

## 多重解像度表現による組合せ最適化を用いた離散平面認識

中力 雅人<sup>†1</sup> 清水 郁子<sup>†1</sup> Rita Zrour<sup>†2</sup>  
剣持 雪子<sup>†3</sup> Hugues Talbot<sup>†2</sup> 杉本 晃宏<sup>†4</sup>

コンピュータビジョンの研究の中で、直線・平面の認識は主要な技術の一つである。離散幾何学の考え方に基づく離散直線・平面認識手法のひとつに、外れ値の存在するデータに対応する Zrour らの手法がある。この手法は混合整数計画問題として定式化されており、直線・平面を構成しうるすべての点の組合せの中で最適解を求めることができるという利点があるが、点の数や外れ値の数が増えると、計算時間が増大することが指摘されている。そこで本論文では、多重解像度表現を用いて、混合整数計画問題として定式化された離散直線・平面認識の計算時間を短縮する方法を提案する。本手法では、粗い解像度で直線・平面のパラメタの探索空間、および、直線・平面に含まれる点の候補を絞る。実験により、提案手法で計算時間を大幅に削減することを確認した。

### Digital plane recognition using combinatorial optimization by multiresolutional representation

MASATO CHURIKI,<sup>†1</sup> IKUKO SHIMIZU,<sup>†1</sup> RITA ZROUR,<sup>†2</sup>  
YUKIKO KENMOCHI,<sup>†3</sup> HUGUES TALBOT<sup>†2</sup>  
and AKIHIRO SUGIMOTO<sup>†4</sup>

Line and plane recognition is one of essential tasks in the field of computer vision. While methods for digital plane recognition utilizing digital geometry can detect the optimal solution, they require enormous computational time. In this paper, we propose a method to reduce a computational time of digital line/plane recognition as mixed integer linear programming problem using multiresolutional representation. Our method reduces both the outliers of points and the search space of the line/plane parameters at low resolution. Experimental results show that the computational time was much reduces by our method.

### 1. はじめに

コンピュータビジョンの研究の中で、直線あるいは平面の認識は主要な技術の一つであり、物体の認識<sup>1)</sup>、画像分割<sup>2)</sup>、パラメタ推定<sup>3)</sup>などに応用されている<sup>4)</sup>。直線認識とは、2次元空間の点集合が与えられたときに、その部分集合で構成される直線を求めるものである。同様に、3次元空間中の点集合が与えられたとき、その部分集合で構成される平面を求めることを平面認識という。

コンピュータビジョンにおける直線・平面認識の研究において、過去には最小二乗法<sup>5)</sup>、M推定<sup>6)</sup>、あるいは Hough 変換<sup>7)</sup> や RANSAC<sup>8)</sup>、LMedS 推定<sup>9)</sup> といった数多くの手法が提案されてきた。一方で、離散幾何学の分野でも直線・平面認識が研究されてきた<sup>10)-15)</sup>。離散幾何学における直線・平面は離散直線・離散平面と呼ばれ、統計的手法によって求められる近似直線とは異なり、2つの平行なユークリッド直線・平面の間にある点の集合として定義される<sup>16)</sup> (図1)。

離散平面認識の手法として、与えられた点集合が平面であるかそうでないかを求める手法<sup>10)-13)</sup>、与えられた点集合のすべての点から構成されるできるだけ小さい平面の厚みを求める手法<sup>14),15)</sup> が数多く研究されている。これらの方法では与えられた点集合には外れ値が含まれないことが仮定されている。これに対し、Zrour らによって提案された手法<sup>17)</sup> は、外れ値が含まれる場合に適用可能であり、あらゆる可能な点の組合せの中から、直線に含まれる点の数が最大になるような組合せを求める。この手法では、直線・平面認識を混合整数計画問題として定式化しており、与えられた点集合に外れ値や複数の離散直線あるいは平面が含まれていても、それらを識別し、最適解を求めることができるという利点がある。しかし、点の数や外れ値の数が増えると、計算時間が増大することが指摘されている。

そこで、本論文では、多重解像度表現を利用することによって、混合整数計画問題として定式化された離散直線・平面認識の計算時間を短縮する手法を提案する。本手法では、低い解像度で、直線・平面のパラメタの探索空間の削減、および、直線・平面に含まれる点の候補の絞り込みを行う。このとき、パラメタ探索空間の削減は、より高い解像度での混合整数計画問題に制約式を追加することにより実現する。また、点の候補の絞り込みは、低い解像度

<sup>†1</sup> 東京農工大学 大学院 Graduate School, Tokyo University of Agriculture and Technology

<sup>†2</sup> 電子電気工学技術高等学院 ESIEE

<sup>†3</sup> 仏国立科学研究所 CNRS

<sup>†4</sup> 国立情報学研究所 National Institute of Informatics

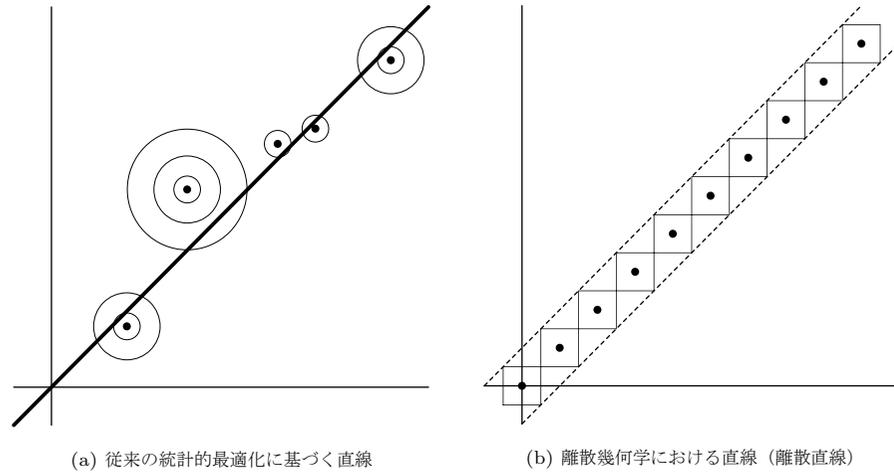


図1 ユークリッド幾何学と離散幾何学における直線  
Fig.1 A euclidean line and discrete line

で直線に含まれると判定された点に対応する点のみに限定することで行う。

## 2. 離散幾何学における直線と平面の表現

ユークリッド空間  $\mathbb{R}^n$  ( $n = 2, 3$ ) における直線  $\mathbf{L}$  や平面  $\mathbf{P}$  は,  $a, b, c \in \mathbb{R}$  とすると, それぞれ次の式で定義される:

$$\mathbf{L} = \{(x, y) \in \mathbb{R}^2 : ax + y + b = 0\}, \quad (1)$$

$$\mathbf{P} = \{(x, y, z) \in \mathbb{R}^3 : ax + by + z + c = 0\}. \quad (2)$$

上の連続なモデルに対して, 離散幾何学では離散モデルを扱う。離散幾何学では, 座標は整数値しか取らない。すべての整数値の集合を  $\mathbb{Z}$  とし, 離散空間  $\mathbb{Z}^n$  ( $n = 2, 3$ ) を考える。このとき,  $\mathbf{L}$  の離散化である離散直線  $\mathbf{D}(\mathbf{L})$  は,  $a, b, \omega \in \mathbb{R}$  とすると, 次の式で定義される:

$$\mathbf{D}(\mathbf{L}) = \{(x, y) \in \mathbb{Z}^2 : 0 \leq ax + y + b \leq \omega\}. \quad (3)$$

$\mathbf{D}(\mathbf{L})$  は 2 つの平行な直線  $ax + y + b = 0$  と  $ax + y + b = \omega$  の間に存在する格子点の集合であり, これらの直線の間  $y$  軸 (垂直) 方向の距離は  $\omega$  である。このとき,  $y$  軸を主軸という。これに対し,  $x$  軸 (水平) 方向が距離  $\omega$  の場合には, 単純に式 (3) の  $x$  と  $y$  を入れ替えればよい。このとき, 主軸は  $x$  軸となる。

同様に,  $\mathbf{P}$  の離散化である離散平面  $\mathbf{D}(\mathbf{P})$  は,  $a, b, c, \omega \in \mathbb{R}$  とすると, 次の式で定義さ

れる:

$$\mathbf{D}(\mathbf{P}) = \{(x, y, z) \in \mathbb{Z}^3 : 0 \leq ax + by + z + c \leq \omega\}. \quad (4)$$

主軸の入れ替えについても同様に,  $x, y, z$  の置換によって実現できる。

## 3. 混合整数計画問題による離散直線・平面認識の定式化

### 3.1 混合整数計画問題とは

1 次式で表される等式や不等式の制約条件の下で, 1 次式で表される目的関数を最小化, もしくは最大化する最適化問題のことを線形計画問題という。すなわち, 行列  $\mathbf{A}$  とベクトル  $\mathbf{b}, \mathbf{c}$  が与えられた時に, 制約条件  $\mathbf{Ax} \geq \mathbf{b}, \mathbf{x} \geq 0$  のもとで,  $\mathbf{c}^T \mathbf{x}$  を最小化, もしくは最大化するベクトル  $\mathbf{x}$  を求める問題である。

特に,  $\mathbf{x}$  が整数値のみをとるとき整数計画問題, 整数値と実数値をとる変数が混在するとき混合整数計画問題と呼ばれる。線形計画問題には多項式時間アルゴリズムが存在するのに対し, (混合) 整数計画問題は NP 困難であることが知られている。しかし, 近年の計算機パワーの増大に加え, 最適化アルゴリズムそのものが格段に進化していることから, 最先端のアルゴリズムを実装した最適解ソルバーの性能は数年前に比べて飛躍的に向上している<sup>18)</sup>。

### 3.2 混合整数計画問題としての定式化

まず, 離散直線認識について考える。  $N$  を点集合の要素数とし,  $p_i$  ( $i = 1, \dots, N$ ) を,  $(x_i, y_i)$  が離散直線の不等式 (3) を満たすとき  $p_i = 0$ , 満たさないとき  $p_i = 1$  となるような 2 値変数とする。これにより, 直線に含まれる点の数が最大になる (外れ値の数が最小になる) 直線を求める問題は, 次のような混合整数計画問題で表すことができる:

$$\text{minimize } \sum_{i=1, \dots, N} p_i \quad (5)$$

$$\text{subject to } -Mp_i \leq ax_i + y_i + b \leq Mp_i + \omega \quad \text{for all } i = 1, \dots, N \quad (6)$$

$$p_i \in \{0, 1\}. \quad (7)$$

ただし,  $M$  は十分に大きな定数であるとする。

点  $(x_i, y_i)$  が離散直線に含まれている場合,  $p_i = 0$  であるので, 式 (6) は,

$$0 \leq ax_i + y_i + b \leq \omega \quad (8)$$

となり, 式 (3) の形になる。一方, 点が離散直線に含まれていない, すなわち  $p_i = 1$  なら, 式 (6) は,

$$-\infty \leq ax_i + y_i + b \leq \infty \quad (9)$$

となる。このような, 十分大きな定数  $M$  を用いた定式化の手法を big-M 法と呼ぶ<sup>18)</sup>。

同様に離散平面認識についても、 $N$  を点集合の要素数とし、 $p_i (i = 1, \dots, N)$  を、 $(x_i, y_i, z_i)$  が離散直線の不等式 (4) を満たすとき  $p_i = 0$ 、満たさないとき  $p_i = 1$  となるような 2 値変数とする。これにより、平面に含まれる点の数が最大になる (外れ値の数が最小になる) 平面を求める問題は、次のような混合整数計画問題で表すことができる:

$$\text{minimize } \sum_{i=1, \dots, N} p_i \quad (10)$$

$$\text{subject to } -Mp_i \leq ax_i + by_i + z_i + c \leq Mp_i + \omega \quad \text{for all } i = 1, \dots, N \quad (11)$$

$$p_i \in \{0, 1\}. \quad (12)$$

以下では、幅  $\omega$  が 1 であるとして説明する。ただし、任意の幅で同様のアルゴリズムが適用可能である。

#### 4. 多重解像度表現の適用

##### 4.1 ダウンサンプリングによる多重解像度表現

混合整数計画問題の最適化ソルバーが高速になったといっても、前節のように定式化された離散平面・直線認識を行った場合に、非常に時間がかかることがわかっている<sup>17)</sup>。特に、外れ値の割合が増えた場合には、現実的な時間で解くことができない。そこで、本論文では、多重解像度表現を用いて計算時間を削減する。

多重解像度表現を得るために、本論文では、以下に示すようにダウンサンプリングを行う。まず、元の解像度に対し、 $1/2$  の解像度の表現を得るときには、図 2 のように、高い解像度で  $(2x_i, 2y_i), (2x_i + 1, 2y_i), (2x_i, 2y_i + 1), (2x_i + 1, 2y_i + 1)$  のいずれか 1 つでも点が存在するとき、低い解像度で  $(x_i, y_i)$  に点が存在すると考える。このように定義することで、解像度を落とした画像は一意的に得られる。ダウンサンプリングを  $k$  回繰り返すことで、 $2^{-k}$  倍の解像度の点集合を得る。

##### 4.2 高解像度への反映について

提案手法では、低い解像度で求めた離散直線・離散平面の認識結果を、高い解像度での計算に反映させることで計算時間を短縮する。具体的には、以下に説明するように、点集合の削減および直線・平面のパラメタの探索空間の削減を行う。

###### 4.2.1 点集合の削減

低い解像度で求めた離散直線または離散平面のパラメタを用いて、高い解像度でその離散直線または離散平面に含まれるであろう点をサンプリングし、その点だけで離散直線あるいは離散平面認識を行う。これで、外れ値であろう点の数を減らすことができるので、計算時

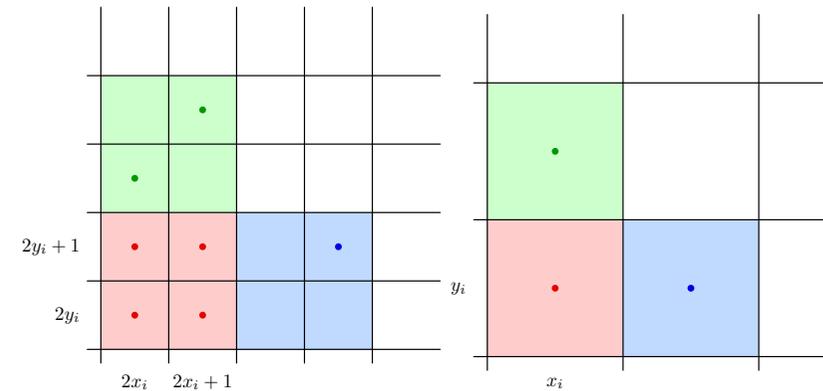


図 2 ダウンサンプリングのモデル  
Fig. 2 A model of downsampling

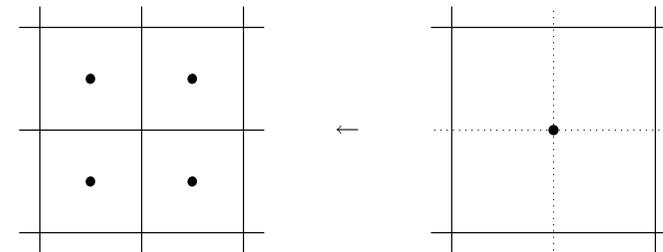


図 3 異なる解像度での対応点  
Fig. 3 Corresponding points in different resolutions

間の大幅な短縮が期待される。これは、高い解像度で最も点の数が多き離散直線や平面は、低い解像度でも同様であるという根拠に基づくものである。

2次元で考えると、低い解像度で求められた離散直線のパラメタを  $a, b$  とすると、その 1 つ上の解像度で直線に含まれる可能性のある点は、次の式を満たす。

$$0 \leq a(\lfloor x/2 \rfloor + 0.5) + (\lfloor y/2 \rfloor + 0.5) + b < 1 \quad (13)$$

ここで、低い解像度でのある点は、その一つ上の解像度では、そのまわりに存在する 4 つの点であった可能性がある (図 3)。そのため、座標を 2 で割るときには、ダウンサンプリングのモデルから、床関数  $\lfloor x \rfloor$  を使って整数値にする必要がある。

式 (13) を満たす点のみを入力点集合とし、混合整数計画問題を繰り返し解くことで直

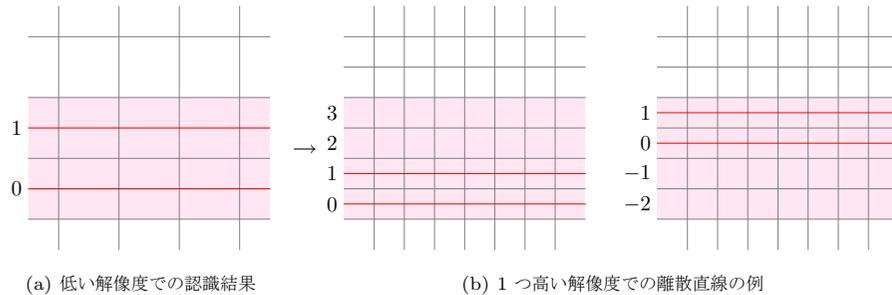


図 4 異なる解像度の離散直線の関係  
Fig. 4 Relationship between lines of different resolutions

線を検出する。なお、幾何学的構造が崩れるほどに解像度を落としてしまうと、正しい結果が得られない可能性があるので注意しなければならない。

また、外れ値が点集合全体に散らばっているような場合には、解像度を落とすことで点の数が減っても、外れ値の数は減らないので、結果として点の数に対する外れ値の割合が高くなってしまふことが考えられる。

#### 4.2.2 パラメタの探索空間の削減

前節では、低い解像度で求めたパラメタから、点集合の点の数を減らすことで計算時間を短縮したが、混合整数計画問題のパラメタの探索空間も同様に削減することで、計算時間のさらなる短縮を狙う。

2次元で考えると、ある解像度の一つ低い解像度で求められた直線は、ある解像度では幅が3倍の直線になると考えられる。例として、図4(a)で赤い離散平面が求められたとすると、点が存在する可能性のある領域は、薄赤で塗りつぶされた部分となる。1つ高い解像度でも薄赤の領域に点があるはずなので、その解像度で離散直線認識をしたとき、図4(b)に示すような赤い離散直線が求められる。すなわち、低い解像度で  $(x_i, y_i)$  が直線に含まれるとしたら、求めたいパラメタ  $a, b$  は次の不等式を満たすはずである。

$$-2 \leq ax_i + y_i + b \leq 3 \quad (14)$$

そこで、離散平面認識においては式(6)を混合整数計画問題の制約条件として使っているが、前述した理由から、一つ低い解像度で直線に含まれるとされた点について、さらに式(14)の条件を加えることができる。

```

Input:  $n$  次元 ( $n = 2$  or  $3$ ) の  $N$  個の点を含む集合  $S$ , 初期解像度  $1/2^k$ 
Output:  $n$  次元超平面のパラメタと  $S$  のそれぞれの点についての  $p_i$ 
1  foreach 仮定した主軸それぞれ  $x, y, z$  軸について do
2   $S$  から解像度  $1/2^k$  の点集合  $S^k$  を生成
3  混合整数計画問題を解き, パラメタを求める
4  end foreach
   上で求めた中でもっとも外れ値が少ない軸について
5  for  $r = k - 1, k - 2, \dots, 1$  do
6  1つ低い解像度で求められたパラメタを基に, 解像度  $1/2^r$  の点集合  $S^r$  を生成
7  混合整数計画問題を解き, パラメタを求める
8  end for
9  求めたパラメタを最適値とする
10 return
    
```

図 5 多重解像度表現を用いた混合整数計画問題による離散直線・平面認識のアルゴリズム  
Fig. 5 An algorithm for digital line/plane recognition using combinatorial optimization by multiresolutional representation

#### 4.3 連結性の利用

提案手法では多重解像度表現を用い、解像度を落とすことで外れ値を減らし、計算時間を短縮することを狙っているが、直線や平面が密に存在するような点集合では、解像度を落とすことで空白の(点のない)領域が埋まってしまう、直線や平面でなくなってしまうことがある。これは、解像度を落とすすぎないことで対応できるが、解像度を落とさないより計算に時間がかかってしまう。

そこで、実データのように膨大な数の点を扱う場合に、8近傍(あるいは26近傍)すべてに点が存在するようなある点は、そのある点と8近傍(あるいは26近傍)すべての点を計算には用いないこととする。

#### 4.4 多重解像度表現を用いた混合整数計画問題による離散直線・平面認識のアルゴリズム

主軸が不明なため、初期解像度においてはすべての軸について混合整数計画問題を解く。それ以降は、最も点の数が多(外れ値の数が少ない)軸のみを対象として離散直線・平面認識を行う。最適解が複数得られた場合には、それぞれのパラメタについて高い解像度で計算する。

前節までに述べた手法を用い、図5のようなアルゴリズムで離散直線・平面認識を行う。

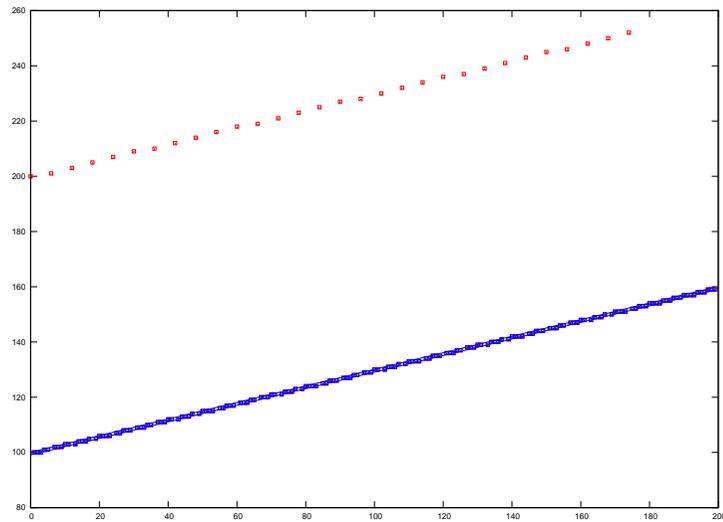


図 6 2次元合成データの例とその認識結果 (赤: 外れ値, 青: 離散直線)

Fig. 6 An example of 2D synthetic data and a recognized line by our method (red: outliers, blue: digital line)

## 5. 実験と考察

提案手法を評価するため, 実際に合成データに対して離散直線・平面認識を行う. なお, 実行時間にデータの読み込みなどを含んでいる.

### 5.1 合成データに対する離散直線認識

手動で生成した点集合を入力として, 提案手法と, 多重解像度表現を用いない手法 (単一解像度) で離散直線認識を行い, 計算時間を測定, 比較する. なお, 混合整数計画問題のソルバーとして SCIP Verson 1.10<sup>\*1</sup>を用いた.

使用する点集合の例を図 6 に示す. 点集合は, (A) 外れ値の数を 10 に固定して, 点の数を 100, 200, 300, 400, 500 と変化させたものと, (B) 点の数を 200 に固定して, 外れ値の数を 10, 20, 30, 40, 50 と変化させたものを用意した.

(A) の計算時間のグラフを図 7 に, (B) の計算時間のグラフ図 8 に示す. なお, 提案手法

\*1 <http://scip.zib.de/>

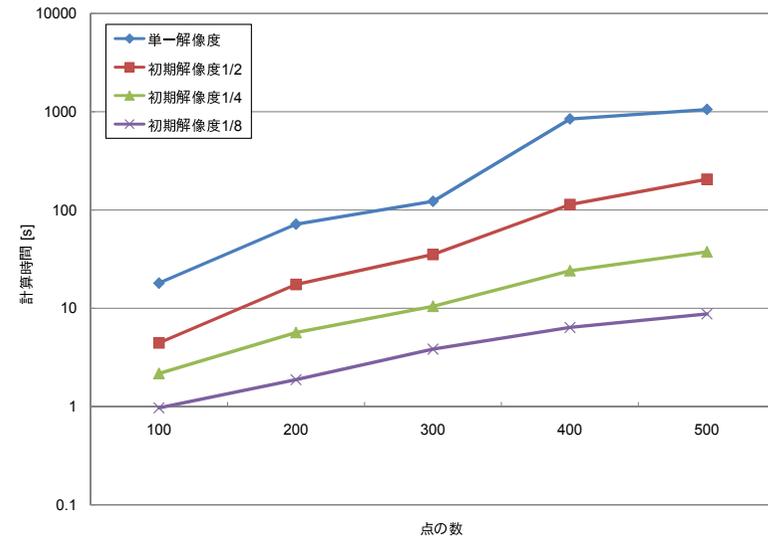


図 7 離散直線認識における点の数を変化させたときの計算時間との関係 (外れ値の数 10)

Fig. 7 Relation of computational time to number of points for digital line recognition (the number of outliers is 10)

と, 多重解像度表現を用いない手法とともに同じ離散直線が得られた. 図 7, 図 8 ともに縦軸は対数目盛りとなっており, 単一解像度の場合の計算時間に比べて, 初期解像度 1/8 の場合の提案手法は計算時間が大幅に短縮されていることがわかる.

### 5.2 合成データに対する離散平面認識

離散直線認識と同様に, 手動で生成した 3 次元の点集合についても, 比較実験を行った. なお, 混合整数計画問題のソルバーとして SCIP Verson 1.10 を用いた.

点集合の例を図 9 に示す. 点集合は, 点の数を 50 に固定して外れ値の数を 10, 20 と変化させたものと, 点の数を 100 に固定して外れ値の数を 10, 20, 30, 40, 50 と変化させたものを用意した.

計算時間のグラフを図 10 に示す. なお, 提案手法と, 多重解像度表現を用いない手法 (単一解像度) とともに同じ離散直線が得られた. 図 10 の縦軸は対数目盛りとなっており, 単一解像度の場合の計算時間に比べて, 初期解像度 1/8 の場合の提案手法は計算時間が大幅に短縮されていることがわかる.

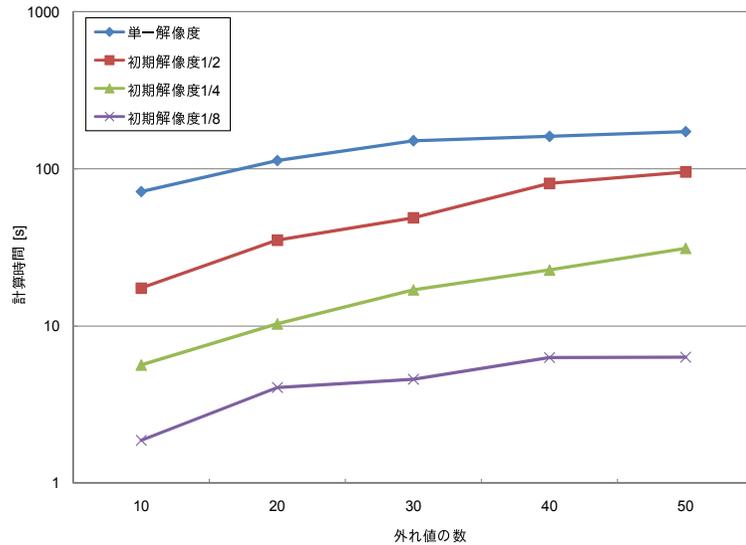


図 8 離散直線認識における外れ値の数を変化させたときの計算時間との関係 (点の数 200)

Fig. 8 Relation of computational time to number of points and outliers for digital plane recognition (the number of points is 200)

### 5.3 実データに対する離散直線・平面認識

参考までに、提案手法を図 11 に示す実データに対して適用した。2次元の点集合に対する離散直線認識の結果を図 12 に、3次元の点集合に対する離散平面認識の結果を図 13 に示す。この節の離散直線・平面認識には、混合整数計画問題のソルバーとして lp\_solve\*1 を用いた。

実データは点の数や外れ値の数が合成データで使用した点集合よりも多いため、計算時間も長く必要となる。また、3次元の点集合は、レンジセンサを用いて得られた計測データを元としているが、多重解像度を用いてもこのような点集合から離散平面を認識することは時間上難しいので、事前に点の数を減らすなど前処理を施している。

### 5.4 RANSAC との比較

提案手法と RANSAC を用いて点集合を離散直線認識し、比較する。使用する点集合は

\*1 <http://lpsolve.sourceforge.net/5.5/>

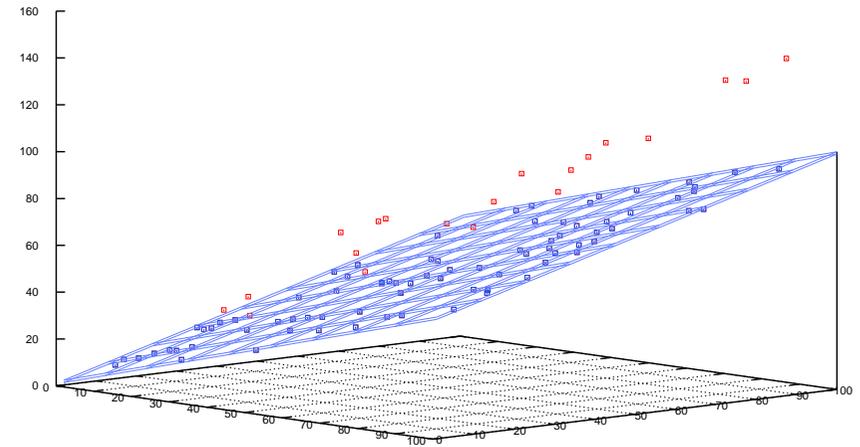


図 9 3次元合成データの例とその認識結果 (赤: 外れ値, 青: 離散直線)

Fig. 9 An example of 3D synthetic data and a recognized plane by our method (red: outliers, blue: digital line)

傾きを変えた角丸四角形を用いた。ほとんどの点集合で同じ結果が得られたが、傾きを  $45^\circ$  にしたときに RANSAC では最適解が得られなかった。その結果を提案手法、RANSAC それぞれ図 14, 図 15 に示す。離散直線の色は認識順であり、特別な意味はない。RANSAC は実際には 1 つの直線を求め、その  $\pm 0.5$  の幅に含まれる点が多ければ直線を構成していると考えられる。提案手法では最適解が得られているのに対し、RANSAC では離散直線の両端付近で離散直線に含まれていない点があることがわかる。これは、ランダムに 2 点をサンプリングし、直線を求めるという手法のためである。

### 5.5 考 察

2次元、3次元いずれの結果とも、点の数や外れ値の割合が増えると、計算時間が急増していることがわかる。また、多重解像度表現を用いることで、確実に計算時間が短縮できている。

もし係数の大小関係だけでも分かっていたら、距離 1 の軸方向を特定できるため、解かなければならない混合整数計画問題の数も減らすことができる。また、解く必要のない混合整数計画問題を解くことは、非常に外れ値の多い点集合から離散直線・平面を見つけることとであり、計算時間増大の大きな要因であるため、係数の大小関係を事前に求めることができ

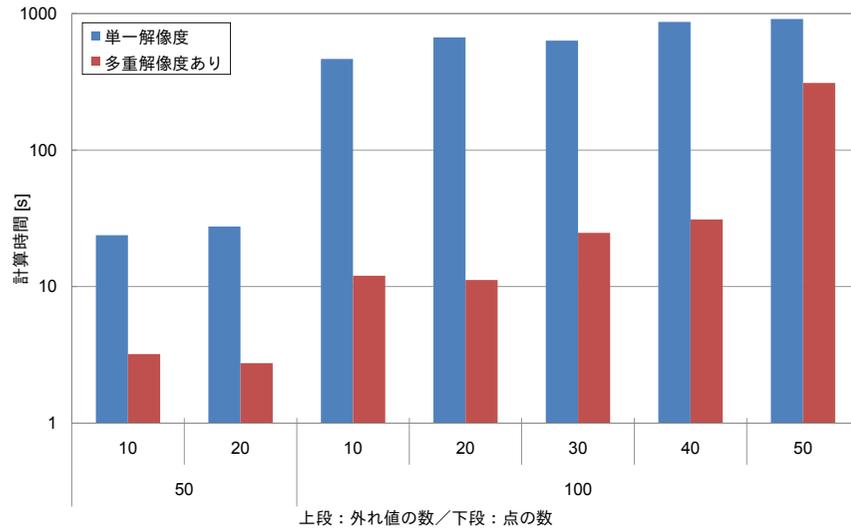


図 10 離散平面認識における点の数および外れ値の数と計算時間との関係

Fig. 10 Relation of computational time to number of points and outliers for digital plane recognition

れば、大幅な計算時間の短縮が期待できる。

初期解像度をいくりにするかは、今のところ点の数や座標等から経験的に求めるしかない。解像度を落とさないと計算に時間がかかり、落としすぎると幾何学的構造が崩れ、望ましい結果が得られなかったり、実行不可能解となってしまうので注意が必要である。なお、解像度を落とすと直線や平面の幅が 1 より大きくなることもあるが、最終的には幅 1 の直線や平面に含まれるので問題ない。ただし、低い解像度での計算時に外れ値の数が増えることになるので、計算時間も増える可能性がある。

## 6. おわりに

本論文では、混合整数計画問題によって定式化された離散直線・平面認識の手法の計算時間を、多重解像度表現を用いることで短縮する手法を提案した。従来手法は計算に時間がかかることが問題だったが、多重解像度表現を用いて、少ない点集合の低い解像度で大まかに直線を求め、高い解像度の点集合やパラメタ空間を削減するという方法によって、計算時間



図 11 2次元実データ

Fig. 11 2D real data

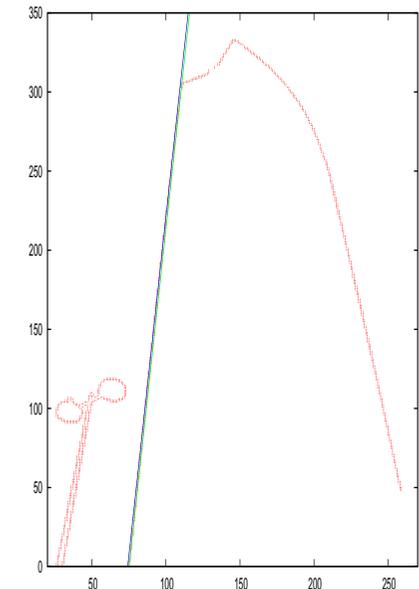


図 12 2次元実データに対する離散直線認識

Fig. 12 Digital line recognition for 2D real data by our method

の削減が期待できる。

実際に提案手法を 2 次元の合成データに適用した結果、大幅に計算時間を短縮できることが確認された。また、離散直線や離散平面が検出され、認識結果に問題がないことも確認された。

ただし、 $x$ ,  $y$ ,  $z$  方向それぞれの値の分散が同程度でない場合、解像度を落としすぎると分散が小さい軸方向での幾何学的構造が維持できなくなってしまうなどの問題点がある。

また、幅を推定できるような手法を考案することが今後の課題である。

## 参考文献

- 1) Faugeras, O. and Hebert, M.: The representation, recognition, and locating of 3-d objects, *International Journal of Robotics Research*, Vol.5, No.3, pp.27-52 (1986).

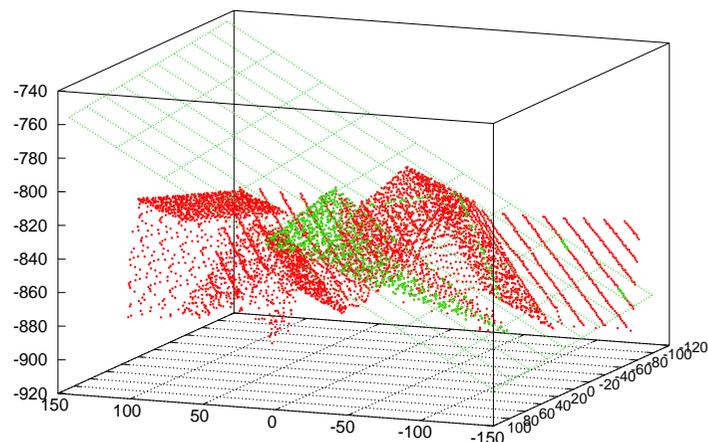


図 13 3次元実データに対する離散平面認識  
Fig. 13 Digital plane recognition for 3D real data

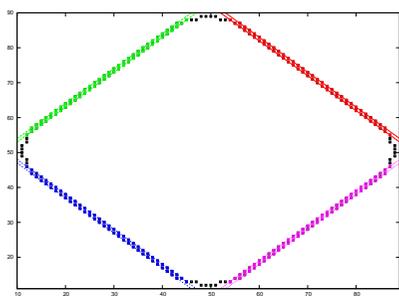


図 14 提案手法による離散直線認識  
Fig. 14 Line recognition by our method

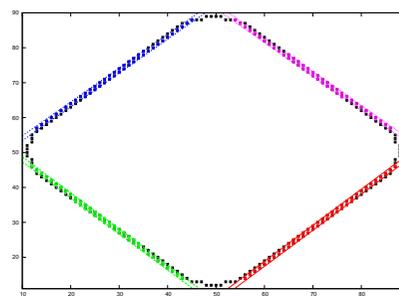


図 15 RANSAC による離散直線認識  
Fig. 15 Line recognition by RANSAC

- 2) Köster, K. and Spann, M.: MIR: An approach to robust clustering — application to range image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.22, No.5, pp.430–444 (2000).
- 3) Hartley, R. and Zisserman, A.: *Multiple view geometry in computer vision*, Cambridge University Press (2003).
- 4) Forsyth, D.A. and Ponce, J.: コンピュータビジョン, 共立出版 (2007). 大北 剛 訳.

- 5) 金谷健一: 空間データの数理, 朝倉書店 (1995).
- 6) Huber, P.J.: Finite sample breakdown of  $M$ - and  $P$ -estimators, *The Annals of Statistics*, Vol.12, No.1, pp.119–126 (1984).
- 7) Duda, R.O. and Hart, P.E.: Use of the Hough transformation to detect lines and curves in pictures, *Communications of the ACM*, Vol.15, No.1, pp.11–15 (1972).
- 8) Fischler, M.A. and Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM*, Vol.24, No.6, pp.381–395 (1981).
- 9) Rousseeuw, P.: Least median of squares regression, *Journal of the American statistical association*, Vol.79, No.388, pp.871–880 (1984).
- 10) Coeurjolly, D. and Brimkov, V.: Computational aspects of digital plane and hyperplane recognition, *Lecture Notes in Computer Science*, Vol.4040/2006, pp.291–306 (2006).
- 11) Stojmenović, I. and Tosić, R.: Digitization schemes and the recognition of digital straight lines, hyperplanes and flats in arbitrary dimensions, *Contemporary Mathematics*, Vol.119, pp.197–212 (1991).
- 12) Gerard, Y., Debled-Rennesson, I. and Zimmermann, P.: An elementary digital plane recognition algorithm, *Discrete Applied Mathematics and Combinatorial Operations Research and Computer Science*, Vol.151, pp.169–183 (2005).
- 13) Gérard, Y.: A fast and elementary algorithm for digital plane recognition, *Electronic Notes in Discrete Mathematics*, Vol.12, pp.142–153 (2003).
- 14) Debled-Rennesson, I., Rémy, J. and Rouyer-Degli, J.: Linear segmentation of discrete curves into blurred segments, *Discrete Applied Mathematics*, Vol.151, pp.122–137 (2005).
- 15) Provot, L., Buzer, L. and Debled-Rennesson, I.: Recognition of blurred pieces of discrete planes, *Lecture Notes in Computer Science*, Vol.4245/2006, pp.65–76 (2006).
- 16) Klette, R. and Rosenfeld, A.: *Digital Geometry: Geometric Methods for Digital Picture Analysis*, Morgan Kaufmann (2004).
- 17) Zrour, R., Kenmochi, Y., Talbot, H., Shimizu, I. and Sugimoto, A.: Combinatorial optimization for robust digital line and plane detection, *WCVM of PSIVT09* (2009).
- 18) 宮代隆平, 松井知己: ここまで解ける整数計画, システム/制御/情報, Vol.50, No.9, pp.363–368 (2006).