Efficient, General Point Cloud Registration with Kernel Feature Maps

Hanchen Xiong, Sandor Szedmak, Justus Piater

Institute of Computer Science
University of Innsbruck

30 May 2013
Outline

1. Background
2. Rigid Transformation in Hilbert Space
3. Rigid Transformation in $\mathbb{R}^3$
4. Experiment Results
5. Conclusion
1. Background
Problem statement

- **3D point cloud registration**
  
  Given two point clouds $X_1 = \{x_{i}^{(1)}\}_{i=1}^{l_1}$, $X_2 = \{x_{j}^{(2)}\}_{j=1}^{l_2}$, find the correct correspondences between $x_{i}^{(1)}$ and $x_{j}^{(2)}$, based on which two point clouds can be aligned.
Related Work

Registration

- **Iteration Closest Point (ICP)**;
  - match nearest neighbours as correspondences ⇔ minimize the distances between correspondences

- **Gaussian Mixture**;
  - fit point clouds to distributions + correlation, L2 distance or kernel methods

- **SoftAssign / EM-ICP**
  - one-to-many correspondences
  - optimize w.r.t. correspondence matrix ⇔ optimize w.r.t. transformation.
\begin{equation}
\{R^*, b^*\} = \arg\min_{R, b} \sum_{i=1}^{l_1} \sum_{j=1}^{l_2} \left( R x_i^{(1)} + b - x_j^{(2)} \right)^2 w_{i,j}
\end{equation}

where $R, b$ denote rotation and translation in $\mathbb{R}^3$.

- **ICP**: $w_{i,j} \in \{0, 1\}$, determined by shortest-distance criterion;
- **Guassian Mixtures**: and $w_{i,j} = \frac{1}{l_1 l_2}$ for all $i, j$ (uniformly);
- **SoftAssign/EM-ICP**: $w_{i,j}$ is interpreted as the probability of the correspondence;
2. Transformation in Hilbert Space
By applying a kernel function on 3D points $K(x_i, x_j)$, a $\mathbb{R}^3 \rightarrow \mathcal{H}$ feature map $\phi$ is implicitly induced:

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$$ (2)

and $\mathcal{H}$ is called Hilbert space, which is usually much higher or possibly infinite dimensional:

$$K(x_i, x_j) = e^{-\|x_i - x_j\|^2/2\sigma^2} \quad \rightarrow \quad \phi(x_i) \propto f(\xi) = e^{-\|\xi - x_i\|^2/2\sigma^2}$$ (3)
Gaussian in Hilbert Space

mean:

\[ \mu_{\mathcal{H}}^{(1)} = \frac{1}{l_1} \sum_{i=1}^{l_1} \phi(x_i^{(1)}) \]  
\[ \mu_{\mathcal{H}}^{(2)} = \frac{1}{l_2} \sum_{i=1}^{l_2} \phi(x_i^{(2)}) \]  

(4)

(5)

covariance:

\[ C_{\mathcal{H}}^{(1)} = \frac{1}{l_1} \sum_{i=1}^{l_1} \left( \phi(x_i^{(1)}) - \mu_{\mathcal{H}}^{(1)} \right) \left( \phi(x_i^{(1)}) - \mu_{\mathcal{H}}^{(1)} \right)^\top \]  
\[ C_{\mathcal{H}}^{(2)} = \frac{1}{l_2} \sum_{i=1}^{l_2} \left( \phi(x_i^{(2)}) - \mu_{\mathcal{H}}^{(2)} \right) \left( \phi(x_i^{(2)}) - \mu_{\mathcal{H}}^{(2)} \right)^\top \]  

(6)

(7)
Kernel PCA

Assume all points are already centralized:

\[
C = \frac{1}{l} \sum_{i=1}^{l} \phi(x_i)\phi(x_i)^\top
\]  

(8)

the none-zero eigenvalue \(\lambda_k\) and corresponding eigenvector \(u_k\) of \(C\) should satisfy:

\[
\lambda_k u_k = Cu_k
\]  

(9)

by substituting (8) into (9), we can have:

\[
u_k = \frac{1}{\lambda_k} Cu_k = \sum_{i=1}^{l} \alpha_i^k \phi(x_i)
\]  

(10)

where \(\alpha_i^k = \frac{\phi(x_i)^\top u_k}{\lambda_k l}\). Therefore, all eigenvectors \(u_k\) with \(\lambda_k \neq 0\) must lie in the span of \(\phi(x_1), \phi(x_2), \ldots, \phi(x_l)\), and (10) is referred to as the dual form of \(u_k\).
Kernel PCA, cont.

By left multiplying $\sum_{j=1}^I \phi(x_j)^\top$ on both sides of equation (9):

$$\sum_{j=1}^I \phi(x_j)^\top \lambda_k u_k = \sum_{j=1}^I \phi(x_j)^\top C u_k$$

$\iff$ \quad $\lambda_k \sum_{i,j=1}^I \alpha_i^k \langle \phi(x_i), \phi(x_j) \rangle = \frac{1}{I} \sum_{i,j=1}^I \alpha_i^k \langle \phi(x_i), \phi(x_j) \rangle^2$

$\iff$ \quad $\lambda_k \sum_{i,j=1}^I \alpha_i^k K(x_i, x_j) = \frac{1}{I} \sum_{i,j=1}^I \alpha_i^k K(x_i, x_j)^2$

$\iff \underbrace{\lambda_k}_{\eta_k} \alpha^k = K \alpha^k$

it can be seen that $\{\eta_k, \alpha^k\}$ is actually an eigenvalue-eigenvector pair of matrix $K$. In this way, the eigenvector decomposition of bilinear form $C$ in $\mathcal{H}$ can be transformed to the decomposition of matrix $K$. 
Figure: (a) A point cloud of table tennis racket; (b–d) reconstruction using the first 1–3 principal components. For each point in the bounding-box volume, the darkness is proportional to the density of the Gaussian in the feature space $\mathcal{H}$. 
Un-centralized Case

\[ u^k = \sum_{i=1}^{l} \alpha_i^k (\phi(x_i) - \mu) \]
\[ = \sum_{i=1}^{l} \alpha_i^k \left[ \phi(x_i) - \frac{1}{l} \sum_{m=1}^{l} \phi(x_m) \right] \]
\[ = \phi(M)^\top (I_l - \frac{1}{l} 1_l 1_l^\top) \alpha^k \]

(12)

(\text{where } \phi(M)^\top = [\phi(x_1), \phi(x_2), \ldots, \phi(x_l)], \ 1_l \text{ is a } l \text{ dimension vector with all entry equal } 1, \ I_l \text{ is } l \times l \text{ identity matrix, } \alpha^{k\top} = [\alpha_1^k, \alpha_2^k, \ldots, \alpha_l^k] )

\[ u^k \top u^h = 0, \quad \forall k \neq h \]

(13)

\[ u_1^k = \phi(M_1)^\top I_1^E \alpha^k \]

(14)

\[ u_2^k = \phi(M_2)^\top I_2^E \alpha^k \]

(15)
Rotation in Hilbert Space

Only $D$ eigenvectors are used to represent the covariance of high dimension Gaussian distribution of each point cloud:

$$U_1 = \begin{bmatrix} u_1^1, \cdots, u_k^1, \cdots, u_D^1 \end{bmatrix}$$  \hspace{1cm} (16)

$$U_2 = \begin{bmatrix} u_1^2, \cdots, u_k^2, \cdots, u_D^2 \end{bmatrix}$$  \hspace{1cm} (17)

Align $U_1$ with $U_2$: $U_2 = R_\mathcal{H} U_1$

$$U_2 = R_\mathcal{H} U_1 \quad \iff \quad U_2 U_1^\top = R_\mathcal{H} U_1 U_1^\top$$  \hspace{1cm} (18)

$$\iff \quad R_\mathcal{H} = U_2 U_1^\top = \sum_{k=1}^{D} u_k^2 u_k^1 \top$$

$$= \phi(M_2)^\top I_2^E \left( \sum_{k=1}^{D} \alpha_k^2 \alpha_k^1 \top \right) I_1^E \phi(M_1)$$  \hspace{1cm} (19)
Translation in Hilbert Space

if $M_1$ has already been centered, i.e. $\mu^{(1)}_H = 0$

$$b_H = \mu^{(2)}_H = \frac{1}{l_2} \phi(M_2)^\top 1_{l_2}$$  \hspace{1cm} (20)
3. Rigid Transformation in $\mathbb{R}^3$
Consistency

consistency error:

\[
\begin{align*}
x_t^{(1)} & \xrightarrow{\phi} \phi(x_t^{(1)}) \\
\{R, b\} \downarrow & \phi \downarrow \{R_H, b_H\} \\
R x_t^{(1)} + b & \xrightarrow{\phi} \phi(R x_t^{(1)} + b) \sim R_H \tilde{\phi}(x_t^{(1)}) + b_H \\
\{R^*, b^*\} & = \arg \min_{R, b} \frac{1}{l} \sum_{t=1}^{l} \| \Psi_t - \Phi_t \|^2 \quad (21)
\end{align*}
\]

Because \( \| \Phi(x) \|^2 \) can preserve constant under any translation \( b \) and rotation \( R \), and \( \Psi_t \) is fixed, :

\[
\{R^*, b^*\} = \arg \max_{R, b} \frac{1}{l} \sum_{t=1}^{l} \Phi_t^\top \Psi_t \quad (22)
\]
Objective Function

\[ \{R^*, b^*\} = \underset{R, b}{\arg \max} \frac{1}{l_1} \sum_{t=1}^{l_1} \Phi_t^\top \Psi_t \]

\[ O = \frac{1}{l_1} \sum_{t=1}^{l_1} \left\{ \Phi_t^\top \left[ \phi(M_2)^\top \gamma \phi(M_1) \left( \phi(x_t^{(1)}) - \frac{1}{l_1} \phi(M_1)^\top 1_{l_1} \right) \right] + \frac{1}{l_2} \phi(M_2)^\top 1_{l_2} \right\} \]

\[ = \frac{1}{l_1} \sum_{t=1}^{l_1} K(Rx_t^{(1)} + b, M_2)^\top \left[ \gamma \left( K(x_t^{(1)}, M_1) - \frac{1}{k_1} K_{1l_1} \right) + \frac{1}{l_2} 1_{l_2} \right] \]

\[ = \frac{1}{l_1} \sum_{t=1}^{l_1} \sum_{i=1}^{l_2} K(Rx_t^{(1)} + b, x_i^{(2)}) \rho_{t,i} \]
only a small number of points \( D + 1 \) is enough:

\[
\{ R^*, b^* \} = \arg \max_{R, b} \frac{1}{D + 1} \frac{1}{l_2} \sum_{t=1, i=1}^{D+1, l_2} K(Rx_S^{(1)} + b, x_i^{(2)}) \rho_{t,i}
\]  

(25)

where \( S \) denotes the randomly selected subset of \( M_1 \)
Implicit Correspondence

Figure: (a) Two identical point clouds with exactly the same point permutation. (b) Visualization of $\rho_t$ computed for all pairs of points.
Relation to Other Approaches

- our method:
  \[
  \{R^*, b^*\} = \arg\max_{R,b} \frac{1}{D+1} \frac{1}{l_2} \sum_{t=1,i=1}^{D+1,l_2} K(R_{S_t^1} + b, x_{i}^{(2)}) \rho_{t,i} \]  

- SoftAssign /EM-ICP
  \[
  \{R^*, b^*\} = \arg\min_{R,b} \frac{1}{l_1} \sum_{t=1}^{l_1} \sum_{i=1}^{l_2} - \log K(R_{x_t^{(1)}} + b, x_{i}^{(2)}) w_{t,i} \]  

- Gaussian Mixtures
  \[
  \{R^*, b^*\} = \arg\max_{R,b} \frac{1}{l_1} \sum_{t=1}^{l_1} \sum_{i=1}^{l_2} K(R_{x_t^{(1)}} + b, x_{i}^{(2)}) \]
Relation to Other Approaches, cont.

pseudo Gaussian mixture alignment:

\[ u_1^k = \phi(M_1)^\top I_1^E \beta_k \]
\[ = \sum_{i=1}^{l_1} \beta_i^k \phi(x_i^{(1)}) \]
\[ = \sum_{i=1}^{l_1} \tilde{\beta}_i^k N(\xi; x_i^{(1)}, \sigma) \]

Remark:
- pseudo Gaussian mixture: \( \tilde{\beta}_i^k \) can be negative;
- \( D \) pseudo Gaussian mixtures are aligned simultaneously.
Qualitative Experiments

Figure: Test of the proposed algorithm in typical challenging circumstances for registration: (a) large motion; (b) outliers; (c) nonrigid transformation
Figure: More test results on KIT 3D object database
Quantitative Experiments

Accuracy and Robustness

Figure: Test of four registration algorithm on (a) different scales of motions; (b) different portion of outliers added.
### Efficiency

<table>
<thead>
<tr>
<th>Point cloud size $n$</th>
<th>complexity</th>
<th>200</th>
<th>500</th>
<th>1000</th>
<th>2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our method</td>
<td>$n \log n$</td>
<td>1.489</td>
<td>2.162</td>
<td>5.126</td>
<td>21.165</td>
</tr>
<tr>
<td>ICP [BM92]</td>
<td>$n \log n$</td>
<td>0.023</td>
<td>0.051</td>
<td>0.154</td>
<td>0.469</td>
</tr>
<tr>
<td>GaussianMixtures [JV11]</td>
<td>$n^2$</td>
<td>3.998</td>
<td>15.245</td>
<td>43.570</td>
<td>172.4</td>
</tr>
<tr>
<td>SoftAssign [GRLM97]</td>
<td>$n^2$</td>
<td>4.801</td>
<td>83.925</td>
<td>592.1</td>
<td>3812</td>
</tr>
</tbody>
</table>

**Table:** Average execution time (seconds)
Conclusion

- kernel feature map point cloud to Hilbert space;
- align projections of point clouds in Hilbert space;
- project alignment back to $\mathbb{R}^3$;
- accurate and robust to large motion and outliers;
- much faster than state-of-the-art methods;
END

Questions are welcome!
P. J. Besl and H. D. Mckay.
A Method for Registration of 3-D Shapes.

Steven Gold, Anand Rangarajan, Chienping Lu, and Eric Mjolsness.
New Algorithms for 2D and 3D Point Matching: Pose Estimation and Correspondence.

Bing Jian and Baba C. Vemuri.
Robust Point Set Registration Using Gaussian Mixture Models.
Computation Complexity Reduction

\[
\langle \Phi_t, \Psi_t \rangle = \phi(\mathbf{P} x_1^{(1)})^\top \left( \sum_{k=1}^D \tilde{u}_2^k \tilde{u}_1^k \top \left( \phi(x_1^{(1)}) - \mu_1 \right) + \mu_2 \right) \\
= \sum_{k=1}^D \langle \tilde{u}_2^k, \phi(\mathbf{P} x_1^{(1)}) \rangle \langle \tilde{u}_1^k, \phi(x_1^{(1)}) - \mu_1 \rangle + \langle \mu_2, \phi(\mathbf{P} x_1^{(1)}) \rangle \\
= \sum_{k=1}^D \langle \tilde{u}_2^k, \phi(\mathbf{P} x_1^{(1)}) \rangle \langle \tilde{u}_2^k, \mathbf{R}_\mathcal{H} \phi(x_1^{(1)}) - \mu_1 \rangle + \langle \mu_2, \phi(\mathbf{P} x_1^{(1)}) \rangle
\] (30)

where we can see that \( \phi(\mathbf{P} x_1^{(1)}) \) and \( \mathbf{R}_\mathcal{H} \phi(x_1^{(1)}) - \mu_1 \) are projected onto \( D \) eigenvectors \( \{ \tilde{u}_2^k \}_{k=1}^D \) respectively, and an additional projection of \( \phi(\mathbf{P} x_1^{(1)}) \) onto \( \mu_2 \). Therefore, the computation of the objective function is actually done in a space spanned by \( D \) eigenvectors \( \{ \tilde{u}_2^k \}_{k=1}^D \) and one \( \mu_2 \), which is a subspace of \( \mathcal{H} \).