

実環境での測定を考慮したレーザスキャナからの点群データを用いたブレイクライン抽出  
Automated extraction of break lines in TLS data of real environment.

北村 和男† Nicola D'Apuzzo‡ 高地 伸夫† 金子 俊一††  
Kazuo Kitamura†, Nicola D'Apuzzo‡, Nobuo Kochi†, Shun'ichi Kaneko††

†(株)トプコン 研究開発センター 画像・計測研究室

‡HOMETRICA CONSULTING ††北海道大学大学院 情報科学研究科

†Imaging and Measuring Laboratory, R&D Center, Topcon Corporation

‡HOMETRICA CONSULTING

††Hokkaido University Graduate School of Information Science and Technology

E-mail: k\_kitamura@topcon.co.jp

### Abstract

我々は、地上据置型レーザスキャナ (Terrestrial Laser Scanner: TLS) から得られた点群データを用いて、建築物などのブレイクラインを抽出する方法についての研究をしている。今回開発したアルゴリズムは、面の抽出 (セグメンテーション) と異なる2面の交線をブレイクラインとして抽出する方法である。我々のアルゴリズムから得られる結果には TLS の計測誤差や計算誤差などが含まれるため、図面作成、モデリングまたは形状理解のアシスト、簡易図作成 (下絵) など、作業の前処理としての位置づけとなる。しかし、それらの前処理においても、現状ではマニュアル作業や複数アプリケーションを用いて行われている。そのため、自動で面やブレイクラインを抽出する本アルゴリズムを用いることで、作業効率の大幅な改善が見込める。また、点群データから面や線などのポリゴンデータに変換することで、データ軽量化の効果も得られる。更に我々のアルゴリズムは、実環境での測定を考慮し、TLS と対象物間を通過する歩行者や車両、草木によるノイズを自動で除去しながら対象物のセグメンテーションとブレイクラインを抽出できるという特徴を有している。

## 1 はじめに

実環境において TLS を用いて取得される点群データには、計測中に TLS と対象物の間を通過する歩行者や車両、草や木によるノイズが含まれる。TLS から得られた点群を用いて、建築物のブレイクラインを抽出したり、モデリングしたりする方法については既に多くの報告がなされている[6][10][11]。しかし、センサーノイズを考慮したり[1]、データに人工的なノイズを付加したりする

[2][3]報告はあるものの、ここで検討しているような歩行者、車両、草や木などによるノイズを考慮した報告はなされていない。このようなノイズが含まれる場合、従来は取得した点群データからマニュアル操作により、これらノイズを除去する必要があった。しかし、近年のレーザスキャナの急速な進歩により、測定対象物から数千万点規模の点群データを短時間で取得できるようになった。また、大規模な対象物の場合、複数視点から計測することも多く、オペレータへの負担は更に大きくなっている。こうした背景より、我々はこのようなノイズを含んだデータからでも対象物の面やブレイクラインを自動で抽出できる方法を開発した。

ここでは、実際の屋内、屋外シーンのサンプルを用いた実験結果を示す。2章では処理全体の概要を、3章でレンジ画像の作成、4章で平面と曲面の抽出、5章でセグメンテーション、6章で交線抽出、7章で実験結果、8章でまとめと今後について述べる。

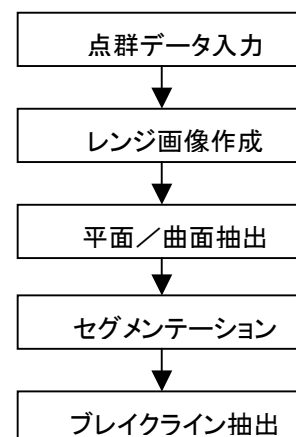


図1 処理のフローチャート

## 2 処理概要

我々のアルゴリズムでは面やブレイクラインを抽出するにあたり、以下の順で処理を進めている。最初に点群データから2次元レンジ画像の作成、次にセグメンテーションの前処理としての平面/曲面抽出、その後にセグメンテーションを行い、最後にブレイクラインの抽出を行っている。(図1参照)。

また、以下の説明で出てくるパラメータは幾つかのサンプルを用いて実験的に決めた値である。

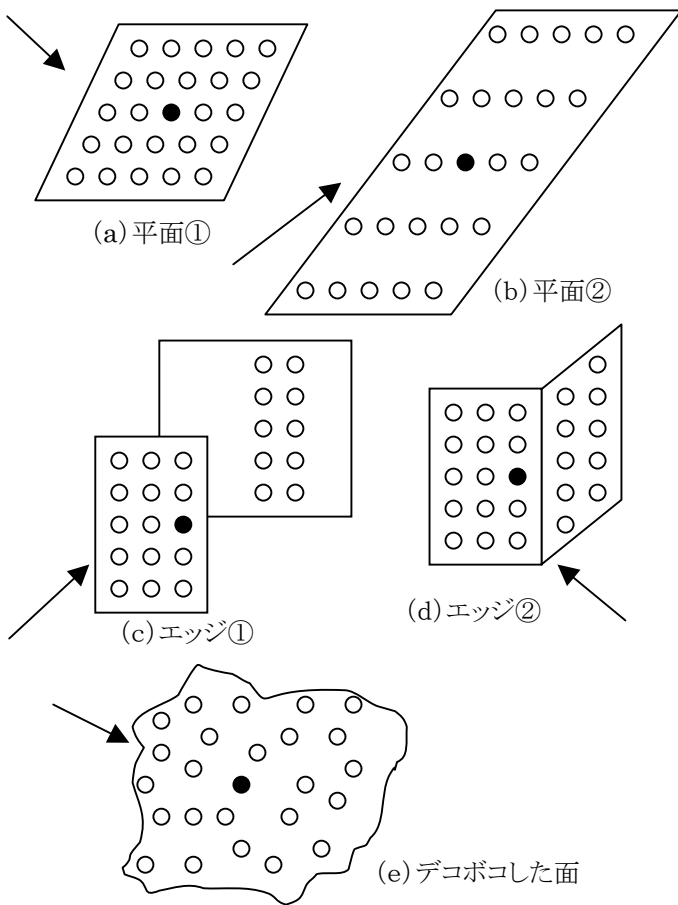


図2 局所面のパターン

黒丸は局所面の中心、矢印はスキャン方向

### 法線ベクトルについて

ここでは、まずセグメンテーションとその前処理で使っている局所面の法線ベクトルの計算方法についての説明を行う。注目点近傍の $n \times n$ 点の3次元座標を用いて、最小二乗法により局所面を計算している。我々の実験では $n$ を5または7としている。この計算を点群データ全点で行う。図2に $5 \times 5$ 点で計算する場合の局所面の状

況を示す。局所面の計算に用いる中心点を黒丸で、近傍点を白丸で示す。矢印は面に対するスキャンの方向である。(a)と(b)はどちらも平面をスキャンした場合である。(a)の場合、対象物の面の法線と近い角度でスキャンしているため、点間距離は非常に短い。一方、(b)の場合、対象物の面に対して浅い角度でスキャンしているため、奥行方向の点間距離が長くなっている。この点が(a)の場合と異なる。ただし、どちらの場合も同一平面上に点があるため、局所面は正しく計算できる。次に、エッジ近傍をスキャンしたのが(c)と(d)である。(c)は、スキャナに近い面Aとその背後にある面Bの境界にあることを示している。我々は、このような部分をジャンプエッジと呼んでいる。また(d)は、面Aと面Bの境界にあることを示している。ただし、(c)とは異なり、それらの面はエッジを境界にして、つながっている。(e)は、デコボコした平面でも曲面でもない面を示している。

### セグメンテーションとその前処理について

ここで、セグメンテーションに話を戻す。点群のセグメンテーション手法には大きく分けて3つある。元々あるモデルに点群を当てはめるモデルベース手法[8][9]、レンジ画像からエッジ検出を行い、それらによって囲まれる部分を1つの領域とするエッジベース手法[2][4][5]、法線ベクトルやレーザの反射強度などを用いて局所的に類似している点をラベリングするリージョンベース手法[6][7][10][11]がある。モデルベースやエッジベースでは、建物の大まかな外観など単純形状のセグメンテーションには向いている。しかし、我々は窓枠や天井、柱などの詳細な構造のセグメンテーションを目的としている。そのため、我々は局所法線ベクトルを用いたリージョンベースのアルゴリズムを採用している。同じくリージョンベースを用いている今野ら[11]は、ラベリングの適切な開始位置の抽出に特徴線抽出を用いている。しかし、それは明確な特徴線で無ければならぬため、平面にしか対応できない。また実環境の場合、面の境界が必ずしも明確な特徴線として抽出できないことも多く、ラベリングに失敗する可能性がある。また Stamos ら[6]は、フィッティング計算に用いる点を、注目点からの距離の固定閾値で決めているが、図2(b)の様に面に対して浅い角度でスキャンした場合、同一平面内でも点間隔は長くなるため、図2(c)のジャンプエッジとの区別がつかなくなり、やはり期待したようなラベリングの結果にならない可能性がある。

一方、我々はラベリングの前処理として、開始点が平面、もしくは曲面上になる様な処理を施している。注目点における局所面のフィッティング精度、曲率、近傍点までの距離を用い、その点が平面や曲面上にいるかどうかの判定を行っている。この判定により、我々はラベリングの開始点を適切に与えることができる。また、初期ラベリングはそういった面上にいと判定された点のみを対象に行われ、注目点と隣接点の法線ベクトルの向きの相違を算出し、その差が予め設定した閾値以下ならば、それらの点に同一のラベル(番号)を付ける。次に、ラベルの付いていない点を対象にラベルの拡張処理を行う。これは、ラベル付きの点と隣接する、ラベルの付いていない点を対象に、注目点近傍のラベル付きの点から局所面を算出し、注目点からその局所面までの距離などを用いて行う。そして、最後にラベル付けされた異なる2平面を取り出し、その交線を求め、ブレイクラインとする。

### 3 レンジ画像作成

TLS は、ビームの放射方向の方位角 $h$ と仰角 $v$ の2つの方向で測定範囲を制御している。スキャンすると、各点における3次元座標 $(x, y, z)$ と反射強度、RGB の色情報が得られる。ここで、図3のように照射方向の $h$ と $v$ を2軸とした平面への投影とみなすと、3次元座標と対応した2次元画像を生成することができる。ここでは、この2次元画像をレンジ画像と呼んでいる。つまり、点群データの1点ごとに、3次元座標 $(x, y, z)$ 、レンジ画像座標 $(h, v)$ 、反射強度、RGB を持つことになる。スキャナによっては放射方向の $h$ と $v$ が等間隔で与えられているものもあり、その場合には TLS から取得されたデータをそのままレンジ画像として用いることができる。しかし、オペレータにより編集されたデータなどでは、レンジ画像上での座標情報の消失や、照射方向が等間隔になっていない場合がある。そのため、その様なデータを扱うにはレンジ画像を作り直す必要がある。そこで、我々のアルゴリズムでは、最初にこのレンジ画像を作成する。指定した解像度で、投影面上に正規グリッドを作り、そのメッシュ交点に最も近い点を登録する。この画像を作ることで、後に行う法線ベクトルの計算やセグメンテーション処理を、2次元の画像処理のように行うことができる。対象物を複数方向からスキャンした場合には、スキャンごとにこのレンジ画像を作成する。通常、解像度は放射方向の範囲とデータ点数から設定するが、意図的に解像度を

変更することも可能である。図4の上を示したのが、元の点群データを3次元空間上に表示したもので、それから作ったレンジ画像が図4の下に示したものである。

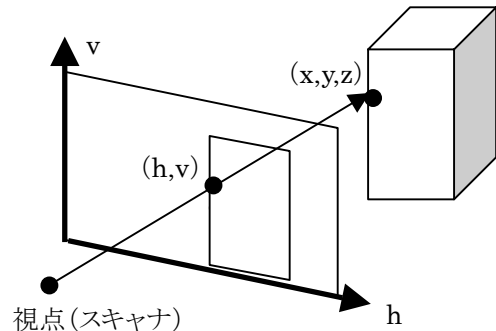


図3 レンジ画像の座標と物体の3次元座標

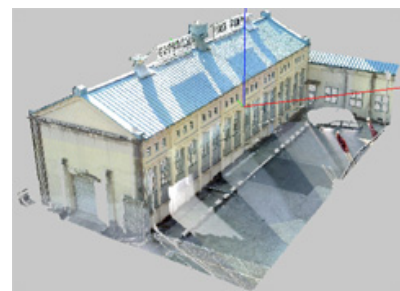


図4 上:スキャンデータ(3D)、下:レンジ画像(2D)

### 4 平面／曲面の抽出

これは、次章のセグメンテーション処理において、ラベリングの適切な開始位置を与えるために行う前処理である。セグメンテーションは、同一の平面、または曲面に属する点群に同一のラベル(番号)を貼る処理である。その判定として、注目点と隣接点の法線ベクトルの向きの相違を用いている。しかし、注目点が図2のデコボコした面(e)やエッジ上やエッジ近傍にある場合(c)(d)、法線ベクトルの計算は間違っている可能性がある。そのため、その様な点を開始点としてラベリングを始めると、処理に失敗する可能性がある。そこで、我々は、以下3つの判定方法で、平面または曲面に属さない点を予め除去し、残りの部分にラベリングを施すことにした。

### (1) フィッティング精度による判定

注目点でフィッティングされた局所面からフィッティングに用いた近傍の各点までの最大距離を、フィッティング精度の判定に用いた。この距離が、予め設定した閾値  $d_{\max}$  よりも大きい場合、この注目点でのフィッティングは正しく行えなかったとみなす。図2(e)のデコボコした部分やエッジ上の点図2(c) (d)の除去が目的である。ここでは  $d_{\max}=0.02[\text{m}]$ とした。

### (2) 曲率を用いた判定

注目点において、近傍  $n \times n$  点の法線ベクトルの各軸方向における大きさの標準偏差 ( $\sigma_{NVx}$ ,  $\sigma_{NVy}$ ,  $\sigma_{NVz}$ ) を求め、更にそれらを用いて、曲率  $C$  を計算する。図2(d)の様なエッジ近傍では法線の向きが変わるため、この値は大きくなる。そのため、そのような位置にある点を除去することを目的としている。まず、レンジ画像全体で各点における曲率を計算し、その最大値  $C_{\max}$  と最小値  $C_{\min}$  を求める。閾値  $C_{th}$  を下記式で計算するが、パラメータとして  $\gamma$  を 0~1.0 の範囲で設定する。ここでは  $n=5$ ,  $\gamma=0.5$  とした。

$$c = \sqrt{\sigma_{NVx}^2 + \sigma_{NVy}^2 + \sigma_{NVz}^2} \quad \text{式(1)}$$

$$c_{th} = (c_{\max} - c_{\min}) \times \gamma + c_{\min} \quad \text{式(2)}$$

### (3) 周辺との距離を用いた判定

最後に、注目点から近傍点までの距離を用いた判定を行う。これは図2(c)に示すようなジャンプエッジの近傍では、点間距離が大きく変化する。そのため、注目点から近傍点までの距離の最大値と最小値を求め、その比(最大値/最小値)を持って判定する。ここではその比の閾値を  $R_{th}=2.0$  とし、この値よりも大きい場合にはジャンプエッジ近傍と判断し、除去した。

## 5 セグメンテーション

### 初期ラベリング

前章で平面または曲面上にあると判断された点を対象に行う。前述した様に、注目点と隣接点の法線ベクトルの向きを比較し、その差が閾値  $\theta_{th}$  以下ならば、それらの点に同一のラベル(番号)を貼る。レンジ画像の左上から処理対象点を探し出し、そこを開始点とする。レンジ画像を使うため、2次元のラベリング処理と同様のアルゴリズムで行える。ここでは  $\theta_{th}=2.0[\text{deg}]$ とした。

### ラベルの拡張

次に、初期ラベリングでラベルの付かなかった点を対象にラベルの拡張処理を行う。前章の前処理により、エッジ近傍などではラベリング処理の対象から除外されてしまうが、近傍の面に属している場合もある。そのため、そういった点にラベルを付け直すのがこの処理である。まず、ラベルのついている点と隣接しているラベルの付いていない点に注目する。注目点近傍( $k \times k$ 点)のラベルの付いている点から、同じラベルを持つ点を用いて局所面を計算する。ラベルが複数ある場合には、ラベルごとに局所面を計算する。次に注目点からそれらラベルの局所面までの距離を計算し、最も近いラベルを選ぶ。ただし、その距離が閾値  $L_{Eth}$  以上ならば、拡張はしない。更に、拡張前後での局所面の法線ベクトルの向きの変化( $\theta_{Eth}$ )と、注目点と近傍ラベルまでの距離と局所面を計算するのに使った近傍領域の平均点間隔の比( $R_{Eth}$ )が閾値以内ならば、注目点は近傍の対象ラベルの面に属していると判断し、拡張する。ここでは  $k=7$ ,  $L_{Eth}=0.01[\text{m}]$ ,  $\theta_{Eth}=1.5[\text{deg}]$ ,  $R_{Eth}=2.0$  とした。

ここまでの処理でラベルの付かなかった点について、ノイズと判断する。ただし、こうした残りの点に対して  $\theta_{th}$  を大きくして、再ラベリングを行うことも可能である。

## 6 ブレイクラインの抽出

レンジ画像を用いて隣接する2つのラベル(面)を取り出し、それらの面の交線をブレイクラインとして算出した。線分の始点と終点の位置は、各ラベルの交線近傍の点を交線に投影させて求めた。しかし、2つのラベルが平行な場合、3次元空間上で離れている場合、対象物の輪郭となる部分では交線を持たない。そのため、ラベルの境界線(輪郭の点を結んだ線)も表示、出力できるようにした。この境界線はラベルを囲むため、閉曲線となっている。

## 7 結果

ここでは室内(図5)、交差点(図6)、建物(図4、7)の3つのサンプルデータに対して、我々のアルゴリズムを適用した結果を示す。それらの図のセグメンテーションの結果は、ラベルごとに色付されている。

実験に用いた TLS はトプコン製 GLS-1000[12]、PC は CPU: Inter Core2 Duo P8700 2.53GHz, RAM: 2.8GB である。各データの点数、処理時間をまとめたものを表

1、元のデータサイズ(txt形式)、ブレイクラインと境界線をdxf形式で出力した場合のサイズを表2に示す。

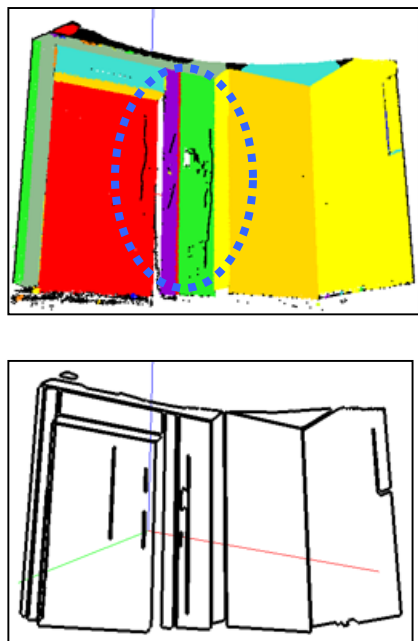


図5 室内サンプルの結果 上:セグメンテーションの結果 黒はノイズを示し、点線で囲まれた部分は通行人によるノイズ 下:ブレイクラインと境界線

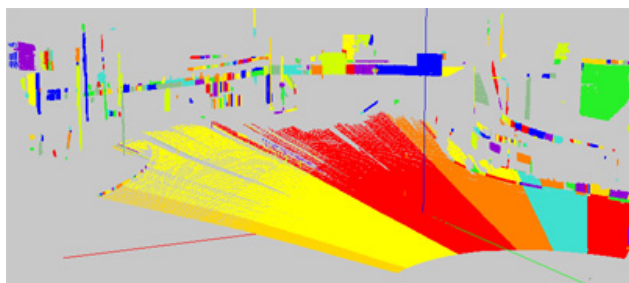
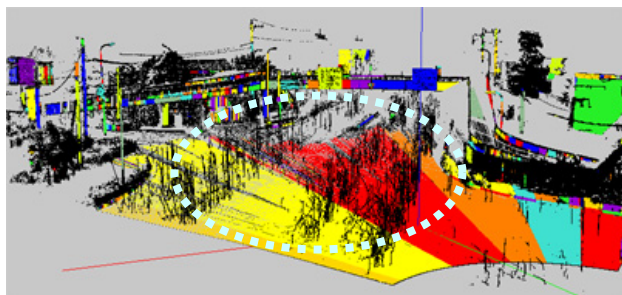


図6 交差点サンプルの結果 上:セグメンテーションの結果 図の点線で囲まれた部分が通過車両によるノイズ 下:上の図からノイズを除去して表示

図5、6より、TLS と対象物間を通過する通行人や車両によるノイズ(黒で表示している点)を除去しているのが分かる。特に図6の交差点のデータでは路面から草が生えているように見える車両によるノイズが大量に含まれている。しかし、我々のアルゴリズムではそういった場合でも、ノイズを除去し、路面のセグメンテーションが行えていることが分かる(図6下)。

サンプル名	点数[千点]	処理時間
室内	181	9 秒
交差点	1276	1 分 34 秒
建物	10673	38 分 12 秒

表1 点数と処理時間

サンプル名	入力サイズ(txt)	出力サイズ(dxf)
室内	4	1
交差点	63	7
建物	378	66

表2 入力と出力サイズ[MB]

表1に各サンプルとその点数と処理時間を示す。建物データでは1000万点を超えているため、処理時間が掛かってしまった。しかし、2章で説明したレンジ画像作成時に解像度を落とすことで、結果の精度は落ちるものの処理時間を短縮させることも可能である。その場合、精度も下がるが、簡易的に形状理解する場合などに用いることはできる。表2には、入力時と出力時のデータサイズを示している。入力時の4~9倍に圧縮することができる。

また、建物のデータは CAD 図面を持っているので、我々のアルゴリズムが出力したブレイクラインと境界線のずれを測定することにした。図面と出力結果から窓枠のコーナーを32点選び出し、トプコン製 3次元計測ソフト Image Master を用いて、図面上のずれを測定した。その結果を表3に示す。

	x	y
RMS	25.9	26.2
MAX	51.5	59.8

表3 建物データの出力結果とCAD図面の比較 図面上のずれのRMSとMAX[mm]

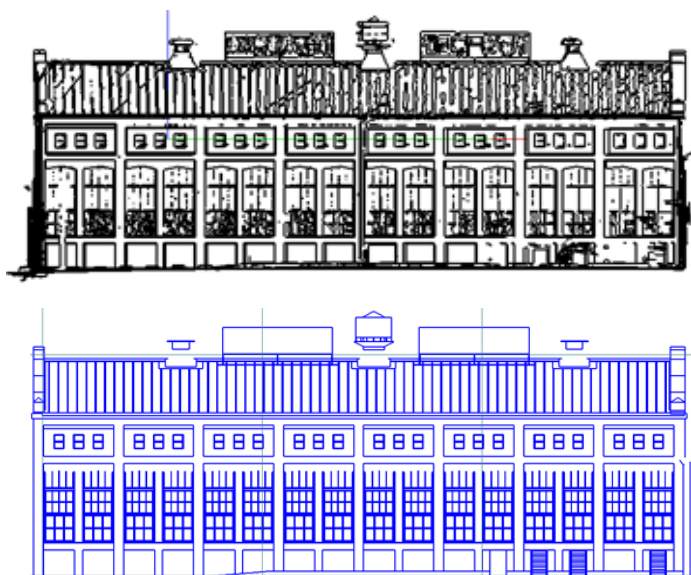


図7 上:建物データのブレイクラインと境界線 下:建物のCADデータ

## 8 まとめ

我々は TLS での測定中に、TLS と対象物間を通過する歩行者、車両、草木によるノイズが含まれた場合でも、それらを除きながら、セグメンテーションやブレイクラインを抽出できるアルゴリズムの開発を行った。室内、交差点の結果より、ノイズが除去されている様子が分かる。特に交差点データ程度のノイズがある場合、従来はマニュアルで数時間の作業時間を強いられていた。しかし、我々のアルゴリズムを用いれば約1分半で処理でき、数十倍以上の時間短縮効果を得られた。

また、建物データでは CAD データと比較したところ、誤差 RMS が 25~26mm であった。これより、我々のアルゴリズムを用いることで、点群データから図面の下絵レベルの結果を得られることが分かった。

更に、点群データをブレイクライン、境界線に置き換えることで、データサイズが元の点群データの 4~9 倍に圧縮することができ、測定現場にて高スペックな PC が無い場合でも、対象物の外観や形状などの確認ができ、形状理解の手助けとなる。建物のサンプルデータでは処理時間が約40分となってしまったが、レンジ画像を作る時の解像度を落とすことで、更に処理時間を短縮することもできる。

我々の究極の目的は点群データからの自動図面作成である。しかしながら、得られる結果には TLS の計測誤差や計算誤差などが含まれるため、現状では形状理

解のアシスト、簡易図作成(下絵)など、作業の前処理としての利用となる。ここでは、そういった前処理として我々のアルゴリズムを利用した場合、従来と比べてどの程度の効果があるのかを示すことができた。

ここでは1視点からの点群データでしか処理を行わなかったが、今後は複数視点から得られた点群データを統合する処理を開発する予定である。

## 参考文献

- [1] X. Jiang, H. Bunke. "Edge Detection in Range Images Based on Scan Line Approximation." *Computer Vision and Image Understanding Vol.73, No.2, February*, pp.183-199, 1999.
- [2] Y. Zhang, Y. Sun, H. Sari-Sarraf, M. Abidi. "Impact of Intensity Edge Map on Segmentation of Noisy Range Images." In *Proc. SPIE Conf. On Three-Dimensional Image Capture and Applications III, Vol.3958m, pp.260-269, Quebec, Canada, 2001*.
- [3] N. Mitra, A. Nguyen. "Estimating Surface Normals in Noisy Point Cloud Data." *Proc. Of the 9<sup>th</sup> Annual Symposium on Computational Geometry, 2003*.
- [4] O. R. P. Bellon and L. Silva. "New Improvements to Range Image Segmentation by Edge Detection." *IEEE Signal Processing Letters, Feb. 2002*.
- [5] A.D.Sappa and M.Devy. "Fast Segmentation by an Edge Detection Strategy." In *Proc. 3<sup>rd</sup> International Conference on 3-D Digital Imaging and Modeling, pp292-299, Quebec, Canada, 2001*.
- [6] I. Stamos and P. K. Allen. "Geometry and Texture Recovery of Scenes of Large Scale." *Journal of Computer Vision and Image Understanding, Nov. 2002*.
- [7] Y. Yu, A. Frensz and J. Malik, "Extracting objects from range and radiance images." *IEEE Trans. Visualization and Computer Graphics, Vol.7, No.4, pp.351-364, 2001*.

- [8] H. Wang and D. Suter, “A model-based range image segmentation algorithm using a novel robust estimator.” In Proc, 3rd International Workshop on SCTV, 2003.
- [9] J. Min and K. Bowyer, “Improved range image segmentation by analyzing surface fit patterns.” Computer Vision and Image Understanding, Vol.97, Issue.2, pp.242-258, 2005.
- [10] C.Chen and I Stamos “Range Image Segmentation for Modeling and Object Detection in Urban Scene.” The 6th International Conference on 3-D Digital Imaging and Modeling, Montreal, Canada, August 21-23, 2007.
- [11] T. Konno, K. Konno, N. Chiba, “Ridge Lines Extraction by Hierarchical Planer Segmentation of Measured Point Clouds.” Nicograph, Vol6, No4, pp.197-206, 2007.
- [12] <http://www.topcon.co.jp/>