q_{\lambda(C,i)}$ means the corresponding points in Q to i^{th} feature points in image C and A_C is non-full rank 3-by-4 matrix that can reflect the 3d point to the image.

7.2. Problem simplify and Optimization

7.2.1 Problem simplify

Actually, we don't need to solve all T_i at the same time, we can suppose that Q_1 is the point cloud that has already matched, so the loss function will be much more simple.

$$\min_{\{T\}} f_d(Q_1, T_2(P_2)) + \sum_i^{n_C} g_d(T_2(P_2), c_i) \tag{7}$$

Put equation 5 and equation 6 into consideration.

$$\min_{\{T\}} \sum_{(i,j)\in\Omega} \|q_i - Rp_j - t\|_2^2 + \sum_i^{n_C} \sum_j^{n_{fc}} (\|A_i \begin{bmatrix} p_{\Lambda(C,j)} \\ 1 \end{bmatrix} - c_{i,j}\|_2^2) \tag{8}$$

7.2.2 Prepare

First, simplify the subscript of the formula.

$$\min_{\{T\}} \sum_i^{size(\Omega)} \|q_i - Rp_i - t\|_2^2 + \sum_j^{n_C * n_f} (\|A_j \begin{bmatrix} p'_j \\ 1 \end{bmatrix} - c_j\|_2^2) \tag{9}$$

Transposition can be devide into two steps.

$$T(p) = Rp + t \tag{10}$$

In the process below, we will use two nature of R.

$$RR^T = I \tag{11}$$

$$R \approx R_0(I + r\times) \tag{12}$$

where R_0 is the R before this iteration which will be I in first iteration, and r is the rotation angel in axis x, y and z respectively, r$\times$ will be a matrix of cross operation about r. We can alternate update parameter R and t. And we will update R ahead of t.

7.2.3 Optimize R

Consider the first item.

$$\begin{aligned} \|q_i - Rp_i - t\| &= (Rp_i + (t - q_i))^T (Rp_i + (t - q_i)) \\ &= p_i R^T R p_i^T - 2(t - q_i)^T R p_i + \|t - q_i\|_2^2 \\ &= const - 2(t - q_i)^T R p_i \\ &\approx const - 2(t - q_i)^T R_0 (I + r\times) p_i \\ &= const - 2(t - q_i)^T R_0 (r \times p_i) \\ &= const - 2r^T (t - q_i)^T R_0 \times p_i \end{aligned} \tag{13}$$

As for the second item, we first define $A_j = \begin{bmatrix} E & e \end{bmatrix}$, where E is a 3-by-3 matrix and e is a 3-by-1 matrix. And we

let $R_0^T E_j^T E_j R_0 = H_j$, where H_j obviously is a symmetry matrix.

$$\begin{aligned}
&\|A_j \begin{bmatrix} Rp_j + t \\ 1 \end{bmatrix} - c_j\| \\
=&\| \begin{bmatrix} E_j & e_j \end{bmatrix} \begin{bmatrix} Rp_j + t \\ 1 \end{bmatrix} - c_j\| \\
=&\|E_j R p_j + (E_j t + e_j - c_j)\| \\
=&(E_j R p_j + (E_j t + e_j - c_j))^T (E_j R p_j + (E_j t + e_j - c_j)) \\
=&p_j^T R^T E_j^T E_j R p_j + 2 p_j^T R^T E_j^T (E_j t + e_j - c_j) + \\
&\|(E_j t + e_j - c_j)\|_2^2 \\
=&p_j^T R^T E_j^T E_j R p_j + 2 p_j^T R^T E_j^T (E_j t + e_j - c_j) + const \\
=&p_j^T (I + r\times)^T R_0^T E_j^T E_j R_0 (I + r\times) p_j + \\
&2 p_j^T (I + r\times)^T R_0^T E_j^T (E_j t + e_j - c_j) + const \\
=&p_j^T H_j p_j + p_j^T (r\times)^T H_j p_j + \\
&p_j^T H_j (r\times) p_j + p_j^T (r\times)^T H_j (r\times) p_j + \\
&2 p_j^T (I + r\times)^T R_0^T E_j^T (E_j t + e_j - c_j) + const \\
=&2 p_j^T H_j (r\times) p_j + p_j^T (r\times)^T H_j (r\times) p_j + \\
&2 p_j^T (r\times)^T R_0^T E_j^T (E_j t + e_j - c_j) + const \\
=&2 r^T p_j^T H_j \times p_j - r^T p_j^T \times H_j \times p_j r - \\
&2 r^T p_j^T \times R_0^T E_j^T (E_j t + e_j - c_j) + const
\end{aligned} \tag{14}$$

Thus now, we can renew the loss function to r,throwing all constant items.let

$$\begin{aligned}
&(t - q_i)^T R_0 \times p_i = u_i \\
&p_j^T H_j \times p_j - p_j^T \times R_0^T E_j^T (E_j t + e_j - c_j) = v_j \\
&p_j^T \times H_j \times p_j = w_j \\
&U = \sum_i u_i \\
&V = \sum_j v_j \\
&W = \sum_j w_j
\end{aligned} \tag{15}$$

Then we have,

$$\begin{aligned}
\min_r &\sum_i (-2 r^T (t - q_i)^T R_0 \times p_i) + \\
&\sum_j (2 r^T p_j^T H_j \times p_j - r^T p_j^T \times H_j \times p_j r - \\
&2 r^T p_j^T \times R_0^T E_j^T (E_j t + e_j - c_j))
\end{aligned} \tag{16}$$

After simplify, we have

$$\min_r -2 r^T U + 2 r^T V - r^T W r \tag{17}$$

This a relative simple question, we can easily get

$$r^* = 2(W + W^T)^{-1}(V - U) \tag{18}$$

7.2.4 Optmize t

Consider the first items,

$$\begin{aligned}
\|q_i - R p_i - t\| &= p_i R^T R p_i^T - 2(t - q_i)^T R p_i + \|t - q_i\|_2^2 \\
&= const - 2 p_i^T R^T t + t^T t - 2 q_i^T t
\end{aligned} \tag{19}$$

As for the second items,

$$\begin{aligned}
&\|A_j \begin{bmatrix} Rp_j + t \\ 1 \end{bmatrix} - c_j\| \\
=&p_j^T R^T E_j^T E_j R p_j + 2 p_j^T R^T E_j^T (E_j t + e_j - c_j) + \\
&\|(E_j t + e_j - c_j)\|_2^2 \\
=&2 p_j^T R^T E_j^T E_j t + \|(E_j t + e_j - c_j)\|_2^2 + const \\
=&2 p_j^T R^T E_j^T E_j t + t^T E_j^T E_j t + 2(e_j - c_j)^T E_j t + const
\end{aligned} \tag{20}$$

Thus now, we can renew the loss function to t,throwing all constant items.let

$$\begin{aligned}
&-2 p_i^T R^T - 2 q_i^T = u_i' \\
&2 p_j^T R^T E_j^T E_j + 2(e_j - c_j)^T E_j = v_j' \\
&U' = \sum_i u_i' \\
&V' = \sum_j v_j' \\
&W' = \sum_i I + \sum_j E_j^T E_j
\end{aligned} \tag{21}$$

Then we have,

$$\begin{aligned}
\min_t &\sum_i (-2 p_i^T R^T t + t^T I t - 2 q_i^T t) + \\
&\sum_j (2 p_j^T R^T E_j^T E_j t + t^T E_j^T E_j t + 2(e_j - c_j)^T E_j t)
\end{aligned} \tag{22}$$

After simplify, we have

$$\min_t U t + V t + t^T W t \tag{23}$$

This a relative simple question, we can easily get

$$t^* = -(W + W^T)^{-1}(V + U)^T \tag{24}$$

References

[1] Moos Hueting, Maks Ovsjanikov, and Niloy J. Mitra. Crosslink: Joint understanding of image and 3d model collections through shape and camera pose variations. *ACM Trans. Graph.*, 34(6):233:1–233:13, Oct. 2015. 1

[2] Yangyan Li, Qian Zheng, Andrei Sharf, Daniel Cohen-Or, Baoquan Chen, and Niloy J. Mitra. 2d-3d fusion for layer decomposition of urban facades. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 882–889, Washington, DC, USA, 2011. IEEE Computer Society. 1

[3] P. Musialski, P. Wonka, D. G. Aliaga, M. Wimmer, L. Gool, and W. Purgathofer. A survey of urban reconstruction. *Comput. Graph. Forum*, 32(6):146–177, Sept. 2013. 1