SOLVING THE POINT-PLANE PROBLEM FOR THE CLASS OF AFFINE TRANSFORMATIONS AND DEVELOPMENT OF A FAST ITERATIVE ALGORITHM FOR Registering of 3D POINT CLOUDS

A. Vokhmintsev¹, A. Melnikov², T. Botova³

¹Information Technology and systems, Research Laboratory, Chelyabinsk State University, Chelyabinsk, Russian Federation
²Ugra State University, Khanty-Mansiysk, Russian Federation
³Intelligent information technology and systems, Research Laboratory, Chelyabinsk State University, Chelyabinsk, Russian Federation

Abstract: Solution to the point-plane problem for the class of affine transformations will be found, and a fast accuracy iterative algorithm for registering 3D point clouds will be designed.

1. Introduction

The scientific problem at solving which the project is aimed consists in the development of methods of generating a three-dimensional combined dense map are of the accessible environment around mobile a robot. To construct the qualitative three-dimensional models it is necessary to share data received from different sensors (images, clouds of three-dimensional points, depth maps) at each moment of time. Depth sensors and laser scanners are using to obtain information about the shape of a three-dimensional object [1-3]. Kinect camera is widely used to obtain the depth map of a scene. Existing algorithms for solving this problem can be divided into two groups. The algorithms of the first group are based on the classical algorithm ICP [4] and obtain a sparse cloud of three-dimensional points by calculating the structure of the motion sensor through a set of images. The disadvantage of this approach is the need for user input during the initial mapping phase. Algorithms of the second group are based on comparison of features in two-dimensional space of descriptors (SIFT, SURF, ORB) [5, 6]. The disadvantage of the second group of algorithms is the high computational complexity and the assumption of a sufficient number of common features on images of neighboring frames.

A common registration algorithm solves the variational problem of finding the optimum geometric (orthogonal or affine) transformation, which best matches two clouds of points with a given correspondence between points [4]. Different functionals lead to various methods for registration of point clouds. The most commonly used methods for searching correspondence between a pair of clouds are point-to-point and point-to-plane. For the class of orthogonal transformations, the solution of the point-to-point problem was explicitly given by un Horn [7,8]. An exact solution of the point-to-point problem for the case of an arbitrary affine transformation is given in following paper6. The variational problem of the point-to-point in the class of orthogonal transformations is usually solved using either the Levenberg-Marquardt iteration algorithm or the linearization method for small angles [9]. For the variational problem of the point-plane for the class of affine transformations, we proposed the exact solution [10]. An approximate solution of the point-to-plane problem for the class of orthogonal transformations was obtained using Khoshelham algorithm [11]. Note that the point-plane method for the class of orthogonal transformations is more robust to noise of sensors, but for this method the explicit solution of the problem has not been found yet. This complicates the use of the method in real-time registration applications.

One of the main problems that has not been acceptably solved in the field of SLAM is that of dynamic investigated environment. The preliminary results of the project have shown that the use of an approximate 3D map of the surrounding environment [12] significantly improves the quality of recognition and localization of objects in dynamic, bulky scenes, especially with partial or complete occlusion of a object by other scene objects [13, 14]. Basically, the quality of the depth map obtained from the Kinect camera is good. However, there are some problems with the output information. So in the output data, invalid areas are formed due to the following reasons: the structured light of the radiation does not come to the camera after reflection, the resolution of the measurement of the depth decreases by a quadratic law with increasing depth, and a fast movement of the camera also leads to loss of data. Within the framework of this project, we are going to analyze existing calibration and correction algorithms and to develop a new algorithm for constructing an accurate 3D map of the surrounding environment for dynamic, contextually complicated and large-scale scenes [15-17]. In the context of this project new accurate methods for reconstruction of three-dimensional maps of the environment will be developed in terms of complex navigation tasks of mobile robot in unknown dynamic environment with obstacles [18, 19]. We will present the foundation for required estimates of accuracy of the reconstruction of the environment and computational complexity based on adaptive approach. The proposed method will show the accurate reconstruction of scenes are not less than a specified threshold.

Within the framework of the proposed project, new accurate method for reconstruction of three-dimensional map of the unknown dynamic environment will be provided. The paper is organized as follows. In Section 2, the 3d scene model is presented. In Section 2 the proposed registration algorithm of 3D point clouds is presented and discussed. Computer simulation results are provided in Section 3. Section 4 summarizes our conclusions.

2. Fast iterative algorithm for registering of 3D point clouds

Let \( P = \{p_1, ..., p_n\} \) be a source point cloud, and \( Q = \{q_1, ..., q_n\} \) be a destination point cloud in \( \mathbb{R}^3 \). Suppose that the relationship between points in \( P \) and \( Q \) is given in such a manner that for each point \( p_i \) there is a corresponding point \( q_i \). The ICP algorithm is commonly considered as a geometrical transformation for rigid objects mapping \( P \) to \( Q \):

\[
RP_i + t, \quad (1)
\]

where \( R \) is a rotation matrix, \( t \) is a translation vector, \( i = 1, ..., n \). The S-ICP algorithm uses a slightly different geometrical transformation given as

\[
SRP_i + t, \quad (2)
\]

where \( S \) is a scaling matrix.

The group of affine transformations in the dimension of three has 12 generators. It means that the affine transformation in the dimension of three is a function of 12 variables. Let us consider the ICP variational problem for an arbitrary affine transformation in the point-to-plane case. Denote by \( S(Q) \) a surface constructed from the cloud \( Q \), by \( T(q') \) a tangent plane of \( S(Q) \) at point \( q' \). Let \( J(A, T) \) be the following function:

\[
J(A) = \sum_{i=1}^{n} <Ap' - q'n'>^2, \quad (3)
\]

where \( <;> \) denotes the inner product, \( A \) is a matrix of an affine transformation in the homogenous coordinates:
The sum \( \sum_{i=1}^{n} n'_i < q', n'_i > \) is constant with respect to \( A \).

denote it by \( c_j: \sum_{i=1}^{n} n'_i < q', n'_i > = c_j \).

The expression (10) takes the form:

\[
\sum_{i=1}^{n} < A p'_i, n'_i > n'_i = c_j, j = 1, \ldots, 3.
\]

Denote by the \( (PN)^i \) the following matrix:

\[
(PN)^i = p'^i n'' = \begin{pmatrix}
    p'_1 n''_1 & p'_2 n''_2 & p'_3 n''_3 \\
    0       & 0       & 0
\end{pmatrix}
\]

Let us consider the following expression:

\[
< A p'_i, n'_i > = tr (A (PN)^i).
\]

It follows from

\[
< A p'_i, n'_i > = \begin{pmatrix}
    a_{11} & a_{12} & a_{13} & t_1 \\
    a_{21} & a_{22} & a_{23} & t_2 \\
    a_{31} & a_{32} & a_{33} & t_3
\end{pmatrix} \begin{pmatrix}
    n''_1 \\
    n''_2 \\
    n''_3 \\
    0
\end{pmatrix} >=
\begin{pmatrix}
    p'_1 n'_1 + a_{13} p'_3 n'_1 + t_1 n'_1 \\
    p'_2 n'_2 + a_{23} p'_3 n'_2 + t_2 n'_2 \\
    p'_3 n'_3 + a_{33} p'_3 n'_3 + t_3 n'_3
\end{pmatrix} = 0
\]

\[
(14)
\]

and

\[
< A p'_i, n'_i > = \begin{pmatrix}
    a_{11} & a_{12} & a_{13} & t_1 \\
    a_{21} & a_{22} & a_{23} & t_2 \\
    a_{31} & a_{32} & a_{33} & t_3
\end{pmatrix} \begin{pmatrix}
    n''_1 \\
    n''_2 \\
    n''_3 \\
    0
\end{pmatrix} >=
\begin{pmatrix}
    a_{11} p'_1 n'_1 + a_{12} p'_2 n'_1 + a_{13} p'_3 n'_1 + t_1 n'_1 \\
    a_{21} p'_2 n'_2 + a_{22} p'_2 n'_2 + a_{23} p'_3 n'_2 + t_2 n'_2 \\
    a_{31} p'_3 n'_3 + a_{32} p'_3 n'_3 + a_{33} p'_3 n'_3 + t_3 n'_3
\end{pmatrix} = 0
\]

\[
(15)
\]

\[
A = \begin{pmatrix}
    a_{11} & a_{12} & a_{13} & t_1 \\
    a_{21} & a_{22} & a_{23} & t_2 \\
    a_{31} & a_{32} & a_{33} & t_3
\end{pmatrix}
\]

Then new project solutions to 3D total variation regularization will be obtained. The function \( J(A, T) \) can be rewritten:

\[
J(A) = \sum_{i=1}^{n} < A p_i - q_i, n_i >^2 = \sum_{i=1}^{n} < A p_i - q_i, n_i >^2,
\]

\[
= \sum_{i=1}^{n} < A p_i, n_i >^2 - 2 < A p_i, q_i > + < q_i, n_i >^2
\]

where \( p_i \) is a point from the point cloud \( P \), \( n_i \) is the unitary normal for \( T(q) \).

The term \( < q_i, n_i > \) is a constant with respect to \( A \), thus the variational problem (6) takes the form:

\[
\sum_{i=1}^{n} < A p_i, n_i >^2 - 2 < A p_i, q_i > + < q_i, n_i >^2 = \min_A
\]

Let us rewrite the inner product \( < A p_i, n_i > \) as follows:

\[
< A p_i, n_i > = (a_{i1} p'_1 n'_1 + a_{i2} p'_2 n'_1 + a_{i3} p'_3 n'_1 + t_i n'_1) +
(+ (a_{i1} p'_1 n'_2 + a_{i2} p'_2 n'_2 + a_{i3} p'_3 n'_2 + t_i n'_2) +
+ (a_{i1} p'_1 n'_3 + a_{i2} p'_2 n'_3 + a_{i3} p'_3 n'_3 + t_i n'_3) + (0)
\]

\[
J(A) \text{ is a function of 12 variables. Let us consider components of the gradient } \nabla J(A).
\]

**Partial derivatives with respect to \( t \)**

Let us consider the partial derivatives with respect to \( a_{ij}, i, j = 1, \ldots, 3; \)

\[
\frac{\partial J}{\partial t_j} = \sum_{i=1}^{n} 2 < A p'_i, n'_i > n'_j -
- 2 n'_j < q', n'_i > = 0, j = 1, 2, 3.
\]

\[
\sum_{i=1}^{n} < A p'_i, n'_i > n'_j - \sum_{i=1}^{n} n'_j < q', n'_i > = 0.
\]
Let us rewrite Eq. (13) taking into account Eq. (15) as
\[ \sum_{i=1}^{n} n_i^j \text{tr} \left( A \left( PN \right)^j \right) c_j, \quad j = 1, \ldots, 3. \] (16)

Denote by \((n_j PN)^j\) the following matrices:
\[ (n_j PN)^j = \begin{pmatrix} p_i^j n_i^j & p_i^j n_i^j & p_i^j n_i^j & 0 \\ p_i^j n_i^j & p_i^j n_i^j & p_i^j n_i^j & 0 \\ p_i^j n_i^j & p_i^j n_i^j & p_i^j n_i^j & 0 \\ n_i^j & n_i^j & n_i^j & 0 \end{pmatrix}. \] (17)

The following condition holds:
\[ n^j \text{tr} \left( A \left( PN \right)^j \right) = \text{tr} \left( A \left( n_j \left( PN \right)^j \right) \right). \] (18)

Taking into account Eq. (18), we rewrite Eq. (16) as
\[ \sum_{i=1}^{n} \text{tr} \left( A \left( n_j \left( PN \right)^j \right) \right) = c_j. \] (19)

Using the property of the matrix trace, we arrive to
\[ \sum_{i=1}^{n} \text{tr} \left( A \left( n_j \left( PN \right)^j \right) \right) = \text{tr} \left\{ \sum_{i=1}^{n} A \left( n_j \left( PN \right)^j \right) \right\} = c_j. \] (20)

Denote by \(M_j, j = 1, \ldots, 3\), the following matrices:
\[ \sum_{i=1}^{n} (n_j PN)^j = M_j. \] (21)

\[ M_j = \begin{pmatrix} m_{11}^j & m_{12}^j & m_{13}^j & 0 \\ m_{21}^j & m_{22}^j & m_{23}^j & 0 \\ m_{31}^j & m_{32}^j & m_{33}^j & 0 \\ m_{41}^j & m_{42}^j & m_{43}^j & 0 \end{pmatrix}. \] (22)

The equation (22) can be rewritten as
\[ \text{tr} \left( A M^j \right) = c_j, \quad j = 1, \ldots, 3. \] (23)

\[ A M^j = \begin{pmatrix} a_{11} & a_{12} & a_{13} & t_1 \\ a_{21} & a_{22} & a_{23} & t_2 \\ a_{31} & a_{32} & a_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} m_{11}^j & m_{12}^j & m_{13}^j & 0 \\ m_{21}^j & m_{22}^j & m_{23}^j & 0 \\ m_{31}^j & m_{32}^j & m_{33}^j & 0 \\ m_{41}^j & m_{42}^j & m_{43}^j & 0 \end{pmatrix} = \begin{pmatrix} a_{11} m_{11}^j + a_{12} m_{12}^j + a_{13} m_{13}^j + t_1 m_{41}^j \\ a_{21} m_{11}^j + a_{22} m_{12}^j + a_{23} m_{13}^j + t_2 m_{42}^j \\ a_{31} m_{11}^j + a_{32} m_{12}^j + a_{33} m_{13}^j + t_3 m_{43}^j \\ 0 \end{pmatrix}. \] (24)

We get the following equations for \( j = 1, \ldots, 3 \):
\[ (a_{11} m_{11}^j + a_{12} m_{12}^j + a_{13} m_{13}^j + t_1 m_{41}^j) + \]
\[ + (a_{21} m_{11}^j + a_{22} m_{12}^j + a_{23} m_{13}^j + t_2 m_{42}^j) + \]
\[ + (a_{31} m_{11}^j + a_{32} m_{12}^j + a_{33} m_{13}^j + t_3 m_{43}^j) = c_j, \]
\[ j = 1, \ldots, 3, i = 1, \ldots, n \]

Partial derivatives with respect to \( a \).
Let us consider the partial derivatives with respect to \( a_{ij}, i, j = 1, \ldots, 3 \).
\[ \frac{\partial J}{\partial a_{ij}} = \sum_{k=1}^{n} 2 < A p^k, n^k > p_{ij}^k, n_{ij}^k - \]
\[ - 2 p_{ij}^k, n_{ij}^k < q^k, n^k >= 0, j = 1, \ldots, 3, \]
\[ k = 1, \ldots, n \]
\[ = \sum_{i=1}^{n} < A p^k, n^k > p_{ij}^k, n_{ij}^k - \]
\[ - \sum_{i=1}^{n} p_{ij}^k, n_{ij}^k < q^k, n^k >= 0 \] (26)

The partial sum \( \sum_{k=1}^{n} p_{ij}^k, n_{ij}^k < q^k, n^k > \) is a constant with respect to \( A \):
\[ \sum_{k=1}^{n} p_{ij}^k, n_{ij}^k < q^k, n^k > = c_{ij}, i, j = 1, \ldots, 3. \] (28)

The expression (27) takes the form:
\[ \sum_{k=1}^{n} < A p^k, n^k > = p_{ij}^k, n_{ij}^k = c_{ij}. \] (29)

Taking into account Eqs. (3) and (4) can be rewritten as
\[ \sum_{k=1}^{n} p_{ij}^k, n_{ij}^k < A (PN)^k > = c_{ij}, i, j = 1, \ldots, 3. \] (30)

Denote by \((p_{ij}^k, n_{ij}^k PN^k)\) the following matrices:
Finally, we get 9 equations for $i, j = 1, \ldots, 3$:

$$\begin{align*}
\alpha_{11} m_{11}^q + \alpha_{12} m_{21}^q + \alpha_{13} m_{31}^q + t_1 m_{41}^q \\
\alpha_{21} m_{12}^q + \alpha_{22} m_{22}^q + \alpha_{23} m_{32}^q + t_2 m_{42}^q \\
\alpha_{31} m_{13}^q + \alpha_{32} m_{23}^q + \alpha_{33} m_{33}^q + t_3 m_{43}^q = c_{ij},
\end{align*}$$

(35)

where matrix $Z$ has size $9 \times 12$ and consist of the elements $m_{ij}, C$ is the column consist of 12 numbers $c_i$ and $c_9$, and the column $\Omega$ :

$$\begin{align*}
\Omega &= (a_1, a_{12}, a_{13}, t_1, a_{21}, a_{22}, a_{23}, t_2, a_{31}, a_{32}, a_{33}, t_3)^T.
\end{align*}$$

(38)

3. Computer simulation

In this section, computer simulation results are presented and discussed. We used Kinect as RGB-D camera. The following Indoor Segmentation and Support Inference from RGB Images ECCV 2012 (Indoor Segmentation and Support database) was used for the tests (http://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html#raw_parts), which contains semantic space labeling. We have selected four datasets of large areas of the reference database with insufficient RANSAC inliers. To obtain this result we discovered denendancy the accuracy of the three-dimensional reconstruction with respect to the number and spatial distribution of singular points in the frame, in a sequences of experiments we found out a minimal number of features providing a given accuracy of the three-dimensional reconstruction for the type of each method (descriptor) in Table 1. We have made a argumentation for estimates of the accuracy and computational complexity of the proposed method. It was found that the accuracy of 3D reconstruction of scene depends with respect to the number of feaures in the frame in a nonlinear way. This function has one distinct extreme for all types of descriptors. This conclusion is very important to assess the convergence of the visual SLAM algorithms in real-time.

$$\begin{align*}
(a_{11} m_{11}^q + a_{12} m_{21}^q + a_{13} m_{31}^q + t_1 m_{41}^q ) + \\
(a_{21} m_{12}^q + a_{22} m_{22}^q + a_{23} m_{32}^q + t_2 m_{42}^q ) + \\
(a_{31} m_{13}^q + a_{32} m_{23}^q + a_{33} m_{33}^q + t_3 m_{43}^q ) = c_{ij},
\end{align*}$$

(36)

The expression (1) can be rewritten as

$$\sum_{k=1}^{n} tr(\mathbf{A} \mathbf{P} \mathbf{K} \mathbf{H}) = c_{ij}. \quad (32)$$

Denote by $\mathbf{M}(j, i, j = 1, \ldots, 3)$, the following matrices:

$$\sum_{k=1}^{n} (\mathbf{P} \mathbf{K} \mathbf{H}) = \mathbf{M}(j, i). \quad (33)$$

$$AM^j = \begin{pmatrix}
a_{11} & a_{12} & a_{13} & t_1 \\
a_{21} & a_{22} & a_{23} & t_2 \\
a_{31} & a_{32} & a_{33} & t_3 \\
0 & 0 & 0 & 1
\end{pmatrix} \times
\begin{pmatrix}
m_{11}^q & m_{12}^q & m_{13}^q & 0 \\
m_{21}^q & m_{22}^q & m_{23}^q & 0 \\
m_{31}^q & m_{32}^q & m_{33}^q & 0 \\
m_{41}^q & m_{42}^q & m_{43}^q & 0
\end{pmatrix} =
\begin{pmatrix}
a_{11} m_{11}^q + a_{12} m_{21}^q + a_{13} m_{31}^q + t_1 m_{41}^q \\
a_{21} m_{12}^q + a_{22} m_{22}^q + a_{23} m_{32}^q + t_2 m_{42}^q \\
a_{31} m_{13}^q + a_{32} m_{23}^q + a_{33} m_{33}^q + t_3 m_{43}^q
\end{pmatrix}, \quad (35)

Table 1. Quality of reconstructing a three-dimensional combined dense map with respect to descriptor

<table>
<thead>
<tr>
<th>Method</th>
<th>Dataset-1</th>
<th>Dataset-2</th>
<th>Dataset-3</th>
<th>Dataset-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFT</td>
<td>93</td>
<td>93</td>
<td>94</td>
<td>95</td>
</tr>
<tr>
<td>SURF</td>
<td>57</td>
<td>61</td>
<td>63</td>
<td>59</td>
</tr>
<tr>
<td>ORB</td>
<td>78</td>
<td>82</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>HOGs</td>
<td>93</td>
<td>96</td>
<td>97</td>
<td>96</td>
</tr>
<tr>
<td>The processing time, s</td>
<td>17.2</td>
<td>26.21</td>
<td>12.82</td>
<td>18.9</td>
</tr>
<tr>
<td>SIFT</td>
<td>0.23</td>
<td>0.7</td>
<td>0.4</td>
<td>0.44</td>
</tr>
<tr>
<td>SURF</td>
<td>3.34</td>
<td>3.42</td>
<td>5.15</td>
<td>3.33</td>
</tr>
<tr>
<td>ORB</td>
<td>0.79</td>
<td>1.11</td>
<td>0.68</td>
<td>0.81</td>
</tr>
</tbody>
</table>

4. Conclusion
In this project the following main results were obtained:

- Solving the point-plane problem for the class of affine transformations;
- Development of a new fast iterative method for constructing a 3D accurate map of the surrounding environment using sequences of sparse point clouds obtained from depth sensors for dynamic, contextually complicated and large-scale scenes. It is necessary to design an algorithm for registration cloud scenes with arbitrary spatial resolution and scale relative to each other.
- Development of algorithms that take into account a 3D map of the surrounding environment, for reliable recognition and localization of objects in dynamic, contextually complicated scenes, especially with partial or complete occlusion of persons by other scene objects.

Acknowledgments
The work was supported by the RFBR, project no 16-08-00342.

References