Exercise 3D Machine Vision

Sample solution

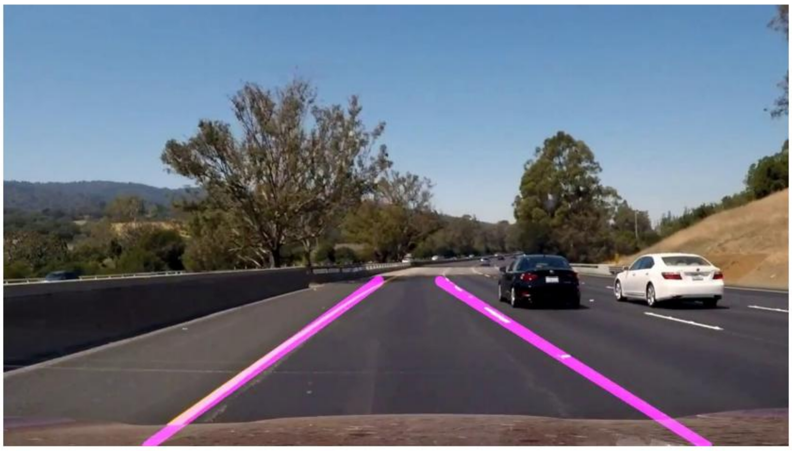
Prof. Dr.-Ing. Volker Willert

Exercise sheet 3

In both image processing and point cloud processing, many sub-problems can be solved with a compensation calculation. For example,

- determining intrinsic camera parameters based on 2D-3D point correspondences, • fitting a linearly parameterized 2D curve (e.g. straight line or parabola) into a set of image point coordinates dinates, or
- calculating the camera orientation based on more than three world points, etc.

In particular, the fitting of contours is formulated and solved as a balancing problem. 2D coordinates along a contour are given. The model parameters of the contour, e.g. a straight line, are sought. Such calculations are used in the field of automated mapping or visually supported driver assistance, among other things.



The general problem of the adjustment calculation is:
Given are N data pairs (x_i , t_i), i = 1, ..., N and k < n functions f_1 to f_k. We are looking for a linear combination $f(x) = \sum_{j=1}^{k} w_j f_j(x)$ of this f_j , so that the sum of the squared deviations of f at the points x_i to t_i is minimal becomes:

$$\operatorname{argmin}_w \sum_{i=1}^{N} (f(x_i) - t_i)^2 = \operatorname{argmin}_w \sum_{i=1}^{N} \left(\sum_{j=1}^{k} w_j f_j(\mathbf{x_i}) - t_i\right)^2$$

This problem is also called *Gaussian least squares* or *regression.*

Task 3.1: Fit 2D line

The following n = 3 pixel coordinates are given: (x_1, y_1) = (1, 1), (x_2, y_2) = (2, 1), (x_3, y_3) = (3, 2).

- Specify all k functions fj and determine the pseudoinverse.

Answer:

$$f(x_i) = a_1 x_i + a_2 \quad \Rightarrow \quad \begin{array}{l} f_1(x_i) = x_i \\ f_2(x_i) = 1 \end{array} \quad \Rightarrow \quad f(x_i) = \begin{pmatrix} x_i \\ 1 \end{pmatrix} \quad \Rightarrow \quad F(x) = \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 1 \end{pmatrix}$$

$$y = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}$$

Determination of the pseudoinverse F^+:

$$F^+(x) = (F^T(x)F(x))^{-1}F^T(x)$$

$$= \left(\begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 1 \end{pmatrix} \right)^{-1} \begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 14 & 6 \\ 6 & 3 \end{pmatrix}^{-1} \begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \end{pmatrix}$$

$$= \frac{1}{6} \begin{pmatrix} 3 & -6 \\ -6 & 14 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \end{pmatrix}$$

$$= \frac{1}{6} \begin{pmatrix} -3 & 0 & 3 \\ 8 & 2 & -4 \end{pmatrix}$$

- Calculate the optimal parameters w of the line using the pseudonormal solution.

Answer:

$$w = F^+(x)y = \frac{1}{6} \begin{pmatrix} -3 & 0 & 3 \\ 8 & 2 & -4 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} = \frac{1}{6} \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{3} \end{pmatrix}$$

- Evaluate the quality of your best fit lines using the correlation coefficient.

Answer:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{1}{3}(1+2+3) = 2$$

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{3}(1+1+2) = \frac{4}{3}$$

$$r_{xy} = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}}{\sqrt{\left(\sum_{i=1}^{n} x_i^2 - n\bar{x}^2\right)\left(\sum_{i=1}^{n} y_i^2 - n\bar{y}^2\right)}} = \frac{(1+2+6-3\cdot 2\cdot \frac{4}{3})}{\sqrt{(1+4+9-3\cdot 4)(1+1+4-3\cdot \frac{16}{9})}} = \frac{1}{\sqrt{2\cdot \frac{2}{3}}} \approx 0.87$$

Since the value of the empirical correlation coefficient is less than 1, there is no strictly linear relationship between x and y . However, since its value is “close” to 1, the best-fit line approximates the data relatively well.

Task 3.2: Interpolation of depth values

Given four depth values at four neighboring pixels arranged in a square in a depth image: Z(100, 50) = 6, Z(101, 50) = 10, Z(100, 51) = 4 and Z(101, 51) = 1.
Approximate the depth values Z_i of this 2 × 2 image section with a second order polynomial $Z_i \approx w_1 x_i^2 + w_2 x_i y_i + w_3 y_i^2$.

Answer:

$$f(x_i) = \begin{pmatrix} x_i^2 \\ x_i y_i \\ y_i^2 \end{pmatrix} \quad \Rightarrow F(x) = \begin{pmatrix} x_1^2 & x_1 y_1 & y_1^2 \\ \vdots & \vdots & \vdots \\ x_4^2 & x_4 y_4 & y_4^2 \end{pmatrix} = \begin{pmatrix} 10000 & 5000 & 2500 \\ 10201 & 5050 & 2500 \\ 10000 & 5100 & 2601 \\ 10201 & 5151 & 2601 \end{pmatrix}, \quad y = \begin{pmatrix} 6 \\ 10 \\ 4 \\ 1 \end{pmatrix}$$

First, the pseudoinverse $F(x)^+$ calculated:

$$F^+(x) = \left(F^T(x)F(x)\right)^{-1} F^T(x) = \begin{pmatrix} -0.2469 & 0.2515 & 0.2510 & -0.2554 \\ 0.9849 & -0.9951 & -1.0049 & 1.0147 \\ -0.9820 & 0.9842 & 1.0059 & -1.0076 \end{pmatrix}$$

With this result, the pseudonormal solution a^+ can be determined:

$$a^+ = F^+(x)y = \begin{pmatrix} 1.7820 \\ -7.0460 \\ 6.9659 \end{pmatrix}$$

The sought hyperparabola therefore has the form

$$Z_i(x_i) = 1,782 x_i^2 - 7.046 x_i y_i + 6.9659 y_i^2$$

Task 3.3: The RANSAC algorithm

Four image point coordinates are given: $x_1 = (1, 1)$, $x_2 = (2, 1)$, $x_3 = (3, 2)$ and $x_4 = (1, 3)$, where three points are near a straight edge and one is not (outlier). Determine these three points and the straight line equation that best approximates the edge using the RANSAC algorithm. Proceed as follows:

- Plot the four points in a coordinate system. • RANSAC Step 1:

  Randomly select as many points from the data points as are necessary to calculate the parameters of the model. This is done in the expectation that this set is free of outliers. (For practice purposes, select “randomly” x_1 and x_4 in the first iteration step, and x_2 and x_3 in the second).

- RANSAC Step 2: Determine the model parameters (line parameters) using the selected points.

- RANSAC Step 3: Determine the subset of the measured values whose distance from the model curve is smaller than a certain limit g (g = 0.5 is chosen for this example). This subset is called *the consensus set* . If it contains a certain minimum number of values, a good model has probably been found and the consensus set is saved.

- Repeat steps 1 − 3 several times.

The RANSAC algorithm is now applied to all 6 possible combinations of two image point coordinates in order to find a straight line approximation y = ax + b of the points, which is not disturbed by outliers.

in the following the notation $x_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix}$.

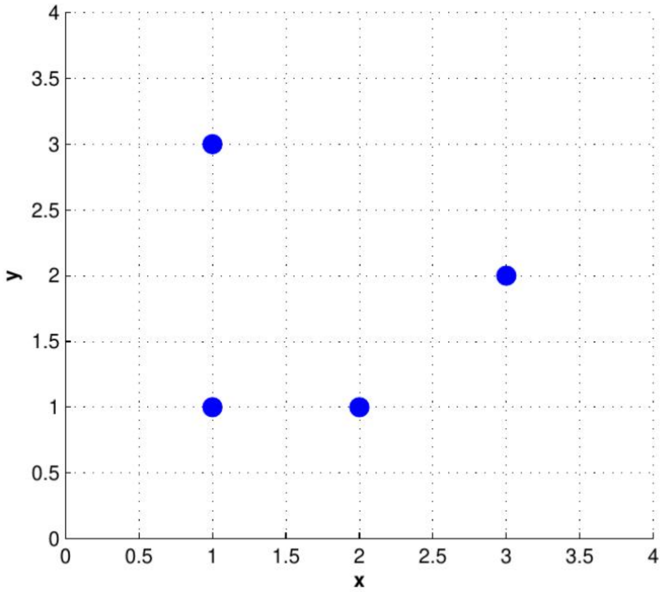


Figure 1: Plot of pixel coordinates

1st iteration: x1 = 1 1 , x4 = 1 3

First, a straight line equation is determined for these points. This can be done by plotting the line in the coordinate system and reading the parameters, or analytically. Using the Hesse normal form, it is easy to determine the distance of a point from a straight line, so that the RANSAC steps are then calculated analytically.

y = ax + b = y4 ÿ y1 y4 ÿ y1 x + y1 ÿ x1 x4 ÿ x1 x4 ÿ x1

ÿ (x4 ÿ x1)y = (y4 ÿ y1)x + y1(x4 ÿ x1) ÿ x1(y4 ÿ y1)

ÿ 0 = 2x ÿ 2

$\boxed{G1 : x = 1}$

The Hesse normal form has the general form

x cos ÿ + y sin ÿ ÿ p = 0,

where ÿ is the angle between the abscissa and the normal vector from the origin to the line and is calculated from the Connection

ÿ = ÿ arctan a 1

calculated. For the randomly selected data of the first iteration step, the line representation in Hesse normal form is

1 ÿ = ÿ arctan a = 0 ÿ x ÿ 1 = 0.

With this representation, by inserting the remaining points into the equation, the distances to this Specify straight lines:

|d(x2, G1)| = 1 > 0, 5 |d(x3,

G1)| = 2 > 0.5

Since both points are above the permissible limit of g = 0.5 , there is no point for this iteration that supports the line model. The *consensus set* is therefore empty:

S1 = {}

The RANSAC steps 1-3 are then applied to further random combinations of two pixel coordinates.

2nd iteration: x2 = 2 1 , x3 = 3 2

y = ax + b = 1 x + 1 ÿ 2 · 1 = x ÿ 1 ÿ G2 : y = x ÿ 1 1

G2 in Hesse normal form:

1 ÿ = ÿ arctan a 1ÿ 4 ÿ x cos ÿ + y sin ÿ ÿ p = ÿ 2 1 x ÿ 1 y ÿ ÿ 2 ÿ 2 1 = 0

|d(x1, G2)| = 1 ÿ 2 · 1 ÿ 1 ÿ 2 · 1 ÿ 1 ÿ 2 = 1 2 > 0.5 ÿ

|d(x4, G2)| = 1 ÿ 2 · 1 ÿ · 3 ÿ ÿ 1 2 ÿ 2 1 = 3 > 0.5 ÿ 2

ÿ S2 = {}

3rd iteration: x1 = 1 1 , x2 = 2 1

0 0 y = ax + b = x + 1 ÿ 1 · = 1 ÿ G3 : y = 1 1 1

G3 in Hesse normal form:

1 ÿ = ÿ arctan a = ÿ 2 ÿ x cos ÿ + y sin ÿ ÿ p = y ÿ 1 = 0

|d(x3, G3)| = |2 ÿ 1| = 1 > 0, 5 |d(x4, G3)|

= |3 ÿ 1| = 2 > 0.5

ÿ S3 = {}

4th iteration: x1 = 1 1 , x3 = 3 2

1 1 y = ax + b = x + 1 ÿ 1 · 2 2 = 1 2 x + 1 2 G4 : y = x + 2 2 1 1 ÿ

G4 in Hesse normal form:

ÿ = ÿ arctan a 1 = ÿ1.1071 ÿ x cos ÿ + y sin ÿ ÿ p = 0.4472x ÿ 0.8944y + 0.4472 = 0

|d(x2, G4)| = |0.4472 · 2 ÿ 0.8944 · 1 + 0.4472| = 0.4472 < 0.5 |d(x4, G4)|

= |0.4472 · 1 ÿ 0.8944 · 3 + 0.4472| = 1.7888 > 0.5

ÿ S4 = {x2}

5th iteration: x2 = 2 1 , x4 = 1 3

2 y = ax + b = ÿ x + 1 ∓ 2 · 1 1 $\frac{2}{}$ = ÿ2x + 5 ÿ G5 : y = ÿ2x + 5

G5 in Hesse normal form:

ÿ = ÿ arctan a $\frac{1}{}$ = 0.4636 ÿ x cos ÿ + y sin ÿ ÿ p = 0.8944x + 0.4472y ÿ 2.2361 = 0

|d(x1, G5)| = |0.8944 · 1 + 0.4472 · 1 ÿ 2.2361| = 0.8945 > 0.5 |d(x3, G5)|

= |0.8944 · 3 + 0.4472 · 2 ÿ 2.2361| = 1.3415 > 0.5

ÿ S5 = {}

6th iteration: x3 = $\begin{matrix} 3 \\ 2 \end{matrix}$, x4 = $\begin{matrix} 1 \\ 3 \end{matrix}$

1 y = ax + b = ÿ x + 3 2̄ 2 $\frac{1}{}$ 1 7 x + 2 $\frac{}{2}$ y = ÿ x + 2 2 1 7 ÿ G6 :

G6 in Hesse normal form:

ÿ = ÿ arctan a $\frac{1}{}$ = 1.1071 ÿ x cos ÿ + y sin ÿ ÿ p = 0.4472x + 0.8944y ÿ 3.1305 = 0

|d(x1, G6)| = |0.4472 · 1 + 0.8944 · 1 ÿ 3.1305| = 1.7889 > 0.5 |d(x2, G6)|

= |0.4472 · 2 + 0.8944 · 1 ÿ 3.1305| = 1.3417 > 0.5

ÿ S6 = {}

Since set S4 contains the most elements, the model of the fourth iteration is chosen. As can be seen from the following plot, this represents a good approximation for the three image coordinates near the edges, without the outlier disturbing this model.

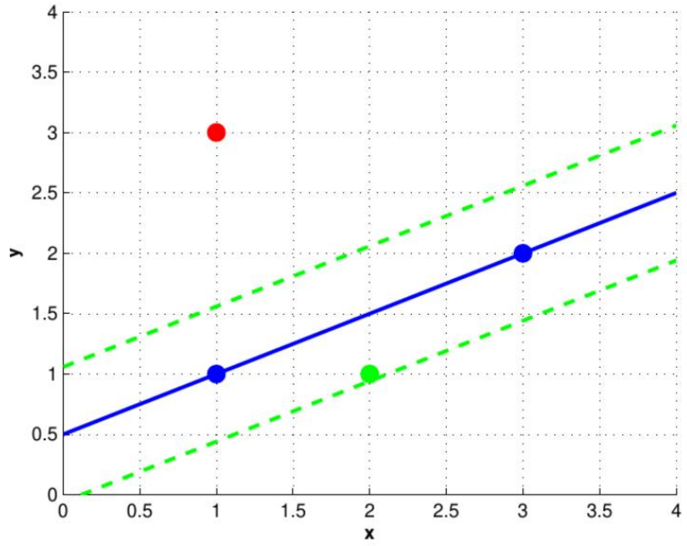


Figure 2: Result of the RANSAC algorithm with g = 0.5

After several iterations have been carried out, the subset that contains the most points is selected (if one was found). Only this subset is used to calculate the model parameters using one of the usual adjustment methods. An alternative variant of the algorithm ends the iterations early if enough points support the model in step 3. This variant is called preemptive - i.e. prematurely terminating - RANSAC. With this procedure, it must be known in advance how large the proportion of outliers is, so that it can be estimated whether enough measured values support the model.

The algorithm essentially depends on three parameters:

1. the maximum distance of a data point from the model up to which a point is not considered a gross error;
2. the number of iterations and
3. the minimum size of the consensus set, i.e. the minimum number of points consistent with the model.

• What happens if you choose the limit g = 0.1? Three points are never assigned. • What happens if you choose the limit g = 2? All four points are always assigned. • What does this mean for the choice of limit? Not too small and not too large, depends on the distribution of the data points in the data room.

The number of repetitions can be set so that an outlier-free subset of the data points is selected at least once with a certain probability pn(outlier = 0) . To do this, we define the following parameters: the number of data points s that are necessary to calculate a model, the relative proportion of outliers in the data ÿ and the number of repetitions n.

The minimum number n of required repetitions depends only on the relative proportion of outliers, the number of model parameters s and the probability of the occurrence of at least one outlier-free subset with n repetitions pn(outliers = 0), but not on the total number of measured values. It can be calculated as follows:

$$n = \frac{\text{ln(1 ÿ pn(outlier = 0)) ln(1 ÿ (1}}{\text{ÿ) s)}} .$$

Calculate the minimum number n of iterations required to fit a line to a point cloud with 4 points that contains an outlier. The probability of selecting at least one outlier-free subset from all data points is set to pn(outlier = 0) = 0.99.

Answer: For the example from task 4.1, with pn(A = 0) = 0.99, s = 2 and = 0.25, the following minimum number of repetitions results:

$$n = \frac{\ln(1 - 0,99)}{\ln(1 - (1 - 0,25)^2)} = 5,57 = 6$$