Ransac Based Out-of-Core Point-Cloud Shape Detection for City-Modeling

Ruwen Schnabel, Roland Wahl, Reinhard Klein
Institut für Informatik II, Universität Bonn
Römerstr. 117
53117 Bonn
{schnabel,wahl,rk}@cs.uni-bonn.de

1 Introduction

Due to the increasing size and complexity of geometric data sets there is an ever-growing demand for concise and meaningful abstractions of this data. Especially when dealing with digitized geometry, e.g. acquired with a laser range scanner or reconstructed by stereo algorithms, no handles for modification of the data are available to the user other than the digitized points themselves. However, in order to be able to make use of the data effectively, the raw digitized data has to be enriched with abstractions and possibly semantic information, providing the user with higher-level interaction possibilities. Only such handles can provide the interaction required for involved editing processes, such as deleting, moving or resizing certain parts and hence can make the data more readily usable for modelling purposes. One mean to quickly derive higher levels of abstraction is detecting instances of a set of primitive geometric shapes in the point sampled data. In our previous work [1] we focussed especially on finding an *efficient* algorithm for point-cloud shape detection, in order to be able to deal even with large point-clouds. We presented a high performance RANSAC [2] algorithm that is capable to extract a variety of different types of primitive shapes, while retaining such favourable properties of the RANSAC paradigm as robustness, generality and simplicity. At the heart of this algorithm are a novel, hierarchically structured sampling strategy for candidate shape generation as well as a novel, lazy cost function evaluation scheme, which significantly reduces overall computational cost. The method detects planes, spheres, cylinders, cones and tori, but additional primitives are possible. The algorithm allows for the reliable extraction of these shapes from the data, even under adverse conditions such as heavy noise. An in depth discussion of application areas for this algorithm can also be found in [1] and a follow up report on shape retrieval methods [3]. Unfortunately, the algorithm suffers of two major problems preventing its application to data sets like combined aerial and terrestrial scans of whole cities. Firstly our original algorithm does not support out-of-core data, i.e. for processing the complete data set has to be loaded into main memory. Secondly, by applying the RANSAC paradigm our algorithm extracts the different primitive shapes in a sequential order according to the size of their support, i.e. the number of points belonging to the primitive. An estimate of the probability $P_m(C)$ that no better candidate than primitive shape C was overlooked is used to terminate the sampling process. While the number of samples that have to be drawn to ensure that $P_m(C)$ has a reasonable value, e.g. 99% in our setting, is moderate when decomposing a point cloud of a single model like an oil pump, or even when processing the exterior scan of a whole church, it increases drastically in the case of a high number of equally sized primitive shapes like facades or roofs are contained in a city data set. In this paper we present an out-of-core extension of our previous algorithm that solves both problems. By decomposing the data set into smaller parts that can be loaded into main memory and processed individually, the data set can be handled either in sequential order on a single processor or in parallel on a multi-processor system. Furthermore, at the same time the number of samples needed to determine the best candidate shape remains small leading to an overall performance gain of the extended algorithm.

2 RANSAC for point cloud shape detection

The RANSAC paradigm extracts shapes by randomly drawing minimal sets from the point data and constructing corresponding shape primitives. A minimal set is the smallest number of points required to uniquely define a given type of geometric primitive. The resulting candidate shapes are tested against all points in the data to determine how many of the points are well approximated by the primitive (called the *score* of the shape). After a given number of trials, the shape which approximates the most points is extracted and the algorithm continues on the remaining data. RANSAC exhibits the following, desirable properties:

- It is conceptually simple, which makes it easily extensible and straightforward to implement

Abb. 1: Extract shapes in the point cloud P

- It is very general, allowing its application in a wide range of settings
- It can robustly deal with data containing more than 50\% of outliers [4]

Its major deficiency is the considerable computational demand if no further optimizations are applied. In their pioneering work Bolles and Fischler [2] apply RANSAC to extract cylinders from range data, [5] use RANSAC and the gaussian image to find cylinders in 3D point clouds. Both methods, though, do not consider a larger number of different classes of shape primitives. Roth and Levine [4] describe an algorithm that uses RANSAC to detect a set of different types of simple shapes. However, their method was adjusted to work in the image domain or on range images, and they did not provide the optimization necessary for processing large unstructured 3D data sets. A vast number of extensions to the general RANSAC scheme have been proposed. Among the more recent advances, methods such as MLESAC [6] or MSAC [7] improve the robustness of RANSAC with a modified score function, but do not provide any enhancement in the performance of the algorithm, which is the main focus of our work. Nonetheless the integration of a MLESAC scoring function is among the directions of our future work. Nister [8] proposes an acceleration technique for the case that the number of candidates is fixed in advance. As it is a fundamental property of our setup that an unknown large number of possibly very small shapes has to be detected in huge point-clouds, the amount of necessary candidates cannot, however, be specified in advance.

2.1 Overview

Given a point-cloud $P=\{p_1,...,p_N\}$ with associated normals $\{n_1,...,n_N\}$ the output of our algorithm is a set of primitive shapes $\Psi=\{\psi_1,\ldots,\psi_n\}$ with corresponding disjoint sets of points $P_{\psi_1}\subseteq P,\ldots,P_{\psi_n}\subseteq P$ and a set of remaining points $R=P\setminus\bigcup_{i=1}^{n}P_{\psi_i}$ Similar to [9] and [10], we frame the shape extraction problem as an optimization problem defined by a score function. The overall structure of our method is outlined in pseudo-code in algorithm [Abb.1]. In each iteration of the algorithm, the primitive with maximal score is searched using the RANSAC paradigm. New shape candidates are generated by randomly sampling minimal subsets of *P* using a novel sampling strategy introduced in [1]. The types of candidate shapes are plane, sphere, cylinder, cone and torus. Candidates of *all* these

shape types are generated for every minimal set and each of these candidates is collected in the set C. Thus no special ordering has to be imposed on the detection of different types of shapes. After new candidates have been generated the shape m of highest score is computed employing an efficient lazy score evaluation scheme. The best candidate is only accepted if, given the size $|m|$ (in number of points) of the candidate and the number of drawn candidates $|C|$ the probability $P(|m|,|C|)$ that no better candidate was overlooked during sampling is high enough (see section Probabilities). If a candidate is accepted, the corresponding points P_m are removed from P and the candidates C_m generated with points in P_m are deleted from C. The algorithm terminates as soon as $P(\tau,|C|)$ for a user defined minimal shape size τ is large enough.
In our implementation we use a standard score function that counts the number of compatible points for a shape candidate [2, 6]. The function has two free parameters: ε specifies the maximum distance of a compatible point while α restricts the deviation of a point's normal from that of the shape. We also ensure that only points forming a connected component on the surface are considered.

2.2 Probabilities

The complexity of RANSAC is dominated by two major factors: The number of minimal sets that are drawn and the cost of evaluating the score for every candidate shape. As we desire to extract the shape that achieves the highest possible score, the number of candidates that have to be considered is governed by the probability that the best possible shape is indeed detected, i.e. that a minimal set is drawn that defines this shape.
Consider a point cloud P of size N and a shape ψ therein consisting of n points. Let k denote the size of a minimal set required to define a shape candidate. If we assume that any k points of the shape will lead to an appropriate candidate shape then the probability of detecting ψ in a single pass is:

$$P(n) = \binom{n}{k} \Big/ \binom{N}{k} \approx \left(\frac{n}{N}\right)^k \tag{1}$$

The probability of a successful detection $P(n,s)$ only after s candidates have been drawn equals the complementary of s consecutive failures:

$$P(n,s) = 1 - (1 - P(n))^s \tag{2}$$

Solving for s tells us the number of candidates T required to detect shapes of size n with a probability $P(n,T) = p_t$

$$T \geq \frac{\ln(1-p_t)}{\ln(1-P(n))} \quad (3)$$

For small $P(n)$ the logarithm in the denominator can be approximated by its Taylor series $\ln(1-P(n)) = -P(n) + O\left(P(n)^2\right)$ so that:

$$T \approx \frac{-\ln(1-p_t)}{P(n)} \quad (4)$$

Given the cost *Co* of evaluating the cost function, the asymptotic complexity of the RANSAC approach is $O(T \cdot Co) = O\left(\frac{1}{P(n)} Co\right)$.

For huge city models this inverse polynomial dependency of the asymptotic run-time complexity on the probability *P(n)* for detecting a certain shape becomes a bottleneck. In this situation we have to deal with thousands of individual buildings most of them of the same size. Let us consider for example an aerial scan with a spatial resolution of 7cm of the city of Munich roughly containing 200.000 individual buildings and roughly being spread over about 300 km^2. In this case the total raw data size consists of $61 \cdot 10^9$ points. For simplicity let us further assume that each building consists of only 6 individual faces with a total surface area of approximately 1000m^2 resulting in about 32000 points per primitive shape. To find one of the individual faces with a probability of 99% with the standard RANSAC approach we would have to draw

different minimal sets from the point-cloud which is prohibitively large. In our previous method this was already improved using local sampling which in this case would lead to

necessary candidates. This is still too much especially since for each of the resulting individual shapes the score function must be evaluated. One way to solve this problem is to adapt the size of the primitive to be extracted to the total number of points in the model, i.e. to keep the ration between *n* and *N* bounded, which is discussed in detail in the following section.

3 Out-of-Core

Since buildings typically do not exceed a maximal side length, we can safely apply our original RANSAC scheme to local parts of the input point-cloud. Such local parts have to be large enough so that no structures in the data can be missed. Moreover, to be effective, these parts should be chosen such that the ratio $\frac{1}{P(n)}$ is bounded locally. Since arbitrarily small shapes can appear in general, the local parts have to be hierarchical. To construct such local parts of the input point-cloud, we propose to sort the point-cloud into a global, out-of-core octree data structure, but other hierarchies might be suitable as well.

In principle, the shape detection is then executed for each of the cells in the octree. Of course, some additional care has to be taken to ensure sufficient overlap between cells. Also the octree should be constructed in a way such that the side lengths of the cells correspond to the expected maximal side length of structures in the point data.

3.1 The octree data structure

The side length of the largest cells is computed based on a maximal area *A* and a maximal aspect ratio *R* of the surfaces contained in the data. These parameters need to be specified by the user. The side length *S* of the cells is then given as $S = \sqrt{RA}$. The bounding cube of the octree is chosen such that on some level *L* the side length of the contained cells is exactly *S*. In order to achieve overlap between cells, when the shape detection is executed for cell *C*, the points in neighbouring cells on level $L+1$ are included during the shape detection as well (these have side length $\frac{1}{2}S$). The minimal size of a shape is set to $\frac{1}{4}A$ during the detection, as smaller shapes will be searched on the next finer level in the octree. This way the ratio $\frac{1}{P(n)}$ is effectively bounded (since the side length of a cell and the minimal size of a shape are coupled via the hierarchy) and the data can be processed out-of-core due to the local nature of the algorithm.

Thus, once the detection has been executed on level L, we continue in the same manner on level $L+1$ until all shapes have been detected.

4 Results and conclusion

We tested our modified algorithm on a part of the Graz dataset kindly provided by Heiko Hirschmüller from the DLR. The first subset we considered contains about 15 million sample points generated using the stereo reconstruction algorithm from aerial photographs described in [11]. We restricted the maximum area of a shape to 1000 m^2 and the aspect ratio of an individual shape to 1:10. We extracted 1120 shapes. About 3.5 million points, mainly belonging to other geometry like natural cover, were not assigned. The total extraction time was about 259 sec. For this medium sized data set we were able to compare this to our original algorithm which delivered a similar result with respect to the number of shapes and remaining points: 1000 shapes with 3.8 million remaining points were extracted in about 900 sec. Enhancing the area considered to a subset of about 25 million points of the Graz dataset and using our modified algorithm we extracted about 2000 shapes with 5.7 million remaining points in about 470 sec, which is an approximately linear increase of runtime. The parameters were again a maximum area of 1000 m^2 per shape and an aspect ratio of 1:10. A comparison with the original algorithm was not possible in this case due to the lack of main memory and the resulting disk-swapping. In summary we can state that partitioning the raw data set in parts that are adapted to the size of the sought primitives, enables the design of an efficient out-of-core RANSAC method for point-cloud shape detection that is especially well suited for huge city models. A probability analysis shows that by adapting the size of the local subsets of the global point-cloud to the existent primitive shapes, the run-time complexity can be kept approximately linear in the overall number of points in the model, a speedup that is also suggested by the experiments.

Literatur

1. R. Schnabel, R. Wahl, R. Klein: Efficient RANSAC for Point-Cloud Shape Detection. in *Computer Graphics Forum*, Vol. 26, No. 2, Blackwell Publishing, June 2007, pages 214-226
2. Fischler M.A., Bolles R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM 24* , 6 (1981), 381--395.
3. R. Schnabel, R. Wahl, R. Wessel, R. Klein: Shape Recognition in 3D Point Clouds, Technical report, No. CG-2007-1, ISSN 1610-8892, May 2007
4. Fitzgibbon A.~W., Eggert D.~W., Fisher R.~B. : High-level CAD model acquisition from range images. *Computer-aided Design 29* , 4 (1997), 321--330.

5. Chaperon T., Goulette F.: Extracting cylinders in full 3-d data using a random sampling method and the gaussian image. In *VMV01* (2001), pp. 35--42.
6. Torr P., Zisserman A. : Robust computation and parametrization of multiple view relations. In *ICCV '98: Proceedings of the Sixth International Conferenceon Computer Vision* (Washington, DC, USA, 1998), IEEE Computer Society, p. 727.
7. Torr P., Zisserman A.: MLESAC : A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding 78* (2000), 138--156.
8. Myatt D., Torr P., Nasuto S., Bishop J., Craddock R. : Napsac: High noise, high dimensional robust estimation - it's in the bag. In *BMVC* (2002), Computer Vision Tools.
9. Nistér D. : Preemptive {RANSAC for live structure and motion estimation. *Mach. Vision Appl. 16* , 5 (2005), 321--329.
10. Décoret X., Durand F., Sillion F., Dorsey J. : Billboard clouds for extreme model simplification. In *Proceedings of the ACM Siggraph* (2003), ACM Press
11. Hirschmüller H., Stereo Processing by Semi-Global Matching and Mutual Information, accepted for publication in *IEEE Transactions on Pattern Analysis and Machine Intelligence*.