

Lines to Planes Registration

Ales Jelinek, Adam Ligocki

Central European Institute of Technology, Purkynova 123, Brno, Czech Republic
{Ales.Jelinek, Adam.Ligocki}@ceitec.vutbr.cz

Keywords: Line, Plane, Registration, Least squares, Optimization, Rigid transformation, Data fitting

Abstract: This paper deals with the Line-Plane registration problem. First we introduce the matter of geometrical object fitting and show the specificity of the Line-Plane combination. Then the method is derived to fulfil all usual demands we put on a registration algorithm. Our algorithm works on any number of corresponding Line-Plane pairs higher than three, providing more accurate results and better convergence, as this number gets higher. Although the computation is non-linear by its nature, benefits of the least squares optimization were preserved. The algorithm is divided into a linear part dependant on the amount of input data and a non-linear part, which is not, keeping efficiency in case of large data sets. All important features were experimentally tested and proved to work.

1 INTRODUCTION

Registration of two sets of geometrical objects is a longly solved problem in multiple disciplines of science and engineering. Finding an optimal transformation, which brings one set as close to the other as possible is an underlying principle connecting robotic mapping (Cadena et al., 2016), object recognition (Grimson and Lozano-Pérez, 1984), pose determination (Markley and Mortari, 2000) and other related tasks. There is not a single type of transformation to be discussed and neither a definition of "closeness" is universal. Moreover, the geometrical objects may vary in their nature (points, lines, planes, vectors and rigid bodies to name the most common) and there might not even be the same kind objects in both sets. Another defining difference is, whether we know which objects from both sets correspond with each other, or not.

All this diversity leads to a large number of approaches developed over the last decades (Cadena et al., 2016), (Pomerleau et al., 2015), (Tam et al., 2013). We are going to focus on rigid transformation and known correspondences. There is plenty of algorithms for registration problem with points in at least one of the data sets (Wientapper and Kuijper, 2017), (Olsson et al., 2006), (Ramalingam et al., 2010). The Line-Line (Li et al., 2016) and Plane-Plane (Forstner and Khoshelham, 2017) problem is addressed much less, but we can still find relevant literature. Surprisingly, the Line-Plane problem draw minimal attention

over time and we were able to locate a single paper (Chen, 1991) with working, yet somewhat unwieldy algorithm.

In many practical situations, the lines are just approximations computed from point-like data, so the Line-Plane task can be easily converted into a Point-Plane equivalent. Though it is a viable option, storing point-like data might get very memory demanding and sometimes these data do not exist at all. If extra memory requirements are impractical, extra sampling of lines to obtain points is terribly inefficient. Linear and planar approximations can be used for further processing (Cadena et al., 2016) while point-based methods provide no additional value. Becoming independent on points by direct solution of the Line-Plane problem is therefore a useful addition to the state of the art of the registration algorithms. This paper aims to provide an alternative to (Chen, 1991) with more favorable properties listed below:

- Our algorithm works on any number of corresponding pairs $N \geq 3$ in contrast to (Chen, 1991) which needs exactly $N = 3$.
- The algorithm consists of two stages: I. Simple linear computation with $O(N)$ complexity (proportional to the number of correspondences) and II: Non-linear optimization independent on N .
- Addition and removal of a single correspondence from the sets results in a constant time operation on results precomputed in the first stage.
- Initial guess to speed up the second stage can be

supplied.

- Provides metric directly describing quality of the registration.
- Works for general planes and lines in 3D as well as for general planes and lines lying in a single plane (the case of line laser scanning mentioned in (Chen, 1991), etc.).

To put the Line-Plane registration into the context and provide fluent introduction to actual derivation of our method, we will discuss it together with a closely related topic of methods designed for Vector-Vector matching.

2 RELEVANT FACTS TO LINE-PLANE REGISTRATION

As the work directly related our topic is so rare, we will not start with traditional literature review, but rather state the problem and than discuss its connections and differences from more established domains.

2.1 Problem statement

Rigid transformation can be described by a rotation matrix R and translation vector \mathbf{t} acting on a vector in the well known manner: $\mathbf{x}' = R\mathbf{x} + \mathbf{t}$. A line can be described by a unit directional vector \mathbf{d} and another vector \mathbf{p} representing a point lying on it. A general plane is uniquely determined by a unit normal vector \mathbf{n} and the shortest distance l along this vector to the origin. (Chen, 1991) formulates the problem as finding R and \mathbf{t} satisfying the equations:

$$\mathbf{n}_i \cdot R\mathbf{d}_i = 0 \quad (1)$$

and

$$\mathbf{n}_i \cdot (R\mathbf{p}_i + \mathbf{t}) = l_i, \quad (2)$$

for $i = 1, 2, 3$. As we are going to work with $N \geq 3$ number of corresponding pairs, we will search for R and \mathbf{t} minimizing the error functions derived from (1) and (2). For rotation R we get:

$$E_r(R) = \sum_{i=1}^N (\mathbf{n}_i \cdot R\mathbf{d}_i)^2, \quad (3)$$

but generalization of (2) is not that straightforward. (Chen, 1991) uses only three equations to find R , therefore the perfect match is guaranteed and the three equalities (1) for $i = 1, 2, 3$ will be exactly valid. When using $N \geq 3$ equations, residual error depending on parametrization of the line will appear. \mathbf{p}_i is an arbitrary point lying on the i -th line, so adding

some multiple of \mathbf{d}_i to \mathbf{p}_i should not influence anything. Simple rearrangement of (2) gives:

$$\mathbf{n}_i \cdot \mathbf{t} = l_i - \mathbf{n}_i \cdot R\mathbf{p}_i. \quad (4)$$

Plugging $\mathbf{p}_i + \delta\mathbf{d}_i$ instead of \mathbf{p}_i into (4) should not influence the equality, but the following happens:

$$\mathbf{n}_i \cdot \mathbf{t} = l_i - \mathbf{n}_i \cdot R\mathbf{p}_i - \delta\mathbf{n}_i \cdot R\mathbf{d}_i. \quad (5)$$

The last term $\delta\mathbf{n}_i \cdot (R\mathbf{d}_i)$ is guaranteed to be zero if and only if $N = 3$, otherwise this is not necessarily true and selection of \mathbf{p}_i could influence the result, which is undesirable. To overcome this issue, we have decided to use the point on the line closest to the origin of coordinates ($\mathbf{p}_{i0} = \mathbf{p}_i - (\mathbf{p}_i \cdot \mathbf{d}_i)\mathbf{d}_i$) in place of arbitrary \mathbf{p}_i , so the computation will get independent of the choice of \mathbf{p}_i . Keeping the original notation we arrive at the error function for translation:

$$E_t(R, \mathbf{t}) = \sum_{i=1}^N (\mathbf{n}_i \cdot (R(\mathbf{p}_i - \mathbf{d}_i(\mathbf{d}_i \cdot \mathbf{p}_i)) + \mathbf{t}) - l_i)^2. \quad (6)$$

(Kamgar-Parsi and Kamgar-Parsi, 2011) advocate even more robust approach to translation estimation, pointing out dependency of the solution on selection of the coordinate system. We found it much less severe than the dependency on parametrisation of the lines, so we have taken no precautions. In case of necessity, our method should be modifiable in that manner at cost of additional computational complexity.

Since evaluation of E_t depends both on R and \mathbf{t} and there is a separate formula for R only, it is possible to start with minimization of E_r alone and than minimize E_t with given R . This way we follow order of the operations during transformation and obtain sensible results for both parameters, as presented in 2D case of line segments in (Jelinek, 2018). Let us examine the rotational part at first.

2.2 Optimal Rotation

Lines and planes can be, among many other ways, described using vectors. If only orientation is a concern, than for lines, we extract the directional vectors and for planes the normal vectors. In Line-Line and Plane-Plane problems, the rotational part of registration boils down to alignment of two sets of vectors. This operation is well described in literature and the first appearance was probably the Wahba's problem (Wahba, 1965) formulated as follows:

Given two sets of N points $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N\}$ and $\{\mathbf{v}'_1, \mathbf{v}'_2, \dots, \mathbf{v}'_N\}$, where $N \geq 2$, find the rotation matrix R (i.e. the orthogonal matrix with determinant +1) which brings the first set into the best least squares coincidence with the second. That is, find R which minimizes:

$$\sum_{i=1}^N \|\mathbf{v}'_i - R\mathbf{v}_i\|^2. \quad (7)$$

2.2.1 A state of the art solution for vectors

There are plenty of solutions, for example a decomposition method by (Farrell et al., 1966), the Kabsch algorithm (Kabsch, 1976) and many other papers, nicely surveyed in (Markley and Mortari, 2000). Solution using singular value decomposition (SVD) is widely accepted and proceeds as follows.

Stationary vectors \mathbf{v}'_i and rotated vectors \mathbf{v}_i are organized in matrices:

$$V = \begin{bmatrix} v_{x1} & v_{y1} & v_{z1} \\ v_{x2} & v_{y2} & v_{z2} \\ \vdots & \vdots & \vdots \\ v_{xN} & v_{yN} & v_{zN} \end{bmatrix}, V' = \begin{bmatrix} v'_{x1} & v'_{y1} & v'_{z1} \\ v'_{x2} & v'_{y2} & v'_{z2} \\ \vdots & \vdots & \vdots \\ v'_{xN} & v'_{yN} & v'_{zN} \end{bmatrix}, \quad (8)$$

which provide the covariance matrix C :

$$C = V^T V', \quad (9)$$

with T for transposition. SVD of the covariance matrix expresses it as a product of three matrices:

$$C = USV^T. \quad (10)$$

The optimal rotation matrix is given by equation:

$$R = V \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{pmatrix} U^T, \quad (11)$$

where $d = \det(VU^T)$. Note that the rotation R is analytically expressed in terms of linear algebra providing a single optimal result. Although orthonormality of the rotation matrix imposes non-linear constraints on its nine elements, there is no need to reflect them in the computation. SVD approach (8) - (11) represents state of the art solution, but it does not clearly illustrate geometrical reason, why is that possible.

2.2.2 On unnecessary of non-linear constraints in Vector-Vector registration

The example below briefly shows, why for Vector-Vector registration the constraints are naturally fulfilled and why the Line-Plane case is more complicated.

Let us again start with the two sets of unitary corresponding vectors (8). Coordinate frames of each set are described by the orthonormal bases of three vectors $E = \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ and $E' = \{\mathbf{e}'_1, \mathbf{e}'_2, \mathbf{e}'_3\}$, both equal to $E_0 = \{\{1, 0, 0\}, \{0, 1, 0\}, \{0, 0, 1\}\}$ at the beginning. Then for every corresponding pair, we find a rotation R_i which brings the registered vector \mathbf{v}_i into coincidence with static \mathbf{v}'_i . Axis of such rotation is determined by the cross product $\mathbf{v}_i \times \mathbf{v}'_i$ and the angle corresponds to $\arccos(\mathbf{v}_i \cdot \mathbf{v}'_i)$. Frame E would then

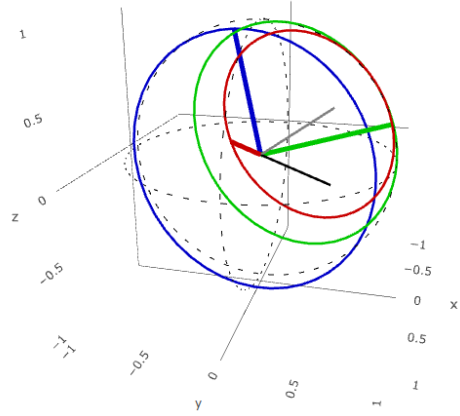


Figure 1: All possible poses of the coordinate frame E bringing vector \mathbf{v}_i (black) into perfect coincidence with \mathbf{v}'_i (grey) from the second set. Red, green and blue circular traces demarcate possible locations of the basis vectors of E .

correspond to $R_i E_0$ as depicted in Fig. 1, where RGB vectors are used to plot its axes.

But that is only one of infinitely many possible rotations. After the first alignment, we can rotate E about \mathbf{v}'_i by an arbitrary angle and both vectors will still perfectly coincide. Further rotation about \mathbf{v}'_i results into a circular trace around \mathbf{v}'_i for each \mathbf{e}_j (see again Fig.1). One corresponding pair of vectors is therefore not enough to specify orientation of the new basis. On the other hand, we know exactly the magnitudes of projections of the basis vectors \mathbf{e}_j onto \mathbf{v}'_i , which remain constant for all possible poses of E . With $N \geq 3$, we can for every \mathbf{e}_j construct a system of linear equations:

$$A_j = V', \quad B_j = \begin{bmatrix} (R_1 \mathbf{e}_{j0}) \cdot \mathbf{v}_1 \\ (R_2 \mathbf{e}_{j0}) \cdot \mathbf{v}_2 \\ \vdots \\ (R_N \mathbf{e}_{j0}) \cdot \mathbf{v}_N \end{bmatrix}. \quad (12)$$

For noise-less vector sets V and V' , the least squares solutions of the $A_j \mathbf{e}_j = B_j$ systems yield the coordinate frame E , which directly represents the rotation R we sought for. Indeed, the approach sketched above is inferior to the SVD solution, because it requires $N \geq 3$ even though it is proved that $N \geq 2$ is enough (Markley and Mortari, 2000), SVD works on noised data as well and for large N its computational costs are much lower. On the other hand, this example clearly illustrates, why no non-linear constraints are required in Vector-Vector registration. Distribution of the invariant projections in matrices B_j and (for noised data) the properties of SVD encode enough information to retrieve proper rotation matrix R .

2.2.3 The Line-Plane problem specificity

The geometric intuition behind the computation procedure described above can be easily extended to Line-Plane registration. Using the notation from introductory part of Section 2, the rotation which makes a directional vector \mathbf{d}_i of a line orthogonal to a normal vector \mathbf{n}_i of a plane, can be easily computed using the products $\mathbf{d}_i \times \mathbf{n}_i$ and $\mathbf{d}_i \cdot \mathbf{n}_i$, similar to the Vector-Vector example above. After that operation, we can freely rotate the coordinate frame not only about \mathbf{n}_i , which corresponds to the previous example as well, but also about the, now orthogonal, \mathbf{d}_i .

This is a defining difference, because two degrees of freedom in rotation cause loss of the constant size projections, leaving us with no invariants to rely on. Depiction in sense of Fig. 1 would show three overlapping ring-like bands spanned around the \mathbf{n}_i vector. It would be possible to use inequalities to describe all possible locations of the basis vectors, but the advantage of implicitly present constraints embodied in invariants would be lost anyway. This thought experiment shows, inevitable non-linearity of the rotational registration problem when lines and planes are in question.

2.3 Optimal Translation

Contrary to the complicated matter of search for optimal rotation, the translational part of the registration turns out to be somewhat simpler. In case of formula (2) by (Chen, 1991) it is a linear system of three equations with three variables. Due to our extension to $N \geq 3$ corresponding pairs we have to adopt the adjustment described in the problem statement in Section 2.1. Then the optimal solution to minimization of (6) turns out to be a straightforward least squares computation as we will show near the end of the following section.

3 DERIVATION OF THE METHOD

Previous discussion revealed, that our main concern is minimization of (3) in search for optimal rotation matrix R . We denote its elements as follows:

$$R = [\mathbf{r}_x \quad \mathbf{r}_y \quad \mathbf{r}_z] = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (13)$$

and use them to write down the constraints, which ev-

ery proper rotation matrix has to fulfil:

$$\begin{aligned} \mathbf{r}_x \cdot \mathbf{r}_x - 1 &= 0, \\ \mathbf{r}_y \cdot \mathbf{r}_y - 1 &= 0, \\ \mathbf{r}_z \cdot \mathbf{r}_z - 1 &= 0, \\ \mathbf{r}_x \cdot \mathbf{r}_y &= 0, \\ \mathbf{r}_y \cdot \mathbf{r}_z &= 0, \\ \mathbf{r}_z \cdot \mathbf{r}_x &= 0 \end{aligned} \quad (14)$$

and then we define a vector \mathbf{r} of R elements such that:

$$\mathbf{r} = [r_{11} \quad r_{12} \quad r_{13} \quad r_{21} \quad \cdots \quad r_{33}]. \quad (15)$$

Next we take an outer product of \mathbf{n}_i and \mathbf{d}_i :

$$\mathbf{n}_i \mathbf{d}_i^T = \begin{bmatrix} n_{xi}d_{xi} & n_{xi}d_{yi} & n_{xi}d_{zi} \\ n_{yi}d_{xi} & n_{yi}d_{yi} & n_{yi}d_{zi} \\ n_{zi}d_{xi} & n_{zi}d_{yi} & n_{zi}d_{zi} \end{bmatrix} \quad (16)$$

and organize it similar to \mathbf{r} into a vector \mathbf{s}_i :

$$\mathbf{s}_i = [n_{xi}d_{xi} \quad n_{xi}d_{yi} \quad n_{xi}d_{zi} \quad n_{yi}d_{xi} \quad \cdots \quad n_{zi}d_{zi}]. \quad (17)$$

After this preparation, the error function (3) can be expressed as:

$$E_r(R) = \sum_{i=1}^N (\mathbf{n}_i \cdot R \mathbf{d}_i)^2 = \sum_{i=1}^N (\mathbf{s}_i \cdot \mathbf{r})^2. \quad (18)$$

Equation (18) in conjunction with constraints (14) forms a starting point for further computation. In principle, we try to find a least squares solution best fulfilling equality $\mathbf{s}_i \cdot \mathbf{r} = 0$ for $i = 1 \cdots N$, but non-linear constraints prevent us from the straightforward way of acquiring the result. Instead, we employ Lagrange multipliers λ_j and formulate a Lagrange function:

$$\mathcal{L}_{Er}(\mathbf{r}, \lambda_{1, \dots, 6}) = \sum_{i=1}^N (\mathbf{s}_i \cdot \mathbf{r})^2 - \sum_{j=1}^6 \lambda_j c_j(\mathbf{r}), \quad (19)$$

where $c_j(\mathbf{r})$ are constraints (14). Partially differentiating this formula with respect to \mathbf{r} yields the following

set of equations:

$$\begin{aligned}
2\lambda_1 r_{11} + \lambda_4 r_{12} + \lambda_5 r_{13} &= 2 \sum_{i=1}^N n_{xi} d_{xi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_2 r_{12} + \lambda_4 r_{11} + \lambda_6 r_{13} &= 2 \sum_{i=1}^N n_{yi} d_{yi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_3 r_{13} + \lambda_5 r_{11} + \lambda_6 r_{12} &= 2 \sum_{i=1}^N n_{zi} d_{zi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_1 r_{21} + \lambda_4 r_{22} + \lambda_5 r_{23} &= 2 \sum_{i=1}^N n_{yi} d_{xi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_2 r_{22} + \lambda_4 r_{21} + \lambda_6 r_{23} &= 2 \sum_{i=1}^N n_{yi} d_{yi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_3 r_{23} + \lambda_5 r_{21} + \lambda_6 r_{22} &= 2 \sum_{i=1}^N n_{yi} d_{zi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_1 r_{31} + \lambda_4 r_{32} + \lambda_5 r_{33} &= 2 \sum_{i=1}^N n_{zi} d_{xi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_2 r_{32} + \lambda_4 r_{31} + \lambda_6 r_{33} &= 2 \sum_{i=1}^N n_{zi} d_{yi} \mathbf{s}_i \cdot \mathbf{r}, \\
2\lambda_3 r_{33} + \lambda_5 r_{31} + \lambda_6 r_{32} &= 2 \sum_{i=1}^N n_{zi} d_{zi} \mathbf{s}_i \cdot \mathbf{r}.
\end{aligned} \tag{20}$$

Differentiating of $\mathcal{L}_{Er}(\mathbf{r}, \lambda_{1,\dots,6})$ with respect to $\lambda_{1,\dots,6}$ gives back the constraints (14). (20) and (14) form a system of non-linear equations to be solved for \mathbf{r} and $\lambda_{1,\dots,6}$, directly leading to the rotation matrix R . The crucial result is the independence of the size of the non-linear system on the number of corresponding pairs N . The information from an arbitrarily high number of corresponding pairs is condensed in the right side sums of (20). Detachment of search for R from data aggregation allows better computational optimization of both stages of the algorithm. Furthermore, because every set of corresponding pairs is fully specified by its sums, separate precomputation allows constant time merging and splitting of sets, for which the sums are known. This feature might be utilized in correspondence search algorithms, making our approach worth usage in this field as well.

In many situations, an initial guess supplied to the numerical solver is mandatory or beneficial at least. Any proper rotation matrix R_0 implies the values of vector \mathbf{r} , but we need a good guess of Lagrange multipliers $\lambda_{1,\dots,6}$ at the beginning as well. Fortunately, proper R_0 automatically satisfies constraints (14) and plugging its elements in place of \mathbf{r} in the system of equations (20) leaves us with linear system of nine equations for six unknowns $\lambda_{1,\dots,6}$. Common least squares solution provides us with good initial setting of $\lambda_{1,\dots,6}$ corresponding to R_0 .

The last step of the registration process is enumeration of the optimal translation. In Section 2.1 we have arrived at the error function (6):

$$E_t(R, \mathbf{t}) = \sum_{i=1}^N (\mathbf{n}_i \cdot (R(\mathbf{p}_i - \mathbf{d}_i(\mathbf{d}_i \cdot \mathbf{p}_i)) + \mathbf{t}) - l_i)^2.$$

With R being known, we can reformulate the problem into N equations:

$$\mathbf{n}_i \cdot \mathbf{t} = l_i - \mathbf{n}_i \cdot R(\mathbf{p}_i - \mathbf{d}_i(\mathbf{d}_i \cdot \mathbf{p}_i)), \tag{21}$$

with variable \mathbf{t} and the right side being known. Another overdetermined linear system suitable for least squares solution have arisen in our computation. Solving it yields the optimal translation we have been looking for, while the sum of squared residuals provides accompanying error metric.

4 PRACTICAL VERIFICATION

Each theoretical result calls for empirical verification and comprehensive testing is mandatory for every new method being deployed to practice. To test our registration procedure, we have designed a few synthetic tests exploring its main properties. The testing code is written in Python 3.6.5 using NumPy 1.14.2 and SciPy 1.0.1 packages. We have not aimed for high computational performance and the implementation is not optimized in that respect. Underlying linear algebra libraries and non-linear solvers are going to vary a lot on different platforms in different applications, so reasonably general benchmarking is highly problematic. To provide at least some numbers, we can say that on a 4.0A GHz CPU with lowly elaborated single threaded implementation, the mean registration time measured was a few milliseconds. Indeed, correspondence count N and quality of initial guess influence that number.

For error enumeration, there is a metric provided by the method itself. For rotation, it is E_r obtained from equations (3) or (18) and for translation we use E_t (6), both available after every computation. We refer it as an error function value. For the needs of synthetic testing, we also employ Frobenius norm of the difference between R estimated and the true rotation matrix R_t used during data generation. Denoting this difference R_d , the Frobenius error E_f of the registration is:

$$F_r(R) = \sqrt{\sum_{j=1}^3 \sum_{k=1}^3 r_{jk}^2}, \tag{22}$$

where r_{jk} are elements of R_d . This error metric is obviously not available in real-life application, because R_t is unknown, but in synthetic testing it provides useful comparison with the ground truth. Similar computation applies for translational part:

$$F_t(\mathbf{t}) = \sqrt{(\mathbf{t} - \mathbf{t}_t) \cdot (\mathbf{t} - \mathbf{t}_t)}, \tag{23}$$

4.1 Stability and Convergence

The first test is focused on reliability of convergence to correct R , when imprecise initial guess is supplied to the algorithm. If the guess is precise, the non-linear solver terminates after the first iteration leaving the guess untouched, regardless the size of N . However if the guess does not comply with the data, the solver might get caught in local minimum and provide suboptimal result. To test this behavior, we have generated precisely corresponding pairs of line directions and plane normals and then rotated the direction with matrix R_t . Next, the noise rotation matrix R_n was composed from arbitrary axis and angle between zero and a predefined maximum. The initial guess was than given by the product $R_n R_t$. Without any noise in the data it is easy to detect suboptimal results, because for correct registration, both E_r and F_r are zero (up to certain negligible numerical error).

Success ratios for various N and initial guess error amplitudes are organized in Tab. 1. One can clearly see importance of data set size for convergence of the algorithm. For minimal input of three correspondences, even a slight deviation of 3° led to potential mismatches, however only two more Line-Plane pairs moved this limit to 30° , which is tremendous improvement for its cost. Increasing N above a few tens of pairs does not seem to help anything, when convergence is in question.

Computation of optimal translation is a linear problem featuring a single minimum for which an analytical solution exists, so convergence is not an issue in this case.

4.2 Noise sensitivity and accuracy

Second important experiment was performed to explore sensitivity of the method, when noise is present in the input data. We have composed the noise matrix

N	Error of the initial guess in degrees					
	1	3	5	10	30	50
3	1,00	0,99	0,98	0,97	0,89	0,77
5	1,00	1,00	1,00	1,00	0,99	0,91
10	1,00	1,00	1,00	1,00	1,00	0,99
30	1,00	1,00	1,00	1,00	1,00	1,00
50	1,00	1,00	1,00	1,00	1,00	1,00
100	1,00	1,00	1,00	1,00	1,00	1,00
300	1,00	1,00	1,00	1,00	1,00	1,00
500	1,00	1,00	1,00	1,00	1,00	1,00
1000	1,00	1,00	1,00	1,00	1,00	1,00

Table 1: Success ratio of correct convergence of the optimal rotation search with given maximal error in initial guess of the matrix and given number N of corresponding pairs used.

using the same rules as in the previous experiment, but now it was applied in conjunction with R_t , when the direction vectors were rotated. In this case, even the optimal rotation produces some amount of error, so neither error metric is zero any more.

Figures 3 and 2 summarize the outcomes of this experiment in case of rotation. For low N , especially for $N = 3$ and 5 we see an interesting behaviour. (Chen, 1991) showed, that for non-degenerate set of three Line-Plane pairs, there is an exact solution to the registration problem, but we observe a non-zero error E_r . More than one reason is possible. First, in some cases, the solution returned by the algorithm might not be optimal (thus exhibiting non-zero error) increasing the overall mean error from all experiments. The second possible cause of these results is the number of vectors being involved. Three or five pairs are simply not enough to provide a reliable statistics. There is a good chance, that none of the vectors involved was influenced by the noise with full amplitude, resulting in more precise data than expected, therefore lowering the mean error values. For higher N , we see a gradual decrease of both error metrics, proving an expectation, that larger data set should lead

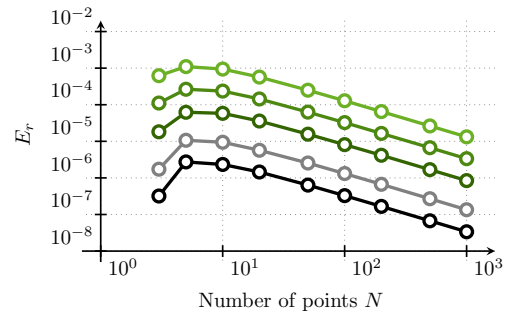


Figure 2: Error function (3) divided by N^2 reflecting expected deviation from true rotation matrix for various number of points N . Paths in the chart belong to $1^\circ, 2^\circ, 5^\circ, 10^\circ, 20^\circ$ peak error in line directions in bottom up order.

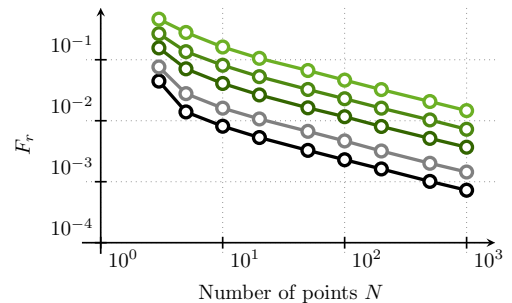


Figure 3: Frobenius error reflecting real deviation from true rotation matrix for various number of points N . Paths in the chart belong to $1^\circ, 2^\circ, 5^\circ, 10^\circ, 20^\circ$ peak error in line directions in bottom up order.

to more precise outcomes. Comparison of E_r and F_r characteristics also proves feasibility of the E_r error metric. Although it tends to underestimate the error for very small N , it corresponds to reliable F_r based on the original true rotation matrix.

The translational part of the registration was of course tested as well. As its linear nature suggests, we observe a gradual decrease in error for higher N , (see Figures 5 and 4), but if $N = 3, 5$ there is more to discuss. First, for cases when $N = 3$, there always exist an exact solution to the system of equations (21), making the expected error E_t zero (not plotted in Fig. 4). Nevertheless, the Frobenius error F_t (Fig. 5) derived from true translation shows non-proportionally high values for $N = 3$. This behavior is even stronger argument against Line-Plane registration with only three samples, than similar effect manifesting in rotation enumeration. The $N = 5$ case is still influenced by error underestimation, but at least, we get reasonable values to alarm our attention. For higher N , E_t nicely reflects true error behavior described by F_t and can be safely used to determine expected error of the computation.

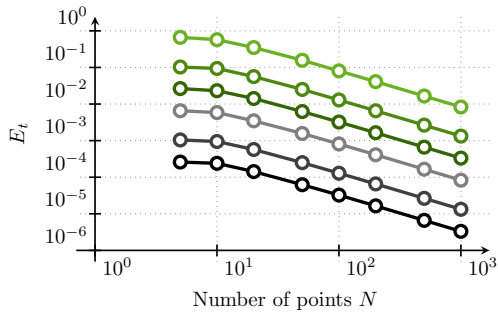


Figure 4: Error function (6) divided by N^2 reflecting expected deviation from true translation vector for various number of points N . Paths in the chart belong to 1, 2, 5, 10, 20 and 50 percent peak error with respect to maximal mutual distance of lines and planes (in bottom up order).

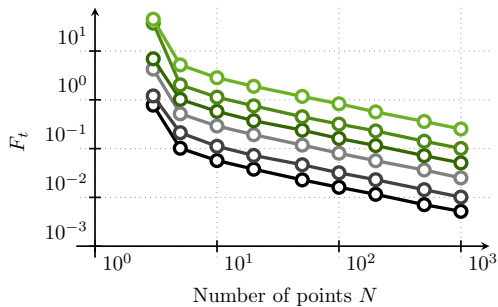


Figure 5: Frobenius error reflecting real deviation from true translation vector for various number of points N . Paths in the chart belong to 1, 2, 5, 10, 20 and 50 percent peak error with respect to maximal mutual distance of lines and planes (in bottom up order).

4.3 Real data testing

Truly empirical experiments turned out to be somewhat problematic to perform, because acquiring highly accurate control measurements is quite hard. Our real-life application is calibration of surround sensing devices for mobile robots (Zalud et al., 2015), (Kocmanova et al., 2013), in this case a mutual pose of 3D and 2D sensing laser scanners (see Fig. 6). Each of these devices has its own frame of coordinates, but its origin is inside of the body of the device, so taking exact measurements is extremely difficult. Verification therefore does not come from direct measurement, but rather from comparison of overlay of the point clouds measured by the devices, before and after the calibration transformation is applied. This overlay can be enumerated by multitude methods resulting in many numerical values (Pomerleau et al., 2015), but the essence of all comparisons is simple: The point clouds after calibration fit together much better than without it (see Fig. 7), despite the metric used. However humble this number-less experiment looks on the paper, it is a clear evidence of functionality of the registration method described above.



Figure 6: The Atlas sensory platform designed for surround sensing of autonomous vehicles. Line-Plane registration is used for mutual pose calibration of the FARO laser scanner and both Velodyne laser scanners.

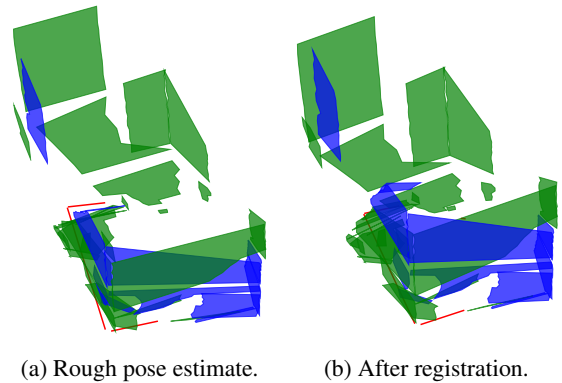


Figure 7: Line-Plane registration in a calibration scenario.

5 CONCLUSIONS

The Line-Plane registration problem is surprisingly unexplored in the literature. This paper provides intuitive introduction into peculiar details of the topic and, to our best knowledge, presents a solution with properties exceeding any other method approaching this problem.

Our algorithm works with any number N of the corresponding Line-Plane pairs greater than three a splits the computation into $O(N)$ complex linear part and N -independent non-linear part. The error metric is supplied, which allows reliability evaluation of the computation and the method also works, if all lines lie in a single plane. Increasing size N of the dataset leads to higher stability of the computation as well as higher precision of the results. If an initial guess of the transformation is provided, the computation is faster and more robust. All these results were practically verified in Section 4. Based on the experiments, we strongly advocate to use at least $N = 5$ for better robustness, which contradicts many methods requiring $N = 3$.

ACKNOWLEDGEMENTS

The research was supported by ECSEL JU under the project H2020 737469 AutoDrive - Advancing fail-aware, fail-safe, and fail-operational electronic components, systems, and architectures for fully automated driving to make future mobility safer, affordable, and end-user acceptable. This research has been financially supported by the Ministry of Education, Youth and Sports of the Czech republic under the project CEITEC 2020 (LQ1601).

REFERENCES

- Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. J. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, 32(6):1309–1332.
- Chen, H. (1991). Pose determination from line-to-plane correspondences: existence condition and closed-form solutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):530–541.
- Farrell, J. L., Stuelpnagel, J. C., Wessner, R. H., Velman, J. R., and Brook, J. E. (1966). A Least Squares Estimate of Satellite Attitude (Grace Wahba). *SIAM Review*, 8(3):384–386.
- Forstner, W. and Khoshelham, K. (2017). Efficient and Accurate Registration of Point Clouds with Plane to Plane Correspondences. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 2165–2173. IEEE.
- Grimson, W. E. L. and Lozano-Pérez, T. (1984). Model-Based Recognition and Localization from Sparse Range or Tactile Data. *The International Journal of Robotics Research*, 3(3):3–35.
- Jelinek, A. (2018). On registration of vector maps with known correspondences. In *2018 ELEKTRO*, pages 1–6. IEEE.
- Kabsch, W. (1976). A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5):922–923.
- Kamgar-Parsi, B. and Kamgar-Parsi, B. (2011). Matching 2D image lines to 3D models: Two improvements and a new algorithm. In *CVPR 2011*, pages 2425–2432. IEEE.
- Kocmanova, P., Zalud, L., and Chromy, A. (2013). 3D proximity laser scanner calibration. In *2013 18th International Conference on Methods & Models in Automation & Robotics (MMAR)*, pages 742–747. IEEE.
- Li, X., Zhang, Y., Shang, Y., and Liu, H. (2016). Probabilistic approach for maximum likelihood estimation of pose using lines. *IET Computer Vision*, 10(6):475–482.
- Markley, F. L. and Mortari, D. (2000). How to estimate attitude from vector observations. *Advances in the Astronautical Sciences*, 103(PART III):1979–1996.
- Olsson, C., Kahl, F., and Oskarsson, M. (2006). The Registration Problem Revisited: Optimal Solutions From Points, Lines and Planes. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06)*, volume 1, pages 1206–1213. IEEE.
- Pomerleau, F., Colas, F., and Siegwart, R. (2015). A Review of Point Cloud Registration Algorithms for Mobile Robotics. *Foundations and Trends in Robotics*, 4(1):1–104.
- Ramalingam, S., Taguchi, Y., Marks, T. K., and Tuzel, O. (2010). P2Pi: A Minimal Solution for Registration of 3D Points to 3D Planes BT - *Computer Vision/ECCV ... Computer Vision/ECCV ...*, 6315(Chapter 32):436–449.
- Tam, G. K. L., Zhi-Quan Cheng, Yu-Kun Lai, Langbein, F. C., Yonghuai Liu, Marshall, D., Martin, R. R., Xian-Fang Sun, and Rosin, P. L. (2013). Registration of 3D Point Clouds and Meshes: A Survey from Rigid to Nonrigid. *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1199–1217.
- Wahba, G. (1965). A Least Squares Estimate of Satellite Attitude. *SIAM Review*, 7(3):409–409.
- Wientapper, F. and Kuijper, A. (2017). Unifying Algebraic Solvers for Scaled Euclidean Registration from Point, Line and Plane Constraints. In *Lecture Notes in Computer Science*, pages 52–66. Springer International Publishing.
- Zalud, L., Kocmanova, P., Burian, F., Jilek, T., Kalvoda, P., and Kopecny, L. (2015). Calibration and Evaluation of Parameters in A 3D Proximity Rotating Scanner. *Elektronika ir Elektrotechnika*, 21(1):3–12.