Model Fitting and Robust Regression Methods

CMPE 264: Image Analysis and Computer Vision
Winter’03, Hai Tao
1/21/03, 1/23/03
Fitting lines and ellipses to image data

- Fit the best line or ellipse model to a given group image data points
- Ellipse fitting problem

Let \( p_1, \ldots, p_N \) be a set of \( N \) image points, \( p_i = [x_i, y_i] \). Assume the implicit equation of the generic ellipse, defined by its parameter vector \( \mathbf{a} = [a, b, c, d, e, f] \) is

\[
f(\mathbf{p}, \mathbf{a}) = \mathbf{x}^T \mathbf{a} = ax^2 + bxy + cy^2 + dx + ey + f = 0
\]

where \( \mathbf{x} = [x^2, xy, y^2, x, y, 1] \)

Find the optimal parameter \( \mathbf{a}^* \) which satisfies \( \| \mathbf{a}^* \| = 1 \) and

\[
\mathbf{a}^* = \min_{\mathbf{a}} \arg \sum_{i=1}^{N} [D(p_i, \mathbf{a})]^2
\]

where \( D(p_i, \mathbf{a}) \) is a suitable distance (residual)
Algebraic distance fit - ellipse

- The algebraic distance of point $p_i$ from a curve $f(p,a) = 0$ is $|f(p_i,a)|$.
- The ellipse fitting problem becomes

$$ a^* = \min \arg \sum_{i=1}^{N} |x_i^T a|^2 $$

subject to

$$ a^T \begin{bmatrix} 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} a = a^T Ca = b^2 - 4ac = -1 $$

Remarks:

The constraint makes sure the resultant curve is an ellipse.
C is called the constraint matrix.
This problem is often called least squares fitting.
Algebraic distance fit - ellipse

Write the distance in matrix form

\[ \sum_{i=1}^{N} |x_i^T a|^2 = \|Xa\|^2 = a^T X^T Xa = a^T Sa \]  \hspace{1cm} (1)

where

\[ X = \begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ x_2^2 & x_2 y_2 & y_2^2 & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^2 & x_N y_N & y_N^2 & x_N & y_N & 1 \end{bmatrix} \]

is the design matrix

\[ S = X^T X \]

is the scatter matrix

The ellipse fitting problem is rewritten as

\[ a^* = \min_{a} \arg a^T Sa \quad \text{subject to} \quad a^T Ca + 1 = 0 \]
Algebraic distance fit - ellipse

- Using Lagrange multipliers

\[
\min_{a, \lambda} L(a, \lambda) = a^T S a - \lambda (a^T C a + 1)
\]

- To obtain solution, we need to solve

\[
\frac{\partial L}{\partial a} = 2 S a - 2 \lambda C a = 0
\]

\[
\frac{\partial L}{\partial \lambda} = a^T C a + 1 = 0
\]

- From the first equation (generalized eigenvalue problem), \(a\) should satisfy \(S a = \lambda C a\)

- Substitute in (1) \(a^* = \min \arg a^T S a = a^T \lambda C a = -\lambda\)

- It can be proved that the solution is the eigenvector \(a^*\), which corresponds to the only negative negative eigenvalue
Algorithm ALG_ELLIPSE_FIT

- Build the design matrix $X$ from the data points
- Build the scatter matrix $S$ as $S = X^T X$
- Build the constraint matrix $C$
- Solve the generalized eigenvalue problem $S\mathbf{a} = \lambda C\mathbf{a}$

The output is the eigenvector corresponding to the only negative value

Example

Figure 5.3 Example of best-fit ellipses found by ALG_ELLIPSE_FIT for the same arc of ellipse, corrupted by increasingly strong Gaussian noise. From left to right, the noise varies from 3% to 20% of the data spread (figure courtesy of Maurizio Pilu, University of Edinburgh).
Algebraic distance fit - line

The generic line model is

\[ f(p, a) = x^T a = ax + by + c = 0 \]

Find the solution

\[ a^* = \min \arg \sum_{i=1}^{N} |x_i^T a|^2 = a^T X^T X a \]

subject to \( a^T a - 1 = 0 \)

where \( X = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & 1 \end{bmatrix} \)

The solution is the eigenvector corresponding to the the least eigenvalue of \( X^T X \)  (Homework: Prove this !)
Algebraic distance fit with Euclidian fit

- In general, the algebraic distance $|f(p_i, a)|$ is different from the Euclidian distance. In addition, two points with the same algebraic distance to the curve may have different Euclidian distances to the curve.

- In most computer vision algorithms, Euclidian distance is more desirable, but using algebraic distance often makes algorithms simpler.
Algebraic distance fit with Euclidian fit

Since weights around the flat parts of the ellipse are larger, the Euclidian errors at flat area are smaller than the errors in the pointed part. As a consequence, the fit tends to be fatter.

Figure 5.4 Illustration of the low-eccentricity bias introduced by the algebraic distance. ALG_ELLIPSE_FIT was run on 20 samples covering half an ellipse, spaced uniformly along \( x \), and corrupted by different realizations of rather strong, Gaussian noise with constant standard deviation \( (\sigma = 0.08, \text{about} \ 10\% \text{ of the smaller semiaxis}) \). The best-fit ellipse (solid) is systematically biased to be “fatter” than the true one (dashed).
Euclidian fit - ellipse

The Euclidian distance fitting problem becomes

$$a^* = \min_{a} \arg \sum_{i=1}^{N} \| p_i - \hat{p}_i \|^2$$

subject to \( f(\hat{p}_i, a) = 0 \)

Solution

- Using Lagrange multipliers

$$\min_{a, \lambda} L(a, \lambda) = \sum_{i=1}^{N} (\hat{p}_i - p_i)^T (\hat{p}_i - p_i) - 2\lambda_i f(\hat{p}_i, a)$$

- The solution should satisfy

$$\frac{\partial L}{\partial \hat{p}_i} = 2(\hat{p}_i - p_i) - 2\lambda_i \nabla f(\hat{p}_i, a) = 0$$

or

$$(\hat{p}_i - p_i) = \lambda_i \nabla f(\hat{p}_i, a)$$
Euclidian fit - ellipse

- If we approximate $\nabla f(\mathbf{p}_i, \mathbf{a})$ by $\nabla f(\mathbf{p}_i, \mathbf{a})$, the previous equation can be rewritten as
  \[ (\mathbf{p}_i - \mathbf{p}_i) = \lambda_i \nabla f(\mathbf{p}_i, \mathbf{a}) \] (2)

- If we approximate the curve as
  \[ 0 = f(\mathbf{p}_i, \mathbf{a}) \approx f(\mathbf{p}_i, \mathbf{a}) + (\mathbf{p}_i - \mathbf{p}_i)^T \nabla f(\mathbf{p}_i, \mathbf{a}) \]
we obtain
  \[ 0 = f(\mathbf{p}_i, \mathbf{a}) + \lambda_i \nabla^T f(\mathbf{p}_i, \mathbf{a}) \nabla f(\mathbf{p}_i, \mathbf{a}) \]
therefore
  \[ \lambda_i = \frac{-f(\mathbf{p}_i, \mathbf{a})}{\|
abla f(\mathbf{p}_i, \mathbf{a})\|^2} \]
Substituting in (2)
  \[ (\mathbf{p}_i - \mathbf{p}_i) = \frac{-f(\mathbf{p}_i, \mathbf{a})\nabla f(\mathbf{p}_i, \mathbf{a})}{\|
abla f(\mathbf{p}_i, \mathbf{a})\|^2} \]
or
  \[ \|
abla f(\mathbf{p}_i, \mathbf{a})\|^2 = \frac{f(\mathbf{p}_i, \mathbf{a})^2 \nabla^T f(\mathbf{p}_i, \mathbf{a}) \nabla f(\mathbf{p}_i, \mathbf{a})}{\|
abla f(\mathbf{p}_i, \mathbf{a})\|^4} = \frac{f(\mathbf{p}_i, \mathbf{a})^2}{\|
abla f(\mathbf{p}_i, \mathbf{a})\|^2} \]
Euclidian fit - ellipse

The original objective function becomes

$$a^* = \min \arg \sum_{i=1}^{N} \frac{f(p_i, a)^2}{\| \nabla f(p_i, a) \|^2}$$

Now the objective function only depends on $a$. This can be solved using gradient descent methods.

Remarks:
- The Euclidian distance is approximated as the algebraic distance divided by the magnitude of the gradient.
- To make this algorithm work, a good initial $a$ is needed. This can be obtained using a fitting algorithm that uses the algebraic distance.
- Both algorithms are least square methods, even a small number of outliers can degrade the result badly.
Robust algorithms

Figure 5.6  Comparison of ALG_ELLIPSE_FIT and ROB_ELLIPSE_FIT when fitting to data severely corrupted by outliers. The circles show the data points, the asterisks suggest the robust fit, the solid line show the algebraic fit, and the dots the true (uncorrupted) ellipse.
Robust algorithms

Three criteria are often used for evaluating robust algorithms (Meer, Mintz, Rosenfeld and Kim, 1991)

- Relative efficiency – the ratio between the lowest achievable variance for the estimated parameters and the actual variance provided by the method
- Breakdown point – the smallest amount of outlier contamination that may force the value of the estimate outside an arbitrary range. Example: the breakdown point of the mean is 0 since a single large outlier can corrupt the result.
- Time complexity – computational complexity. Feasible algorithms should be at most $O(N^2)$
Robust algorithms

- Solution 1 – change the distance measure, so that it is less sensitive to large errors: using absolute values instead of squares

- Algorithm ROB_ELLIPSE_FIT
  - Run ALG_ELLIPSE_FIT to obtain the initial solution $a_0$
  - Using $a_0$ as the initial solution, using gradient descent method to find the optimal parameters using the following objective function

\[
a^* = \min \arg \sum_{i=1}^{N} |x_i^T a|
\]
Robust algorithms - results

Figure 5.6  Comparison of ALG_ELLIPSE_FIT and ROB_ELLIPSE_FIT when fitting to data severely corrupted by outliers. The circles show the data points, the asterisks suggest the robust fit, the solid line show the algebraic fit, and the dots the true (uncorrupted) ellipse.
Robust algorithms – LMedS method

- The least-median-of-squares method (Rousseeuw 1984)

\[ \mathbf{a}^* = \min_{\mathbf{a}} \arg\med \left[ D(p_i, \mathbf{a}) \right]^2 \]

- Breakdown point is 50%

- Problem statement- we want to estimate coefficient \( \beta_j \)
for

\[ z_i = \sum_{j=0}^{p-1} \beta_j x_j(i) \]

- Solution
  - Solve \( \beta_j, j = 1, ..., p - 1 \)
  - Find the mode of \( z_i - \sum_{j=1}^{p-1} \beta_j x_j(i) \) as \( \beta_0 \)
Robust algorithms – RANSAC

- RANdom SAmple Consensus (Fischler and Bolles, 1981)

- Algorithm
  - Compute a model by solving a system of equations defined for a randomly chosen subset of data points
  - All the data is then classified relative to this model. The points within some error tolerance are called the consensus set of the model
  - If cardinality of the consensus set exceeds a threshold, the model is accepted and its parameters recomputed based on the whole consensus set
  - If the model is not accepted, a new set of points is chosen and resulting model is tested for validity
  - The error tolerance and the consensus set acceptance threshold must be set a priori
  - Of the largest consensus set within a fixed number of trials are used
  - Breakdown point is 50%
The generic line model is

\[ f(p,a) = x^T a = ax + by + c = 0 \]

Prove that the solution

\[ a^* = \min \arg \sum_{i=1}^{N} |x_i^T a|^2 = a^T X^T X a \]

subject to \( a^T a - 1 = 0 \)

where \( X = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & 1 \end{bmatrix} \)

is the eigenvector corresponding to the smallest eigenvalue of \( X^T X \)