

Estimation of Projection Matrices from a Sparse Set of Feature Points for 3D Tree Reconstruction from Multiple Images

Štefan Kohek¹, Damjan Strnad¹, Borut Žalik¹, Simon Kolmanič¹

¹University of Maribor, Faculty of Electrical Engineering and Computer Science

Article Info

Article history:

Received May 24th, 2017

Revised Aug 18th, 2017

Accepted Oct 18th, 2017

Keyword:

3D reconstruction
simulated annealing
projection matrix

ABSTRACT

3D reconstruction of trees is an important task for tree analysis but the most affordable approach to capture real objects is with a camera. Although, there already exist methods for 3D reconstruction of trees from multiple photographs, they mostly handle only self-standing trees captured at narrow angles. In fact, dense feature detection and matching is in most cases only the first step of the reconstruction and requires a large set of features and high similarity between individual pictures. However, capturing trees in the orchard is in most cases possible only at wider angles between the individual pictures and with overlapping branches from other trees, which prevents reliable feature matching. We introduce a new approach for estimating projection matrices to produce 3D point clouds of trees from multiple photographs. By manually relating a smaller number of points on images to reference objects, we substitute the missing dense set of features. We assign to each image a projection matrix and minimize the projection error between the images and reference objects using simulated annealing. Thereby, we produce correct projection matrices for further steps in 3D reconstruction. Our approach is tested on a simple application for 3D reconstruction of trees to produce a 3D point cloud. We analyze convergence rates of the optimization and show that the proposed approach can produce feasible projection matrices from a sufficiently large set of feature points. In the future, this approach will be a part of a complete system for tree reconstruction and analysis.

Corresponding Author:

Štefan Kohek,

University of Maribor, Faculty of Electrical Engineering and Computer Science,

Smetanova ulica 17, 2000 Maribor, Slovenia

Email: stefan.kohek@um.si

1. Introduction

Reliable measurement of branches is important for successful examination of trees. However, manual measurement and examination of trees inside orchards is a tedious task, which is prone to errors and also requires extensive workforce. It is more preferable to capture real trees and perform examination on digitized trees inside the office. Therefore, numerous algorithms for automatic 3D reconstruction of trees have been developed. Many of them require special devices for tree capturing, e.g. terrestrial LIDAR [1] and Kinect [2], which are not widely deployed. On the other hand, cameras (e.g., on mobile phones) are the most affordable and widely adopted approach to capture real objects.

Pipelines for 3D reconstruction from images in most cases [3] consist of numerous steps: detecting feature points on a sequence of images, matching feature points between image pairs, camera calibration (i.e., pose estimation), and triangulation [4]. The biggest issue with this approach lies in the heavy dependence on detecting and matching feature points. Most commonly, SIFT [5] or SURF [6] are used for this task, but they require distinct features on images. When trying to perform 3D reconstruction of trees inside the orchards, we cannot easily produce enough distinct feature points because trees might be captured at wider angles, have

repeating patterns, look quite similar and therefore lack distinct features. Overlapping trees in the orchards further aggravate the aforementioned problem.

3D tree reconstruction inside orchards is a problem which cannot be easily solved with the standard 3D reconstruction pipeline. Although numerous approaches for 3D reconstruction of trees from photographs were developed, many of them are designed mostly for convincing visualization or animation of reconstructed 3D models and therefore perform more coarse reconstruction [7], [8], [9], [10]. On the other hand, other more exact approaches [11], [12] have specific requirements, such as clear background, no overlapping trees, or smaller recording angles. Because we cannot adhere to aforementioned requirements, we must use alternative approach for 3D reconstruction. In our case, an approach similar to voxel coloring [13] is used, which performs 3D reconstruction in inverse direction than ordinarily (i.e., with triangulation [3]) and it thus does not require matched feature points. On the other hand, it requires correct camera projection matrices in advance.

Evolutionary algorithms are useful tool for camera calibration when calculating intrinsic parameters [14] or even extrinsic parameters by considering relation between real 3D object with known coordinates and its occurrence on images [15]. However, considering solely relation between 3D object and its occurrence on images is not enough when 3D object covers only part of the image. In this case, the projective error on other parts of the image increases, which prevents correct 3D reconstruction.

In this work, we introduce new approach for estimating camera matrices which considers in addition to few correspondences between points on a known object and its occurrences on multiple images also correspondences between points on multiple images. We use simulated annealing [16] for calculating projection matrices because it requires only few parameters and is quite successful on various applications (e.g., automated tree pruning [17]). We show that consideration of direct relations between images lowers projective error and thus enables 3D reconstruction.

2. Overview of the method

Automatic feature extraction and matching from our photographs is not reliable, therefore we provide alternative data for estimating matrices of images. We can easily provide just a set P of few corresponding points between neighbor images and reference object. Thus before recording, we place a unit cube inside the scene. After taking pictures, we manually match each vertex of the cube with its occurrence on all pictures. For each vertex p of the cube, we thus define its 3D coordinates \mathbf{u}_p inside the scene and its 2D position $\mathbf{v}_{p,i}$ on each picture i . Additional data are provided by manually relating few tips of branches between the images. Therefore, we provide $\mathbf{v}_{p,i}$ for branch tips and thus match each point p on a branch tip between different pictures.

Our approach for estimating projection matrices is designed to consider input data which can be easily provided. To find 3×4 camera projection matrix \mathbf{M}_i of each image i from a set of images I , two-step approach is used. In the first step, we calculate initial estimation of projection matrices, where relations between known object in 3D space and its projection in images are considered. Here, a draft approximation of metric reconstruction is obtained. In the second step, we refine matrices from the previous step by considering also the correspondences between points (tips of branches) on multiple images.

2.1. Initial projection matrix estimation

The objective of the first step is to find for each image i a coarse projection matrix for metric reconstruction. With simulated annealing [16], we calculate projection matrix \mathbf{M}_i of picture i by minimizing for vertices $p \in P$ the distance between projection \mathbf{M}_i of 3D point \mathbf{u}_p and its 2D position $\mathbf{v}_{p,i}$ on the image i . Criteria/energy function of matrix \mathbf{M}_i is:

$$f_1(\mathbf{M}_i) = \frac{\sum_{p \in P} g(\mathbf{M}_i, \mathbf{u}_p, \mathbf{v}_{p,i})}{\|P\|}, \quad (1)$$

where function $g(\mathbf{M}, \mathbf{u}, \mathbf{v})$ calculates distance between point \mathbf{v} and projection \mathbf{M} of point \mathbf{u} (in pixels):

$$g(\mathbf{M}, \mathbf{u}, \mathbf{v}) = \left\| \mathbf{v} - \frac{(\mathbf{M}\mathbf{u})_{xy}}{(\mathbf{M}\mathbf{u})_z} \right\|. \quad (2)$$

The initial state of simulated annealing is for the first image a random matrix which has values in the range of a typical projection matrix:

$$\mathbf{M}_1 = \begin{bmatrix} 1000 r_1 & r_2 & r_3 & 1000 r_4 \\ r_5 & 1000 r_6 & r_7 & 1000 r_8 \\ 1000 r_9 & 1000 r_{10} & 1000 r_{11} & r_{12} \end{bmatrix}. \quad (3)$$

Here, $r_i \in [0, 1]$ is a random number from the uniform distribution.

New candidate projection matrix \mathbf{M}'_i is calculated from \mathbf{M}_i . One random element at row r and column c of matrix \mathbf{M}_i is in each iteration of simulated annealing perturbed according to parameter F and current temperature T :

$$\mathbf{M}'_{i,r,c} = \mathbf{M}_{i,r,c} + T F \mathbf{M}_{i,r,c} (2r - 1). \quad (4)$$

Other elements are perturbed with probability C . If candidate \mathbf{M}'_i is better solution than \mathbf{M}_i , it is accepted as the new state (\mathbf{M}_i), otherwise it is accepted with Metropolis criterion [18].

The best estimated projection matrix of image $i-1$ is used as the initial state of image i in order to speed up the estimation of subsequent matrices.

2.2. Final estimation of matrices

In this step, we refine matrices from the first step by considering also the correspondence between points on multiple images. With simulated annealing from the previous step we produce a set of projection matrices \mathbf{M} . Initial state of simulated annealing is thus a set of matrices from the first step $\mathbf{M} = \{\mathbf{M}_1, \dots, \mathbf{M}_{|\mathbb{I}|}\}$. Criteria function for minimization in the second step is:

$$f_2(\mathbf{M}) = \frac{\sum_{i \in \mathbb{I}} f_1(\mathbf{M}_i)}{2 \|\mathbb{I}\|} + \frac{\sum_{p \in \mathbb{P}} \sum_{i \in \mathbb{I}} g(\mathbf{M}_i, t(p, \mathbf{M}), \mathbf{v}_{p,i})}{2 \|\mathbb{P}\| \|\mathbb{I}\|}. \quad (5)$$

Here, function $t(p, \mathbf{M})$ calculates 3D position from correspondences \mathbf{v}_p of point p by averaging triangulated positions between image-pairs [19].

Unlike the first step, each iteration of simulated annealing changes one random element of randomly chosen matrix. Other elements are perturbed with probability C .

3. Results and discussion

Our approach for estimating projection matrices was tested on a set of 8 orchard images (Fig. 1) which are obtained synthetically from EduApple [21]. For each test we executed simulated annealing 10 times. At first, we estimated coarse projection matrix of the first image. Here, we used next parameters for simulated annealing: $T_0=1$, $F=0.5$, and $C=0$. Fig. 2 displays convergence curves of median, highest, and lowest errors. Results indicate successful estimation as best solution had errors (i.e. average distance of pixels) smaller than 10 pixels.

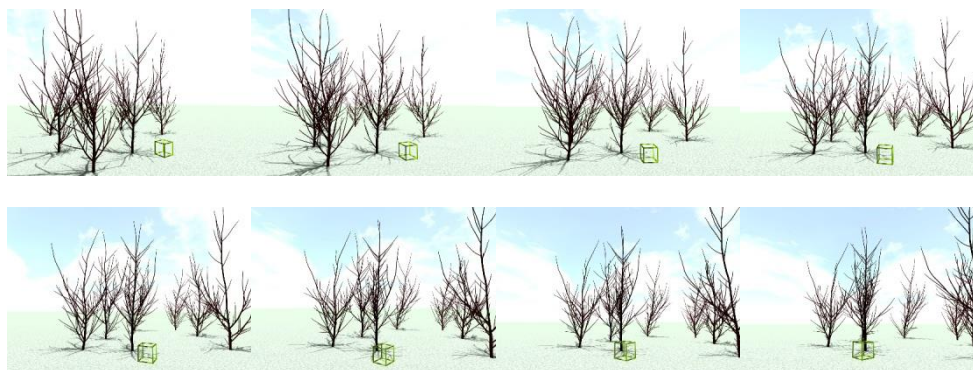


Figure 1. A sequence of images used in 3D reconstruction.



Figure 2. Convergence curves of initial projection matrix estimation (median/best/worst estimations).

The best solution of each estimation of \mathbf{M}_i was used for each subsequent image (Fig. 3). Here, the parameters of simulated annealing were: $T_0=1$, $F=0.1$, and $C=0.5$. For the first image, we again used previous estimation and lowered the error even further. The errors of other images were also significantly lowered from the initial estimation, although initial errors were not so high because projection matrices of consecutive images are already similar to final estimation.

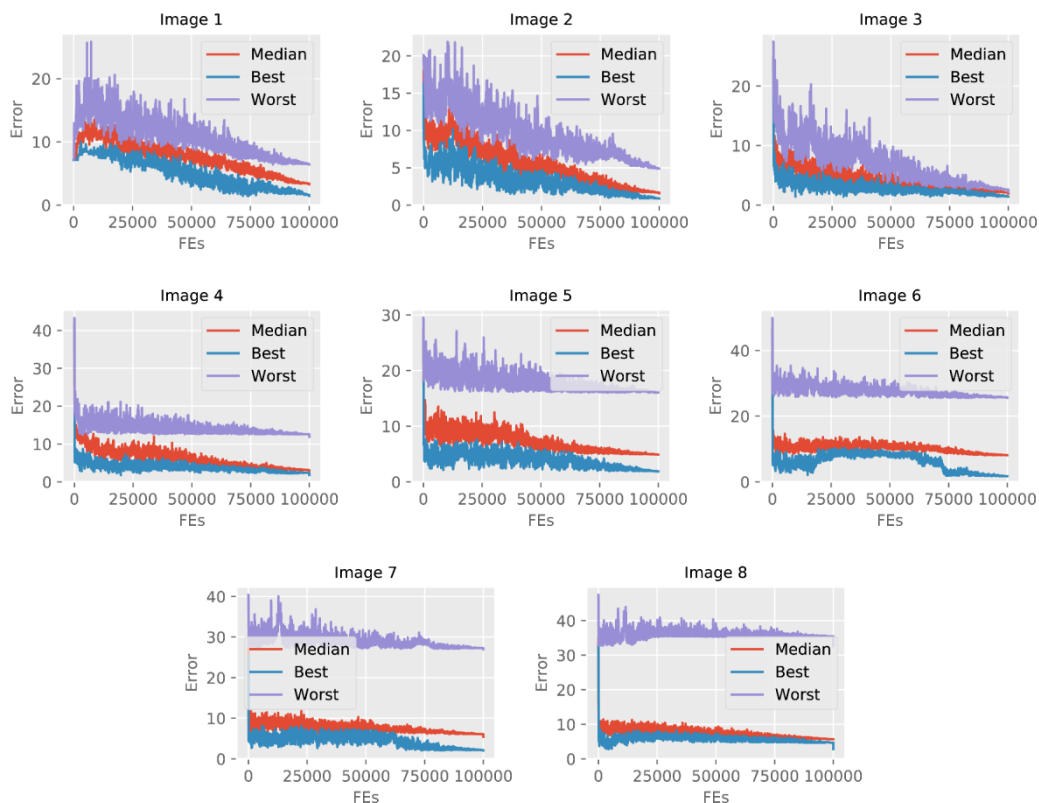


Figure 3. Convergence curves of initial estimation of all projection matrices (median/best/worst estimations).

Fig. 4 shows convergence rates of the second step of matrix estimation. Here, we used next parameters: $F=0.1$, $T_0=0.01$, and $C=0.05$. Lower initial temperature was used because matrices were in most cases already oriented towards the optimal solution. Smaller variations between best and worst estimation confirm the aforementioned fact. The final error was just few pixels, which enables further steps of 3D reconstruction. Here, we used resulting matrices for initial version of 3D reconstruction pipeline, which works similar to voxel coloring [13]. Fig. 5 shows result of successful 3D reconstruction, where produced tree is similar to the tree at the cube (Fig. 1).

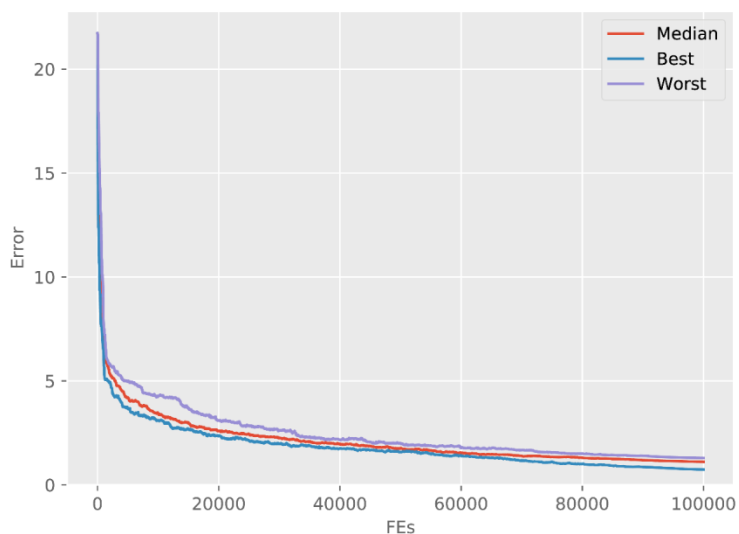


Figure 4. Convergence curves of final optimization of all eight matrices (median/best/worst estimations).

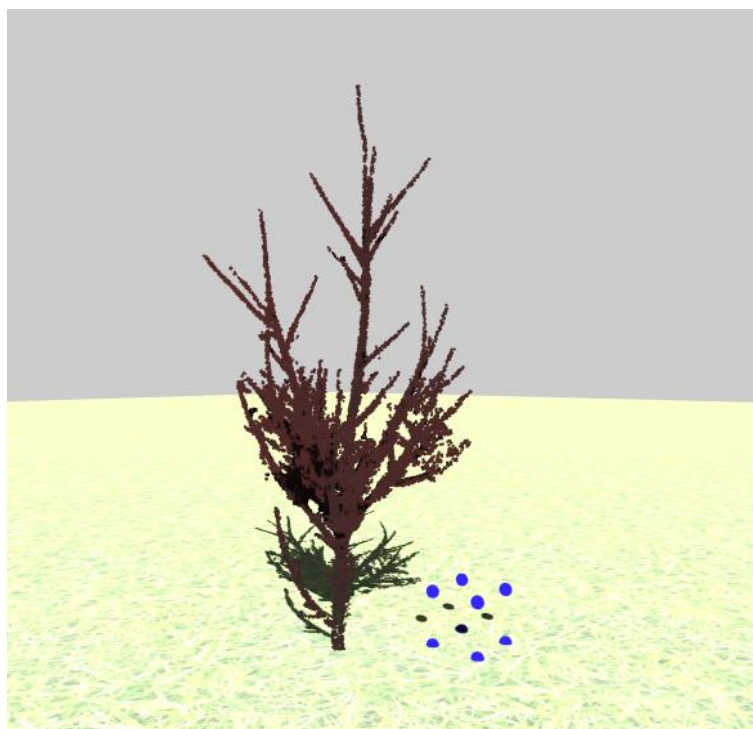


Figure 5. Resulting 3D structure of the reconstructed tree.

4. Conclusions

In this paper, we presented alternative approach for estimating projection matrices and show that two-step approach was able to correctly estimate projection matrices and enable further steps of 3D reconstruction. Simulated annealing supported steady convergence of our optimization problem. Initial estimation of projection matrices lowered complexity of the problem for the second step and thus increased estimation speed of final projection matrices. Finally, we have successfully produced 3D point cloud of simple tree from multiple images inside the orchards. 3D structure was similar to input images and thus appropriate for subsequent analyses.

In the future, we plan to develop a complete framework for tree reconstruction. Matching corresponding branches manually is tedious, therefore we want to automate this task to a greater extent. Simulated annealing in current version requires large number of criteria function evaluations. Therefore, we plan to increase convergence speed of the presented approach by tuning input parameters. Simulated annealing is also only one of the algorithms for global optimization. We plan to verify other algorithms for global optimization, e.g., DE [22] or jDE [23], or even multi-criteria optimization algorithms [24]. Finally, we plan to use our approach for further growth analysis.

Acknowledgements

The authors acknowledge the project J2-6764 and research program P2-0041 were financially supported by the Slovenian Research Agency.

References

- [1] J. R. Rosell, J. Llorens, R. Sanz, J. Arno, M. Ribes-Dasi, J. Masip, A. Escolà, F. Camp, F. Solanelles, F. Gràcia, et al., “Obtaining the Three-Dimensional Structure of Tree Orchards from Remote 2D Terrestrial Lidar Scanning,” *Agricultural and Forest Meteorology*, vol. 149, no. 9, pp. 1505–1515, 2009.
- [2] N. M. Elfiky, S. A. Akbar, J. Sun, J. Park, and A. Kak, “Automation of Dormant Pruning in Specialty Crop Production: An Adaptive Framework for Automatic Reconstruction and Modeling of Apple Trees,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 65–73, 2015.

- [3] C. Li, J. Zheng, C. Dang, and H. Zhou, “A Method of 3D Reconstruction from Image Sequence,” in *Image and Signal Processing, 2009. CISP'09. 2nd International Congress on*, pp. 1–5, IEEE, 2009.
- [4] R. I. Hartley and P. Sturm, “Triangulation,” *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 146–157, 1997.
- [5] D. G. Lowe, “Object Recognition from Local Scale-Invariant Features,” in *Computer Vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2, pp. 1150–1157, Ieee, 1999.
- [6] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-up Robust Features (Surf),” *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [7] P. Tan, G. Zeng, J. Wang, S. B. Kang, and L. Quan, “Image-Based Tree Modeling,” *ACM Transactions on Graphics*, vol. 26, 2007.
- [8] P. Tan, T. Fang, J. Xiao, P. Zhao, and L. Quan, “Single Image Tree Modeling,” *ACM Transactions on Graphics*, vol. 27, pp. 1081–1087, 2008.
- [9] B. Neubert, T. Franken, and O. Deussen, “Approximate Image-Based Tree-Modeling Using Particle Flows,” *ACM Transactions on Graphics*, vol. 26, no. 3, 2007.
- [10] A. Zamuda, J. Brest, B. Bošković, and V. Žumer, “Differential Evolution for Parameterized Procedural Woody Plant Models Reconstruction,” *Applied Soft Computing*, vol. 11, no. 8, pp. 4904–4912, 2011.
- [11] C. Wu, L. He, X. Du, S. Chen, and K. Ni, “3D Reconstruction of Chinese Hickory Tree for Dynamics Analysis,” *Biosystems Engineering*, vol. 119, pp. 69 – 79, 2014.
- [12] D. Zhang, N. Xie, S. Liang, and J. Jia, “3D Tree Skeletonization from Multiple Images Based on PyrLK Optical Flow,” *Pattern Recognition Letters*, vol. 76, pp. 49 – 58, 2016.
- [13] S. M. Seitz and C. R. Dyer, “Photorealistic Scene Reconstruction by Voxel Coloring,” *International Journal of Computer Vision*, vol. 35, no. 2, pp. 151–173, 1999.
- [14] A. Whitehead and G. Roth, “Estimating Intrinsic Camera Parameters from the Fundamental Matrix using an Evolutionary Approach,” *EURASIP Journal on Advances in Signal Processing*, vol. 2004, no. 8, p. 412751, 2004.
- [15] Q. Ji and Y. Zhang, “Camera Calibration with Genetic Algorithms,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 31, no. 2, pp. 120–130, 2001.
- [16] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, et al., “Optimization by Simulated Annealing,” *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [17] D. Strnad and Š. Kohek, “Novel Discrete Differential Evolution Methods for Virtual Tree Pruning Optimization,” *Soft Computing*, pp. 1–13, 2015.
- [18] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of State Calculations by Fast Computing Machines,” *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.
- [19] R. Hartley, R. Gupta, and T. Chang, “Stereo from Uncalibrated Cameras,” in *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pp. 761–764, IEEE, 1992.
- [20] P. C. Schuur, “Classification of Acceptance Criteria for the Simulated Annealing Algorithm,” *Mathematics of Operations Research*, vol. 22, no. 2, pp. 266–275, 1997.
- [21] Š. Kohek, N. Guid, S. Tojnko, T. Unuk, and S. Kolmanič, “Eduapple: Interactive Teaching Tool for Apple Tree Crown Formation,” *HortTechnology*, vol. 25, no. 2, pp. 238–246, 2015.
- [22] R. Storn and K. Price, “Differential Evolution– A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [23] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Žumer, “Self-adapting Control Parameters in Differential Evolution: A comparative Study on Numerical Benchmark Problems,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, pp. 646–657, 2006.
- [24] T. Robič and B. Filipič, “Demo: Differential Evolution for Multiobjective Optimization,” in *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 520–533, Springer, 2005.

