CS-465 Computer Vision

Nazar Khan

PUCIT

10. Robust Estimation – RANSAC

Outliers

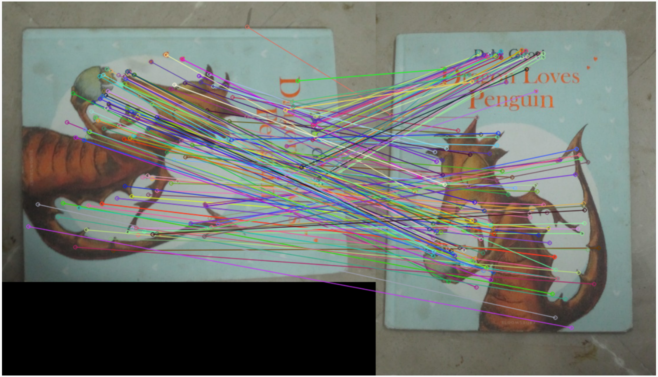


Figure: Correspondences obtained after matching SIFT descriptors. Most correspondences seem to be correct but some *outlier* correspondences can also be seen. Outliers will affect the estimation of the transformation between the two images.

Line Fitting

- Given points $(x_1, y_1), \ldots, (x_N, y_N)$, find the parameters (m, c) of the *best-fit* line.
- One definition of best-fit could be in terms of sum-squared-error (SSE).

$$E(m, c) = \frac{1}{2} \sum_{n=1}^{N} (y_n - mx_n - c)^2$$

  so that $(m^*, c^*) = \arg\min_{m,c} E(m, c)$.
- Equating gradients to zero yields

$$\sum_{n=1}^{N} -x_n(y_n - m^* x_n - c^*) = 0$$
$$\sum_{n=1}^{N} -(y_n - m^* x_n - c^*) = 0$$

Line Fitting

- Can be written as a linear system

$$\begin{bmatrix} \sum_{n=1}^{N} x_n^2 & \sum_{n=1}^{N} x_n \\ \sum_{n=1}^{N} x_n & \sum_{n=1}^{N} 1 \end{bmatrix} \begin{bmatrix} m^* \\ c^* \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{N} x_n y_n \\ \sum_{n=1}^{N} y_n \end{bmatrix}$$

- In such a fitting squared error due to point (x_n, y_n) is $(y_n - mx_n - c)^2$.
- For a line that passes through the inliers, the outliers will contribute large values to the SSE.

> Therefore, minimizing SSE will cause the best-fit line to pass closer to the outliers.

Line Fitting

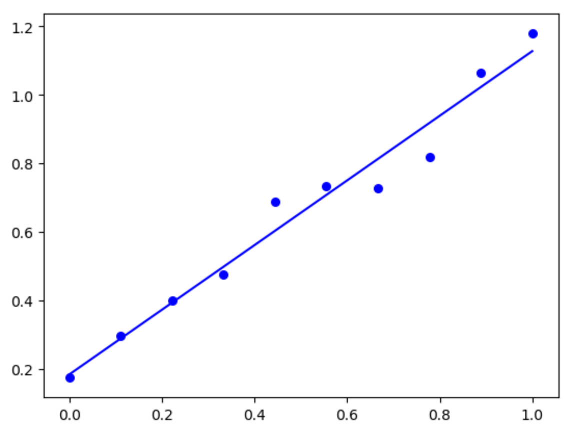


Figure: Best-fit line for data with no outliers. Author: N. Khan (2018)

Line Fitting

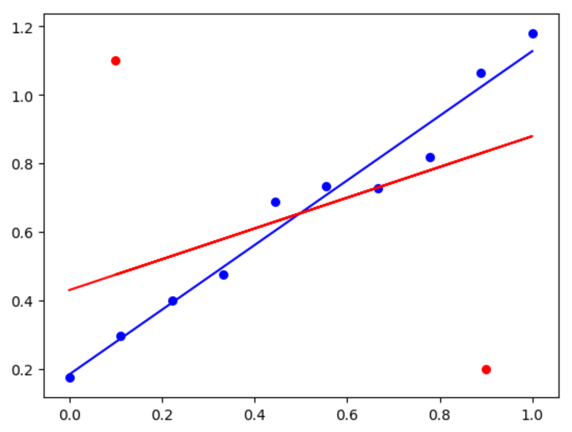


Figure: Best-fit line for data with outliers. Author: N. Khan (2018)

Robust Estimation via RANSAC

- In the presence of outliers, how can we fit a model *robustly*?
- One solution is the *RAndom Sample And Consensus* algorithm[1]. Shortened as *RANSAC*.

Basic Idea

1. Choose a small subset of points uniformly at random.
2. Fit a model to that subset.
3. Find all points that are 'close' to the model and reject the rest as outliers.
4. Repeat step 1-3 many times and choose model with fewest outliers.

[1] Fischler and Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography".

Robust Line Fitting via RANSAC

- A line requires at least two points. The RANSAC algorithm applied to line fitting is as follows:
  1. Pick any two points randomly.
  2. Fit a line (m, c) between them.
  3. Consider data point (x_n, y_n) to lie on the line if $(y_n - mx_n - c)^2 < \tau_1$. Such points are *inliers*. Remaining points do not support this line. They are *outliers*.
  4. If ratio of inliers is above a threshold τ_2, then stop. Otherwise, go to step 1.

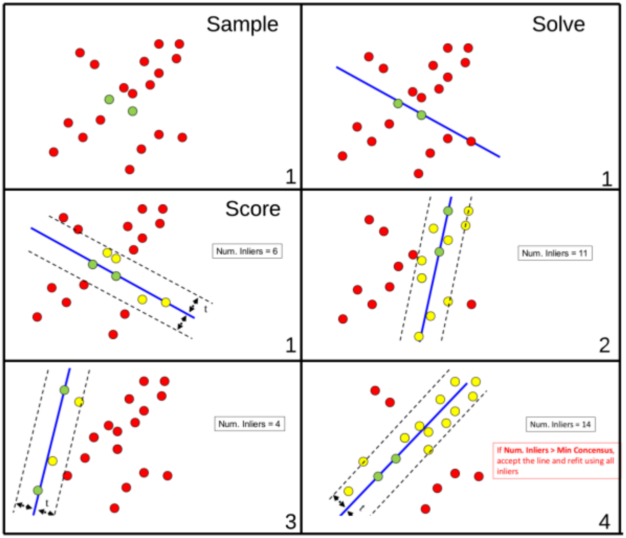
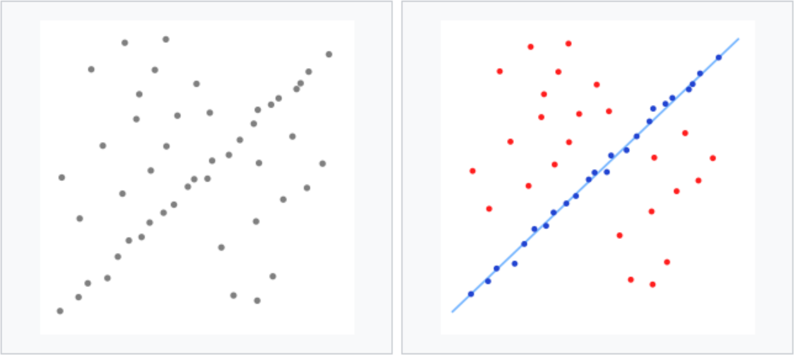


Figure: Author: F. Moreno



A data set with many outliers for which a line has to be fitted.

Fitted line with RANSAC; outliers have no influence on the result.

Figure: Source: `https://en.wikipedia.org/wiki/Random_sample_consensus`

Robust Estimation via RANSAC
In General

- The RANSAC algorithm can be applied to any general estimation problem in the presence of outliers.
- Denoting model parameters by θ, error function by $f(\theta)$ and assuming that minimum points required to fit the model is K, RANSAC is as follows:
  1. Pick any K points randomly.
  2. Fit the parameters θ for selected points.
  3. If ratio of inliers ($f(\theta) < \tau_1$) is above a threshold τ_2, then stop. Otherwise, go to step 1.
- Can be used to *robustly estimate* parameters of
  1. Affine Transformation
  2. Homography
  3. Fundamental Matrix

Robust Homography Estimation via RANSAC

Figure: Projective warping by homography estimated from all SIFT matches (inliers and outliers).

Robust Homography Estimation via RANSAC

Figure: Projective warping by homography estimated from inliers only.

OpenCV Example

```
# Find homography
M, mask = cv.findHomography(src_pts, dst_pts)
M_ransac, mask = cv.findHomography(src_pts, dst_pts, cv.RANSAC)

# Use homography
height, width, channels = img2.shape
img1Reg = cv.warpPerspective(img1, M, (width, height))
img1Reg_ransac = cv.warpPerspective(img1, M_ransac, (width, height))
```