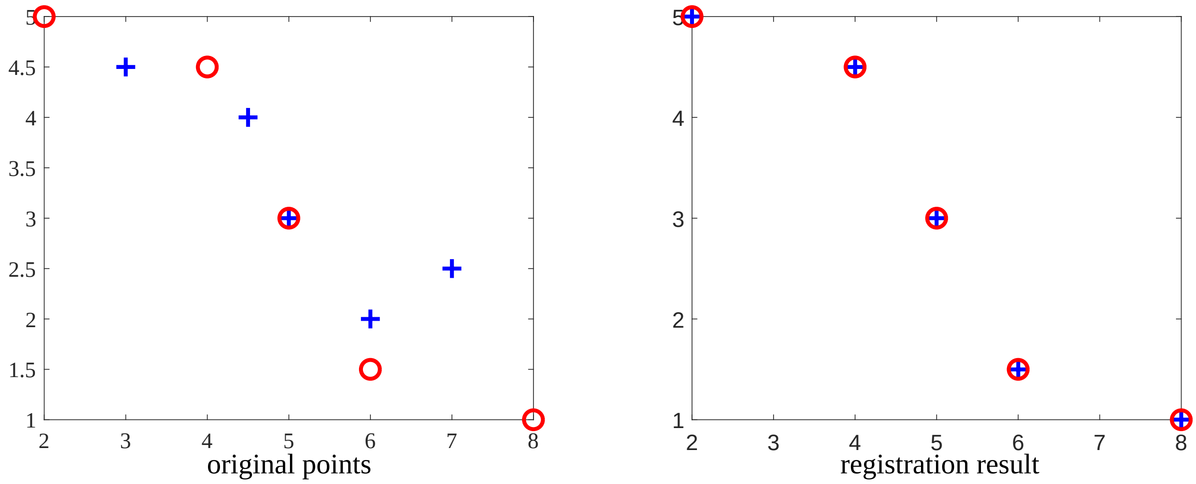
A numerical example

We provide a simple numerical example to illustrate the ultrarobust support vector registration (Ur-SVR) algorithm.



(a) (b)

Figure 1: A numerical example for Ur-SVR. The goal is to warp the source point set ('+') to the target point set ('○'). The initial state of the point sets is presented in (a). The registration result is presented in (b).

Let $\boldsymbol{X}$ be the source point set and $\boldsymbol{Y}$ be the target point set:

$$\boldsymbol{X} = \begin{bmatrix} 3.0 & 4.5 \\ 4.5 & 4.0 \\ 5.0 & 3.0 \\ 6.0 & 2.0 \\ 7.0 & 2.5 \end{bmatrix}, \qquad \boldsymbol{Y} = \begin{bmatrix} 2.0 & 5.0 \\ 4.0 & 4.5 \\ 5.0 & 3.0 \\ 6.0 & 1.5 \\ 8.0 & 1.0 \end{bmatrix},$$

where they are expressed as 5×2 matrices, and each row represents the coordinate of a point. The goal is to match $\boldsymbol{X}$ to $\boldsymbol{Y}$. Since the points in $\boldsymbol{X}$ and $\boldsymbol{Y}$ are few, we use all points throughout the whole matching process by setting the coarse-to-fine threshold $\delta = 100\%$ and the number of basis functions $\Psi = 5$. The matching procedure is introduced as follows:

1. Build the feature correspondences $\boldsymbol{\mathcal{M}}$ based on the SC descriptor:

$$\boldsymbol{\mathcal{M}} = \left\{ \left(\begin{bmatrix} 3.0 \\ 4.5 \end{bmatrix}, \begin{bmatrix} 2.0 \\ 5.0 \end{bmatrix} \right), \left(\begin{bmatrix} 4.5 \\ 4.0 \end{bmatrix}, \begin{bmatrix} 4.0 \\ 4.5 \end{bmatrix} \right), \left(\begin{bmatrix} 5.0 \\ 3.0 \end{bmatrix}, \begin{bmatrix} 5.0 \\ 3.0 \end{bmatrix} \right), \left(\begin{bmatrix} 6.0 \\ 2.0 \end{bmatrix}, \begin{bmatrix} 6.0 \\ 1.5 \end{bmatrix} \right), \left(\begin{bmatrix} 7.0 \\ 2.5 \end{bmatrix}, \begin{bmatrix} 8.0 \\ 1.0 \end{bmatrix} \right) \right\}.$$

2. Normalize data to have zero mean and unit variance before transformation estimation:

$$\boldsymbol{X} = \begin{bmatrix} -1.8074 & 1.1189 \\ -0.5164 & 0.6885 \\ -0.0861 & -0.1721 \\ 0.7746 & -1.0328 \\ 1.6353 & -0.6025 \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} -1.6641 & 1.1094 \\ -0.5547 & 0.8321 \\ 0 & 0 \\ 0.5547 & -0.8321 \\ 1.6641 & -1.1094 \end{bmatrix}.$$

3. Compute the Laplacian matrix $\boldsymbol{L}$:

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

There are only five points in $\boldsymbol{X}$ and they are far from each other, resulting in the obtained $\boldsymbol{L}$ being a zero matrix. A more detailed calculation process about $\boldsymbol{L}$ is presented at the end of the document.

4. Compute the Gram matrix $\boldsymbol{K}_\Psi$:

$$\boldsymbol{K}_\Psi = \left(e^{-\tau\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}\right)_{5\times5} = \begin{bmatrix} 1.0000 & 0.0246 & 0.0001 & 0 & 0 \\ 0.0246 & 1.0000 & 0.1569 & 0.0001 & 0 \\ 0.0001 & 0.1569 & 1.0000 & 0.0517 & 0.0018 \\ 0 & 0.0001 & 0.0517 & 1.0000 & 0.1569 \\ 0 & 0 & 0.0018 & 0.1569 & 1.0000 \end{bmatrix},$$

where $\tau = 2$, and $\boldsymbol{x}_i, \boldsymbol{x}_j \in \boldsymbol{X}$, $i, j = 1, 2, ..., 5$.

5. Determine parameter $\boldsymbol{\varepsilon}$ using Eq. (20) and Eq. (3): $\boldsymbol{\varepsilon} = 0.0205$. Note that all quoted formulas in the document come from the paper that proposes Ur-SVR.

6. Initialize the weight coefficients of the model in Eq. (11): $c_i = 1$, $i = 1, 2, ..., 5$.

7. Estimate the transformation f by the coordinate descent method under the current c_i, $i = 1, 2, ..., 5$:

   Initialize the constraint conditions of the model in Eq. (11): $\boldsymbol{r}_i = \mathbf{0}$ and $\hat{\boldsymbol{r}}_i = \mathbf{0}$, $i = 1, 2, ..., 5$.

   1) Because the use of the displacement enables greater robustness, we estimate a displacement function g, defined as $g(\boldsymbol{x}) = f(\boldsymbol{x}) - \boldsymbol{x}$, instead of estimating f directly. Accordingly, $\boldsymbol{Y}$ in Eq. (19) is assigned the difference of the target point set minus

the source point set. With $\boldsymbol{V} = \boldsymbol{K}_{\Psi}$, $\boldsymbol{U} = \boldsymbol{V}$, $\lambda_1 = 0.1$, and $\lambda_2 = 0.1$, we update the displacement function g (i.e., coefficient matrix $\bar{\boldsymbol{A}}$) using Eq. (19):

$$\bar{\boldsymbol{A}} = \begin{bmatrix} 0.1377 & -0.0118 \\ -0.0548 & 0.1168 \\ 0.1010 & 0.1345 \\ -0.2235 & 0.2626 \\ 0.0607 & -0.5223 \end{bmatrix}.$$

2) Update $\boldsymbol{r}_i$ and $\hat{\boldsymbol{r}}_i$, $i = 1, 2, ..., 5$ using Eq. (13):

$$\boldsymbol{R} = \begin{bmatrix} 0.0273 & 0.0199 \\ 0.0177 & 0.0263 \\ 0.0255 & 0.0272 \\ 0.0093 & 0.0336 \\ 0.0235 & 0 \end{bmatrix}, \quad \hat{\boldsymbol{R}} = \begin{bmatrix} 0.0136 & 0.0211 \\ 0.0232 & 0.0146 \\ 0.0154 & 0.0137 \\ 0.0316 & 0.0073 \\ 0.0174 & 0.0466 \end{bmatrix},$$

where $\boldsymbol{R} = (\boldsymbol{r}_1, \boldsymbol{r}_2, ..., \boldsymbol{r}_5)^T$ and $\hat{\boldsymbol{R}} = (\hat{\boldsymbol{r}}_1, \hat{\boldsymbol{r}}_2, ..., \hat{\boldsymbol{r}}_5)^T$.

Alternate between the above 1) and 2) until $\bar{\boldsymbol{A}}$ converges or the number of iterations reaches the maximum, obtaining:

$$\bar{\boldsymbol{A}} = \begin{bmatrix} 0.1155 & -0.0093 \\ -0.0417 & 0.0992 \\ 0.0800 & 0.1169 \\ -0.1935 & 0.2284 \\ 0.0459 & -0.4771 \end{bmatrix}.$$

Compute the transformed point set $f(\boldsymbol{X})$ using Eq. (18):

$$f(\boldsymbol{X}) = \boldsymbol{U}\bar{\boldsymbol{A}} + \boldsymbol{X} = \begin{bmatrix} -1.6929 & 1.1120 \\ -0.5427 & 0.8059 \\ -0.0225 & -0.0288 \\ 0.5924 & -0.8733 \\ 1.6509 & -1.0436 \end{bmatrix}.$$

8. Estimate the weight coefficients c_i, $i = 1, 2, ..., 5$ by the EM algorithm under the current f:

Initialize the model parameters $\Theta = \{\sigma_k^2, \pi_k\}_{k=1}^{3}$ in Eq. (5):

$$\boldsymbol{\sigma}^2 = \begin{bmatrix} 0.0010 \\ 0.0011 \\ 0.0012 \end{bmatrix}, \quad \boldsymbol{\pi} = \begin{bmatrix} 0.3 \\ 0.3 \\ 0.3 \end{bmatrix},$$

where $\boldsymbol{\sigma}^2 = (\sigma_1^2, \sigma_2^2, \sigma_3^2)^T$ and $\boldsymbol{\pi} = (\pi_1, \pi_2, \pi_3)^T$.

1) E step: Update p_{ik}, $i = 1, 2, ..., 5$ and $k = 1, 2, 3$ using Eq. (7) with $v = 7.3846$ (i.e., the area of the bounding box of the $\boldsymbol{Y}$):

$$\boldsymbol{P} = \begin{bmatrix} 0.3523 & 0.3327 & 0.3148 \\ 0.3524 & 0.3327 & 0.3147 \\ 0.3451 & 0.3333 & 0.3214 \\ 0.3195 & 0.3347 & 0.3454 \\ 0.3001 & 0.3349 & 0.3642 \end{bmatrix},$$

where the ikth entry of $\boldsymbol{P}$ is p_{ik}.

2) M step: Update σ_k^2 and π_k, $k = 1, 2, 3$ using Eq. (10):

$$\boldsymbol{\sigma}^2 = \begin{bmatrix} 0.0010 \\ 0.0011 \\ 0.0011 \end{bmatrix}, \qquad \boldsymbol{\pi} = \begin{bmatrix} 0.3339 \\ 0.3337 \\ 0.3321 \end{bmatrix},$$

where $\boldsymbol{\sigma}^2 = \left(\sigma_1^2, \sigma_2^2, \sigma_3^2\right)^T$ and $\boldsymbol{\pi} = \left(\pi_1, \pi_2, \pi_3\right)^T$.

Alternate between the above 1) and 2), until Q in Eq. (8) converges or the number of iterations reaches the maximum, obtaining:

$$\boldsymbol{P} = \begin{bmatrix} 0.3381 & 0.3336 & 0.3284 \\ 0.3381 & 0.3336 & 0.3283 \\ 0.3364 & 0.3337 & 0.3298 \\ 0.3307 & 0.3342 & 0.3351 \\ 0.3263 & 0.3345 & 0.3392 \end{bmatrix},$$

where the ikth entry of $\boldsymbol{P}$ is p_{ik}, $i = 1, 2, ..., 5$ and $k = 1, 2, 3$.

Compute the weight coefficients c_i, $i = 1, 2, ..., 5$:

$$\boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

where $\boldsymbol{C}$ is a diagonal matrix with the iith entry being the $c_i = \sum_{k=1}^{3} p_{ik}$. The converged probabilistic model concludes that the matching probability of each point pair in $\boldsymbol{\mathcal{M}}$ approximates or equals 1.

9. Iterate between the above step 7 and step 8 until $\bar{\boldsymbol{A}}$ converges or the number of iterations reaches the maximum, obtaining the estimated transformation f under the optimized c_i, $i = 1, 2, ..., 5$:

$$\bar{\boldsymbol{A}} = \begin{bmatrix} 0.1155 & -0.0093 \\ -0.0417 & 0.0992 \\ 0.0800 & 0.1169 \\ -0.1935 & 0.2284 \\ 0.0459 & -0.4771 \end{bmatrix}, \quad f\left(\boldsymbol{X}\right) = \begin{bmatrix} -1.6929 & 1.1120 \\ -0.5427 & 0.8059 \\ -0.0225 & -0.0288 \\ 0.5924 & -0.8733 \\ 1.6509 & -1.0436 \end{bmatrix}.$$

10. Update the source point set $\boldsymbol{X}$ by $f(\boldsymbol{X})$ and inverse normalize $\boldsymbol{X}$:

$$\boldsymbol{X} = \begin{bmatrix} 1.9481 & 5.0047 \\ 4.0216 & 4.4528 \\ 4.9594 & 2.9481 \\ 6.0680 & 1.4257 \\ 7.9763 & 1.1187 \end{bmatrix}.$$

11. Repeat steps 1−10 until $\boldsymbol{X}$ sufficiently approaches $\boldsymbol{Y}$ or the number of repeats reaches the maximum, obtaining:

$$\boldsymbol{X} = \begin{bmatrix} 2.0000 & 5.0000 \\ 4.0000 & 4.5000 \\ 5.0000 & 3.0000 \\ 6.0000 & 1.5000 \\ 8.0000 & 1.0001 \end{bmatrix}.$$

The calculation results of the above steps are output by the program of Ur-SVR with only four decimal places retained. The initial state and the registration result of $\boldsymbol{X}$ and $\boldsymbol{Y}$ are shown in Figure 1.

The calculation process of the Laplacian matrix $\boldsymbol{L}$:

Given a weighted graph $G = (V, E, \omega)$, the vertex set V includes all $\boldsymbol{x}_i$, the edge set E contains edges e_{ij} joining vertices $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, and ω_{ij} represents the weight of edge e_{ij}, with

$$\omega_{ij} = \begin{cases} e^{-\frac{1}{t}\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}, & \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 \leqslant \eta, \\ 0, & \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 > \eta, \end{cases} \qquad (t = 0.2,\ \eta = 0.2)\,.$$

Since

$$\left(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2\right)_{5\times 5} = \begin{bmatrix} 0 & 1.8519 & 4.6296 & 11.2963 & 14.8148 \\ 1.8519 & 0 & 0.9259 & 4.6296 & 6.2963 \\ 4.6296 & 0.9259 & 0 & 1.4815 & 3.1481 \\ 11.2963 & 4.6296 & 1.4815 & 0 & 0.9259 \\ 14.8148 & 6.2963 & 3.1481 & 0.9259 & 0 \end{bmatrix},$$

we have

$$\boldsymbol{W} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

with the ijth entry being ω_{ij}. Hence, the diagonal matrix $\boldsymbol{D}$ can be constructed:

$$\boldsymbol{D} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

in which the iith entry is $d_{ii} = \sum_{j=1}^{5} \omega_{ij}$.

Therefore, we have

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$