

分类号: P229

密 级: 公 开

U D C :

单位代码: 10424

学 位 论 文

测深异常数据的自动检测

曲 萌

申请学位级别: 硕士学位 专业名称: 大地测量学与测量工程

指导教师姓名: 阳 凡 林 职 称: 副 教 授

山 东 科 技 大 学

二 零 零 八 年 五 月

论文题目：

测深异常数据的自动检测

作者姓名： 曲 萌

入学时间： 2005 年 9 月

专业名称： 大地测量学

研究方向： 现代测量数据处

理与测量工程

理论及其应用

指导教师： 阳 凡 林

职 称： 副 教 授

论文提交日期： 2008 年 5 月

论文答辩日期： 2008 年 6 月

授予学位日期：

**AUTOMATIC DETECTING OF OUTLIERS
FOR BATHYMETRIC DATA**

A Dissertation submitted in fulfillment of the requirements of the degree of

MASTER OF PHILOSOPHY

from

Shandong University of Science and Technology

by

Qu Meng

Supervisor: Associate Professor Yang Fanlin

Geomatics College

May 2008

声 明

本人呈交给山东科技大学的这篇硕士学位论文，除了所列参考文献和世所公认的文献外，全部是本人在导师指导下的研究成果。该论文资料尚没有呈交于其它任何学术机关作鉴定。

硕士生签名：

日 期：

AFFIRMATION

I declare that this dissertation, submitted in fulfillment of the requirements for the award of Master of Philosophy in Shandong University of Science and Technology, is wholly my own work unless referenced or acknowledge. The document has not been submitted for qualification at any other academic institute.

Signature:

Date:

摘 要

由于海洋环境的复杂性,致使采集的水深数据包含诸多异常数据。为了真实反映海底地形,必须首先进行水深异常数据的检测,剔除异常水深值。由于水深测量的数据量巨大且影响因素很多,其数据处理工作量非常大,因此,研究出一种快速、高效、自动的异常数据定位方法对于测深数据的处理十分重要。

本文叙述了水深测量的基本原理,即海洋测深的空间结构、潮汐改正等;介绍了传统的水深测量异常数据处理的方法,包括单波束系统和多波束系统(传统的人工交互式编辑和自动编辑)的方法以及数字图像处理中图像去噪的一般方法,并归纳了其各自的特点;又针对多波束测深系统获取的数据的特点,参考数字图像处理中图像去噪的方法,提出了一种水深异常数据的自动探测方法,其借助于投影后的水深密度,自动搜索出投影面上正常数据的边缘线,以此对异常数据进行自动判断的方法;最后利用 VC++ 开发工具,具体实现了该算法,并通过实测数据说明了该算法进行水深异常数据的自动探测的方便性、可靠性及实用性。

通过实践证明,本文提出的水深异常数据的自动探测方法,操作简单合理,在保证剔除异常数据的稳健性的基础上,可以实现水深异常数据的快速有效剔除,是一种快速稳健可靠的异常数据自动探测方法。

关键词: 水深测量, 异常值, 自动检测

ABSTRACT

Ocean environment is very complex, so bathymetric survey data contain many outliers. In order to reflect the real seafloor topography, outliers must be detected and eliminated firstly. The work to process bathymetric survey data is great, for water depth data is a great mount and contains a lot of influencing factors, so we must research an automatic, rapid and robust approach to remove outliers and noises.

This paper introduced the basic principles of bathymetric survey, including the spatial structure of marine sounding, tidal theory and so on; discussed the various methods of outliers data processing, including traditional single beam system, multi-beam sounding system (the interactive manual editing and the automatic editing) and digital image de-noising in digital image processing, and then inducts their own characteristics; for the characteristics of multibeam echo sounding data, with the help of digital image de-noising technology, a kind of automatic algorithm of detecting outliers is proposed, which is similar to the interactive manual editing and can automatically search the edge of the normal data with the density projection. By using VC++, the module is designed and can automatically detect outliers.

It is proved in practice that the algorithm can realize the rejection of outliers quickly and effectively, and can guarantee the robust data processing method to eliminate outliers, which is a kind of rapid, robust algorithm to detect the outliers automatically.

Keywords: sounding, outlier, automatic detection

目 录

1 绪 论.....	1
1.1 引 言.....	1
1.2 国内外研究现状.....	2
1.3 本文研究意义.....	3
1.4 研究的目的和内容.....	4
2 海洋测深的技术和方法.....	6
2.1 海洋测深空间结构.....	6
2.2 深度基准面.....	9
2.3 潮汐影响.....	10
3 水深异常数据的检测方法.....	14
3.1 水深数据编辑的理论基础.....	14
3.2 单波束水深数据编辑.....	16
3.3 多波束水深数据编辑.....	16
3.4 数字图像处理中的滤波方法.....	25
4 基于密度投影的多波束异常数据自动检测.....	45
4.1 基于密度投影的多波束异常数据自动检测.....	45
4.2 实际数据分析.....	70
5 结论与展望.....	79
5.1 研究成果.....	79
5.2 展望.....	79
致谢.....	81
攻读硕士期间主要成果.....	82
参考文献.....	83

Contents

1 Preface.....	1
1.1 Introduction.....	1
1.2 Research in home and abroad.....	2
1.3 Research Significance of the Paper.....	3
1.4 Structure of the Paper.....	4
2 The Technology and Method of Marine Sounding.....	6
2.1 Basic Space Structure of Marine Sounding.....	6
2.2 Depth Datum.....	9
2.3 The Influence of Tidal.....	10
3 The Methods of Automatic Detecting Outliers.....	14
3.1 The Editing Principles of Depth Data.....	14
3.2 The Editing Methods of Single Beam Data.....	16
3.3 The Editing Methods of Multibeam Data.....	16
3.4 The Filtering Method of Digital Image Processing.....	25
4 Automatic Detecting Outliers of Multibeam Data based on Intensity Projection.....	45
4.1 Automatic Detecting Outliers of Multibeam Data based on Intensity Projection.....	45
4.2 Practical Data Analysis.....	70
5 Main Research Results and Conclusion.....	79
5.1 Main Research Results.....	79
5.2 Conclusion.....	79
Acknowledgements.....	81
Main Work Achievement of the Author during Working on Master Paper..	82
References.	83

1 绪 论

1.1 引 言

面积占地球 70%的海洋是现代社会的交通要道、人类社会的资源宝库，更是人类生命得以延续的基本保证。随着地球人口的急增，环境的恶化，陆上资源的加速枯竭，今天海洋已成为人类生存和发展的重要空间，是人类生命支持系统的重要组成部分，可持续发展的宝贵财富。

海洋测量是一切海洋活动的基础，是人类认识海洋的主要方式。海洋测量是一门综合性很强的学科，包含海底地形测量、海道测量、海洋大地测量、海洋重力测量、海洋磁力测量、海洋各种工程测量和海洋调查测量等，是对整个海洋空间，包括海面水体和海底进行全方位、多要素的综合测量，以获取包括大气(气温、风、雨、云、雾等)、水文(海水温度、盐度、密度、潮汐、波浪、海流等)以及海底地形、地貌，地质、重力、磁力、海底扩张等各种信息和数据并绘制成各种使用用途的专题图件，为经济、军事和科学服务的基础科学。海洋测量技术在海洋测量手段和方式日益多样化的前提下获得了迅速的发展，在建设海洋工程、开发海洋资源、发展海洋科学、维护海洋权益等方面都发挥着极为重要的作用^{[1][2]}。

而海底地形测量则是一切海洋工程活动的基础，它最基本的任务是测量海水的深度，测绘海底地形图(水深图)。作为人类开发和利用海洋的工具，海底地形测量发挥着十分重要的作用。主要表现为^[3]：(1)世界海洋调查(2)海洋工程建设(3)海洋资源开发(4)海洋科学研究(5)海洋权益维护。

水深测量是在动态的条件下进行的数据采集，受大气运动、海水物理性质、海洋生物活动及测量传感器的各方面的影响，造成勘测的水深数据中包含许多误差。水深测量误差具有显著的多源性，引起这些误差的因素主要分为两类，即仪器内部因素引起的误差和外部环境因素引起的误差。其中内部因素引起的误差是指系统内部各种传感器自身产生的误差，主要包括姿态测量误差、定位系统的误差等。而外部因素引起的误差则是指在系统测深过程中因海水介质的变化运动和探测基面的变化等所引起的影响测深的误差。由于海洋测量的动态性，无论是单波束测深系统，还是先进的多波束测深系统都不

可能采用重复测量来检测异常数据并提高精度，所以，测深系统进行水下测量时，需要进行一系列的参数校正以及水深数据的改正。单波束及多波束测深系统工作示意图如图 1.1 所示。

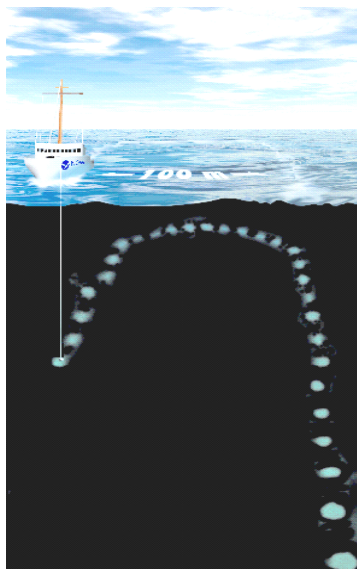
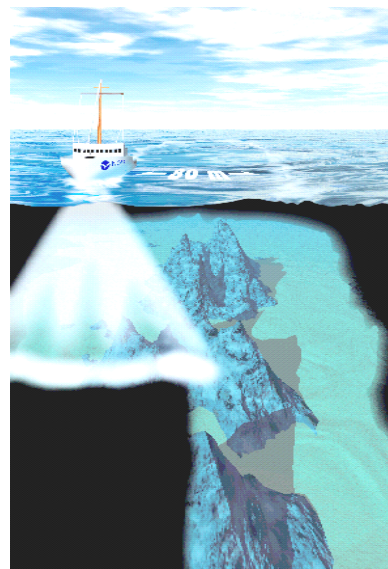


图 1.1(a) 单波束测深系统工作示意图

Fig. 1.1(a) the single beam system working
schematic diagram



(b) 多波束测深系统工作示意图

(b) the multibeam system working
schematic diagram

1.2 国内外研究现状

测深异常数据直接影响着水深数据对海底真实情况的反映，所以需要对测深数据进行异常数据的剔除。传统的数据编辑的模式可划分为两类，即交互式编辑和自动编辑。

交互式编辑是指通过显示的测深数据，根据海底地形是连续平稳变化的特点，利用人的经验知识直观的判断异常数据，手工进行标注、剔除异常数据的方法。关于交互式编辑的方法，通常是采用投影法，即先将数据沿测线前进方向正投影、正交测线方向投影或垂直正投影，然后在投影面上手工对数据进行编辑，剔除异常数据^[4]。人工交互式滤波方式对有明显错误的异常值的判断比较成功，国外最初在进行数据处理的时候，大多数采用的就是这种方式，但是这种方式具有效率低的明显缺点，特别是处理大量数据时，这种缺点尤为明显，并且编辑结果因操作者的不同而各异且不能重复。因此不适用于大批量、多波束数据的处理的要求，因此研究快速、可靠、自动的异常数据定位方法

的要求越来越迫切。

自动编辑则是指根据测深数据的特点,利用一定的判定准则,自动区分测深数据中的异常值和正常值,进行异常数据检验的方法。在海底地形连续平缓变化的假设下,国内外学者在自动异常数据的检测方面取得了很多成果。目前,自动异常数据编辑方法主要分为两类,一类是基于经典统计理论的误差统计检验法,另一类是基于函数或统计推值的比较判别法。即在一定区域范围内用经典的统计量,如标准差、方差、均值,设置门限对测深数据进行统计,识别出不符合一定标准的数据^{[5][6]};或者是通过经典滤波的方法,如中值滤波^[7]和卡尔曼滤波来平滑数据^{[8][9]};还有就是采用趋势面分析的理论,即用最小二乘法、多项式曲面函数、*Bezier*方法和 *B*样条方法等方法进行曲面拟合,通过真实测深数据和拟合测深数据的比较判别异常数据^{[10][11][12]}。由于测深数据中通常含有粗差,上述的方法都是依据最小二乘的原理,其估值是有偏的,所以黄谟涛^[11]等人尝试应用现代抗差估计理论来提高海洋测量异常数据统计检验的抗差能力,提出一种基于抗差估计的选权迭代插值比较检验法。近两年出现了比以上处理方法更先进的方法,*Brian Calde*提出了 *CUBE* (*Combined Uncertainty and Bathymetry Estimator*) 算法,该算法通过对立方格中多个深度测量值的不确定性分析来推演该位置的真实水深,标识异常数据^{[13][14]}。这种方法的不同之处是,根据测深数据的不确定性生成一个代表真实深度的最优的海底表面模型,而不是根据个别的测深数据生成海底模型。因此操作者只需要检查区域里正确的假设值是否被选择,所以这种处理方法大大减少了处理的时间,简化了操作者的工作步骤。操作者能够利用他的空间分析能力确保正确的海底面被选择,而不是运用他的鼠标来处理多波束数据。实验数据表明,其编辑处理消耗的时间与传统的交互式编辑方法的比例是 1:25-37,数据后处理编辑的效率大大提高。

总之,水深测量的复杂性,测深误差的多源性,使得测深数据中包含大量的异常数据,所以研究稳健、快速、自动的测深异常数据的检测是目前研究的热点。

1.3 本文研究意义

测深系统及实际海况条件的复杂性,造成测量的水深数据中包含有很多误差。这些误差概括起来可以分为两大类——深度误差和位置误差。造成这些误差的原因主要有:

①海况的影响，即海水声速结构的变化、海水潮汐变化、海水风浪变化、海水中浮游生物群、海水气泡混响以及海底表面散射条件等的影响。②组成多波束测深系统传感器的测量误差，如船的姿态测量误差、声速剖面误差、定位系统的误差等。③多波束测深系统本身产生的测量误差。

上述影响测深数据的这些误差，如内部因素引起的误差有一定的规律性可以通过系统分析加以改正或消减，而还有一些误差为随机误差，没有规律，如海洋噪声、仪器噪声等因素，这些噪声使得测深数据中包含大量的异常数据。这些异常数据的存在，将会导致假信号，产生虚假地形。由于水深测量的不可见性，为了提高测量学成果的可靠性，必须对测深数据进行编辑，滤除这些虚假信息，为后处理成图做准备。

1.4 研究的目的是内容

1.4.1 研究目的

海底测深数据的准确性，直接影响海底真实地形的反映，对水深测量数据进行编辑处理，获得高质量的海底地形地貌数据，是人类认识海洋资源利用海洋资源的首要步骤。本文所研究的测深异常数据的自动检测就是针对这一目的而提出的。

1.4.2 研究内容

本文主要从水深测量中的异常数据自动处理方面展开研究，针对现有的异常水深数据自动提取的算法，总结其各种方法的优缺点的基础上，结合实际水深数据特点，提出了一种基于密度投影的区域生长异常数据自动检测法。

本论文主要包括 5 个部分。

第一章，绪论，主要介绍了论文的研究目的和意义。

第二章，论述了水深测量的基本技术和方法。

第三章，概述了一些经典的水深数据滤波的方法，然后根据本文内容介绍了数字图像处理中相关的图像滤波的方法。

第四章，算法设计与实现。针对多波束测深数据的分布特征，基于数字图像滤波的方法，提出了一种自动确定真实水深，剔除异常水深值的算法。该算法可以在投影面上

自动搜索正常数据区域，剔除异常水深。最后，结合实际水深数据，利用 *VC++ 6.0* 开发工具编辑了该算法，验证了所提出的算法的可行性及优点。

第五章，总结全文，总结了论文所做的工作，提出了目前存在的不足，并对今后的研究工作做出了展望。

2 海洋测深的技术和方法

水深测量，是通过声波在水介质中往返传播的时间及其在水介质中的传播速度，来计算水深值的。公式如下^[4]：

$$H_o = \frac{1}{2} Ct \quad (2.1)$$

式中， H_o 为换能器与海底之间的水深， C 为波束在海水中传播的平均速度， t 为波束往返传播的旅行时。

2.1 海洋测深空间结构

海洋测深的本质是确定海底表面至某一基准面的差距。目前世界上常用的基准面为深度基准面、平均海面 and 海洋大地水准面。前一种是指按潮汐性质确定的一种特定深度基准面，即狭义上的深度基准面，这也是海洋测深实际用到的基准面。

2.1.1 海洋测深空间结构

海洋测深基本空间结构示意图^[3]，如图 2.1 所示：

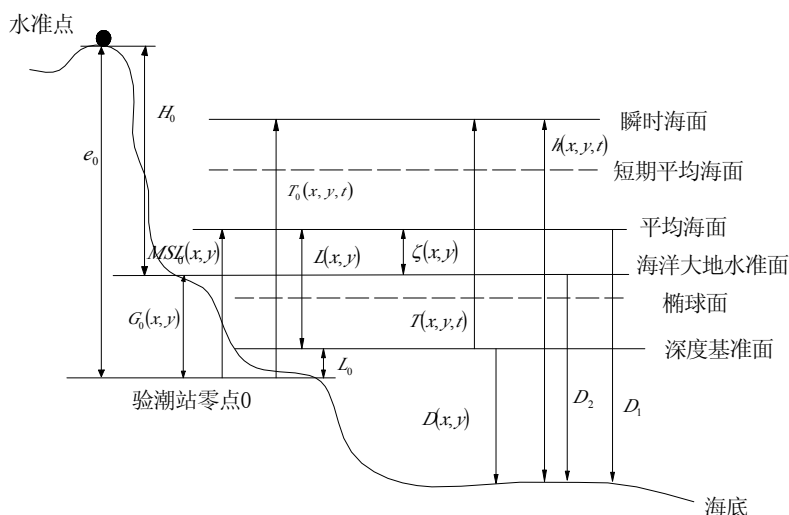


图 2.1 海洋测深基本空间结构示意图

Fig. 2.1 the basic space structure chart of marine survey

以深度基准面为基准的海底深度为：

$$D(x, y) = h(x, y, t) - T(x, y, t) \quad (x, y) \in s \quad (2.2)$$

式中： x, y 为平面位置坐标， s 为测区位置， t 为时间参数；

$D(x, y)$ 为从深度基准面起算的海底深度值，即深度基准面与海底表面的差距；

$h(x, y, t)$ 为瞬时测深值，即瞬时海面与海底表面的差距，通常由海上船只有水声学原理直接测得，为直接观测值；

$T(x, y, t)$ 为瞬时潮位高度值，即瞬时海面与深度基准的差距。由验潮站的潮位观测来获得，或由随船的仪器获得。这是唯一精确标定瞬时海面起伏的观测值。

上式表明，只要已知 $h(x, y, t)$ 和 $T(x, y, t)$ ，便可以确定出 $D(x, y)$ 。

2.1.2 海洋测深基本空间的时空特性

海洋测深所涉及到的海洋界面的特性是不一致的，各有其不同的时空特性^[3]。海洋测深中各界面示意图，如图 2.2 所示。

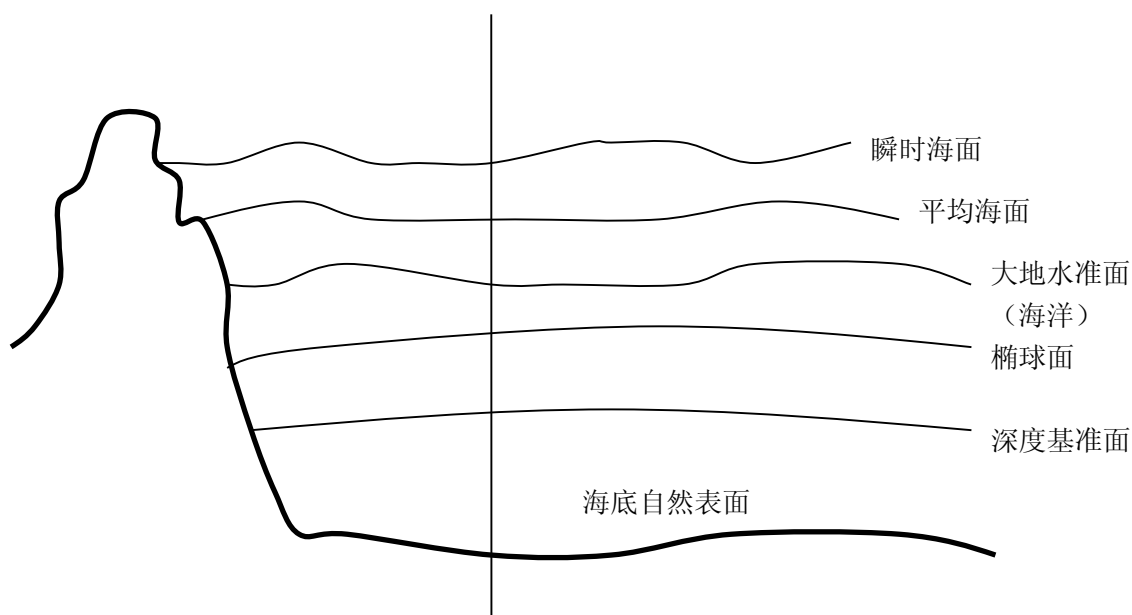


图 2.2 海洋测深基本界面示意图

Fig. 2.2 the basic space planes chart of marine survey

1、时变特性

瞬时海面，由于受潮汐、波浪等因素的影响，每时每刻都处于变化中。

平均海面，因为滤除了一些短周期分潮的影响，其稳定性增强，变化的幅度与取平均期间的长短有关、取平均的时间越长稳定性越好。通常，潮汐学认为 18.61 年是一个重要的会合周期，18.61 年（或 19 年）的平均海面可以认为是消除了各种波浪、潮汐等因素的平均海面。我国高程基准确定时，就采用了这种平均海面。该平均海面，在海洋测量工作期间，可认为是不随时间变化的常量。

海洋大地水准面是国家大地水准面的一部分，在海洋测量工作的短期间内，仍可认为是不随时间变化的常量。

深度基准面是根据潮汐调和常数计算得出的。一般来说，潮差越大，其与平均海面之间的差距值 $L(x, y)$ 越大，深度基准面的位置越低。在海洋测量工作期间，可认为某一地点深度基准面为常量。

海底地形的表面，虽然由于地壳的运动及海水腐蚀等的影响，存在着缓慢的变动，但是在测深工作期间，可认为海底表面是稳定不变的。

这样，在海洋测深中主要涉及的五个海洋界面（瞬时海面、平均海面、深度基准面、海洋大地水准面、海底地形的表面），只有瞬时海面随时间 t 变化，而其它界面可认为是不随时间 t 变化的，为常量。

2、区域特性

实际工作中，除了在沿岸及岛屿上可以设定一定数目的验潮站外，在广阔的海面上却无法设立永久的验潮站。另外，在测深过程中，还需要实时测定出任一测深点处的瞬时海面高度值。然而不可能在海面上任一点都设立验潮站，这就需要利用已知的验潮站的界面信息内插出测区内任一点处海洋界面的信息，这实际上就涉及到各种海洋界面的区域分布形态。

(1) 平均海面、深度基准面、海洋大地水准面的区域变化特性

与陆地地形测量相似，对这三种测深基准面分布形态的约束，是通过缩小测区范围来实现的。在每一测区范围内，使得这三种测深基准面为线性分布或为常量。

(2) 瞬时海面的区域变化特性

与上述三种基准面不同，瞬时海面主要受潮汐、波浪等短期因素的影响，分布起伏大，区域内的形状不能认为是常量或线性分布，需要根据潮汐区域传播性质来建模内插

除任意点的 $T(x, y, t)$ 。

2.2 深度基准面

水深测量活动是在起伏不定的运动海面上进行，同一海底目标在不同时间测量的水深是不同的，这会给海图使用者带来极大的不便。为此，须将变化的水深归算到一个相对稳定(或固定)的基准面，称为深度基准面。

深度基准面的确定既要考虑到船舶航行的安全，又要考虑到航道或水域水深资源的利用效率。为此海图深度基准面确定方法要有科学性并在计算方法上力求统一^[3]。深度基准面以上的水深就是潮位，在实测水深中减掉了潮位，并进行了其他改正项的修订，就得到了实际水深。

目前实际上有以下几种方法确定深度基准面^[4]：

- 1、最低天文潮面
- 2、平均大潮低潮面
- 3、最低大潮低潮面
- 4、平均低潮面
- 5、平均低低潮面
- 6、略最低低潮面

$$H_d = MSL_0 + H_{M2} + H_{K1} + H_{O1} \quad (2.3)$$

上式中， MSL_0 为平均海面，后面为 $M2$ 、 $K1$ 、 $O1$ 分潮的振幅。

- 7、理论深度基准面

$$H_d = MSL_0 + \min[K_1 \cos \varphi_{K1} + K_2 \cos(2\varphi_{K1} + 2g_{K1} - g_{K2} - 180^\circ) - (R_1 + R_2 + R_3)] \quad (2.4)$$

上式的第二项里的系数为 8 个分潮的各种组合值，由潮汐方法计算出。

- 8、近最低潮面

$$H_d = MSL_0 + (2.341 - 0.913S + 0.764K)\delta \quad (2.5)$$

上式中， δ 为分潮的标准差， S 为分潮的偏度， K 为分潮的峰度。

50 年代初期,我国采用略最低低潮面。由于它在大潮低潮期间有时高于实测的低潮面,1954 后,在正规半日潮和不正规半日潮海区也曾在此面上增加了 k_2 分潮和 N_2 潮的改正。1957 年后,我国采用前苏联的可能(理论)最低潮面作为海图深度基准面。1977 年中国科学院海洋研究所方国洪和国家海洋局海洋情报研究所王骥等提出了“近最低潮面”,用概率统计的方法计算海图深度基准面。在多波束进行海洋海洋大地测量中,我们一般用可能(理论)最低潮面作为理论深度基准面^[3]。

一般来说,潮差越大,深度基准面与平均海面的差距值越大,其位置越低。

2.3 潮汐影响

潮汐使得瞬时海面发生变化,要获得准确的海洋测深就必须消除潮汐的影响,即进行潮汐改正。为了正确的表示海底地貌,就要把瞬时测得的深度化为从规定的深度基准面(我国采用理论深度基准面)起算的深度,这个过程中需要进行潮汐改正。

2.3.1 潮汐改正原理

潮汐改正的原理,见图 2.3。

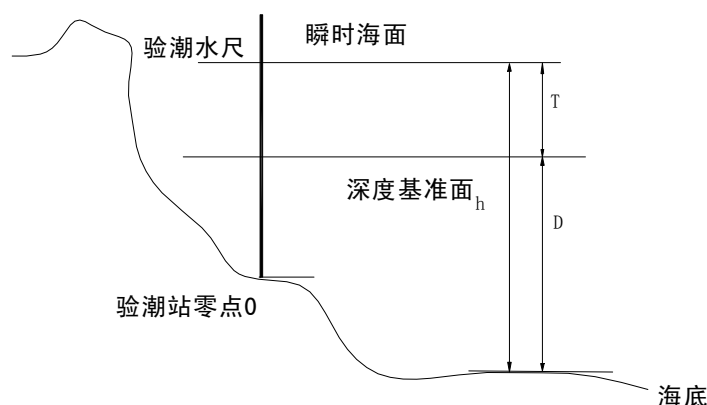


图 2.3 潮汐改正

Fig. 2.3 the tide adjustment

在测区某一点,瞬时水深值为 h , 从深度基准面起算的水位(潮汐)高度为 T , 则从测深基准面起算的水深值 D 应为^[4]:

$$D = h - T \quad (2.6)$$

T 通常由水位观测（或由瞬时海面形状数学模型内插）求定，其深度基准面的数值通常由潮汐调和常数理论计算求定。

2.3.2 潮汐观测方法

潮汐观测方法主要由水尺验潮、浮子式与引压钟式验潮仪、声学式验潮仪、压力式验潮仪、GPS 验潮和潮汐遥感测量等方法^{[16][17]}。

水尺验潮是将特制的水尺安装于水中，在码头可直接安装在港池壁上，在野外一般要竖一木桩，再将水尺固定在桩上。水尺验潮是最古老但又是最简易的验潮方法，同时也是短期、临时性质验潮中普遍采用的方法。

浮子式与引压钟式验潮仪均属于有井验潮仪。井式验潮方法一般适合固定于岸边（港口、码头等）的观测站进行长期潮位观测。它的滤波方法属于机械性滤波（即利用硬件设备实现滤波），即采用小孔阻尼滤波或长管阻尼滤波。

声学式验潮仪属无井验潮仪，根据其声探头（换能器）安装在空气中或水中而分为两类。探头安置在空气中的声学式验潮仪是在海面以上固定位置安放一声学发射接收探头，探头定时垂直向下发射超声脉冲，声波通过空气到达海面并经海面反射返回到声学探头，通过检测声波发射与海面回波返回到声探头的历时来计算出探头至海面的距离，从而得到海面随时间的变化。由于其安装位置可距海面较近，声波在空气中的行程短，因此精度较高。探头安置在水中的声学式验潮仪是将一声学探头安放在海底，定时垂直向上发射声波，并接收海面的回波以测量安放点的水深，或者采用类似于测深仪的原理，选一块平坦的海区，将声探头放置于海面固定载体上，定时向海底发射声波，通过检测海底回波以检测载体所在位置的水深，精度较低。

压力式验潮仪是将验潮仪安置于水下固定位置，通过检测海水的压力变化而推算出海面的起伏变化。

GPS 验潮分为船载 GPS 验潮和 GPS 浮球验潮。船载 GPS 验潮思想是利用载波相位差分技术进行高精度定位，从而获得精确的船载 GPS 接收机天线相位中心的高程，并根据高程的变化，结合一定的几何关系，获得瞬时海面高程，从而达到验潮的目的。GPS 浮球验潮系统也是由基准站和流动站两部分组成。其中基准站的组成同船载 GPS 验潮系统相同，但流动台站不相同，其载体变成了浮球。值得注意的是，GPS 测出的是其在 WGS-

84 椭球中的位置，与以往所采用的验潮仪验潮有所不同，验潮仪所测出的是相对与海底的水深，显然 GPS 验潮不但包含了海水的潮汐变化，还包含了地壳的固体潮。前面已提到在引潮力的作用下，地壳的起伏可达十几至几十厘米，因此在采用 GPS 验潮时，要设法修正和减小固体潮的影响。

现代潮汐测量的手段很多，其中目前较为常用的是声学式和压力式验潮仪^[16]。

2.3.3 验潮站的有效作用距离

计算验潮站的有效作用距离，对合理布设验潮站以及决定水位改正的方法，有着重要的意义。验潮站的有效作用距离取决于水深测量的精度、验潮站与水深点上瞬时（同一时间）的最大潮高差。根据测深区附近的已有两个验潮站的潮汐调和常数计算其间的瞬时最大潮高差，并按两个验潮站的距离计算测深精度相对应的距离，即为按测深精度要求的验潮站的有效作用距离^[2]。海道测量中验潮站有效距离^[4]：

$$d = \frac{\delta_z S}{\Delta h_{\max}} \quad (2.7)$$

其中， d 为验潮站有效距离(km)， δ_z 为测深精度(cm)， S 为两站之间的距离(km)， Δh_{\max} 为两站在同一时间的最大可能潮高差(cm)。

利用（2.7）估计有效距离，关键是求 Δh_{\max} ，通常有三种方法^[1]：

- 1、同步观测比对法。根据两站同步观测资料，绘出大潮期间几天的水位变化曲线(从平均海面起算)，从图上找出 Δh_{\max} ；
- 2、解析计算法。利用两站的 4 个主要分潮构成的准调和潮高模型，计算出 Δh_{\max} ；
- 3、数值计算比对法。利用两站的 11 个分潮构成的调和潮高模型，计算两站一段时期的潮高值，选出 Δh_{\max} 。

2.3.4 潮位的改正

当测区的范围不大，在一个验潮站的有效范围内，直接利用该站的水位观测资料改正。当范围较大时，超过一个站的控制范围，采用两个站的水位观测资料分带改正。当两个站不能有效控制时，采用三个站的三角分带法改正。

3 水深异常数据的检测方法

水深测量是在受波浪影响的、有潮起潮落的海面上进行的，受风浪、潮汐、海水结构等海况因素的影响，其测量都往往具有一次性的结果，即不能重复测量。其测深过程中，由于各种参数设置的不正确、海况因素、仪器噪声等的影响，可能产生一些异常数据，形成虚假地形。这些虚假地形的存在影响了真实海底地形的反应，所以需要在后处理的水深数据编辑中进行剔除。目前水深数据处理的方法主要有两种：人工交互式检测和自动异常数据检测。人工交互式检测，通常采用投影法，即在测线前进方向投影、正交测线方向投影或垂直正投影平面上对水深数据进行人工的编辑检测，其优点是直观性好，可靠性高，但主观性强、效率很低，并不适用于目前密度很大的多波束测深数据^[4]。自动异常数据检测，适用于大批量的多波束处理的要求，能够快速、可靠、高效的定位异常数据，可以满足目前各种各样的海洋测量的要求，是目前研究的热点。

3.1 水深数据编辑的理论基础

3.1.1 水深数据编辑的理论基础

1、地形是连续变化的

真实海底地形是连续变化的，即使有海底峡谷或海山，也是由平坦海底过渡而来，因此测深系统测得的数据也应该反映出连续的地形变化。如果出现孤立、突变的数据点，则可肯定是噪声，可通过地形趋势面的判断或分析拟合，采用交互式编辑或自动编辑方法去除噪声点。

2、相邻测线对比判断边缘变化数据

将相邻测线放在一起进行编辑，可以了解地形全貌及变化趋势，很容易区分噪声和真实地形数据。因此，对一些在条带边缘的突变数据区，必须比对相邻条带数据。

3、中央波束精度高

多波束勘测系统，其波束开角越来越大，可高达 150° 。而声速剖面的不准确，使得边缘波束质量较差，甚至出现与航迹线方向一致的假地形。另外，边缘波束声程路径比中央波束要长，回波信号能量较中央波束弱，信噪比降低，故相对于边缘波束，中央波

束的质量是比较可靠的，对于中央波束与相邻条带边缘波束重叠区域，应以中央波束为准。

4、水深处于一定范围内

在一定区域勘测时，水深肯定处于一定的范围内，如果了解这个测区水深大致范围，则可通过设置阈值将超过这个范围的数据点全部自动滤除。

3.1.2 常规统计检测法

常规统计检测方法通常以每个波束点为中心，计算该波束点邻域内所有波束的统计量，从而判断该波束点是否为异常值。通常的统计量有水深平均值、水深中间值。

1、通过水深平均值和标准差判断异常数据

如果该波束点水深与平均值之差在 3 倍标准差外，则认为是异常点，否则为正常信号。

$$\left. \begin{aligned} (f(x_i, y_i) - \bar{f}) &\leq 3\sigma && \text{正常信号} \\ (f(x_i, y_i) - \bar{f}) &> 3\sigma && \text{异常数据} \end{aligned} \right\} \quad (3.1)$$

式中， $f(x_i, y_i)$ 为点 (x_i, y_i) 处的水深， \bar{f} 为邻域内所有波束平均水深， σ 为标准差。

式 (3.1) 成立的基础是邻域内波束水深成正态分布，而实际海底地形不一定如此。另外，异常数据也参与了 σ 的计算，使得 σ 含有偏差，故该方法可靠性不强，在异常数据点较多时可靠性急剧下降。

2、通过水深中间值和比例系数判断异常数据

如果该波束点水深与邻域内中间值之差在中间值的固定倍数外，则认为是异常点，否则为正常信号。

$$\left. \begin{aligned} (f(x_i, y_i) - f_m) &\leq a \cdot f_m && \text{正常信号} \\ (f(x_i, y_i) - f_m) &> a \cdot f_m && \text{异常数据} \end{aligned} \right\} \quad (3.2)$$

式中， f_m 为邻域内所有波束排序后的中间水深， a 为 0 到 1 之间的固定系数。这种方法适用于非平坦海域，且水深范围未知^[18]。

3、通过水深中间值和固定值判断异常数据

如果该波束点水深与邻域内中间值之差在固定数外，则认为是异常点，否则为正常信号。

$$\left. \begin{aligned} (f(x_i, y_i) - f_m) &\leq c && \text{正常信号} \\ (f(x_i, y_i) - f_m) &> c && \text{异常数据} \end{aligned} \right\} \quad (3.3)$$

式中, c 为固定的常量。相对于前一种方法, 这种方法适用于非平坦的深水海域, 且水深范围未知^[18]。

3.2 单波束水深数据编辑

单波束水深测深仪, 在实际工作中, 波束开角一般为 $0^\circ \sim 30^\circ$, 每次仅能测量一个“点”, 仅能完成航迹线上的水深测量, 其测深数据沿航迹线水深数据密集, 而在测线之间却没有数据。它仍然是一种非全覆盖的测量方法, 造成测线间可能存在的航行障碍物和特征地形的漏测。

常用的单波束滤波方法为门限法, 即采用设置深度门来消除粗差^[3]。

门限法, 即在每条测线的起点首先选定一个可信得水深值, 以此为标定值来设定深度门, 将后面的新测深度与门限相比较, 假如落入门内, 则认为是合格。再以此深度为标定值, 重新设立深度门, 检验后面的水深值。如果新测的水深值在固定门限外, 则在该深度处设立一个临时门, 假如有数个深度值通过临时门, 则会被认为是合格水深, 否则设定为临时门处的数据将被舍弃, 即被滤除。

门的长宽值是可变的, 需根据海底起伏情况确定。如果门太窄或太长, 海底的详细信息就会被忽略; 相反, 门太宽或太短, 就会保留不合格的水深。因此, 需要设定恰当的门限, 以确保数据的有效滤除。

3.3 多波束水深数据编辑

3.3.1 多波束数据格式分析

1、多波束数据格式

多波束测深系统, 是由多传感器组成的水深测量系统, 其原始测量数据包含的信息相当丰富, 包括多波束本身提供的信息和测量船的实时姿态信息两个方面。

具体来说, 多波束原始测深数据通常都包括:

(1) 水深测量的基本信息: Ping 号、波束号、经度、纬度、水深值;

- (2) 测量船的位置信息：纵摇、横摇、吃水、航向、速度；
- (3) 中央波束的信息：中央波束北坐标、中央波束东坐标、侧向距、航向距；
- (4) 时间信息及其他：年、日月、小时、分、秒、旅行时、角度。

这些数据提供了确定水深点的位置和深度的信息，每个波束产生时瞬时船姿变化的数据等。不同的仪器采集的多波束数据的格式是不同的，但总体上都包含上述这些信息。本文采用的数据格式如图 3.1 所示：

```
Ping, 波束号, 经度, 纬度, 水深, Pitch, Draft, Roll, Heave, Draft, 速度, Heading, 侧向距, 航向距, 旅行时, 波束角,
1, 30, 114.9696740, 13.609788, 3490.0, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -3324.600, 24.100, 6.41300, 45.00
1, 31, 114.9712883, 13.609809, 3461.8, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -3149.900, 24.000, 6.22700, 43.64
1, 32, 114.9717650, 13.609816, 3461.0, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -3098.300, 24.000, 6.18100, 43.15
1, 33, 114.9729007, 13.609831, 3452.6, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2975.400, 23.900, 6.06400, 42.02
1, 34, 114.9734652, 13.609838, 3444.4, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2914.300, 23.900, 6.00300, 41.49
1, 35, 114.9749630, 13.609860, 3470.6, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2752.200, 24.100, 5.89300, 39.58
1, 36, 114.9756154, 13.609868, 3450.3, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2681.600, 24.000, 5.81400, 39.00
1, 37, 114.9765255, 13.609883, 3476.8, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2583.100, 24.200, 5.76200, 37.71
1, 38, 114.9772583, 13.609892, 3455.0, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2503.800, 24.100, 5.67700, 37.00
1, 39, 114.9787700, 13.609910, 3406.6, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2340.200, 23.800, 5.49900, 35.52
1, 40, 114.9789889, 13.609915, 3434.3, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2316.500, 24.000, 5.51200, 35.00
1, 41, 114.9801070, 13.609929, 3423.4, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2195.500, 23.900, 5.41100, 33.63
1, 42, 114.9806577, 13.609937, 3431.0, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2135.900, 24.000, 5.37700, 32.83
1, 43, 114.9813553, 13.609947, 3432.6, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2060.400, 24.000, 5.32600, 31.87
1, 44, 114.9818644, 13.609954, 3432.8, 0.00, -1.40, -0.55, 0.00, 29.86, 359.20, -2005.300, 24.000, 5.28900, 31.17
```

图 3.1 原始多波束数据格式

Fig. 3.1 the raw multibeam data format

这里的水深数据值是已经经过横摇、纵摇、吃水、声速改正等实时处理后的水深值，代表了每个探测波束的具体位置。后文中的水深数据处理就是基于此种格式的数据。

2、多波束数据特征

与传统的陆地测量方式相比，多波束测深数据的特点是：

(1) 数据海量

多波束系统通常采用几十个甚至上百个电子独立的波束，发射和接收的更新率很快，如EM3000、SeaBat8111等每秒可达20余次。因此，每次所采集的海底数据量很大，根据测区面积的不同需以几十甚至几百G来存储。

(2) 扫描密集

多波束系统发射和接收船龙骨方向两侧的几十个至几百个独立的波束，在海底形成一个声照射带。一个测幅(Swath)的宽度可达到水深的1-12倍，所以说扫测点密集，可实现海底的全覆盖测量。

(3) 内容丰富

多波束系统不仅可以获得海底高精度的、高分辨率的水深地形数据，还可同时接收

和处理声反向散射数据，获得类似侧扫声纳的海底声像图。而且，还可以利用多波束系统采集的水深和反向散射数据进行海底底质分类。目前，多波束勘测技术已成为集海底地形、地貌和底质类型测量为一体的综合性手段。

(4) 噪声多源

海洋本身复杂的物理性质，使海洋测量比陆地测量具有更多的噪声干扰源。仪器噪声、海况因素或仪器参数设置不合理等因素，使得测深数据中包含大量的异常数据。这些异常值直接影响着测深数据对真实海底的反映，所以，必须对数据进行编辑处理。

3.3.2 基于多波束特征的异常数据检测法

该法主要与多波束勘测系统的特征有关，主要有 3 种算法：

1、直接剔除边缘波束

由于多波束勘测系统的边缘波束的质量通常较差，如果系统有一定的覆盖，则可删去处于覆盖区的边缘波束。图 3.2 为原始多波束水深数据，图 3.3 为在两边各剔除外部 5 个波束的结果。可以看出，边缘波束有不少是异常数据。

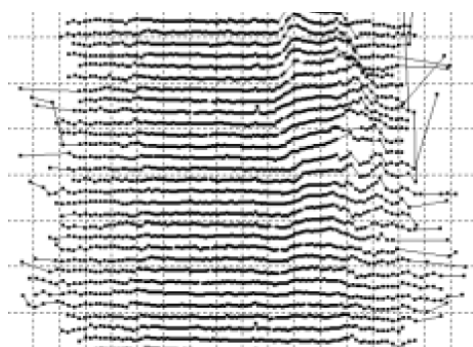


图 3.2 原始多波束水深数据

Fig. 3.2 the raw multibeam data

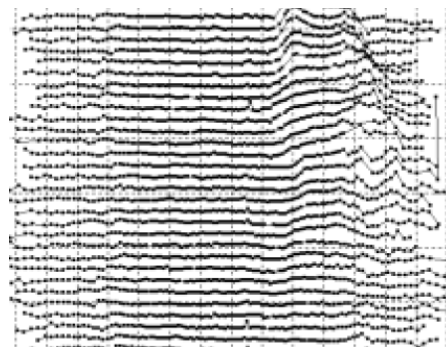


图 3.3 在两边各剔除外部 5 个波束的结果

Fig. 3.3 the final result after 5 beams filtering

2、脊状假地形

对于连续的 ping 的同一波束号的波束点，如果出现较其他波束浅许多或侧向距相对其他波束也小许多，在三维地形图上表现为与航向平行的脊状假地形，则判断为异常数据并剔除。

3、单 ping 有太少的正常数据

如果某 ping 数据的异常数据太多，达到一定阈值，则认为该 ping 数据不可靠，全部剔除。

3.3.3 人工交互式异常数据检测

根据海底地形连续变化原则，采用交互式的投影编辑方法，将每条带或条带子区多波束数据投影到3个平面视图上，包括测线前进方向正投影、正交测线方向侧投影和水平投影。为了减少条带中地形较大变化所带来的误差，可根据条带的长度，对长的条带即剖面数或ping数多的条带进行分区，再进行投影。

1、沿测线前进方向正投影编辑

测线前进方向正投影，就是把水深点投影到与测线正交的平面上，对整 ping 异常的数据比较容易识别。

沿测线前进方向进行投影，可以方便地判断海底地形区域，从而去除此区域外的噪声。如图 3.4，显示有两个条带的数据，相邻条带数据是为了对比。左边绿色条带数据在红色多边形内的数据，可认为是噪声。经过编辑后，最终得到投影面上光滑连续的海底地形。

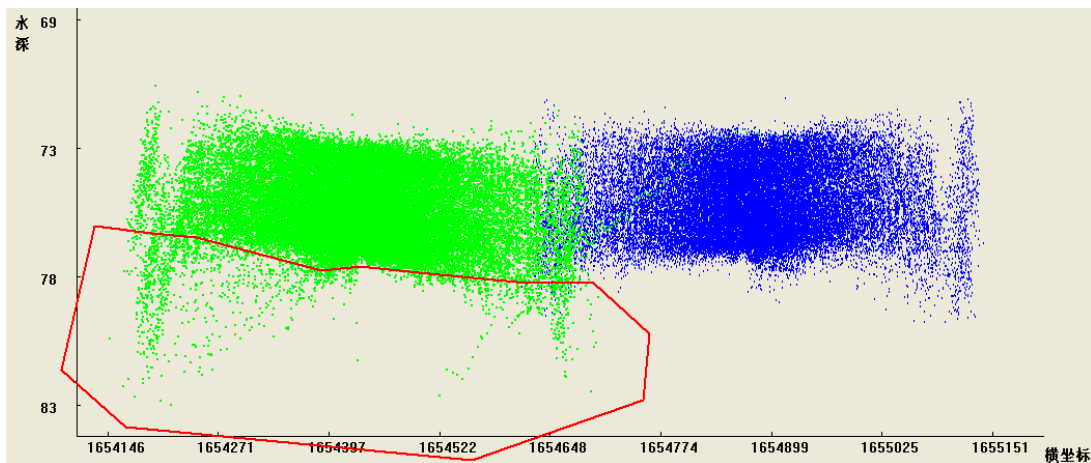


图 3.4 垂直正投影平面上的数据编辑

Fig. 3.4 the raw projection in orthogonal direction

在此投影方式下，可以选择单 ping 编辑方式，类似于单波束剖面编辑方式。优点是能发现被条带正常水深区域隐蔽的噪声，缺点是速度慢。

这种投影方法适用于海底比较平坦的区域，而对于海底地形起伏较大、变化复杂的区域，因判断海底地形区域很困难，故需采用其他投影方法^[11]。

2、沿正交测线方向侧投影编辑

正交测线方向侧投影以 ping 号为横轴，水深为纵轴，在投影面上水深数据是以一个个波束的形式显示的，能很好展现沿测线前进方向的海底变化趋势，能够在海底地形变

化复杂海区，即在测线前进方向投影方式下不能很好编辑的情况下，能有效地进行编辑^[11]。如图 3.5，海底地形区域比较容易判定，下部多边形区域内的红色的数据点，可认为是噪声，应去除。

在此投影方式下，也可选择该条带的所有 ping 的某个波束或多个波束进行编辑，优点是能根据剖面连续变化的原则可判定处于条带正常水深范围内的噪声，缺点是速度较慢。

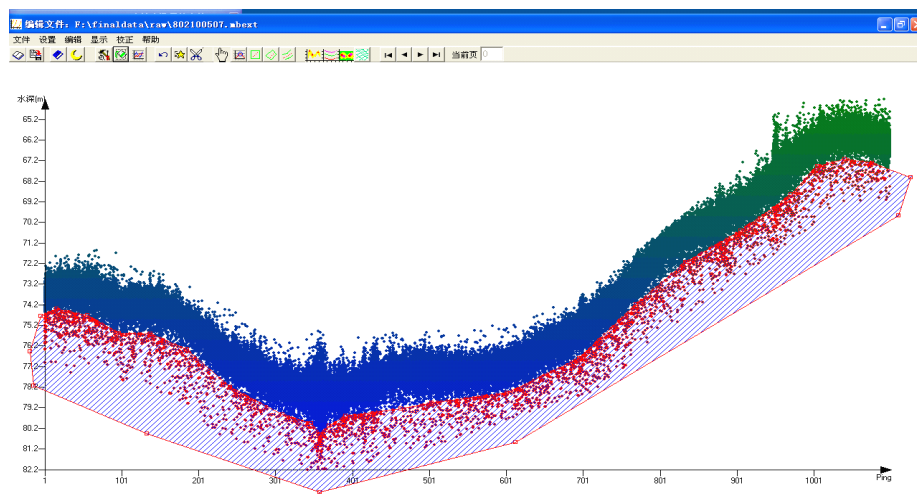


图 3.5 正交测线方向侧投影平面的数据编辑

Fig. 3.5 the raw projection in side direction

3、水平投影编辑

在海底地形变化极其复杂的区域，需要在水平投影方式下进一步编辑。其投影方式就是把测深数据按经度、纬度坐标或平面坐标投影到水平面上。如图 3.6，红色多边形框内的孤立点即为噪声，需去除。此方式下虽不能直观地判断出海底地形的变化趋势，但通过等深线，或者对等深线进行彩色充填，也可以间接判断出当前编辑区域的地形变化特征，从而能判断哪些是真实数据，哪些是噪声或异常数据。在这种投影方式下，甚至可以对数据进行水深分层，然后在分层的基础上进行细致编辑。通过水平投影在分层的基础上进行编辑，基本上可以剔除所有的数值稍大的噪声数据。

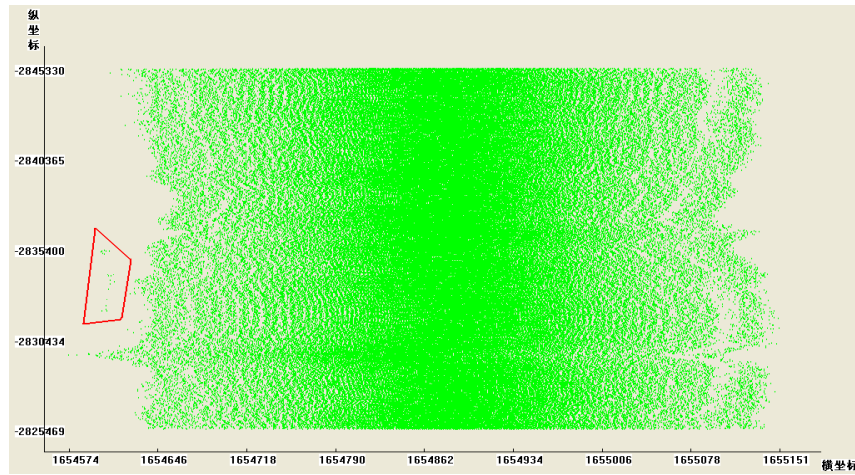


图 3.6 水平投影平面的数据编辑

Fig. 3.6 the raw projection in level direction

3.3.4 多波束异常数据的自动检测

多波束测深异常数据自动检测的方法基本上都是根据测深点的统计特性来进行自动检测的。用统计学方法处理异常数据,就是给定一个置信概率,找出相应的置信区间,只要在此区间外的数据就视为异常数据,并予以剔除,即将被检查点的深度与相邻测点的深度通过比较来做出接受或者拒绝判断。应用比较多的统计方法主要有 *COP* 法、*Ware* 法、*Du* 法、*Binstat* 法、趋势面滤波检验法、中值滤波检验法等。

1、*COP* 方法^{[21][22]}

COP 方法针对 *SeaBeam* 等条带测深数据,根据波束沿航迹方向的时间和垂直航迹方向的距离来选择邻近波束(必须有最小数量的邻近波束存在),如果存在合适的邻近波束,计算这些邻域波束深度的平均值和标准差,通过计算每个邻域波束的深度与平均深度的差值,拒绝大于两倍标准差的邻近波束。重复该过程直到没有邻近波束不符合这个准则,用最终合适的邻域波束深度平均值和标准差评定被检查波束的合理性,并做出接受或拒绝判断。

考虑波束 Z_i 与被接受的邻近波束的平均值 μ_i 比较,并且必须通过下面的准则:

$$|Z - \mu_i| > \beta + \gamma \mu_i \quad (3.4)$$

式中, $\beta=10.0$ ——是个固定值,

$\gamma=1.5\%$ ——对 *SeaBeam* 系统的数据

固定值 β 被认为是由系统的不确定性产生的误差, γ 是水深不确定的一个比值。

2、Ware 方法^{[21][22]}

Ware方法的目的是鉴别条带数据覆盖的区域内,是否存在相对于区域的平均值特别高的深度标准差。在真实的海底表面是光滑连续的假设下,用所有可用的条带测深数据,计算表面的加权移动平均,依据每个水深和加权移动平均水深对于标准差的差值来进行选择判断。

用该方法进行异常数据检验时需要考虑权函数半径的定义和对拒绝的水深使用表面平均和标准差的方法。①权函数的半径将控制有多少测深值用于计算移动平均和标准差。较小的半径测深数将难以获得稳健的参数;相反,大的半径将影响到标准差,很可能改变低于半径波长的海底粗糙度。半径的选择应反映测深的密度和波束足印的大小。②对拒绝的水深使用表面平均和标准差的方法有两种,一是仅使用标准差帮助交互式编辑,这是经常采用的方式。二是根据用户选择的准则对水深分类,选择准则的依据是每个水深和平均水深对于标准差的差值。

3、Du 方法^{[19][20]}

Du方法是使用一个聚类模式自动探测界外值的一个算法,这个算法是仿真操作者做出一个交互式编辑的决策过程。该方法处理水深数据分为两个阶段。

① 分割和第一次标记:对将要检测的所有测深数据进行平面分区(工作窗口),大小近似于 $10\text{ping} \times 10$ 个波束的子集。对每个分区,产生一个直方图用于寻找主群与界外值之间的间隙(如果存在),如图 3.7 所示。

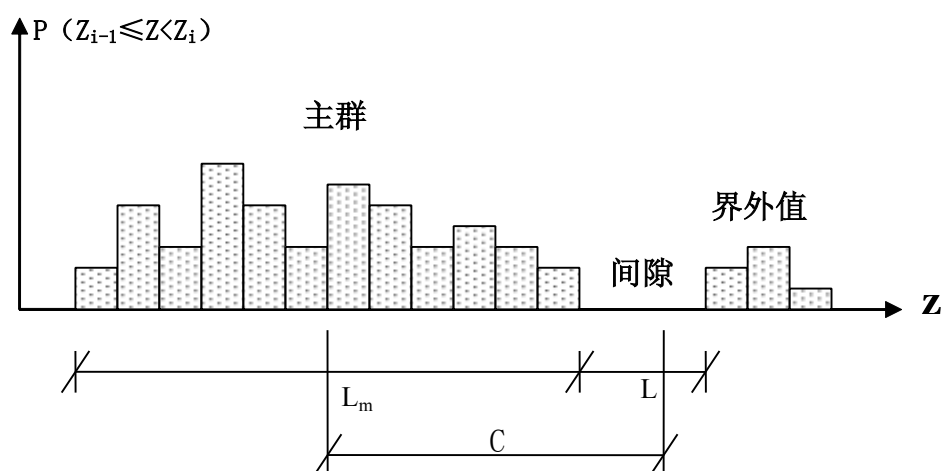


图 3.7 用模式寻求分类

Fig. 3.7 to Seek classification by model

标记的准则是:

$$\begin{aligned} L &\geq L_m / 2 \\ L + L_m / 2 &> C \end{aligned} \quad (3.5)$$

这里 L 是主群与界外值之间的间隙, L_m 是主群的跨距, C 是一个关于测深值标准差的常量, C 通常设为期望标准差的 3 倍。这个准则既可用于界外值比主群浅, 也可用于界外值比主群深。

②第二次标记: 在这个阶段, 每个水深与同测幅中最近的 6 个邻近波束比较, 7 个水深相互排序。在此情况下, L 是考察水深与邻近最浅和最深水深之间的距离, L_m 是最浅和最深波束之间的距离。

现在的标记准则成了:

$$\begin{aligned} L &\geq L_m \\ L + L_m / 2 &> C \end{aligned} \quad (3.6)$$

该方法的优点是: 每次检测不止一个波束, 假定所有的邻近波束都可以接受, 使用分区中所有的波束, 并寻求这些波束中正常水深与奇异水深的自然分离。不清楚的是 10 波束 \times 10 ping 这样的分区大小是否合适。这个方法存在着剔除短波长的异常地形的可能。

4、Binstat 方法^{[21][22]}

Binstat 方法是 *Simrad* 公司多波束后处理软件 *Neptune* 使用的数据清理方法。该方法假设每个数据单元内的地形可以用一个倾斜的平面来表示。这个平面是用最小二乘拟合的方法计算单元内所有水深点到这个平面的位置来获得的 (如图 3.8)。

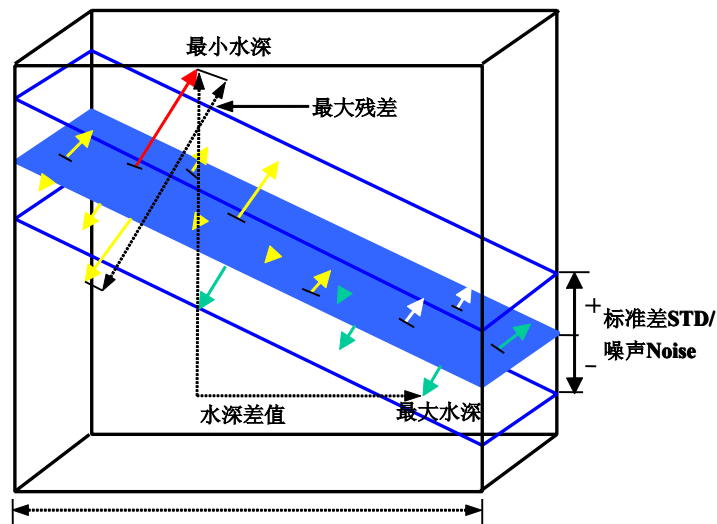


图3.8 Binstat 数据单元的最佳拟合平面, 在这个数据单元中有17个有效点

Fig. 3.8 the best fitted plane of data unit

数据单元的大小是由用户定义的, 定义的原则主要取决于使用的多波束系统每个

Ping 的波束数和波束足印的大小,单元内的数据太少会产生不好的统计结果,当少于 4 个点时不作统计计算。标记或拒绝的准则由用户根据水深、使用的系统等因素来确定,一般选择大于标准差或者噪声的 2~3 倍作为拒绝的准则,通常较大的水深允许较大的标准差。该方法的优点是对数据单元内的所有数据都假定是可接受的并参与统计计算,缺点是数据单元的大小、拒绝的准则要由用户来确定,也存在着剔除短波长异常地形的可能。

5、趋势面滤波法^{[10][17]}

趋势滤波分析法也称之趋势曲线分析、曲线拟合或曲线回归,它是迄今为止研究最多,也最为流行的预测方法。它是根据已知的历史资料来拟合一条曲线,使得这条曲线能反映负荷本身的增长趋势,然后按照这个增长趋势曲线,对要求的未来某一点估计出该时刻的预测值。

趋势面滤波法主要根据波束点邻域拟合一个曲面表示海底,并根据 3σ 原则实现粗差标定。

$$\left. \begin{aligned} (f(x_i, y_i) - \hat{f}(x_i, y_i)) &\leq 3\sigma && \text{正常信号} \\ (f(x_i, y_i) - \hat{f}(x_i, y_i)) &> 3\sigma && \text{异常数据} \end{aligned} \right\} \quad (3.7)$$

式中, $f(x_i, y_i)$ 为点 (x_i, y_i) 处的水深, $\hat{f}(x_i, y_i)$ 为邻域内所有波束拟合的海底水深, σ 为标准差。

曲面拟合方法通常有多项式曲面函数、*Bezier* 方法和 *B* 样条方法等。这里主要介绍应用较广的多项式曲面拟合法。

多项式曲面拟合法,通常利用二次或三次多项式函数表示海底,根据邻域内所有波束点水深和平面位置采用最小二乘法求解系数,在利用确定的多项式拟合的海底判断邻域中央某点波束是否为异常数据。图 3.9 是利用这种方法对一个小测区的测深数据进行了滤波,(a) 部分为滤波前的三维地形图,(b) 部分为滤波前的等深线图,(c) 部分为滤波后的三维地形图,(d) 部分为滤波后的等深线图,图中平面位置(B, L)单位为度,水深为米,各坐标量减去了大数^[17]。

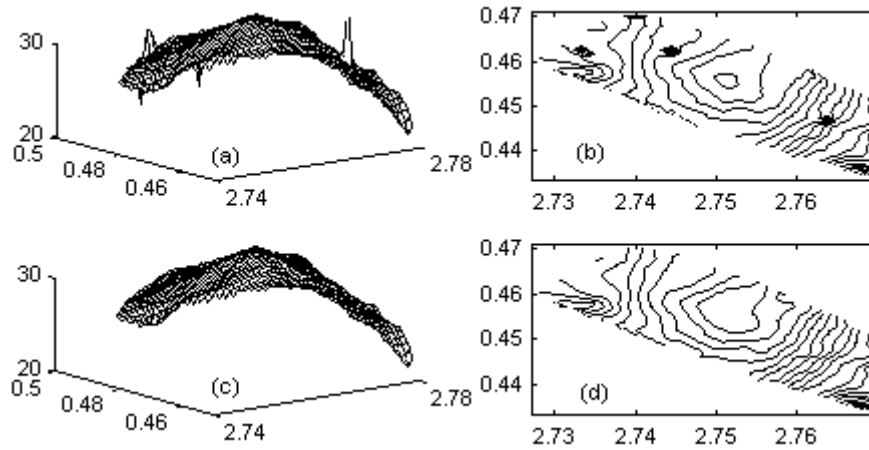


图 3.9 趋势面滤波效果图

Fig. 3.9 the trend surface filtering effect drawing

趋势面滤波操作简单，可方便的实现较大粗差的标定，但不适用于小粗差的标定。滤波时，多项式阶数的选择非常重要。阶数太高，大的粗差不能剔除；阶数太低，可能会将一些有用的信息作为粗差滤掉。对于地势起伏相对平坦的情况非常适用，但若地形变化复杂，将会导致滤波不彻底。

6、中值滤波检验法^{[19][20]}

该方法结合中值滤波和统计偏差检测的优点，仅采用中值滤波后的水深计算每个波束邻域内的平均值，从而计算标准偏差，以此判断该波束是否为异常值。

中值滤波法是一种非线性滤波，它首先确定一个奇数像元的窗口 W ，窗口内各像元按灰度或数值大小排序，并用序列中间值 $g(x, y)$ 作为原数值 $f(x, y)$ 。

$$g(x, y) = \text{median}\{f(x-k, y-l), (k, l) \in W\} \quad (3.8)$$

在一定的条件下，它可以克服线性滤波器所带来的细节模糊，而且对滤除脉冲干扰非常有效。由于异常数据主要成脉冲状，否则如果连续就可能是海底真实地形，比如说飞机、轮船的残骸，也同样可以造成海底地形突变，但是它们的突变是延续的，而不是脉冲状的。中值滤波可以较好的滤去异常数据，但是对高斯噪声和白噪声等，滤波效果较差，另外，对细节也会产生一定的影响。

3.4 数字图像处理中的滤波方法

数字图像在形成、传输、接收和处理的过程中，均会存在着一定程度的噪声干扰。噪声恶化了图像质量，使图像模糊，特征淹没，给分析带来了困难。因此，图像去噪是

图像处理中的一个重要内容，它对图像分割、特征提取、图像识别等更高层次的处理均具有直接的影响。

根据数字图像中包含的噪声的类型、实际图像的特点及频谱分析的规律，数字图像滤波的方法，主要包括两种，即空域滤波和频域滤波。

3.4.1 空域滤波法

空域滤波，其基本思想是用所选的邻域中的各像素灰度的平均值来代替中心像素的灰度值^[23]。但是，空域平滑算法有一个共同的不足，就是它不仅平滑噪声，而且使图像中的细节模糊化。

1、邻域平均法

邻域平均法，即均值滤波，是一种简单的空域处理方法。这种方法的基本思想是用几个像素灰度的平均值来代替每个像素的灰度，实现图像的平滑。最简单的均值滤波方法为非加权邻域平均，它均等的对待邻域中的每个像素。假定有一幅 $N \times N$ 个像素的图像 $f(x, y)$ ，平滑处理后得到一幅图像 $g(x, y)$ 。 $g(x, y)$ 由下式决定：

$$g(x, y) = \frac{1}{M} \sum_{(m, n)} f(m, n) \quad (3.9)$$

式中 $x, y=0, 1, 2, \dots, N-1$ ， (m, n) 点邻域中点的坐标的集合，但其中不包括 (m, n) 点， M 是集合内坐标点的总数。常用的邻域为 4-邻域和 8 邻域。

均值滤波，是一种线性滤波，相当于低通滤波，其算法简单，计算速度快，但在平滑噪声的同时使图像产生模糊，尤其是在图像的细节和边缘处。而且邻域越大，在去噪声能力加强的同时模糊程度越严重。

2、中值滤波

中值滤波是在 1971 年由 *J.w.Jukey* 首先提出并应用于一维信号处理技术(时间序列分析)中的，后被二维图像信号处理技术所引用。所谓中值滤波，是指把以某点 (x, y) 为中心的小窗口内的所有象素的灰度按从大到小的顺序排列，将中间值作为 (x, y) 处的灰度值(若窗口中有偶数个象素，则取两个中间值的平均)。

中值滤波是一种非线性滤波，能抑制图像中的噪声。它是基于图像是由像素数较多、面积较大的小块构成，而噪声往往以孤立的点的形式出现，这些点对应的象素很少的特

点考虑的。该滤波的特点是可以克服图像细节模糊，能够很好的滤除脉冲干扰，但对杂乱的高斯白噪声却无能为力。

3、多图像平均法

如果一幅图像包含有加性噪声，这些噪声对于每个坐标点是不相关的，并且其平均值为零，在这种情况下就可能采用多图像平均法来达到去掉噪声的目的^[23]。

设 $g(x, y)$ 为有噪声图像， $n(x, y)$ 为噪声， $f(x, y)$ 为原始图像，可用下式表示：

$$g(x, y) = f(x, y) + n(x, y) \quad (3.10)$$

多图像平均法是把一系列有噪声的图像 $\{g_j(x, y)\}$ 迭加起来，然后再取平均值以达到平滑。

3.4.2 频域滤波法

由于噪声主要存在于图像的高频部分，所以可以采用频域滤波来去除噪声改善图像质量。在分析图像信号的频率特性时，一幅图像的边缘、跳跃部分以及颗粒噪声代表图像信号的高频分量，而大面积的背景区则代表图像信号的低频分布。用滤波的方法滤除其高频部分就能去掉噪声，使图像得到平滑^[23]。由卷积定理可知，

$$G(u, v) = H(u, v) \cdot F(u, v) \quad (3.11)$$

其中 $F(u, v)$ 是含有噪声的图像的傅里叶变换， $G(u, v)$ 是平滑处理后的图像之傅里叶变换， $H(u, v)$ 是传递函数。选择传递函数 $H(u, v)$ ，利用 $H(u, v)$ 使 $F(u, v)$ 的高频分量得到衰减，得到 $G(u, v)$ 后再经反傅里叶变换就可以得到所希望的平滑图像 $g(x, y)$ 。根据前面的分析，显然 $H(u, v)$ 应该具有低通滤波特性，所以这种方法被称之为低通滤波法平滑化处理^[23]。低通滤波平滑化处理流程如图 3.10 所示。

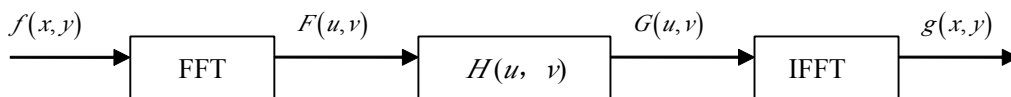


图 3.10 线性滤波器处理框图

Fig. 3.10 the linear filter treatment block diagram

常用的频域低通滤波器(*low pass filter*)有四种，理想低通滤波器、*Butterworth* 低通滤

波器、指数低通滤波器、指数低通滤波器^[23]。

1、理想低通滤波器

一个理想的二维低通滤波器的传递函数由下式表示：

$$H(u, v) = \begin{cases} 1 & D(u, v) \leq D_o \\ 0 & D(u, v) > D_o \end{cases} \quad (3.12)$$

式中 D_o 是一个规定的非负的量，叫做理想低通滤波器的截止频率。 $D(u, v)$ 是从频率域的原点到 (u, v) 点的距离，即 $D(u, v) = \sqrt{u^2 + v^2}$ 。在理论上， D_o 内的频率分量无损通过；而在 $D > D_o$ 的分量却被除掉。然后经逆傅立叶变换得到平滑图像。由于高频成分含有大量的边缘信息，因此采用该滤波器在去噪声的同时将会导致边缘信息损失而是图像边缘模糊，并会产生振铃效应。

2、*Butterworth* 低通滤波器

n 阶 *Butterworth* 低通滤波器的传递函数为：

$$H(u, v) = \frac{1}{1 + \left[\frac{D(u, v)}{D_o} \right]^{2n}} \quad (3.13)$$

Butterworth 低通滤波器传递函数的透视图和其剖面图如图所示。它的特性是连续性衰减，而不像理想滤波器那样陡峭和明显的不连续性。因此采用该滤波器滤波在抑制噪声的同时，图像边缘的模糊程度大大减小，没有振铃效应产生。

3、指数低通滤波器

指数低通滤波器是图像处理中常用的另一种平滑滤波器。它的传递函数为：

$$H(u, v) = e^{-\left[\frac{D(u, v)}{D_o} \right]^n} \quad (3.14)$$

式中， n 决定指数的衰减率。采用该滤波器滤波在抑制噪声的同时，图像边缘的模糊程度较用 *Butterworth* 滤波产生的大些，无明显的振铃效应。

4、梯形低通滤波器

梯形低通滤波器是理想低通滤波器和完全平滑滤波器的折中。它的传递函数为：

$$H(u, v) = \begin{cases} 1 & D(u, v) < D_o \\ \frac{D(u, v) - D_1}{D_o - D_1} & D_o \leq D(u, v) \leq D_1 \\ 0 & D(u, v) > D_1 \end{cases} \quad (3.15)$$

式中 D_o 、 D_1 是规定的。它的性能介于理想低通滤波器和指数滤波器之间，滤波的图像有一定的模糊和振铃效应。

用低通滤波器进行平滑处理可以使噪声伪轮廓等寄生效应减低到不明显的程度，但是由于低通滤波器对噪声滤除的同时，对有用的高频部分也有滤除作用，因此，这种去噪方法是以牺牲清晰度为代价换取的。

3.4.3 数学形态学滤波方法

近些年来，随着数学各分支在理论和应用上的逐步深入，使得数学形态学、模糊数学、遗传算法、小波理论等在图像去噪技术应用中取得很大进展，产生了不少新的去噪算法。这里根据本文所研究的内容主要介绍基于数学形态学的滤波方法。

数学形态学的基本思想^[24]是具有一定形态的结构元素去量度和提取图像中的对应形状以达到对图象分析和识别的目的。数学形态学的基本运算有 4 种：膨胀（或扩张）、腐蚀（或侵蚀）、开运算和闭运算。它们在二值图象中和灰度（多值）图像中各有特点。基于这些基本运算还可推导和组合成各种数学形态学实用算法。

1、膨胀运算

膨胀的算符为 \oplus ，A 用 B 来膨胀写作 $A \oplus B$ 。膨胀运算的作用可以扩张图像的边界。

2、腐蚀运算

腐蚀的算符为 \otimes ，A 用 B 来腐蚀写作 $A \otimes B$ 。腐蚀运算的作用则为收缩边界。

3、开运算

开运算，即先对图像进行腐蚀然后膨胀的结果。开运算的算符为 \circ ，A 用 B 来开启写作 $A \circ B$ ，其定义为：

$$A \circ B = (A \otimes B) \oplus B \quad (3.16)$$

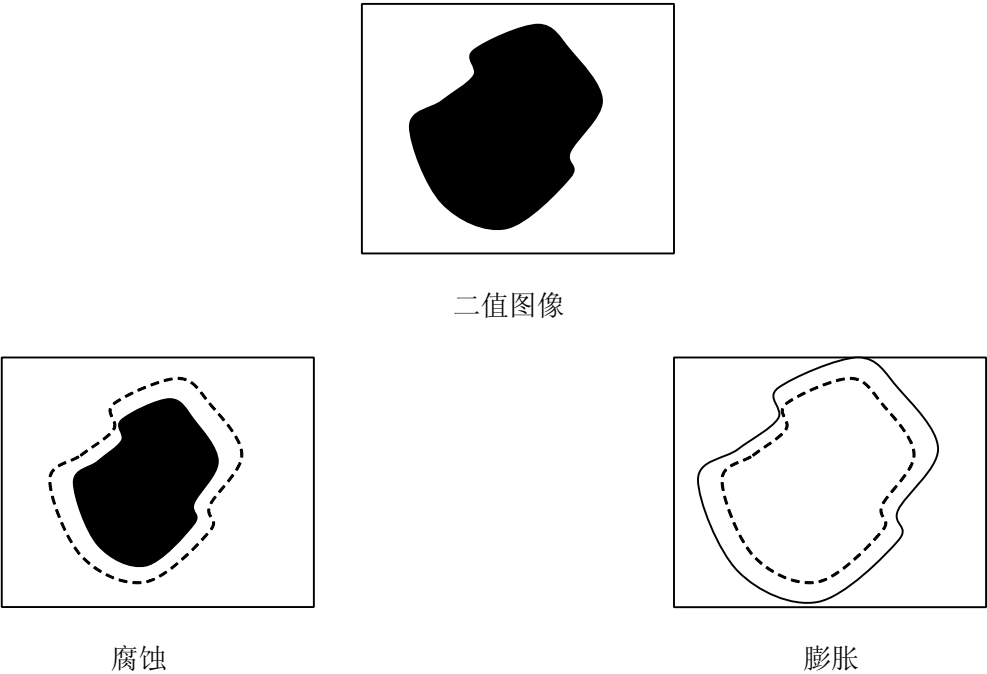
开运算可以平滑边界，滤掉突刺，切断细长搭接而起到分离作用。

4、闭运算

闭运算，即先对图像进行膨胀然后腐蚀的结果。闭运算的算符为 \bullet ，A 用 B 来闭合写作 $A \bullet B$ ，其定义为：

$$A \bullet B = (A \oplus B) \otimes B \tag{3.17}$$

闭运算，在平滑边界的同时，可以连接段的间断，填充小孔。



3.11 腐蚀和膨胀示意图

Fig. 3.11 the erosion and delation schematic diagram

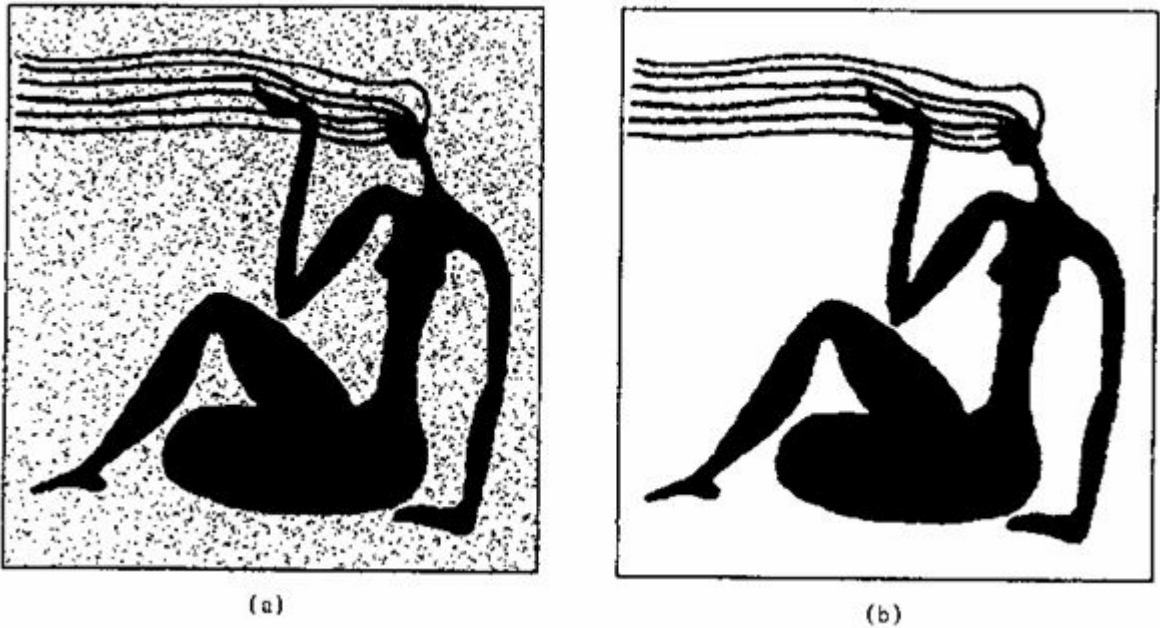


图3.12 (a) 被噪声污染图

(b) 开闭滤波后的图像

Fig. 3.12 (a) the noise pollution image

(b) the filtered images

4 基于密度投影的多波束异常数据自动检测

4.1 基于密度投影的多波束异常数据自动检测

水深数据测量大量异常值的出现，影响了测深数据对真实海底的反映，需要滤除。传统的手工交互式编辑方法速度慢且主观性强，不适用大批量的多波束测深数据的编辑要求。本文根据多波束勘测数据的分布特性，提出的基于密度投影的自动异常数据检测方法，从真实数据范围出发，根据判断准则，可以快速有效的合并数据区域，客观的发现和剔除噪声数据，正确的反应真实水深点。

4.1.1 基于密度投影的多波束异常数据自动检测的基本原理

4.1.1.1 数据投影

首先采用交互式的投影编辑方法，将每条带或条带子区多波束数据投影到一个平面上，可以包括测线前进方向正投影、正交测线方向侧投影等。进行投影后的数据，可以更直观的体现多波束条带数据的整体性，并且在此投影平面上，可以更容易的进行异常数据的判断。下面介绍测线前进方向正投影的方法。

测线前进方向正投影，就是把采集的水深点投影到与测线正交的平面上， x 轴为波束值，而 y 轴为水深值。该投影方法对整 ping 的异常数据比较好识别，适用于海底比较平坦的区域。如图 4.1，即测深数据的测线前进方向正投影，从图中可以看出该测线测得的水深数据，最大值为 4411.6m，最小值为 4129.9m，中央波束精度较高，而边缘波束粗差较多。

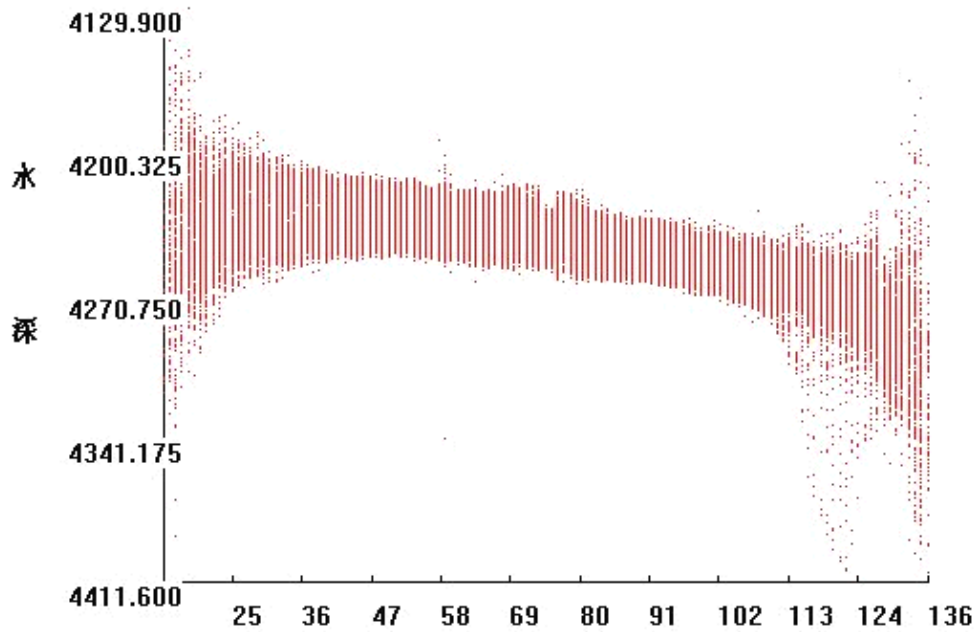


图 4.1 沿测线前进方向正投影图

Fig. 4.1 the raw data projection in orthogonal direction

4.1.1.2 数据分块

1、格网划分

由于多波束数据采集密度大、数量多，直接应用原始的采集数据进行编辑和分析，不仅对计算机硬件要求很高，而且需要占用大量的计算机资源，因此需要在数据处理钱进行网格计算，压缩数据，以提高处理速度。本文采用的内插方法为距离加权内插法，该内插方法考虑了数据点对内插点的不同影响程度，适合多波束测深数据的特点^[4]。

距离加权内插法根据局部窗口内的点与内插点的距离远近来定权，距离内插点越近，对内插点的影响越大。内插公式为：

$$f(X,Y)=\left[\sum_{i=1}^n W(d_i)Z_i\right]/\sum_{i=1}^n W(d_i) \quad (4.1)$$

式中， $W(d_i)$ 是权函数； Z_i 是第 i 个已知点的数据； d_i 是第 i 个已知点到未知点之间的距离。

实践证明, $W(d_i)=1/d_i^2$ 是较优的选择, 即 $W(d_i)$ 的大小由 d_i^2 平方的反比来决定并计算。

2、数据分块

在数据网格化的基础上, 本文提出根据统计落入每个格网中水深点的密度, 来确定大范围的真实水深数据, 即进行数据分区。这里数据分块的概念, 相当于数字图像处理中的区域生长, 即根据相邻小区域的相似性, 把特征相同的小邻域合并到一起, 最后可以形成特征不同的各区域。不同的是, 数字图像处理中分析的是图像的像素值, 而我们这里分析的是网格内水深点的密度值, 以密度最大的格网为种子合并格网。其中网格大小的选择要根据数据的水深范围, 合理的进行选择。

基于网格点密度的数据分块首先要解决三个问题: ① 选择种子点的依据; ② 确定相似性准则, 即分块方式; ③ 终止分块的条件。具体来说, 本算法将密度最大的网格点作为种子点, 即初始的搜索位置。首先依据已设定的密度阈值, 去掉低于该密度的水深数据区。然后, 根据统计学的原理, 密度较大的数据区域应该为真实的数据区, 因此, 搜索投影图像中与种子点相邻的具有水深点的网格, 依据分区准则, 可以获得一个大的数据块和(或)几个不相连的子数据块。通常, 获得的大数据块为真实的水深数据区域, 与之不相连的小的子块为异常数据, 即可滤除。

下面给出基于网格点密度的数据分块的步骤。

(1)对投影后的水深数据划分网格, 统计出每个格网点的密度, 定位密度最大的网格点, 并设定阈值, 去掉低于密度阈值的测深数据。

(2)从密度最大的网格开始, 将其周围(4—邻域或 8—邻域)不属于其他区域的网格密度同该网格点的密度进行比较, 若密度差值小于阈值, 则合并到同一区域。

(3)从新合并的网格点开始, 反复进行(2)的操作。

(4)反复进行(2)、(3)的操作, 直至不能再合并。

3、算法实例

数据分块, 即根据格网密度进行区域增长, 可以快速定位主块范围, 滤除远离主块的异常数据。

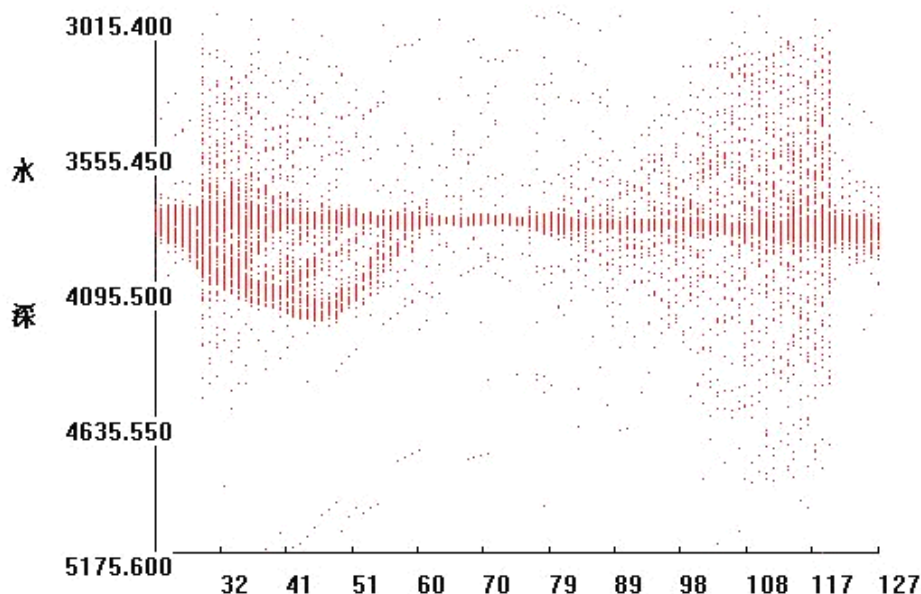


图 4.2 区域增长前水深数据投影

Fig. 4.2 the projection of water depth before region growing

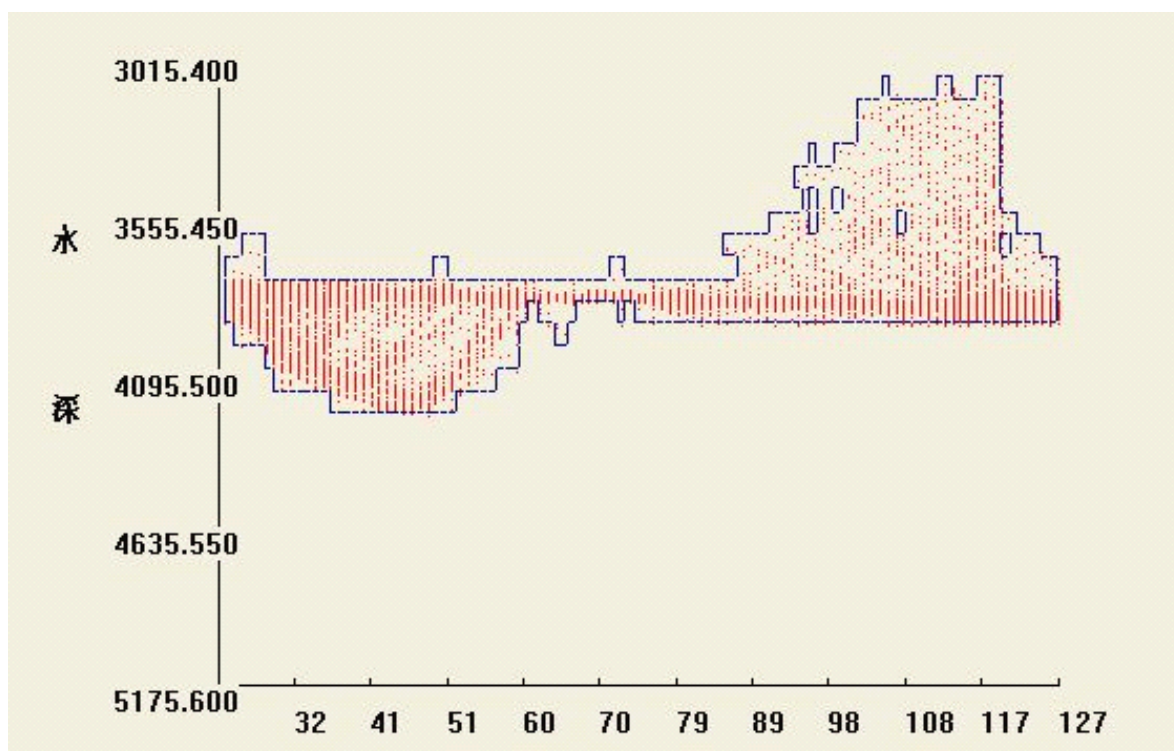


图 4.3 区域增长后水深数据投影

Fig. 4.3 the projection of water depth after region growing

由图 4.2 和图 4.3 对比，区域增长可滤除远离主块的异常数据，但对与主块相连的细小子块中的异常数据滤除效果不好。所以，对于水深异常数据的定位过滤，不能单独运用区域增长的方法，还需结合其他的方法对水深数据进行处理，达到过滤目的。

4.1.1.3 数学形态学滤波

1、数学形态学滤波

通过数据分区可以删除与主块分离的子块，可以迅速定位异常较大的水深数据和大片的异常水深数据。但，对于投影后与主块相连的异常数据块，用此种方法就无能为力了，这时可以借助数学形态学的方法对数据进行处理，以剔除这些相连的异常水深数据。

数学形态学^[25]主要通过腐蚀、膨胀、开、闭等四种基本运算对数据进行滤波处理的。其运算主要用于二值图像的处理。腐蚀运算，可以消除边界点，使边界向内部收缩，可以消除小且无意义的物体。膨胀运算，可以将与物体接触的所有背景点合并到该物体中，使边界向外部扩张的过程，可用来填补物体内的空洞。开运算，则是可以平滑边界，切断细长搭接，消除突刺，实际是就是对图像进行先腐蚀后膨胀的结果。而闭运算，则相反，是对图像进行先膨胀后腐蚀的结果，可以在平滑边界的同时，连接段的间断，填充小孔。实际处理图像的过程中，都是根据实际的图像特点，将该四种运算相互结合处理，已达到图像处理目的。

根据不同的数据，可将投影后的数据主块进行相应的腐蚀、膨胀运算，以滤除与主块相连的异常水深值，确定新的主块。

2、算法实例

数字图像处理方法中，可以运用腐蚀膨胀算法滤除二值图中的噪声点。利用这一原理，将原始的多波束数据二值化后可利用腐蚀膨胀算法进行异常值剔除。原始的多波束数据二值图、投影图分别见图 4.4、图 4.5。

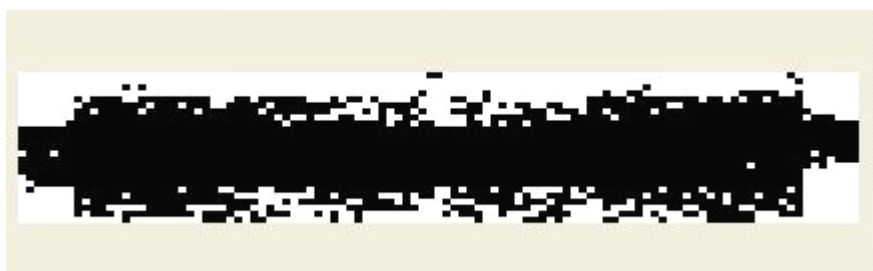


图 4.4 原始的多波束数据二值图

Fig. 4.4 the binary image of raw multibeam data

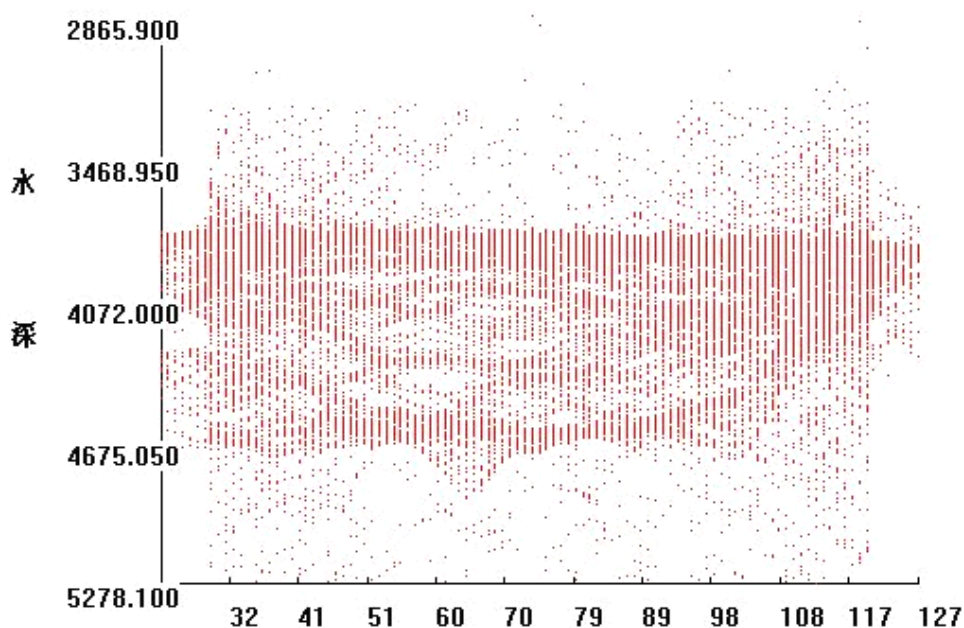


图 4.5 原始的多波束数据投影图

Fig. 4.5 the projection of raw multibeam water depth data

图像进行多次腐蚀及膨胀运算，可消除主块外的散点。但不同的运算次数，所获得的结果不同，如下图所示。

(1) 腐蚀膨胀 3 次后示意图：

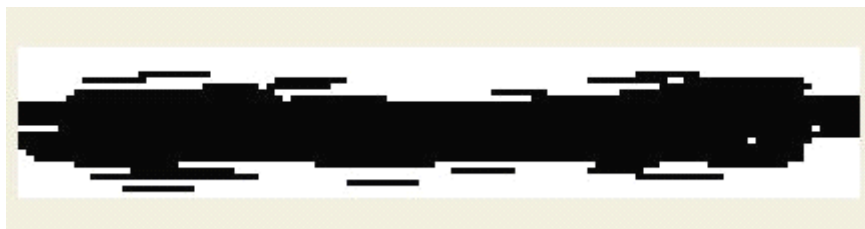


图 4.6 腐蚀膨胀 3 次后的二值图

Fig. 4.6 the binary image after 3 times erosion and dilation

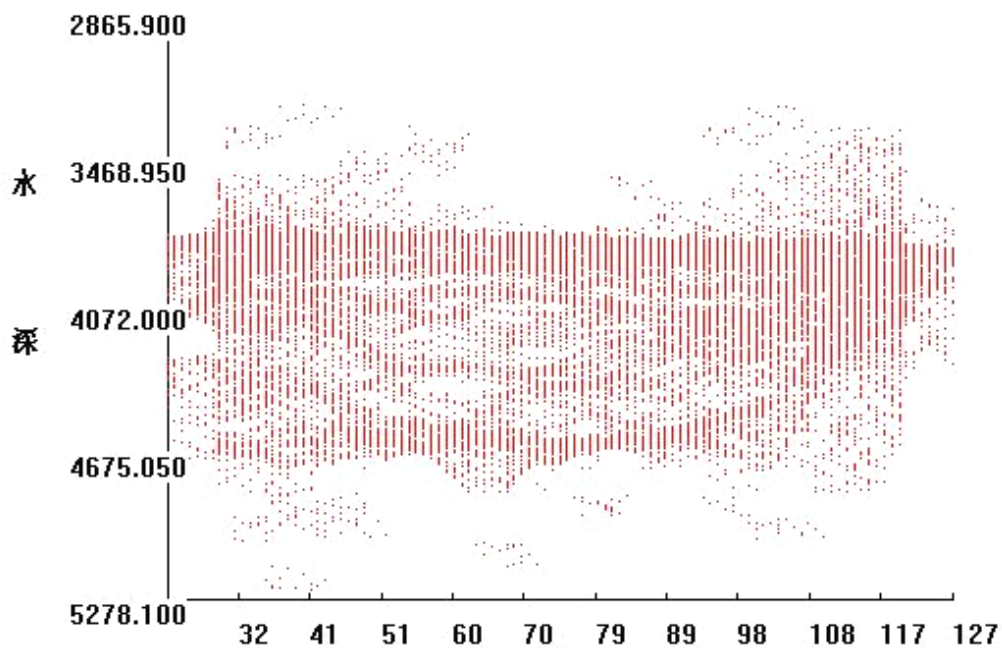


图 4.7 腐蚀膨胀 3 次后的散点图

Fig. 4.7 the projection of multibeam data after 3 times erosion and dilation

(2) 腐蚀膨胀 12 次后示意图:

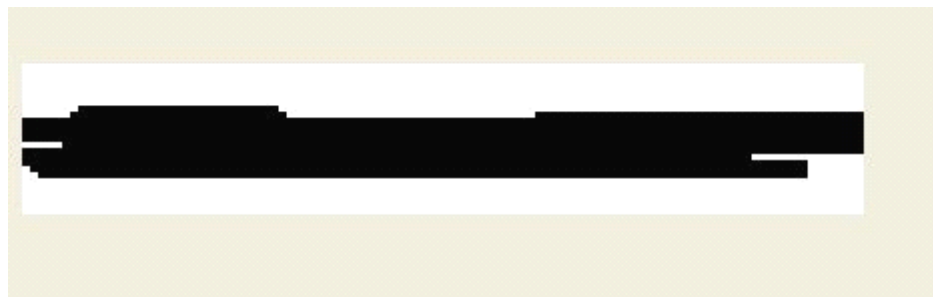


图 4.8 腐蚀膨胀 12 次后的二值图

Fig. 4.8 the binary image after 12 times erosion and dilation

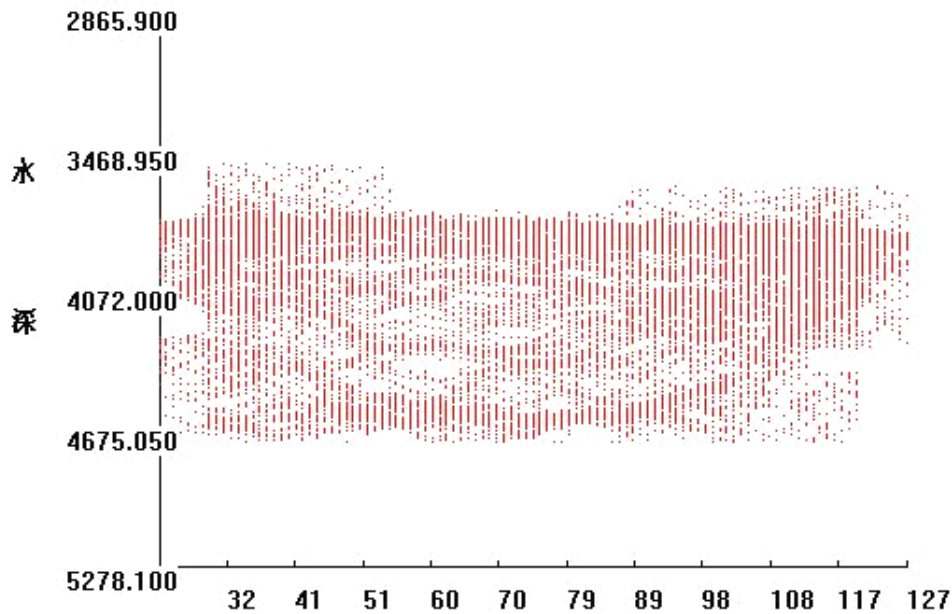


图 4.9 腐蚀膨胀 12 次后的散点图

Fig. 4.9 the projection of multibeam data after 12 times erosion and dilation

由上图比较可知，腐蚀膨胀算法可以去除离散于正常数据外或与正常数据相连异常水深值，并且运算次数的不同，去除效果不同。但并不是运算次数越多越好，而且根据网格大小的不同，滤除异常数据的效果也是有一定限度的。

同时，虽然腐蚀膨胀算法可以有效的滤除异常水深数据，但是对于数据分块较多的水深数据，则该方法不适用，所以不能直接利用该方法进行异常值的检测，而应该结合其它处理方法，去除噪声数据。

4.1.1.4 中值滤波

1、中值滤波

经数据分块及腐蚀膨胀算法后，已经可以定位出大部分的异常水深数据，包括异常较大的水深数据、投影后和主块不相连的子块异常数据及与主块相连的异常数据块，但对于地形变化复杂，异常值在条带水深变化范围内的，仍未被探测出，为此，可以借助中值滤波算法。

前面已经介绍过，在一定条件下，中值滤波可以克服线性滤波器带来的细节模糊，而且可以有效的滤除脉冲噪声，但是对大面积的噪声污染，如高斯噪声和白噪声等，滤波效果较差。

而真实海底的地形都是连续变化的，海洋水深测量中的异常数据通常都成脉冲状，

即使是飞机、轮船等的残骸，其突变也是延续变化的，而不是脉冲形状。因此，可以利用中值滤波对测深数据进行处理，剔除脉冲状异常测深值。脉冲状异常值如图 4.10 所示。

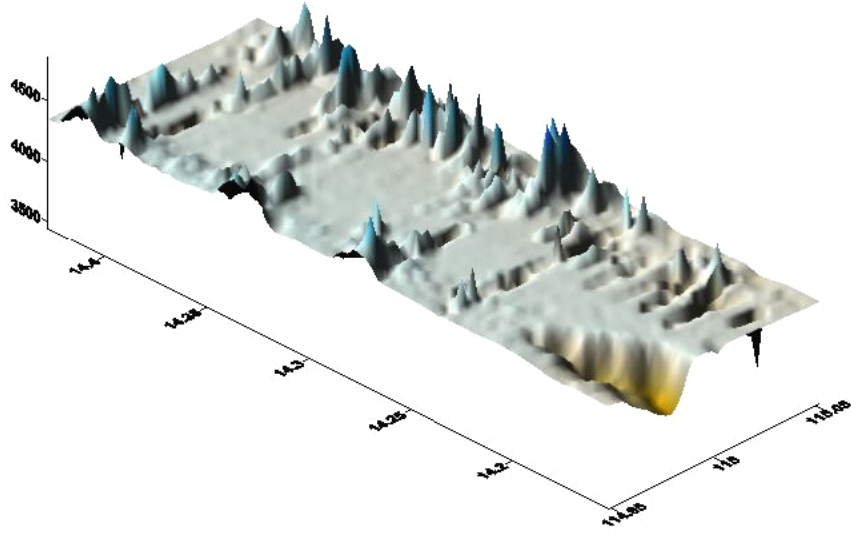


图 4.10 脉冲状水深异常值

Fig. 4.10 the pulse outlier of water depth

为了避免上述问题，中值滤波不作为最后的结果，而仅通过它获得的局部方差，并利用局部方差别除异常点^[20]。

局部方差的计算为：

$$\hat{\sigma}_i^2 = \frac{1}{j-1} \sum_{i=1}^j [g_i(x, y) - u_i]^2 \quad (4.2)$$

其中， u_i 是中值滤波后局部窗口内的点的深度的平均值， j 是点数， $g_i(x, y)$ 为测深值。若 $g_i(x, y)$ 与 u_i 的差值超过 $3\hat{\sigma}_i$ ，则确定是异常值，并利用局域内插值作为该点的深度。

2、算法实例

对比图 4.11 与图 4.12，中值滤波可以有效去除脉冲噪声，但对大面积的高斯白噪声无能为力。因此，过滤多波束异常数据，不能仅仅运用中值滤波这个方法，而需结合其他方法，这样才能高效的判断异常数据，进行数据清理。

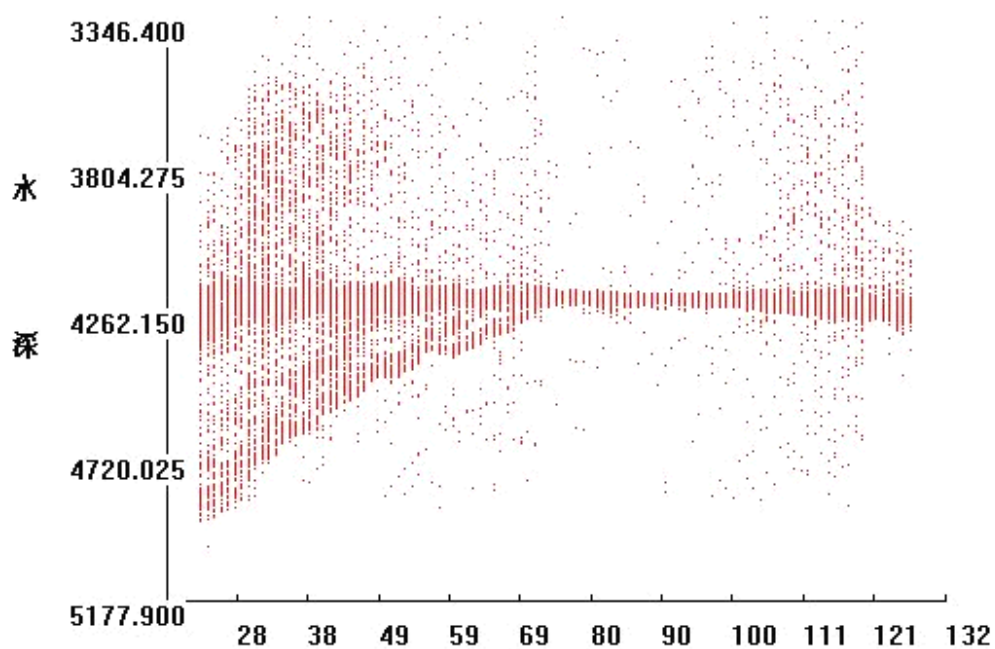


图 4.11 滤波前示意图

Fig. 4.11 the projection of multibeam data before filtering

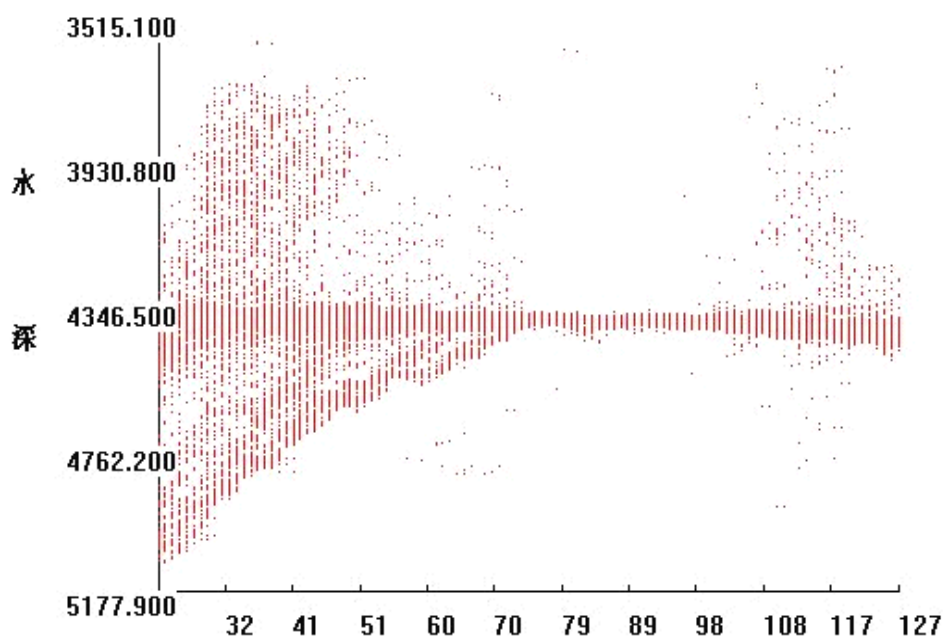


图 4.12 滤波后示意图

Fig. 4.12 the projection of multibeam data after filtering

4.1.1.5 总结

从上面各算法的分析中可以看到, 单独运用这些算法, 异常数据的处理都不是很理想, 所以将上述几种处理方式集合起来, 可达到理想的处理效果。但同时, 处理结果的好坏还与格网的划分有关, 理论上网格越小, 处理的效果越好, 但代价是处理的速度变慢, 因此并不是网格的大小越小越好。所以, 实际处理时网格参数的设置很关键, 影响着数据处理结果的好坏。

4.1.2 算法实现及部分代码

本算法基于 *VC++6.0* 编译。*Visual C++ 6.0* 是一个可视化的软件开发工具, 其功能强大、界面友好、代码效率高、执行速度快, 能方便地形成应用软件中所需的各种用户界面 (*GUI*) 和工具, 并可直接与系统及底层硬件交换数据。其中集成的微软基础类库 *MFC* (*Microsoft foundation class*), 功能齐全, 用户调用已有的类库函数, 就可实现自己的需要。具体的数据处理过程如下:

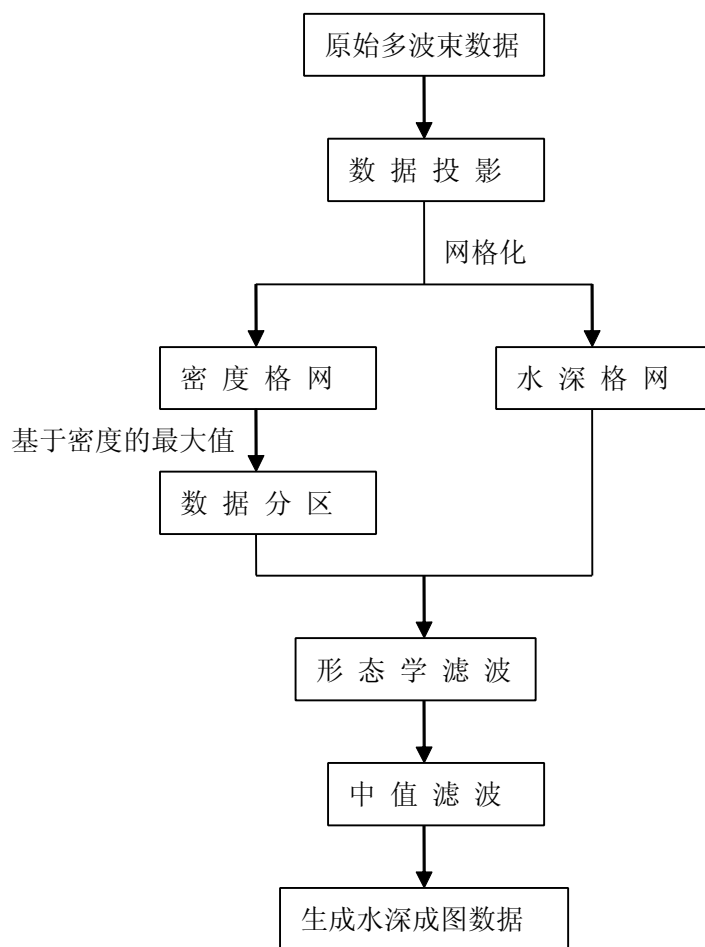


图 4.13 数据处理流程图

Fig. 4.13 the flow chart of data processing

1、数据提取

首先构造一个结构体以存储需要进行处理的数据，结构体中定义的元素就是原始多波束测深数据中所包含的所有数据的类型。定义的结构体如下所示：

```

struct MinePoint {
    int ping ;           //ping 号
    int beam;           //波束号
    double x;           //x 坐标，单位 m
    double y;           //y 坐标，单位 m
    double deep;         //深度
    double longitude;    //经度
    double latitude;     //纬度
    double pitch;        //纵摇
    
```

```
double roll;           //横摇
double heave;          //起伏
double draft;          //吃水
double velocity;       //速度
double heading;        //航向
double acrosstrack;    //侧向距
double alongtrack;     //航向距
double trvaltime;      //旅行时
double angle;          //波束角

bool flag;             //数据标记，即水深数据是否为异常数据的标记。
int index;             //所在网格索引，即水深点所在的格网的索引值。

};
```

其中, x , y , $deep$ 分别为测深点的高斯平面坐标和深度值, 即在水深异常值检测中主要处理的数据。

读取原始多波束数据的相关代码, 如下所示:

```
bool CWDeepDoc::Mine_explode(double *out, CString in, int max, char div = ',')
{
    int i=0;
    int begin=0,end=0;
    while(true)
    {
        if(i>=max) break;
        end = in.Find(div,begin);
        if(end<0)
        {
            end = in.GetLength();
            out[i] = atof(in.Mid(begin,end - begin));
            break;
        }
    }
}
```

```

        out[i++] = atof(in.Mid(begin, end - begin));
        begin = end + 1;
    }
    return true;
}

```

其中导入数据时，需将水深点的大地坐标换算成高斯直角坐标，以便进行成图处理。

2、网格生成

建立数据格网的目的是为了压缩数据，以提高处理速度。其中水深数据网格建立的方法是距离加权内插法^[4]。

生成网格的格式为：第一行为格网的行数和列数，形式为“***, *col*, *row*”；第二行为格网纵向的宽度，“***, *m_iGrid*”；第三行开始则为计算的网格值。

```

bool CWDeepDoc::Mine_BuildGrid()
{
    ofstream fout;
    int beam = m_iMinBeam;
    int deep = m_iGrid * int(m_fMinDeep/m_iGrid);
    int col = ceil((m_fMaxDeep - deep)/m_iGrid);
    int row = m_iMaxBeam - m_iMinBeam + 1;
    int all = col*row;
    double *tmp = new double[all];
    double *tmp1 = new double[all];
    //建立密度格网
    for(int i=0;i<m_iMaxLine;i++)
    {
        int m = m_pMyPoint[i].beam - beam;
        int n = (m_pMyPoint[i].deep - deep)/m_iGrid;
        int index = n*row + m;
        if(tmp[index]<0)
            tmp[index] = 1;
        else

```

```
        tmp[index] += 1;
    }
    //写入密度格网数据文件
    fout.open(m_sFile_dgrid);
    fout<<"*,"<<col<<','<<row<<'\n';
    fout<<"*,"<<m_iGrid<<'\n';
    for(i=0;i<all;i++)
    {
        if((i+1)%row==0)
        {
            if(tmp[i]<0)
                fout<<"0\n";
            else
                fout<<tmp[i]<<'\n';
        }
        else
        {
            if(tmp[i]<0)
                fout<<"0,";
            else
                fout<<tmp[i]<<',';
        }
    }
    fout.close();
    //建立深度格网
    memset(tmp,0,all * sizeof(double));
    for(i=0;i<m_iMaxLine;i++)
    {
        int m = m_pMyPoint[i].beam - beam;    //行值
        int n = (m_pMyPoint[i].deep - deep)/m_iGrid;
```

```
int index = n*row + m;
if(index>=all)
{
    MessageBox(NULL,"计算错误","警告窗口",MB_OK|MB_ICONSTOP);
    delete[] tmp1;
    return false;
}
double middle;
middle = m_fMinDeep + m_iGrid * (n + 0.5);
double p;
p = 1.0 / pow((m_pMyPoint[i].deep - middle),2);    //计算权值
if(tmp1[index]<0)
    tmp1[index] = m_pMyPoint[i].deep * p;
else
    tmp1[index] += m_pMyPoint[i].deep * p;
tmp[index] += p;
m_pMyPoint[i].index = index;
}
for(i=0;i<all;i++)
{
    if(tmp[i]>0)
        tmp1[i] = tmp1[i]/tmp[i];
}
//写入深度格网文件
fout.open(m_sFile_wgrid);
fout<<"*,"<<col<<','<<row<<'\n';
fout<<"*,"<<m_iGrid<<'\n';
for(i =0;i<all;i++)
{
    if((i+1)%row==0)
```



```

{
    if(tmp1[i]<0)
        fout<<"0\n";
    else
        fout<<tmp1[i]<<"\n";
}
else
{
    if(tmp1[i]<0)
        fout<<"0,";
    else
        fout<<tmp1[i]<<',';
}
}
fout.close();

delete[] tmp;
delete[] tmp1;

return true;
}

```

3、数据分块

数据分块，是为了剔除远离主块的分散异常数据，达到去除异常数据的目的。其代码如下：

```

bool CMyDlgGrow::Mine_RegionGrow(int nSeedX,int nSeedY, double *pUnchInput,
                                   int nWidth, int nHeight, double *pUnRegion,
                                   CRect &R, int nThreshold)
{
    int nDx[] = {-1,1,0,0};
    int nDy[] = {0,0,-1,1};

```

```

int nSaveWidth = nWidth;

int * pnGrowQueX ;

int * pnGrowQueY ;

pnGrowQueX = new int [nWidth*nHeight];
pnGrowQueY = new int [nWidth*nHeight];

int nStart ;

int nEnd ;

nStart = 0 ;

nEnd = 0 ;

pnGrowQueX[nEnd] = nSeedX;
pnGrowQueY[nEnd] = nSeedY;

int nCurrX ;

int nCurrY ;

int k ;

int xx;

int yy;

while (nStart<=nEnd)
{
    nCurrX = pnGrowQueX[nStart];
    nCurrY = pnGrowQueY[nStart];
    for (k=0; k<4; k++)
    {
        xx = nCurrX+nDx[k];
        yy = nCurrY+nDy[k];
        if((xx<nWidth)&&(xx>0)&&(yy>0)&&(yy<nHeight)&&(pUnRegion[yy*nWi
            dth+xx]<=0)&&(pUnchInput[yy*nSaveWidth+xx]
            >0)&&abs(pUnchInput[yy*nSaveWidth+xx]-
            pUnchInput[nCurrY*nSaveWidth+nCurrX])<=nT
            hreshold)
        {

```

```

        nEnd++;
        pnGrowQueX[nEnd] = xx;
        pnGrowQueY[nEnd] = yy;
        pUnRegion[yy*nWidth+xx] = 255 ;
    }
}
nStart++;
}
delete []pnGrowQueX;
delete []pnGrowQueY;
pnGrowQueX = NULL ;
pnGrowQueY = NULL ;
return FALSE;
}

```

4、腐蚀与膨胀运算

通过腐蚀与膨胀算法，可以断开与主块连接的细小的异常水深值，以剔除异常水深值。以下是腐蚀算法的代码：

```

void CWDeepDoc::Mine_Erosion(int * data, int nWidth, int nHeight)
{
    int all = nWidth * nHeight;
    int* newdata = new int[all];
    memcpy(newdata,data,all*sizeof(int));
    //水平腐蚀
    for(int i = 0; i < nHeight; i++)
    {
        for(int j = 1; j < nWidth - 1; j++)
        {
            int index = i*nWidth + j;    //计算出该点的位置
            if(0 == newdata[index])
            {

```

```
data[index] = 0;    //先设置元素为 0
for(int m = 0; m < 3; m++)
{
    if(255 == newdata[index+m-1])
    {
        //自身及左右邻居中若有一个不是黑点，则将该点腐蚀成白点
        data[index] = 255;
        break;
    }
}
}
}

//垂直腐蚀
/* for(i = 1; i < nHeight -1; i++)
{
    for(int j = 0; j < nWidth; j++)
    {
        int index = i*nWidth + j;    //计算出该点的位置
        if(0 == newdata[index])
        {
            data[index] = 0;
            for(int m = 0; m < 3; m++)
            {
                if(255 == newdata[index+(m-1)*nWidth])
                {
                    //自身及上下邻居中若有一个不是黑点，则将该点腐蚀成白点
                    data[index] = 255;
                    break;
                }
            }
        }
    }
}
```

```

    }
}
*/
delete[] newdata;
}

```

以下是腐蚀算法的代码：

```

void CWDeepDoc::Mine_Dilation(int *data, int nWidth, int nHeight)
{
    int all = nWidth * nHeight;
    int* newdata = new int[all];
    memcpy(newdata,data,all*sizeof(int));
    //水平膨胀
    for(int i = 0; i < nHeight; i++)
    {
        for(int j = 1; j < nWidth - 1; j++)
        {
            int index = i*nWidth + j;    //计算出该点的位置
            if(255 == newdata[index])
            {
                data[index] = 255;    //先设置元素为 0
                for(int m = 0; m < 3; m++)
                {
                    if(0 == newdata[index+m-1])
                    {
                        //自身及左右邻居中若有一个不是黑点，则将该点腐蚀成白点
                        data[index] = 0;
                        break;
                    }
                }
            }
        }
    }
}

```

```

    }
    //垂直膨胀
    /* for(i = 1; i < nHeight - 1; i++)
    {
        for(int j = 0; j < nWidth; j++)
        {
            if(255 == data[j])
            {
                int index = i * nWidth + j;    //计算出该点的位置
                for(int m = 0; m < 3; m++)
                {
                    if(0 == data[index+(m-1)*nWidth])
                    {
                        //自身及上下邻居中若有一个不是黑点，则将该点膨胀成白点
                        data[index] = 0;
                        break;
                    }
                }
            }
        }
    }
    */
}

```

5、中值滤波算法

中值滤波的目的，是利用它来计算数据的局部方差，即通过计算统计阈值来过滤异常数据，其代码如下：

```

void CWDeepDoc::Mine_Median(double *data, int nWidth, int nHeight)
{
    int all = nWidth * nHeight;
    double* newdata = new double[all];
    memcpy(newdata, data, all * sizeof(double));
    double *temp;

```

```
for(int y=0;y<nHeight;y++)
{
    for(int x=2;x<nWidth-2;x++)
    {
        double a[5];
        int t = y*nWidth+x;
        a[0]=newdata[t-2];
        a[1]=newdata[t-1];
        a[2]=newdata[t];
        a[3]=newdata[t+1];
        a[4]=newdata[t+2];
        for(int i=0;i<5;i++)
        {
            for(int j=4;j>i;j--)
            {
                if(a[j]<a[j-1])
                {
                    double t;
                    t=a[j];
                    a[j]=a[j-1];
                    a[j-1]=t;
                }
            }
        }
        data[t] = a[2];
    }
}
delete[] newdata;
}
```

根据中值滤波计算局部方差检测异常数据的具体算法如下：


```
void CDialogMedian::OnOK()
{
    if(m_file_save.IsEmpty() || m_file_mgrid.IsEmpty())
        return;
    //读数据部分
    CStdioFile f(m_file_mgrid,CFile::modeRead);
    CString str;
    f.ReadString(str);
    double a[3];
    m_pDoc->Mine_explode(a,str,3,',');
    m_col = a[1];
    m_row = a[2];
    f.ReadString(str);
    m_pDoc->Mine_explode(a,str,2,',');
    m_grid = a[1];
    int all = m_col*m_row;
    m_data_mgrid = new double[all];    //存储水深数据
    int iCount = 0;
    while(f.ReadString(str)) {
        double *a = new double[m_row];
        m_pDoc->Mine_explode(a,str,m_row,',');
        memcpy(m_data_mgrid+iCount*m_row,a,m_row*sizeof(double));
        m_MyDlgProgress->SetPoint(iCount);
        iCount++;
        for(int m=0;m<m_row;m++)
            if(m_maxValue<(int)a[m])
                m_maxValue = (int)a[m];
        delete[] a;
    }
    f.Close();
}
```

//数据读取完毕

m_pDoc->Mine_Median(m_data_mgrid,m_row,m_col); //中值滤波计算

double* m_data_median = new double[m_col*m_row]; //存储计算的方差

//计算局部方差

for(int i=0;i<all;i++) {

double sum = 0.0; //求和

int times = 0; //计算个数

for(int j=0; j < m_pDoc->m_iMaxLine; j++)

if(m_pDoc->m_pMyPoint[j].index == i)

{

sum += pow((m_pDoc->m_pMyPoint[j].deep - m_data_mgrid[i]) , 2);

times++;

}

if(times!=0)

m_data_median[i] = sum / (double)times;

else

m_data_median[i] = 0;

m_data_median[i] = sqrt(m_data_median[i]);

}

for(int j=0; j < m_pDoc->m_iMaxLine; j++) {

if(m_data_mgrid[m_pDoc->m_pMyPoint[j].index] == 0)

{

m_pDoc->m_pMyPoint[j].flag = false; //异常数据标记为 false

continue;

}

if((m_pDoc->m_pMyPoint[j].deep - m_data_mgrid[m_pDoc->

m_pMyPoint[j].index])>=

3 * m_data_median[m_pDoc->m_pMyPoint[j].index])

m_pDoc->m_pMyPoint[j].flag = false; //异常数据标记为 false

else

```
m_pDoc->m_pMyPoint[j].flag = true;    //正常数据标记为 true
}
m_pDoc->OnFileSave(m_file_save);    //保存数据

delete[] m_data_median;
delete[] m_data_mgrid;
}
```

4.2 实际数据分析

实验数据，测区最大水深 5177.2m，最小水深 3629.6m，最小波束号 23，最大为 127，共有 329 个条带，31950 个波束数据。

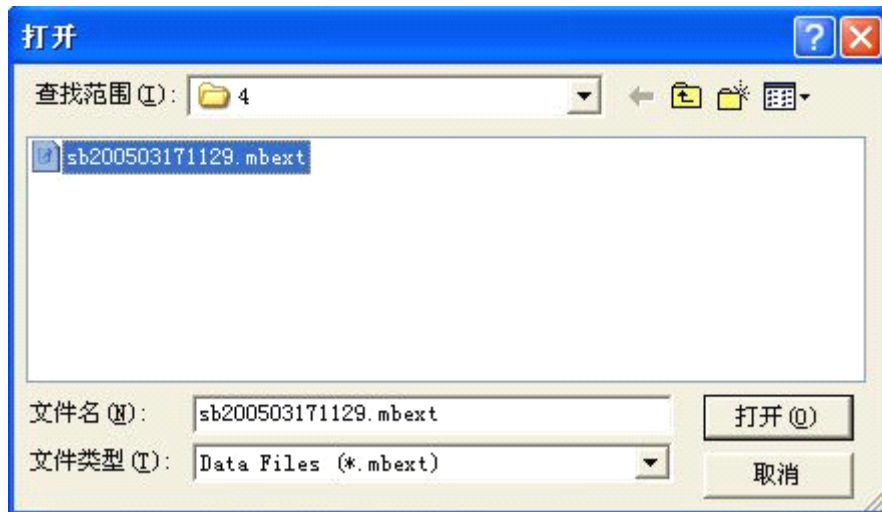


图 4.13 读取数据对话框

Fig. 4.13 the opening dialog

其条带水深沿测线前进方向正投影见图 4.14 所示。由图可见，点云密集的地方，即横向的条带处为正常水深数据，而该条带上下存在的大量散点为异常水深值，即需要滤除的数据。

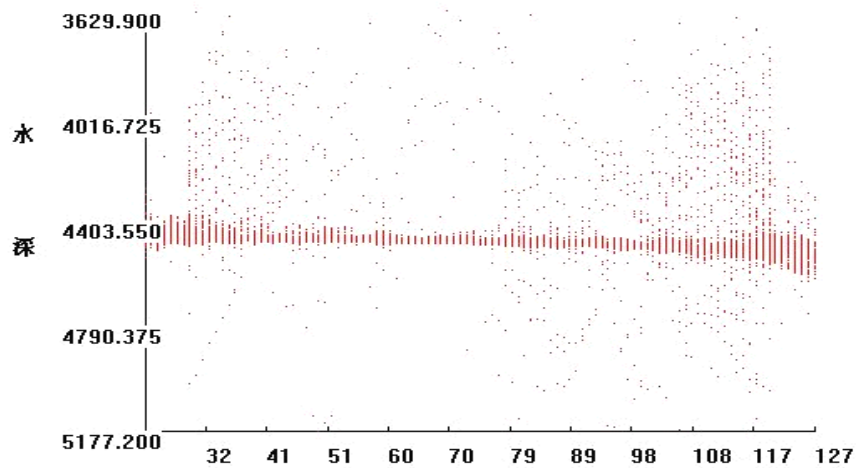


图 4.14 沿测线前进方向正投影图

Fig. 4.14 the raw data projection in orthogonal direction

根据距离加权内插法将数据网格化，网格宽度为 50 米。

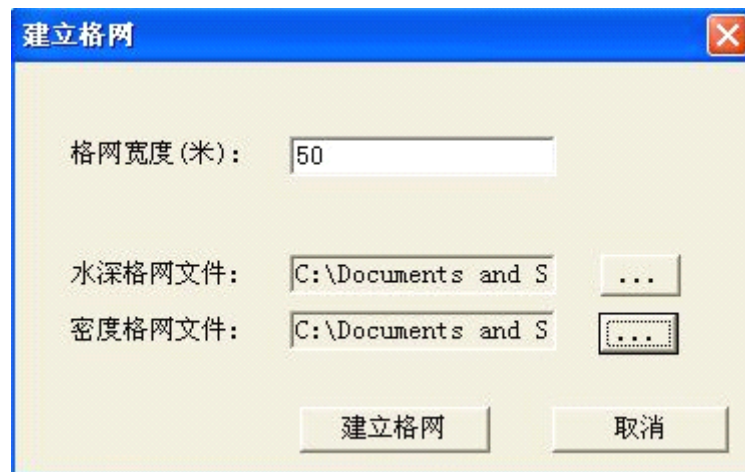


图 4.15 建立格网对话框

Fig. 4.15 the grid-built dialog

统计每个格网内数据点的密度，其对应的密度图，如图 4.16 所示。其中，红色的区域表示水深点密集的地方，即正常水深值；浅蓝色代表网格内水深值稀少的区域，即可认为是异常水深值，需要滤除。

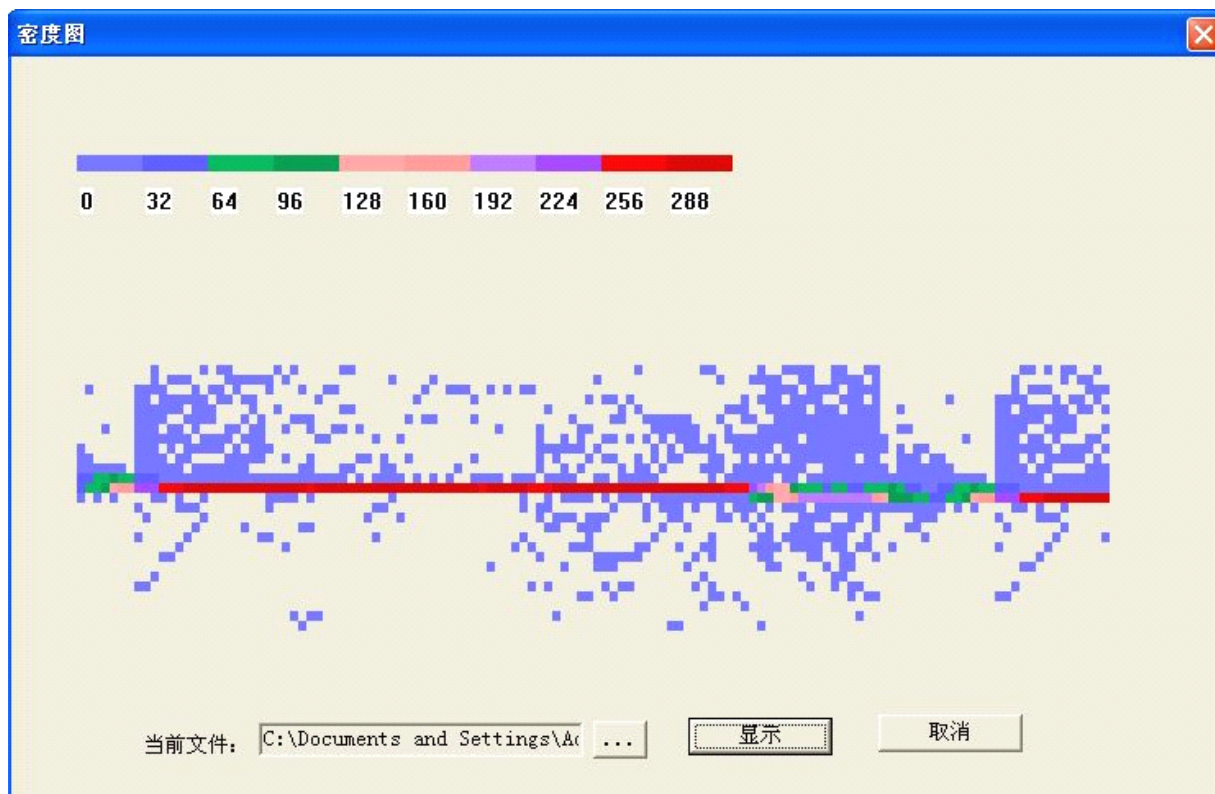


图 4.16 对应的密度图

Fig. 4.16 the corresponding density map

根据判断准则, 首先设定阈值为 40, 然后基于密度最大的格网, 向外搜索相邻的网格, 最终得到一个大的连同区域, 如图 4.17 所示, 离散于横向条带的异常水深数据都被剔除了。



图 4.17 数据分块后的密度图

Fig. 4.17 the data density map after block

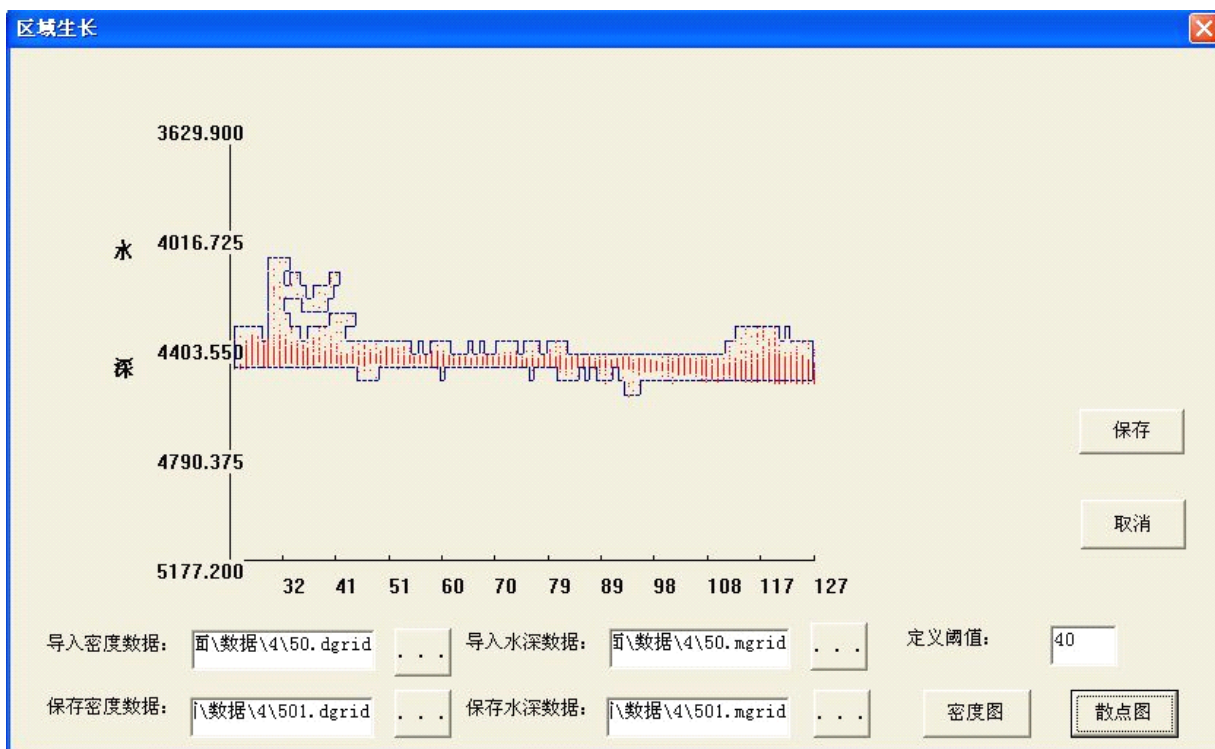


图 4.18 数据沿测线前进方向正投影图

Fig. 4.18 the scatter image after erosion and dilation

如图 4.18，该区域仍包含一些与主块连通的小的异常数据块，接着使用形态学腐蚀及膨胀算子进行操作，最终剔除这些小的与主块相连的数据块。

数据分块后的数据二值图，见图 4.19，腐蚀膨胀 3 次后的数据二值图，见图 4.20，散点图，见 4.21。



图 4.19 原始二值图

Fig. 4.19 the raw binary image



图 4.20 腐蚀膨胀后二值图

Fig. 4.20 the binary image after erosion and dilation

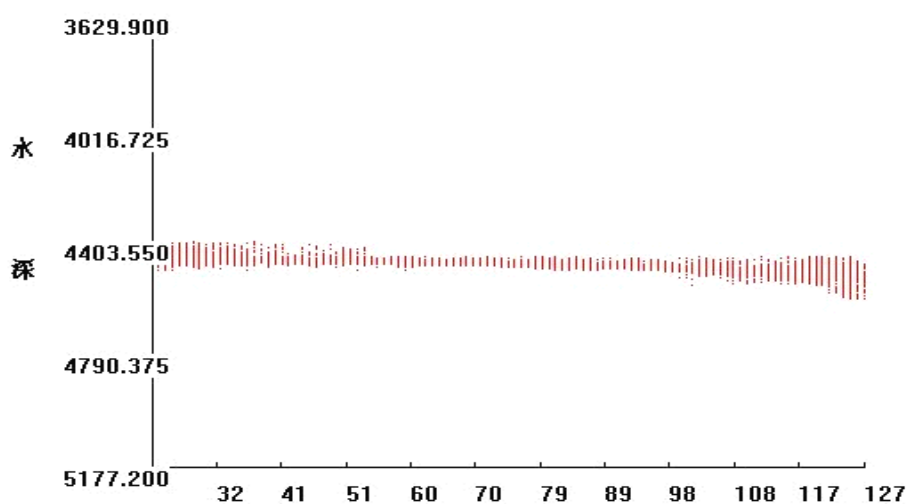


图 4.21 腐蚀膨胀后投影图

Fig. 4.21 the projection after erosion and dilation

最后使用统计滤波方法，滤除小的脉冲噪声。即对每 ping 数据进行一维中值滤波，

并计算其局部方差 $\hat{\sigma}_i$ ，统计阈值即超过 $3\hat{\sigma}_i$ ，则认为是异常数据。由于前几步工作已经滤除了水深数据中所包含的大量异常值，处理后的水深数据中包含有少量的脉冲噪声，需进行中值滤波处理，处理后的效果图，如图 4.23 所示。

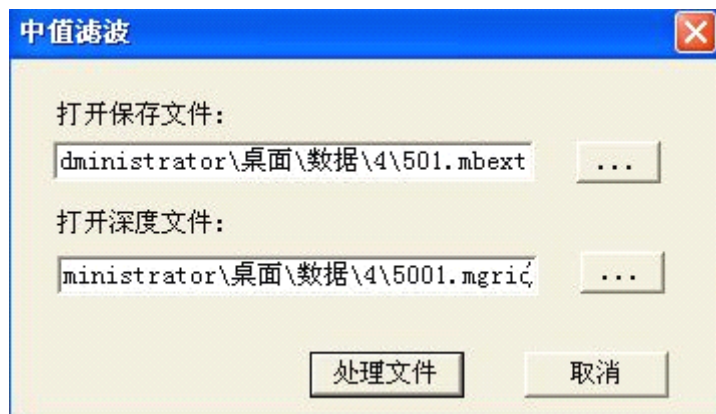


图 4.22 中值滤波对话框

Fig. 4.22 the dialog of median filter

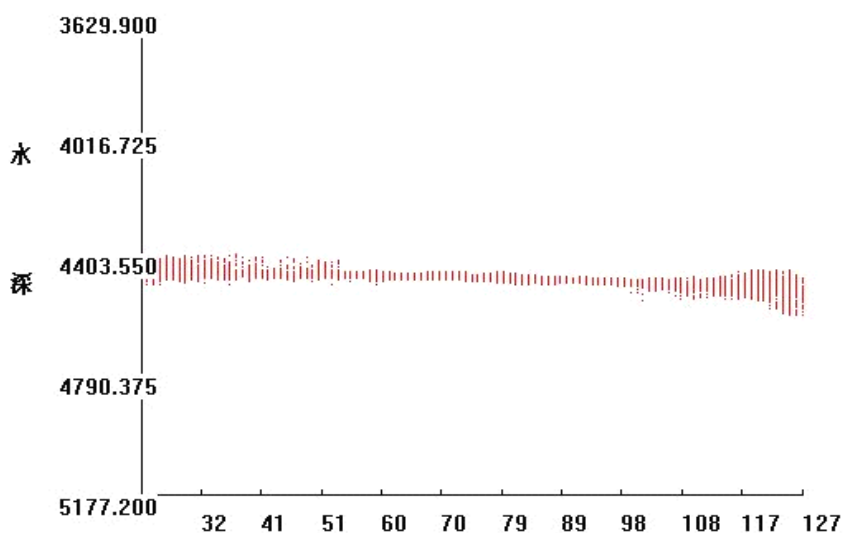


图 4.23 中值滤波后的散点图

Fig. 4.23 the projection after filtering

最终的干净数据见图 4.23，即真实的海底地形。对比图 4.24 和图 4.25，我们可以看到数据编辑前后，水深数据所反映的海底地形有了很大的变化，该算法取得了良好的滤波效果。

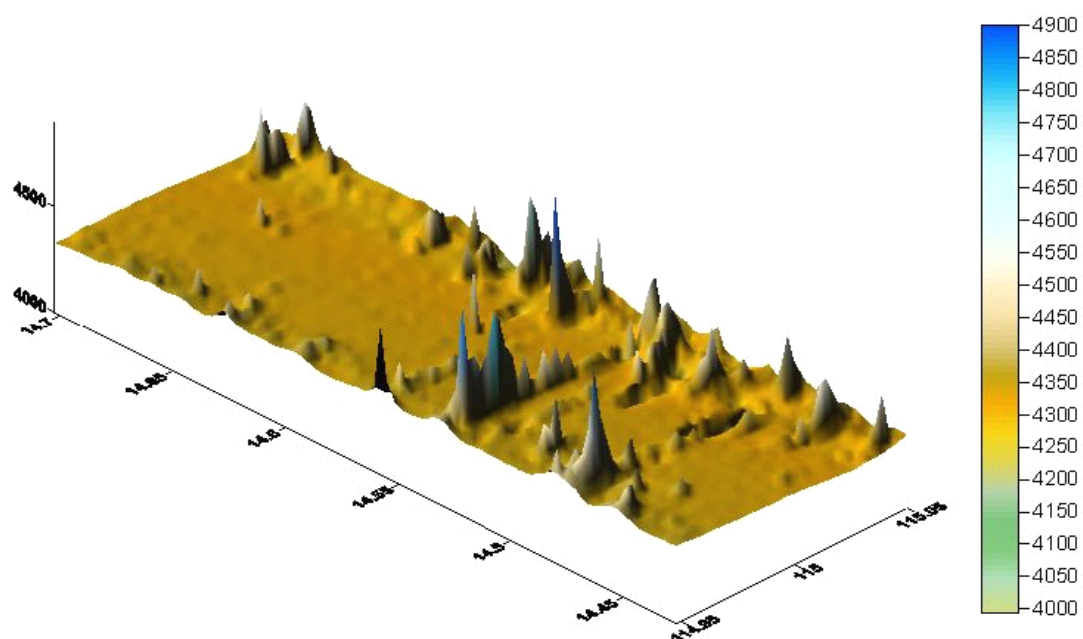


图 4.24 原始数据 3D 图像

Fig. 4.24 the raw 3D seafloor before outliers filtered

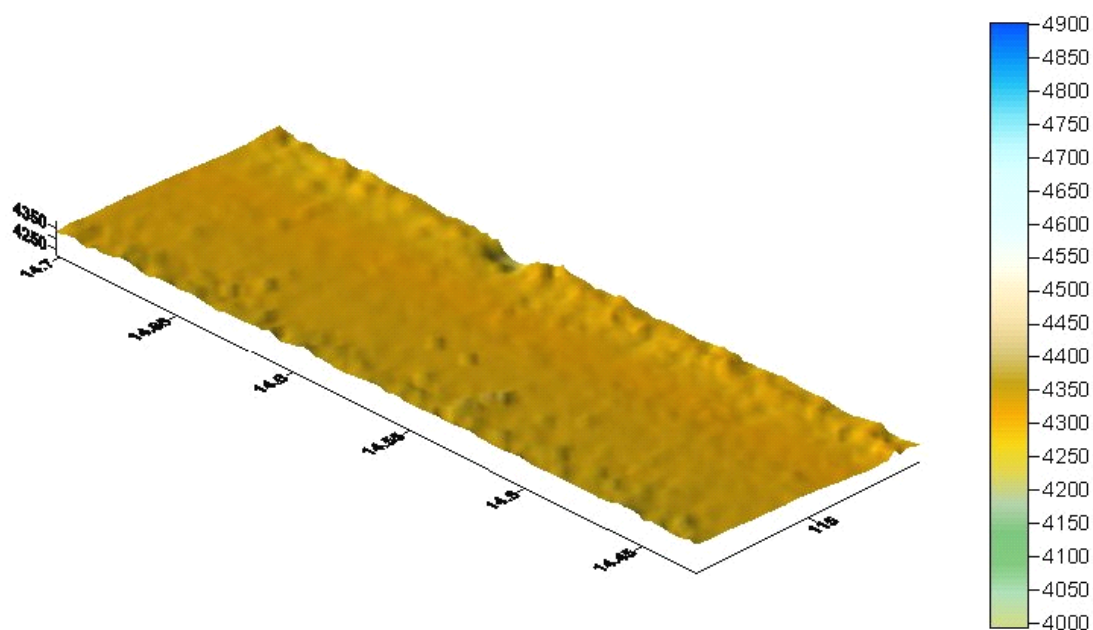


图 4.25 异常数据检测和滤波后的 3D 图像

Fig. 4.25 the final result after outliers automatically filtered

对比表 4.1 和表 4.2 可知，水深数据编辑前后，数据个数由 31403 减少为 30490，标

准偏差由 117.508 减小为 18.3573，可见处理后水深数据的质量提高了。由于该算法是基于密度最大值（即真实数据）所获取的干净的水深值，所以说算法稳健性好、可靠性高、且设置的参数少，处理的速度快。

表 4.1 编辑前数据统计量

Table 4.1 the statistics before editing

数值个数	31403
最小值	3015.4
最大值	5175.6
均值	4325.43
标准偏差	117.508

表 4.2 编辑后数据统计量

Table 4.2 the statistics after editing

数值个数	30490
最小值	4203
最大值	4399.2
均值	4322.26
标准偏差	18.3573

5 结论与展望

5.1 研究成果

海深测量是人类认识海洋世界的基础，其中水深异常数据的滤除，是进行海洋测量最基本的工作，而自动化的滤除方法则是其一直研究的重点。本文结合数字图像处理中噪声滤除的方法，围绕着异常数据的自动检测展开研究工作。

本文研究了多波束异常数据的自动检测方法，主要的研究工作如下：

1、归纳了传统水深数据编辑的方法，包括单波束的编辑方法及现在应用广泛的多波束勘测的数据编辑方法，各自的特点及不足。

2、在分析多波束数据格式的基础上，总结了多波束测深数据的特点，并根据其特点，结合上述滤波方法，提出了基于测深点密度，区域增长、形态学滤波、中值滤波以及局部方差检测相结合的自动异常数据检测方法。该算法利用了交互式编辑可靠性强的优点，以密度投影为基础，运用区域增长算法快速探测异常水深值，搜索正确水深数据的边界线，并通过腐蚀和膨胀算法有效的定位与主块相连的细小的异常数据块，最后采用中值滤波计算局部方差，去除脉冲状的异常数据。实验证明，该算法操作简单，数据稳健性强、可靠性高，运算速度快，自动化程度高，适用于多波束勘测中异常水深值的探测和剔除。

3、*Microsoft Visual C++*是 *Windows* 平台下构建 32 位应用程序的强大的开发工具，其应用十分广泛。尤其 *VC++*是面向对象的可视化编程工具，其代码效率高，可移植性好，并且可以利用功能强大的微软基础类库 MFC 中提供的一整套的对系统操作的封装类，方便的实现自己需要的功能。因此，本文运用 *VC++ 6.0* 作为开发工具，编写该算法，通过实例的分析，说明了该算法的可靠性及实用性，实现了快速有效的自动探测异常数据。

5.2 展望

本文围绕水深异常数据的自动提取做了部分工作，但还有很多问题需要进一步的探

讨:

1、 本文提出的算法进行腐蚀和膨胀处理过程中,结构元素选择时,只考虑了单一的结构元素,即水平或垂直的结构元素,而实际上结构元素的选择是腐蚀和膨胀处理时关键的一个步骤,不同的结构元素处理效果是不相同的。所以,在以后的研究中还需进一步的深入研究不同数据元素的选择,以获得满意的处理效果。

2、 本文提出的算法,与传统的手工交互式编辑的方法比较,速度快,效率高,但是判断准则单一。而针对面积较大的勘测区域,仅用一套单一的判断准则有可能滤除局部的特殊区域,所以实际中最好根据水深范围进行分区处理,以免造成数据的误删。而数据分块处理的方法则有待进一步的探讨。

致谢

岁月如歌，光阴似箭，三年的研究生学习生活即将结束，漫漫十九载的求学路也就走向尾声。作为三年求学生活答卷的这份论文所带给我的启示，值得永远的珍惜，而引导我、帮助我、激励我获得这份启示的人们，也值得永远的铭谢在心。

首先要感谢的是我的导师阳凡林老师，从论文的主题确立到写作到定稿，倾注了导师大量的心血。从导师身上收获是多方面的，从他渊博的学识、严谨的治学中，我体会到了知识与研究的魅力；从他认真负责的工作作风、正直的为人中，我学习到了生活与为人处事的道理。能够遇到这样一位师长与朋友，我为自己感到庆幸，毕业在即，在此谨向阳老师表示我最衷心的感谢！

在此要感谢测绘学院里的老师们，卢秀山教授、于胜文教授、姜岩教授、韩晓东教授、独知行教授、徐泮林教授、成枢教授、刘国林教授、栾元重教授、江涛教授、郑文华副教授、刘智敏副教授等，是你们的教导让我学到了丰富的专业知识，更让我懂得了许多做人的道理。

何当共剪西窗竹，却话巴山夜雨时。我的师弟王煜、田绘波，感谢你们的鼓励和帮助；我的室友们，我的同学们，特别是同学吴静、石娟、周动力、赵勇、高丽丽、景冬，感谢你们的关心和友谊。同门同窗之谊，将是我人生最珍贵的记忆。

最后，需要特别感谢的是我的家人们，感谢父亲、母亲，是他们的爱，激励着我学习、前进。

新的人生旅途就要启程，话别昨天，追寻未来，祝愿我自己，也祝愿亲人、师长、同学和朋友未来一切顺利。

攻读硕士期间主要成果

发表论文情况：

曲萌. 多波束数据处理方法 [J]. 山东科技大学学报, 2007(4).

参加科研项目：

1. 国家自然科学基金项目（40506023）：《水下目标定位和识别的综合研究》
2. 国家自然科学基金项目（40704001）：《遗传算法在 GPS RTK 整周模糊度解算上的应用研究》

参考文献

1. 吴白银, 郑玉龙, 初凤友, 等. 海底浅表层信息声探测技术研究现状及发展 [J]. 地球科学进展, 2005, (11)
2. 金翔龙. 海洋地球物理技术的发展 [J]. 东华理工学院学报, 2004, (3)
3. 刘雁春. 海洋测深空间结构及其数据处理 [D]. 武汉: 武汉测绘科技大学, 1998
4. 李家彪, 等. 多波束勘测原理技术与方法 [M]. 北京: 海洋出版社, 1999
5. Tianhang Hou, Lloyd C Huff, Larry Mayer. Automatic Detection of Outliers in Multibeam Echo Sounding Data [J]. Center for Coastal and Ocean Mapping, University of New Hampshire
6. Scott Shaw, James Arnold. Automated error detection in multibeam bathymetry data [J]. Proc. IEEE oceans, 1993
7. 黄谟涛, 翟国君, 王瑞, 等. 海洋测量异常数据的检测 [J]. 测绘学报, 1999, 28(3)
8. 李明叁, 刘雁春, 暴景阳, 等. 海洋测深数据的抗差 Kaman 滤波 [J]. 测绘科学, 2006, (9)
9. Calder B R, Mayer L A. Robust Automatic Multibeam Bathymetric Processing [S/OL]. [Http://www.thsoa.org/pdf/h01/3_4.pdf](http://www.thsoa.org/pdf/h01/3_4.pdf), 2001
10. 赵建虎. 多波束深度及图像数据处理方法 [D]. 武汉: 武汉大学, 2002
11. 何义斌, 吴书帮, 谢洪燕, 等. 多波束异常测深数据检测方法实践 [J]. 测绘科学, 2004, (2)
12. 胡光海, 周兴华. 趋势面分析在水深测量数据处理中的应用 [J]. 测绘工程, 2004, (9)
13. 黄谟涛, 翟国君, 欧阳永忠, 管铮, 等. 海洋测量误差处理技术研究 [J]. 海洋测绘, 2003, (5)
14. Calder, BR (2003). Automatic Statistical Processing of Multibeam Echosounder Data [J]. Int. Hydro.Review, Vol. 4, No. 1, pp. 53-68

15. Kammerer E, Charlot D, Guillaudeux S, et al. Comparative Study of Shallow Water Multibeam Imagery for Cleaning Bathymetry Sounding Errors [J]. Oceans, MTS/IEEE Conference and Exhibition, 2001
16. 阮锐. 潮汐测量与验潮技术的发展. 海洋技术, 2001, 20(3)
17. 何义斌, 吴书帮, 谢洪燕, 等. 多波束异常测深数据检测方法实践 [J]. 测绘科学, 2004, (2)
18. Schmidt V., Chayes D., Caress D. The MB-System Cookbook
19. M. Mann, P. Agathoklis, and A. Antoniou. Automatic outlier detection in multibeam data using median filtering [J]. IEEE Pacific Rim Conf. Comm., Comp. and Signal Processing, Victoria, pp. 690-693, Aug. 2001
20. 阳凡林, 刘经南, 赵建虎. 多波束测深数据的异常检测和滤波 [J]. 武汉大学学报, 2004, (1)
21. kongsberg Simrad Company. Kongsberg Simrad EM 950 Multibeam Echo Sounder Operator Manual. Kongsberg Simrad company, 1999, 1~10
22. Simrad EM 3000 Multibeam Echo Sounder Baseversion, Kongsberg Simrad, 1999, 20~50
23. 阮秋琦. 数字图像处理学 (第二版) [M]. 北京: 电子工业出版社, 2007
24. 何斌, 马天予, 王运坚, 等. Visual C++ 数字图像处理 (第二版) [M]. 北京: 人民邮电出版社, 2002
25. 贾永红. 计算机图像处理与分析 [M]. 武汉: 武汉大学出版社, 2002
26. 侯俊杰. 深入浅出 MFC (第二版) [M]. 武汉: 华中科技大学出版社, 2001
27. 孙鑫, 余安萍. VC++深入详解 [M]. 北京: 电子工业出版社, 2006
28. 王育坚. Visual C++面向对象编程教程 [M]. 北京: 清华大学出版社, 2003
29. 阳凡林. 多波束水深和图像数据精细处理方法及其应用 [D]. 杭州: 国家海洋局第二海洋研究所, 2007
30. 浅水多波束勘测技术研究 [M]. 青岛: 国家海洋局第一海洋研究所, 1999