Studienarbeit

Patrick Tutzauer

Line Based Object Recognition and Tracking using Sequential Monte Carlo Methods

Motivation

In Computer Vision there are many different approaches to detect objects in images.
This research project deals with object recognition in video sequences, utilizing a specific kind of filter - the *Particle Filter* (PF). Obviously, the challenge in the case of a video sequence is to perform all calculations in (quasi) real-time. Therefore computation intensive approaches like *Hough-Transformation* don't come into consideration. When attempting to track objects in image sequences there are several methods for this issue. All of them have some kind of prediction and update step in common, as for example the *Kalman-Filter.* The weakness of most of these filters is the sensitivity to great amounts of outliers. In contrast, the *Particle Filter* is able to handle outliers quite well. The task of this research project is to implement an object recognition and tracking algorithm by means of *RANSAC* and the *Particle Filter.* Image acquisition is performed by a uEye UI225x SE-M camera with a 12mm focal length object lens. The test object is a small wooden winner's podium with ground truth. The object has 16 edge points. These points lead to 22 lines which represent the object's shape.

Approach

The algorithm can be divided into 3 main stages:
1. *Sobel Filtering* and Binarization
2. *RANSAC* for Line Detection
3. *Particle Filtering*

1. *Sobel Filtering* and Binarization

In the beginning some very basic image processing steps are applied to the image. First of all the *Sobel operator* is applied to the image in x and y direction to extract edges. Subsequently, a gradient directions matrix and a matrix containing absolute values of the edge images are computed. Based on the absolute image a fixed-level threshold binarization is performed. The obtained binary image serves as input for the line detection.

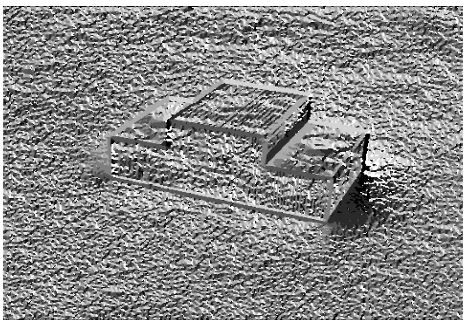

Figure 1: Gradient directions image

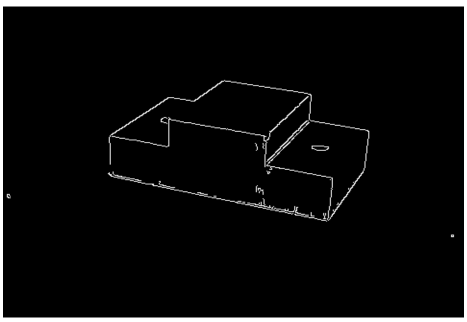

Figure 2: Binary image

2. *RANSAC* for Line Detection

The chosen line representation for this work is the normal form, rather than the more common slope-intercept form. Using this form, a line is defined by its perpendicular distance from the origin d and the line rotation angle α. The location of all pixels with grey value 255 (white) in the binary image are determined, since considered lines will only pass through edge pixels. From this set one pixel is randomly selected. Knowing the location of the randomly selected edge pixel in the binary image, one can pick the direction of the gradient from the previously computed gradient directions matrix at the same location. This gradient direction defines the line rotation angle. Based on the x and y coordinate of the randomly selected pixel and its corresponding α, the perpendicular distance from the origin is determined. For all remaining edge pixels, based on the given α, this distance d is determined. If the distance difference between the randomly chosen pixel and each remaining one is smaller than a predefined threshold for a sufficient number of pixels, a line is considered to be found. This procedure is done iteratively. In every step the amount of edge pixels is reduced by removing the pixels belonging to the current detected line. Once a valid line has been found, an additional *Least Squares* estimation for the parameter set is performed to increase accuracy of the line.

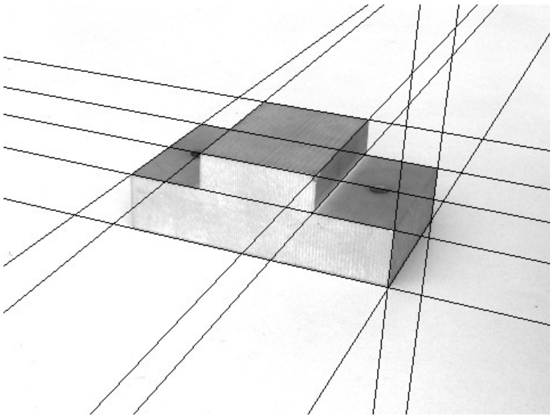

Figure 3: Lines detected by RANSAC and subsequent Least Squares estimation

3. *Particle Filtering*

The *PF* is a non-parametric filter. It is a probabilistic approach based on the theorem of Bayes. Posterior probabilities in the probability density function (PDF) are represented by randomly chosen discrete samples with associated weights. In this case the PDF and therefore the particles refer to the exterior orientation (EO) of the camera acquiring the video sequence. Therefore every particle can be regarded as the representation of a guessed camera viewpoint.

The *PF* process consists of the following steps:

- Initialization
- Projection and Line Parameter Determination
- Evaluation
- Weighting
- Spreading

Initialization

Initialization means the spreading of the first set of particles. The total number of particles is chosen as 500. Each particle contains 6 parameters defining the EO (translation and rotation angles). Finding a reasonable set of initial particles is an issue to cope with. Since the parameters of the EO represent six degrees of freedom a completely random initialization would produce too many possibilities for the guessed camera viewpoint, leading to unreasonable results in the subsequent evaluation.

Projection and Line Parameter Determination

With the initialized particles a projective transformation can be performed. Based on each EO the 3D coordinates defining the corners of the object are projected into the 2D image coordinate system. Subsequently for every particle all possible 22 line parameter sets are determined.

Evaluation
The aim of this step is to evaluate the quality of each particle and each estimated EO, accordingly. Detected lines and their corresponding line parameters in the actual image can be referred to "measured" values. The 22 line parameters for each particle represent "model" values. Based on the euclidean distance in parameter space between "measured" values and their "nearest model neighbours"s the quality for each particle is determined.

Weighting
Basic idea for this step: the smaller the euclidean distance for a particle calculated in the *Evaluation* step, the higher its weight.

Spreading
The best estimated EO gets the highest weight and therefore the highest number of tracking particles is spread around this particle. Based on the weights computed earlier the PF tries to concentrate its particles in areas, where good EOs led to good matches of the line parameters. Additionally some particles will be produced with an extremely wide spreading range to handle fast and large changes of the object position and simulate noise, to increase the detection speed once the object tracking lost trace.

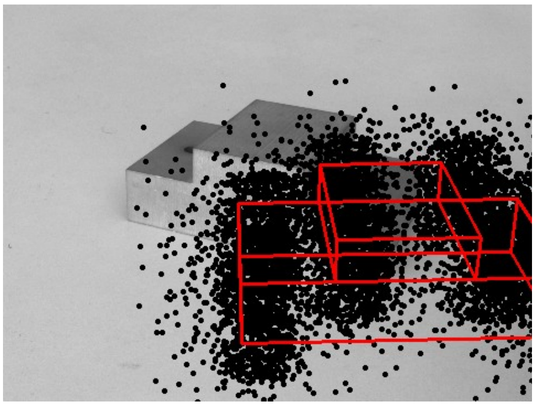

Figure 4: Initialization of the PF. The black dots are the edge points of the object projected into the image based on the exterior orientations of each particle. The red lines show the projection of the best evaluated particle.

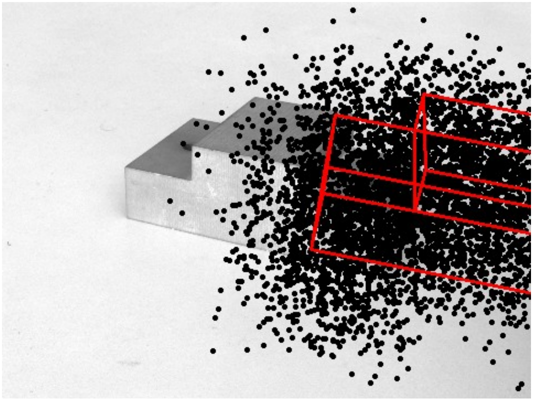

Figure 5: Few iterations later one can see the changing of the projected points. The best match leads to an already rotated object in comparison to the result of the Initialization.

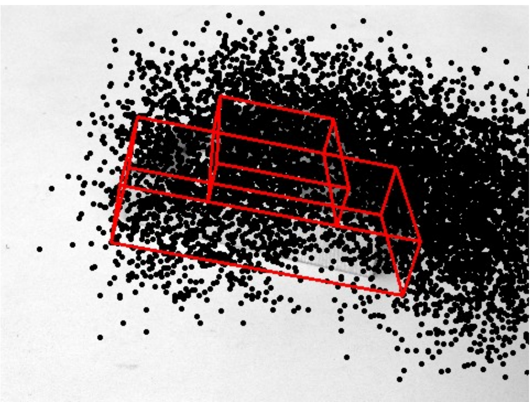

Figure 7: In this figure the PF has already converged to the actual location of the object, but is still rotated in a wrong direction.

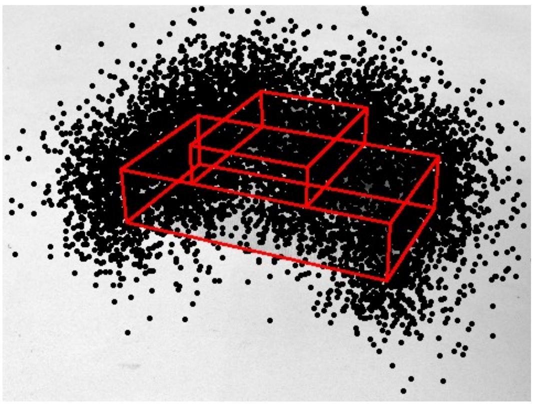

Figure 6: Finally the best projection is rotated in the right direction and the estimated EO leads to an perfect match with the actual position of the object. The object has successfully been detected.

Conclusion and Outlook

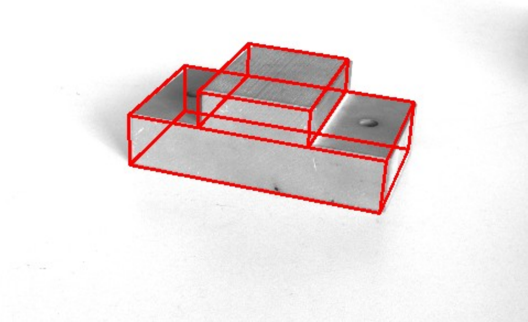

Figure 8: Figure without the projected edge points for better visibility of the match.

The presented work shall depict a mixture of template-based and feature-based tracking algorithms. Instead of the Hough Transformation, RANSAC is applied for line detection. This has the advantage of low computational costs. In addition, artificial views for the object are generated and evaluated on the fly, which again saves memory for LUTs. In this project the test object was line-based. However, the approach is expandable to different shapes and segments, for example circles, since it is only a question of evaluating different parameters in parameter space. When putting further effort in the Evaluation and Spreading, this approach will surely be a fast and robust way to recognize and track objects in video sequences efficiently.