Robust Estimators for Geodetic Transformations and GIS

Dr. sc. techn. Entela Kanani

Zurich, 2000

This publication is an edited version of:

DISSERTATION ETH No. 13521

Robust Estimators for Geodetic Transformations and GIS

A dissertation submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of

Doctor of Technical Sciences

presented by

Entela Kanani

Geodetic Engineer (with honors), University of Tirana

born on 01 December, 1967

from Albania

accepted on the recommendation of

Prof. Dr. Alessandro Carosio, examiner
Prof. Dr. Alfio Marazzi, co-examiner

Zurich, 2000

IGP Mitteilung Nr. 70
Robust Esimators for Geodetic Transformations and GIS — Entela Kanani

Copyright © 2000

Institut für Geodäsie und Photogrammetrie
Eidg. Technische Hochschule Zürich
ETH Hönggerberg
8093 Zürich

Alle Rechte vorbehalten

ISBN 3-906467-26-0
ISSN 0252-9335

Vorwort

Die Ausgleichungsrechnung in ihren Formen ist heute ein Werkzeug, das zunehmend Beachtung in den unterschiedlichsten Disziplinen erlangt. In der Geodäsie ist sie aber bereits seit 200 Jahren das Verfahren für die Parameterschätzung. Die Grundidee von Gauss und Legendre hat sich während zwei Jahrzehnten als optimal erwiesen.

Erst Ende der sechziger Jahre wurden die Geodäten auf neuere Modelle der Parameterschätzung aufmerksam gemacht. Die Mathematiker (besonders bekannt sind die Arbeiten von P. Huber in Berkley, Zürich und Harward) hatten die Bedeutung der Robusten Statistik erkannt. Die Möglichkeit, gute Resultate zu erreichen – auch wenn ein Anteil grober Messfehler darin enthalten ist – war für die Geodäsie von grossem Interesse.

Erste Versuche in den achtziger Jahren – vor allem in der Schweiz und in Dänemark – waren erfolgreich und sicherten der Schweiz einen bis jetzt aufrechterhaltenen Vorsprung. An unserem Institut ist besonders die Arbeit von F. Wicki bemerkenswert, welcher die Robuste Ausgleichung in der Triangulation als Standardverfahren in der Schweiz durchsetzte.

Frau Kanani übernahm 1995 das Folgeprojekt „Robuste Schätzer mit hohem Bruchpunkt für geodätische Transformationen", das vom Nationalfonds finanziert wurde. Seit zwei Jahren stehen die experimentellen Teile zur Verfügung und jetzt ist auch die vollständige wissenschaftliche Dokumentation abgeschlossen.

Das Ziel von Frau Kanani war, die Robuste Statistik nutzbringend in zwei Bereichen unserer Aktivität einzusetzen:

- Die Parameterschätzung in linearen Transformationen
- Die automatische Interpretation von kartographischen Bildern

Die vorliegende Publikation, die in Zusammenarbeit mit der UNI-Lausanne entstanden ist, beweist in eindrücklicher Art, dass Robuste Ausgleichungsverfahren einen signifikanten Qualitäts- und Effizienzgewinn in allen Applikationen ermöglicht, wo Modellunsicherheiten vorliegen.

Die vorliegende wissenschaftliche Publikation beinhaltet eine generelle Beschreibung des mathematischen Modells der Robusten Ausgleichung mit den unterschiedlichen

Schätzerkategorien. Besondere Aufmerksamkeit ist den Schätzern mit hohem Bruchpunkt und den Schätzern mit begrenzten Einflussfunktionen gewidmet.

Die Anwendungen der Robusten Verfahren in den geodätischen linearen Transformationen werden dann bis ins ausführliche Detail behandelt. Es wird nachgewiesen, dass sowohl Schätzer mit hohem Bruchpunkt als auch diejenigen mit begrenzten Einflussfunktionen je nach Applikation notwendig sind. Als typische Beispiele wurden der BIBER-Schätzer gewählt, der sich in der Triangulation bereits bewährte und der LMS-Schätzer (mit hohem Bruchpunkt). Die professionelle Software TRINETH, die daraus entstanden ist, ermöglicht, die Methoden in der Praxis einzusetzen. Die durchgeführten Vergleiche mit im Ausland untersuchten Modellen zeigten die überdurchschnittliche Qualität des Modells von TRINETH.

Als zweite Applikation hat die Autorin die Anwendung der Robusten Verfahren für die automatische Interpretation bestehender topographischer Karten, insbesondere für die Vektorisierung von kartographisch dargestellten Bauten gewählt. Die Besonderheit des realisierten Verfahrens ist seine Anwendbarkeit, auch wo gewisse geometrische Eigenschaften nur teilweise zutreffen. Die Auswertung (Vektorisierung) lässt sich so modellieren, dass auch unscharfe Eigenschaften (z.B. „oft rechtwinkling" oder „in vielen Fällen rechteckig" usw.) berücksichtigt werden können. Die Software, die Frau Kanani entwickelt hat (RobVec), wird erfolgreich für die wirtschaftliche Produktion von GIS-Daten sowohl bei staatlichen Stellen als auch bei privaten Unternehmungen eingesetzt.

Die vorliegende Publikation ist klar und übersichtlich geschrieben, sie ist sorgfältig verfasst und dokumentiert. Entela Kanani hat die Arbeit auf Grund sorgfältiger Recherchen in den Grundlagen erarbeitet. Sie hat aber die Bedürfnisse der Praxis nicht ausser Acht gelassen und dadurch Marktanforderungen erfüllt. Ihre Leistung wurde als Promotionsarbeit der ETH angenommen und die Autorin mit dem Titel Dr. sc. techn. ausgezeichnet.

Ich danke und gratuliere Frau Entela Kanani für den erfolgreichen Abschluss dieser bemerkenswerten Arbeit und allen anderen Beteiligten – insbesondere Prof. Dr. Alfio Marazzi von der Universität Lausanne – danke ich für die wirksame Unterstützung.

Zürich, 4. Mai 2000 Prof. Dr. Alessandro Carosio

Contents

Abstract 1

Zusammenfassung 2

1. Introduction 3

1.1 Motivation 3

1.2 Research Objectives 6

2. Estimation of Parameters in Geodesy 7

2.1 Introduction 7

2.2 The Adjustment Model in Geodesy 9
2.2.1 The Stochastic Model 9
2.2.2 The Functional Model 9
2.2.3 The Gauss – Markov Model 10
2.2.4 The Least Square Method 11

2.3 The Robust Adjustment Model in Geodesy 14
2.3.1 The Stochastic Model of the Robust Adjustment 15
2.3.2 Outliers in Geodesy 15
2.3.3 The Functional Model of the Robust Adjustment 18

3. Classes of Robust Methods 19

3.1 The Huber-Type Estimator 20

3.2 Bounded Influence Estimates 22
3.2.1 The Mallows-Type Estimator 22
3.2.2 The Schweppe-Type Estimator 23

3.3 High Breakdown Point Estimates 24
3.3.1 The Breakdown Point – Mathematical Definition 24
3.3.2 The Least Median of Squares Estimate 25
3.3.3 The Least Trimmed Square Estimate 26

3.3.4 The S-Estimators 26

4. Robust Estimators for Geodetic Transformations 27

4.1 Linear Transformations – Introduction 27
4.1.1 Helmert Transformation – Mathematical Model 28
4.1.2 Affine Transformation – Mathematical Model 30

4.2 Linear Transformations in Geodesy 31

4.3 Determination of the Unknown Parameters by Least Square Method 34
4.3.1 Helmert Transformation 34
4.3.2 Affine Transformation 37
4.3.3 Outliers in the Transformation Model 39

4.4 Determination of the Unknown Parameters by Robust Estimators 39
4.4.1 The Transformation Program TRINETH 40
4.4.2 LMS - estimator 41
Definition 41
Method of computation 41
The choice of the observation equations in the determination of the unknown parameters 44
Calculation procedure 45
Bounding the influence of outliers using the LMS estimator 49
Estimation of s_0 (LMS - estimator) 49

4.4.3 BIBER - estimator 50
Definition 50
Calculation procedure 52
Bounding the influence of outliers using the BIBER - estimator 56
Estimation of s_0 (BIBER - estimator) 56
Calculation of the matrix of the cofactors of the unknown parameters 57

4.4.4 Implementation in the Software TRINETH 59
User input 59
Linearisation of the observation equation and determination of the approximated values 60
Adjustment process 61
Numerical problems 61
Important indicators and listing 63

4.5 Applications and Results 64

4.5.1 Example 1: Simulated Deformations in Lago Bianco 64
LMS - estimator 64
BIBER - estimator 66
Huber - estimator 67

4.5.2 Example 2: Simulated Deformations in Lago Bianco 70
LMS - estimator 71
BIBER - estimator 72
Huber - estimator 72

5. Robust Estimators in the Automatic Structuring of Raster Data 75

5.1 Introduction 75
5.2 Topographic Map 76
5.2.1 The Content of the Map 76
5.2.2 Map Accuracy 79

5.3 From Analogue Copy to Numerical Data 79
5.3.1 Manual Vectorisation 80
5.3.2 Automatic Vectorisation 81
5.3.3 Combined Solution 82

5.4 Vectorisation of Topographic Features 83
5.4.1 Vectorising Lines 83
5.4.2 Vectorising Solid Shapes 83

5.5 Pattern Recognition based Vectorisation 84
5.5.1 Pattern Recognition Techniques – Overview 84
5.5.2 Extraction of Areal Objects – Method and Results 85

5.6 Vectorisation – Contour Extraction 88

5.7 Simplification and Smoothing of Digitised Lines 89

5.7.1 Simplification of Digitised Lines 90
Douglas and Peucker algorithm adapted in a closed contour 91
Determination of the tolerance parameters at the Douglas – Peucker algorithm 92

5.7.2 Smoothing of Digitised Lines 95

5.8 Adjustment of the Approximated Contour – New Approach 97

5.8.1 Mathematical Model in the Adjustment of Areal Objects 98
Observations and unknown parameters 98
Equations of the model 99
Outliers and model characteristics 101
The stochastic model 102

5.8.2 Computation Procedure 103
Linearisation of the observation equations 103

Estimation of the unknown parameters 103

5.8.3 Implementation of the Algorithms in RobVec 105

5.8.4 Applications and Results 110
Statistical results 110
Current projects 112

6. Summary and Conclusions 121

References 123

Acknowledgements 133

Abstract

Most parameters in geodesy are determined by statistical estimator procedures. The method most commonly employed is that of Least Squares - an optimal estimator for normal distributed measurements. In practical evaluation, however, there often occur many observations which do not conform to the expected normal distribution. This occurs, for example, when a part of the observations contain outliers. With the help of robust statistics, it is possible to solve such problems, since they are not sensitive to outliers.

This work concerns the evaluation of data using robust methods, particularly in the fields of linear transformation (geodesy) and vectorisation of areal objects from scanned topographic maps for Geographic Information Systems (GIS). Both the theoretical basis and the practical implementation are described.

Besides the classical method of data evaluation, the Least Square, it is now possible to determine the transformation parameters at the geodetic transformation, by two robust estimators: the LMS estimator (from the class of the high breakdown point estimators) and the BIBER estimator (from the class of bounded influence estimators). It is shown that the LMS estimator, as a very good outlier detector, can be used when data is highly contaminated (up to 50%); whereas the BIBER estimator is more appropriate in instances where the number of outliers is small, and the assumption of the normal distribution can still be accepted. A combination of both estimators gives optimal results, even when the data are contaminated (up to 50%).

A novel application in the field of GIS and cartography is demonstrated, involving the employment of robust techniques for the automatic vectorisation of buildings extracted from topographic and general maps. This application was motivated by the growing need for the automatisation of data acquisition processes in GIS, which are otherwise expensive, and laborious. An appropriate adjustment model, which incorporates robust techniques (BIBER estimator) for the evaluation of data, delivers excellent results, in a short period of time, using a fully automatic process.

The developed algorithms are implemented in prototype software and have shown to be good tools for the evaluation of data in these two fields.

Zusammenfassung

In geodätischen Wissenschaften werden in der Regel stochastische Verfahren verwendet um gesuchte Parameter zu schätzen. Das am heftigsten gebrauchte Verfahren ist die Ausgleichung nach der Methode der kleinsten Quadrate, die für normalverteilte Ausgangsdaten optimale Ergebnisse liefert. In vielen Fällen werden wir aber mit Daten konfrontiert, bei denen die Annahme der Normalverteilung nicht vollständig zutrifft, so zum Beispiel, wenn ein Teil der Beobachtungen verfälscht ist (Ausreisser). Solche Probleme lassen sich mit Hilfe robuster Schätzverfahren lösen, die auf Messfehler viel weniger empfindlich reagieren als die klassischen Methoden.

Diese Arbeit befasst sich mit der Ausgleichung von Daten mittels robuster Schätzverfahren, insbesondere mit linearen Transformationen in der Geodäsie und der Vektorisierung flächiger Objekte aus gescannten topographischen Karten für GIS. In dieser Arbeit werden sowohl die theoretischen Untersuchungen als auch verschiedene Realisierungen beschrieben. Im ersten Teil werden, neben der klassischen Ausgleichungsmethode, der Methode der kleinsten Quadrate, zwei robuste Schätzverfahren für die Bestimmung der Transformationsparameter vorgestellt: Der LMS-Schätzer (aus der Klasse der Schätzer mit hohem Bruchpunkt) und der BIBER-Schätzer (aus der Klasse der Schätzer mit begrenztem Einfluss der Verbesserungen). Es wird gezeigt, dass der LMS-Schätzer sich besonders für Datensätze mit vielen Ausreissern (bis 50%) und schlechter Messanordnung eignet. Der BIBER-Schätzer wiederum ist besser geeignet, wenn die Anzahl der Ausreisser klein ist und die Annahme einer Normalverteilung der Daten immer noch akzeptiert werden kann. Eine Kombination beider Schätzer hingegen liefert optimale Resultate, selbst bei vielen Ausreissern.

Im zweiten Teil, wird eine neuartige Applikation auf dem Gebiet von GIS und Kartographie aufgezeigt, welche die Techniken robuster Schätzverfahren für die automatische Vektorisierung von Gebäuden verwendet, die aus topographischen Karten und Übersichtsplänen extrahiert werden. Diese Applikation soll eine Lösung für das wachsende Bedürfnis nach einer Automatisierung des arbeits- und kostenintensiven Datenerfassungsprozesses für GIS aufzeigen. Ein geeignetes Ausgleichungsmodell, welches die entwickelten robusten Ausgleichsverfahren verwendet, liefert in einem vollautomatischen Prozess nach kurzer Zeit hervorragende Ergebnisse.

Die entwickelten Algorithmen wurden in zwei Softwareprototypen implementiert und haben sehr gute Resultate bei der Auswertung von Daten in den oben erwähnten Gebieten gezeigt.

Chapter 1

Introduction

1.1 Motivation

Most parameters in geodesy are determined by statistical estimator procedures. These procedures determine the requested values through both a multivaried random realisation (sequence of measurements) and a functional model. The method most commonly employed is that of Least Squares - an optimal estimator for normal distributed measurements. In practical evaluation, however, there often occur many observations which do not conform to the expected normal distribution. The classical compensation method of Least Squares is extremely sensitive and quickly leads to unusable results when significant deviations from the normal distribution are present. Two pathways leading to the solution of this problem can be followed:

- purging the data set before the process of evaluation
- using algorithms which are not sensitive to a significant deviation from the normal distribution provided that these errors do not exceed a given fraction of the entire data set

The first method is the classical procedure. It deals with the definition of an outliers' criteria on the basis of the accepted pattern conception (test of the standardised residuals, F-test and so on). Observations surpassing the significance level of the test, being considered outliers, will be rejected [Pope, 1976; Pelzer, 1979]. The sensitive analysis of the set up test, led to the development of the reliability theory [Barda, 1968; Carosio, 1983].

The second alternative approach is currently of considerable interest and algorithms for this purpose are being developed by many groups working in the field of statistics. Statistical procedures that are resistant to outliers and stable with respect to deviations from the given distributional model are termed robust. P.J. Huber and F. Hampel undertook pioneering work in the 70s, through which the robust statistic first became known. The high breakdown points estimators [Rousseeuw and Leroy, 1987] could be considered a kind of third generation in robustness theory, coming after minimax variance [Huber, 1964] and the influence function [Hampel, 1974]. They not only remain insensitive to large residuals and measurements disposition, but also give good results when a relatively large number of outliers is included in the observations.

Up to now, research directed towards the applications of robust estimators in geodesy, have mostly been conducted either in employing the robust estimators in the outlier detection, or in the determination of unknown parameters (e.g. in the adjustment of geodetic networks, alignment models, photogrammetric adjustment models etc.). Almost all publications in these fields demonstrate the influence of robust procedures based on M-estimators. The robust estimators are used to detect outliers in the observations [Benning, 1978, 1995; Werner, 1984; Caspary and Borutta, 1986; Kampmann, 1986] and the determination of the unknown parameters is accomplished by the Least Square estimator. In the adjustment of geodetic networks, alignment models and photogrammetric adjustment models, robust estimators are used for the determination of unknown parameters [Wicki, 1992, 1999; Fuchs, 1980; Meissl, 1980; Klein and Förstner, 1984].

Robust estimators can also be employed in linear geodetic transformation (e.g. deformation measurements) where the geodetical task involves the transformation of coordinates from one reference system to the other. The robust estimators are employed to determine the transformation parameters of the transformation model in order to compare repeatedly determined positions of reference points [Carosio, 1982; Caspary *et*

al., 1983; Hahn and Bill, 1984; Somogyj and Zavoti, 1990; Kampmann, 1994]. In solving this task, less attention has been paid to the high breakdown estimators and their implementation. Although this group of estimators is generally associated with a large amount of calculation time, their implementation, e.g. for the analysis of deformation measurements, can be of considerable value. The number of parameters to be estimated in a linear transformation is limited (4, 6, 7 and so on), which restricts the calculation time to manageable proportions. Preliminary work undertaken at ETH[1] Zurich has been very promising. It has been shown that inserting a robust estimator with a high breakdown point, to replace or complete the compensation process in the determination of transformation parameters in the Helmert Transformation, has noteworthy advantages [Danuser, 1992]. The geodetic task at the linear transformation involves not only the identification of outliers, but also the delivery of reliable results comparable to the LS estimator. In some cases, the amount of outliers could be considerable and it is often difficult to say in advance where the gross errors should be, in other cases the number of outliers can be small but the measurement disposition is not very suitable. It is worthwhile, therefore, to investigate which estimators are suitable for the parameter determination, in different circumstances.

Automatisation of processes is also currently a topic of considerable importance since every day, thousands of data are processed, very quickly, by computers. As a very fast and fully automated process, the task is often performed with little or no editing of the data; errors might therefore creep in unnoticed. Errors can be intuitively recognised only in special cases but not when complex data have to be adjusted or processed.

To our knowledge, no research has been directed towards extending robust estimations to other fields related to geodesy. One such field is the data acquisition in GIS and cartography, which is normally very expensive and laborious. The vectorisation of areal objects, i.e. raster to vector conversion of extracted cartographic solid shape symbols from topographic maps, will be our topic of interest. Because we are concerned with an automated process, the robust techniques could be of great benefit in the evaluation process. If the robust estimators are not used, a considerable amount of work has to be performed after the process is complete or else the precision will be very poor.

[1] **E**idgenösiche **T**echnische **H**ochschule

1.2 Research Objectives

Our research aims to investigate and develop new concepts of robust estimators for two geodetic applications:

- *linear geodetic transformations (Helmert and affine);*
- *automatic vectorisation of areal objects (e.g. buildings) from scanned topographic maps.*

Linear transformations are used in Geodesy to compare repeatedly determined positions of reference points. Coordinates previously determined have to be adjusted to the reference system at a later time. The resulting differences between previous and present coordinates are the parameters to be interpreted. The aim is to provide a significant improvement in the speed of the network's evaluation and guarantee an efficient analysing tool for deformations measurements, comparing model variants, monitoring and others.

With **Automatic Vectorisation,** the goal is to obtain structured objects (e.g. buildings) from topographic maps in raster format facilitated by robust estimation functions in an automated process. The implemented algorithms should provide a high rate of automatisation making possible the use of this information for several GIS applications such as environmental planning, 3D-modelling or spatial analysis of different themes.

This work addresses the following aspects:

- establishment of a systematic overview of robust estimators and their characteristics
- identification of modern estimators which could or should be used in a given geodetic application considering their characteristics
- investigation and development of new concepts and algorithms suitable for linear geodetic transformation and raster to vector conversion
- design and implementation of the chosen procedures in prototype software
- flexibility of the implemented solution through parameterisation in order to cover many applications
- testing the quality of the obtained estimators by applying them to real data

Chapter 2

Estimation of Parameters in Geodesy

2.1 Introduction

When we talk about "*geodesy*" and "*adjustment*" we automatically think about observations, estimation of unknown parameters and the Least Squares method. This method is commonly regarded as the one and only estimation method in the determination of unknown parameters, after Gauss proved that when the errors of the observations obey the normal distribution, the Least Squares method gives the best estimation for the unknown parameters. Its basic idea is to optimise the fit by minimising the square of the residuals r_i ($\sum \mathbf{r}_i^2 = \mathbf{min}$). The absence of computers at that time and the simple way of computation, made LS very popular. Furthermore, the statistical characteristics of the required parameters (estimated observations, unknown parameters or residuals) can be described very efficiently. Since then LS has become the basic technique of fitting data. Later it was realised that real data usually do not conform to the classical assumption [Box 1953; Tukey, 1962] and that Least Squares is not

always the optimal estimator. Therefore, the necessity for new ways and methods to model data has become an important topic in the field of statistics.

There are two fundamentally different approaches to model data which do not conform to a normal distribution (see Fig. 2.1). The first is to construct the so called *outlier diagnostics* which attempt to spot influential observations, which are corrected or deleted. The modified data set can then be refitted using the standard Least Squares method. The other option is to employ the *robust technique* which adopts estimators that are not, or only partly, affected by outliers. Outlier diagnostics and robust techniques both have the same goal, but these approaches are distinct in that they go about the task in the opposite order. The outlier diagnostics first attempt to identify the outliers and then fit the data with a LS whereas a robust method first fits a regression that satisfies the majority of the data, and then identifies the outliers of those observations which posses large residuals (displayed in bold) from the robust solution.

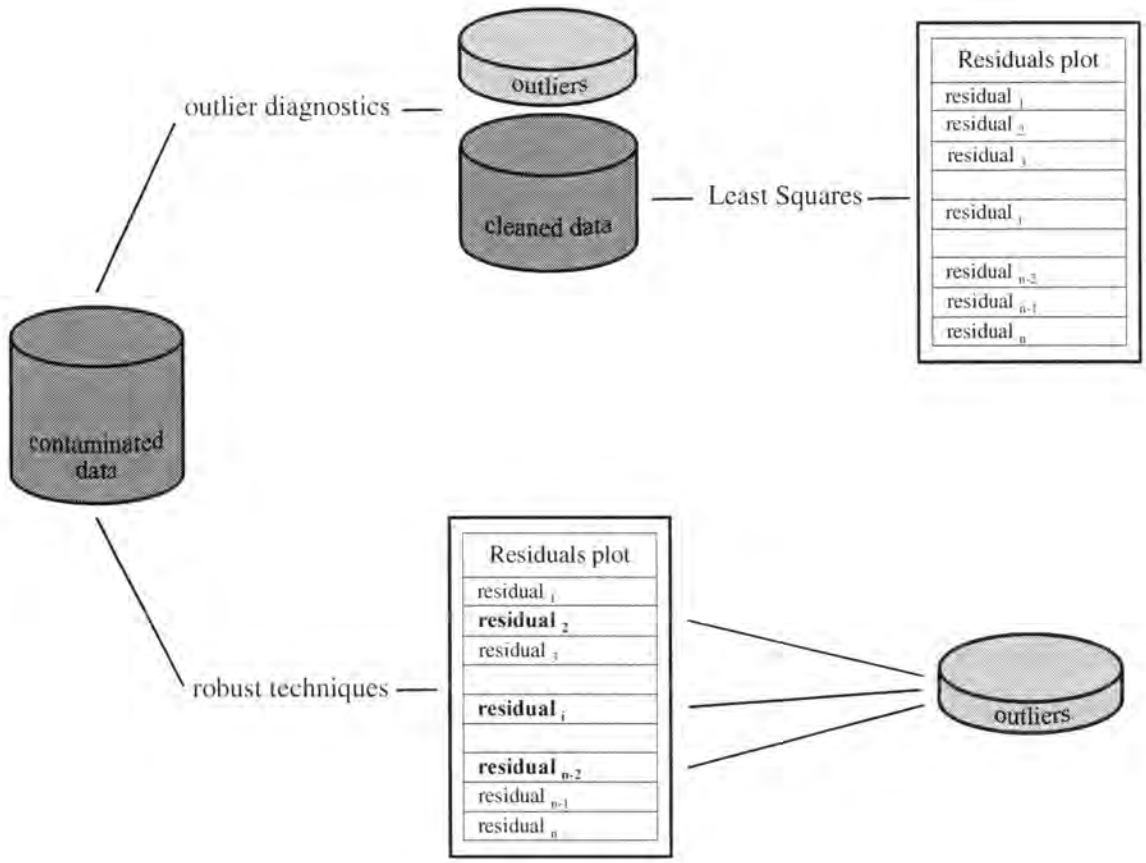


Fig. 2.1 Two ways of fitting data: outlier diagnostics and robust techniques

Outlier diagnostics are well known and commonly employed in the statistical analysis. A detailed description can be found in Rosseeuw and Leroy [1987] and Heck [1981]. In this work only a survey of *robust techniques* will be presented.

2.2 The Adjustment Model in Geodesy

2.2.1 The Stochastic Model

The stochastic part of the mathematical model in geodesy describes the characteristics or properties of the random variables (observations $l_1, l_2, \ldots, l_i, \ldots, l_n$). The assumption is that the observations obey the normal distribution $N(\underline{l}, \Sigma_{ll})$.

Its presentation in the simplest way can be given through the covariance matrix of the observations Σ_{ll} as per 2.1. The σ_i^2 represent the variance of the ith observation while the σ_{ij} represent the covariances between the observations $l_1, l_2, \ldots, l_i, \ldots, l_n$.

$$\Sigma_{ll} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \ldots & \sigma_{1j} & \ldots & \sigma_{1n} \\ & \sigma_2^2 & \ldots & \sigma_{2j} & \ldots & \sigma_{2n} \\ & & \ldots & \ldots & \ldots & \ldots \\ & & & \sigma_i^2 & \ldots & \sigma_{in} \\ & & & & \ldots & \ldots \\ & & & & & \sigma_n^2 \end{bmatrix} \tag{2.1}$$

The standard deviations σ_i may be estimated from experience, whereas the covariance values σ_{ij} ($i \neq j$) are taken as zero assuming that the correlation between observations is zero.

2.2.2 The Functional Model

The functional model in the adjustment model most commonly used, expresses the relation between the expected values of the observations and the unknown parameters that have to be estimated; every observation is given as a function of the unknown parameters:

$$\underline{l}_i = F_i(x_i, \ldots, x_p) \qquad \text{for } i = 1, \ldots, p \tag{2.2}$$

where $\underline{l}_i$ is usually interpreted as the real value of the ith observation.

2.2.3 The Gauss-Markov Model

In geodesy, the function that describes the relation between unknown parameters and observations is normally non linear. Therefore, models that substitute such a function, by a linear one, have been developed. The *Gauss-Markov* adjustment model, regarded in geodesy as the standard model, is given by the following linear equation,

$$\underline{l}_i = a_{i1}x_1 + \ldots + a_{ip}x_p \ , \tag{2.3}$$

where $\{a_{i1}, \ldots, a_{ip}\}$ represents a row of the design matrix *A*; $\underline{l}_i$ the real value of the ith observation; $\{x_1, \ldots, x_p\}$ the vector of the unknown parameters.

The stochastic model of the Gauss-Markov model is given by,

$$l + \underline{e} = \underline{l} \ , \tag{2.4}$$

where l is the vector of the observations and $\underline{e}$ the vector of the errors. From 2.3 and 2.4 can be written,

$$\underline{e} = Ax - l \ . \tag{2.5}$$

The following equivalent matrices are important in the Gauss-Markov model

$$\Sigma_{ll} = E[ee^T]; \qquad Q_{ll} = \frac{1}{\sigma_o^2}\Sigma_{ll}; \qquad W = Q_{ll}^{-1} \ . \tag{2.6}$$

The errors normally obey the normal distribution with,

$$\underline{e} \approx N(0, \sigma_o^2 Q_{ll}) = N(0, \Sigma_{ll}) \ . \tag{2.7}$$

In the equations above the following symbols mean:

$\underline{l}$	vector of the real values of the observations l_i
$\bar{l}$	vector of the observations l_i
$\underline{e}$	vector of the errors e_i
A	design matrix
x	vector of unknown parameters
Σ_{ll}	covariance matrix of the observations
Q_{ll}	matrix of the cofactors of the observations
W	matrix of weight of the observations
σ_o^2	variance when the weight is equal to one

When the weight matrix W is known, it is always possible to transform the model given in 2.5 and 2.7 into a homogeneous model. In this case the weight matrix W is equal to the identity matrix I. By that transformation, the unknown parameters x remain unchanged. The transformation steps are not given in this work. They can be found in Wicki [1999].

After this transformation, the covariance matrix of the observation is given by,

$$\Sigma_{ll} = \sigma_o^2 I \ . \tag{2.8}$$

In order to keep the calculation and presentation of all equations and formulae in the following chapters simple we assume that the above mentioned transformation has been applied.

2.2.4 The Least Square Method

In reality the observations l_i do not fullfil the functional model, therefore, estimated values $\bar{l}_i$ should be determined. The so-called, residuals, should be added to the observed variables in order to obtain the expected values of the observations,

$$\bar{l}_i = l_i + r_i \ . \tag{2.9}$$

In order to determine the r_i, and so the $\bar{l}_i$, an estimator is needed. The assumption is that the measurements obey the normal distribution and are independent from each other. The classical adjustment model is that where the function to be minimised is the squared residuals, and is known as the *Least Square Method*. The estimated model is given by,

$$r = Ax - l \tag{2.10}$$

and the matrix of cofactors of the observations gives the stochastic model,

$$Q_{ll} = \frac{1}{\sigma_o^2}\Sigma_{ll} = I \ . \tag{2.11}$$

Therefore, for a single observation the residual is given by,

$$r_i = \sum_{j=1}^{p}(a_{ij}x_j) - l_i, \tag{2.12}$$

and the diagonal element of the matrix of cofactors of the observations is,

$$(q_{ll})_{ii} = \frac{1}{\sigma_o^2}(\Sigma_{ll})_{ii} = 1 \ . \tag{2.13}$$

The Least Squares estimate is obtained as a solution to the problem,

$$\sum r_i^2 = \min . \tag{2.14}$$

We can obtain the normal equation by differentiating 2.10 with respect to the unknown parameters x and equating the partial derivatives to zero;

$$A^T Ax = A^T l \ . \tag{2.15}$$

The unknown parameters x_i are given by,

$$x = (A^T A)^{-1} A^T l . \tag{2.16}$$

The matrix of the cofactors of the unknown parameters is determined by applying the law of the propagation of errors and is given by,

$$Q_{xx} = (A^T A)^{-1}, \tag{2.17}$$

and s_o is estimated by,

$$s_o^2 = \frac{r^T r}{n-p} \tag{2.18}$$

The matrix *H.* given through equation 2.19, is called *the hat matrix:*

$$H = A(A^T A)^{-1} A^T . \tag{2.19}$$

Through the law of the propagation of errors and equations 2.10 and 2.16 it is possible to calculate the matrix of the cofactors of the residuals which is given in 2.20. The steps of the calculation can be found in Carosio [1996],

$$Q_{rr} = Q_{ll} - H = I - H . \tag{2.20}$$

Another important variable that warrants attention is z_i, known as the *local reliability* or *partial redundancy* of an observation. The z_i is mathematically expressed [Carosio, 1990] by,

$$z_i = \frac{(q_{rr})_{ii}}{(q_{ll})_{ii}} , \tag{2.21}$$

where $(q_{rr})_{ii}$ is the ith diagonal element of the matrix of cofactors of the residuals and $(q_{ll})_{ii}$ is the same element of the matrix of cofactors of the observations. The sum of the local reliability coefficients z_i gives the redundancy of the system and is given by,

$$\sum_{i=1}^{n} z_i = n - p , \tag{2.22}$$

The redundancy of the system could also be given by,

$$\sum_{i=1}^{n} (1 - h_{ii}) = n - p , \tag{2.23}$$

from which the *local reliability* z_i of an observation, can be derived,

$$z_i = (q_{rr})_{ii} = 1 - h_{ii} , \tag{2.24}$$

where the h_{ii} is the diagonal element of the *"hat matrix"* H.

Another important variable that can be calculated for each observation is the standardised residual. For the ith observation this can be calculated by dividing the residual r_i by the standard deviation of the residual σ_{r_i}.

$$r^*_i = \frac{r_i}{\sigma_{r_i}} = \frac{r_i}{\sigma_o \sqrt{(q_{rr})_{ii}}} \tag{2.25}$$

2.3 The Robust Adjustment Model in Geodesy

In geodesy, and in the evaluation of data in general, it is generally accepted that the observations obey the normal distribution. This is true only if the observations are free from systematic and gross errors. Intensive efforts are made by surveyors to deliver good data (precise data which conform to normal distributed measurements). This includes employing rigorous measuring methods, which limit the correlation between observations, continuously checking of measuring protocols, etc. Nevertheless, deviations from the normal distribution are often found and are explained with the existence of gross errors or *outliers*, as they are known in the theory of statistics. Whereas in geodesy we are looking for gross errors in the observations, in the multiple regression model they represent outliers in the response variables. These gross errors distort the assumption model and lead to distributions different from the normal. In order to cope with these gross errors, the robust estimators can be employed in the evaluation process. Caspary [1996] has formulated the qualities or characteristics that a robust estimator should achieve. The three most important are:

- A robust estimator should have a high breakdown point (see 3.3.1). This means that the estimator should be able to cope with gross errors provided that these errors do not exceed a certain amount of the data set given in advance.

- The robust estimator should have a continuos and bounded influence function, *i.e.* the gross errors should not influence the estimator in comparison to the rest of the data set [Hampel *et al.*, 1986], see also Fig. 3.2 (influence function of Huber estimator).

- The estimator should achieve a high asymptotic efficiency. In other words the dispersion of the estimator around the expected value should be small.

The question of which and when of the above mentioned characteristics is more important than the others, depends on the requirements where the estimator is going to be implemented, the degree of contamination of the data, the number of unknown

parameters to be estimated, etc. For example, for adjustment of geodetic networks, where both, the number of observations and the number of unknown parameters to be estimated can vary from around ten to several hundred, the M-estimators are quite suitable; whereas when the contamination of data is large the breakdown point of the estimators employed for the determination of the parameters is more important. Still, it is very important to note that all these three characteristics are very much interrelated.

2.3.1 The Stochastic Model of the Robust Adjustment Model

The assumption that is the most widely assumed in Geodesy is that the errors of the measurements obey the normal distribution, $N(0,\sigma)$. When gross errors are present, we assume that the distribution that they obey is roughly classified into two groups:

(a) most of the errors obey a normal distribution Φ
(b) a part of the errors (gross errors) obey a distribution H.

Therefore, the stochastic model for the standardised errors can be represented by,

$$F(x) = (1-\varepsilon)\Phi(x) + \varepsilon H(x)\,, \tag{2.26}$$

where ε is the small probability of the existence of gross errors and Φ represents the standard normal distribution.

2.3.2 Outliers in Geodesy

Outliers or gross errors in geodesy can be classified into two groups: (a) outliers in the observations and (b) outliers in the coefficients of the A-matrix, which are also known as *leverage points*.

(a) Outliers in the observations can arise as result of misplacement of decimal points, incorrect targeting, systematic errors of the instruments, erroneous data transmission, etc.

(b) Leverage points or distant outliers in geodesy can be defined as observations which, because of their geometry (which is portrayed in the coefficients of the A-matrix), have values far removed from the rest of the data set and therefore have a large influence on the estimation of the parameters. It is easy to see that such an outlier is very difficult to detect.

From the mathematical point of view, *leverage points* can be viewed as observations with small partial redundancy. This is explained below:

The vector of residuals is given through the following equation,

$$r = -(I - H)l \quad , \tag{2.27}$$

and for a single observation the residual is equal to,

$$r_i = (1 - h_{ii})l_i + \sum h_{ij} l_j \ . \tag{2.28}$$

The *partial redundancy* or *local reliability* of an observation is given in 2.24. From this equation, it is very easy to see that if the diagonal element of the "*hat matrix*", h_{ii}, is close to one, and therefore z_i close to zero, then a single outlier in the observation l_i (equation 2.28) is not going to be detected. Therefore, outliers in the observations with small partial redundancy can greatly influence the estimated parameter. These kinds of observations are then called *leverage points*. At the same time it is important to note that the contrary is not true, meaning that a leverage point should always have a large h_{ii}. If there is only one leverage point, then the h_{ii} is large and the "*hat matrix*" is useful, but if other points are in its vicinity (see equation 2.28) the h_{ii} are smaller and the outliers do not show up through the "*hat matrix*" [Rousseeuw and Leroy, 1987].

Below is shown a simple example to illustrate the two types of outliers **in the simple regression** (p=2). The classical linear model in the simple regression is given by,

$$l_i = a_i x_1 + x_2 + e_i \ , \tag{2.29}$$

in which the slope x_1 and the intercept x_2 are to be estimated. To visualise the data in the simple regression model we can make a plot of the (a_i, l_i). Often this is called a *scatterplot*. After applying a Least Square fit we may face two kinds of outliers: one

kind of outliers are gross errors occurring in the observations, which are called *outliers in the observation space* or *outliers in the y-direction* (Fig. 2.2).

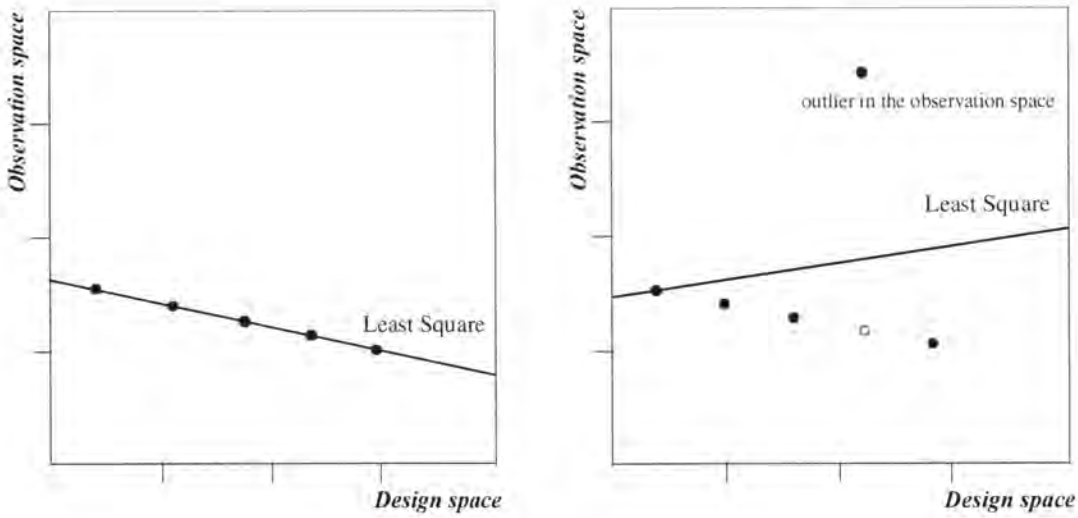


Fig. 2.2 (a,b) Least Squares fit without outliers, Least Squares fit with outliers in the observation space [Rousseeuw and Leroy, 1987]

The other kind are observations that lie far away from the data set (distant outliers). These are called *leverage points* in the design space [Huber, 1981]. Rousseeuw and Leroy [1987] have shown with many examples that an ordinary LS does not show the distant outliers (Fig. 2.3b).

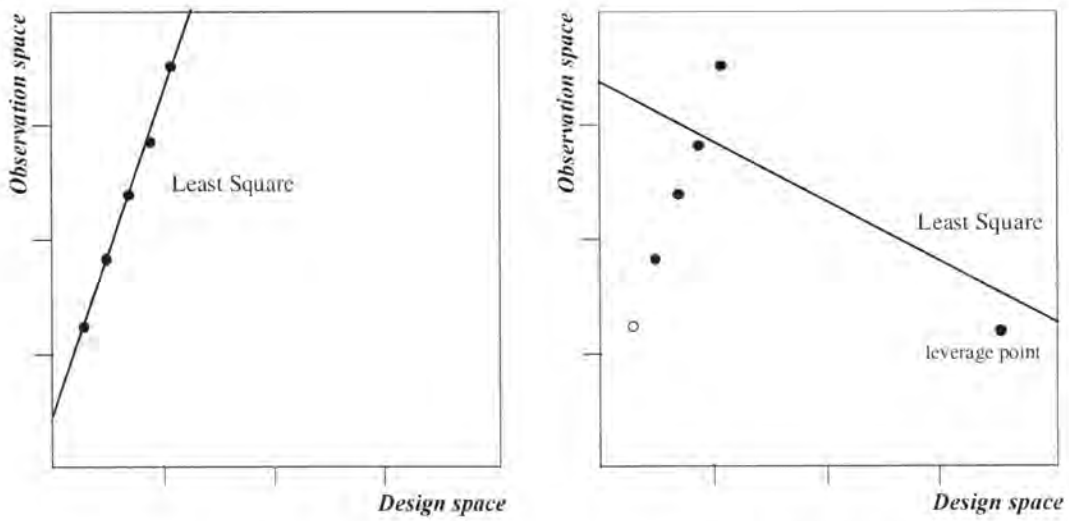


Fig. 2.3 (a,b) Least Squares fit without outliers , Least Squares fit with outliers in the design space [Rousseeuw and Leroy, 1987]

2.3.3 The Functional Model of the Robust Adjustment

A vector of estimators

$$\overline{l}_i = g_i(l_1, l_2, ..., l_n) \quad \text{for } i = 1, ..., m \tag{2.30}$$

is robust when it delivers good estimated values (see also 2.3) even when the observations or the data do not obey the normal distribution.

In chapter 3 classes of robust estimators shall be presented. The form of the robust estimators, employed for the determination of the transformation parameters in linear geodetic transformations, and raster to vector conversion, shall be given in detail in the chapters 4 and 5.

Chapter 3

Classes of Robust Methods

There are almost as many robust estimators as there are outlier diagnostics and it is very important to measure their effectiveness in order to differentiate between them. The main purpose of robust procedures is to give good results in the presence of outliers. In order to achieve this stability robust procedures bound the influence of outliers in various ways according to the type of the problem for which they have been designed. Marazzi [1993] has classified the robust methods into the following groups:

- Problems with outliers in the y-direction (see Fig. 2.2) The most commonly robust procedures for solving this type of problems are based on the Huber-type estimator.

- Problems with a moderate percentage of outliers in the coordinate space (see Fig. 2.3) also known as *levarage points*. The estimators employed for solving this type of problem are the Mallows and Schweppe–type estimators. These types of estimators are also known as *bounded influence estimators*.

- Problems when the frequency of outliers in both y and x direction is high. The estimators employed in this case are the high breakdown estimators.

We shall be concerned with their underlying mathematical principals and their effectiveness in finding outliers. The major references go to Huber, [1981] and Hampel *et al.* [1986] and for the high breakdown point estimators to Rousseeuw and Leroy, [1987].

3.1 The Huber-Type Estimator

Let $A = (a_{i,j})$ be a real n x p matrix, $l = (l_1, ... l_n)^T$ a given vector of observations and $x = (x_1, ..., x_p)^T$ an unknown vector of parameters that has to be estimated. We shall denote by $r = Ax - l$ the vector of residuals for a given value of x and a_i^T the ith row of the matrix A. Classically, the problem of estimating the unknown parameters x_i is solved by minimising the sum of squares $\sum_i (\sum_j a_{ij} x_j - l_i)^2$.

Huber proposes to minimise:

$$\sum \rho \left(\frac{\sum_{i=1}^{n} \rho (\sum_{j=1}^{p} a_{ij} x_j) - l_i)}{\sigma} \right), \tag{3.1}$$

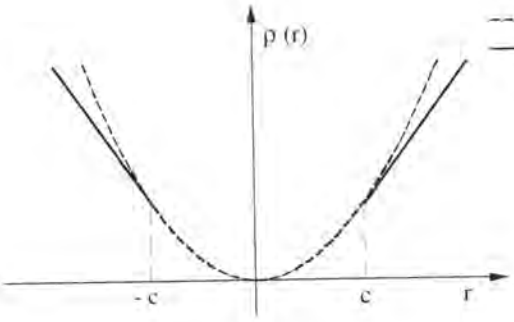
where ρ is a convex function of the residuals that increases less rapidly than the squared function and σ the standard deviation of the observations. By taking partial derivatives of this function with respect to $x_1, \ldots, x_p$ and defining $\psi = \rho'$ (the influence function) we obtain the following system of equations for x_1 to x_p:

$$\sum_{i=1}^{n} \psi \left(\frac{r_i}{\sigma} \right) a_{i,j} = 0 , \qquad \text{for } j = 1, ..., p. \tag{3.2}$$

The solution of this equation is called an *M-estimate* of x_1 to x_p *of Huber type* where the function ρ is an arbitrary function that must be chosen. Huber proposes

$$\rho(r) = \begin{cases} \frac{1}{2} r^2 & \text{for } |r| < c \\ c|r| - \frac{1}{2} c^2 & \text{for } |r| \geq c \end{cases} \tag{3.3}$$

where c is called the *tuning constant* and its value depends on the data dispersion. This function is identical with the Least Squares objective function in the interval (-c, c) and outside this interval is a linear function. The estimator defined by function ρ, ψ and equation (3.3) is called the *Huber estimator*. As $c \to \infty$ the Huber estimator is the classical Least Square estimator. The ρ and ψ functions of Least Square- and Huber-estimator are displayed respectively in figures 3.1 and 3.2.



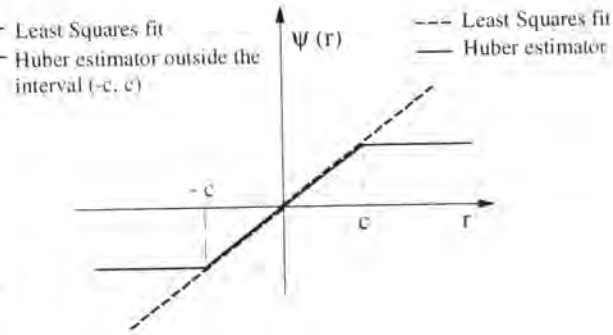


Fig. 3.1 Objective function of Least Square and Huber estimator

Fig. 3.2 Influence function of Least Square and Huber estimator

We can obtain the normal equation by differentiating (3.3) with respect to the unknown parameters and equating the partial derivatives to zero. The system of normal equations consists of three parts corresponding to three different cases of the function 3.3, $r \leq -c$, $-c < r < c$ and $r \geq c$. It is not known from the beginning in which of the above intervals the residual r_i is to be found. Therefore, a possible solution of (3.3) is computed numerically with the help of general-purpose iterative algorithms [Marazzi, 1993]. The first value for the iterative process could be a Least Squares fit even the result is not satisfactory. The procedure continues "by pulling the outliers towards their fitted values, refitting iteratively until convergence is obtained" [Huber, 1981]. This process is displayed in Fig. 3.3 (a,b,c,d) for simple regression.

Assuming that $\sigma = 1$ the Huber estimator is defined as the solution of the vector equation $\sum \psi(r_i) a_i = 0$. Interpreting this computation as a procedure for cleaning the data as in the figure 3.3 (a,b,c,d) the Huber estimator bounds the influence of the outliers in the y-direction but does not take into account their position.

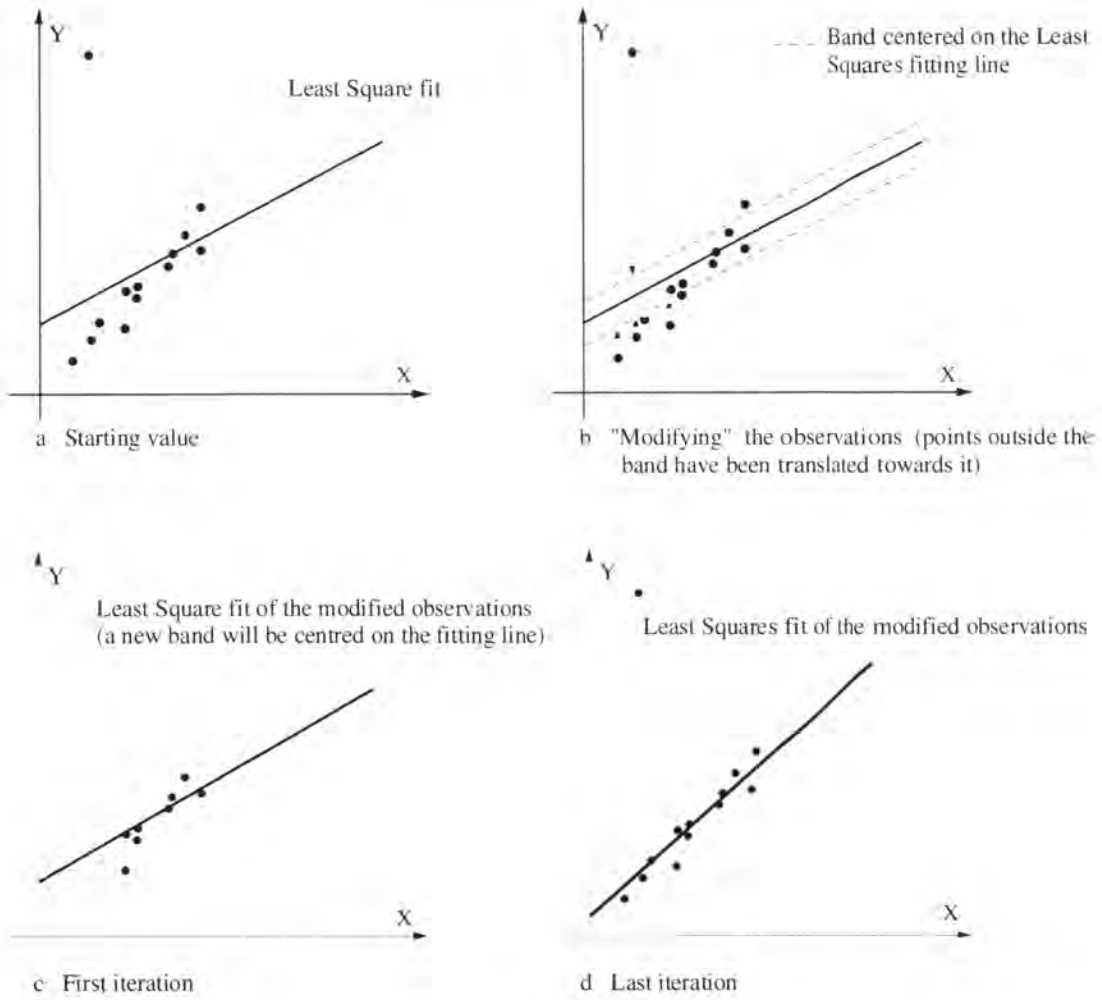


Fig. 3.3 (a,b,c,d) Computation of Huber estimator regression line [Marazzi, 1993]

3.2 Bounded Influence Estimates

In order to bound the influence of outliers in the design space (influence of position) the so called *bounded influence estimates* were introduced. The Mallows and Schweppe type estimators belong to this class of estimators.

3.2.1 The Mallows-Type Estimator

The Mallows-type estimator is defined as a solution of,

$$\sum_{i=1}^{n} \psi\left(\frac{r_i}{\sigma}\right) u_i a_{i,j} = 0 \qquad \text{for } j = 1, \ldots p, \tag{3.4}$$

where u_i are positive weights. Assuming that $\sigma = 1$, equation (3.4) becomes $\sum_{i=1}^{n} \psi(r_i)\, u_i a_{i,j} = 0$. The weights $u_i = u(a_i)$ depend on a measure of the magnitude of the vector a_i (e.g. its distance from the centre of gravity of the data set). Thus the Mallows - estimator bounds the influence of position (Fig. 3.4) without considering whether the residual is large or small, which can imply a loss of efficiency.

3.2.2 The Schweppe –Type Estimator

A Schweppe-type estimate is defined as the solution of the system,

$$\sum_{i=1}^{n} \psi\left(\frac{r_i}{\sigma w_i}\right) u_i a_{i,j} = 0. \tag{3.5}$$

For $\sigma = 1$ the Schweppe-type estimator is $\sum_{i=1}^{n} \psi\left(\frac{r_i}{w_i}\right) u_i a_{i,j} = 0$. This estimator bounds the influence of leverage points only if the corresponding residual is large (Fig. 3.5).

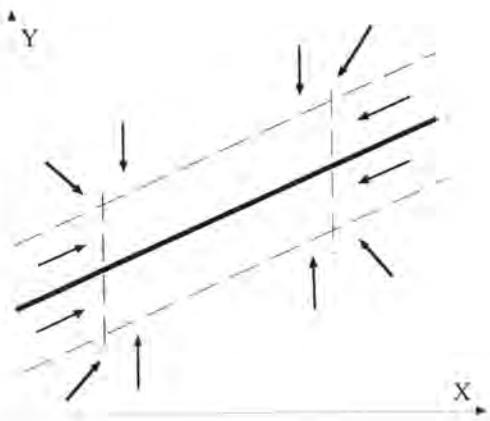


Fig. 3.4 Bounding the influence of outliers (Mallows estimator)

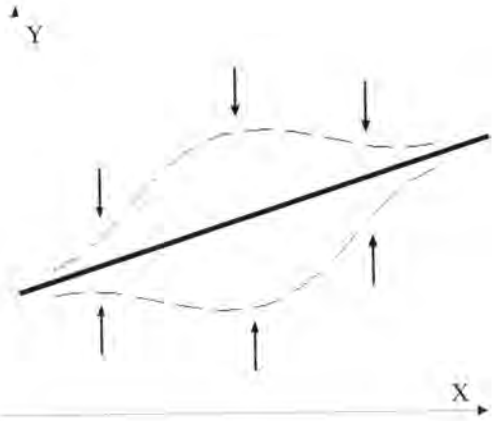


Fig. 3.5 Bounding the influence of outliers (Schweppe estimator)

3.3 High Breakdown Point Estimates

3.3.1 The Breakdown Point - Mathematical Definition

The goal of robust estimators is to deal with contaminated data. In section 2.3.2 we demonstrated that a single distant outlier could totally ruin a Least Squares estimator. We then introduced the Huber-type estimator that can find outliers only in the y direction, and bounded influence estimates that can deal with data containing a certain percentage of outliers. In order to formalise this aspect, we now introduce the *breakdown point*.

The breakdown point has been formally defined by Hodges [1967] and Hampel [1971]. The finite sample breakdown point was first introduced by Donoho and Huber [1983] and has been subject to attention from others [Rousseeuw, 1983; Bloomfield and Stieger, 1983; Rousseeuw and Leroy, 1987].

The mathematical description begins by applying a regression estimator T to a data set Z to obtain the solution of the sample $(x_1, \ldots, x_p)$. If any m original data points are replaced with arbitrary values we obtain a contaminated data set Z'. We define the bias of the estimate T as a function of

$$\text{bias}(m/n; T, Z) = \sup_{Z'} |T(Z') - T(Z)| \tag{3.6}$$

It is always possible that Z' has an infinite bias, meaning that m outliers have an arbitrary large effect on the regression estimator. In this instance the estimator is said to "break down". The breakdown point ε_n^* for the sample Z is defined as

$$\varepsilon_n^* (T, Z) = \min (m/n; \text{ when } \text{bias}(m/n; T,Z) = \text{infinite}) \tag{3.7}$$

In simpler terms, *the breakdown point indicates what fraction of observations in the sample can be regarded as bad values without causing the value of the estimate to change greatly.*

Together with the definition of the breakdown point it is important to know if a theoretical maximum exists for which the estimator takes arbitrary values. Rousseeuw and Leroy [1987] have proved that the breakdown point of an estimator complies with the condition,

$$\varepsilon_n^*(T,Z) \le ([(n-p)/2]+1)/n. \tag{3.8}$$

The breakdown point for the sample **mean** is $\varepsilon^* = 1/n$, *i.e.* one gross error and the bias of the estimator approaches infinity. If the data set is very large its breakdown point tends to zero. Therefore, it can be said that the arithmetic mean has a breakdown point of 0%.

By contrast, a simple resistant estimate is the **median** of the observations

$$\begin{aligned} \mathrm{med}(l_i) &= l_{[(n+1)/2]} && \text{for } n \text{ odd} \\ &= (l_{[n/2]} + l_{[n/2+1]})/2 && \text{for } n \text{ even} \end{aligned} \tag{3.9}$$

where $l_{[1]} \le l_{[2]} \le \ldots \le l_{[n]}$ are the ordered observations. It is obvious that the sample median can tolerate gross errors in about 50% of the observations without driving the bias to infinity.

Investigations concerning the breakdown point of some estimators can be found in Bloomfield and Steiger [1983] and Rousseeuw and Leroy [1987]. Maronna *et al.* [1979] have shown that the breakdown point of the *bounded-influence estimators* decreases with increasing dimension p, that is when more outliers are likely to occur.

3.3.2 The Least Median of Squares Estimate

Rousseeuw and Leroy [1987] proposed "high breakdown point" methods to cope with a large fraction of errors in the regression analysis. The main idea of their proposal is to design robust estimators with a high breakdown point based on the median. Therefore, Rousseeuw [1984] proposes to **minimise the median of the squared residuals (LMS)** based on the idea of Hampel [1975]. The Least Median of Squares is given by,

$$\min(\mathrm{med}(r_i^2)) . \tag{3.10}$$

The LMS estimate is equivariant with respect to linear transformations of the A-matrix and has a breakdown point of 50%. A disadvantage remains its poor asymptotic efficiency (see also 2.3).

3.3.3 The Least Trimmed Square Estimate

To improve the asymptotic efficiency of the LMS, Rousseeuw [1983, 1984] introduced the Least Trimmed Squares estimator (LTS) given by,

$$\min \sum_{i=1}^{h} (r^2)_{i:n}, \tag{3.11}$$

where $(r^2)_{1:n} \leq \ldots \leq (r^2)_{n:n}$ are the ordered squared residuals (the residuals are first squared and then ordered). This estimator is very similar to LS, the only difference is that the largest squared residuals are not used in the summation. The best robustness properties are when h is approximately $n/2$, in which case the breakdown point attains 50% [Rousseeuw and Leroy, 1987].

Both LMS and LTS are defined by minimising a robust estimator of the scatter of the residuals. This idea is generalised in the S-estimators, which minimise a certain type of M-estimate of the scale of residuals.

3.3.4 The S - estimators

The S estimators are defined by

$$\text{Min}(s(x)), \tag{3.12}$$

where s(x) is a certain type of M-estimate of the scale of residuals. These estimators have the same asymptotic efficiency as the M-estimators but retain a very high breakdown point [Rousseeuw and Leroy, 1987].

Chapter 4

Robust Estimators for Geodetic Transformations

4.1 Linear Transformations - Introduction

Under the term 'transformation of coordinates' is to be understood the transformation of the coordinates of a given point from one coordinate system to another. In order to determine the form of the function needed to carry out the transformation, points with known coordinates in both coordinate systems are required. The issuing coordinate system is often called the *local coordinate system*, whereas the destination coordinate system is commonly referred to as the *global coordinate system*. Identical points in space found in both coordinate systems are called *passpoints*.

Coordinates X,Y, in the global coordinate system are related to the coordinates x,y, in the local coordinate system through the transformation,

$$(X,Y) = F(x,y), \tag{4.1}$$

where the coefficients of the function are determined from the passpoints.

Any transformation problem is solved in the following steps:

1. The unknown parameters of the transformation function are determined by the co-ordinates of the passpoints. These parameters are normally estimated through a Least Square estimator.

2. Through the transformations parameters determined in the first step the coordinates of points of the local system are transformed in the global system.

The function F is very important since it gives the qualities of the transformation. The two most typical transformation methods employed in geodesy are the Helmert transformation and the general linear transformation, also known as the affine transformation.

4.1.1 Helmert Transformation - Mathematical Model

In the Helmert Transformation the form of figures does not change upon transformation from one coordinate system to the other. This kind of transformation is also known as a similarity transformation (Fig. 4.1).

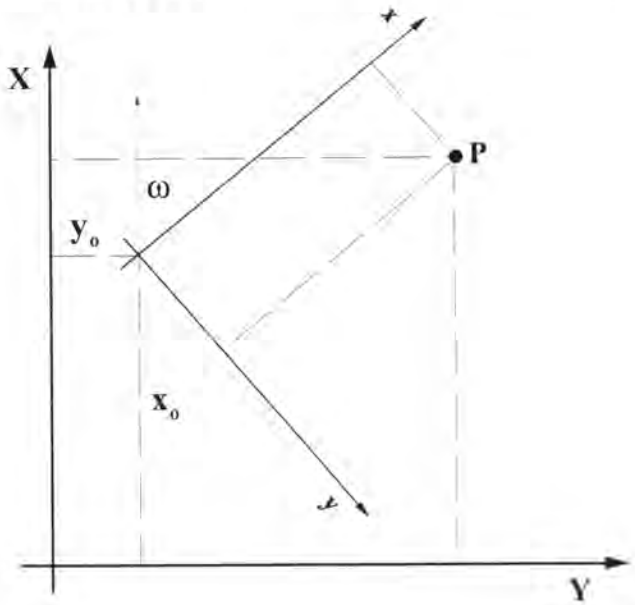


Fig. 4.1 Transformation of coordinates through the Helmert Transformation

The parameters used for the Helmert Transformation of coordinates are two translations $\mathbf{x_0}$ and $\mathbf{y_0}$, one rotation ω and one scaling factor **m** (see Fig. 4.1). The mathematical function for the transformed coordinates X and Y is given by,

$$\begin{aligned} Y &= y_0 + m\cos\omega \cdot y + m\sin\omega \cdot x \\ X &= x_0 - m\sin\omega \cdot y + m\cos\omega \cdot x \end{aligned} \quad , \tag{4.2}$$

where,

y_0	translation in y direction
x_0	translation in x direction
m	the scaling factor
ω	rotation about the z axis.

These four parameters are called the transformation parameters of the Helmert Transformation.

Equation 4.2 can be transformed into linear form by the following substitution,

$$\begin{aligned} a &= m \cdot \cos\omega \\ b &= m \cdot \sin\omega \end{aligned} \quad , \tag{4.3}$$

from which we obtain,

$$\begin{aligned} Y &= y_0 + a \cdot y + b \cdot x \\ X &= x_0 - b \cdot y + a \cdot x \end{aligned} \quad . \tag{4.4}$$

The parameters y_0, x_0, a and b are normally determined by the Least Square method, providing that the number of *passpoints* is at least two.

Some of the characteristics of the Helmert transformation method are:

- it has four transformation parameters
- the scaling factor is the same in all directions and all places (conform resembling)
- the form of figures in the local coordinate system remains unchanged after the transformation
- the correlation between the fictitious observations is assumed, in general, to be zero.

4.1.2 Affine Transformation - Mathematical Model

If the condition of similarity in the Helmert Transformation is abandoned, equation 4.4 can be written in general form as per equation 4.5,

$$\begin{aligned} Y &= a + by + cx \\ X &= d + ey + fx \end{aligned} \tag{4.5}$$

This equation expresses the affine transformation, where six transformation parameters are to be determined (**a**, **b**, **c**, **d**, **e** and **f**). It is also possible to express the transformation parameters as follows:

y_o, x_o	two translations (in the *y* and *x* directions)
ω_x, ω_y	two rotations
m_x, m_y	two scaling factors.

The affine transformation can then be expressed by,

$$\begin{aligned} Y &= y_o + m_y \cdot \cos\omega_y \cdot y + m_x \cdot \sin\omega_x \cdot x \\ X &= x_o - m_y \cdot \sin\omega_y \cdot y + m_x \cdot \cos\omega_x \cdot x \end{aligned} \tag{4.6}$$

As in the Helmert Transformation, the transformation parameters of the affine transformation are determined through the Least Square method, providing that the number of *passpoints* is at least three.

Some of the characteristics of the mathematical model for the affine transformation are:

- it has six transformation parameters
- the form of figures does not necessarily stay the same after the transformation
- the angles will be distorted
- the circles will become ellipses.

4.2 Linear Transformations in Geodesy

Linear transformations are used in Geodesy in order to compare repeatedly determined positions of reference points. The Helmert transformation can be used to compare the resultant coordinates of points of triangulation networks and deformation measurements. It can also be used as a first step in a complex interpolation process, when the precondition is that the local and global coordinates have to be in the same space. The affine transformation can be used for the elimination of a deformation represented in the main direction of the paper plan (when a transformation of digitised points in a plan to the global system of coordinates is needed) or in an interpolation process in a triangulated area.

In all these applications there are two principal cases to be distinguished:

1. The comparison of the same network measured twice, independently from each other (e.g. comparison of triangulation networks, deformation measurements, etc.). The coordinates previously determined need to be compared with those of a later time in order to ascertain whether the values of the coordinates in both systems belong to the same points and how large possible deviations are.

   Example 1: An existing geodetic network (Fig. 4.2) around the power plant is to be monitored and, if necessary, supplemented in order to provide an early indicator of possible movements of the power plant. By a Helmert transformation, it is possible to compare the measurements of different epochs, and, determine which of the points have moved in this period of time.

   Characteristics of the example:
   - *The number of passpoints is normally large, therefore, the number of redundant observations is also large*
   - *The number of points that might have moved could be relatively large*
   - *The errors could be in the local coodinates as well as in the global coordinates.*

2. The fitting of a free geodetic network in the fixed points system (e.g. datum transformation). In a problem like this the determined coordinates in a local system need to be adjusted at the existing reference system (global system).

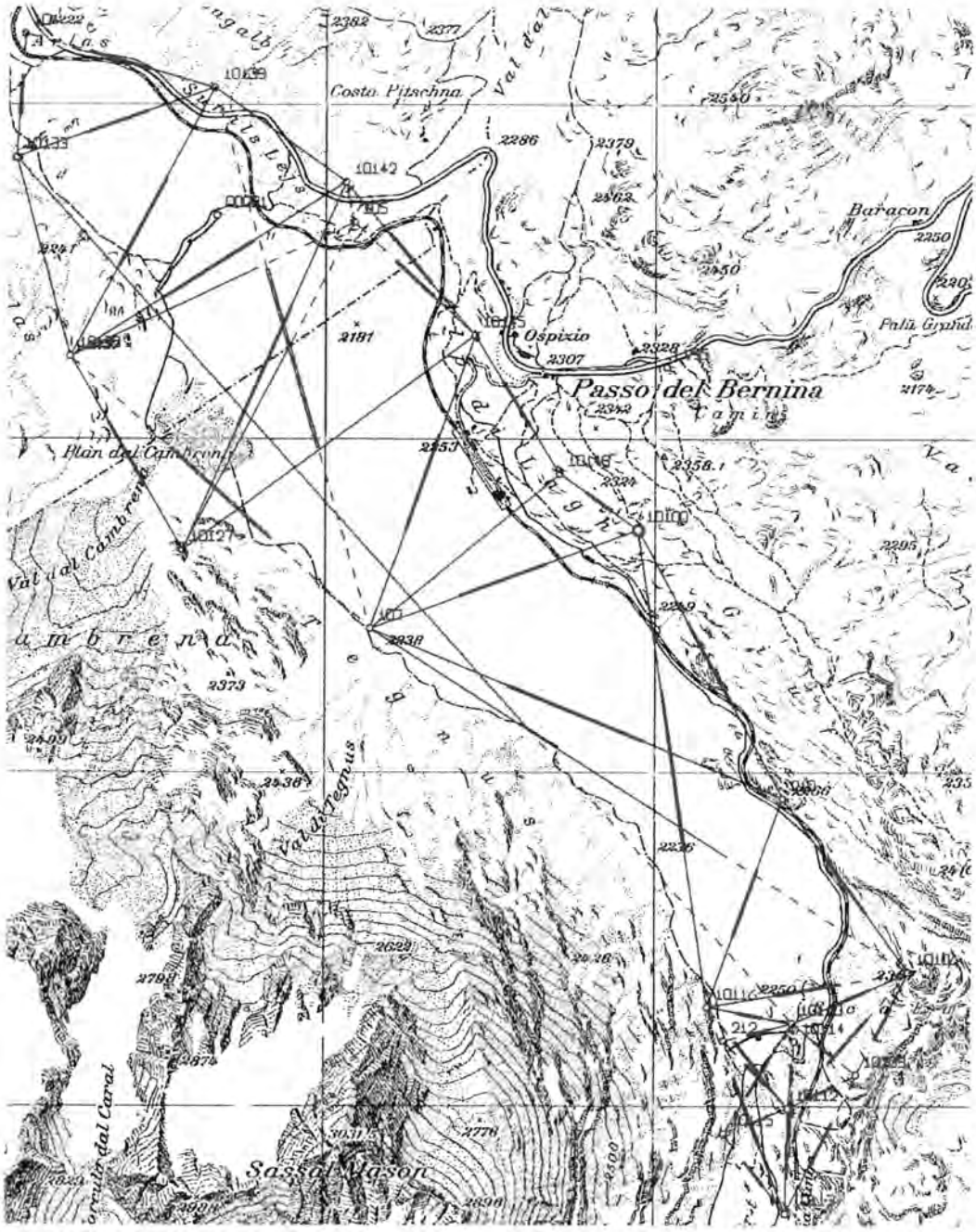


Fig. 4.2 Geodetic network around the Lago Bianco

Example 2: A set of points (Fig. 4.3) has been digitised with a certain accuracy from a general map of scale 1:10000. Their coordinates lie in an arbitrary coordinate system. The coordinates of 11 points are also known in the official projection system (global coordinates). The goal is to provide the transformation parameters needed to transform all points from the local system of coordinates into the global system of

coordinates. The resulting differences between the previous (local) and the fix (global) coordinates are the parameters that are to be interpreted.

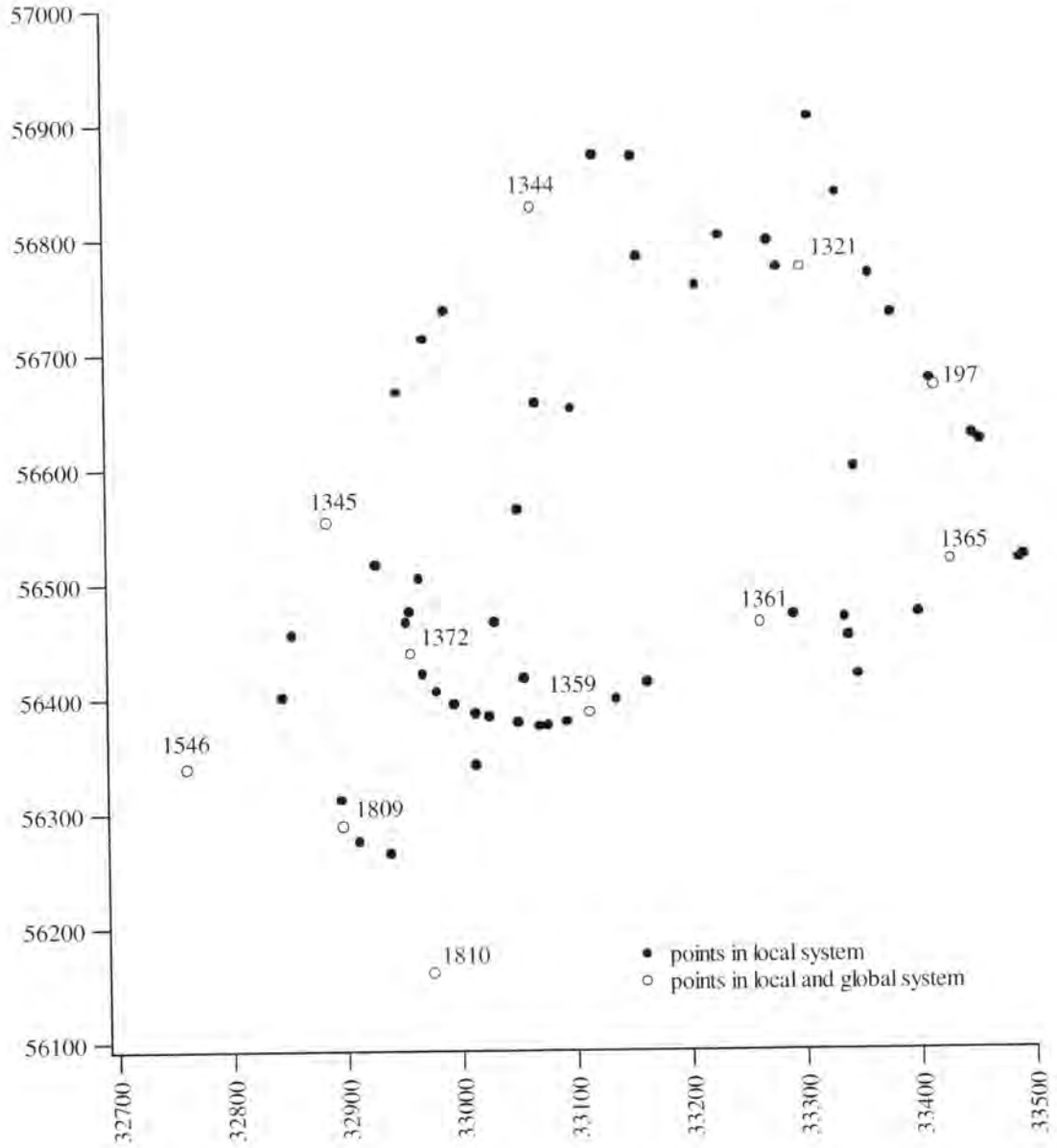


Fig. 4.3 Digitised points from a cadastral plan

Characteristics of the example:

- *The number of outliers is expected to be relatively small*
- *Outliers can be either points (in the local system) that have been measured with an error, or any of the fixed points (global system) that might have moved in the passage of time*

- *Often the local system of coordinates is more precise than the global one*

The transformation parameters in both examples should be determined in such a way that the fit of the local coordinates into the global coordinates should be as good as possible. The unknown parameters of the transformation method employed can be determined either through a Least Square or robust adjustment. These two ways of parameter determination are presented below. The advantages and disadvantages of the methods employed accompanied with numerical examples shall be presented at the end of this chapter.

4.3 Determination of the Parameters by Least Square Method

An adjustment process is possible only if the number of the observations is greater than the number of unknown parameters. Therefore, the number of passpoints needed for the determination of the unknown parameters should be at least two at the Helmert Transformation (four unknown parameters) and three at the affine Transformation (six unknown parameters). For the case where the number of passpoints is equal to two for the Helmert Transformation and three for the affine Transformation, the residuals are equal to zero. The steps of the adjustment process, for the determination of the transformation parameters by the Least Square method, are outlined below:

4.3.1 Helmert Transformation

Let n be the total number of observations and n/2 the number of passpoints. The equation of the ith residual, regarding the global coordinates as the source of the error, is given by,

$$\begin{aligned} Y_i + r_{y_i} &= y_0 + ay_i + bx_i \\ X_i + r_{x_i} &= x_0 - by_i + ax_i \end{aligned}, \quad \text{for } i = 1, \ldots, n/2 \tag{4.7}$$

The equation of residuals (4.7) can be expressed in matrix form by the following equation,

$$r = Ax - l, \tag{4.8}$$

where x is the vector of the unknown parameters y_o, x_o, a, and b.

If the unknown parameters are determined by the Least Square method, the function to be minimised is that of the sum of the squared residuals. The ***design matrix A*** is given by,

	y_o	x_o	a	b
Y_1	1	0	y_1	x_1
X_1	0	1	x_1	$-y_1$
...	...	...	...	...
...	...	...	...	...
$Y_{n/2}$	1	0	$y_{n/2}$	$x_{n/2}$
$X_{n/2}$	0	1	$x_{n/2}$	$-y_{n/2}$

(4.9)

Matrix A (n,n)

and the ***vector with the free terms*** $l^T = (X_1, Y_1, ..., X_{n/2}, Y_{n/2})$. (4.10)

We can obtain the normal equation by differentiating 4.8 with respect to the unknown parameters y_o, x_o, a, and b and equating the partial derivatives to zero:

$$A^T A \cdot x = A^T l \ , \tag{4.11}$$

where the **matrix of the normal equation** $A^T A = N$ is equal to,

$A^T A$			
n/2	0	[y]	[x]
...	n/2	[x]	-[y]
...	...	$[x^2 + y^2]$	0
...	...	...	$[x^2 + y^2]$

(4.12)

the vector of the unknown parameters

$$x = \begin{bmatrix} y_o \\ x_o \\ a \\ b \end{bmatrix}, \quad (4.13) \quad \text{and the vector } A^T l = Z = \begin{bmatrix} [Y] \\ [X] \\ [yY]+[xX] \\ [xY]-[yX] \end{bmatrix}. \quad (4.14)$$

In the matrices above [x] denotes the sum of the elements of the vector x.

To remedy numerical problems (the loss of accuracy during the calculation process) it is recommended to shift the origin to the centre of gravity (or close to that) for both coordinate systems. Therefore, the coordinates x_i, y_i employed for the local system of coordinates and X_i, Y_i for the global system of coordinates) are calculated relative to the centre of gravity. The calculation of the new coordinates is explained below:

We denote by $\mathbf{x_s}$, $\mathbf{y_s}$ and $\mathbf{X_s}$, $\mathbf{Y_s}$ the coordinates of the centres of gravity of the local and global system of coordinates. The coordinates $\mathbf{x_s}$, $\mathbf{y_s}$ and $\mathbf{X_s}$, $\mathbf{Y_s}$ are

$$x_s = \frac{[x]}{n/2}, \quad y_s = \frac{[y]}{n/2}, \tag{4.15}$$

$$X_s = \frac{[X]}{n/2}, \quad Y_s = \frac{[Y]}{n/2}. \tag{4.16}$$

Each coordinate of the passpoints in the local and global system of coordinates expressed by the coordinates of the centre of gravity will be,

$$\begin{aligned} x_{i(new)} &= x_i - x_s, \quad y_{i(new)} = y_i - y_s \\ X_{i(new)} &= X_i - X_s, \quad Y_{i(new)} = Y_i - Y_s \end{aligned}. \tag{4.17}$$

From equations 4.14, 4.15 and 4.16 the following relations are determined:

$$\begin{aligned} [x_{i(new)}] &= [x_i] - (n/2)\cdot[x_s] = [x_i] - [x_i] = 0 \\ [y_{i(new)}] &= [y_i] - (n/2)\cdot[y_s] = [y_i] - [y_i] = 0 \\ [X_{i(new)}] &= [X_i] - (n/2)\cdot[X_s] = [X_i] - [X_i] = 0 \\ [Y_{i(new)}] &= [Y_i] - (n/2)\cdot[Y_s] = [Y_i] - [Y_i] = 0 \end{aligned} \tag{4.18}$$

In equations 4.17 and 4.18 the abbreviation "new" indicates coordinates calculated from the centre of gravity of the system. If the coordinates employed are calculated this way, the unknown parameters can be calculated very easily from the system of normal

equation 4.11. In this equation, as per 4.18, the sum [x], [y], [X], [Y] is equal to zero, which leads to the following solution of the system of the normal equations:

$$y_o = Y_s - y_s, \quad x_o = X_s - x_s, \quad a = \frac{[y \cdot Y] + [x \cdot X]}{[x^2 + y^2]}, \quad b = \frac{[x \cdot Y] - [y \cdot X]}{[x^2 + y^2]}. \tag{4.19}$$

Nevertheless, irrespective of the way the coordinates are calculated, the solution of the normal equations delivers the unknown parameters.

4.3.2 Affine Transformation

The equation of the ith residual at the affine transformation is given by,

$$\begin{aligned} Y_i + r_{y_i} &= a + by_i + cx_i \\ X_i + r_{x_i} &= d + ey_i + fx_i \end{aligned} \tag{4.20}$$

The matrix form of the equation 4.20 is given by equation 4.8. The ***design matrix A*** of the coefficients of the equations of the residuals is given by,

	a	**d**	**b**	**c**	**e**	**f**
$\mathbf{X_1}$	1	0	0	0	$-y_1$	x_1
$\mathbf{Y_1}$	0	1	y_1	x_1	0	0
...	...	...	...	...	...	...
...	...	...	...	...	...	...
...	...	...	...	...	...	...
$\mathbf{X_{n/2}}$	1	0	0	0	$-y_{n/2}$	$x_{n/2}$
$\mathbf{Y_{n/2}}$	0	0	$y_{n/2}$	$x_{n/2}$	0	0

Matrix A (n,n) (4.21)

and the ***vector of free terms*** $l^T = (X_1, Y_1, ..., X_{n/2}, Y_{n/2})$. (4.22)

If the unknown parameters are determined by the Least Square method, the function to be minimised is that of the sum of the squared residuals. The matrix $N = A^TA$ that contains the coefficients of the normal equations is given by,

A^TA					
n/2	0	0	0	[y]	[x]
	n/2	[y]	[x]	0	0
		$[y^2]$	[xy]	0	0
			$[x^2]$	0	0
	sym.			$[y^2]$	-[xy]
					$[x^2]$

(4.23)

x, the vector of the unknown parameters (a, b, c, d, e and f) and $Z = A^Tl$ are given by,

$$X = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} \quad (4.24) \qquad Z = \begin{bmatrix} [X] \\ [Y] \\ [yY] \\ [xY] \\ -[yX] \\ [xX] \end{bmatrix}. \quad (4.25)$$

If the origins of the local and global systems of coordinates are shifted to the centres of gravity of these two systems, then the sums [x], [y], [X], [Y] are equal to zero. Therefore, the solution of the system of the normal equations leads to the following transformation parameters:

$$a = X_s - x_s \qquad b = \frac{[x^2]\cdot[yY]-[xy]\cdot[xY]}{[x^2]\cdot[y^2]-[xy]\cdot[xy]} \qquad c = \frac{[y^2]\cdot[xY]-[xy]\cdot[yY]}{[x^2]\cdot[y^2]-[xy]\cdot[xy]}$$

$$d = Y_s - y_s \qquad e = \frac{-[x^2]\cdot[yX]-[xy]\cdot[xX]}{[x^2]\cdot[y^2]-[xy]\cdot[xy]} \qquad f = \frac{[y^2]\cdot[xX]+[xy]\cdot[yX]}{[x^2]\cdot[y^2]-[xy]\cdot[xy]} \quad (4.26)$$

Once again, irrespective of the way the coordinates are calculated, the solution of the normal equations delivers the transformation parameters a, b, c, d, e and f.

4.3.3 Outliers in the Transformation Model

Outliers in the transformation model may arise in both observations and coefficients of the A-matrix:

- Outliers in the observations are errors in the local coordinates (e.g. the local coordinates have been determined with an error) or errors in the global coordinates (e.g. points that have changed their position over a period of time).

- If the coordinates in the local system contain gross errors, comparable to the number and size of the errors present in the coordinates of the global system, the coefficients of the A-matrix will be incorrect. These errors can then be called outliers in the design matrix A.

When outliers are present the transformation parameters will be wrong if the Least Square estimator determines them. Therefore, implementing a robust estimator, that is able to detect outliers, both in the observations and the coefficients of the A-matrix, is of great advantage.

4.4 Determination of the Unknown Parameters by Robust Estimators

To determine the unknown parameters, an estimation function is needed. In chapter 3 an overview of classes of robust functions was presented. The high breakdown point estimators are suitable for the determination of unknown parameters in the transformation model for the following reasons:

- high breakdown point estimators can cope with a large number of outliers (up to 50%)

- high breakdown point estimators not only identify outliers in observations, but in the design matrix as well

- the number of unknown parameters in the linear transformation is limited (2 for translation, 3 for translation and rotation, 4 for Helmert transformation and 6 for

affine transformation), therefore the calculation time can be confined to manageable proportions

- With a high breakdown point estimator, it is possible to determine the approximation parameters needed for the linearisation process, very precisely. This is very advantageous if the final determination of the unknown parameters is performed by employing a non-high breakdown point estimator (e.g. a bounded influence estimators).

A disadvantage, which can be mentioned, is the poor asymptotic efficiency of a LMS estimator.

In spite of the advantages of the high breakdown point estimators, it is also important to obtain a good estimation of parameters. Therefore, in cases when only a few large outliers are present (see example 2 in 4.2) a bounded influence estimator, which can limit the influence of outliers and is also efficient (small variance), is more appropriate.

4.4.1 The Transformation Program TRINETH

One of the aims of this work is to implement robust estimators in prototype software that enables transformation of coordinates. TRINETH is a later version of the software program TRANSINT of the Swiss Federal Office of Topography [Carosio, 1981], which enables geometrical transformation and interpolation between coordinates.

The transformation software TRANSINT provides the linear transformation Helmert (four unknown parameters) and the affine (six unknown parameters). It is also possible to compute a transformation with just a translation (two unknown parameters) or translation and rotation (three unknown parameters).

The transformation parameters may be determined by the following procedures:

Least Square method — The determination of the parameters by the Least Square method is explained in 4.3. It is clear that a single outlier can ruin this method of parameter determination.

Huber estimator

By the Huber estimator (see also 3.1) it is possible to identify outliers in the data points. However as an M-type estimator, the identification of outliers is confined only to the outliers in the observations, and not in the design matrix A.

Taking into consideration the problems that often transpire in geodetic transformations (see 4.2) and the advantages of robust estimators outlined in 4.4, we have concentrated our research towards the development and implementation, in TRINETH, of appropriate algorithms for the geometrical transformation of coordinates based on two robust estimators:

- LMS-estimator from the class of high breakdown point estimators and
- BIBER-estimator [Wicki, 1999] from the class of bounded influence estimators

4.4.2 LMS-Estimator

In Rousseeuw and Leroy [1987] the properties for the LMS estimator were presented. In this work we shall mention only the characteristics that are important for the practical implementation of this estimator.

Definition

The LMS-estimator is defined as a solution of the system

$$\underset{i}{\text{Minimise}}\,(\text{med}(r_i^2)) \tag{4.27}$$

Rouseeuw and Leroy [1987] have proved that there always exists a solution of the minimum problem, but the number of local minima of the objective function is typically very large. An algorithm that calculates the global minimum is required.

Method of computation

There are two computation algorithms that take us to the solution of the LMS-estimator; *the combinatorial* and *the resampling algorithm:*

Let $(a_{i,j}, l_i)$ be a given complete data set for i = 1, ..., n and j = 1, ..., p. We are looking for the solution $x^T = (x_1, ..., x_p)$ for which the value of the objective function $\mathrm{med}(r_i^2)$ is minimum.

We will denote by p the number of unknown parameters and by q $(q \geq p)$ a number of data points drawn from the matrix A. These q data points (observations) form a subsample (see Fig. 4.4).

$$\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} \\ a_{5,1} & a_{5,2} & a_{5,3} & a_{5,4} \\ a_{6,1} & a_{6,2} & a_{6,3} & a_{6,4} \\ \cdots & \cdots & \cdots & \cdots \\ a_{i,1} & a_{i,2} & a_{i,3} & a_{i,4} \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ a_{n-1,1} & a_{n-1,2} & a_{n-1,3} & a_{n-1,4} \\ a_{n,1} & a_{n,2} & a_{n,3} & a_{n,4} \end{bmatrix} \longrightarrow \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} \\ a_{6,1} & a_{6,2} & a_{6,3} & a_{6,4} \\ a_{n-1,1} & a_{n-1,2} & a_{n-1,3} & a_{n-1,4} \end{bmatrix}$$

matrix A

a possible subsample for $q \geq p$

Fig. 4.4 Complete data set and choice of subsamples

If we are interested in all possible subsamples without repetition, then the total number of the subsamples to be created is,

$$C_q^n = n!/(q! \cdot (n-q)!) . \tag{4.28}$$

For each subsample we can compute an LS estimate, determine the residuals and calculate the $\mathrm{med}(r_i^2)$. The jth subsample that provides the smallest $\mathrm{med}(r_i^2)$ is the solution x for which the objective function is minimum. Because the minimum of

function 4.27 is determined this way (by the exhaustive search of all subsamples) this computation method is called the *combinatorial algorithm*.

For the linear geodetic transformation (2, 3, 4 and 6 parameters) the number of subsamples that can be created is given in table 4.1 for different numbers of passpoints. It is obvious that for other geodetic applications, where the number of unknown parameters is greater than six, the number of the systems to be solved is extremely large.

q / n	2 *translation*	3 *translation and rotation*	4 *Helmert transformation*	6 *affine transformation*
10	45	120	210	210
20	190	1140	4845	38760
50	1225	19600	230300	15890700
100	4950	161700	3921225	1192052400
500	124750	20708500	2573031125	$2.10577 \cdot 10^{13}$

Tab. 4.1 Number of systems to be solved for linear transformation (combinatorial algorithm)

Since the *combinatorial algorithm* is quite unrealistic in the majority of cases, a *resampling algorithm* is a reasonable approach. Using the resampling algorithm, it is possible to set the number of subsamples (NSUB) in advance, instead of computing all subsamples. Rouseeuw and Leroy [1987] have chosen the NSUB number in such a way that the probability α of at least one of the subsamples being "good" is almost one. If we denote by ε the fraction of bad observations in a sample, then a subsample is "good" if it consists of q "good" observations. Assuming that NSUB/q is large, this probability is,

$$\alpha = 1-(1-(1-\varepsilon)^{q})^{NSUB}. \tag{4.29}$$

This number is almost equal to 99% if the NSUB $\approx 4.6 \cdot 2^{q}$. It is quite interesting to see that the probability α is independent of the number of the observations n, but depends only on the number of the subsamples and the number of the observations belonging to a subsample. In the table below is given an approximated number of NSUB for the unknown parameters at the transformations employed in our algorithm. The number of subsamples outlined in Table 4.2 corresponds to α greater than 99%, and ε equal to 50%.

q	2 *translation*	3 *translation and rotation*	4 *Helmert transformation*	6 *affine transformation*
NSUB	1000	1500	2000	3000

Tab. 4.2 Number of systems to be solved for linear transformations (resampling algorithm)

The choice of the observation equations in the determination of the unknown parameters

It is very important to notice that both calculation algorithms presented before, take us to the required solution only if the matrices of the system of equations of each subsample of q data points are not singular. In other words, all p unknown parameters have to be determined from the q data points of the subsample.

A preselection of the observation equations will contribute to the acceleration of the algorithms proposed, since the number of systems to be solved will be reduced. For the Helmert Transformation this preselection is given below:

From the equation of residuals, it is seen that all four parameters can be determined if the subsamples are created as follows:

a. Combination of two passpoints (two observations of type x and two of type y)

b. Combination of three observations of the type x and one of type y

c. Combination of three observations of the type y and one of type x

Any subsample of the combination of four observations of x or four observations of y leads to a singular matrix of the system of the normal equations. Therefore, an additional algorithm that omits these subsamples could be added to the combinatorial algorithm presented before.

If we denote by n the total number of observations (*the number of passpoints is* n/2), the half are observations of x type and the other half are observations of y type. The number of subsample combinations that should be avoided is C_3^n which corresponds to almost ten percent of the total number of subsamples of the combinatorial algorithm in the exhaustive search. It is important to note that by taking into consideration only the

subsamples created as a combination of only two passpoints, the minimum of the function 4.27 is not going to be very precise compared to the results of the exhaustive search.

A similar situation is found for the affine transformation, where the subsamples created from only x or only from y observations are going to produce a singular matrix of the system of normal equations.

Different opinions also exist for the choice of the number of observations that complete a subsample. As presented in the calculation procedure (combinatorial or resampling algorithm) the choice q (number of observations in a subsample) is variable:

- q = p represents the case where the number of the observations chosen for a subsample is equal to the number of unknown parameters

- q > p represents the case where the number of observations extracted to create the subsample, is larger than the number of the unknown parameters. Aktinson and Weisberg [1991] have shown that the results of the resampling algorithm for q = p are very "rough" if n is small compared to p. The use of q > p seems to produce better results in those cases, but the number of systems of normal equations to be solved increases rapidly. Marazzi and Joss [1990] have shown that for q = p+1, it is always possible to find the absolute minimum of the LMS estimator in the combinatorial algorithm.

Calculation procedure

From the foregoing we can say that the algorithm chosen for the determination of the unknown parameters by a LMS estimator at the linear transformations depends upon:

- *the number of observations (number of passpoints)*
  The number of observations needed to determine the unknown parameters by the LMS estimator influences directly the results of the algorithm and the calculation time. It is recommended to employ this estimator when n > 5p [Aktinson and Weisberg, 1991].

- *the number of unknown parameters to be determined*
  At the linear transformation this number cannot reach values larger than six. For a small number of passpoints therefore, the combinatorial algorithm comes into question. However, even when the number of passpoints is large, the resampling algorithm may not deliver reliable values. An alternative remains a bounded influence estimator.

- *the number of the observations drawn to form a subsample*
  If q > p the number of subsamples increases rapidly and so does the calculation time. Still, this option is recommended when data is sparse [Marazzi, 1993].

- *the type of the observations drawn in a subsample*
  As explained before, both of the aforementioned algorithms take us to the required solution only if the matrices of the system of equations of each subsample of *q* data points are not singular. Therefore, the omission of these types of subsamples will accelerate the computation.

It is obvious that all of these factors are interrelated and should be taken in consideration in the algorithm procedure for the parameter determination.

The calculation procedure that is implemented in TRINETH is explained below:

We denote by

a_{ij}, l_i	a complete data set
n	number of observations
p	number of unknown parameters
q	number of observations in a subsample
NSUB	number of subsamples
NIT	number of iterations
s*	initial value of the objective function $\mathrm{med}(r_i^2)$
s	value of the objective function for the jth subsample
x	final solution for the unknown parameters when the objective function is minimum
ϑ_j	solution of the LS for the ith subsample
ϑ_i^*	solution of the LS when objective function $\mathrm{med}(r_i^2)$ is minimum

The LMS estimator is calculated through the following steps:

First step - Compute the total number of subsamples NSUB = C_q^n or set the NSUB for the resampling algorithm from table 4.2, then set the number of iteration NIT to 1 and a large initial value s^* to the objective function.

Second step - Draw a subsample of $q \geq p$ data points.

Third step - If the subsample drawn at step two produces a single matrix, check if the number of iterations is less than NSUB. If yes, then NIT = NIT + 1; go to step two.

Fourth step - Compute the Least Square fit $\vartheta_1,\ldots,\vartheta_p$ for each subsample (the fit is exact if q = p).

Fifth step - Compute the residuals $r_i = \sum_{j=1}^{p} a_{ij}\vartheta_j - l_i$

Sixth step - Compute the value s of the objective function of the residuals $\mathrm{med}(r_i^2)$

Seventh step - if $s < s^*$, then $s^* = s$ and $\vartheta_j^* = \vartheta_j$.

Eighth step - if the number of iterations is less than NSUB, then NIT = NIT + 1; go to step two

Ninth step - Set $x = \vartheta^*_1,\ldots,\vartheta^*_p$ and $s = s^*$.

The *x* and *s* are the approximated values of the desired estimate and of the minimum of the objective function.

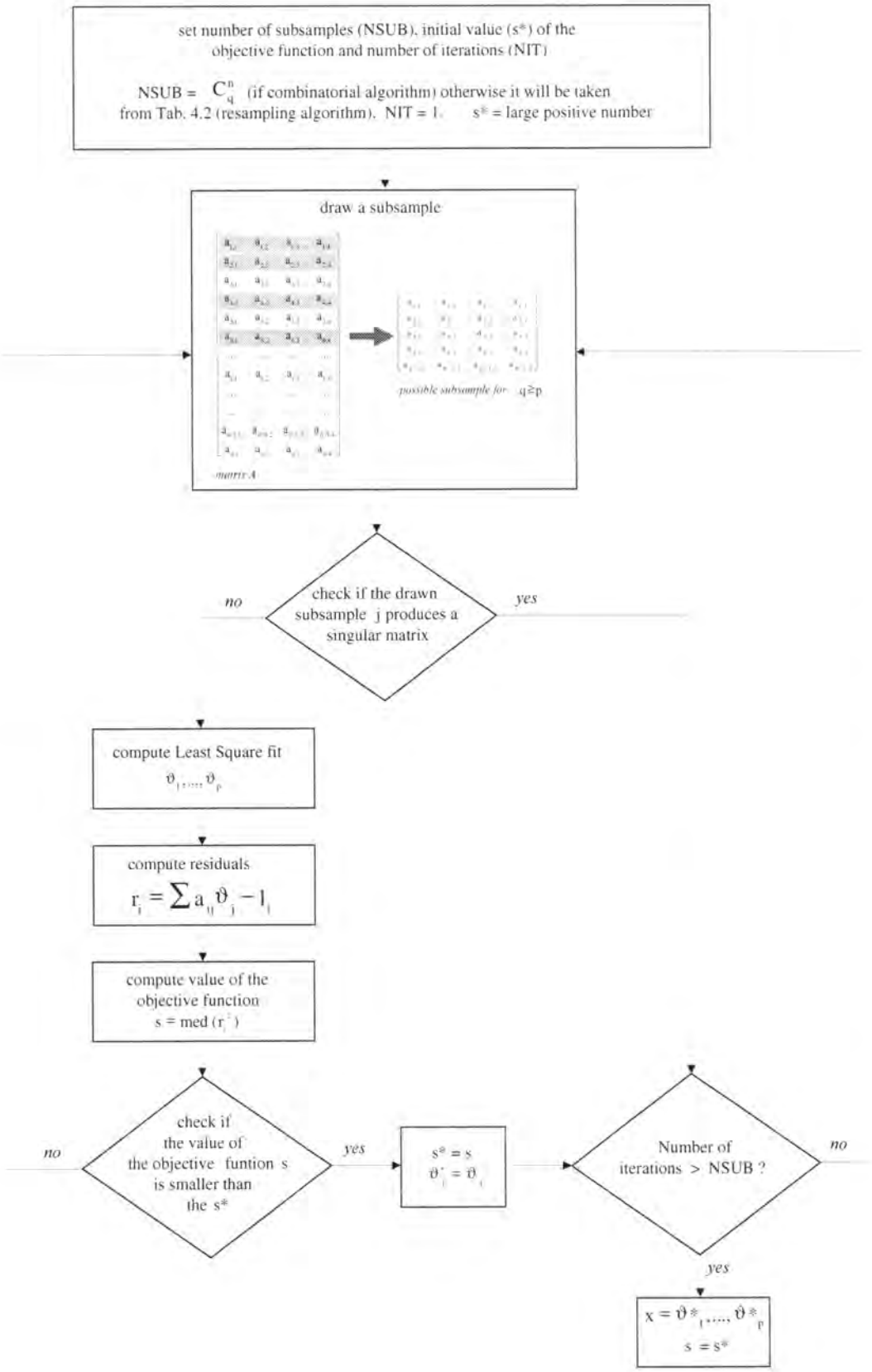


Fig. 4.5 Flow-diagram illustration showing the various steps in the algorithm procedure

Bounding the effect of outliers using the LMS - estimator

As presented in 2.3.2, the leverage points can be viewed as observations with small partial redundancy. The ordinate of such points is far away from the rest of the data set and its residual is,

$$r_i = a_{ij}\, x_j - l_i\,, \tag{4.30}$$

where a_{ij} is the ith row of the matrix A. In the two dimensional case, a_{ij} represents the ordinate of the leverage point.

A good robust estimator should bound the influence of outliers in such observations. The LMS estimator reduces the influence of large residuals by using the median. The large residuals, therefore, have no influence in the determination of parameters, providing that the amount of the contaminated data is not more than 50%.

However, once again it is emphasised: For unfavourable measurements disposition, even if the number of bad observations is less than 50%, the LMS estimator could have a large bias (see also 2.3.2). For example, in a geodetic network, when many outliers occur in the same network section. In this instance, the observations found at certain sections of the network are not verified from other observations.

Estimation of s_0 (LMS - estimator)

Rousseeuw and Leroy [1987] suggest a two step approach for the estimation of s_0:

First step

An initial scale estimate s^0 shall be determined based on the minimal median and multiplied by a correction factor (which depends on the *n* and *p*),

$$s^0 = 1.4826\,(1 + 5/(n-p))\sqrt{\underset{i}{\mathrm{med}}(r_i^2)}\,. \tag{4.31}$$

The factor $1.4826 = 1/\Phi^{-1}(0.75)$ was introduced [Rouseew and Leroy, 1987] because $\mathrm{med}_i|Z_i|/\Phi^{-1}(0.75)$ is a consistent estimator of σ when the Z_i are distributed like $N(0, \sigma^2)$.

From some empirical examples it is evident that the factor 1.4826 alone delivers values, which are too optimistic for the scale parameter, especially when data is sparse. Therefore, the additional factor (1+5/(n-p)) is added. It is obvious that this additional factor tends to one when *n* is large.

Second step

To reduce the influence of large residuals, the final scale parameter s_0 is determined as per,

$$s_0 = \sqrt{\sum (w_i \cdot w_{i\,lms} \cdot r_i^2) / (\sum w_{i\,lms} - p)} \tag{4.32}$$

where w_i are the weights of the observations. The preliminary value s^0 is used to determine the weights $w_{i\,lms}$

$$w_{i\,lms} = \begin{cases} 1 & \text{if } |r_i / s^0| \le k_{lim} \\ 0 & \text{otherwise} \end{cases} . \tag{4.33}$$

The choice of the value k_{lim} should be made in such a way that the final estimation of s_0 by the LMS estimator is comparable to that calculated by the LS (when no outliers are present). From our comparisons, when data is sparse this value can be taken 2.7. For a large number of passpoints (more then 15) this value can be taken 2.

4.4.3 BIBER-Estimator

Definition

This estimator is related to the M-estimators and more precisely to the Shweppe-type estimator (see 3.2.2). It is defined as a solution of the system,

$$\sum_{i=1}^{n} \Psi\left(\frac{r_i}{w_i}\right) \mu_i a_{ij} = 0 , \tag{4.34}$$

where the standard deviations of the residuals are used as additional weights μ_i [Wicki, 1999],

$$\mu_i = \sigma_{r_i} = \sigma_0 \sqrt{(q_{rr})_{ii}} \quad . \tag{4.35}$$

These additional weights are dependent only upon the given weights of the observations and the given design of the geodetic network. The functions ρ and ψ of this estimator are defined by:

$$\rho\left(\frac{r_i}{\sigma_{r_i}}\right) = \begin{cases} \frac{1}{2\sigma_{r_i}} r_i^2 & \text{for } \left|\frac{r_i}{\sigma_{r_i}}\right| < c \\ \frac{c}{\sigma_{r_i}}|r_i| - \frac{1}{2}c^2 & \text{for } \left|\frac{r_i}{\sigma_{r_i}}\right| \geq c \end{cases} \quad . \tag{4.36}$$

$$\Psi\left(\frac{r_i}{\sigma_{r_i}}\right) = \begin{cases} \frac{r_i}{\sigma_{r_i}} & \text{for } \left|\frac{r_i}{\sigma_{r_i}}\right| < c \\ \text{sign}(r_i)c & \text{for } \left|\frac{r_i}{\sigma_{r_i}}\right| \geq c \end{cases} \quad . \tag{4.37}$$

By the substitution of equation 2.25 in 4.37, the function Ψ is expressed as a function of the standardised residuals in the following form:

$$\Psi_c\left(\frac{r_i}{\sigma_{r_i}}\right) = \Psi_c(r^*_i) = \begin{cases} r^*_i & \text{for} \left| r^*_i \right| < c \\ \text{sign}(r^*_i) & \text{for} \left| r^*_i \right| \geq c \end{cases} \quad . \tag{4.38}$$

Since σ_{r_i} is always positive, $\text{sign}(r^*_i) = \text{sign}(r_i)$.

The limiting value c defines three different cases for the standardised residuals r^*_i: $r^*_i \leq c$, $-c < r^*_i < c$ and $r^*_i \geq c$. If the standardised residuals fall in the interval $-c < r^*_i < c$, then the unknown parameters are determined by the Least Square estimator, otherwise the influence of the observation(s) is limited.

The constant *c*, is determined by the user. From experience, in geodetic applications, it can be stated that the value of *c* generally falls between 2.5 to 4.

Equation 4.37 can be expressed also as a function of the residuals r_i.

$$\Psi_{k_i}(r_i) = \begin{cases} r_i & \text{for } |r_i| < k_i \\ \text{sign}(r_i)k_i & \text{for } |r_i| \geq k_i \end{cases}. \tag{4.39}$$

The cut off values $k_i = \sigma_{r_i} c$ are different for every residual, but remain invariant during the adjustment process since they are computed with the σ_{r_i} a priori, which also stays unchanged [Wicki, 1999].

From 4.38 and 4.39 the following relation can be written,

$$\Psi_{k_i}(r_i) = \sigma_{r_i} \Psi_c(r_i^*). \tag{4.40}$$

The BIBER estimator has been employed very successfully in the adjustment of geodetic networks, with hundreds of observations and unknown parameters [Wicki, 1999].

Calculation procedure

In order to solve equation 4.39 (as well as equation 4.38) a minimum problem has to be solved. The function that has to be minimised is different in three intervals. Therefore, residuals have to be classified in the three intervals. Because the residuals are correlated, meaning that an observation with an error causes a large residual to an observation without error, their classification is possible only iteratively. Wicki [1999] proposes the following calculation procedure:

First step – Least Square fit

By assuming that the number of outliers is not very large, the first adjustment could be performed by a Least Square method. A Least Square estimator gives the possibility of classifying all the observations to the middle interval.

Second step – choose the observations with the largest $|r_i^*|$

The observation with the largest absolute value of the standardised residual r^*_i, will be placed in the outside intervals. Since the residuals are correlated, it is very difficult to spot the observation with the real gross error. Therefore, we choose the observation that possesses the largest residual in absolute value (as well as the largest standardised residual r^*_i, in absolute value), if $|r_i^*|$ is greater than c.

Third step – calculate the unknown parameters and new residuals

After the observation with the largest residual has been identified (step two) it is possible to calculate new unknown parameters and residuals, where the influence of the observation, classified with the largest residual, is reduced.

The influence of such an observation can be reduced by applying one of the following methods:

1. Recalculating a part of the matrix of the normal equations
2. Correcting the observation l_i
3. Diminishing the weight w_i of the observation l_i

It is very important to emphasise that all three methods proposed lead to the same results. The employment of the first method means recalculating the matrix of the normal equations for each iteration (every time that an observation falls outside the middle interval). A similar situation is presented also when the weight of the observation l_i has to be modified. Both procedures require a lot of calculation time, therefore, Wicki [1999] employs the modification of the observation l_i (method 2). In this case no new inversion of the matrix of the normal equations is needed since its elements remain unchanged. This method goes about the task in the following way:

From the function Ψ of the BIBER - estimator (equation 4.39) it is easy to see that an observation with a large residual, which falls outside the interval $[-k_i, k_i]$, influences the rest of the data to the same extent as an observation with a residual equal to the value k.

Therefore, the goal of the algorithm is to modify all the observations that fall outside the interval $[-k_i, k_i]$ in such a way that their residual becomes exactly $-k_i$, or $+k_i$. The algorithm starts with the observation l_i that possesses the largest residual. The modified observation $l_{i\,(mod)}$ then has a small influence on the unknown parameters. The residual r_i of the observation l_i can be given as,

$$r_i = -(q_{rr})_{i1} l_1 - \ldots - z_i l_i - \ldots - (q_{rr})_{in} l_n . \tag{4.41}$$

We are looking for the modified observation where,

$$\mathrm{sign}(r_i) k_i = -(q_{rr})_{i1} l_1 - \ldots - z_i l_{i\,(mod)} - \ldots - (q_{rr})_{in} l_n . \tag{4.42}$$

We define d_i as.

$$d_i = r_i - \mathrm{sign}(r_i)k_i \qquad \text{for } |r_i| > k_i \tag{4.43}$$

Comparing 4.41 with 4.42 the modified observation will be.

$$l_{i(mod)} = l_i - \frac{d_i}{-z_i}. \tag{4.44}$$

After the observation has been modified, it is possible to calculate the new unknown parameters (x_{rob}) and the new residuals (r_{rob}). The inversion of the matrix of the normal equations is not necessary, since the coefficients of the design matrix A, and the weights w_l, have not changed.

The unknown parameters and residuals can be described in matrix form as per,

$$x_{rob}^{(1)} = x - (A^T A)^{-1} A^T \begin{pmatrix} 0 \\ \dots \\ \dots \\ \frac{d_i}{-z_i} \\ \dots \\ \dots \\ 0 \end{pmatrix}, \tag{4.45}$$

$$r_{rob}^{(1)} = Ax_{rob} - (l - \begin{pmatrix} 0 \\ \dots \\ \dots \\ \frac{d_i}{-z_i} \\ \dots \\ \dots \\ 0 \end{pmatrix}). \tag{4.46}$$

After this calculation, it must be established whether the new $r_{i\,rob}$ falls in the interval $[-k_i, k_i]$. This procedure is repeated iteratively until all observations with residuals outside the interval $[-k_i, k_i]$ either fall inside the interval or are modified. Upon completion of the process explained above, the robust parameters are determined. All residuals (of each iteration) brought to an observation l_i can be summarised together in,

$$\Delta d_i = \frac{1}{-z_i} \sum_{t=1}^{s-1} d_i^{(t)} , \tag{4.47}$$

where s is the number of the iterations within a cycle.

The total residual of the observation l_i can be calculated as follows,

$$r_{i\,tot} = k_i - \Delta d_i \ . \tag{4.48}$$

The equations 4.45 and 4.46 can be given in matrix form as per,

$$x_{rob} = x - (A^T A)^{-1} A^T \Delta d \ , \tag{4.49}$$

$$r_{rob} = A x_{rob} - (l - \Delta d) \ , \tag{4.50}$$

$$r_{tot} = A x_{rob} - l \ , \tag{4.51}$$

where the vector Δd is,

$$\Delta d = \begin{bmatrix} 0 \quad \text{or} \quad \frac{1}{-z_1} \sum (d_1) \\ \dots \qquad \dots \\ \dots \qquad \dots \\ \dots \qquad \dots \\ 0 \quad \text{or} \quad \frac{1}{-z_n} \sum (d_n) \end{bmatrix} . \tag{4.52}$$

The following equation relates r_{tot} and r_{rob},

$$r_{tot} = r_{rob} - \Delta d \ . \tag{4.53}$$

Bounding the effect of outliers using the BIBER-estimator

From the influence function of the BIBER – estimator (equation 4.39) it can be seen that this estimator bounds the influence of leverage points by the variation of the value k. This value depends upon the local reliability (or partial redundancy) of the observations.

The value k_i can be expressed as function of c and σ_{r_i} as per,

$$k_i = c\sigma_{r_i} = c\sigma_o\sqrt{z_i} \tag{4.54}$$

Therefore, when an observation has a small redundancy, its influence is limited. The BIBER estimator does not provide any results when the local reliability z_i is equal to zero.

Estimation of s_o (BIBER-estimator)

The estimation of s_o at the Least Square method (when the observations obey the normal distribution) is determined by the following relation,

$$s_o^2 = \frac{1}{n-p}\sum_{i=1}^{n}\Psi^2(r_i) = \frac{1}{n-p}\sum_{i=1}^{n} r_i^2 , \tag{4.55}$$

By analogy to 4.55, Wicki [1999], estimates the s_o at the BIBER estimator as per,

$$s_o^2 = \frac{1}{(n-p)\beta}\sum_{i=1}^{n}\Psi_{k_i}^2(r_i) , \tag{4.56}$$

where β is a constant such that s_o is asymptotically unbiased for normal errors,

$$\beta = \left(c^2 + (1-c^2)(2\Phi(c)-1) - 2c\varphi(c)\right), \tag{4.57}$$

and,

Φ: standard normal distribution,

φ: density of the standard normal distribution.

If σ_o is equal to one, then the estimation s_o in equation 4.56 can be expressed with the help of 4.39 as per,

$$s_o^2 = \frac{1}{(n-u)\beta}\left(\sum_I r_i^2 + c^2 \sum_{II} z_i\right), \tag{4.58}$$

where,

$\sum_I$ sum over all observations where $|r_i| < k_i$,
$\sum_{II}$ sum over all observations where $|r_i| \geq k_i$.

Calculation of the matrix of the cofactors of the unknown parameters

Wicki, [1999] proposes the following simple method for the computation of the matrix of cofactors:

$$Q^*_{xx} = \frac{1}{\sigma_o^2}\Sigma^*_{xx} \approx (A^T W^* A)^{-1}, \tag{4.59}$$

where Σ^*_{xx} is calculated from the law of the propagation of errors as per,

$$\Sigma^*_{xx} = \sigma_o^2 (A^T W^* A)^{-1} A^T W^* W^* A (A^T W^* A)^{-1}, \tag{4.60}$$

taking into consideration that the covariance matrix of the observation is $\Sigma_{ll} = \sigma_o^2 I$, and the matrix of the reduced weights W* [Wicki, 1999] is equal to,

$$w_i^* = \frac{\Psi_{k_i}(r_i)}{|r_i|}. \tag{4.61}$$

The matrix of cofactors of the unknown parameters calculated with the aid of 4.60 is,

$$Q^*_{xx} = (A^T W^* A)^{-1} A^T W^* W^* A (A^T W^* A)^{-1}. \tag{4.62}$$

Q^*_{xx} given in 4.59 approximates to Q^*_{xx} given in 4.62 when the number of gross errors is not large and the weight matrix W* is not greatly changed from the matrix W (equal to the identity matrix).

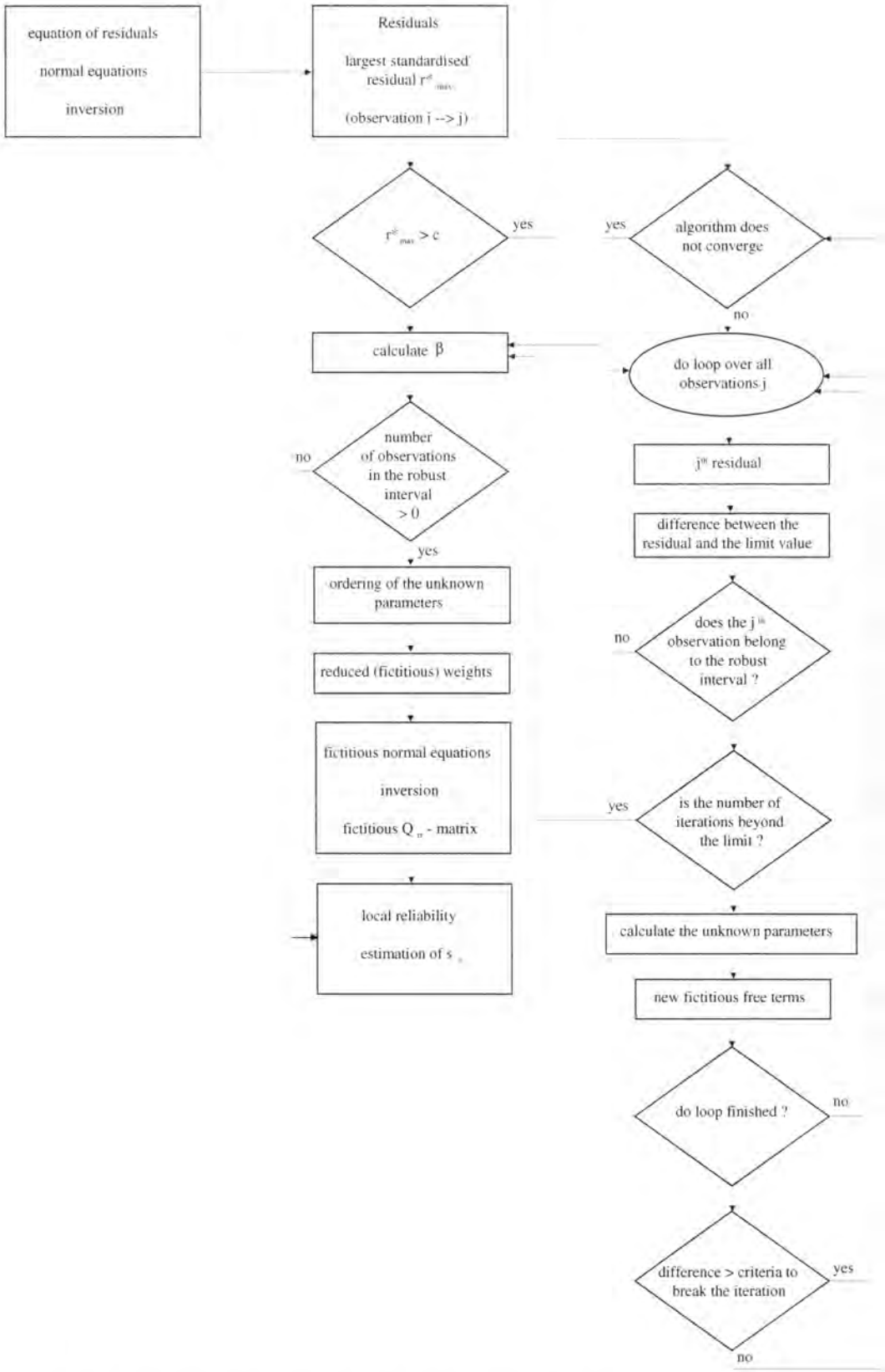


Fig. 4.6 Flow-diagram illustration of the calculations steps of the BIBER-estimator

4.4.4 Implementation in the Software TRINETH

User Input

The following estimators are integrated in TRINETH:

- LS estimator
- LMS estimator
- BIBER estimator

```
--- Transformation - Interface ---

1. Transform passpoints ? [Yes/No]                  : Yes
2. Number of unknown parameters  [2/3/4/6]          : 4

--- Given Parameters ---
3. Y-Translation [m]                                :
4. X-Translation [m]                                :
5. Rotation [gon]                                   :
6. Scaling factor [km/mm]                           :

--- Reduction of Coordinates to the Centre of Gravity ---
7. Global centre of gravity (Y/X) [m]               : 600000 150000
8. Local centre of gravity (Y/X) [m]                : 600000 200000

--- Adjustment Method ---
9. BIBER-estimator (c-robust)                       : 3.0
10. LMS-estimator [Yes/No]                          : No
```

Fig. 4.7 Transformation-Interface at TRINETH

In principle it is possible to use any one of the three procedures for a given problem; the user determines which of the procedures is more suitable.

The choice between different methods of parameter determination can be made very easily from the Transformation - Interface, where the only input to be given is the value of c for the BIBER-estimator, and for LMS *Yes* or *No*. When the parameter c is 0, the adjustment will follow by the Least Square method (a convention), Fig. 4.7.

Linearisation of the observation equation and determination of the approximated values

The transformation software TRINETH allows the transformation of coordinates by 2, 3, 4 or 6 parameters. The functions of the equations of the observations contain non linear functions only in the instance of the transformation with three parameters. Therefore, in order not to alter the calculation procedure, we apply the linearisation for all equations. The linearisation can be either analytical (a) or numerical (b).

In both cases, approximated values for the unknown parameters are needed. It is important to note that the result of the transformation depends on the way these approximated parameters are determined, particularly when the data is contaminated (gross errors are present). The coefficients of the A-matrix normally do not influence the results because the evaluation process is calculated iteratively. However, if the estimators used cannot cope with a large amount of outliers, then the bad approximated parameters will influence the results (the algorithm does not converge).

In TRINETH we determine four approximated values (Helmert transformation); since, after having these four parameters, it is possible with simple analytical formulae to determine the six parameters needed at the affine transformation.

One way of parameter determination is the Least Square Method, with the disadvantage that if outliers are present the approximated parameters are wrong. We suggest a high breakdown point estimator (LMS), since it is possible to identify the outliers, and as the number of unknown parameters is always four, the calculation time is reasonable. The approximated parameters determined this way are very good, which allows the employment of a non-high breakdown point estimator for the final determination of the unknown parameters.

The linearisation method employed is the numerical linearisation.

Adjustment process

As per the adjustment method chosen (the flow diagrams are given in Fig 4.5 and Fig. 4.6) the program proceeds with the calculation of transformation parameters and other important indicators. The adjustment process occurs iteratively, which allows the improvement of the linearisation process.

A flow chart given in Fig. 4.8 illustrates the algorithm procedure at TRINETH.

Numerical problems

In order to remedy the loss of accuracy during the calculation process, we suggest translating the coordinates of the points close to the centre of gravity of the two systems: this decision is left free for the user. If this is not done, the software produces the following message:

> *Warning:*
>
> *The coordinates are not reduced to the origin of the local and global systems (or close to the centre of gravity of the local and global systems). The program will calculate but the results could be wrong !!!*
>
> *Press <**Return**> to continue or <**Opt**> to return to the Interface*

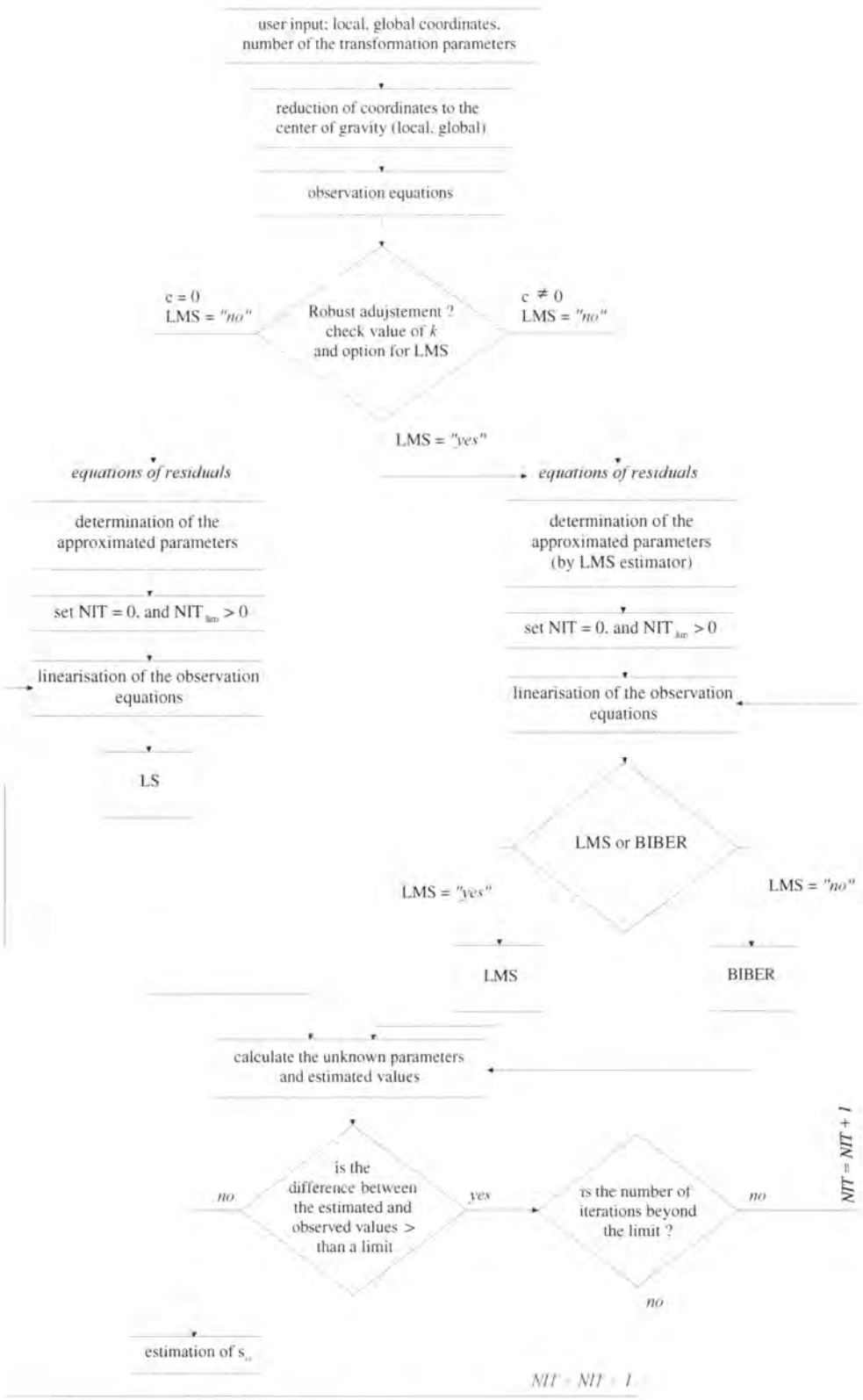


Fig. 4.8 Flowchart illustration of the adjustment process at TRINETH

Important indicators and listing

From the listing the following are of particular importance:

- *The method of estimation employed*
  LS, BIBER or LMS estimator

- *The transformation parameters as per the transformation method employed*
  Either Helmert; affine; translation; or translation and rotation.

- *The estimation of s_0*
  For each estimator, the s_0 can be estimated. For LS, see equation 4.55, BIBER estimator equation 4.58, LMS equation 4.32.

- *The estimation of the cofactors matrix of the unknown parameters*
  The determination of the matrix of the cofactors of the unknown parameters by the BIBER estimator, is calculated as per 4.59. When the LMS estimator is employed, the cofactor matrix of the unknown parameters is not calculated at all, since this type of estimator is not suitable for such a determination.

- *The passpoints with the residuals of the transformation*
  The residuals are the parameters that we are going to interpret in order to determine whether the deviations between coordinates are acceptable or not.

- *The local reliability z_i*
  When the parameters are determined by the LS the z_i represents the local reliability calculated by 2.24. For the BIBER estimator, the local reliability can be calculated in a similar way:

$$z_i^* = (q_{rr}^*)_{ii} w_i^*, \qquad (4.63)$$

where w* is calculated as per 4.61. For the LMS estimator, the local reliability, loses its significance, therefore, in the listing in the column of z_i is written *"LMS"*.

4.5 Applications and Results

In this section, through numerical examples, shall be presented the effectiveness of the different estimators employed. The comparison is going to take place between, Huber (software TRANSINT), BIBER and LMS estimator (software TRINETH) employed in two typical cases of geodetic problems that are solved by transformation of coordinates (see also 4.2).

4.5.1 Example 1: Simulated Deformations in Lago Bianco

The example given in 4.2, concern one of the typical cases meet in Geodesy; Comparison of two geodetic networks, measured in different times when the number of points that might have moved could be relatively large (e.g. slide movements).

In Fig. 4.9 is simulated this kind of movement when errors of the size of 29 to 35 cm have been introduced in the global coordinates of points 10100, 10130, 10142, 10145. The total number of passpoints is 9 and the standard deviation of the points in the local and global system of coordinates is assumed as 2.5 mm.

The results of the adjustment by three methods employed show the following:

LMS - estimator

Robust adjustment with LMS (Least Median of Squares)

s_o = 0.9143

Standard deviation (global, local) [mm]:	2.5	2.5
Origin for the global system [m]:	700000.000	130000.000
Origin for the local system [m]:	700000.000	130000.000

Transformation Parameters

Number	4		
Y_o *[m]:*	0.117	σ_{Y_o} *[mm]:*	****[2)]
X_o *[m]:*	3.238	σ_{X_o} *[mm]:*	****

[2)] the symbol **** indicates that the matrix of cofactors for the unknown parameters is not calculated when the LMS estimator is applied (see also 4.4.2)

Rotation ω [gon]:	0.0019931	σ_ω *[cc]:*	****
Scaling factor m [mm/km]:	-5.3259	σ_m *[mm/km]:*	****

Number of passpoints: 9

Number of iterations: 1

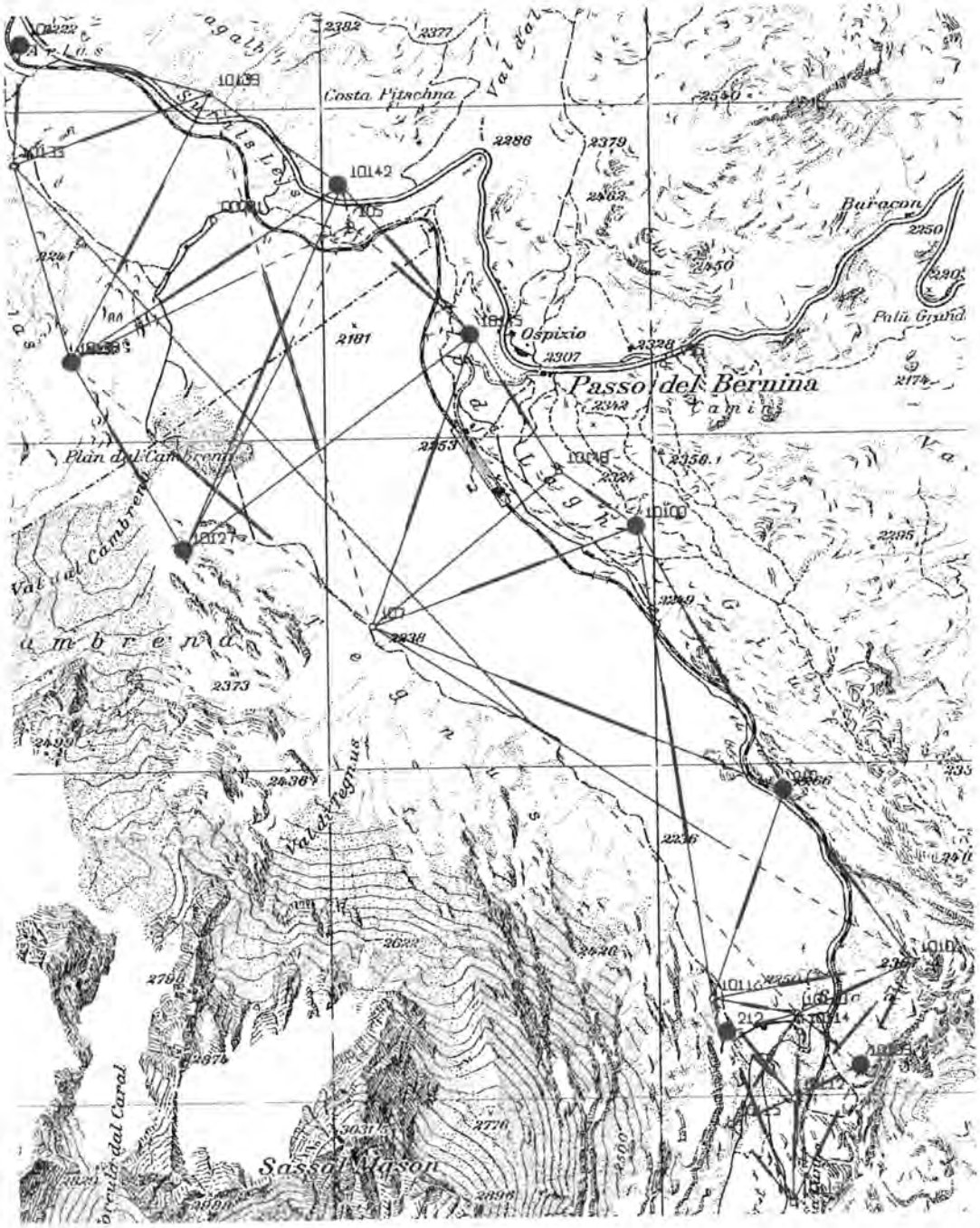


Fig. 4.9 Geodetic network around the Lago Bianco (example 1)

Passpoints and Residuals

PASSPOINT	Y_{loc} [m]	X_{loc} [m]	Y_{gl} [m]	X_{gl} [m]	r_y [mm]	r_x [mm]	outliers
10100	798950.728	142727.984	798950.403	142727.706	312.5	349.3	*
10109	799613.000	141092.504	799612.933	141092.561	.1	3.2	
10127	797570.308	142667.702	797570.299	142667.820	2.4	-2.4	
10130	797231.171	143246.832	797230.885	143246.664	299.2	291.1	*
10142	798061.867	143767.554	798061.650	143767.307	242.7	340.9	*
10145	798455.145	143309.890	798454.806	143309.653	348.1	321.1	*
104	797093.038	144211.252	797093.085	144211.378	-2.6	-3.6	
210	799389.186	141921.017	799389.145	141921.081	1.8	-1.1	
212	799212.630	141194.572	799212.571	141194.647	-2.0	-3.5	

BIBER - estimator

Robust Transformation with $c = 3$

s_o = 1.2035

Standard deviation (global, local) [mm]: 2.5 2.5

Origin for the global system [m]: 700000.000 130000.000
Origin for the local system [m]: 700000.000 130000.000

Transformation Parameters

Number: 4

Y_o *[m]:*	6.781	σ_{Y_o} *[mm]:*	218.16
X_o *[m]:*	-12.664	σ_{X_o} *[mm]:*	225.42
Rotation ω *[gon]:*	-0.0085570	σ_ω *[cc]:*	1.45
Scaling factor m [mm/km]:	-53.0189	σ_m *[mm/km]:*	2.17

Number of passpoints: 9

Number of iterations: 2

Passpoints and Residuals

PASSPOINT	Y_{loc} [m]	X_{loc} [m]	Y_{gl} [m]	X_{gl} [m]	r_y [mm]	r_x [mm]	outliers
10100	798950.728	142727.984	798950.403	142727.706	68.2	227.9	*
10109	799613.000	141092.504	799612.933	141092.561	-5.4	70.8	
10127	797570.308	142667.702	797570.299	142667.820	-165.0	-349.7	
10130	797231.171	143246.832	797230.885	143246.664	52.3	-140.5	*
10142	798061.867	143767.554	798061.650	143767.307	-130.8.	21.8	*
10145	798455.145	143309.890	798454.806	143309.653	31.3	89.3	*
104	797093.038	144211.252	797093.085	144211.378	-402.7	-504.8	
210	799389.186	141921.017	799389.145	141921.081	-130.1	-10.8	
212	799212.630	141194.572	799212.571	141194.647	-4.9	-7.2	

Huber - estimator

The same example has been calculated also with the Huber estimator (software TRANSINT). The algorithm does not converge.

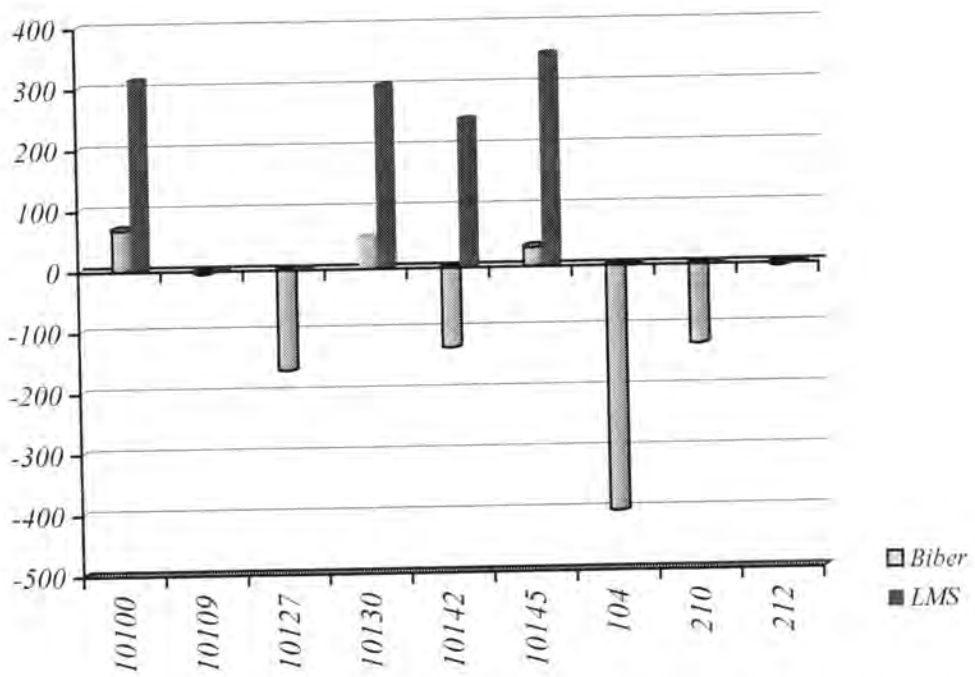


Diagram 4.1 Comparison of resultant residuals in y determined by the BIBER and LMS estimators (residuals are in mm)

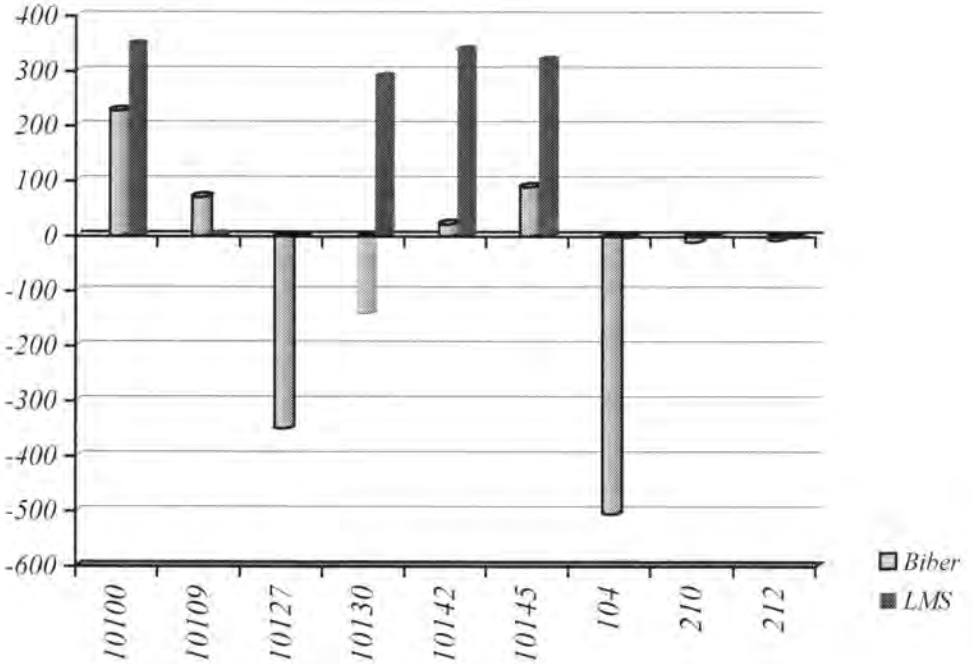


Diagram 4.2 Comparison of resultant residuals in x determined by the BIBER and LMS estimators (residuals are in mm)

After the comparison of the results of the three different calculations, and consideration of the characteristics of the example (see also 4.2, example 1) the following can be stated:

In cases where the number of outliers is high (almost 40%) the LMS estimator is the only estimator that can spot the outliers. The residuals (by LMS estimator) of the points 10100, 10130, 10142 and 10145 are almost equal to the errors introduced. The results of the Huber (algorithm does not converge) and BIBER estimators are useless, because, in this example, the breakdown point of the estimator is one of the most important aspects. In situations similar to this one, therefore, the LMS estimator can be recommended for the outlier detection (as a first step of the parameter determination). The detected outliers can then be removed, and the final adjustment can follow by the LS estimator.

Below shall be given the displacement vectors for all points calculated by LMS and BIBER estimator.

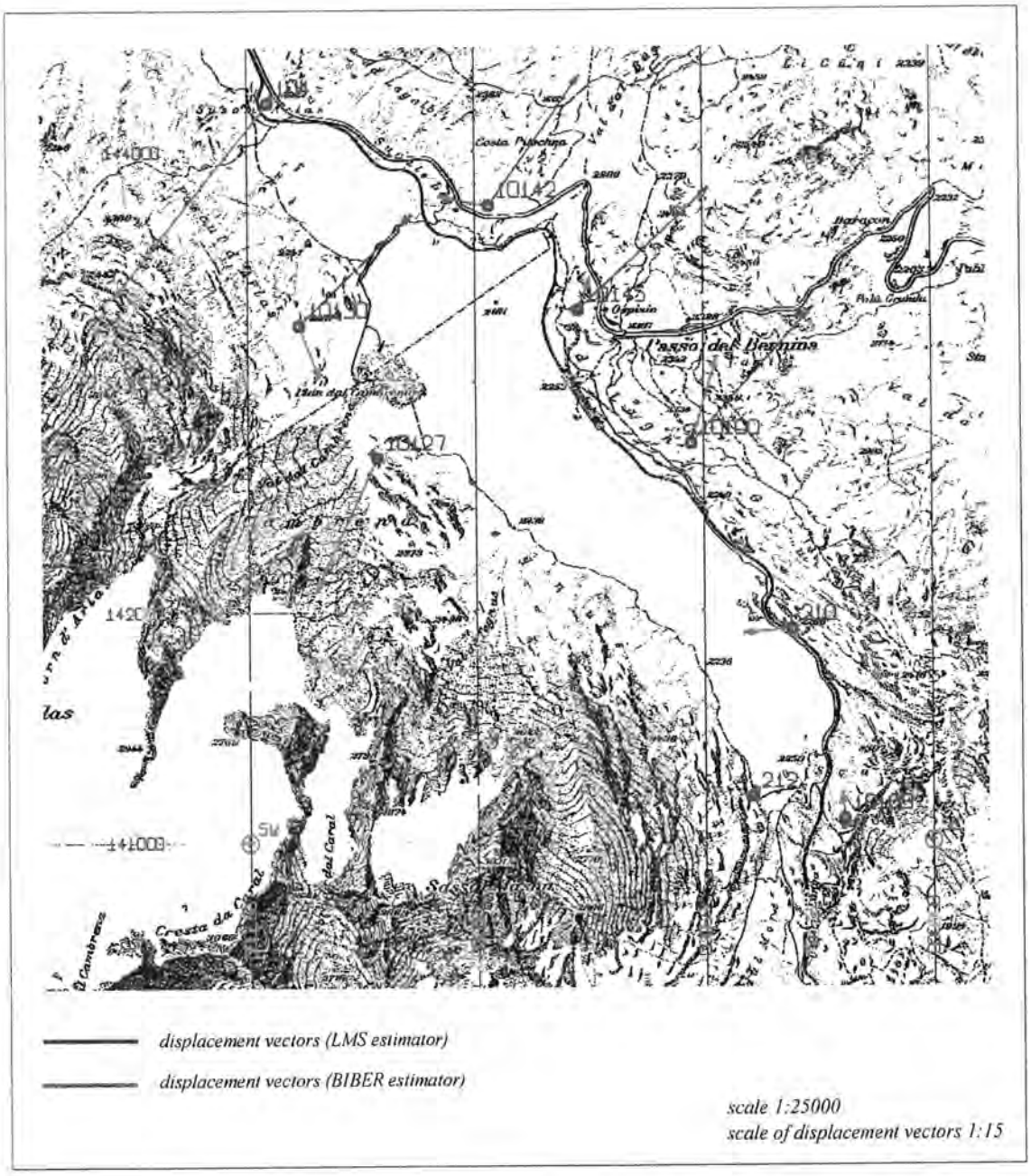


Fig. 4.10 Comparison of the displacement vectors calculated by the LMS and BIBER estimators (example 1)

4.5.2 Example 2: Simulated Deformations in Lago Bianco

In this example the number of passpoints is smaller and errors have been introduced only in the point 10109 (16 cm in y and x). The most important characteristic of the example is that the geometry is not very favourable, since the point 10109 is far from the rest of the others. The results of transformation with different estimators are presented below:

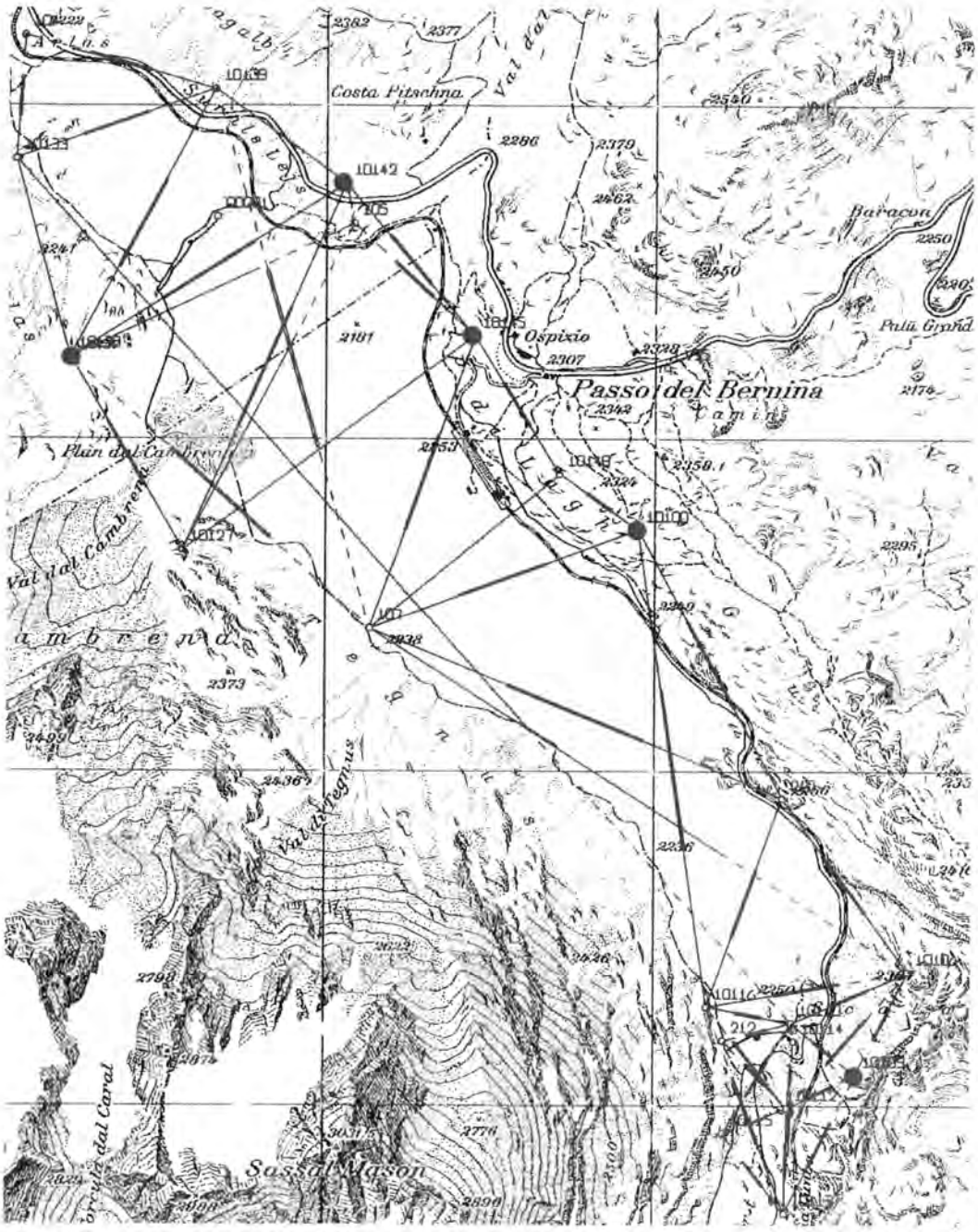


Fig. 4.11 Geodetic network around the Lago Bianco (example 2)

LMS - estimator

Robust adjustment with LMS (Least Median of Squares)

$s_0 = 1.4690$

Standard deviation (global, local) [mm]: 2.5 2.5

Origin for the global system [m]: 700000.000 130000.000
Origin for the local system [m]: 700000.000 130000.000

Transformation Parameters

Number	4		
Y_0 *[m]:*	0.772	σ_{Y_0} *[mm]:*	****
X_0 *[m]:*	4.036	σ_{X_0} *[mm]:*	****
Rotation ω *[gon]:*	0.0024443	σ_ω *[cc]:*	****
Scaling factor m [mm/km]:	-12.9281	σ_m *[mm/km]:*	****

Number of passpoints: 5

Number of iterations: 2

Passpoints and Residuals

PASSPOINT	Y_{loc} [m]	X_{loc} [m]	Y_{gl} [m]	X_{gl} [m]	r_y [mm]	r_x [mm]	outliers
10100	798950.728	142727.984	798950.713	142728.056	-3.9	-0.4	
10109	799613.000	141092.504	799612.773	141092.401	137.1	171.2	*
10130	797231.171	143246.832	797231.185	143246.964	9.6	-0.4	
10142	798061.867	143767.554	798061.900	143767.647	0.4	-0.5	
10145	798455.145	143309.890	798455.156	143309.973	-0.4	0.5	

BIBER - estimator:

Robust adjustment with $c = 3$

$s_0 = 1.1292$

Standard deviation (global, local) [mm]:	2.5	2.5
Origin for the global system [m]:	700000.000	130000.000
Origin for the local system [m]:	700000.000	130000.000

Transformation Parameter

Number	4		
Y_0 *[m]:*	-0.146	σ_{Y_0} *[mm]:*	264.12
X_0 *[m]:*	4.778	σ_{X_0} *[mm]:*	264.96
Rotation ω *[gon]:*	0.0029966	σ_ω *[cc]:*	1.72
Scaling factor m [mm/km]:	-4.7708	σ_m *[mm/km]:*	2.64

Number of passpoints: 5

Number of iterations: 1

Passpoints and Residuals

PASSPOINT	Y_{loc} [m]	X_{loc} [m]	Y_{gl} [m]	X_{gl} [m]	r_y [mm]	r_x [mm]	outliers
10100	798950.728	142727.984	798950.713	142728.056	-4.9	-13.3	
10109	799613.000	141092.504	799612.773	141092.401	127.3	139.2	*
10130	797231.171	143246.832	797231.185	143246.964	-0.9	5.9	
10142	798061.867	143767.554	798061.900	143767.647	1.2	2.9	
10145	798455.145	143309.890	798455.156	143309.973	-0.4	-3.3	

Huber - estimator

The same example has been calculated also with the Huber estimator (software TRANSINT). The algorithm does not converge as in the example 1.

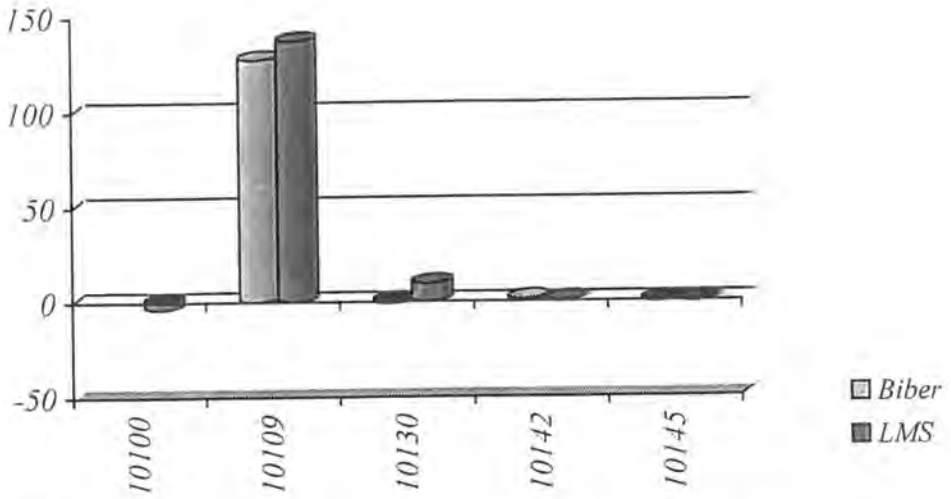


Diagram 4.3 Comparison of resultant residuals (example 2) in y determined by the BIBER and LMS estimators (residuals are in mm)

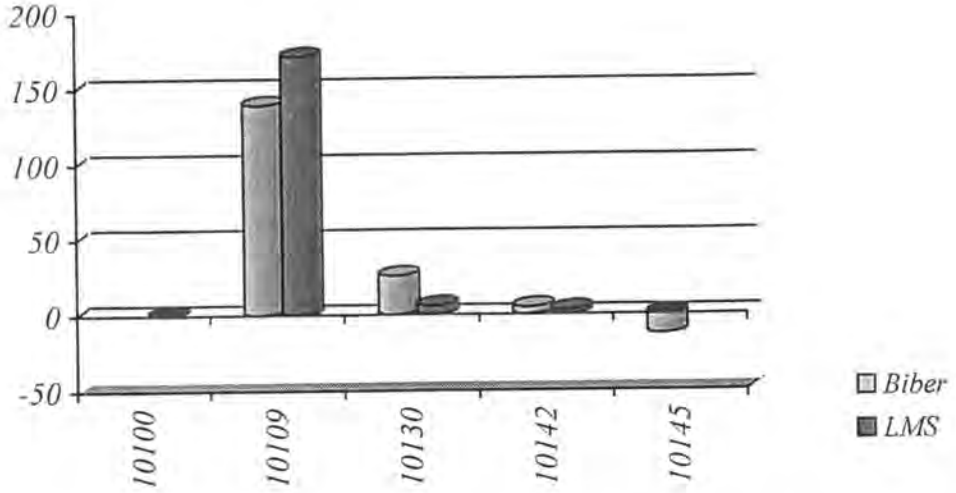


Diagram 4.4 Comparison of resultant residuals (example 2) in x determined by the BIBER and LMS estimators (residuals are in mm)

In this example, the geometry of the network, rather than the number of outliers is the important characteristic. There is a single outlier, placed in an unfavourable position. Upon comparing the results of the adjustment, it can be noticed that the algorithm of the Huber estimator does not converge (software TRANSINT), whereas the LMS and BIBER estimators deliver the correct results. Since in this case (where the number of outliers is small) the LMS estimator is not necessarily required, it is recommended to

employ the BIBER estimator since its asymptotic efficiency is high. This estimator limits the influence of the outliers and maintains the properties of the LS estimator. The displacement vectors for the three calculations are given in Fig. 4.12.

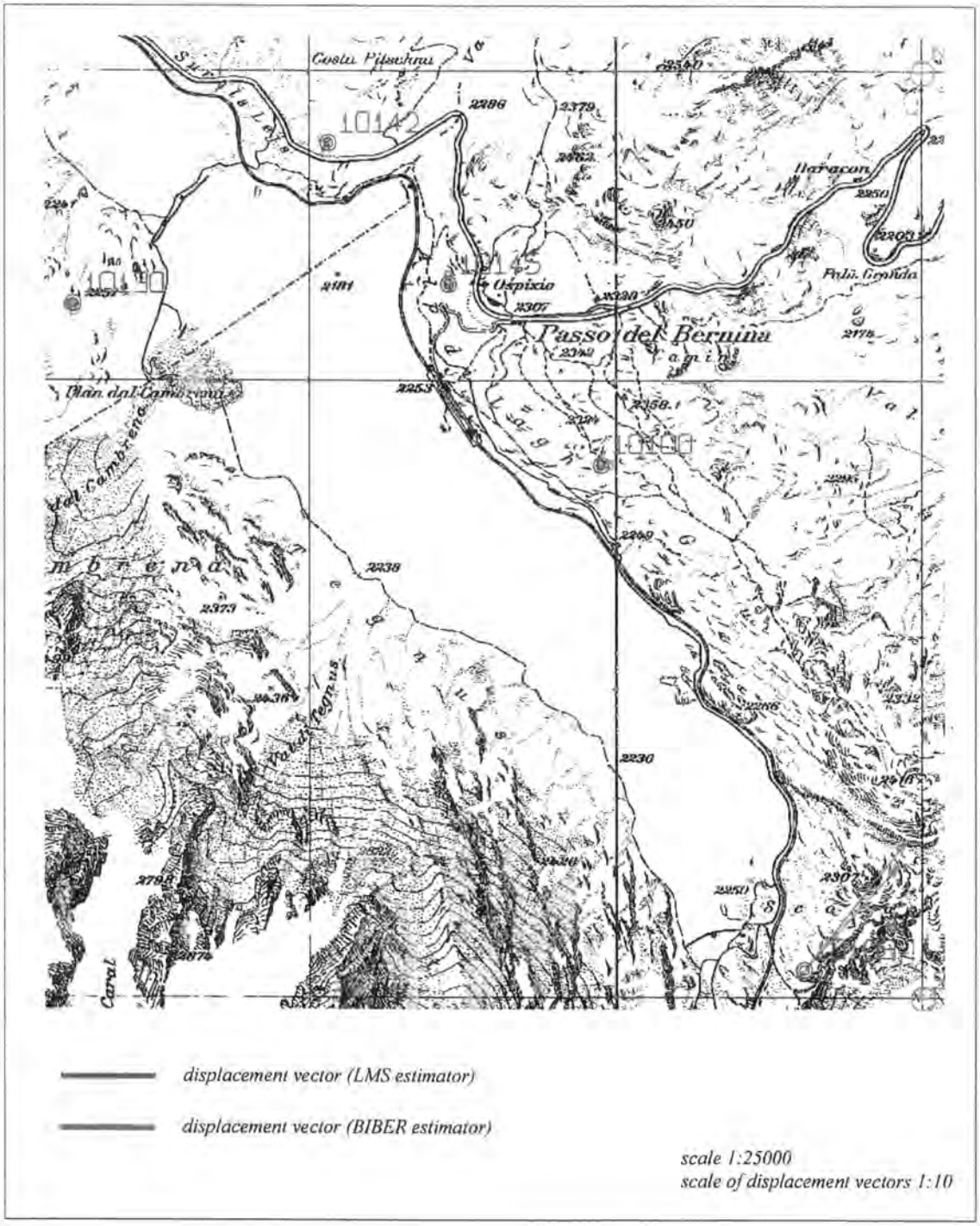


Fig. 4.12 Comparison of the displacement vectors calculated by the LMS and BIBER estimators (example 2)

Chapter 5

Robust Estimators in the Automatic Structuring of Raster Data

5.1 Introduction

Over several years, the demand for digital data has grown concurrently with the implementation of Geographic Information Systems (GIS). To fully exploit the capabilities of GIS, accurate and up-to-date information must be provided. This information can be obtained from various sources such as: existing numerical data (point coordinates, statistical data, databanks), graphical information (topographic maps, maps of cadastral surveying, thematic maps, etc.), geodetic works (measurements, photogrammetry, satellite data, etc.) and other sources such as from questionnaires.

In recent years, many countries have paid much attention to the ways and means by which geographic information may be made available for a wide range of applications. A testament to this is the advent of national databanks in Germany (ATKIS) [Jäger, 1997] and Switzerland (GETDB) [Sonder and Jäger, 1998] where the digital topographic map is the basic data source. These maps contain a wealth of information

which can be used in many GIS applications. The automatic extraction and structuring of geoinformation from digital topographic maps is a topic of considerable current interest in cartography. A human being has an impressive ability to provide high quality interpretation of noisy or corrupted data. However, as this process is slow and laborious, there is a growing need for new algorithms and methods which allow the extraction and structuring of cartographic symbology, as well as areal and linear objects, in a quick and effective way.

One of the research objectives of this work is to structure the two dimensional extracted information (e.g. buildings) from topographic or general maps automatically and show how the robust estimators can be in use of such purposes. Due to map degradation or difficulties during the extracting process, the true shape of a building can be misrepresented. Falsely assigned parts of areal objects (e.g. extension of railways, roads or grid lines of the map), rounded shaped corners and geometrical constraints not maintained, such as perpendicularity and parallelism, are common. Because we are concerned with an automated process where a huge amount of data has to be processed, the robust estimators techniques could be of a great benefit in the evaluation process.

Therefore, in this chapter the transformation process of areal objects from raster to vector data, and the developed algorithms concerning the robust estimators employed in this process, shall be presented.

5.2 Topographic Map

Major references go to Hake [1975, 1994] and Hake and Grünreich [1994].

5.2.1 The Content of the Map

The topographic map consists of a representation of the situation, relief and font. *The situation* denotes all two dimensional elements of the map apart from the relief. The situation includes urban areas and separate buildings, roads and railway networks, territorial waters, land cover and single objects [Hake, 1975; 1994]. We are concerned principally with the acquisition and evaluation process of all two dimensional elements

of the map and especially of urban areas and separate buildings. The representation of relief will not be treated in this work.

The representation of buildings and building blocks in the topographic map can be classified into one of two groups [Hake, 1975; Spiess, 1979]: (a) *true contour representation* and (b) *approximated contour representation*. In the first group (a) buildings are given as objects with a scale, i.e. their original size with all details can be fully reproduced. The minimal dimension of 0.3mm for the smallest possible object presented in the map represents, in reality, 1.5m for a map of scale 1:5000. The scale of 1:10000 is normally the limit for a true contour representation. Here the minimal object dimension which can be represented is 3m. The other group (b) applies to topographic maps of smaller scale where buildings are represented by an approximated contour. Details of their representation are omitted and the process of cartographic generalisation is applied.

"Generalisation can be defined as the process of reducing the amount of detail so that the character or essence of the original features is retained at successively smaller scales" [Robinson, 1995]. This process involves omission, aggregation, simplification, displacement, exaggeration and symbolisation of individual feature or groups of features with consideration to the following rules: the minimal building, minimal distance between houses, building orientation and the typical distribution of the original houses. Details of single buildings (e.g. new buildings, court yards, etc.) are simplified and small buildings eliminated. Whenever it is necessary to maintain some particular features of the house then *dimension exaggeration* comes into play. As the scale becomes smaller the degree of symbolisation becomes bigger (Fig 5.1).

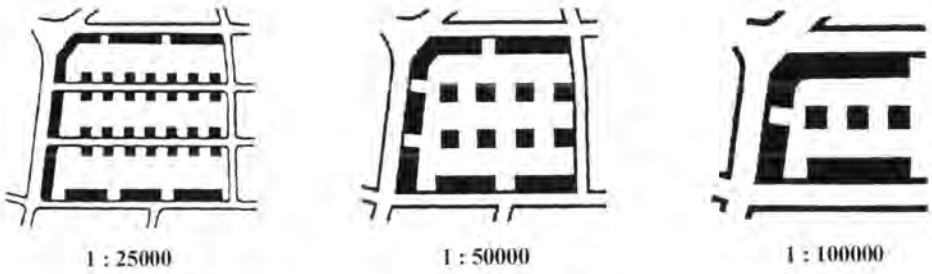


Fig. 5.1 Street and house generalisation [Morehouse, 1995]

For a map of scale 1:25000 the buildings are represented as blocks in the part of the city where they are densely spaced (Fig. 5.2).

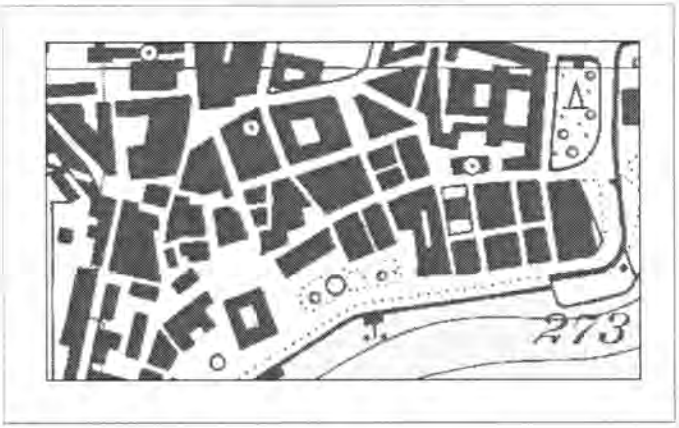


Fig. 5.2 Representation of building blocks (scale 1:25000)

For topographic maps of scale between 1:300000 and 1:500000 the inner structure of a city is represented only by a contour around the built up area. For smaller scales, the contour representation is substituted by a signature which can vary in size, form, type of filling and sometimes are accompanied with useful information such as the number of inhabitants.

Road and railway networks: A true contour representation of roads and railway networks is possible up to a scale of 1:5000. For maps of smaller scale the representation is constructed from signatures that classify the roads and railways by their importance and/or size. The generalisation process in the representations of roads includes omission of confluence and enlargement of the width. In topographic maps of scale 1:25000 the road widths are often larger than in reality, which affects the size of buildings that are placed by the side of such roads.

Territorial waters: Territorial waters signify surfaces, which are, covered with water either continuously or seasonally. Thin water surfaces are represented in the topographic maps by a single line. For maps of scale 1:5000 and smaller, a double line represents them. The generalisation process includes the omission of curves and narrow parts, which results in a shortening of their length.

Land cover: Land cover signifies gardens, forests, meadows, wine gardens, etc. They are represented as closed contour polygons of different signatures (e.g. rows of points) as long as this is possible. The type of land cover is denoted by colour. Sometimes the

signatures and colouring are combined but for smaller scales the coloured surfaces become increasingly important and the range of representation also diminishes.

Single objects denote the reminding local objects such as churches, monuments, petrol stations, triangulation points, view points, etc. Also line-like objects such as hydro-electric dams, walls or fences, etc. are classified under single objects.

5.2.2 Map Accuracy

The accuracy of the elements of the situation in the topographic maps depends on two factors: the standard deviation of a drawing point in the map and the map scale. The accepted standard deviation s is about 0.15mm which represents in reality 0.15 multiple the map scale. For a map scale of 1:5000 or 1:25000 these equate to 0.8 and 3.8m respectively [Hake and Grünreich, 1994]. Unfortunately, the actual accuracy is far greater than these values, especially as the map scale diminishes or the generalisation process is activated (the enlargement or the omission process acts negatively in the map accuracy). In the colour map production process, the accuracy cannot be maintained either, because of the difficulties encountered when the colour layers are joined.

5.3 From Analogue Copy to Numerical Data

Spatial information can be stored in three different ways: (a) raster structure, (b) vector structure and (c) hybrid structures. The method of transforming from analogue copy to numerical data depends on the required data structure:

If the required data structure is in *raster form (a)* then the analogue copy (e.g. topographic maps) may be converted by scanning technologies from analogue to digital form quickly and effectively [Eidenbenz, 1989]. The raster image can then be described numerically as an amount of quadratic picture elements (or pixels). Raster data are ideally suited for capturing, storing and presenting visual thematic information in analogue or digital form. Furthermore, scanning remains the most rapid and most automated acquisition process.

If the required data structure is *vector structure (b)* then one must differentiate between spaghetti data (*vectors* that are represented either by two points, or assigning a direction and magnitude from a single point) or vector data structured in thematic layers.

Although vectors are most appropriate for the representation of lines and line-like components, many authors have also used vectorisation for other types of components, primarily as an intermediate step in the process of converting to a more expressive representation or different analysis such as recognition of primitives. Vector data can be obtained by manual digitising or by automatic structuring of information [Stengele, 1995; Frischknecht 1996, 1999; Mayer, 1993; Frischknecht and Kanani, 1998]. Compared to raster data they are efficient and easy to work with and need less storage requirements.

Under the definition of hybrid data (c) comes a combination between raster and vector data within a data model. Up to now information systems contain separate raster and vector data. The first applications of hybrid data structures where the vector and raster data are still structured (saved) in separate files, are to be found in Nebiker [1997].

In this section only the vector data and the conversion of raster to vector data *(vectorisation process)* are presented. Detailed information about the two other types of data can be found in Hurni [1996] and Nebiker [1997].

The vectorisation is achieved through a manual or automated process or a combination of the two. The combined solutions are termed interactive, half automated or automated supported depending on the procedure.

5.3.1 Manual Vectorisation

Manual vectorisation involves ***manual digitalisation*** followed by the generation of lines and polygons. In the first stage, the local coordinates (x_{local}, y_{local}) of discrete points are saved in digital form (acquisition of the geometry). This is achieved by the registration of these points on the analogue copy which is attached to a digitising table. From these points, vectors may be generated either by specifying two points, or a combination of a single point, direction and magnitude. If the final goal is to create a vector-oriented GIS, more information concerning thematic, attributes etc. will be recorded too. The recorded local coordinates will then be transformed into a basic reference system. At this stage

extra conditions such as parallelism, perpendicularity, or existing measurements can be added. These procedures together create a homogenous vector-data set.

The primary disadvantages of manual digitising are that it is time consuming, monotonous and prone to human error. Furthermore, the cost of assuming an operator is much higher than buying the proper infrastructure. Nevertheless, as an old, traditional method of graphical image processing it is still useful in the following circumstances:

- when small areas are to be processed
- when the structure of elements is quite complicated
- when map degradation is very pronounced. In this instance it is very difficult to digitise the features automatically.

5.3.2 Automatic Vectorisation

Automatic vectorisation procedures in GIS attempt to convert **raster images** into a collection of vectors structured in thematic layers (see also 5.3).

The conversion from analogue - to - digital form is achieved by scanning the hard copy. The raster image can be described numerically as an amount of elementary objects which are called grid cells, raster cells or pixels. The amount of information transferred from the hard copy to the raster image is dependent upon the scanner resolution (the size of pixels), which is chosen with due consideration to the amount of data, the desired speed of the scanning process and the information contained in the analogue copy. The *geometrical resolution* is expressed either in terms of lines per millimetre (l/mm) or dots per inch (dpi). A typical scanner resolution in cartography is 400-1000 dpi and the human eye can recognise the quadrate structure of a raster image up to 800 dpi. *The quantitative resolution* is also a key characteristic of raster data. Under this term is to be understood the assignment of a colour value to a pixel. This is often called also *radiometric, thematic or colourimetric resolution.* Depending on the chosen depth of the colour (1 bit, 8 bits, 24 bits) one can distinguish between binary (1 bit), grey value (8 bits) and truecolour (24 bits) images.

For visualisation and extraction of information from raster images, a scanner resolution of 500 dpi suffices. These tests have been performed by cartographers taking due consideration of the minimal object dimension and minimal spaces between objects [Hake, 1994]. Various operations of image processing, such as contrast change,

filtering, geometrical equalisation can then be applied to improve the raster image. The automatic process should be able to interpret and extract information from the raster image with the same proficiency as in the manual digitalisation process. An important approach that allows an automatic data acquisition process is that which is based on *the pattern recognition techniques*. The entire automatic process consists of many steps together: scanning, pre-processing, image analysis through pattern recognition and post processing. This requires a high financial initial investment. Moreover, highly qualified personal are required to instigate the automatic process. The advantages of the automated process are listed below:

- automatic digitalisation allows a quick acquisition of unstructured raster data
- it is not prone to human error.
- it has been shown to result in a higher geometrical precision compared to manual methods

5.3.3 Combined Solution

Despite the tremendous advances which have occurred in the develop of both hardware and software in recent years a fully automated process that allows one to digitise (vectorise) the contents of an image does not exist; some degree of manual digitalisation is always required. The computer should perform the routine processes and the operator (manually) the processes which require a higher intelligence level. Two of the most popular approaches are those of *screen-digitising* and *CAD - overlay techniques*. The screen digitising technique proceeds as follows: Instead of manual digitising in the analogue copy, the digitising is performed in the raster image using a graphic workstation. CAD-overlay techniques go a step further: Here further operations are possible such as comparing vector data simultaneously with the raster image, moving the cursor coordinates over the desired area and increasing the size of pixels in order to get a better view and switching on and off coloured layers of the raster image. These operations are achieved through a flexible visualisation display-editor of vector and raster data. From the point of view of tiredness of the operator, the process is a clear advantage over the manual digitising where the operator has to move between the screen and analogue copy.

A further kind of automatisation is that where the operator is actively supported by the system. Special cartographic features are found and indicated from the system

automatically. The operator has only to say yes or no to the results of the automatic search and enter the coordinates.

Another "automated" process for the data acquisition is that of line following. The line following techniques are used for digitising lines such as stream networks or contour lines and is accomplished with a lot of hardware expenses.

5.4 Vectorisation of Topographic Features

5.4.1 Vectorising Lines

There are two primary methods for establishing a skeleton representation for vectorisation: thinning and contour following [Doermann, 1997].

Thinning is the most straightforward method of generating a set of features to be used to fit vectors. Connected components are reduced to a medial-axis representation by peeling off layers of pixels from the boundary points which have more than one neighbour. After the component is reduced to a single line of pixels, line-simplification, line-smoothing and line fitting algorithms can be used to produce vectors.

Contours: In these methods contours are extracted from the image and line segments are generated from the mid-points between pairs of parallel lines. Compared to other methods, this approach generates a more robust set of features and produces a more stable vector.

5.4.2 Vectorising Solid Shapes

Even though vector data are more suitable for the representation of line features, many authors have used them for representing structures such as solid shapes objects (separate buildings, urban blocks, filled regions such as meadows, gardens, forests, etc.). Nevertheless, vector data cannot be applied for structuring of the information in an entire image and are used mostly as an intermediate step for a further representation.

The most common method employed is that of contour following (see 5.4.1).

5.5 Pattern Recognition - based Vectorisation

5.5.1 Pattern Recognition Techniques - Overview

The main problem in cartographic pattern recognition remains the formulation of a definitive general solution in order to create complex map objects. Pattern recognition techniques can be classified under two general headings: (a) *vector-based approaches* and (b) *raster-based approaches*. Vector-based pattern recognition techniques (a) are described through the following steps: simple vectorisation of the raster image, segmentation and classification of the single objects through geometrical and topological features. The complex map objects are then created through compilation of single objects [Giebels and Weber, 1990; Brenecke, 1990; Yamada *et al.*, 1993; Ohlhof, 1992]. Pattern recognition approaches based on the original raster data (b) enable the extraction of cartographic features direct from the raster image. Although rare, they are appropriate for several applications of automatic image or map interpretation. Samet and Stoffer [1994] present a system which recognises cartographic symbology of separate map layers using the legend of the map for the classification of the geographic symbols. Brügelman [1996] shows an approach of raster-based extraction of hatched cartographic patterns from a cadastral map. Stengele [1995] and Frischknecht [1996, 1999] present a raster-based approach facilitated by template matching techniques and raster local operations for the extraction of cartographic symbols from scanned topographic maps of Switzerland.

Pattern recognition - based vectorisation signifies a raster to vector conversion process which is performed in two steps: first, the information of the raster image is extracted and saved in a separate layer by pattern recognition techniques; second, the structuring of the information of interest is performed through vectorisation. The features that can be vectorised are linear (coastlines, road networks, etc.) and areal objects (buildings).

In Stengele [1995], a comparison between vector- and raster-based pattern recognition approaches is presented with regard to the scale of the automatisation of the process and loss of information during the conversion process (for example, as the image degrades the line features became broken, therefore it is not possible to provide a continuos vector representation).

Because we are concerned with an automated process that enables a vectorisation of solid shape objects after they has been extracted from the raster image, in the following section, only the method and results of the extraction process of areal objects through pattern recognition are presented. The extraction of other topographic features (font, linear elements, single objects, etc.) can be found in Stengele [1995] and Frischknecht [1996, 1999].

5.5.2 Extraction of Areal Objects - Method and Results

Buildings and urban blocks are represented in topographic maps of scale 1:25000 as cartographic objects with scale (see 5.2.1). Stengele [1995] and Frischknecht [1996, 1999] present an approach based on the scanned Swiss topographic maps for the extraction of areal objects (e.g. buildings). The topographic map specifications are: Scale 1:25000, resolution 20 l/mm, 40 l/mm and 50 l/mm, standard raster map dimensions respectively 14'000/28'000/35'000 x 9'600/19'200/24'000 pixels which results in uncompressed sizes of 16/32/40 MB for individual thematic layers.

- ***Method***

The extraction of buildings can be obtained by a combination of different approaches: local raster operations, morphological operations and template matching. These approaches are explained in detail in Stengele [1995]. Here, only the principal of the extraction process, employing the over mentioned methods shall be presented.

Local raster image operations can be defined as a calculation of the raster image where an element (pixel) is calculated from the information of other elements found in the localities. The neighbourhood operations of thinning and expanding are defined as basics of raster local operations.

Thinning and expanding are used in the raster image in order to separate information of the same colour, e.g. lines (grid lines, roads or railway network) from solid shapes objects (buildings). A N4- or N8-neighbourhood pattern or structuring element can be used for this purpose. When thinning the foreground with the N4- or N8-neighbourhood pattern, the result will give only the objects which have dimensions larger than twice the size of the thinning element.

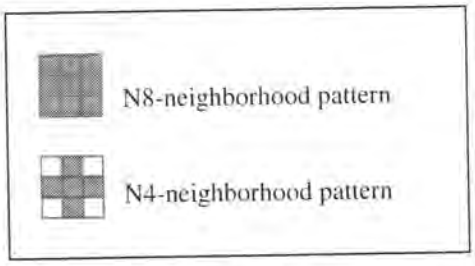


Fig. 5.3 Neighbourhood pattern used for local raster operations

By expanding these remaining objects by the same amount, the areal objects are generated again but separated from the line elements. Stengele [1995] has shown that in order to obtain good results, the two processes have to be repeated several times. The *thinning* amount has to be chosen carefully in order to remedy the deletion of areal objects smaller than twice the amount of the thinning element.

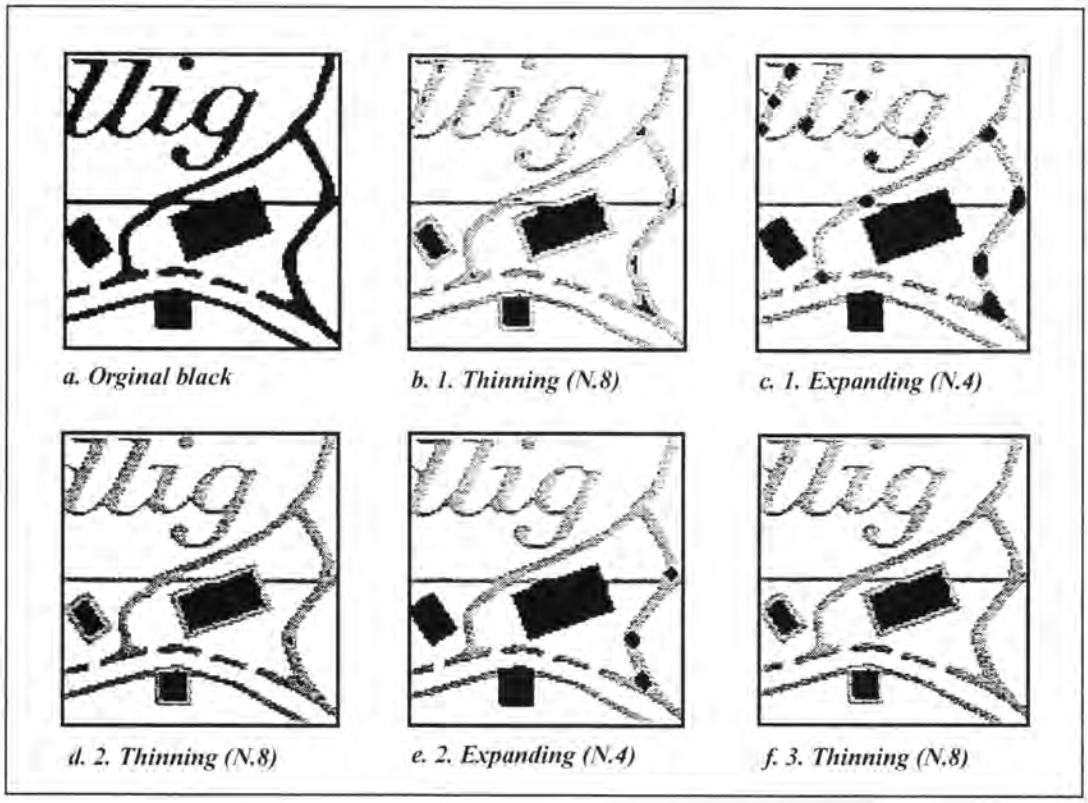


Fig. 5.4 Reduction of the falsely classified objects through flexible local raster operations [Stengele, 1995]

Furthermore, it has been shown that through a flexible thinning and expanding process the form of objects can also be improved [Stengele, 1995].

Morphological Operations. The most extensive class of binary image processing operations is sometimes described as morphological operations [Sierra, 1982; Coster and Chermant, 1985]. The principal idea is to compare the object or set under analysis with an object of known shape called a *structuring element*. This element, which is defined by the user, acts as a probe to investigate the relationships existing between the various parts of the studied object. A cross, a quadrate or another simple structure can serve as a structure element in the image data processing. The basic transformations of mathematical morphology are erosion and dilation. Because these two types of

trnsformation cause a reduction or increase in the size of regions, respectively, they are sometimes known as shrinking and expanding.

Morphological operations and local raster operations. In many cartographic applications, it has been shown that morphological operations can substitute the local raster operations. Indeed, Stengele [1995] has shown that if the structuring element of the morphological operations is the 3x3 square of nine pixels represented in Fig. 5.5 the outcome is the same as applying local raster operations like thinning and expanding.

Morphological operations and Template Matching. The comparison between two images is very often called *Image Matching*. *Template Matching* represents the process by which a synthetically generated pattern (template) is searched for in the raster image. The template is moved through the image, pixel by pixel, and the degree of similarity determines whether the pattern is accepted. In other words, template matching can also play the role of the structuring element in the morphological operations. For example in Fig. 5.5 is shown the template - structuring element for the erosion process for the extraction of buildings from the topographic maps. The template is defined with due consideration to the smallest object dimension of interest. This process is analogous to local raster operations where the procedures of thinning and expanding are performed. Morphological operations are generally preferred as they are less time consuming and more flexible.

- ***Results***

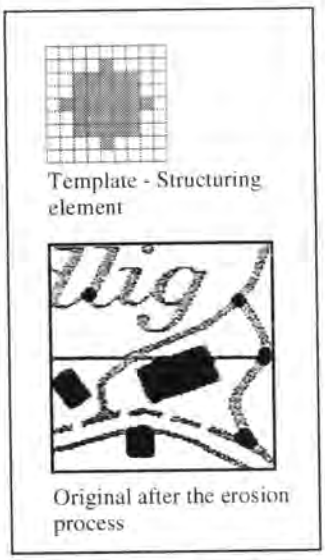


Fig. 5.5 Structuring element for the extraction of buildings [Stengele, 1995]

Through combinations of the methods explained above the extraction of buildings still requires some manual corrections. Two main types of errors are to be mentioned:

Error of first order - Small buildings and small parts of buildings are not recognised (Fig. 5.6 a); error of second order - solid shape map objects that are classified as buildings (Fig. 5.6 b,c). Stengele [1995] classified the errors of second order further. This new classification comes because 70% of the errors of second order are part of falsely recognised font. Therefore, the extraction of font before the extraction of buildings is highly advantageous. A further visual inspection of the results through a raster visualisation software RaVis [Zanini, 1999] is also recommended. So it is possible to remove almost completely the errors of second order.

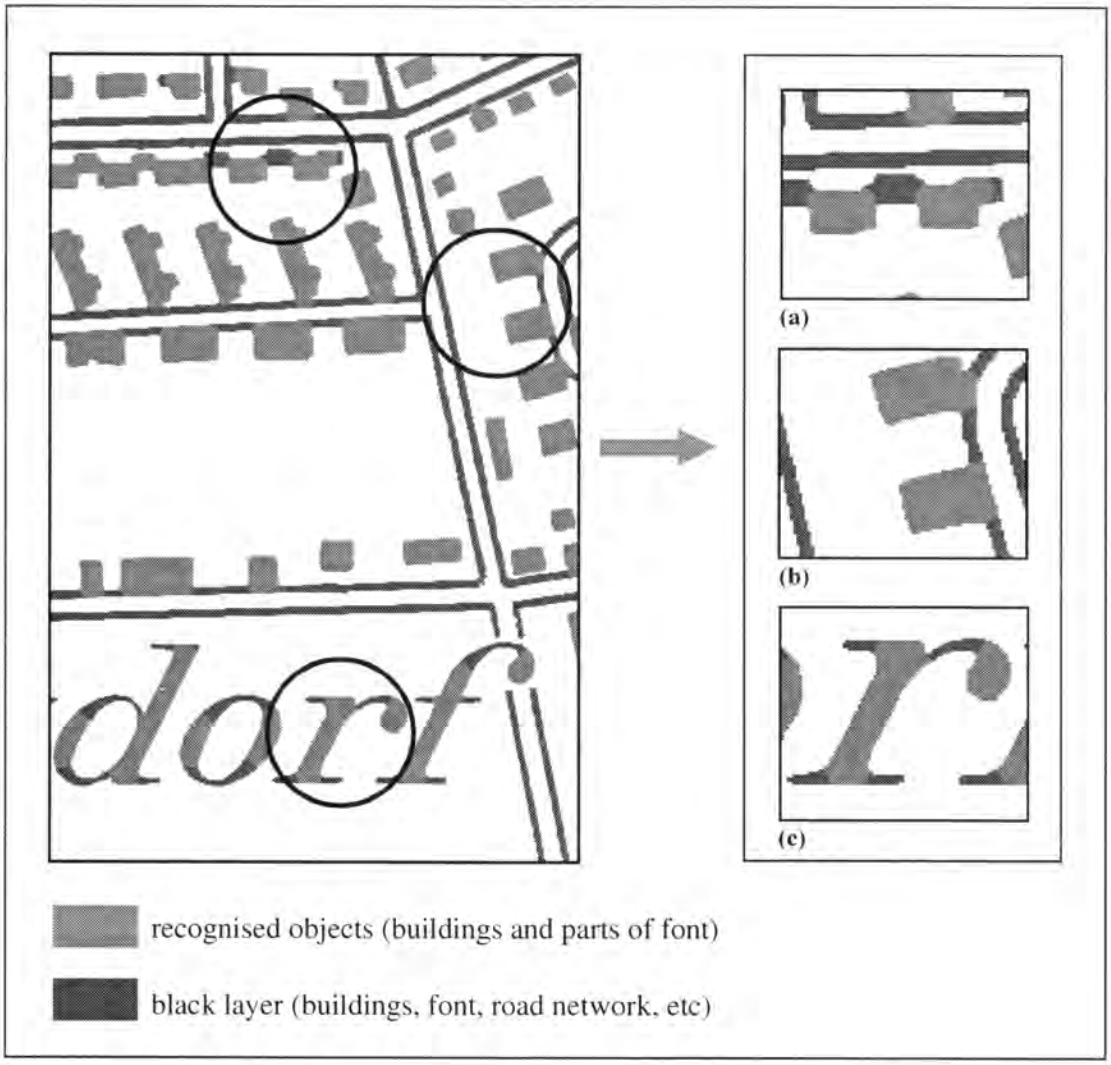


Fig. 5.6 Extracted buildings (Error of first a,b and second order c)

5.6 Vectorisation - Contour Extraction

As presented in 5.4.2 the method used for deriving a vector representation of solid shapes is that of contour following. Nebiker [1995] analysed the area loss during the raster - to - vector conversion by applying the standard contour following method. When small buildings are processed, the contour extraction method by joining the mid-points between pairs of parallel lines (formed by the quadratic structure of the raster image) causes a reduction of the component dimensions which could be of great importance in cases with limited scanner resolution Therefore he proposed a two step approach:
(1) determination of the exterior and interior borders (when present), (2) determination of the (raw) contour.

The resulting line is a closed polygon for each interior and exterior contour which connects the outside corners of the contour pixels (Fig. 5.7). Each joining point is recorded with two orthogonal coordinates (x, y). The bounding line approximates the areal object but it is inappropriate for further use. The number of points which comprise the contour line must be reduced to form a realistic abstraction of the object's shape.

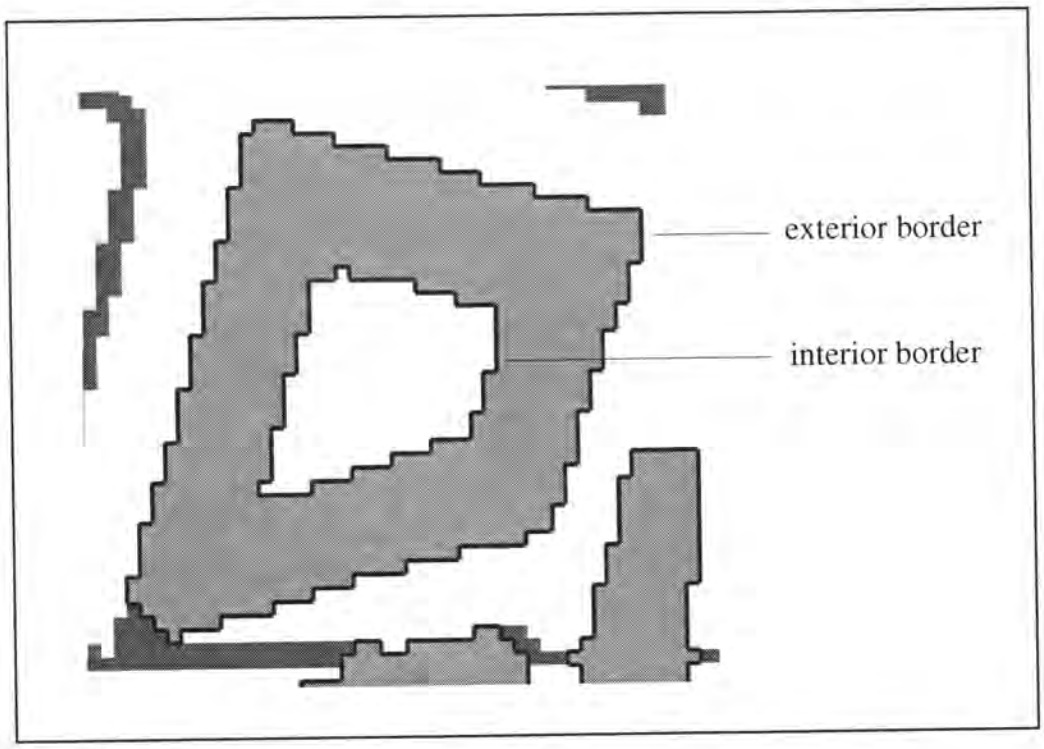


Fig. 5.7 Approximation of the contour line by contour following techniques

5.7 Simplification and Smoothing of Digitised Lines

Point reduction is applied to a map when there is a requirement to reduce the number of points necessary to represent the linear features of the map whilst retaining the basic shape or caricature of these features. Line smoothing is generally applied if it is necessary to improve the aesthetic quality of the map.

Many algorithms have been developed for the purpose of line reduction or smoothing. An overview of the algorithms developed for these two purposes can be found in Spiess [1992]. In Prashker [1991] a software utility is introduced that enables line simplification and smoothing of digitised lines using a broad range of algorithms.

The definition of a good vector representation is dependent upon the application (the nature of the line) and the method used for the data capture. In an automated process it is important to decide whether vectorisation should be performed to precisely represent the object or simply to approximate the original features. Therefore, some of the known algorithms for reduction and smoothing with their advantages and disadvantages are mentioned below.

5.7.1 Simplification of Digitised Lines

In general, all digitising methods record far more points than are necessary to reproduce a line. The elimination of superfluous data such as points along a straight line or in our case the points along the "raw" contour line can be of value for the following reasons:

- it allows us to describe the line with different degrees of abstraction, concentrating on the essential information for different users
- it enhances the possibility of visualisation
- diminishes the storage requirements
- removes unnecessary information and enhances the essential parts geometrically
- makes possible a faster vector processing and a faster vector to raster conversion

Many approaches have been suggested and algorithms written to reduce the number of points required to represent numerically recorded lines. These methods can be classified into two broad categories:

(a) elimination of points along the line according to various criteria

(b) deletion of non significant line characteristics

The automatisation of the last group of methods (b) relies principally on the ability to program the computer to recognise specific cartographic features such as a peninsula etc. and is definitely not suitable for our purpose. The extracted areal objects are approximated by a raw contour of pixels of the same type where no cartographic feature is to be recognised.

From the first category of methods, (a), which eliminate points, some approaches concentrate on the points which are to be deleted while the others are directed towards selecting those points which are to be maintained. The various algorithms which determine the critical points in a digitised line have been presented with different key

words: approximation of a polygon, piecewise linear approximation, determination of corners, data compression, selecting significant points on a line, etc.

These methods can function by establishing (i) a threshold distance, (ii) a threshold angle, (iii) both threshold angle and distance, (iv) polygon approximation, (v) elimination of points further than a tolerance distance, (vi) finding of corners as the curvature changes with the straight segment and *n* points, etc.

A full description of these algorithms is to be found in Spiess [1992]. Below is explained the Douglas and Peucker algorithm, since this method, adapted for a closed contour [Kanani, 1998], is employed for the determination of the approximated object corners of the areal objects (e.g. buildings).

Douglas and Peucker algorithm adapted for a closed contour

Douglas and Peucker [1973] base their idea on eliminating points, but concentrate on the selection of points rather than on their deletion. We have adapted this algorithm in closed contour [Kanani, 1998].

The procedure begins by defining the first point (in a closed contour we define this as the point farthest from the centre of gravity of the contour) as an anchor and the last point (the point farthest from the first one in the whole contour) as a floating point, Fig. 5.8 (a). These two points define a straight segment and separate the object's contour in two polygons, Fig. 5.8 (b). By defining the first and last point this way, it is always possible to determine two corners of the object immediately, since the contour has a very well defined shape (the shape of a building). Then, the procedure continues with the determination of the corners found into the two polygons. For each polygon, the perpendicular distance from the intervening points is calculated to establish the points with the greatest distance. If this distance is less than the maximum tolerance distance established in advance then the segment is the best line to represent our polygon, otherwise the farthest point will be the new floating point, Fig. 5.8 (c). The cycle is repeated, Fig. 5.8 (d) until no more floating points are found. The anchor then takes the place of the floater and the last point of the polygon becomes the new floating point. Upon completion of the final operation, the contour of the polygon is established.

Finally the approximated object's contour is obtained by joining the anchor points found for the two polygons, Fig. 5.8 (f).

Determination of the tolerance parameters at the Douglas-Peucker algorithm

Line simplification algorithms generally require the specification of one or more tolerance parameters in order to perform the process. These parameters may specify, for example, the distance from a previous point or an angle between the segment between two points and the previous line direction.

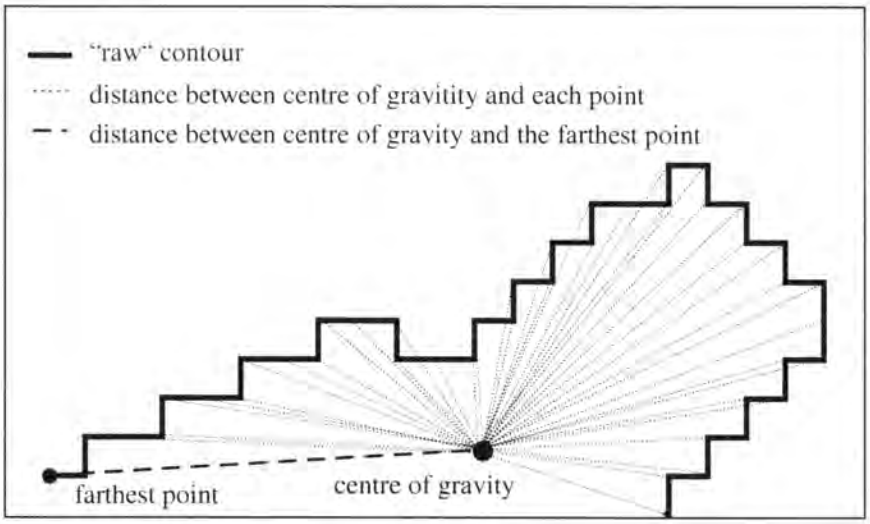


Fig. 5.8 (a) Determination of the first starting point in the contour

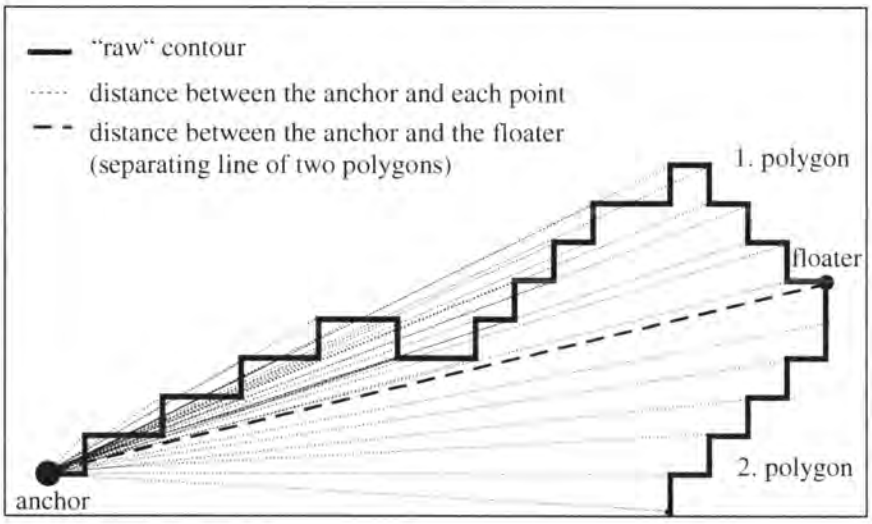


Fig. 5.8 (b) Determination of the first floater - Separation of the line in two polygons

For the Douglas-Peucker algorithm, the tolerance parameter is the distance between the intervening points and the segment anchor-floater (see Fig. 5.8c).

The determination of the tolerance parameters is dependent upon (a) the type of application, (b) the map scale, (c) the scanning resolution, (d) the quality of the analogue map and (e) the quality of the extraction process.

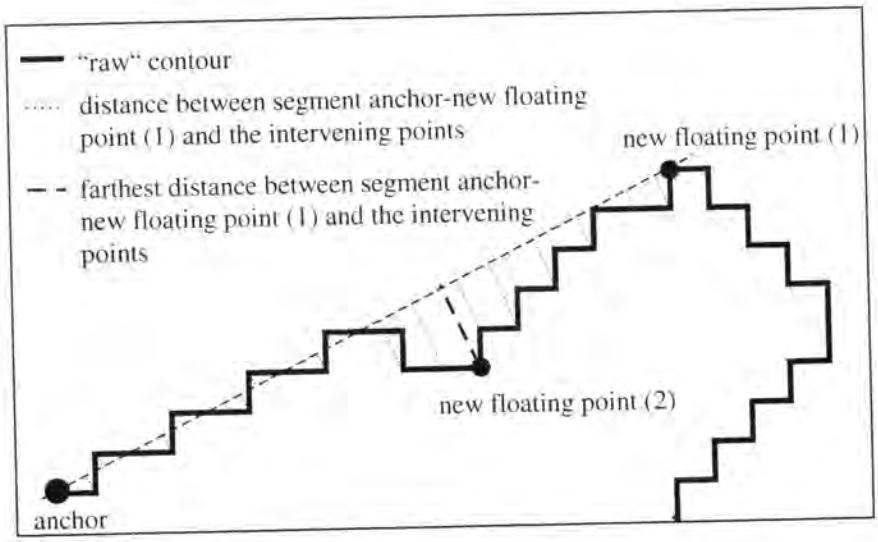


Fig. 5.8 (c) ***Determination of the new floating point***

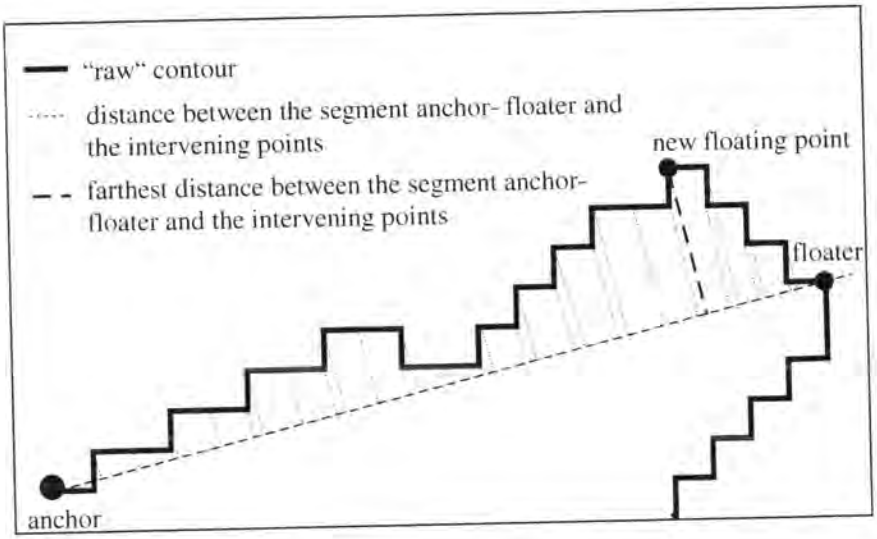


Fig. 5.8 (d) ***Repetition of the cycle Fig. 5.8 (c) until no more floating points are found***

All these factors are strongly interdependent, therefore it is very difficult to judge whether one is more important than the other. Through the following examples it will be possible to understand how these factors are inter-related.

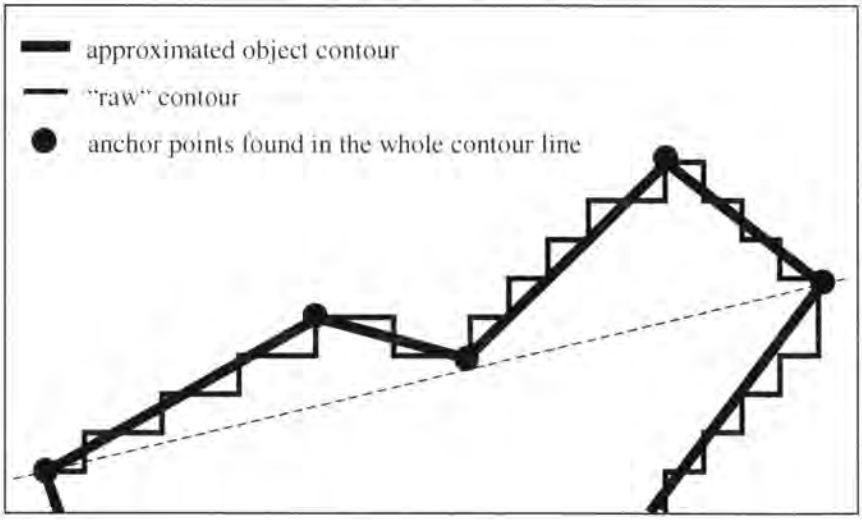


Fig. 5.8 (f) Approximated object contour - joining of the anchor points found for the two polygons

Without knowing the type of application (a) for which the point reduction is required, it is very difficult to estimate the tolerance distance. For example, if only an abstract form of the areal object is required, a rough tolerance distance, set just by eye, is enough whereas if we want to reproduce their form for an application such as a future replacement of the buildings in the topographic map, a better determination should be done. For such an application, the scanning resolution should also be sufficient and the quality of the extraction process very good.

The map scale (b) and the scanning resolution (c) are also strongly inter-dependent. This is because the amount of information that we want to leave out or extract depends upon both these factors. The amount of information in reality that corresponds to a pixel size in the map is calculated by the map scale and the scanning resolution and is given by,

$$amount\ of\ information\ in\ reality\ per\ pixel = \frac{map\ \ scale}{scanning\ \ resolution}$$

It is therefore possible for each application to decide the limit of the tolerance parameter with the help of this indicator. The quality of the analogue copy (d) clearly influences

the extent of point reduction. If the quality is poor then the scanner resolution should be high and the extraction procedure (e) of a high quality in order that a further abstraction can be representative.

By considering all the above mentioned factors, the decision of how much to set the threshold distance, in order to have a good representation of the areal objects, is determined by classifying the information to be maintained (real edges or corners of the buildings, Fig. 5.9, a) and the information to be omitted (falsely assigned edges or corners which may arise due to difficulties in the extraction process, Fig. 5.9, b). Nevertheless, the value of the tolerance distance can vary from case to case, because the practical separation between the information to be maintained and the information to be left out is very difficult to judge and in many cases a compromise value is accepted.

Example: at a scanner resolution of 20 l/mm and map scale 1:25000 a pixel represents in reality 1.25m, therefore a value of one and a half to twice the size of a pixel is acceptable as a threshold distance.

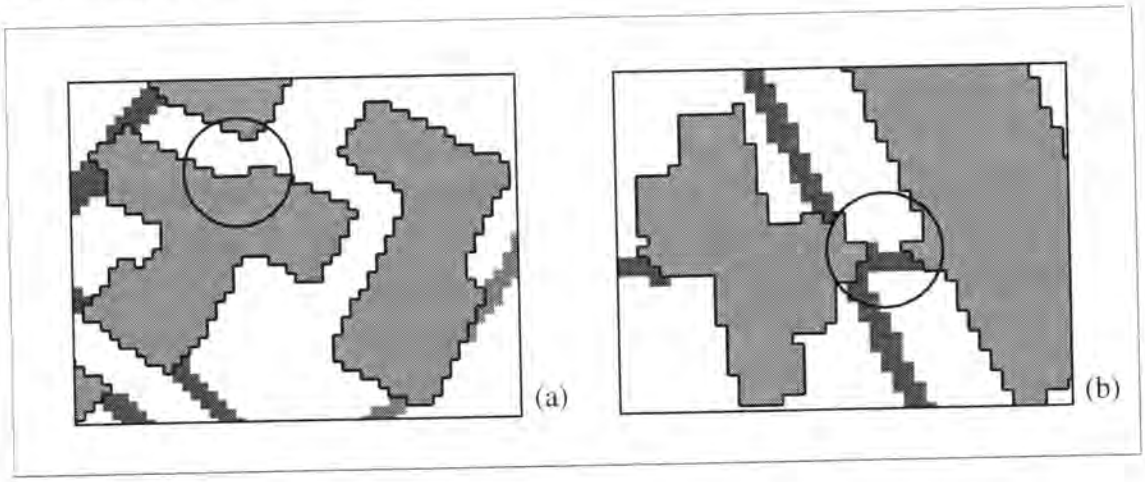


Fig. 5.9 (a,b) Real edges or corners to be maintained (a) and falsely assigned parts to be left out (b)

5.7.2 Smoothing of Digitised Lines

The resultant line after the reduction process represent the building approximately (Fig. 5.10). Further techniques are needed to derive a proper shape of the areal objects. One of these possible techniques is line smoothing that is used when the digitised lines need to be improved aesthetically or angularities from the digitising process need to be removed.

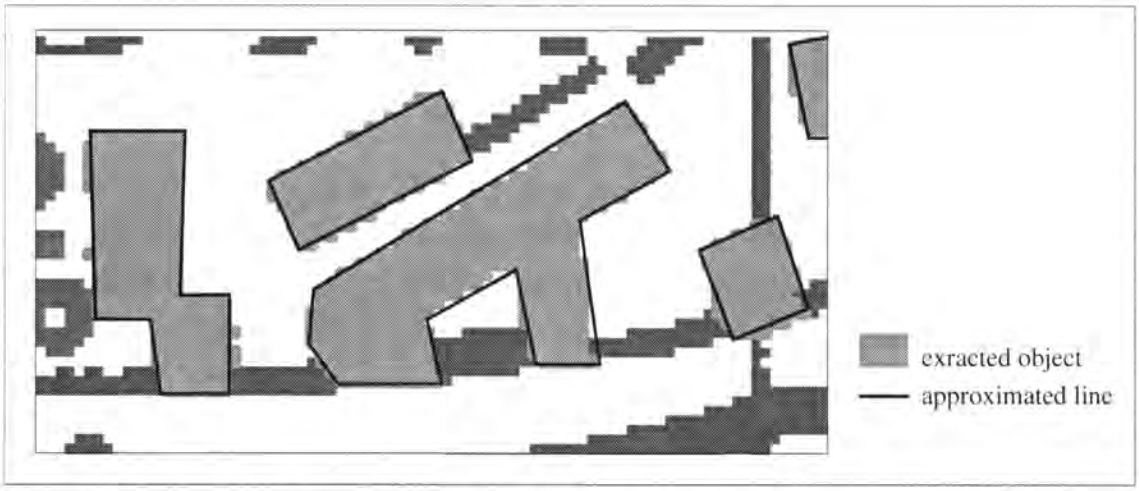


Fig. 5.10 Resultant line after applying the Douglas-Peucker algorithm

There are two methods of line smoothing: (a) applying algorithms to approximate the line and (b) approximation of the line with a mathematical function or interpolations algorithms maintaining the original points.

From the first group (a) the smoothing algorithm based on the *weighted mean* is of particular relevance [Tobler 1966].

The second group (b) seeks to approximate the points of the line with *a mathematical function*. This can be done for the whole line at once or in parts taking a small number of connecting points at a time. A method that falls into this category has been developed by A. R. Boyle [1970]. The algorithm fits a Least Squares line to a given number of points and then steps forward in that direction by a predetermined distance. This method produces lines which are much smoother than the lines they represent but requires a lot of calculation time. Nebiker and Carosio [1994] have proposed a slightly different method. Instead of adopting a Least Squares fit they propose a robust approach - the least absolute deviation estimator (L_1-estimator). The advantage over Least Squares procedure is the possibility to process more data points at once (*ca.* 20% more than the current number of points) without ruining the results of the parameter estimation. Furthermore, due to its robust behaviour, it is possible to detect corners in the raw data set, Fig. 5.11.

Disadvantages remain the minimum number of starting points required and the calculation time.

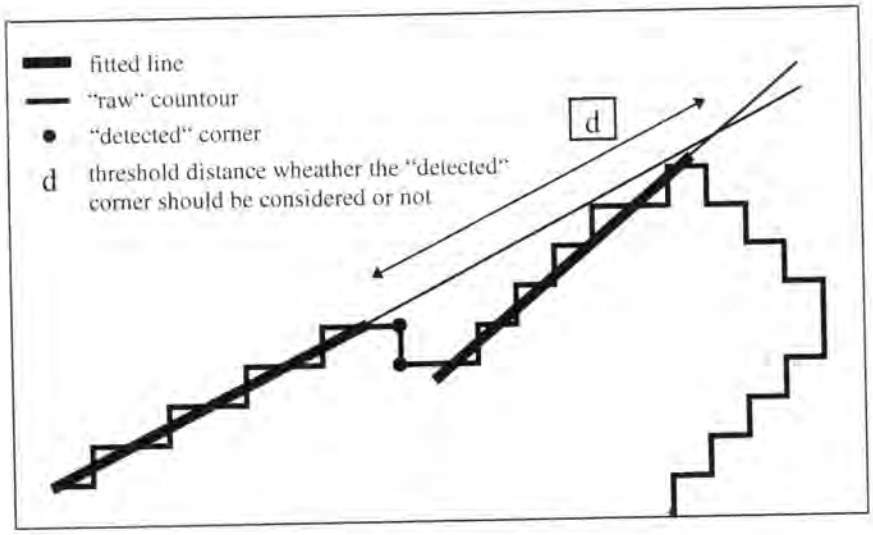


Fig. 5.11 Line reduction by least absolute deviation estimator

Furthermore, extra geometrical conditions such as parallelism and perpendicularity should be implemented in order to improve or reproduce the true shape of buildings.

5.8 Adjustment of Approximated Contour – New Approach

When an operator makes the vectorisation of the areal object by digitising, it is obvious that the resulting line will pass through most of the pixels and also maintain geometric constrains such as parallelism and perpendicularity. After the implementation of the Douglas-Peucker algorithm for reduction of the superfluous points, the resultant line is an approximated contour (closed polygon) defined only by the coordinates of the corners of the object, Fig. 5.10. The rest of the points in the extracted contour (the coordinates of which are also known) have up to now, not been employed in the determination of the vector line.

An *adjustment process* is always possible if the number of observations is larger than the number of parameters to be determined. By inspection of Fig. 5.10, we note that the number of observations needed for the determination of the object's contour is equal to the number of object's corners determined by the Douglas-Peucker algorithm multiple two (every point is given by two orthogonal coordinates). By taking into consideration the other points (observations) found in the raw contour, the adjustment condition is met.

5.8.1 Mathematical Model in the Adjustment of Areal Objects

Observations and unknown parameters

To determine the unknown parameters and observations of the adjustment model, we refer to Fig. 5.12, where a building with its approximated contour and corners is presented. We denote by,

n	the total number of the observations;
n_j	the number of points (pixels) in one segment j, j+1;
p	the number of the object's corners, the number of unknown parameters is 2p;
nd	the sum of the pixels in all contour ($nd = \sum_{j=1}^{p} n_j$);
x^j_i, y^j_i	the coordinates of the points in the raw object contour, for i=1 to n_j (for j = 1, ..., p);
Xo_j, Yo_j	the approximated coordinates of the object's corners, for j = 1,..., p;
Xa_j, Ya_j	the adjusted coordinates (unknown parameters) of the object's corners, for j = 1, ..., p;

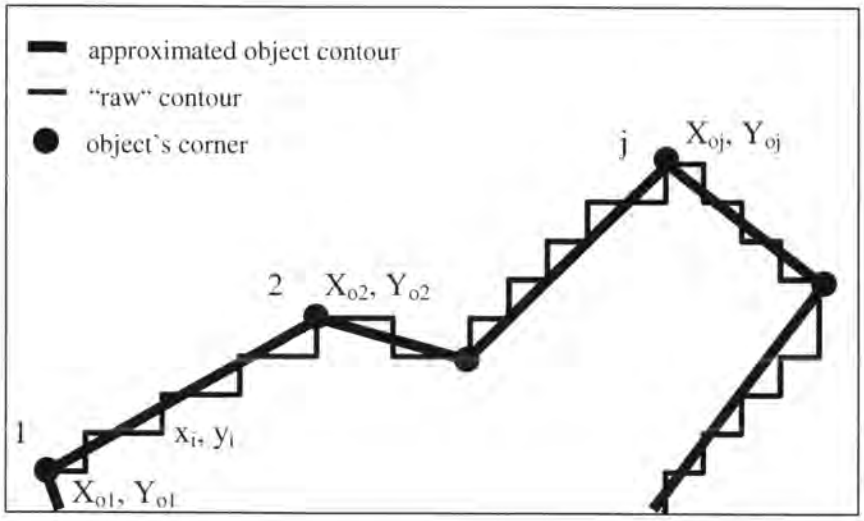


Fig. 5.12 Approximated contour and observations

Equations of the model

The functional model expresses the relation between the observations and the unknown parameters through a mathematical function. In Fig. 5.13 is displayed a building with its approximated (displayed in black) and adjusted (displayed in red) contour.

When determining an appropriate mathematical equation to fit the observations, in order to produce a better representation of the approximated line, we consider the following equations:

(a) assuming that the approximated contour (so the approximated coordinates) of the object's corners determined from the Douglas and Peucker algorithm, are very close to the real position, an equation that maintains the position of the object's corners close to the approximated one is suitable

$$\begin{aligned} Xa_j - Xo_j &= r_j^{(a)} \\ Ya_j - Yo_j &= r_j^{(a)} \end{aligned} \tag{5.1}$$

for j = 1, ..., p.

(b) the angles of the object could be fitted by a 90° degree angle through a function that expresses the condition of the perpendicularity between two straight lines, given by the coordinates of the initial and final points (Fig. 5.13). This assumption came after a survey of the Swiss topographic maps where 80% of the houses are rectangle or a right angle polygon.

$$c_j(Xa_{j-1}, Ya_{j-1}, Xa_j, Ya_j, Xa_{j+1}, Ya_{j+1}) = 0 + r_j^{(b)} \tag{5.2}$$

for j = 1,..., p. When j = p then j+1=1 (the first point)

The function c_j is expressed as per equation 5.3 (the condition of perpendicularity between two segment lines given by the scalar product between two vectors, see Fig. 5.13),

$$\cos(\alpha_{j+1}) = \frac{(Xa_{j+1} - Xa_j)\cdot(Xa_{j+2} - Xa_{j+1}) + (Ya_{j+1} - Ya_j)\cdot(Ya_{j+2} - Ya_{j+1})}{\sqrt{(Xa_{j+1} - Xa_j)^2 + (Ya_{j+1} - Ya_j)^2}\cdot\sqrt{(Xa_{j+2} - Xa_{j+1})^2 + (Ya_{j+2} - Ya_{j+1})^2}} = 0 \tag{5.3}$$

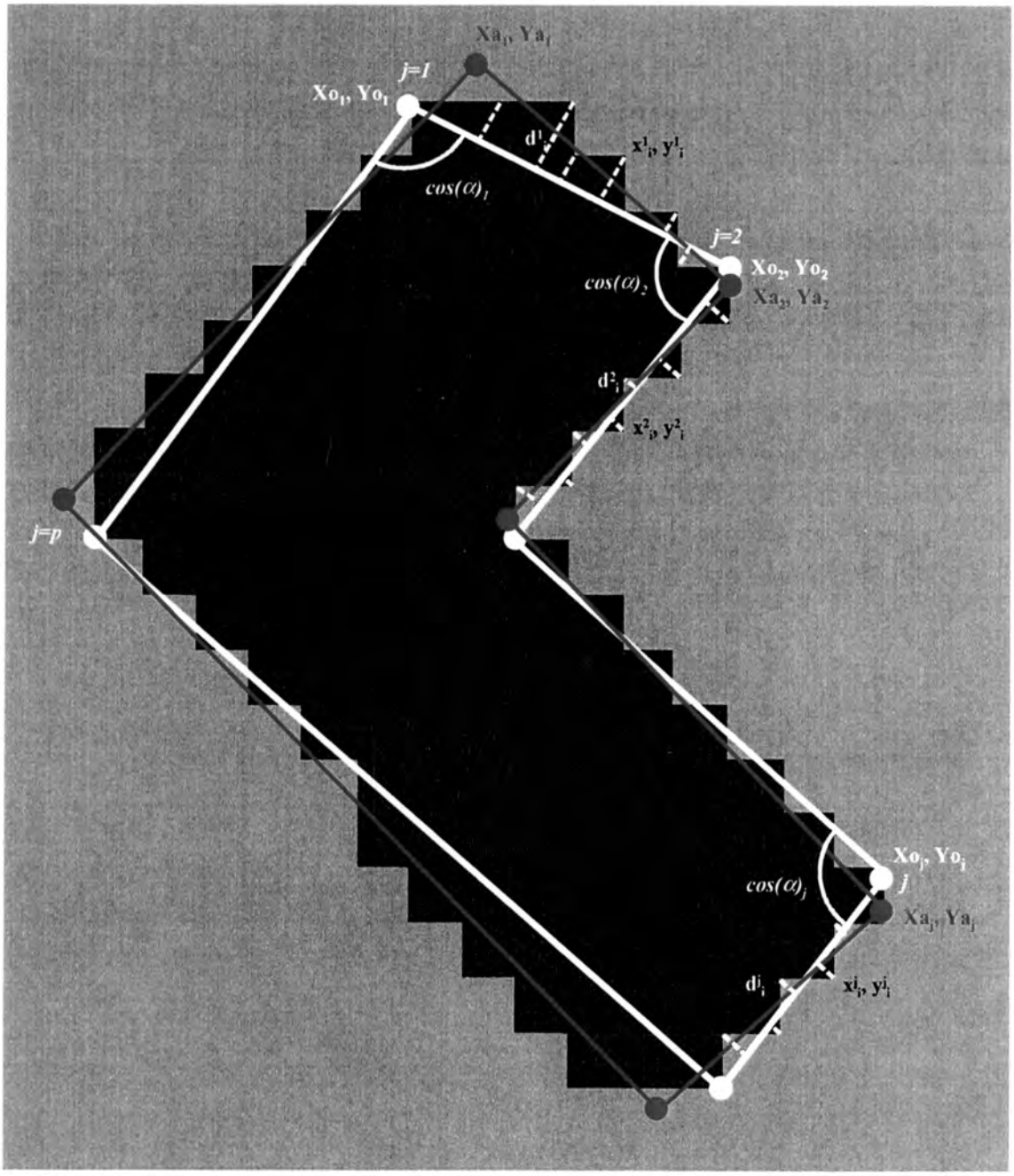


Fig. 5.13 Approximated and adjusted object contour

(c) the points of the raw contour (pixels) can be fitted to a general polygon,

$$d_i^j(Xa_j, Ya_j, Xa_{j+1}, Ya_{j+1}, x_i^j, y_i^j) = 0 + r_i^j. \tag{5.4}$$

for j = 1, …, p and i = 1, …, n_j. When j = p then j+1 = 1 (the first point).

The function d_i^j represents the distance of the ith pixel from the approximated line (Fig. 5.13), and is given by the following equation,

$$d_i^j = \frac{Xa_j Ya_{j+1} + Ya_j x_i + Xa_{j+1} y_i - x_i Ya_{j+1} - y_i Xa_j - Ya_j Xa_{j+1}}{\sqrt{(Xa_j - Xa_{j+1})^2 + (Ya_j - Ya_{j+1})^2}} \qquad (5.5)$$

for $j = 1, \ldots, p$ and $i = 1, \ldots, n_j$.

Ideally, all distances are equal to zero, in which case the pixels will lie on the approximated line. Since this is not the case, we could consider the distances d_i^j as residuals of the pixels that are to be fitted in the line.

Outliers and model characteristics

Our aim is to provide an automatic process for the vectorisation of areal objects (buildings) and this should take account to the following:

- The number of observations is very large (higher the scanner resolution, larger the number of observations)

- The number of unknown parameters (coordinates of the corners) is large (at least six)

- The approximated contour determined by the Douglas and Peucker algorithm gives a very good representation of the building (the coordinates of the corners are particularly close to the correct position).

The following types of outliers are apparent in our model:

1. The equations of type (a) can be considered as without outliers, since the approximated coordinates are very close to the real ones. The Douglas and Peucker algorithm adapted in a close contour delivers a good representation of the building (see Fig. 5.10).

2. The equations of the type right angles could contain gross errors since this condition is not always met. From a survey of the Swiss topographic maps, 80% of the buildings seem to be rectangles or right angle polygons. Therefore, the number of outliers of this type is limited, but when they are present, they present themselves as gross errors.

3. Outliers which present themselves in terms of pixels, which are pixels that are falsely recognised as parts of the areal objects (see errors of first order at Fig. 5.6 a and b). The number of such outliers is small.

The stochastic model

The stochastic model describes the characteristics or properties of the random variables (observations x_i^j, y_i^j, α_j, Xo_j, Yo_j). Its presentation is given through the covariance matrix of the observations Σ_{ll} (see 2.2.1). The elements σ_i can be determined in advance for each observation. The covariance values σ_{ij} (when $i \neq j$) are taken as zero, assuming that the correlation between observations is zero. The standard deviations of the three types of observations are:

σ_i^j standard deviation of points in the raw contour for the observations x_i^j, y_i^j or to be more precise, the distances d_i^j calculated from the coordinates x_i^j, y_i^j. We have taken this value as less than a pixel.

σ_{α_j} standard deviation for the right angles. This value corresponds to 2 degrees.

$\sigma_{Xoj,\ Yoj}$ standard deviation for the approximated coordinates Xo_j, Yo_j. Assuming that the object's corners of the approximated contour are close to the correct position, we accept the value of the standard deviation as two to three pixels.

5.8.2 Computation Procedure

The final goal of our approach is to vectorise the extracted buildings (either from topographic maps or general maps) in an automatic process. The adjustment process goes through the following steps:

Linearisation of the observation equations

The functions d_i^j, c_j contain non linear terms (see equations of type *b* and *c*). The linearisation process could be either analytical or numerical. The Taylor series expansion (analytical linearisation) is used when the form of the functions is not complicated and the first derivatives are easy to determine. When this is not the case (see equations 5.4, 5.5), it is possible to determine the coefficients of the matrix A from the equations of the observations numerically. We have adopted this type of linearisation in our adjustment model.

Estimation of the unknown parameters

For the determination of the unknown parameters an estimator is needed. Considering the model characteristics and the possible outliers present (see 5.8.2) the following should be considered in the choice of the estimator:

- If a Least Square estimator is employed, the disadvantage remains its 0% breakdown point. This estimator will deliver bad results since outliers are present.

- The employment of a high breakdown estimator is also not advantageous in this case, since the number of outliers is not very high. Furthermore, because the number of unknown parameters could be very large (from six to some tens), the calculation time will increase enormously.

- A bounded influence estimator (BIBER-estimator) is more appropriate since it keeps the qualities of the Least Square estimator and can also cope with a small amount of outliers.

The mathematical definition and properties of a BIBER-estimator are given in detail in chapter 4, section 4.4.3. In this section we shall present the practical use of this

estimator in the vectorisation process. The design matrix A with its coefficients is presented below. The abbreviations can be taken from figure 5.13.

Unknown parameters

	Xa_1	Ya_1	Xa_2	Ya_2	Xa_3	Ya_3	...	Xa_{p-2}	Ya_{p-2}	Xa_{p-1}	Ya_{p-1}	Xa_p	Ya_p	
Xo_1	1	0	0	0	0	0	0	0	0	0	0	0	0	*approx. coord.*
Yo_1	0	1	0	0	0	0	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
Xo_p	0	0	0	0	0	0	0	0	0	0	0	1	0	
Yo_p	0	0	0	0	0	0	0	0	0	0	0	0	1	
α_1	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	0	0	0	0	0	0	0	$a_{i,p-1}$	$a_{i,p}$	*right angles*
α_2	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	$a_{i,5}$	$a_{i,6}$	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
α_{p-1}	0	0	0	0	0	0	0	$a_{i,p-5}$	$a_{i,p-4}$	$a_{i,p-3}$	$a_{i,p-2}$	$a_{i,p-1}$	$a_{i,p}$	
α_p	$a_{i,1}$	$a_{i,2}$	0	0	0	0	0	0	0	$a_{i,p-3}$	$a_{i,p-2}$	$a_{i,p-1}$	$a_{i,p}$	
d^1_1	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	0	0	0	0	0	0	0	0	0	*distances*
d^1_2	0	0	$a_{i,3}$	$a_{i,4}$	$a_{i,5}$	$a_{i,6}$	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
d^1_{n1-1}	0	0	0	0	0	0	0	$a_{i,n1-5}$	$a_{i,n1-4}$	$a_{i,n1-3}$	$a_{i,n1-2}$	0	0	
d^1_{n1}	0	0	0	0	0	0	0	0	0	$a_{i,n1-3}$	$a_{i,n1-2}$	$a_{i,n1-1}$	$a_{i,n1}$	
d^2_1	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	0	0	0	0	0	0	0	0	0	
d^2_2	0	0	$a_{i,3}$	$a_{i,4}$	$a_{i,5}$	$a_{i,6}$	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
d^2_{n2-1}	0	0	0	0	0	0	0	$a_{i,n2-5}$	$a_{i,n2-4}$	$a_{i,n2-3}$	$a_{i,n2-2}$	0	0	
d^2_{n2-1}	0	0	0	0	0	0	0	0	0	$a_{i,n2-3}$	$a_{i,n2-2}$	$a_{i,n2-1}$	$a_{i,n2}$	
...	....	....	....	....	....	....	....	....	....	....	....	....	....	
...	....	....	....	....	....	....	....	....	....	....	....	....	....	
...	....	....	....	....	....	....	....	....	....	....	....	....	....	
...	....	....	....	....	....	....	....	....	....	....	....	....	....	
...	....	....	....	....	....	....	....	....	....	....	....	....	....	
d^{p-1}_1	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	0	0	0	0	0	0	0	0	0	
d^{p-1}_2	0	0	$a_{i,3}$	$a_{i,4}$	$a_{i,5}$	$a_{i,6}$	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
$d^{p-1}_{n(p-1)-1}$	0	0	0	0	0	0	0	$a_{i,n(p-1)-5}$	$a_{i,n(p-1)-4}$	$a_{i,n(p-1)-3}$	$a_{i,n(p-1)-2}$	0	0	
$d^{p-1}_{n(p-1)}$	0	0	0	0	0	0	0	0	0	$a_{i,n(p-1)-3}$	$a_{i,n(p-1)-2}$	$a_{i,n(p-1)-1}$	$a_{i,n(p-1)}$	
d^p_1	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	0	0	0	0	0	0	0	0	0	
d^p_2	0	0	$a_{i,3}$	$a_{i,4}$	$a_{i,5}$	$a_{i,6}$	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
d^p_{np-1}	0	0	0	0	0	0	0	$a_{n-1,np-5}$	$a_{n-1,np-4}$	$a_{n-1,np-3}$	$a_{n-1,np-2}$	0	0	
d^p_{np-1}	0	0	0	0	0	0	0	0	0	$a_{n,np-3}$	$a_{n,np-2}$	$a_{n,np-1}$	$a_{n,np}$	

The estimator employed is the BIBER-estimator. The calculation procedure for that estimator can be taken from 4.4.3.

5.8.3 Implementation of the Algorithms in RobVec

The software RobVec (Robust Vectorisation) [Nebiker, 1995; Kanani, 1998; Frischknecht and Kanani, 1998] has been developed at ETH Zürich as a tool for a full automatic vectorisation of areal objects from topographic and general maps. Together with KAMU (Cartographic Pattern Recognition) [Stengele, 1995; Frischknecht, 1999] and RaVis (Raster Visualisation) [Zanini, 1999], RobVec completes a software package that enables the extraction, vectorisation and visualisation of data from topographic and general maps.

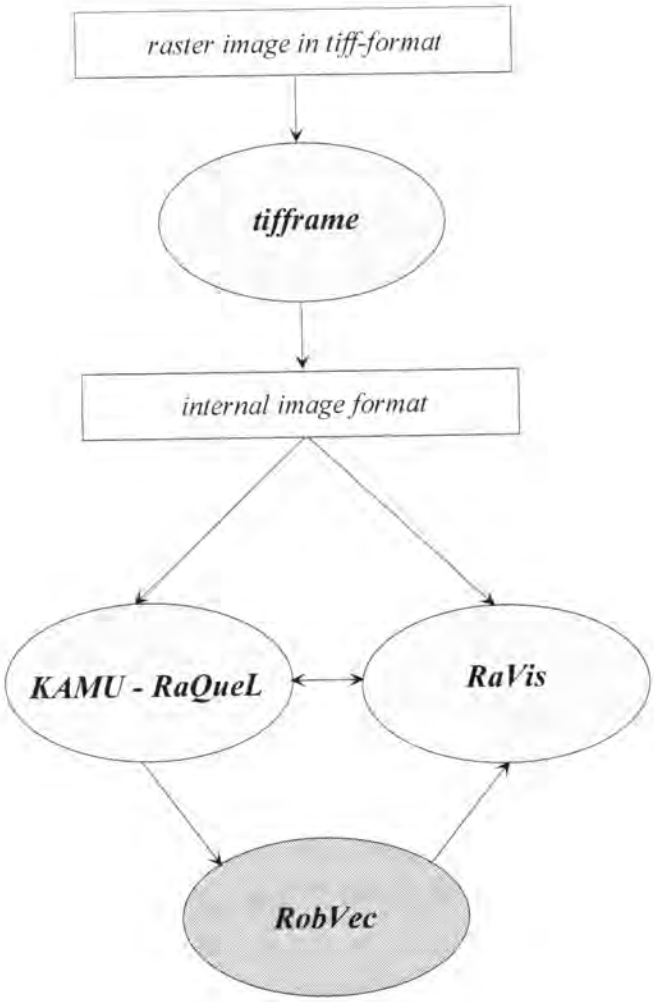


Fig. 5.14 RobVec in the context of other software packages

In this work, we shall present only the module that enables the vectorisation process. Other functions and utilities that the software provides can be taken from the user manual [Nebiker, 1995; Kanani, 1998].

Under the module *Vectorisation* are two important functions:

1. Identification of buildings and determination of the raw contour (see 5.6)
2. Approximation of the raw contour and final adjustment

The approximation of the raw contour and the adjustment process is performed in the following way:

First step: *Determination of the approximated corners of the object contour*
The Douglas and Peucker algorithm adapted for a closed contour is employed at this stage (see 5.7.1). The resulting line is a good approximation of the object contour (Fig. 5.10).

Second step: *Adjustment of the approximated contour*
In the sections 5.8.1 and 4.4.3 the steps of the adjustment process from the determination of the equations of the model, observations, unknown parameters, residuals and estimatior, were given. The choice of the value c, needed for the estimator employed, could be made very easily from the vectorisation interface (see Fig. 5.15). As c is increased, the results tend to the Least Square fit.

Third step: *Alternative models*
After the adjustment process, the adjusted vector contours are analysed to ascertain whether the shapes of some buildings may be represented by "alternative models"; meaning pre-set models of similar shape.

An overview of the Swiss topographic maps has shown that the shape of buildings, in almost 80% of the cases, is a rectangle or a right-angled polygon. The 20% of the buildings which possess more complicated shapes are to be found mostly in the centre of towns (e.g. railway stations, old town areas or other objects of interest). As outlined in 5.8.1, this was also a reason for introducing the equations of the type right angle, since in 80% of the cases this condition is met.

```
ROBVEC  Vectorisation


--PARAMETER: Scanner Resolution

1) Scanner resolution [Pixel/mm]........... 20


1. Step

--PARAMETER: Douglas-Peucker Procedure

2) Maximum Tolerance Distance [m]......... 1.3

3) Minimum Object Dimension [Pixel]....... 4


2. Step

--PARAMETER: Robust Adjustment

4) c-robust (real number)................. 4.0


3. Step

--PARAMETER: Alternative Models

5) Smoothing: Max. Limit Distance [m].... 1.2

6) Rectangle Model - s₀ [m]............... 8
```

Fig. 5.15 Vectorisation Module - Interface at RobVec

The employment of the approach "alternative models" is going to be adopted only for buildings where the number of corners is limited (not more than ten), Fig. 5.16 (a), since the probability that a larger building could be fitted by a rectangle without a loss in accuracy, is very low.

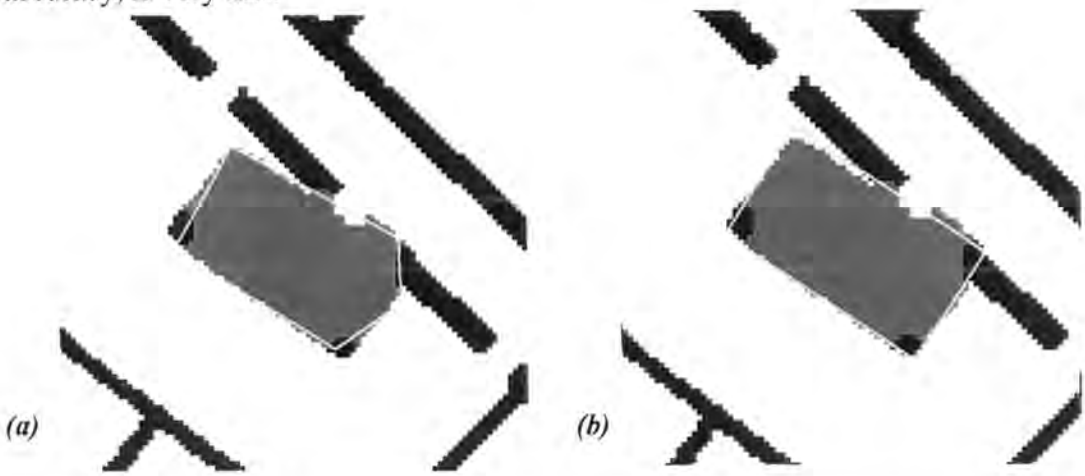


Fig. 5.16 Alternative model of the type rectangle adopted in the model represented in a

The decision whether the simpler model is going to be accepted, is taken according to the following criteria:

Let us denote by $s_{o\ (simple\ model)}$ and $s_{o\ (complicated\ model)}$ the estimation of s_o (see also equation 4.58). These two values are determined for the complicated shape and rectangle model. If the simpler model (Fig. 5.16 b) resembles the old one quite well (Fig. 5.16 a), $s_{o\ (simple\ model)}$ will be of similar magnitude to $s_{o\ (complicated\ model)}$. It is obvious, of course, that $s_{o\ (simple\ model)}$ will be larger than $s_{o\ (complicated\ model)}$. By the statistical test, the analysis of the variance, we determine whether the simplest model is accepted or the comprehensive model is adopted.

With the employment of alternative models and the robust estimators employed in the adjustment process, it is possible to eliminate errors deriving from the false recognition of parts of areal objects (see also Fig. 5.9).

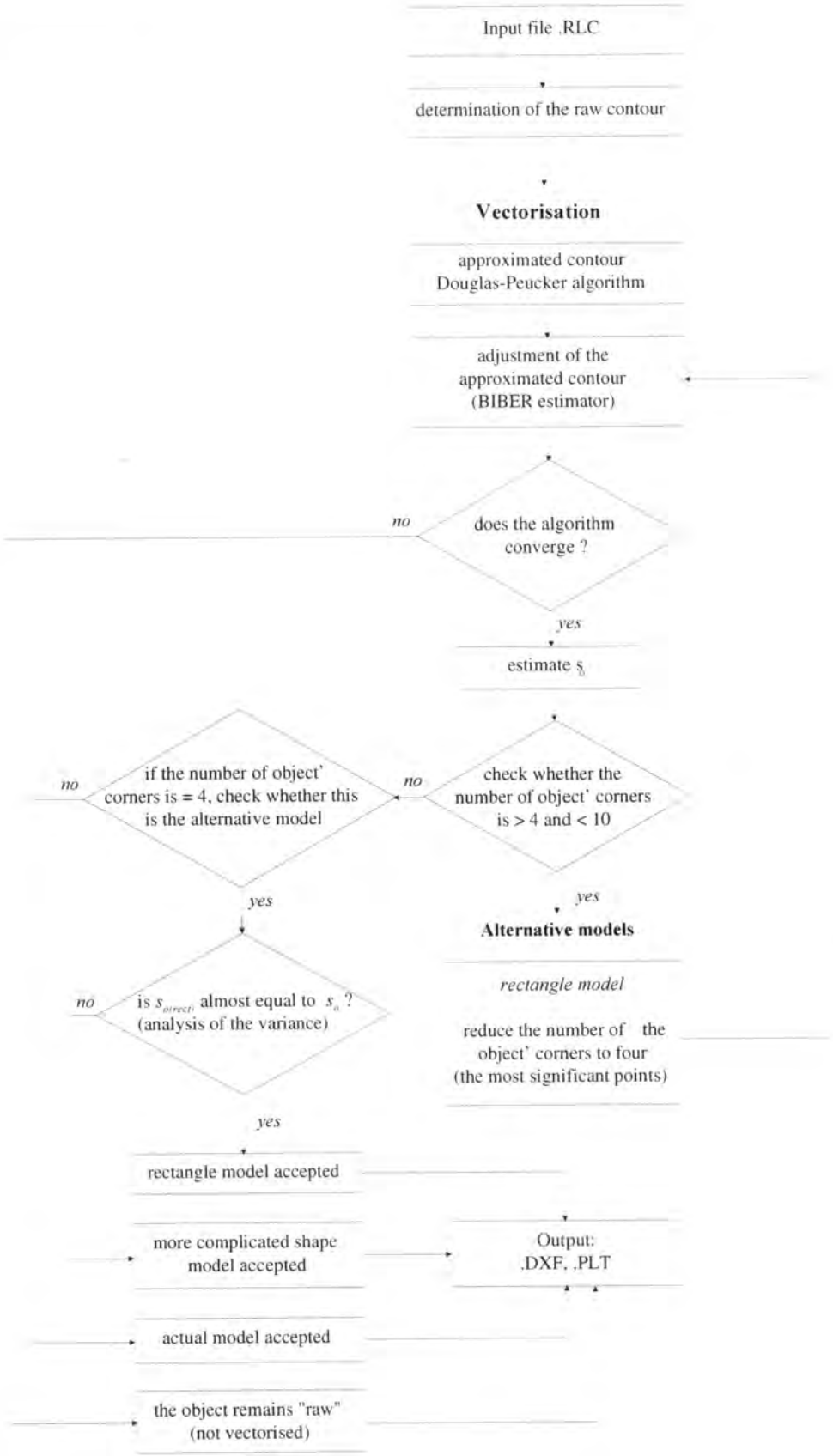


Fig. 5.17 Flowchart-illustration of the steps of the algorithm employed for the automatic vectorisation

5.8.4 Applications and Results

The examples presented in this section show the results of the vectorisation process of the extracted buildings from topographic and general maps.

All calculations are performed using a machine IBM RS/6000, Modell F50 SMP (CPU: 4 x 332 MHZ, RAM 1024 MB).

Statistical results

After the vectorisation the software RobVec [Kanani, 1998] records the following important data:

Topographic and general maps:

Pixel map of city Winterthur (size 3x3 km)

Scale	1:25000
Scanner resolution	40 l/mm
Total number of polygons (exterior and interior borders)	3646
Number of adjusted polygons	3645
Number of not vectorised objects	1
Percentage of vectorisation	99.97
Calculation time	ca. 6 min

Pixel map of city Lugano (size 6x6 km)

Scale	1:25000
Scanner resolution	20 l/mm
Total number of polygons (exterior and interior borders)	7574
Number of adjusted polygons	7550
Number of not vectorised polygons	24
Percentage of vectorisation	99.68%
Calculation time	ca. 6 min

Pixel map of scale 1:200000 (sheet 1) (size 48x48 km)

Scale	1:200000
Scanner resolution	20 l/mm
Total number of polygons (exterior and interior borders)	5134
Number of adjusted polygons	5097
Number of not vectorised polygons	37
Percentage of vectorisation	99.28%
Calculation time	ca. 2 min

Pixel map (general map) of Kanton Aargau (size 1x1.2 km)

Scale	1:10000
Scanner resolution	40 l/mm
Total number of polygons (exterior and interior borders)	145
Number of adjusted polygons	145
Number of not vectorised polygons	0
Percentage of vectorisation	100%
Calculation time	ca. 1 min

From the statistical results shown, a small percentage of polygons (objects) that are not vectorised are evident. These can occur as a consequence of

- a very complicated shape, the algorithm does not provide the needed results
- very small surfaces (small objects which have been identified as buildings)
- no elimination before hand of possible errors (e.g. railway lines) from the extraction process.

From the vectorised objects, 1-2% do not fulfil cartographic requirements and need manual corrections. Manual corrections are also required for objects which do not represent buildings (e.g. font or part of it).

Current projects

The developed approach for the vectorisation of areal objects is in operational use at the Swiss Federal Office of Topography and private companies. An overview of current objects is presented below:

Vector25 - The Vector25 data set derives from the vectorisation of the Swiss topographic map of scale 1:25000, where the software KAMU and RobVec are employed for the extraction and vectorisation of the buildings. The vector data of this data set are topologically structured [Eidenbenz, 1995; Sonder, 1998].

Vector200 - This data is similar to the Vector25 data set, accept that the vector data are from the topographic map of scale 1:200000. The vector data of buildings derive from the employment of KAMU and RobVec.

Vector data from general map - The software packages KAMU and RobVec have found use also for the extraction and vectorisation of buildings from general maps. The vectorisation results are to be seen in Fig. 5.23.

3D - Visualisation - Another research project that is pursued at ETH Zurich addresses how raster images and digital terrain models can be used for the creation and visualisation of three dimensional landscape models [Zanini and Carosio, 1996; Zanini, 1999]. The aim is not only visualisation of data but 3D modelling and graphical presentation of extracted thematic information from raster images. The procedure to extract and structure the information (e.g. buildings) has been presented above. The necessary height information needed for the 3D modelling can either be derived implicitly from thematic information (church, tree, building, etc.) or explicit height data (photogrammetry, statistical data, etc.) can be used (Fig. 5.24).

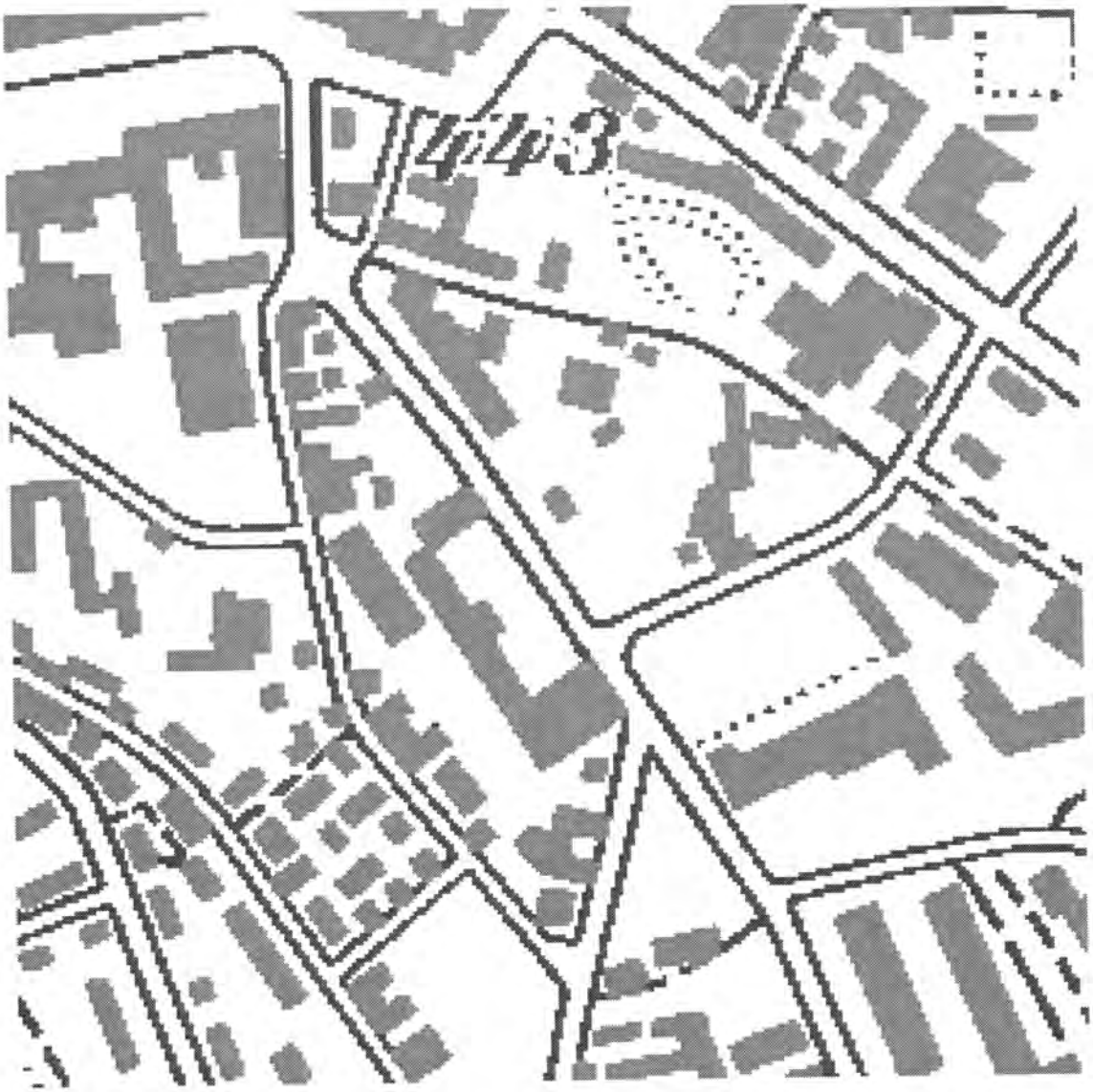


Fig. 5.18 Extracted buildings with the software KAMU *(Area: City Winterthur; Data: Pixelmap PK25, resolution 40l/mm, © Swiss Federal Office of Topography)*

Fig. 5.19 Vectorised buildings with the software RobVec *(Area: City Winterthur; Data: Pixelmap PK25, resolution 40 l/mm, © Swiss Federal Office of Topography)*

Fig. 5.20 Extracted buildings with the software KAMU *(Area: City Lugano; Data: Pixelmap PK25, resolution 20 l/mm. © Swiss Federal Office of Topography)*

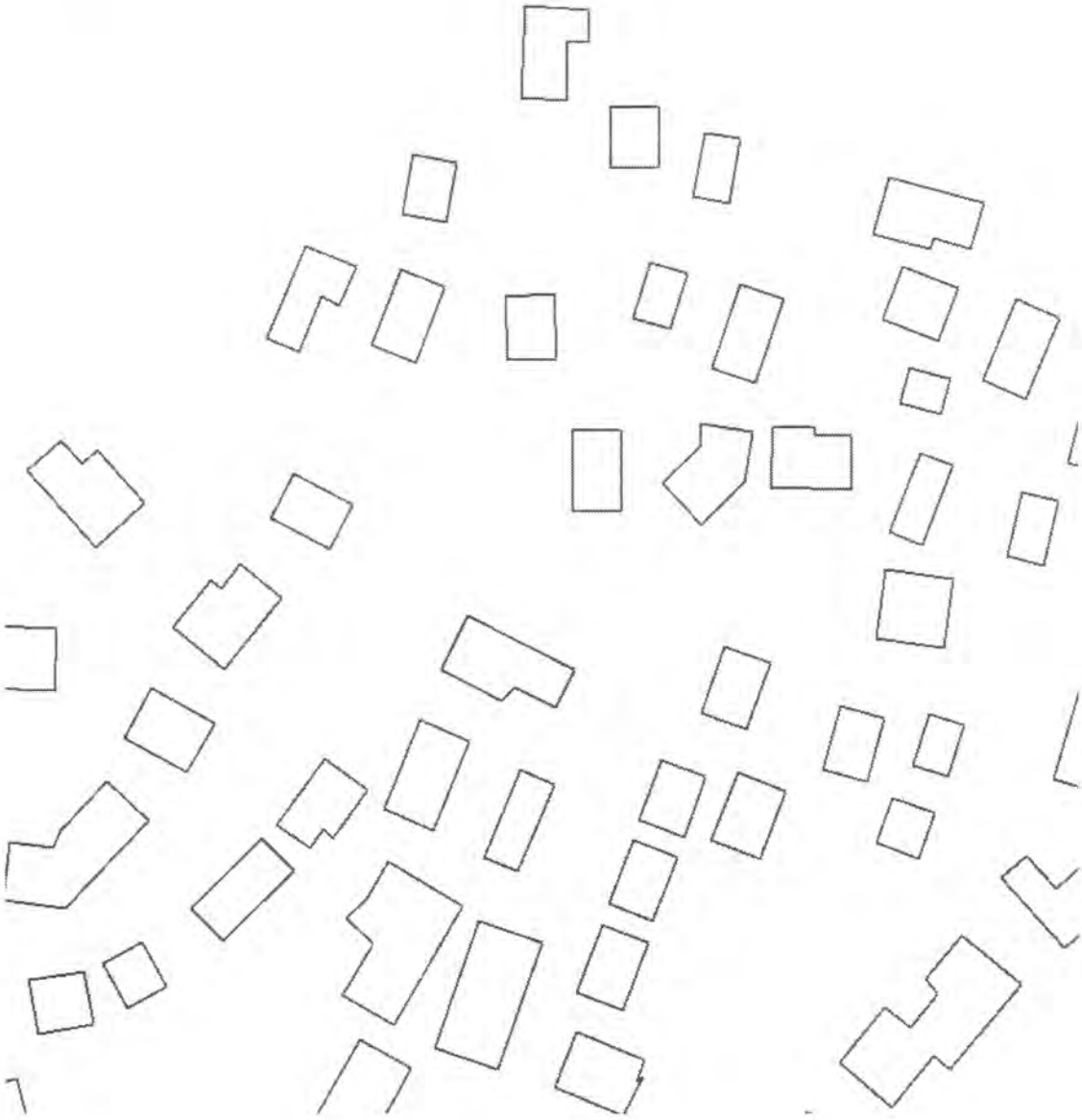


Fig. 5.21 Vectorised buildings with the software RobVec *(Area: City Lugano; Data: Pixelmap PK25, resolution 20 l/mm. © Swiss Federal Office of Topography)*

Fig. 5.22 Extracted buildings with the software KAMU *(Area: Kanton Aargau; Data: Pixel general map, resolution 40 l/mm. © Vemessungsamt der Kanton Aargau)*

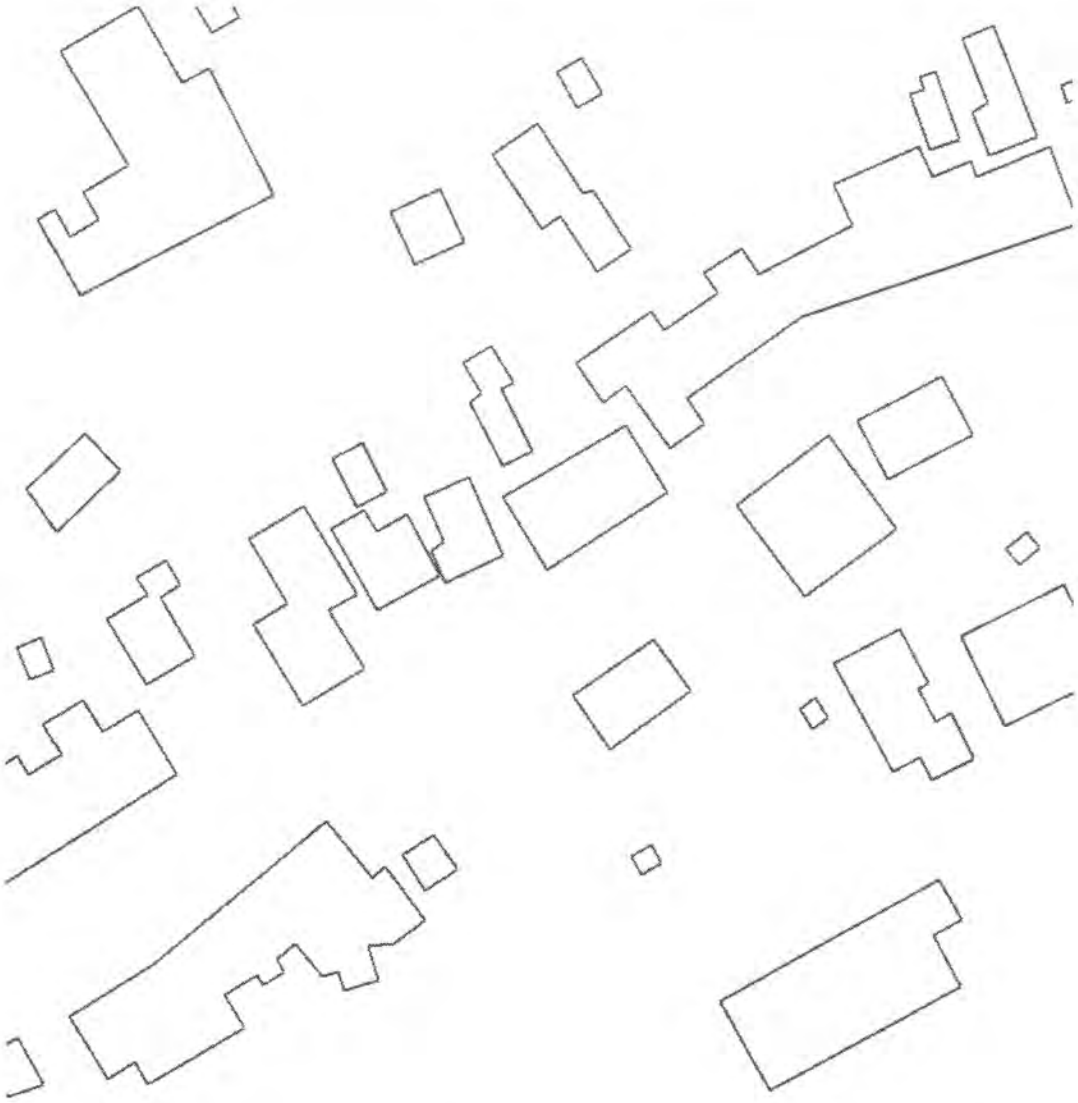


Fig. 5.23 Vectorised buildings with the software RobVec *(Area: Kanton Aargau; Data: Pixel general map, resolution 40 l/mm, © Vemessungsamt der Kanton Aargau)*

Fig. 5.24 3D - Visualisation with extracted buildings and trees *(Area Zurich Altstetten, ETH-Hönggerberg;, Data: Pixelmap PK25, DHM25 © Swiss Federal Office of Topography)*

Chapter 6

Summary and Conclusions

The work presented was motivated by the growing need for more effective methods for data evaluation. Robust estimators were employed for the evaluation of data in the field of geodetic transformations and raster to vector conversion. In these applications and others, it is apparent that the robust statistic is a very useful method by which it is possible to arrive at reliable results very quickly. The developed algorithms have been implemented in software packages and are a good tool for the every day work of private companies and governmental institutions. The most important achievements and conclusions of this work are as follows:

For ***geodetic transformations*** it is now possible to determine the unknown parameters by two robust estimators: the LMS-estimator from the class of high breakdown estimators, and the BIBER-estimator from the class of bounded influence estimators.

- The LMS estimator is a very good outlier detector, it is therefore very advantageous in cases where the number of outliers is high and the priority remains the bounding

of the influence of the outliers rather than the statistical information for the unknown parameters, residuals and others. Its implementation in the first stage of the algorithm process for the determination of the approximation parameters needed for the linearisation process is very useful. This facilitates the effectiveness of the employment of a bounded influence estimator (e.g. BIBER estimator) at a later stage.

- In cases where the number of outliers expected is not that high, and the assumption of the normal distribution can still be accepted for the majority of the data, it has been demonstrated that the employment of the BIBER estimator delivers very good results, for the following reasons:
  - it is practically identical to the LS estimator for normal distributed measurements
  - single errors do not influence the estimator greatly, even when the redundancy of the system is not that high
  - it is possible to estimate the variance of groups of observations
  - it can also work very efficiently for hundreds of measurements and unknown parameters which also allows the employment of this estimator in other fields where a great amount of data need to be evaluated (e.g. photogrammetry).

In this work it has been shown that the robust statistics can also give very good results in other fields related to geodesy. One such field is the ***automatic vectorisation*** of areal objects from scanned topographic maps, where it has been shown that through an appropriate adjustment model, facilitated by robust techniques, it is possible to obtain very good results in a fully automatic process. This is a great advantage when a huge amount of data (hundreds of unknown parameters and observations) needs to be processed very quickly, as is common nowadays.

These applications and others emphasise once again the importance of robust estimators in data evaluation, and therefore they should be further pursued.

References

Baarda, W. (1968): A Testing Procedure for use in Geodetic Networks, Netherlands Geodetic Commission. Geodetic Publication 2/5.

Benning, W. (1978): Zur Auswertung geodätische Messungen und automatisierter Fehlersuche. Allgemeine Vermessungsnachrichten, No. 1.

Benning, W. (1995): Test von Ausreisern bei der Helmerttransformation, Zeitschrift für Vermessungswesen, No. 5.

Brennecke, J. (1990): Mustererkennung zur automatischen Bildschirmführung beim Editing von Scan-Digitalisierungen. Nachrichten aus dem Karten und Vermessungswesen I, No. 105, pp. 19-30.

Bloomfield, P., Stieger, W. L. (1983): Least Absolute Deviation: Theory, Applications, and Algorithms. Birkhäuser, Boston.

Boyle A. R. (1970): Computer Aided Compilation. Hydrographic Conference, January, Ottava.

Borutta, H. (1988): Robuste Schätzverfahren für geodätische Anwendungen. University of the Federal Armed Forces Munich, No. 33.

Box, G. E. P. (1953): Non Normality and Tests on Variances. Biometrika, 40, pp. 318-335.

Brügelmann, R. (1996): Recognition of Hatched Cartographic Patterns, International Archives of Photogrammetry and Remote Sensing (IAPRS), Volume XXXI, Part B3, pp. 83-87, Vienna.

Carosio, A. (1979): Robuste Ausgleichung. Vermessung Photogrammetrie Kulturtechnik, No. 11, pp. 293-297.

Carosio, A. (1981): Das geodätische Programmsystem der Landestopographie. Report of the Calculation Centre of the Swiss Federal Office of Topography. No. 4, 3/81.

Carosio, A. (1982): Robuste Ähnlichkeitstransformation und Interpolation nach dem arithmetischen Mittel. Vermessung Photogrammetry Kulturtechnik, No. 6, pp. 196-200.

Carosio, A. (1983): Verfahren der multivariaten Statistik zur Beurteilung der Qualität der Resultate und der Zuverlässigkeit geodätischer Messsysteme. Dissertation of the ETH Zurich.

Carosio, A. (1990): Die Theorie der Zuverlässigkeit im Vermessungswesen. Weiterbildungstagung Zuverlässigkeit in der Vermessung. ETH Zurich

Carosio, A. (1991): Überblick über Zweck und Verfahren der Numerisierung. Beiträge zur Weiterbildungstagung "Plannumerisierung", September, ETH Zurich.

Carosio, A. (1996): Lecture Notes in Theory of Errors. Institute of Geodesy and Photogrammetry, ETH Zurich.

Caspary, W., Chen, Y. Q., König, R. (1983): Kongruenzuntersuchungen in Deformationsnetzen durch Minimierung der Summe der Klaffungsbeträge. W. Wrlsch (Ed.), Deformationsanalysen '83, Schriftenr. Studieng. Verm. W., HSBw München, Heft 9.

Caspary, W., Borutta, H. (1986): Geometrische Deformationsananalyse mit robusten Schätzverfahren. Allgemeine Vermessungsnachrichten, No. 93, pp. 315-126.

Caspary, W. (1988): Fehlerverteilung, Methode der kleinsten Quadrate und robuste Alternativen, Zeitschrift für Vermessungswessen, No. 3.

Caspary, W. (1996): Anmerkungen zum robusten Schätzen. Allgemeine Vermessungsnachrichten, No. 7, pp. 287-289.

Coster, M., Chermant, J-L. (1985): Prècis D'analyse D'Images. Éditions du Centre National de la Recherche Scientifique, Paris.

Danuser, G. (1992): Robuste Schätzer mit hohem Bruchpunkt. Grey Series No. 207, Institute of Geodesy and Photogrammetry, ETH Zürich.

Doerman, D. (1997): An Introduction to Vectorization and Segmentation, Lecture Notes in Computer Science - Graphics Recognition, Springer, pp. 1-5.

Donoho, D. L., Huber, P. J. (1983): The notation of breakdown point, in A Festschrift for Erich Lehmann, edited by B. Bickel, K. Doksum and J. L. Hodges, Jr., Wadsworth, Belmont, CA.

Douglas, D., Peucker, T. (1973): Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer, Vol. 10, No. 2, pp. 112-122.

Eidenbenz, Ch. (1989): Scannertechnik zur Erfassung von Plänen und Karten, Vermessung Photogrammetrie Kulturtechnik, No. 2, pp. 110-116.

Eidenbenz, Ch. (1995): GIS-Grundlagedaten beim Bundesamt für Landestopographie, Vermessung Photogrammetrie Kulturtechnik, No 5, pp. 330-333.

Fuchs, H. (1980): Untersuchungen zur Ausgleichung durch Minimieren der Absolutsumme der Verbesserungen. Dissertation of the Technical University, Graz.

Frischknecht, St. Kanani, E. (1998): Automatic Interpretation of Scanned Topographic Maps: A Raster-Based Approach. In: Graphics Recognition; Algorithms and Systems; K. Tombre, A. K, Chhabra (Edit.), Berlin (etc.), Springer, pp. 207-220.

Frischknecht, St (1996): Extraction beliebig orientierter Kartensymbole und Schriftelemente aus gescannten topographischen Karten durch rotations-invariante Filterfunktionen und Template Matching,

Proceedings Kartographie Kongress Interlaken 96, Cartographic Series, No. 14 of the Swiss Society of Cartography.

Frischknecht, St. (1999): Eine Abfragesprache für die Geometrie von Rasterelementen für die Rasterorientierte Kartographische Mustererkennung und Datenanalyse. Dissertation of the ETH Zurich.

Giebels, M., Weber, W. (1990): Methoden der Daten Erfassung für das DLM200. Kartographische Nachrichten, No. 5, pp. 169-174.

Hahn, W., Bill, R. (1984): Ein Vergleich der L_1 - und L_2 - Norm am Beispiel Helmerttransformation. Allgemeine Vermessungsnachrichten, No. 11-12, pp. 440-449.

Hake, G. (1974): Kartographie I and II. De Gruyter Verlag, Berlin (etc.).

Hake, G. (1994): Kartographie I and II. De Gruyter Verlag, Berlin (etc.).

Hake, G., Grünreich, D. (1994): Kartographie, De Gruyter Verlag, Berlin (etc.).

Hampel, F. R. (1971): A general qualitative definition of robustness, Annals of mathematical Statistics, 42, pp. 1887-1896.

Hampel, F. R. (1974): The Influence Curve and its Role in Robust Estimation. Journal of the American Statistical Association, 69, pp. 383-393.

Hampel, F. R. (1975): Beyond location parameters: Robust concepts and methods, Bull. Int. Stat. Inst., 46, pp. 375-382.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. R., Stahel, W. A. (1986): Robust statistics: the approach based on influence functions. Wiley, New York.

Heck, B. (1981): Der Einfluss einzelner Beobachtung auf das Ergebnis einer Ausgleichung. Allgemeine Vermessungsnachrichten, No. 1, pp. 17-34.

Hodges, J. L. (1967): Efficiency in normal samples and tolerance of extreme values for some estimates of location, Proceedings of the Fifth Berkley Symposium in Math. Stat. Probab., 1, pp. 163-168.

Huber, P. J. (1964): Robust Estimation of a Location Parameter. Annals of Mathematical Statistics, 35, pp. 73-101.

Huber, P. J. (1981): Robust Statistics, Wiley, New York.

Hurni, L. (1996): Verarbeitung von hybriden Daten, Wieterbildingstagung – September 1996, Edited by F. Golay, EPF Lausanne.

Jäger, E. (1997): Stand und Entwicklungen der Geobasisinformationssysteme ALK und ATKIS. Kartographische Schriften, Vol. 2, pp. 38-46.

Kampmann, G. (1986): Robuster Ausreissertest mit Hilfe der L1-Norm Methode. Allgemeine Vermessungsnachrichten, No. 4, pp. 139-147.

Kampmann, G. (1990): Robuster Ausreissertest mit Hilfe der bedingten L1-Normschätzer. Allgemeine Vermessungsnachrichten, No. 1, pp. 27-32.

Kampmann, G. (1994): Robuste Deformationsanalyse mittels balancierter Ausgleichung. Allgemeine Vermessungsnachrichten, No. 1, pp. 8-17.

Kanani, E. (1998): Unpublished user manual of the software RobVec, Institute of Geodesy and Photogrammetry, ETH Zurich.

Klein, H., Förstner, W. (1984): Realization of Automatic Error Detection in the Block Adjustment Program PAT-M3 Using Robust Estimators. Proceedings of XV Congress of International Society for Photogrammetry and Remote Sensing (ISPRS), Com III, Rio de Janeiro.

Marazzi, A. (1993): Algorithms, routines, and S functions for robust statistics, Wadsworth & Brooks/Cole Advanced Books & Software, Pacific Grove, California.

Maronna R. A., Bustos, O., and Yohai, V. (1979): Bias- and efficiency-robustness of general M-estimators for regression with random carriers, in Smoothing Techniques for Curve Estimation. Edited by T. Gasser and M. Rossenblatt, Springer, New York, pp. 91-116.

Meissl, P. (1980): Adjustment of Levelling Networks by Minimizing the Sum of Absolute Residuals. NAD-Symposium, Ottawa.

Mayer, H. (1993): Automatische wissensbasierte Extraktion von semantischer Information aus gescannten Karten. Dissertation of the Technical University Munich.

Morehouse, S. (1995): GIS-based map compilation and generalization. GIS and GENERALIZATION, Methodology and Practice. GISDATA 1, Taylor and Francis, pp. 21-30.

Nebiker, St. (1995): Unpublished user manual of the software AutoVec, Institute of Geodesy and Photogrammetry, ETH Zurich.

Nebiker, St. Carosio, A. (1994): Automatic Extraction and Structuring of Objects from Scanned Topographical Maps, IAPRS, Vol. XXX, Part 1.

Nebiker, St. (1997): Spatial Raster Data Management for Geo-Information Systems: a Database Perspective. Dissertation of the ETH Zurich.

Ohlohf, T. (1992): Zur Digitalisierung topographischer Karten unter Einsatz von Geo-Informationssystemen. Zeitschrift für Vermessungswesen, No. 7, pp. 377-385.

Pelzer, H. (1979): Some Criteria for the Accuracy and the Reliability of Networks. Paper, press. to the XVII. General Assembly of the IUGG, Canberra.

Pope, A. J. (1976): The statistics of residuals and the detection of outliers. XVI General Assembly of the IUGG, Grenoble.

Prashker, S. (1991): Demonstration of the ZAPPER software - A collection of Line Simplification and Smoothing Algorithms for Maps. Technical Papers, Cartography and GIS/LIS, Vol. 2, ACSM-ASPRS Annual Convention, Baltimore.

Robinson, G. J. (1995): A hierarchical top-down bottom-up approach to topographic map generalization. GIS and GENERALIZATION, GISDATA 1, Taylor & Francis, pp. 235-245.

Rousseeuw, P. J. (1883): Multivariate Estimation With High Breakdown Point, presented at Fourth Pannonian Symposium on Mathematical Statistics and Propability, Bad Tatzmannsdorf, Austria.

Rousseuw, P. J. (1984): Least median of squares regression, J. Am. Stat. Assoc. 79, pp. 871-880.

Rousseeuw, P. J., Leroy A. M. (1987): Robust regression and outlier detection, Wiley, New York.

Samet, H., Soffer, A. (1994): A Legend-Driven Geographic Symbol Recognition System, Proceedings of the International Conference on Pattern Recognition and Neural Networks, Jerusalem,Vol. II, pp. 350-355.

Sierra, J. (1982): Image Analysis and Mathematical Morphology. Academic Press, London.

Siegel, A. F. (1982): Robust regression using repeated medians. Biometrika 69, pp. 242-244.

Somogyj, J., Zavoti, J. (1990): Die Anwendung der Lp – Norm Schätzung für Àhnlichkeitstransformationen. Zeitschrift für Vermessungswesen, No. 1.

Sonder, G. (1998): Vector25, NDK-Räumliche Informationssysteme. Institute of Geodesy and Photogrammetry, ETH Zürich.

Sonder, G., Jäger, P. (1998): 10 Millionen Geo-Objekte, Vom Vektordatensatz zum Topographischen Informationssystem, ArcAktuell Nr. 4.

Spiess, E. (1979): Lecture Notes in Cartography. Institute of Cartography, ETH Zurich.

Spiess, E. (1992): Lecture Notes of Computer aided Cartography. Institute of Cartography, ETH-Zürich.

Stengele, R. (1995): Kartographische Mustererkennung-Rasterorientierte Verfahren zur Erfassung von Geo-Informationen, Dissertation of the ETH Zurich.

Tukey, J. W. (1962): A survey of Sampling from Contaminated Distributions. In: J. Olkin (Ed.), Contributions to Probability and Statistics, pp. 448-485, Standford Univ. Press.

Tobler, W. R. (1966): Numerical Map Generalization. Michigan Inter-University Community of Mathematical Geographers, Discussion Paper No. 8, Department of Geography, University of Michigan.

Werner, H. (1984): Automatic gross error detection by robust estimatiors. Proceedings of XV Congress of International Society of Photogrammetry and Remote Sensing (ISPRS), Com. III, Rio de Janeiro. pp. 1101-1108.

Wicki, F. (1992a): Robuste Ausgleichung geodätischer Netze. Grey Series No. 189, Institute of Geodesy and Photogrammetry, ETH Zürich.

Wicki, F. (1992b): Robuste M-Schätzer und Zuverlässigkeit, Grey Series No. 190, Institute of Geodesy and Photogrammetry, ETH Zürich.

Wicki, F. (1999): Robuste Schätzverfahren für die Parameterschätzung in geodätischen Netzen. Dissertation of the ETH Zürich.

Youcai, H. (1995): On the design of estimators with high breakdown points for outlier identification in triangulation networks. Bulletin Geodäsique, No. 69.

Yamada, H., Yamamoto, K., Hosokawa, K. (1993): Directional Mathematical Morphology and Reformalised Hough Transformation for the Analysis of Topographic Maps. IEEE Transactions on Pattern Analysis and Machine Intelligence, No. 4, pp. 380-387.

Zanini, M., Carosio, A. (1996): Landscape Modelling and Visualization - from 2-D Surfaces to 3-D Volumes, International Journal of Geomatics, No. 5, pp. 6-8.

Zanini, M. (1999): Dreidimensionale synthetische Landschaften: Wissensbasierte dreidimensionale Rekonstruktion und Visualisierung raumbezogener Informationen. Dissertation of the ETH Zürich.

Acknowledgements

First and foremost, I would like to thank Prof. Dr. Alessandro Carosio for giving me the possibility of undertaking this interesting work; and providing me with advice and encouragement throughout the course of my project.

I would also like to thank my co-examiner Prof. Dr. Alfio Marazzi, for his careful reading of my work. His comments and suggestions were of great value, particularly with regard to the statistical problems.

A special thank you goes to my husband, Philip, for his help and patience in reading through my thesis.

Finally, I would like to express my gratitude to my colleagues in my group at ETH Zürich, for providing both a friendly and stimulating environment. In particular, I would like to mention Stephan Nebiker, Marc Zanini and Steffen Frischknecht for collaboration in projects that were connected with my work, Markus Koller, Cedric Moullet and Dante Salvini for the examples in the pattern recognition, raster visualisation and geodetic transformation, and Mattias Pfund for translating the abstract into German.