Chapter 1

Rank constrained optimization problems in computer vision

1.1 Introduction

The claim that "Behind every data modeling problem there is a (hidden) low rank approximation problem" [13] is demonstrated in this book chapter via four problems in computer vision:

- multidimensional scaling,
- conic section fitting,
- fundamental matrix estimation, and
- least squares contour alignment.

A matrix constructed from exact data is rank deficient. The corresponding data fitting problem in the case of noisy data is a rank constraint optimization problem. In general, rank constraint optimization is a hard nonconvex problem, for which application specific heuristics are proposed. In the chapter, I do not describe solution methods for rank constraint optimization but refer the reader to the literature.
Our main contribution is the analytic solution of contour alignment problem presented in Section 1.5.1. This problem is also nonconvex in the original problem variables, however, a nonlinear change of variables, renders the problem convex in the transformed variables. The link to low-rank approximation (the motto of the chapter) is presented in Section 1.5.4, where the problem is shown to be equivalent to the orthogonal Procrustes problem, which is a constrained low-rank approximation problem [12].

1.2 Multidimensional scaling

Consider \( N \) points \( \{x_1, \ldots, x_N\} \) in an \( n \)-dimensional real space and let \( d_{ij} \) be the squared Euclidean distances between \( x_i \) and \( x_j \). The matrix \( D = [d_{ij}] \in \mathbb{R}^{N \times N} \) of the pair-wise squared distances is symmetric, element-wise nonnegative, and has zero diagonal elements. Moreover, since

\[
 d_{ij} := (x_i - x_j)^\top (x_i - x_j) = x_i^\top x_i - 2x_i^\top x_j + x_j^\top x_j
\]

\( D \) has the following structure

\[
 D = \begin{bmatrix}
 x_1^\top x_1 & \cdots & x_N^\top x_N \\
 \vdots & \ddots & \vdots \\
 x_1^\top x_N & \cdots & x_N^\top x_N
\end{bmatrix} - 2 \begin{bmatrix}
 x_1^\top \\
 \vdots \\
 x_N^\top 
\end{bmatrix} \begin{bmatrix}
 1 & \cdots & 1 \\
 \vdots & \ddots & \vdots \\
 1 & \cdots & 1
\end{bmatrix} + \begin{bmatrix}
 1 \\
 \vdots \\
 1
\end{bmatrix} \begin{bmatrix}
 x_1^\top x_1 & \cdots & x_N^\top x_N \\
 \vdots & \ddots & \vdots \\
 x_1^\top x_N & \cdots & x_N^\top x_N
\end{bmatrix},
\]

or

\[
 D = \text{diag}(X^\top X)1_N^\top - 2X^\top X + 1_N \text{diag}(X^\top X) =: \mathcal{S}(X),
\]

(1.1)

where

\[
 X := \begin{bmatrix}
 x_1 & \cdots & x_N
\end{bmatrix} \quad \text{and} \quad 1_N = \begin{bmatrix}
 1 & \cdots & 1
\end{bmatrix}^\top \in \mathbb{R}_N.
\]

In particular, from (1.1) it can be seen that \( D \) is rank deficient:

\[
 \text{rank}(D) \leq n + 2.
\]

The image of the function \( \mathcal{S} : X \mapsto D \) is referred to as the set of element-wise-squared-distance matrices. The inverse of \( \mathcal{S} \) is a set valued function

\[
 \mathcal{S}^{-1}(D) := \{ X \mid (1.1) \text{ holds} \}.
\]

If \( D \) is a distance matrix of a set of points \( X \), \( \mathcal{S}^{-1}(D) \) consists of all rigid transformations (translation, rotation, and reflection) of \( X \). In other words the nonuniqueness in finding \( X \), given \( D \), is up to a rigid transformation.

**Theorem 1.** Let \( D \) be a distance matrix and let \( \bar{X} \) be a particular solution of the equation (1.1). Then

\[
 \mathcal{S}^{-1}(D) = \{ R\bar{X} + c1_N^\top \mid c \in \mathbb{R}^n \text{ and } R \in \mathbb{R}^{n \times n}, \text{ such that } RR^\top = I \}.
\]
The considered problem is defined informally as follows:

Given noisy and incomplete information about the pair-wise squared-distances \( d_{ij} \) among the points \( \{x_1, \ldots, x_N\} \) and the dimension \( n \) of the ambient space, find estimates of the points \( \{x_1, \ldots, x_N\} \), up to a rigid transformation.

With exact data, the problem can be posed and solved as a rank revealing factorization problem (see the appendix). With noisy measurements, however, the matrix \( D \) is generically full rank. In this case, the relative (up to rigid transformation) point locations can be estimated by approximating \( D \) by a rank-\((n + 2)\) matrix \( \hat{D} \). In order to be a valid distance matrix, however, \( \hat{D} \) must have the structure \( \hat{D} = \mathcal{S} (\hat{X}) \), for some \( \hat{X} = [\hat{x}_1 \ldots \hat{x}_N] \), i.e., the estimation problem is a bilinearly structured low-rank approximation problem:

\[
\text{minimize } \| D - \hat{D} \|_F \quad \text{subject to } \hat{D} = \mathcal{S} (\hat{X}),
\]

where \( \| \cdot \|_F \) is the Frobenius norm. Note that the rank constraint (1.2) is automatically satisfied by the structure constraint (1.1).

For comprehensive treatment of applications and solution methods for multidimensional scaling, the reader is referred to the books [4, 2].

1.3 Conic section fitting

A conic section is a static quadratic model. In this section, I show that the conic section fitting problem can be formulated as a low-rank approximation of an extended data matrix. The mapping from the original data to the extended data is called in the machine learning literature the feature map. In the application at hand, the feature map is naturally defined by the conic model, i.e., it is a quadratic function.

Let

\[
\{d_1, \ldots, d_N\} \subset \mathbb{R}^2, \quad \text{where} \quad d_j = \begin{bmatrix} x_j \\ y_j \end{bmatrix},
\]

be the given data. A conic section is a set defined by a second order equation

\[
\mathcal{B}(A, b, c) := \{d \in \mathbb{R}^2 \mid d^\top Ad + b^\top d + c = 0\}.
\]

(1.3)

Here \( A \) is a \( 2 \times 2 \) symmetric matrix, \( b \) is a \( 2 \times 1 \) vector, and \( c \) is a scalar. \( A, b, \) and \( c \) are the parameters of the conic section. In order to avoid a trivial solution \( \mathcal{B} = \mathbb{R}^2 \), it is assumed that at least one of the parameters \( A, b, \) or \( c \) is nonzero.

The representation (1.3) is an implicit representation of the conic section, because it imposes a relation (implicit function) on the elements \( x \) and \( y \) of \( d \). In special cases, it is possible to use explicit representations defined by a function from \( x \) to \( y \) or from \( y \)
to $x$, however, this approach is restrictive as it does not cover all conic sections (e.g., an ellipse cannot be represented by a map from one variable to the other).

Defining the parameter vector

$$
\theta := \begin{bmatrix} a_{11} & 2a_{12} & b_1 & a_{22} & b_2 & c \end{bmatrix},
$$

and the extended data vector

$$
d_{\text{ext}} := \begin{bmatrix} x^2 & xy & x & y^2 & y & 1 \end{bmatrix}^\top,
$$

we have that

$$
d \in \mathcal{B}(\theta) = \mathcal{B}(A, b, c) \iff \theta d_{\text{ext}} = 0.
$$

(The map $d \mapsto d_{\text{ext}}$, defined by (1.4), is the feature map for the conic section model.) Consequently, all data points $d_1, \ldots, d_N$ are fitted by the model if

$$
\theta \begin{bmatrix} d_{\text{ext},1} & \cdots & d_{\text{ext},N} \end{bmatrix} = 0 \iff \text{rank}(D_{\text{ext}}) \leq 5. \tag{1.5}
$$

Indeed, for $\theta \neq 0$, the left-hand-side of the equivalence states that $D_{\text{ext}}$ has a nontrivial left kernel. Since $D_{\text{ext}}$ has 6 rows (see (1.4)), its rank is at most 5. The mapping $D \mapsto D_{\text{ext}}$ is denoted by $\mathcal{S}$.

In the presence of noise, generically, $\text{rank}(D_{\text{ext}}) > 5$. Then, the aim is to

approximate the data points $d_1, \ldots, d_N$ by nearby points $\hat{d}_1, \ldots, \hat{d}_N$ that lie exactly on a conic section.

Minimizing the sum of squares of the orthogonal distances from the data points to their approximations leads to the structured low-rank approximation problem

$$
\text{minimize} \quad \text{over } \hat{D} \in \mathbb{R}^{2 \times N} \quad \|D - \hat{D}\|_F \quad \text{subject to} \quad \text{rank}(\mathcal{S}(\hat{D})) \leq 5,
$$

where

$$
D := \begin{bmatrix} d_1 & \cdots & d_N \end{bmatrix}, \quad \hat{D} := \begin{bmatrix} \hat{d}_1 & \cdots & \hat{d}_N \end{bmatrix}
$$

are the data matrix and the approximating matrix, respectively.

In the computer vision literature, see, e.g., the tutorial paper [23], conic section fitting by orthogonal projections is called geometric fitting. As shown above, the corresponding computational problem is a quadratically structured low-rank approximation problem. The problem is intuitively appealing, however, it is nonconvex and, moreover, leads to an inconsistent estimator. This has motivated work on easier to compute methods [1, 6, 8, 5, 10, 14, 19] that also reduce or even eliminate the bias.
1.4 Fundamental matrix estimation

In two-dimensional motion analysis [11] a scene is captured by two cameras at fixed locations (stereo vision) and \( N \) matching pairs of points

\[
\{ u_1, \ldots, u_N \} \subset \mathbb{R}^2 \quad \text{and} \quad \{ v_1, \ldots, v_N \} \subset \mathbb{R}^2
\]

are located in the resulting images. The corresponding points \( u \) and \( v \) in the two images satisfy what is called an epipolar constraint

\[
[v^T 1] F \left[ \begin{array}{c} u \\ 1 \end{array} \right] = 0,
\]

for some \( F \in \mathbb{R}^{3 \times 3} \), with rank \((F) = 2\). (1.7)

The \( 3 \times 3 \) matrix \( F \neq 0 \), called the fundamental matrix, characterizes the relative position and orientation of the cameras and does not depend on the selected pairs of points. Estimation of \( F \) from data is a necessary calibration step in many computer vision methods.

The epipolar constraint (1.7) is linear in \( F \). Indeed, defining

\[
d_{\text{ext}} := [u_x v_x \quad u_x v_y \quad u_x \quad u_y v_x \quad u_y \quad v_x \quad v_y \quad 1]^T \in \mathbb{R}^9,
\]

where \( u = [u_x u_y] \) and \( v = [v_x v_y] \), (1.7) can be written as

\[
\text{vec}^T(F) d_{\text{ext}} = 0.
\]

Note that, as in the application for conic section fitting, the original data \((u,v)\) is mapped to an extended data vector \( d_{\text{ext}} \) via a nonlinear function (a feature map). In this case, however, the function is bilinear.

Taking into account the epipolar constraints for all data points, we obtain the matrix equation

\[
\text{vec}^T(F) \underbrace{[d_{\text{ext},1} \ldots d_{\text{ext},N}]}_{D_{\text{ext}}} = 0.
\]

The rank constraint imposed on \( F \) implies that \( F \) is a nonzero matrix. Therefore, by (1.9) \( D_{\text{ext}} \) has a nontrivial left kernel and since \( D_{\text{ext}} \) is \( 9 \times N \)

\[
\text{rank}(D_{\text{ext}}) \leq 8
\]

It can be concluded that for \( N \geq 8 \) data points, \( D_{\text{ext}} \) is not full row rank. Moreover, if the left kernel is one dimensional, the fundamental matrix \( F \) can be reconstructed up to a scaling factor from the data.

In the case of noisy data,

the aim is to perturb as little as possible the data (1.6), so that the perturbed data satisfies exactly the epipolar constraints for some \( \hat{F} \) with rank(\( \hat{F} \)) = 2.
The resulting estimation problem is a bilinearly structured low-rank approximation with an additional rank constraint. This problem defines a maximum-likelihood estimator for the true parameter value. As in the conic section fitting problem, the maximum-likelihood estimator is a nonconvex optimization problem and is inconsistent in the measurement errors or errors-in-variables setup. These facts motivated the development of methods that are convex and unbiased, see [21, 3, 9, 14] and the references there in. Closely related to the estimation of the fundamental matrix problem in two-view computer vision is the shape from motion problem [20].

1.5 Least squares contour alignment

Let \( R_{\theta} \) be the operator in \( \mathbb{R}^2 \) that rotates its argument by \( \theta \) rad (positive angle corresponding to anticlockwise rotation) and let \( R_{\theta}' \) be the operator that reflects its argument about a line, passing through the origin, at \( \theta / 2 \) rad with respect to the first basis vector (see Figure 1.1).

It can be shown that \( R_{\theta} \) and \( R_{\theta}' \) have matrix representations

\[
 R_{\theta}(p) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} p = R_{\theta}p
\]

and

\[
 R_{\theta}'(p) = \begin{bmatrix} \cos \theta & \sin \theta \\ \sin \theta & -\cos \theta \end{bmatrix} p = R_{\theta}'p.
\]

In [15], the authors considered transformation by rotation, scaling, and translation, i.e.,

\[
 A_{a, \theta, s}(p) = sR_{\theta}(p) + a,
\]
where $s > 0$ is the scaling factor and $a \in \mathbb{R}^2$ is the translation parameter. The problem of determining the parameters $\theta, s,$ and $a$ of a transformation $\mathcal{A}_{\theta, s}(p)$ that best, in a least squares sense, matches one set of points $p^{(1)}, \ldots, p^{(N)}$ to another set of points $q^{(1)}, \ldots, q^{(N)}$ can be used to align two explicitly represented contours, specified by corresponding points. Although this alignment problem is a nonlinear least squares problem in the parameters $\theta, s,$ and $a$, it is shown in [15] that the change of variables

$$b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = s \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix}, \quad \left[ \begin{array}{c} \theta \\ s \end{array} \right] = \left[ \begin{array}{c} \sin^{-1} \left( \frac{b_2}{\|b\|} \right) \\ \|b\| \end{array} \right]$$

(1.11)

results in an equivalent linear least squares problem in the parameters $a$ and $b$. This fact allowed efficient solution of image registration problems (see, e.g., [17, 16]) with a large number of corresponding points. The invariance to rigid transformation appears also in learning with linear functionals on reproducing kernel Hilbert space, see [22].

It is well known, however, that dilation and rigid transformation involves reflection, in addition to rotation, scaling, and translation. Therefore, the problem occurs of how to align optimally in a least squares sense the set of points

$$\{ p^{(1)}, \ldots, p^{(N)} \} \quad \text{and} \quad \{ q^{(1)}, \ldots, q^{(N)} \}$$

under reflection, rotation, scaling, and translation, i.e., transformation of the type

$$\mathcal{A}_{a, \theta_1, \theta_2, s}(p) = s \mathcal{R}_{\theta_1} \left( \mathcal{R}_{\theta_2}^t(p) \right) + a.$$  

(1.12)

In order to solve the problem of alignment by dilation and rigid transformation, first consider alignment by reflection, scaling, and translation, i.e., transformation of the type

$$\mathcal{A}_{a, \theta, s}(p) = s \mathcal{R}_{\theta}^t(p) + a.$$  

(1.13)

The solution of this latter problem, given in Section 1.5.1, also uses the change of variables (1.11) to convert the original nonlinear least squares problem to a linear one. The derivation given in Section 1.5.1, however, is different from the derivation in [15] and reveals a link between the alignment problems by rotation and reflection.

The solution to the general least squares alignment problem by rigid transformation is given in Section 1.5.2. Since a transformation (1.12) is either rotation, scaling, and translation, or reflection, scaling, and translation, the alignment problem (1.12) reduces to solving problems (1.10) and (1.13) separately, and choosing the solution that corresponds to the better fit.

In Section 1.5.4, I show that least squares alignment by rotation and reflection is equivalent to the orthogonal Procrustes problem [7, Page 601]. An extension of the orthogonal Procrustes problem to alignment by (1.12), presented in [18], gives an alternative solution method for contour alignment by dilation and rigid transformation. An advantage of the approach based on the orthogonal Procrustes problem is that the solution is applicable to data in higher dimensional space, however, the method requires singular value decomposition of a matrix computed from the data, which may be computationally more expensive than solving an ordinary linear least squares problem.
1.5.1 Alignment by reflection, scaling, and translation

Let $C_1$ and $C_2$ be the matrices of the stacked next to each other points $p(1), \ldots, p(N)$ and $q(1), \ldots, q(N)$, respectively, i.e.,

$$C_1 := \begin{bmatrix} p(1) & \cdots & p(N) \end{bmatrix} \quad \text{and} \quad C_2 := \begin{bmatrix} q(1) & \cdots & q(N) \end{bmatrix},$$

and let $\| \cdot \|_F$ be the Frobenius norm, defined as

$$\|C_1\|_F := \sqrt{\sum_{i=1}^{N} \|p(i)\|^2}.$$

The problem considered in this section is least squares alignment by reflection:

$$\begin{align*}
\text{minimize} & \quad \|C_1 - A' a, \theta, s(C_2)\|_F \\
\text{over} & \quad a \in \mathbb{R}^2, \ s > 0, \ \theta \in [-\pi, \pi).
\end{align*} \tag{1.14}$$

Similarly to the alignment by rotation problem

$$\begin{align*}
\text{minimize} & \quad \|C_1 - A' a, \theta, s(C_2)\|_F \\
\text{over} & \quad a \in \mathbb{R}^2, \ s > 0, \ \theta \in [-\pi, \pi),
\end{align*} \tag{1.15}$$

(1.14) is a nonlinear least squares problem in the parameters $\theta$, $s$, and $a$. The change of variables (1.11), however, also transforms problem (1.14) into a linear least squares problem.

**Theorem 2** (Alignment by reflection, scaling, and translation). Problem (1.14) is equivalent to the linear least squares problem

$$\begin{align*}
\text{minimize} & \quad \|\text{vec}(C_1) - [(C_2^\top \otimes I_2)E \ 1_N \otimes I_2] \begin{bmatrix} b_1 \\ a \end{bmatrix}\|_2 \\
\text{over} & \quad a, b \in \mathbb{R}^2, \ b_1, b_2 \text{ is given by (1.11)}.
\end{align*} \tag{1.16}$$

where $\text{vec}(\cdot)$ is the column-wise matrix vectorization operator, $\otimes$ is the Kronecker product,

$$1_N := \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \in \mathbb{R}^N, \ E := \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & -1 \\ 1 & 0 \end{bmatrix}, \text{ and } I_2 := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{1.17}$$

The one-to-one relation between the parameters $\theta$, $s$ and $b_1$, $b_2$ is given by (1.11).

**Proof.** Note that

$$\mathcal{A}_{a, \theta, s}'(C_2) = sR_{\theta}'C_2 - a1_N^\top.$$

Using the identity,

$$\text{vec}(AXB) = (B^\top \otimes A) \text{vec}(X),$$
we rewrite the cost function of (1.14) as

\[
\left\| C_1 I_2 \begin{bmatrix} s \cos \theta & s \sin \theta \\ s \sin \theta & -s \cos \theta \end{bmatrix} C_2 - a \right\|_F = \left\| \text{vec}(C_1) - (C_2^T \otimes I_2) \begin{bmatrix} s \cos \theta \\ s \sin \theta \\ s \sin \theta \\ -s \cos \theta \end{bmatrix} - a \right\|_2
\]

Problem (1.14) and the relation (1.11) follows by setting

\[ b_1 := s \cos \theta \quad \text{and} \quad b_2 := s \sin \theta. \]

\[ \Box \]

**Note 1 (Alignment by rotation, scaling, and translation).** The above solution of problem (1.14) can be modified easily for the corresponding alignment problem with rotation (1.15), giving an alternative shorter proof to Theorem 1 in [15]. Indeed, the only necessary modification is to replace the matrix \( E \) in (1.17) by

\[
E = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}
\]

**Example 3.** As an illustration of the presented alignment procedure consider the contours shown in Figure 1.2. The optimal alignment by rotation, scaling, and translation is shown in Figure 1.3, right, and the optimal alignment by reflection, scaling, and translation is shown in Figure 1.3, left.

1.5.2 **Alignment by rigid transformation**

The problem considered in this section is:

\[
\begin{align*}
\text{minimize} & \quad \| C_1 - \omega_{a, \theta_1, \theta_2, s}(C_2) \|_F \\
\text{over} & \quad a \in \mathbb{R}^2, \ s > 0, \ \theta_1, \theta_2 \in [-\pi, \pi]
\end{align*}
\]

(1.18)

The following fact allows us to reduce problem (1.18) to the already studied problems (1.14) and (1.15).

**Proposition 1.** A transformation by rotation and reflection, \( R_{\theta_1}(R_{\theta_2}(p)) \), is equivalent to a transformation by an orthogonal matrix \( Qp \). Moreover,

\[ Qp = R_{\theta}(p), \quad \text{if} \ det(Q) = 1, \]
FIGURE 1.2: Example of contour alignment problem (1.14): given contours $\mathcal{C}_1$ and $\mathcal{C}_2$ with corresponding points $p^{(i)} \leftrightarrow q^{(i)}$, find a transformation $A_{a,\theta,s}$ that minimizes the distance between $C_1$ and the transformed contour $A_{a,\theta,s}(C_2)$.

FIGURE 1.3: Left: optimal alignment of $\mathcal{C}_2$ to $\mathcal{C}_1$ and $\mathcal{C}_1$ to $\mathcal{C}_2$ by $A_{a,\theta,s}$ (reflection), Right: optimal alignment of $\mathcal{C}_2$ to $\mathcal{C}_1$ and $\mathcal{C}_1$ to $\mathcal{C}_2$ by $A_{a,\theta,s}$ (rotation).
and
\[ Qp = R'_\theta(p), \quad \text{if } \det(Q) = -1, \]
where, in either case,
\[ \theta = \cos^{-1}(q_{11}). \quad (1.19) \]

Proof. The matrix \( R_\theta \) and \( R'_\theta \) are orthogonal matrices and the product of orthogonal matrices is an orthogonal matrix. Next, I show that an orthogonal matrix \( Q \) is either a rotation matrix \( R_\theta \), for some \( \theta \in [-\pi, \pi) \), or a reflection matrix \( R'_\theta \), for some \( \theta \in [-\pi, \pi) \).

Since \( Q \) is orthogonal
\[
\begin{bmatrix}
q_{11} & q_{12} \\
q_{21} & q_{22}
\end{bmatrix}
\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}.
\]
Without loss of generality we can choose
\[ q_{11} = \cos \theta \quad \text{and} \quad q_{12} = \sin \theta. \]
Then, there are two possibilities for \( q_{21} \) and \( q_{22} \)
\[ q_{21} = \cos(\theta + \pi/2) \quad \text{and} \quad q_{22} = \sin(\theta + \pi/2) \]
or
\[ q_{21} = \cos(\theta - \pi/2) \quad \text{and} \quad q_{22} = \sin(\theta - \pi/2). \]
In the first case, \( Q \) is a rotation matrix and in the second case \( Q \) is a reflection matrix. Therefore,
\[ Q = R_\theta \quad \text{or} \quad Q = R'_\theta, \]
where
\[ \theta = \cos^{-1}(q_{11}). \]
It is easy to check that
\[ \det(R_\theta) = 1 \quad \text{and} \quad \det(R'_\theta) = -1, \quad \text{for any } \theta. \]

The result of Proposition 1 shows that problem (1.18), can be solved by the following procedure.

1. Solve the alignment problem by reflection (1.14).
2. Solve the alignment problem by rotation (1.15).
3. Select the solution of the problem that gives smaller cost function value.

Since, problems (1.14) and (1.15), are already solved in Section 1.5.1, we have a complete solution to (1.18).
1.5.3 Invariance properties and a distance measure

It turns out that the minimum value of (1.18)
\[ \text{dist}'(C_1, C_2) := \min_{a \in \mathbb{R}^2, s > 0, \theta_1, \theta_2 \in [-\pi, \pi]} \| C_1 - S(a, \theta_1, \theta_2, s)(C_2) \|_F \]
is not a proper distance measure. (A counter example is given in Example 6.) Proposition 4 (invariance) and Theorem 7 (distance measure) stated and proved in [15] for alignment by (1.10), however, hold for the more general problem of alignment by dilation and rigid transformation.

**Proposition 2.** If the contours \( C_1 \) and \( C_2 \), defined by the sets of corresponding points \( \{ p_i^{(i)} \} \) and \( \{ q_i^{(i)} \} \), are centered (i.e., \( C_1 \mathbf{1}_N = C_2 \mathbf{1}_N = 0 \)), \( \text{dist}'(C_1, C_2) \) is invariant to a rigid transformation, i.e.,
\[
\text{dist}'(C_1, C_2) = \text{dist}'(R_\theta(C_1), R_\theta(C_2)) = \text{dist}'(R_\theta'(C_1), R_\theta'(C_2)), \quad \text{for any } \theta \in [-\pi, \pi].
\]

If, in addition, \( C_1 \) and \( C_2 \) are normalized by \( \| C_1 \|_F = \| C_2 \|_F = 1 \),
\[
\text{dist}'(C_1, C_2) = \text{dist}'(C_2, C_1).
\]

**Example 4.** Consider again the contours from Example 3. The points \( p_i^{(i)} \) and \( q_i^{(i)} \) are preprocessed, so that the resulting contours, say \( C_{1,c} \) and \( C_{2,c} \), are centered. As a numerical verification of (1.20), we have
\[
\text{dist}'(C_{1,c}, C_{2,c}) = \text{dist}'(R_{0.3}(C_{1,c}), R_{0.3}(C_{2,c})) = \text{dist}'(R_{0.3}'(C_{1,c}), R_{0.3}'(C_{2,c})) = 0.40640.
\]

Let, in addition, the points \( p_i^{(i)} \) and \( q_i^{(i)} \) be preprocessed, so that the resulting contours, say \( C_{1,c,n} \) and \( C_{2,c,n} \), are centered and normalized. As a numerical verification of (1.21), we have
\[
\text{dist}'(C_{1,c,n}, C_{2,c,n}) = \text{dist}'(C_{2,c,n}, C_{1,c,n}) = 0.11271.
\]

As in the case of the transformation (1.10), treated in [15, Section III], the following definition gives a distance measure.

**Definition 1** (2-norm distance between contours modulo rigid transformation).
\[
\text{dist}(C_1, C_2) := \frac{1}{\| C_1 \mathbf{1}_N \mathbf{1}_N^\top \|_F} \times \min_{a \in \mathbb{R}^2, s > 0, \theta_1, \theta_2 \in [-\pi, \pi]} \| C_1 - S(a, \theta_1, \theta_2, s)(C_2) \|_F. \quad (1.22)
\]

**Theorem 5.** The distance measure \( \text{dist}(C_1, C_2) \) is symmetric and invariant to dilation and a rigid transformation, i.e.,
\[
\text{dist}(C_1, C_2) = \text{dist}(C_2, C_1) = \text{dist}(S(a, \theta_1, \theta_2, s)(C_1), S(a, \theta_1, \theta_2, s)(C_2)),
\]
for all \( a \in \mathbb{R}^2, \theta_1, \theta_2 \in [-\pi, \pi], \) and \( s > 0 \). (1.23)
Example 6. For the contours in Example 3, we have
\[
\text{dist}'(C_1, C_2) = 0.40640 \quad \text{and} \quad \text{dist}'(C_2, C_1) = 0.20748,
\]
while
\[
\text{dist}(C_1, C_2) = \text{dist}(C_2, C_1) = 0.11271.
\]

1.5.4 Contour alignment as an orthogonal Procrustes problem

As a consequence of Proposition 1, we have that problem (1.18) is equivalent to
\[
\begin{align*}
\text{minimize} & \quad \|C_1 - sQC_2 - a\|_F \\
\text{subject to} & \quad Q^T Q = I_2 \\
& \quad \text{over } a \in \mathbb{R}^2, s > 0, Q \in \mathbb{R}^{2 \times 2}.
\end{align*}
\]

(1.24)

In turn, problem (1.24) is related to the orthogonal Procrustes problem in numerical linear algebra.

**Problem 1** (Orthogonal Procrustes problem). Given \(q \times N\) real matrices \(C_1\) and \(C_2\),
\[
\text{minimize over } Q \quad \|C_1 - QC_2\|_F \quad \text{subject to } \quad Q^T Q = I_q.
\]

The classical solution of the orthogonal Procrustes problem is given by
\[
Q = UV^T,
\]
where \(U\Sigma V^T\) is the singular value decomposition (SVD) of \(C_1^T C_2\), see [7, Page 601].

The orthogonal Procrustes problem does not involve scaling and translation. The extension of the problem to alignment by dilation and rigid transformation is done in [18]. The resulting procedure is summarized in Algorithm 1. It presents an alternative solution approach for solving problem (1.18). Compared to the solution proposed in Section 1.5.4, Algorithm 1 has the advantage of being applicable to data of any dimension \((C_1, C_2 \in \mathbb{R}^{q \times N}\), for any natural number \(q\)), i.e., the solution based on the orthogonal Procrustes problem is applicable to contours in spaces of dimension higher than 2.

The solution based on the orthogonal Procrustes problem, however, uses the singular value decomposition, while the solution proposed in Section 1.5.4 involves two ordinary least squares problems. Therefore, an advantage of the proposed solution is its conceptual simplicity. In particular, exploiting the Kronecker structure of the coefficients matrix in (1.16) one can derive an efficient algorithm for alignment of contours specified by a large number of corresponding points. Furthermore, in the case of sequential but not necessarily corresponding points (see, [15, Section IV]), \(N\) alignment problems are solved, which makes the computational efficiency an important factor.
Algorithm 1 Algorithm for least-squares contour alignment, based on the orthogonal Procrustes problem.

Require: Contours with corresponding points, specified by matrices $C_1$ and $C_2$.

1: Centering of the contours:

$$C_{i,c} := C_i - a^{(i)} 1_N^T,$$

where $a^{(i)} := \frac{1}{N} C_i 1_N$.

2: Alignment of the centered data by orthogonal transformation:

$$Q := UV^T,$$

where $U \Sigma V^T$ is the SVD of $C_2^T C_1^T$.

3: Computation of the scaling parameter:

$$s := \frac{\text{trace}(QC_2 C_1^T)}{\|C_2, c\|_F^2},$$

4: Rigid transformation of $C_2$ to fit $C_1$:

$$\hat{C}_1 := s Q (C_2 - a^{(2)} 1_N^T) + a^{(1)} 1_N^T.$$

Ensure: Rigid transformation parameters:

- $a^{(1)} - s Q a^{(2)}$ — translation,
- $Q$ — orthogonal transformation, and
- $s$ — scaling.
1.6 Conclusions

This chapter illustrated the claim that every data modeling problem is related to a (structured) low-rank approximation problem for a matrix obtained from the data via a nonlinear transformation (feature map) by four specific examples in computer vision: multidimensional scaling, conic section fitting, fundamental matrix estimation, and contour alignment. In multidimensional scaling, the data is the squared distances between a set of points and the structure of the low-rank approximation problem is given by (1.1). This structure automatically makes the constructed matrix rank deficient, so that the low-rank approximation problem has no additional rank constraint. In the conic section fitting problem, the feature map is a quadratic function and, in the fundamental matrix estimation problem, the feature map is a bilinear function. Finally the contour alignment problem was reduced to the orthogonal Procrustes problem, which is a low-rank approximation problem with an additional orthogonality constraint. A summary of the application is given in Table 1.1.

<table>
<thead>
<tr>
<th>Application</th>
<th>Data</th>
<th>Data Matrix</th>
<th>Structure</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>multidim. scaling</td>
<td>distances $d_{ij}$</td>
<td>$[d_{ij}]$</td>
<td>(1.1)</td>
<td>$\dim(x) + 2$</td>
</tr>
<tr>
<td>conic section fitting</td>
<td>points $d_i$</td>
<td>(1.4), (1.5)</td>
<td>quadratic</td>
<td>5</td>
</tr>
<tr>
<td>fundamental matrix estimation</td>
<td>corresponding $u_j$, $v_j$</td>
<td>(1.8), (1.9)</td>
<td>bilinear</td>
<td>8</td>
</tr>
<tr>
<td>contour alignment</td>
<td>corresponding $C_1$, $C_2$</td>
<td>$[C_1] C_2$</td>
<td>unstructured</td>
<td>2</td>
</tr>
</tbody>
</table>

Acknowledgements

Funding from the European Research Council under the European Union’s Seventh Framework Programme (FP7/2007-2013) / ERC Grant agreement number 258581 “Structured low-rank approximation: Theory, algorithms, and applications” is gratefully acknowledged.
Appendix: Position estimation from exact and complete distances

Consider the change of variables
\[ S := X^\top X. \] (A.1)

The inverse transformation \( S \mapsto X \) is a set valued function with nonuniqueness described by the orthogonal transformation \( X \mapsto RX \) \( \text{i.e.,} \) rotation or reflection of the set of points \( X \). A particular solution of the equation (A.1), for given symmetric matrix \( X \) of rank at most \( n \), can be computed by the eigenvalue decomposition of \( X \). Let
\[ S = V \Lambda V^\top = [V_1 \quad V_2] \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} [V_1 \quad V_2]^\top, \]
where the diagonal elements of \( \Lambda_1 \) are all positive, be the eigenvalue decomposition of \( X \). Then
\[ \sqrt{\Lambda_1} V_1^\top = RX, \]
for some orthogonal matrix \( R \).

Equation (1.1) is linear in \( S \). We have,
\[ \text{vec}(D) = (1_N \otimes E + E \otimes 1_N - 2I) \text{vec}(S) =: L \text{vec}(S). \] (A.2)

Furthermore, taking into account the symmetry of \( D \) and \( S \), (A.2) becomes
\[ \text{vec}_s(D) = L_s \text{vec}_s(S). \] (A.3)

The matrix \( L_s \) is of size \( N_s \times N_s \), where \( N_s := N(N + 1)/2 \), and is a submatrix of \( L \in \mathbb{R}^{N \times N} \).

The system of linear equations (A.3) has \( N_s \) equations and \( N_s \) unknowns. The matrix \( L_s \), however, is rank deficient
\[ \text{rank}(L_s) = N_s - N, \]
so that a solution is nonunique. (Assuming that \( D \) is a distance matrix, an exact solution of (A.3) exists.) We are aiming at a solution \( S \) of (A.3) of rank at most \( n \), finding such a solution in the affine set of solutions is a hard problem.

A simple transformation avoids the nonuniqueness issue. The translated set of points
\[ \bar{X} := X - x_1 1_N^\top = \begin{bmatrix} 0 & \bar{x}_2 & \cdots & \bar{x}_N \end{bmatrix} \]
has the same distance matrix as \( X \), \( \text{i.e.,} \) \( \mathcal{D}(\bar{X}) = D \). The change of variables (A.1) then results in a matrix
\[ \bar{S} := \bar{X}^\top \bar{X} = \begin{bmatrix} 0_{1 \times 1} & 0_{N-1 \times 1} \\ 0_{1 \times N-1} & * \end{bmatrix}, \]
so that
\[ \text{vec}_s(\bar{S}) = \begin{bmatrix} 0_{N \times 1} \\ * \end{bmatrix}. \]

From (A.3), we have
\[
\text{vec}_s(D) = L_s \begin{bmatrix} 0_{N \times 1} \\ \bar{s} \end{bmatrix} = : L_s(:,N+1:): \bar{s}.
\] (A.4)

The submatrix \( L_s(:,N+1:) \) of \( L_s \) is full column rank, which implies that \( \bar{s} \) is the unique solution of (A.4)
Bibliography


