Pattern Recognition Systems – Lab 2

RANSAC – fitting a line to a set of points

1. Objectives

The purpose of this laboratory session is to use the RANSAC method for line fitting.

2. Theoretical Background

Random Sample Consensus (RANSAC) is a paradigm for fitting a model to experimental data, introduced by Martin A. Fischler and Robert C. Bolles in 1981 [1].

As stated by Fischler and Bolles "The RANSAC procedure is opposite to that of conventional smoothing techniques: Rather than using as much of the data as possible to obtain an initial solution and then attempting to eliminate the invalid data points, RANSAC uses as small an initial data set as feasible and enlarges this set with consistent data when possible".

The RANSAC algorithm is summarized below [2]:

Objective: Robust fit of a model to a data set S which contains outliers.

Algorithm:

1. Randomly select a sample containing a number of s data points from S and instantiate the model from this subset.

2. Determine the set of data points S_i which is within a distance threshold t of the model. The set S_i, is the consensus set of the sample and defines the inliers of S.

3. If the size of S_i (the number of inliers) is greater than some threshold T, re-estimate the model using all the points in S_i and terminate.

4. If the size of S_i is less than T, select a new subset and repeat the above.

5. After N trials the largest consensus set S_i is selected, and the model is re-estimated using all the points in the subset S_i.

2.1. RANSAC for fitting a line to a set of points

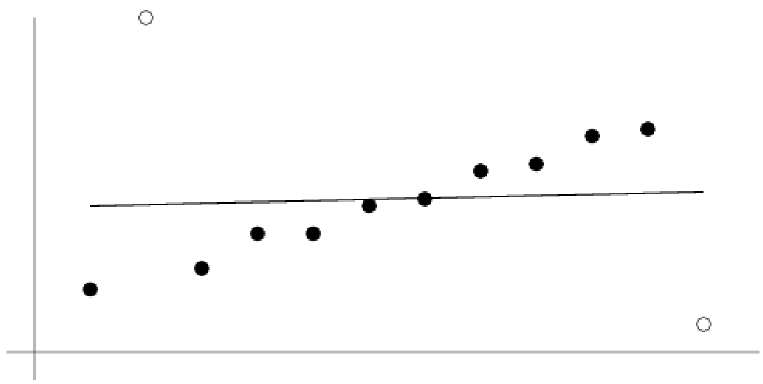

Figure 1-a

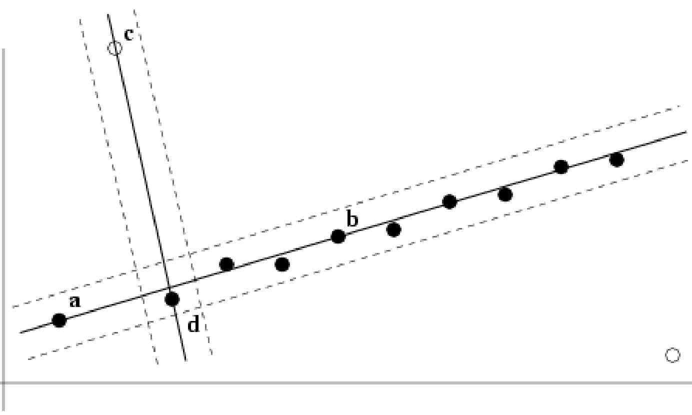

Figure 1-b

The problem, illustrated in Figure 1-a is the following: given a set of 2D data points, find the line which minimizes the sum of squared perpendicular distances (orthogonal regression), subject to the condition that none of the valid points deviates from this line by more than t units. This is actually two problems: a line fit to the data; and a classification of the data into inliers (valid points) and outliers. The threshold t is set according to the measurement noise (for example $t = 3\sigma$), and is discussed below.

The first step is to select two points randomly; these points define a line. The *support* or *consensus set* for this line is measured by the number of points that lie within a distance threshold. This random selection is repeated a number of times and the line with most support is deemed the robust fit. The points within the threshold distance are the inliers (and constitute the eponymous *consensus* set). The intuition is that if one of the points is an outlier then the line will not gain much support.

Furthermore, scoring a line by its support has the additional advantage of favoring better fits. For example, the line (a, b) in figure 1-b has a support of 10, whereas the line (c, d), where the sample points are neighbors, has a support of only 2. We can deduce from this that *c* or *d* is a noise point.

We can discuss three main issues that result from the presented algorithm:

1. **What is the distance threshold?** We would like to choose the distance threshold *t,* such that a point is an inlier with a given probability. For this we require the probability distribution for the distance of an inlier from the model (measurement error model). In practice the distance threshold is usually chosen empirically. However, if it is assumed that the measurement error is Gaussian with zero mean and standard deviation σ, then a value for *t* may be computed.

2. **How many trials?** It is often computationally infeasible and unnecessary to try every possible sample. Instead the number of samples *N* is chosen sufficiently high to ensure with a probability *p,* that at least one of the random samples of *s* points is free from outliers. A typical value for *p* is 0.99. Suppose *q* is the probability that any selected data point is an inlier, and thus q^s is the probability that all *s* points are inliers. The complementary event is that there is at least an outlier among the *s* points has probability $1-q^s$. Then the probability for *N* selections to each have at least 1 outlier is $(1-q^s)^N$ which must be equal to $1-p$. We find the number of required trials *N* is equal to $log(1 - p)/log(1-q^s)$.

3. **How large is an acceptable consensus set?** A rule of thumb is to terminate if the size of the consensus set is similar to the number of inliers believed to be in the data set, given the assumed proportion of outliers, i.e. for *n* data points $T=qn$. For the line-fitting example of Figure 1, a conservative estimate of $q = 0.8$, so that $T = 0.8 \cdot 12 = 9.6$.

3. Mathematical background

The equation of a line through two distinct points (x_1, y_1) and (x_2, y_2) is given by:

$$(y_1 - y_2)X + (x_2 - x_1)Y + x_1y_2 - x_2y_1 = 0$$

The distance from a point (x_0, y_0) to a line given by $aX+bY+c = 0$ is:

$$d = \frac{|ax_0 + by_0 + c|}{\sqrt{a^2 + b^2}}$$

4. Practical background

Opening an image as grayscale:
```
Mat img = imread("filename", CV_LOAD_IMAGE_GRAYSCALE);
```

Creating a grayscale image:
```
Mat dst(height, width, CV_8UC1); //8bit unsigned 1 channel
```

Accessing the pixel at position row *i* and column *j*:
```
uchar pixel = img.at<uchar>(i,j); //unsigned char type
```

A black point from the image at position `(i,j)` corresponds to a point at coordinates `x=j, y=i`:

```
if (img.at<uchar>(i,j)==0){
     Point p; p.x = j; p.y = i;
}
```

Modifying the pixel at position row *i* and column *j*:

```
img.at<uchar>(i,j) = 255; //white
```

Draw a line between two points:

```
line(img, Point(x1, y1), Point(x2, y2), Scalar(B,G,R));
```

Viewing the image:

```
imshow("title", img);
waitKey();
```

5. Practical work

1. Open the image and construct the input point set by finding the positions of all black points.
2. Calculate the parameters *N* and *T* after setting $t=10$, $p=0.99$, $q=0.8$ and $s=2$.
   For points1.bmp use $q=0.3$.
3. Apply the RANSAC method:
   a. Choose two different points;
   b. Determine the line equation passing through the selected points;
   c. Find the distances of each point to the line;
   d. Count the number of inliers;
   e. Save the line parameters *(a,b,c)* if the current line has the highest number of inliers so far;
   f. Write the correct termination conditions (based on the size of the consensus set and the maximum number of iterations).
4. Draw the optimal line found by the method.

6. References

[1] Robert C. Bolles, Martin A. Fischler: *A RANSAC-Based Approach to Model Fitting and Its Application to Finding Cylinders in Range Data*, 1981
[2] Richard Hartley, Andrew Zisserman: *Multiple View Geometry in Computer Vision,* 2003