

# ステレオ画像からの3次元立体復元のためのノイズ除去\*

古川 進<sup>\*1</sup> 磯部 俊夫<sup>\*2</sup> 本間 幸造<sup>\*2</sup>

## A Method for Eliminating Noise from Multiple Images for Reconstructing 3-D Objects

Susumu FURUKAWA<sup>\*1</sup>, Toshio ISOBE<sup>\*2</sup>  
and Kohzo HOMMA<sup>\*2</sup>

### ABSTRACT

Reconstructing 3-D objects from image data is one of the most important problems in enabling a robot to recognize the environment and work independently in a rocket. As the basis for reconstructing 3-D objects, a new method for identifying edges in images is presented. This method consists of the following three subprocedures: (1) Differences of brightness of two subsequent pixels are calculated and binary images are made, (2) Noise included in those image data is roughly eliminated by utilizing the properties of an edge in such images, and (3) Noise not recognized in the previous subprocedure is eliminated by the relation between 3-D objects in 3-D space and images, by using multi image data obtained by a camera moving on a rail. An experimental camera system has been constructed in order to confirm the theoretical consideration, and various examples have been tested. The results show that almost all noise can be recognized and eliminated.

**Keywords:** Computer Vision, Image Processing, Eliminating Noise, Reconstruction of 3-D objects, Stereo-Vision

### 概 要

無人環境の中で働くロボットが周囲の環境を認識するための方法論を確立するための第一歩として、本論文においては、複数の画像データより、3次元立体を復元するための基礎的事項についての検討を行う。具体的には、移動可能なカメラによって得られた複数個の画像情報より、次の手続きによって物体の稜線の検出を目的とするノイズの除去を行う。手続きは(1)差分法によって各画像の2値画像を求め、(2)稜線の持っている性質を利用して2値画像中に含まれるノイズを大まかに取り除く、(3)3次元空間に存在する物体と画像データの関係を利用して、残りのノイズを消去する、というものである。本論文で述べられた方法の正当性を確認するため実際に実験装置を試作し、ノイズの除去を試みたところ、良好な結果が得られた。

### 1. 緒 言

宇宙ロケットなどの無人環境の中で働くロボットには、環境を認識して、自分の行うべき行動を自分で決定しなければならないような作業が非常に多い。目標物の認識を始めとして、必要な機器類の操作、工具の取り付け、取り外しなど、視覚情報処理が必要とされる作業全般にわたっ

ている。

しかし、ロボットにとって、環境を認識し、自律的に動作することは極めて困難な作業であり、多くの研究成果が報告されているにもかかわらず、現在に至るまで、環境認識の方法論は確立されていないといっても過言ではないであろう。ロボットが環境を認識し、かつ必要な作業を行うためには、(1)ロボットの目に当たるカメラを通して、環

\* 平成10年1月29日受付 (received 29 January 1998)

\*1 山梨大学 (Yamanashi University)

\*2 数理解析部 (Computational Sciences Division)

境情報を画像データとして取り込み、(2)何らかの処理を行ってロボットの必要としている対象物を認識し、さらに、その認識対象をその周囲に広げて行き、(3)全体として現在ロボット自身が存在する空間の位置と、自分の行うべき作業を決定する、という手順が必要とされると考えられる。

上記の(2)の作業は従来より画像処理の中心的問題であり、画像情報より物体のエッジを検出する方法<sup>1)・6)</sup>を始めとして、画像から必要とされる立体情報の抽出や3次元立体の復元<sup>7)・10)</sup>に関する問題などについて多くの研究が行われてきた。

画像情報より物体のエッジを検出する方法はさらに(1)画像情報を何らかの方法で2値化する、(2)本来エッジでないもの(以下ではノイズと呼ぶことにする)がエッジと認識されて処理された場合に、ノイズを効果的に消去する、(3)本来エッジであったものが、処理の途中で失われてしまった場合の復元処理、などについての方法論を確立しておかなければならない。しかし、現在においては、2値化処理の際に発生するノイズをどのようにして、消去するか、という問題に限っても、決定的な方法が見いだされておらず、目的とする課題ごとにその制約条件の範囲内で、すべてのエッジが検出されなくても、我慢できる程度のところでやめてしまう、という現実的なやり方をとらざるを得ないような状況にある。

著者等は、複数のカメラを用いて環境情報をイメージデータとしてコンピュータ内に取り込んだ後に、3次元立

体を復元し、さらにあらかじめコンピュータ内に記憶させてある立体の幾何モデルと比較、照合を行い、類似度の判定を行う<sup>11)</sup>ことにより、ロボットが“見ている”対象物が何であり、かつ自分自身はその対象物に対して相対的にどの位置にいるのかを認識できるような方法論を確立することを目指して検討を始めた。

上述の目標を達成するための第一歩として、一台のカメラを直線のガイド上に設置し、カメラがこの直線上を定速度で移動しながら、定期的に画像情報をコンピュータ内に取り込んだとき、これらの複数の画像データを利用して3次元立体の復元が可能かどうかの検討を行った。具体的には(1)3次元空間内に存在する物体を、カメラを通して画像としてコンピュータに取り込む機構および画像データから3次元の点の位置を復元する方法について理論的に考察する、(2)実際にカメラの移動が可能な機構を試作し、画像データをコンピュータに取り込む、(3)理論的考察と一致する結果が得られるかどうかを確認する、という手順で考察を行った。

その結果、3次元稜線の復元を目的としたノイズの除去がある程度正確に行い得ることが分かったので報告する。

## 2. 画像情報よりのエッジの抽出

画像情報より物体のエッジを検出する方法について考察する。第一章で述べたように、画像中に存在するエッジを検出する方法はすでに種々提案されているが、最も単純な方法の一つに、差分法として知られているものがある<sup>3)</sup>。

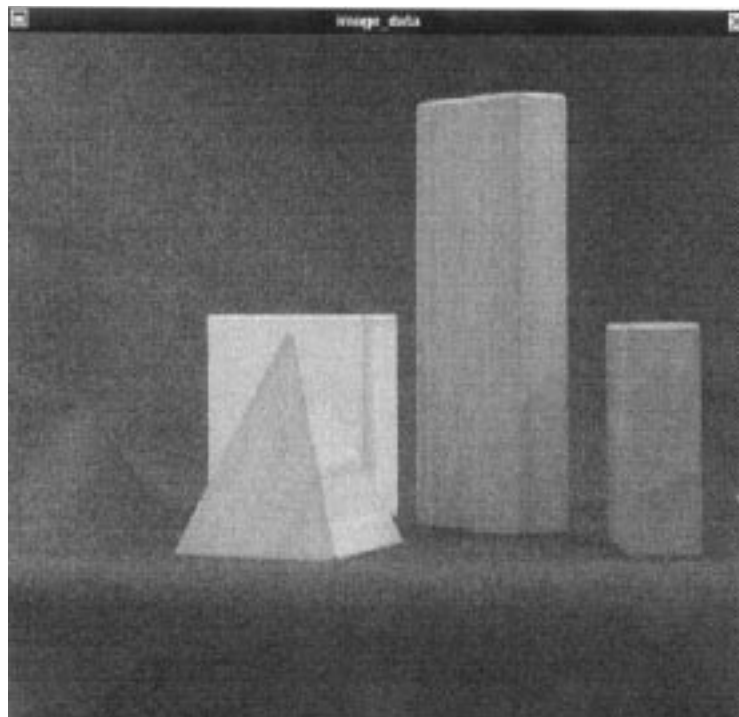


図1 画像データの例

差分法は、エッジは明るさの急激に変化している場所である、という単純な考え方に基づいたものであり、各ピクセルに対して隣のピクセルとの明度差を求め、この差が適当に定めたしきい値よりも大きいとき、この部分にエッジがある、と考える方法である。

図1に示されるような画像を例として、差分法によってエッジの抽出を行ってみた。画像は、大きさが512×480、1ピクセルの明度データが0から255までのものである。図2～4はしきい値 $t$ がそれぞれ64,32,16の場合を表している。これらの図より、しきい値が大きいほどノイズの数が少なく、小さいほどノイズが多くなっていることがわかる。逆に、しきい値が大きいほど本来エッジである筈の部分もエッジと判定されず、しきい値が小さいほどエッジ

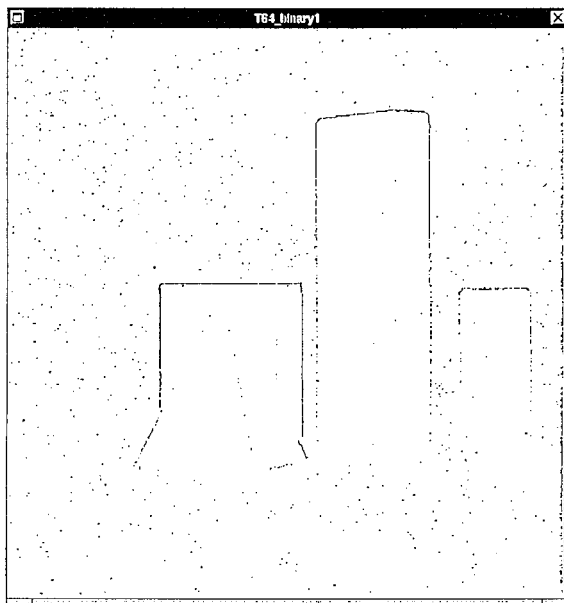


図2 2値化画像の例(しきい値64の場合)

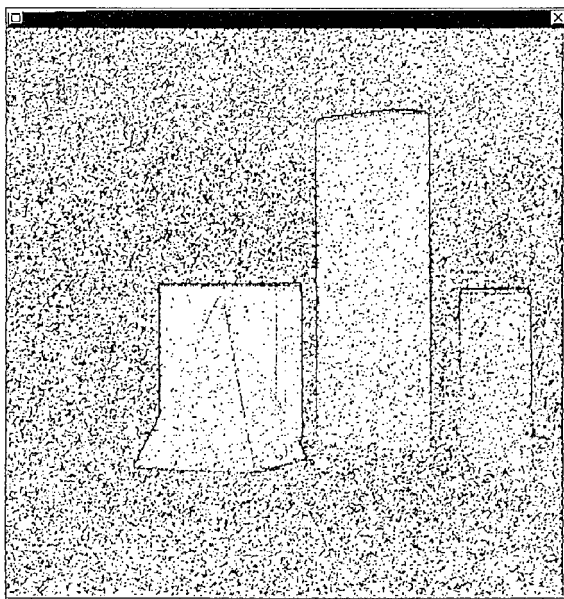


図3 2値化画像の例(しきい値32の場合)

も正確に認識できる、ということがわかる。つまり、エッジを正しく認識するためには、しきい値を小さく設定する必要があり、ノイズを除去するためには、しきい値を大きくする必要がある、ということである。

差分法では上記のしきい値をどの程度に設定すれば良いか、ということが最大の課題である。

しきい値の定め方は画像処理が必要とされる各アプリケーションごとに異なり、必要に応じて柔軟に定める、というのが現在採用されている一般的な考え方のようなのである。しかし、宇宙空間や無人の宇宙船ロケット内で作業を行うべきロボットの場合には、しきい値自体もロボット自身で定めることが要求される場合が多い。

次節以後においては、しきい値を定めることなくエッジを抽出する方法と共に、ノイズ除去の新しい方法について考察する。

### 3. 稜線の持つ性質を利用した稜線の抽出について

前節の図2～4からも明らかなように、差分法によって得られた2値画像は立体の稜線を表すピクセルと共に多くのノイズを含んでいる。ノイズ除去の第一歩として、本節においては稜線の持つ性質について考察しておくことにする。

画像平面上において、ピクセル $p(i, j)$ が稜線上の一点にあると仮定する。図5に示されるように $p(i, j)$ が稜線上の点であるのであれば、稜線の連続性の性質から、ピクセル $(i, j)$ をその中心に含む $3 \times 3$ ピクセルの領域内に点 $p(i, j)$ の他に少なくとも2点は稜線上の点があるはずである。近くに他の稜線が存在しない場合には、領域の大きさを拡大しても同様な性質が成り立つ。以下では、点 $(i, j)$ をその中心に含む $n \times n$ ( $n = 3, 5, 7$ )ピクセルの領域 $S_n$ 内における差分値の統計的な性質について考えてみる。

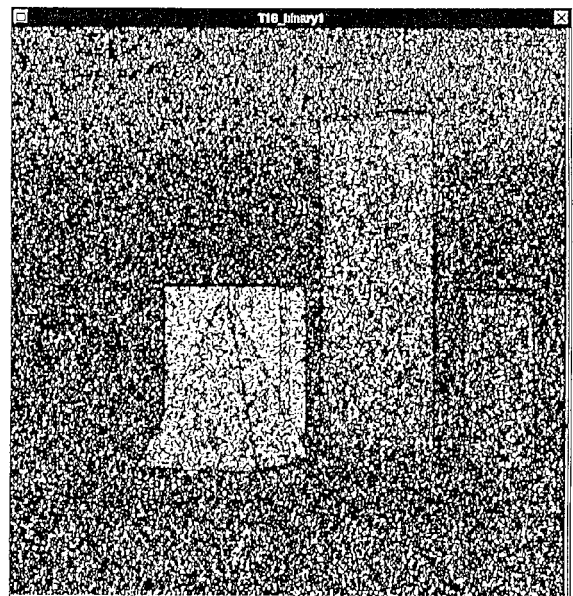


図4 2値化画像の例(しきい値16の場合)

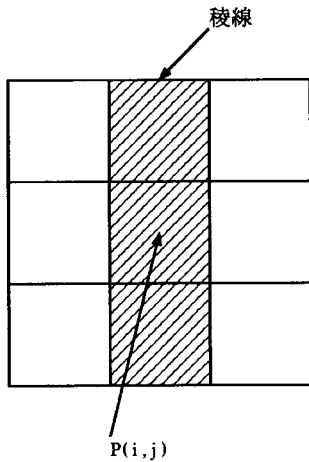


図5 画像平面における稜線

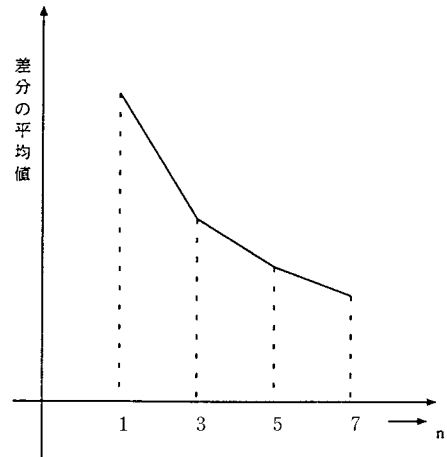


図6 エッジを含む小領域における差分の平均値

稜線を表すピクセルの差分値の平均を  $h_1$ 、稜線以外の差分値の平均値を  $h_n$  とすると、 $n \times n$  ピクセルの領域内で稜線が占めるピクセル数は  $n$  であるから領域  $S_n$  における差分値の平均値  $h_n$  は

$$h_n = \frac{h_1 + (n-1)h_n}{n} \quad (1)$$

で表される。

一般に  $h_1 > h_n$  なる関係が成り立っている筈であるから、式(1)の  $h_n$  の値は図6に示されるように  $n$  が大きくなるのに従って小さくなる。この性質を利用すると、しきい値によらずに、あるピクセルが稜線の上にあるかどうかを判定できる。すなわち、あるピクセルが稜線が否かの判定において、まずそのピクセルにおける差分値の絶対値  $h_1$  (以下差分値の絶対値を単に差分値と記す) を求める。次に対象ピクセルを中心とする  $3 \times 3$  の領域での差分値の平均値  $h_3$  を求める。以下同様に  $h_5, h_7, \dots$  と求める。このようにして求めた  $h_1, h_3, h_5, \dots$  の値が図6に示したように次第に小さくなっているとき、対象とするピクセルを稜線上にあると判断する。以下においては、差分画像の各ピクセルが図6のような性質を持っているもののみを稜線となり得るピクセルとして残し、他はノイズとして消去することにする。たとえば、垂直な稜線上のあるピクセルを中心とした  $7 \times 7$  のピクセルにおいて、その  $x$  方向の差分値は、次のようになっている。

6	7	39	152	3	3	4
16	2	29	122	29	2	15
15	13	15	142	1	14	5
4	16	19	111	29	24	30
7	8	1	128	18	29	7
0	1	18	120	38	45	39
8	15	9	106	20	4	4

中心のピクセルの差分値は、111 である。すなわち、 $h_1$  が 111 ということになる。次にこのピクセルを中心とした  $3 \times 3$  のピクセルの平均値を求めると  $51.5555 (= h_3)$  である。同様に、 $h_5, h_7$  を求める。その結果は、次のようになる。

$$\begin{aligned} h_1 &= 111.00000 \\ h_3 &= 51.55555 \\ h_5 &= 38.96000 \\ h_7 &= 30.44898 \end{aligned}$$

これらの値は、図6に示した曲線の傾向と同様な傾向をもっているといふことができる。本方法では、計算を単純化するため、あるピクセルにおいて、その  $h_1, h_3, h_5, h_7$  の値が、

$$h_1 > h_3 > h_5 > h_7$$

の関係を満たしているとき、そのピクセルは稜線の上にあると判定する。この判定条件に従えば、今対象としているピクセルの右隣のピクセルでは、

$$\begin{aligned} h_1 &= 29.00000 \\ h_3 &= 55.11111 \end{aligned}$$

となり、稜線上にないと判断される。

当然、本方法においても、

- ・稜線上のピクセルが稜線上にない
- ・稜線上にないピクセルが稜線上にある

と誤判断することは、避けられない。稜線上にないピクセルが稜線上にあると判断されたピクセルの周囲  $7 \times 7$  ピクセルの差分値とその  $h_1, h_3, h_5, h_7$  の値の例を次に示す。

14	35	45	18	14	16	13
16	16	15	12	39	41	18
51	18	20	26	19	10	9
16	9	7	29	24	3	6
2	5	12	17	4	2	4
17	5	26	35	2	7	4
4	11	7	1	8	20	33

$h_1 = 29.000000$   
 $h_3 = 17.555556$   
 $h_5 = 16.120000$   
 $h_7 = 16.020408$

上述の方法を適用してみた結果を図7に示す。この図7は、x方向及びy方向の差分値に対して、上述の方法を適用した2つの結果を合成したものである。図4に示される差分画像(しきい値16)に対して、稜線を表すピクセルにはほとんど変化がみられないが、ノイズが大幅に減少していることがわかる。また、この2値画像を得るのに、しきい値を必要としない。これが本方法の最大の利点でもある。このことは、本方法が広い範囲の画像に適用できる可能性を示している。

4. 空間の点の位置の推定とノイズの認識

図8に示されるような水平方向に置かれた2台のカメラによって、環境の情報をステレオ画像としてコンピュータ内に取り込む機構を考える。空間に置かれた物体やカメラ、画像の位置関係を表すために、以下のような3次元左手座標系を設定する。図8において、カメラの右方向をx軸、カメラの高さ方向をy軸、カメラの前方をz軸とする。カメラのフィルム面(画像平面と呼ぶ)にも同一の座標系、すなわち、空間座標系のx、y平面をz軸方向に-fだけ平行移動したものを使用することにする。ここで、fはカメラのレンズからフィルム面までの距離を表す。

カメラが原点を中心としてx軸上にそれぞれ(-d/2, 0), (d/2, 0)の位置に設置されているとすると、空間の点p(x, y, z)の画像平面における位置はそれぞれ

$$x^*_l = -(d/2 + x)(f/z) - (d/2) \tag{2}$$

$$y^*_l = y(f/z)$$

$$x^*_r = (d/2 - x)(f/z) + (d/2) \tag{3}$$

$$y^*_r = y(f/z)$$

で表される。ここで、添え字\*は画像平面上の像であることを意味し、l, rはそれぞれ左、右のカメラによるものであることを意味するものとする。

式(2) (3)から明らかなように、画像平面においては、点pの像は図9に示されるように、左右のフィルム面の同じ高さをもつ水平線上に撮影される。図9のハッチングを施した部分(それぞれi番目、j番目のピクセルとする)が点pの像が射影されたピクセルである。左右のフィルム面上に得られたこれらのピクセル(i, j)を以下においてはピクセルの組と呼び、記号I(i, j)で表すことにする。

図8に示されるカメラ機構によって、点pの位置を逆算すると、線L<sub>l</sub>とL<sub>r</sub>の交点がある位置であることがわかる。ところで、図9に示される各ピクセルはわずかではあるが、大きさを持っている。それ故、実際には点pの位置は

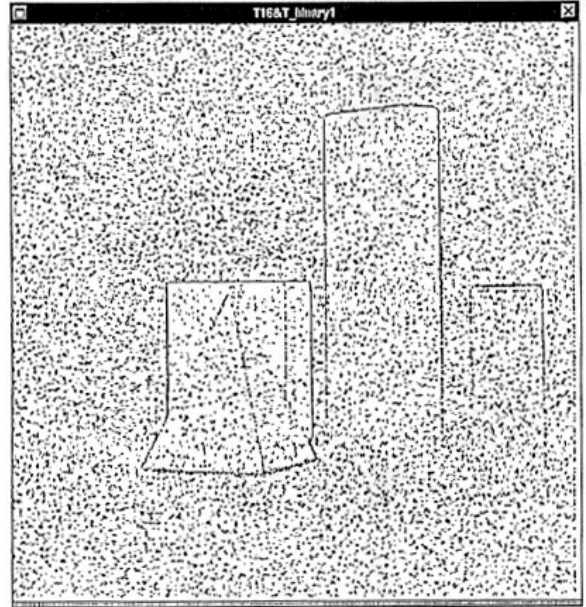


図7 稜線画像の性質を利用した2値化画像の例

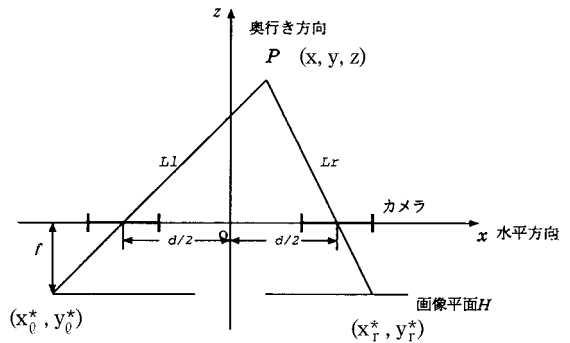


図8 カメラ機構の概略図

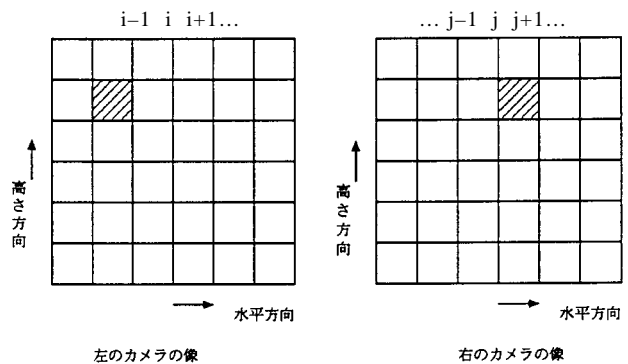


図9 空間の点Pのフィルム面における画像 (フィルム面では同じ高さに射影される)

各ピクセルの左右の境界と各カメラのレンズを結んだ線分によって定義される(図10の斜線部)領域の内部にある、ということが分かるだけである。以下においては、この領域をピクセルの組I(i, j)によって定められるという意味でR(i, j)で表すことにする。

左右のカメラをたとえば、右側に適当な距離だけ移

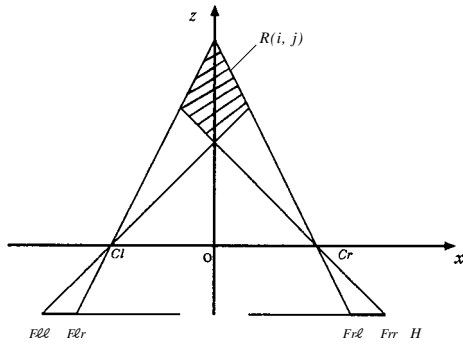


図 10 同一ピクセルの組に射影される点の領域

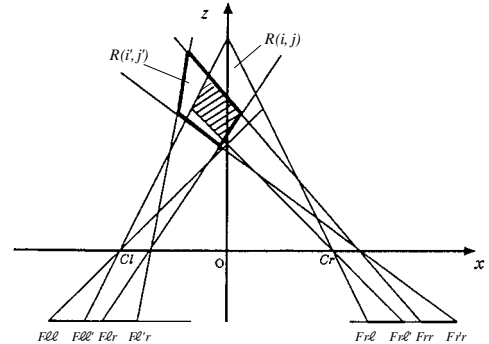


図 11 カメラの移動による点の存在領域の変化

動した場合について考察する。このとき得られる像のフィルム面での位置より、前と同様にして点 p の存在領域を算出すると、図 11 の太線で示される領域  $R(i, j)$  が得られる。カメラの移動前に得られた領域  $R(i, j)$  と移動後に得られた領域の共通領域(図の斜線部)が点 p の存在する可能性のある範囲を表していることは明らかであろう。

さらにカメラの位置を順次たとえば右側に移動しながら、点 p の存在領域を求めて行くと、p は得られた領域の共通部分の内部に存在する。カメラの移動量が連続量であれば、各領域の共通部分はただ一点に収束する筈であり、したがって点 p の位置は正確に求められる。また、カメラが左右どちらの方向にも移動可能である場合には、たとえば 2 分法のような手法を用いて、p の位置を正確に求めることも可能である。カメラの位置を少しずつ移動していったときの領域の変化の様子を図 12 に示す。

カメラの移動に伴って求められた複数個の領域に共通部分がないときには、空間に存在する点の位置を求めることができない。つまり、フィルム面上に現れたピクセルの組は現実には存在しない点の像を表しているか、あるいはこのピクセルの組から空間の点の位置を求めようとするのは適当でない、ということを表している。ピクセルの組  $I(i, j)$  が現実には存在しない点の像を表しているときには、これらは何らかの原因でフィルム面上に現れた雑音(ノイズ)と見なすことにする。

5. 対応点の決定とノイズの除去

図 13 に示されるように、左右のフィルム面の同じ高さの線上に、複数個の像が得られている場合について考察する。なお、以下においては、2 台のカメラによって撮影された一組の画像のそれぞれに対して、4 章で提案した稜線の抽出法に基づいて抽出した稜線により 2 値化した画像を対象とするものとする。したがって、点の像というのは、稜線と認識したピクセルを指すことにする、ということである。

左のフィルム面には  $n_l$  個、右のフィルム面には  $n_r$  個の

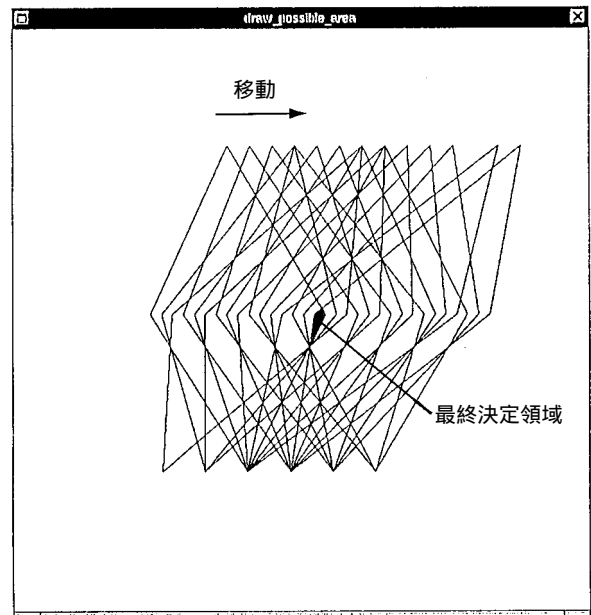


図 12 カメラの移動に伴う点の存在領域の変化

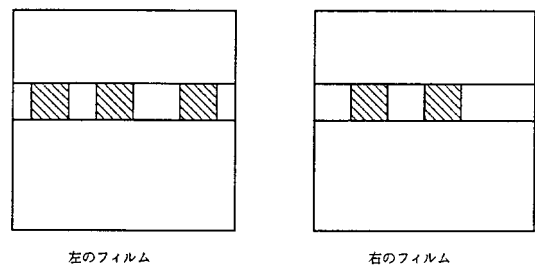


図 13 同一水平線に複数の稜線候補が得られた場合

ピクセル画像が得られている場合について考察する。ピクセルの組  $I(i, j)$  が実際に存在する点の像を表しているか否かは、このピクセルの組より得られる領域  $R(i, j)$  とカメラを少しだけ移動したときに得られる領域  $R(k, l) (k = 1, 2, \dots, n_l; l = 1, 2, \dots, n_r)$  のいずれかと共通部分があるか否かを調べることによって確認できる。いずれの領域  $R(k, l)$  とも共通部分を持たないとき、ピクセルの組  $I(i, j)$  に対応する点は存在しない、と結論づけられる。

移動前の左右のフィルム面上に射影されたピクセルの数

をそれぞれ  $m, n$ , 移動後のそれを  $m', n'$  とすると同一の高さにあるピクセルについてノイズであるか否かを決定するためには,  $m n m' n'$  個の組み合わせについて上述のテストを行う必要がある。コンピュータの演算速度が飛躍的に向上した現在においても, これは非常に時間のかかる作業である。次節においては, この組み合わせの数を減少させる方法について考察する。

## 6. 注目点の設定について

ロボットが注目している点がカメラより距離  $Z_f$  だけ離れたところにあると仮定する。ロボットが注目点付近にあるもののみを認識するだけで良いのであれば, 前節で述べられた方法を適用する場合に, 調査対象となるピクセルの組み合わせの数を大幅に減少させることができる。ロボットの認識すべき対象の物体が図 14 に示されるようにカメラからの距離が  $Z_{fmin}$  から  $Z_{fmax}$  の間にある場合について考察する。ピクセルの組  $I(i, j)$  によって定められる領域  $R(i, j)$  が注目点の付近にない場合, すなわち, 領域  $R(i, j)$  の最遠点が  $Z_{fmin}$  より小さいか最近点が  $Z_{fmax}$  より大きいときには, このピクセルの組は調査対象から除外できるからである。

領域  $R(i, j)$  が  $Z_{fmin}$  よりもカメラに近い位置にある場合には, 右(左)の画像の  $j+1$  ( $j-1$ ) 番目以降(以前)のピクセル, 同様に領域  $R(i, j)$  が  $Z_{fmax}$  よりカメラから遠い位置にある場合には, 左の画像の ( $j-1$ ) 番目以前のピクセルは, 左の  $i$  番目のピクセルとの組み合わせの可否の検証を行う必要がない, といえる。

## 7. 実験例について

前節までに述べられた理論的考察の有効性を確認するため, 写真 1 に示されるような実験装置を試作し, 32 枚の画像の撮影を行った。カメラは写真からもわかるように台上に固定されたレール上を等速度で移動できるようになっている。撮影は 2mm 間隔で行い, 各画像には順序数 1 ~ 32 を割り当てた。第 4 章では 2 台のカメラから画像情報を取り込むことを前提として議論を行ったが, 本実験では便宜上  $i$  と  $i+16$  ( $i = 1, 2, \dots, 16$ ) 番目の画像をそれぞれ一組のステレオ画像と見なして取り扱うことにして, 処理を行ってみた。注目点はカメラから 70cm の位置に設定し, 前後 15cm の幅に入る物体のみを認識の対象とすることとした。実際には画像 1 と 17, 2 と 18, ..., 8 と 24 の 8 組の画像情報を利用してノイズの除去を行った。得られた結果を図 15 に示す。図 15 より明らかなように, ノイズの大半が除去されている。同時に, 本来稜線であるべき部分もノイズと認識されてしまった箇所も認められる。

なお, 実験にはペンティアム 133MHz の CPU を内蔵したパーソナルコンピュータに UNIX を搭載したものを使用

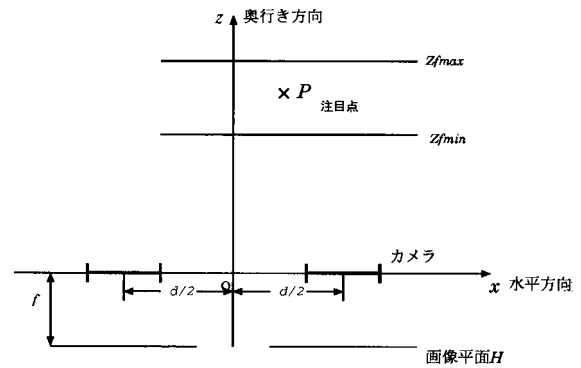


図 14 注目点と認識の範囲の設定



写真 1 試作撮影装置

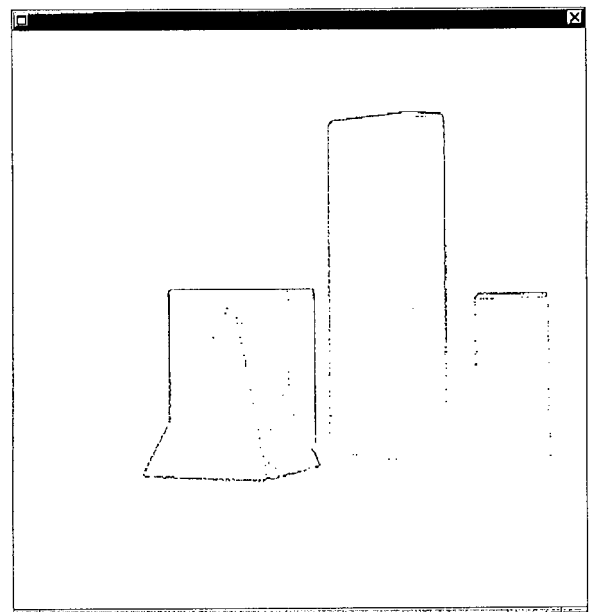


図 15 ノイズ除去後の 2 値画像

した。

このパーソナルコンピュータを使用して, 8 組のステレオ画像からノイズの除去を行った画像(図 15)を計算するに要した時間は約 4 時間 30 分であった。

## 8 . 結 言

移動可能なカメラ機構を用いることを前提として、複数の画像情報より3次元立体の復元を行うための新しい方法について考察を行った。本手法は最も簡単な差分法より出発して、エッジの持つ統計的性質を利用して稜線を抽出し、カメラの移動に伴って雑音を大幅な消去を行った。その結果、本手法は雑音の除去も含めて、しきい値を用いずに稜線を抽出でき、3次元立体の復元手法として使用できる可能性のあることがわかった。しかし、本研究で得られた結果はまだ不十分なものであり、完全に立体を復元、認識するためにはさらに詳細に検討を行うことが必要である。

最後に、本研究の遂行に際して、プログラム作成などで助力された、山梨大学大学院生王朝明さんに感謝の意を表す。

## 参考文献

- 1) Nair H.; Reconstruction of Planar Boundaries from Incomplete Information, Computer Vision, Graphics, and Image Processing, vol. 39, pp. 383-387 (1987)
- 2) 身持圭一, 佐々野祐一, 中山博之; 三次元視覚センサの研究, 三菱重工技法, vol. 25, no. 3, pp. 241-246 (1988)
- 3) 高木幹雄, 下田陽久監修; 画像解析ハンドブック, 東京大学出版会, p. 553 (1991)
- 4) 月瀬寛二, 藤原良一, 壺井芳昭; ハフ変換の量子化誤差評価に基づく線分抽出法, 電子情報通信学会論文誌 D-II, vol. J78-D-II, no. 8, pp. 1176-1183 (1995)
- 5) 郭素梅, 小沢慎治; 微分画像に基づく画像の2値化法における最適しきい値の決定法, 電子情報通信学会論文誌 D-II, vol. J80-D-II, no. 1, pp. 183-189 (1997)
- 6) Stenstrom J.R. and Connolly C.I.; Constructing Object Models from Multiple Images, International Journal of Computer Vision, vol. 9, no. 3, pp. 185-212 (1992)
- 7) 安川祐介, 稲本康, 川上進; ハイパコラム法による濃淡画像からの線分抽出, 信学技法, NLC92-22, PRU92-36, pp. 9-16 (1992)
- 8) 佐藤剛, 山本強; 連続フレーム画像からのテクスチャ付き3次元モデルの推定, 情報処理学会第49回全国大会講演予稿集, pp. 2-397-2-398(1994)
- 9) Seales W.B. and Faugeras O.D.; Building Three-Dimensional Object Models from Image Sequences, Computer Vision and Image Processing, vol. 61, no. 3, pp. 308-324 (1995)
- 10) Kanazawa Y. and Kanatani K.; Reliability of 3-D Reconstruction by Stereo Vision, IEICE., Trans., Inf. & Syst., vol. E78-D, no. 10, pp. 1301-1306 (1995)
- 11) Mukai S., Furukawa S., Obi M. and Kimura F.; An Algorithm for Deciding Similarities of Convex Polyhedra, Computers & Graphics, vol. 18, no. 2, pp. 171-176 (1994)