in: *Proceedings of the IEEE International Conference on Visual Information Engineering*, Bangalore, India, September 26-28, 2006

SUPPLEMENTING BUNDLE ADJUSTMENT WITH EVOLUTIONARY ALGORITHMS

Lars Heyden*, Rolf P. Würtz*, Gabriele Peters†

*Institut für Neuroinformatik, Ruhr-Universität Bochum, D-44780 Bochum, Germany
†Universität Dortmund, Informatik VII, Otto-Hahn-Str. 16, D-44227 Dortmund, Germany
Correspondence: Lars.Heyden@udo.edu

Keywords: 3D Reconstruction, 3D Model Acquisition, Bundle Adjustment, Evolutionary Algorithms.

Abstract

Bundle Adjustment is a common technique to improve results of any multiple view reconstruction algorithm to obtain 3D structure for computer vision and computer graphics. If the error of a reconstruction can be expressed by an error function, this function can be minimized by numerical methods such as the Levenberg-Marquardt algorithm. By this means, the reconstruction can often be significantly improved. Unfortunately, there is no guarantee for the detected minimum of being a global minimum, since numerical optimization algorithms converge at local minima. The idea presented in this paper is to support the optimization process by evolutionary algorithms. While the existence of fast Levenberg-Marquardt algorithms allow for an obviously faster solution than evolutionary algorithms, the latter can mitigate their disadvantage of getting trapped in local minima. We demonstrate the combination of Bundle Adjustment with an evolutionary algorithm by means of 3D reconstruction of objects from visual information only.

1 Introduction

In this paper, we start by presenting a system that acquires an image sequence of an object and reconstructs a 3D-model of this object using exclusively camera images. The model consists of a cloud of unlinked space points. By projecting these space points back into the images a reprojection error can be measured. It can be expressed by an error function and by applying Bundle Adjustment we try to minimize it with a Levenberg-Marquardt algorithm. As the initial reconstruction provided by the system is rather deficient in most cases the algorithm generally converges at an unsatisfactory local minimum. In order to overcome this situation the system uses a second optimization strategy by applying a simple evolutionary algorithm. We will illuminate this strategy in the third section. By regarding an exemplary result in the fourth section, we will show that evolutionary algorithms provide the opportunity to elude or overcome local minima. Afterwards the universality and expendability of the presented ideas are briefly discussed.

2 A System Acquiring Object Models using Visual Information only

In this section, we describe the object acquisition system, which is the framework that allows for the application of the ideas presented in this paper. An extensive description can be found in [3].

2.1 Acquisition of an image sequence

The first step in the work flow of the system is the acquisition of an image sequence. The setup of this procedure is shown in figure 1. An object is placed in the middle of a turntable. A camera is fixed to a robot arm and by moving turntable and robot arm the camera can capture an arbitrary image sequence of the object. Between two pictures of a sequence, the turntable moves about 2 degrees.

The background of all images in the sequence is eliminated by background subtraction. The background images are acquired by generating an additional image sequence after removing the object.

2.2 Generation of Image Point Correspondences

In the next step, point correspondences are generated between all images of the sequence. For this purpose, interesting points are chosen in the images. In the described system, this task is done by calculating difference images between successive images of the sequence and denoting points where changes occur. These points are tracked to the next images. The applied tracking method is based on the Gabor responses at the image points. For details see [5]. In this way, features of the object can be tracked along several images. While processing the sequence tracked points are constantly dropped (e.g. when they are not visible anymore) and replaced by new points.

2.3 Reconstruction of a Projective 3D-Model

On the basis of point correspondences the epipolar geometry between several image pairs can be calculated. Afterwards, for each correspondence a space point can be calculated per triangulation. Because no calibration of the cameras or any other knowledge is assumed we derive a projective model only (i.e., a projective cloud of space points). To cover the whole image sequence we need to consider several image pairs and receive several partial reconstructions that need to be merged. Since the reconstructions are projective they differ by projective transformations that have to be calculated. This can be done if a sufficient number of point correspondences in one image pair is also visible in other image pairs. With the parameters chosen for acquisition it is guaranteed that the images overlap sufficiently, such that each image that is part of one pair is also part of another image pair.

2.4 Measuring and Minimizing the Reprojection Error

The reconstructed space points can be reprojected in each image and the calculated reprojection is then compared to the measured image points that were basis for the calculation of the space point. The aberrations add up to the overall reprojection error. It is expressed by an error function depending on the calculated 3D-points and camera matrices. The system tries to minimize this function by applying a fast Levenberg-Marquardt algorithm (presented in [2]) and, in addition, a simple evolutionary algorithm.

3 Optimizing 3D-Models by Bundle Adjustment and Evolutionary Algorithms

3.1 Setting up the Reprojection Error

The reprojection error of a reconstructed space point in one image can be expressed as

$$e_{\mathbf{x}'} = d(P'\mathbf{X}', \mathbf{x}) \tag{1}$$

where $\mathbf{x}$ is the measured image point, $\mathbf{X}'$ is the reconstructed space point, P' is the reconstructed camera matrix of the image and thus, $P'\mathbf{X}'$ is the reprojected image point. Hence, $e_{\mathbf{x}'}$ expresses the aberration between the original, measured image point and the corresponding backprojected 3D-point of the model.
To express the reprojection error of the whole reconstruction Γ we summarize the squared errors of all points in all images, attaining the overall reprojection error

$$e_\Gamma = \sum_{i,j} d(P'_j X'_i, x_{ij})^2 \tag{2}$$

We assume a Gaussian distributed measurement error and by squaring the single errors we achieve a maximum likelihood estimation.

3.2 Minimizing e_Γ by Bundle Adjustment

A fast Levenberg-Marquardt algorithm minimizing e_Γ is presented in [2]. In equation (2) we expressed e_Γ as an error function depending on a number of parameters, precisely the coordinates of the reconstructed space points and entries of the reconstructed camera matrices. This allows us to apply numerical techniques to find a local minimum in the parameter space. By partitioning the set of parameters and utilizing the sparse structure of the used matrices the minimization can be performed in reasonable time.

3.3 Minimizing e_Γ by an Evolutionary Algorithm

Evolutionary algorithms are mostly applied to "black-box"-problems. A *fitness function* depending on a vector of parameters is given and can be evaluated, but no further knowledge about this function is assumed. The aim is to find a vector of parameters - in the following called *search points* - that maximizes (or minimizes, depending on the particular problem the function models) the function. Often several search points - a *population* of search points - are treated at a time. New search points are added by choosing and modifying existing search points by randomized mutation of some values or by combining the parameters of two (or more) chosen search points. Search points are chosen with greater probability when they result in a higher (or lower, if minimization is desired) function value and search points with contrary results are removed. An extensive description of evolutionary algorithms can be found in [4], for instance.

In the presented system a simple evolutionary algorithm is used. We use equation (2) as fitness function that we want to minimize and the coordinates of the reconstructed space points and entries of the reconstructed camera matrices form the search points. The population consists of a single search point that is modified by randomized mutation of some values. The new search point replaces the old one if an improvement (in this case a decrease of the function value) compared to the previous search point occurs. The probability and variance of the mutation adapt during the processing of the algorithm, i.e., these values vary as well and are taken over if they yield a notable improvement.

4 Results and Examples

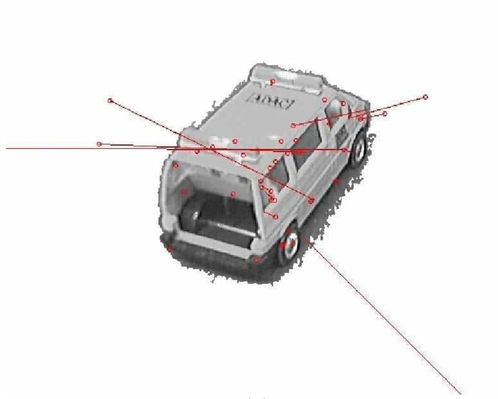
Without doubt no optimization can compensate for deficient initial reconstructions. However, both optimization strategies – Bundle Adjustment with a Levenberg-Marquardt algorithm as well as the evolutionary algorithm – can result in substantial improvement, a fact that might not be surprising. While the Levenberg-Marquardt algorithm converges at a minimum and terminates, the evolutionary algorithm has to be stopped by defining termination conditions. In fact, the runtime was simply confined by defining a maximal number of iterations. This proceeding is justified by the observation that a notable improvement can occur after a long state of stagnation, so other termination conditions do not seem more appropriate.

We applied the evolutionary algorithm on initial reconstructions as well as on reconstructions that were previously optimized by the Levenberg-Marquardt algorithm. The most notable observation was that even in the latter case the evolutionary algorithm could generally improve the results and in some cases in a substantial way. If we applied the Levenberg-Marquardt algorithm afterwards a second time, this could often lead to further improvement.

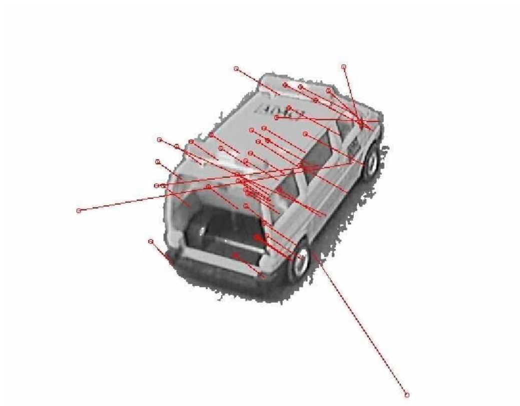
To visualize this results, in figures 2 and 3 we show the reconstruction error in two views of an exemplary reconstruction at different stages of optimization. Each reprojected space point is denoted by a small circle from which a line points to the place where the corresponding image point was measured. As these points should be identical, the lengths of the lines account for the reprojection error.

The initial reconstruction is shown in subfigures labeled (a), in the other subfigures we see the same view after application of the Levenberg-Marquardt algorithm (b), the evolutionary algorithm (c) and again the Levenberg-Marquardt algorithm (d).

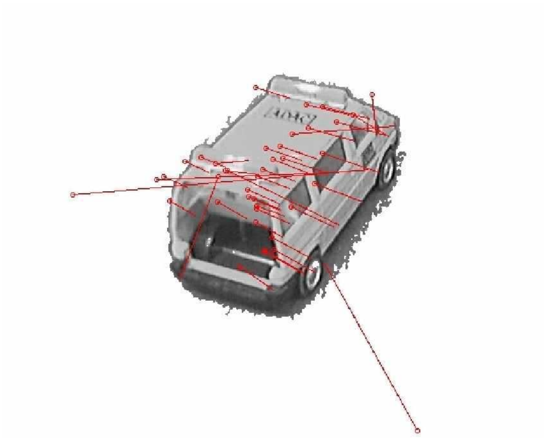
In numbers (the function value corresponding to the squared deviation in pixels), the overall reprojection error of the initial reconstruction that was calculated from 13 views amounted to 1,379,510 in all images and was reduced by the Levenberg-Marquardt algorithm to 938,946. The evolutionary algorithm with 34,408 iterations reduced the error to 627,546 and allowed a second application of the Levenberg-Marquardt algorithm to yield a final error of 204,517.



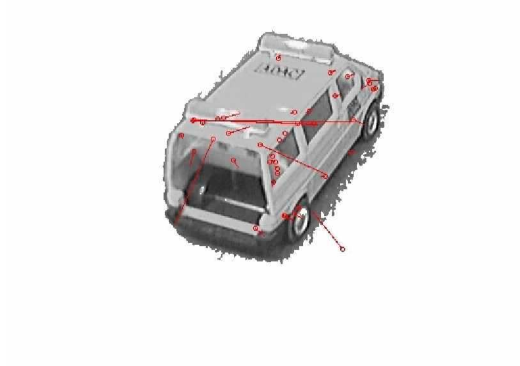
(a)



(b)

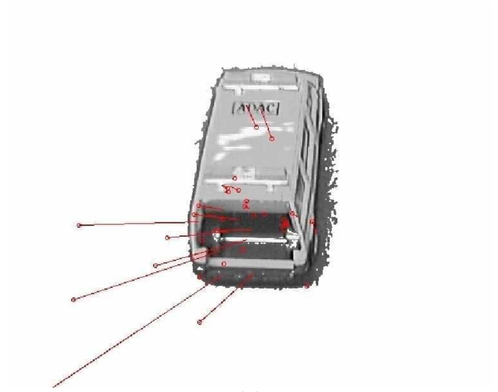


(c)

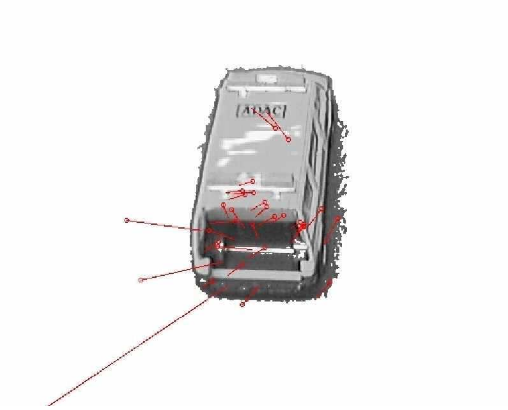


(d)

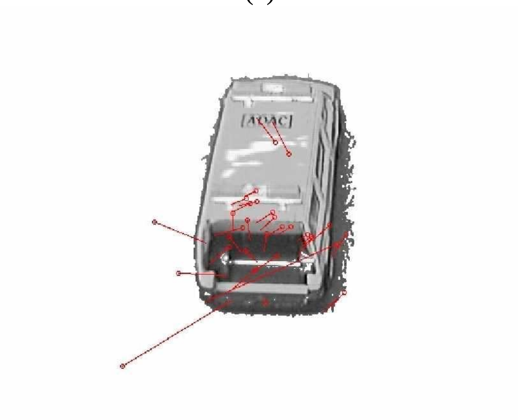
Figure 2: One view of the reconstruction at different stages of optimization



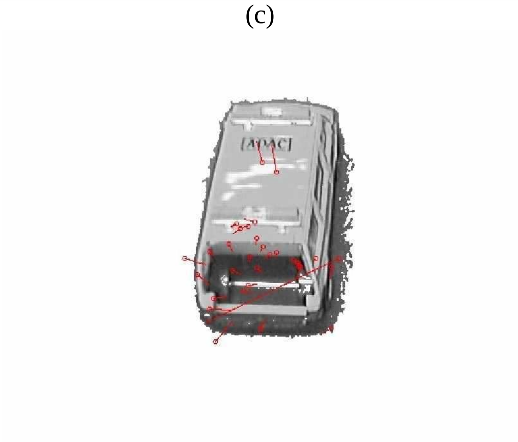
(a)



(b)



(c)



(d)

Figure 3: Another view of the reconstruction at different stages of optimization

5 Conclusions

By presenting an exemplary result we showed how a simple evolutionary algorithm can successfully supplement the optimization process of a 3D-reconstruction that is generally performed by Bundle Adjustment with a Levenberg-Marquardt algorithm.

One could argue that this success can be explained by the deficient initial reconstructions the system delivers, that are far from a global optimum. However, if real-life data is used we always have to deal with these situations.

Without question, the first step to generate an acceptable 3D-reconstruction is to upgrade the system to create better initial reconstructions by applying the trifocal tensor (as recommended in [1], for instance), which is not used in the current system. To receive a metric reconstruction, auto-calibration techniques can be applied (see [2]). These techniques allow for gaining a metric reconstruction without needing extensive further information.

Considering the optimization we suspect that evolutionary algorithms generally can fundamentally contribute. We used a simple evolution strategy only. By using a larger population and recombination an improved optimization behaviour especially in tough situations could be obtained. Of course, this is a broad field still to be investigated.

References

[1] Andrew Fitzgibbon, Andrew Zisserman. Automatic 3D model acquisition and generation of new images from video sequences. *Proceedings of European Signal Processing Conference (EUSIPCO ’98), Rhodes, Greece*, pp 1261–1269, (1998).

[2] Richard Hartley, Andrew Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, second edition. (2004).

[3] Lars Heyden. Inkrementelle Akquisition von 3D-Objektmodellen. Master’s thesis, Universität Dortmund, (2005).

[4] Thomas Jansen. Evolutionäre Algorithmen. http://ls2-www.cs.unidortmund.de/lehre/sommer2003/evoalg/evoAlg-2003.pdf (2003).

[5] Thomas Maurer, Christoph von der Malsburg. Tracking and learning graphs on image sequences of faces. *Proceedings of the ICANN 1996, Bochum*, pp 323–328, (1996).