

Iterative Closest Point Algorithm. L_1 Error Approach via Linear Programming

Yaroslav O. Halchenko[†]

[†]PhD Student, CS@NJIT
yh42@njit.edu

Abstract

The original goal of this project was a straightforward implementation of Iterative Closest Point ICP algorithm to coregister two point clouds. Original ICP uses squared-error function as a criterion of registration performance. In the current project we attempted to come up with an approach which would give us a stable unique solution for rigid co-registration problem while having sum of absolute values as an optimization criterion. As simple simulation results show, L_1 estimate can be more robust in the presence of strong outliers. Advantages and drawbacks of presented method are presented in this paper.

1 Introduction

My main topic of research lies in developing methods for integration of functional brain imaging modalities (IM) (EEG, MRI, fMRI, PET), which is a very challenging task for many people involved in this field. 'Fusion' algorithms are employed as an attempt to construct a spatio-temporal estimate of neuronal activity using data gathered from multiple functional brain imaging modalities. For more information you can refer to [3].

One of the basic preprocessing steps for analysis of any couple of images is co-registration, when one needs to find a transformation to bring second image to the space of the first one. Multiple methods has been developed. It is easy to subdivide them into different classes. There is a class of methods which are usually used to register similar IM of the similar nature (MRI, fMRI, PET): most of them do not use any explicit 3D structure information and based solely on per-voxel values' statistics. Usually minimization of some global error function of misfit is performed (squared error, entropy etc). Another approach is used to register IMs, which do not produce well defined 3D brain 'picture' (EEG, MEG), but rather gather information on the state of the brain activations as viewed from outside of the head. In such case methods common to Computer Vi-

sion are used to find rigid body transformation (rotation + translation) which registers reference coordinate systems of two experiments.

The task to register coordinate system between EEG/MEG and MRI experiments is particularly important, since MRI provides structural information on the gain fields for solving inverse and forward tasks in EEG/MEG brain imaging. Current technology doesn't allow¹ to obtain both signals simultaneously, which makes registration process necessary.

One of the approaches in such cases became prominent after Besl and McKay [1] formalized a simple method ICP to align two clouds of points. In such case we need to align surface of the extracted from MRI (such as in Fig. 1) image scall (Fig. 2(a)) to the digitized brain surface (Fig. 2(b)) taken in the same coordinate system as EEG sensors.

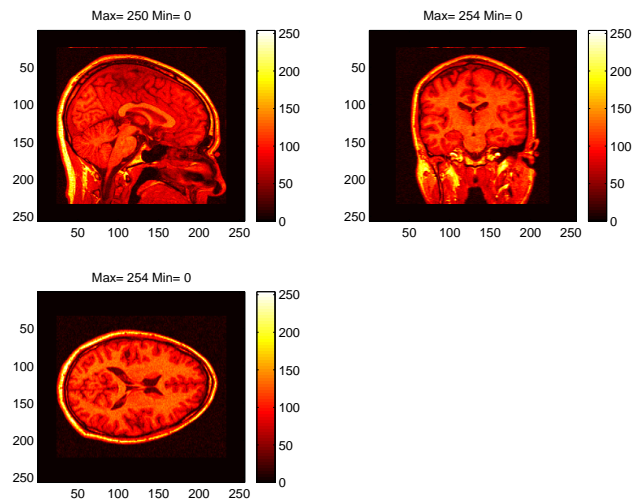


Figure 1: T1 MRI 'image' of the brain (unprocessed).

¹EEG came close to be recorded simultaneously with MRI but it is not perfect yet

2 ICP

Original ICP algorithm [1] is very simple and relies on previous work [6] as the method to find the best rigid body transformation.

ICP for two free-form shapes (model X , object P) can be formulated in common words by the next algorithm:

- Estimate initial transformation
- Iterate next steps to converge to local minima
 1. $\forall p \in P$ find closest point $x \in X$
 2. Find transformation Q which minimizes distances between each p and x
 3. Transform $P_{k+1} \leftarrow Q(P_k)$
 4. Terminate when change in the error falls below a preset threshold
- Choose the best among found solutions for different initial transformations

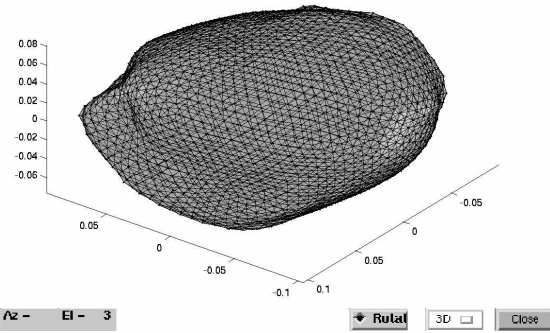
2.1 ICP Problems

While being very simple, ICP method has created a big wave of research to overcome its drawbacks:

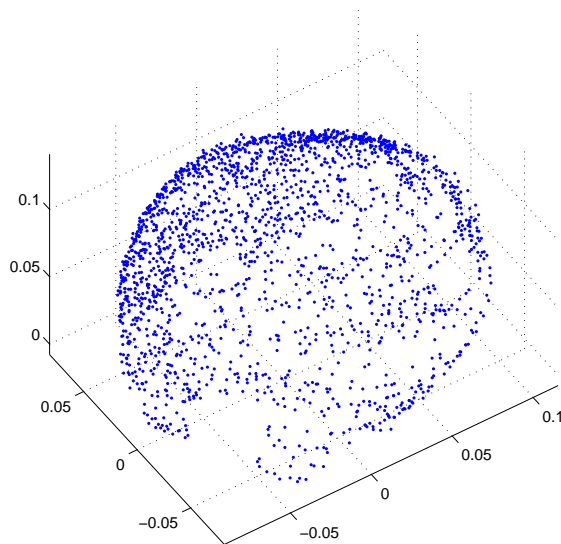
- it is quite computationally demanding to identify the closest point for each object point at each iteration;
- converges to local minima, which might be a global minimum under 'lucky' initial transformation;
- doesn't converge very fast;
- doesn't work properly for some specific shapes (as was mentioned in original Besl and McKay [1] paper), though it is a minor and rare problem;
- can be sensitive to outliers.

Multiple papers [e.g., 7] provided detailed overviews of existing approaches and usually present themselves a new one on how to improve ICP. Most of them have done work in attempt

- to speed-up closest point selection
 - K-d trees, dynamic caching
 - sampling of model and object points
- to avoid local minima
 - removal of outliers
 - stochastic ICP, simulated annealing, weighting



(a) Scaled of the brain extracted from T1 MRI.



(b) Digitized head surface from EEG experiment

Figure 2: Data to be registered.

- use other metrics (point-to-surface vs -point)
- use additional information besides geometry (color, curvature)

Nevertheless nobody tried (or at least didn't report) to try L_1 norm (sum of absolute distances) as an error function. Such error measure brought interesting results in methods such as solving inverse problem for EEG/MEG data analysis [2].

2.2 Matlab Implementation

We have done a simple straightforward implementation of ICP algorithm, source scripts for which you can find on our CIS780 class's project web page [4]. ICP implementation wasn't the crucial part of the project therefore we didn't bring anything innovative - it was a simplistic implementation, because we wanted to spend more time to make L_1 solver work - a task far more interesting than just re-implementing somebodies ideas. So Matlab code for ICP works and has a choice of selecting either default L_2 (squared mean error) error function or L_1 criterion. The functions such as `icp_closest.m` and `icp_initial_align.m` can be modified to easily incorporate some other innovative ideas to overcome problems mentioned in Section 2.1.

3 L_1 Error Minimization

Minimization of sum of absolute distances is much trickier than minimization of sum of squares due to the fact that no efficient and simple approach was ever devised which would lead to unique desired solution. The problem is that $f(x) = |x|$ isn't that easy to handle, it doesn't have 1st derivative defined at the point $x = 0$, so somebody need to use some kind of parametric presentation to approximate this function around 0. But then such solver might have 0 as unstable point of the involved optimization procedure.

To overcome this problem we decided to present given problem as a convex and solvable by efficient Linear Programming problem solver. Matlab has a nice solver shipped with it, but there are many better ones available. We were using MOSEK (<http://www.mosek.com/>) which has free education license and implements internal-point LP solver, which is very efficient for sparse large LP problems. Our presentation of the problem is really sparse and large so this solver is a perfect choice in our case.

The target of the work is to minimize objective function $C = \sum_i \|\vec{p}_i - \vec{x}_i\|$ under rigid-body transformation $\vec{p}_i = R\vec{p}_i + \vec{t}$. So we have unknown parameters of rigid

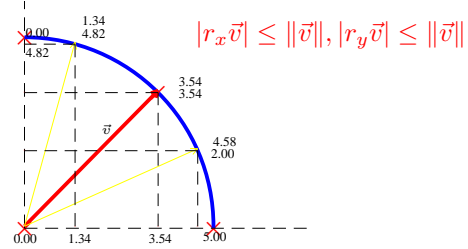


Figure 3: Intuition for Finding Vector Norm Using LP

transformation: rotation matrix R and translation vector \vec{t} .

Next subsections will cover approaches how to present different non-LP terms suitable for LP solver.

3.1 LP Problem Presentation

3.1.1 LP: Absolute Value

$y = |x|$ is commonly presented in LP using some auxiliary variables x^+ and x^- but we decided to go with an equivalent presentation that

$$(y = |x|) \equiv (x \leq y \text{ and } -x \leq y, \text{ while minimizing } y)$$

Current form does not introduce any new variables but enforces minimization of y to get the proper value of $|x|$. In our case it is Ok, because we want to minimize sum of absolute values anyway :-)

3.1.2 LP: Euclidean Norm Approximation

The more intriguing problem is how to present vector norm using LP. To overcome this difficulty we use simple trigonometry. If we take any vector on the plane (in 2D) and start rotating it around the zero-point, at some rotation angle it will be closer or even coincide with one of the axis, so projection of the vector on that axis will roughly correspond to its norm.

Thus for instance Fig. 3 illustrates 2D for vector \vec{v} . If we perform enough rotations, maximal projection will be a good approximation of the vector norm.

For 3D case we have two choices

1. first do 2D vector norm computation in XY coordinate plane using just $\overrightarrow{(v_x, v_y)}$, thus obtaining v_{xy} and then use the same procedure to find $\|\vec{v}\| = \|\overrightarrow{(v_{xy}, v_z)}\|$;

2. go to 3D and do rotations for the set of 3 angles to create Euler rotation matrices, and then use all 3 projections instead of 2 for 2D case.

Both of these methods can give you a very precise approximation if the number of rotations is high. You can see comparison results on Fig. 4. Results suggest that 2D rotations are more efficient (has less inequalities, so LP solver find solution faster) than using 3D rotations to get the same quality of approximation. Further in this work we used only 2 consecutive 2D rotations to approximate vector norm in 3D.

3.1.3 LP: Rigid Transformation

One of the main problem occurred while working on the project is the necessity to constrain 9-elements of matrix R to correspond to a rotation matrix, so $RR^T = I$, or in other words to preserve all distances between object points after the transformation.

We observe, that triangle is a minimal (minimal number of edges) rigid figure in 2D, so it can't be stretched or smeared without changing length of at least one of its sides. For 3D it would be tetrahedron, which has 4 sides connecting 4 points (all-to-all).

Therefore in order to make sure that R is the rotation matrix we need to preserve distances between 4 points of some arbitrary tetrahedron T :

$$\|\vec{p}_j - \vec{p}_k\| - \|\vec{p}_j - \vec{p}_k\| = 0 \quad \vec{p}_i, \vec{p}_j \in T$$

3.2 LP Problem Formulation

Having presented all the 'tricks', now we can present our problem in LP form ie as a set of equations under a set of constraints

$$\begin{aligned} \vec{p} &= R\vec{p} + \vec{t} \\ \|\vec{p}_i - \vec{x}_i\| - d_i &= 0 \quad \forall i, \text{ s.t. } \vec{p}_i \in P, \vec{x}_i \in X \\ \|\vec{p}_j - \vec{p}_k\| - \|\vec{p}_j - \vec{p}_k\| &= 0 \quad \vec{p}_i, \vec{p}_j \in T \end{aligned}$$

Objective: to minimize $C = \sum_i d_i$, where $d_i \geq 0$

3.3 LP Problems and Drawbacks

- Contraction (shrinking):

$$\|\vec{p}_j - \vec{p}_k\| - \|\vec{p}_j - \vec{p}_k\| = 0 \quad \vec{p}_i, \vec{p}_j \in T$$

is actually

$$\|\vec{p}_j - \vec{p}_k\| - \|\vec{p}_j - \vec{p}_k\| \leq 0$$

This is due to the next fact: we can enforce that our object does not stretch but we cannot prevent it from contracting. It follows from the convexity of my presentation of the $|x|$. This problem keeps me from using suggested here L_1 error norm in ICP for now because in case of big miss-alignment between object and model, ICP might tend to contract the object, because then it will 'fit' the model better.

Nevertheless, proposed method works when object is wrapped within the model object, which can happen when we register skull to skull for instance.

- R matrix needs to be "normalized" to the nearest orthonormal matrix due to our $\|x\|$ LP approximation even if no contraction occurred. Horn et al. [5] derived the 'closest rotation matrix' in case of minimal distortion on the residuals for the solution in squared-error sense. We used a simple SVD method by normalizing all of singular values to be unites and recomposing back rotation matrix.

3.4 Experiments

To illustrate the gain from usage of L_1 (sum of absolute values) instead of commonly used L_2 (sum of squares), we've created a simple example - half-sphere model and inscribed half-sphere noisy object with different number of outliers (see Fig. 5). This artificial data was used to compare registration produced by two methods.

Perfect registration would lead to identity rotation matrix and zero translation vector. Sum of absolute values of off-diagonal elements of rotation matrix R and sum of absolute values of translation vector \vec{t} is presented on Fig. 6, which shows that L_1 norm solution is more stable - it didn't produce a lot of unnecessary rotation and much less of translation for quite a long range of present outliers.

4 Conclusion

This project is an attempt to introduce L_1 norm to characterize the error of registration for two points clouds, which can be used in ICP algorithm. Simple presented simulation showed plausibility and advantage of L_1 in place of L_2 norm for registration error.

Presented approach has its drawback. It is impossible to impose rigid constraint in general case, but it is possible to use current method for non-contracting transformations.

Modification of LP solver with explicit consideration of absolute value of variable $|x|$ might make it possible to

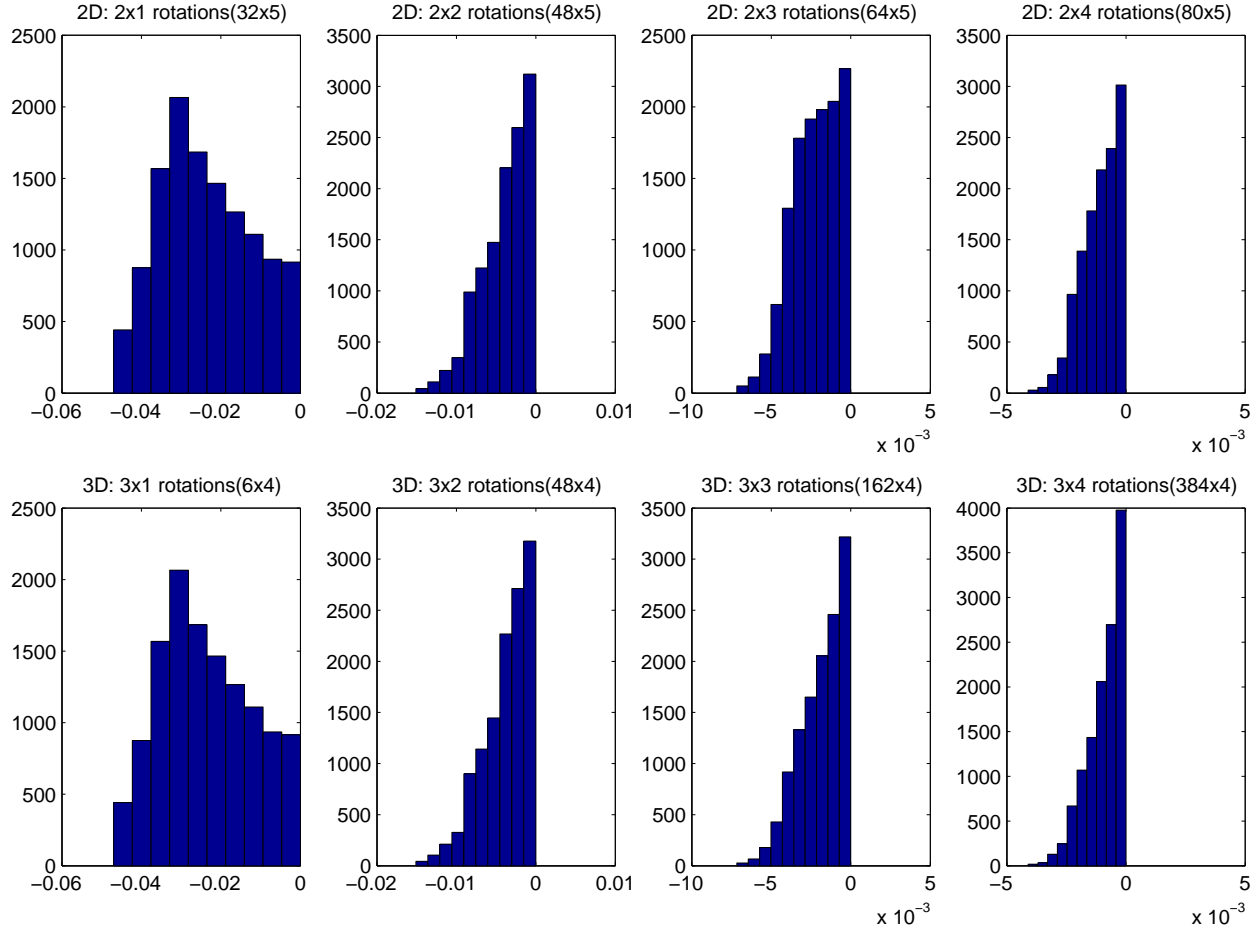


Figure 4: Comparison between 2 consecutive 2D approximations to compute vector norm and single 3D. 1st row corresponds to the first method - 2 consecutive approximations in 2D. 2nd row corresponds to rotations in 3D and considering projections on all 3 axes simultaneously. Different columns correspond to different number of rotations. For 2D case it is a number of rotations for each vector in the plane, and for 3D it corresponds for number of rotations for each Eulerian angle. Numbers in paratneses correspond to total number of inequalities x variables. For 2D rotations variables are $(v_x, v_y, v_z, v_{xy}, \|\vec{v}\|)$ and $(v_x, v_y, v_z, \|\vec{v}\|)$ for 3D rotations.

overcome existing drawback but as to my knowledge no such solver exists yet.

References

- [1] Besl, P. and McKay, N. (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2).
- [2] Fuchs, M., Wagner, M., Kohler, T., and Wischmann, H. A. (1999). Linear and nonlinear current density reconstructions. *J Clin Neurophysiol.*, 16(3):267–295.
- [3] George, J., Schmidt, D., Rector, D., and Wood, C. (2002). *Functional MRI: An Introduction to Methods*, chapter 19. Dynamic functional neuroimaging intergratin multiple modalities, pages 353–382. Oxford University Press.
- [4] Halchenko, Y. (2003). Simple implementation of ICP using Matlab. attempt to devise l_1 fit error solution. <http://www.onerussian.com/classes/cis780>.
- [5] Horn, B., Hilden, H., and Negahdaripour, S. (1998). Closed-form solution of absolute orientation using orthonormal matrices. *JOSA A*, 5(7).
- [6] Horn, B. K. P. (1987). Closed-form solution of absolute orientation using unit quaternions. *Journal of Optical Society of America*, 4(4):629–642.
- [7] Ourselin, S., Stefanescu, R., and Pennec, X. (2002). Robust registration of multi-modal images: Towards real-time clinical applications. *Lecture Notes in Computer Science*, 2489:140–??

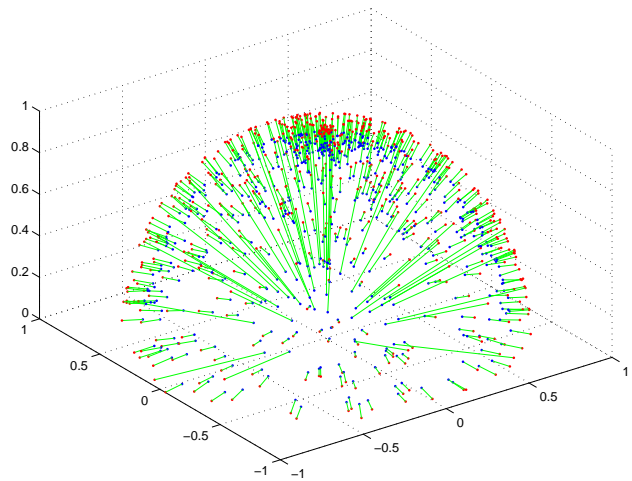


Figure 5: Simulated data. Red dots belong to the model and blue ones form the object. Each corresponding pair is connected by a green line. There are multiple strong outliers which lead object deeper to the sphere.

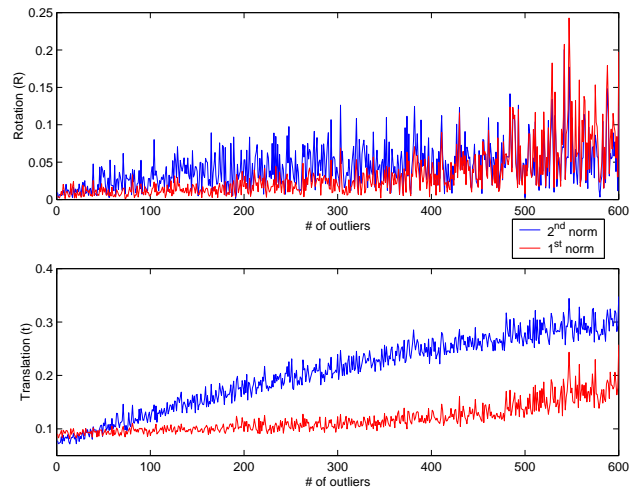


Figure 6: Sum of absolute values of off-diagonal elements of rotation matrix (upper graph) and translation vector, obtained using usual mean-squared error registration (blue) and absolute distances registration (red) for different number of present strong outliers.