

面積 5 階調による画像処理の研究

—文字の抽出—

白 崎 智 義

A Study of Picture Processing using 5 Areal Gradations

Chiyoshi SHIRASAKI

In this paper, the method to extract characters from the picture changed 64 gradations to 5 areal gradations is reported.

1. ま え が き

64階調の画像を面積濃度5階調に変換したノイズの多過ぎる画像の中から文字を抽出する研究をしていたところ、この程その可能性が見えてきたので、現在までに分かってきた研究成果の一部を報告する。

2. システム構成

本研究に使用したシステムは次の通りである。

パソコン演算装置	PC-9801VM II	(NEC)
カラー・ディスプレイ (モニターを兼用)	PC-TV451	(NEC)
C C Dビデオ・カメラ	CCDV8AF	(SONY)
画像取り込みボード	NBC-VIDEO-98	(NBC)
マウス		

画像取り込みボードをパソコン演算装置のスロットバスに装着する。テレビあるいはカメラからの映像をいったんこのボードに取り込んだあと、パソコンの主メモリに転送し、モニターにカラーでディスプレイする。そして、この映像を面積5階調に変換した上で画像処理が行える体制に入る。なお、PC-9801VM II のメモリーは、640KB に増設してはならない。

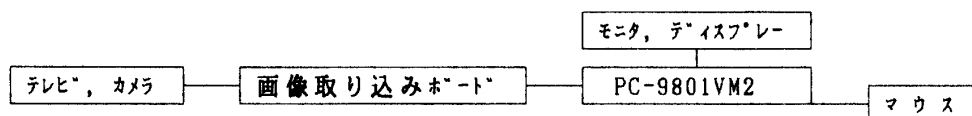


図 - 1 画像処理システム

3. 理論

面積5階調の画像は2*2ドットで1画素を表示している為に、単純に考えてドット単位での画像処理を行うことはよくない。その上、64階調濃度を5段階に区分する範囲の設定、並びに各階調値に対応するドット配置の仕方によって、画像ドットの配色状態が多少変化する。実際に画像の状態を観察するとき、不純なドット・ノイズの存在が目につき、これが画像のノイズとなって気にかかる。この様な状態のもとで、画像の中から文字としたの数字を取り出す画像処理の基本的な方法について述べる。

(1) 5階調の画素

64階調で構成される画像の画素を8色中・8色モードのディスプレイに表示させる為には、これを階調値を5とする画素に変換しておかねばならない。

それには、カラー・コードが、

1 (青)、2 (赤)、3 (紫)、4 (緑)、5 (水)、6 (黄)、7 (白)

であるところから、表-1のように、階調値とドット配列位置との関係表が得られる。


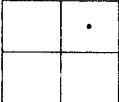
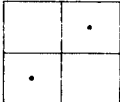
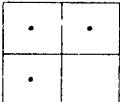
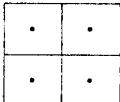
階調値	0	1	2	3	4
ドットの 配列位置					

表-1 5階調の区分表

ところで、各階調値に対応するドット配置の仕方は、同じ階調値であっても、表-2のように各種の配置法がある()内の数字は、表-3を参照)。従って、各階調値ごとにどれを選ぶかによって、画像の模様に変化があるものと考えられるが、今回の研究では*印の示すドット配置を選んだ。

1階調	(1)	*(2)	(3)	(4)		
2階調	(1,2)	(2,4)	(3,1)	(4,3)	*(2,3)	(1,4)
3階調	*(1,2,3)	(1,3,4)	(3,4,2)	(4,2,1)		
4階調	*(1,2,3,4)					

表-2 階調値に対するドット配置の種別

1	2
3	4

表-3 ドット位置番号

また、64階調濃度の範囲をどのように5段階区分するかによっても、画像の色合いが違って来るが、今回の研究では、表-4のように区分した。

区 分	64 階 調	5 階 調
濃 度 の 範 囲	0 ~ 20	0
	21 ~ 26	1
	27 ~ 34	2
	35 ~ 41	3
	42 ~ 63	4

表-4 5階調の濃度範囲

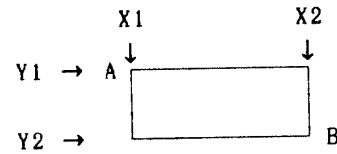


図-2 範囲の設定

(2) 領域の設定

画像処理の対象となる範囲、すなわち図-2で示す長形状の領域をマウスで設定し、A, B 2点の座標 $(X1, Y1)$, $(X2, Y2)$ を求めておく必要があるが、その方法は次の通りである。カラー画像は SCREEN 3, 0, 0, 1 命令による高階像度で表示されている為、この画像の上に直接重ねて境界線を引くことはできない。

この為の対策として、画面全体を青赤緑各色別に分解した画像をいったんフロッピーディスクに格納しておいて、SCREEN 2, 0, 0, 3 の命令でスクリーンモードを白黒に変更し、この画面上で自由に縦横の線を引くことにした。

この画面状態にすれば、縦のX1線, X2線並びに横のY1線, Y2線の4直線をマウスで任意に設定して、原画像を傷めることなく画像処理対象の範囲を決定することができる。すなわち、直線の引き直しの過程の中で、線を消すことがあっても画面まで消すことが無いので都合がよい。しかし、その反面、画面の地の色に変色しているから、領域確保終了後、これを元の正しい色に戻しておく必要がある。この対策として、画面モードをSCREEN 3の命令で、高解像度に戻しておいてから、フロッピーディスクに保存してある色別の3画像を取り出し、これを既に設定してある領域内に重ねることにした。

当然ながら、上記の過程の中から、2点A $(X1, Y1)$, B $(X2, Y2)$ の座標が求められる。この数値は拡大、縮小、2値化などの画像処理をする時に必要なデータである。

(3) 画像データの作成

テレビあるいはビデオ・カメラから、新たに画像を取り込む命令は次の通りである。

```

10 CLEAR &H2700          (マシン語領域を&H2700番地以降に設定      )
20 DEF SEG=&H2700         (セグメント=&H2700番地に設定            )
30 BLOAD "TR5N",&H0        (画像処理ソフト(TR5N、マシン語)をBLOAD  )
40 BLOAD "NTR5N5",&H700    (画像処理ソフト(NTR5N5、マシン語)をBLOAD)
50 POKE &H5FE,0           (画像取り込みメモリ番地のセグメントを
60 POKE &H5FF,&H60        &H&H6000番地に設定                      )
70 OUT &HDC,15            ※ (TV、ビデオの映像の取り込みを指示    )

```

```
80 DEF USR1=&H402  ※ (ユーザーのプログラム(マシ語)の所在番地を指示して、)
90 A=USR1(0)      (そのプログラムを実行する。)
```

これによって、デジタル型の1画面が、画像ボードのフレーム・メモリーに取り込まれると同時に、この画像データが、赤緑青の各色別に主メモリーの &H6000~&HA000 番地に格納される。そして、これを5階調に変化した形でディスプレイ画面上に画像が表示される。

なお、主メモリー内に既存の画像データを繰り返しディスプレイするには、上記命令の中にある※の部分の次のように変更すればよい。

```
70 OUT &HDC,11
80 DEF USR1=&H408
```

(4) 画像データの保存

画像データには、主メモリーの &H6000~&HA000 番地に格納されている64階調のデータと、赤緑青各色別のプレーンで構成されるVRAM領域の中にある画面に表示しているデータとがある。画像データをフロッピー・ディスクに保存するのに BSAVE命令を使う方法とファイル形式で行う方法の2種類を考え、処理時間、用途等を考慮して使い分けた。

7. 主メモリーの &H6000~&HA000 番地にある画像データの保存は BSAVE命令による方法をとった。この番地の領域は、4セグメント即ち 64KB の4倍の広さを持つている為、識別子を使って4個のファイルに分割する必要がある。逆に、フロッピー・ディスクに保存されている画像データ・ファイルを BLOAD命令によって取り出し、主メモリー内に記憶することができる。なお、この画像データは各画素毎に赤緑青別に分離して格納されている為、これを加工して面積5階調あるいは8階調の画像を作ることができる重要な基礎データである。

4. 画面上に表示されている画像は色々に処理される対象となるものであるが、その処理結果の画像を保存するのに BSAVE命令を使う方法とファイル形式の2方式を使い分けた。

VRAM領域は赤緑青各色別のプレーンの重ね合わせで構成されている為、BSAVE 命令でフロッピーディスクに格納する場合は、3個のプレーン別ファイルに分割する必要がある。この場合は、当然ながらBLOAD命令でなければ、フロッピーディスク内の画像データをVRAM上に戻すことができない。

ファイル形式は、画面上のX,Y座標で制限された領域内の画像を一つのデータ・ファイルとして保存する場合に適している。この時のプログラムの骨子は次の通りである。なお、X,Y表示の諸変数は図-2に基づいている。

```
10 OPEN "2:"+FI$ FOR OUTPUT AS #1 (フロッピー・ディスクにデータを書き込むOPEN宣言)
20 PRINT #1,X12,Y12 (X辺の長さ(X12),Y辺の長さ(Y12)を書き込む)
30 FOR J=0 TO Y12 (Y方向の長さを設定)
```

```

40 FOR I=0 TO X12                (X方向の長さを設定)
50 PRINT #1,P0                    (1ドットの濃度値をフロッピー・ディスクに出力)
60 EXIT I,J                      (次の濃度の繰り返し出力を指示)
70 CLOSE                          (ファイル・データの書き込み終了の CLOSE宣言)

```

尚、フロッピー・ディスクから画像データをVRAM上に取り出す命令の骨子は次の通りである。

```

10 OPEN "2:"+FI$ FOR INPUT AS #1 (フロッピー・ディスクからデータを読み込むOPEN宣言)
20 INPUT #1,X12,Y12              (X12,Y12の長さを入力)
30 FOR J=0 TO Y12                (Y方向の長さを指示)
40 FOR I=0 TO X12                (X方向の長さを指示)
50 INPUT #1,P0                  (1ドットの濃度値をフロッピー・ディスクから入力)
60 PSET (X2a+I,Y2a+I),P0        (画面上の位置を指定して、濃度をセットする)
70 EXIT I,J                     (次の濃度の繰り返し入力を指示)
80 CLOSE                        (ファイル・データの入力終了の CLOSE宣言)

```

(5) 画像の位置移動

例えば、図-3においてA図の画面をBの位置に移動したい時は、A図で示す領域内の画素すべての濃度値を POINT命令で計測し、その都度得た数値を直接B図内の対応する位置にPSETする。なお、この場合は2*2の画素を意識しなくてもよい。プログラムは次の通りである。

X1	X2	X1a	X2a	10	X12=X2-X1
↓	↓	↓	↓	20	Y12=Y2-Y1
Y1→				30	FOR J=0 TO Y12
				40	FOR I=0 TO X12
				50	P0=POINT (X1+I,Y1+I)
Y2→				60	PSET (X1a+I,Y1a+J),P0
				70	NEXT I,J

図-3 座標の説明

この他の方法として、A領域内の画像データをファイル形式でいったんフロッピーディスク内に格納しておき、ついで、移動先の新しい位置を決めておいてから、フロッピーディスク内の画像データを一つ一つ取り出して書き込んでもよい。

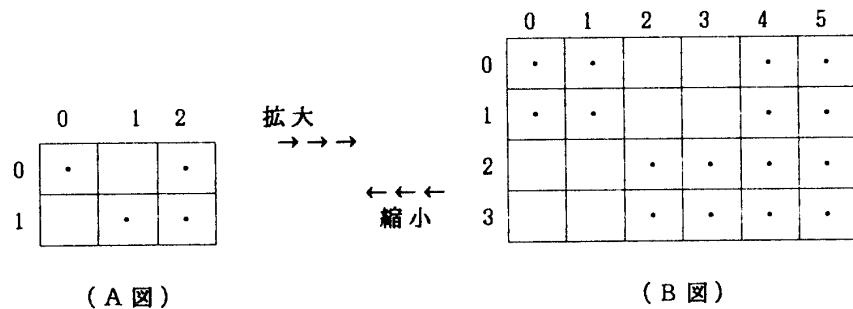
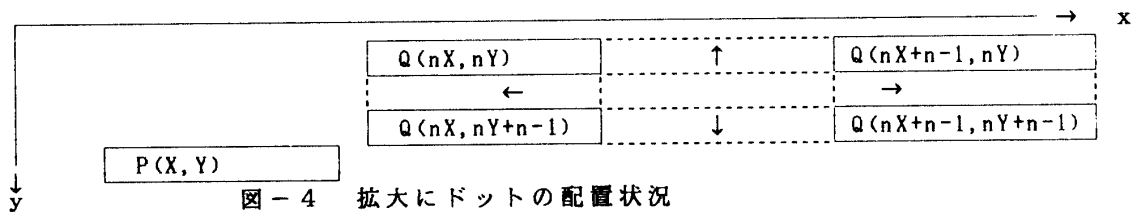
(6) 拡大

7. 画面上の画像を正数倍に拡大処理する方法は、次の考え方をする。

(7) $p(x,y)$ が1ドットの場合について考える。

画像を n 倍に拡大すると、1つの画素 $P(X, Y)$ に対する新たな画素 $Q(X', Y')$ が、縦横ともに n 倍されて、 $(X'=2X \sim 2X+n-1, Y'=2Y \sim 2Y+n-1)$ の座標位置に表示される (図-4)。勿論、実際の表示位置は、原画像を破壊することのない適当な場所を探しておいてから、指定領域内の図の拡大図を移動すればよい。

図-5は、例としてA図を縦横共に2倍に拡大してB図を作る場合の拡大状況を表示したものである。



(イ) 2×2 の4ドットで構成されるものを画素単位として、画像を拡大させる場合は、 X, Y 座標の計数を2ドット飛びにすれば、基本的には上と同じ考え方になる。

なお、画像処理の対象となる画像としては、(イ)よりも、(ア)で処理したものが適していると考えられる。

イ. 画像を拡大するに際して原画像を破壊しないようにする為に、次の2通りの方法をとった。

(ア) 画面上の原画像の位置から離れた場所を決めておいてから、直接原画像を画素単位に拡大したものをPSETしていく。

(イ) 原画像の画素すべてをいったんフロッピーディスクに格納しておいてから、決められた場所へフロッピーディスクの中の画像を画素単位に取り出し拡大しながらPSETする。

(7) 縮小

画像を n 分の1に縮小するには、画像内の決められた領域の左上の座標原点を起点にして、縦横両方向ともに n ドット飛びの画素を抽出して、現画像を破壊することのない位置で圧縮してPSETする (図-5, B図参照)。なお、この場合、画素を一定間隔で飛び飛びに

抽出する為に、当然ながら、画素の欠落現象が生ずる欠点があるのはやむを得ない。

(8) ノイズの除去

5階調画像に発生する一種独特のノイズが画像処理に悪影響を与える為に、これを除去する方法は、画像の上から実際に画像データを取ったところで対処した方がよいのである。

今回の研究で取った方法は次のとおりである。なお、処理の対象となる数字は大多数赤色のドットで構成されている中で、黄色と黒色のドットが多少混在している。そして、数字の周囲には黄色と赤色のドットが可成り点在している状況である。

- ア. 画面全体に地の色として点在する黄色のドットをすべて白色に変更する。
- イ. 文字としての数字を形成する範囲内に、少し点在する黒点はすべて赤色に変更する。
- ウ. 2*2の方形状4ドットの中のどこかに赤のドットが1個ある場合は、これを白色に変更する。
- エ. 3*3マスクを使って、マスク内の9個の画素の濃度値の中央値を求め、これをマスクの中央に位置するドットの濃度値とする。
- オ. スキャニングは縦横両方向とも、ドット連続とするか、1画素置きとするかはその都度判断する。

等、色々の方法を混在させる。

なお、少々ドット・ノイズは、マウスの利用で除去することができるが、今回の研究ではマウス無しにどこまで除去できるかという研究の狙いもあったので止めておいた。

4. 実験結果

(図-6)は数字2の原図であるが、これを1ドット単位で2倍に拡大したのが(図-8)で、2*2ドット単位で拡大すると(図-9)である。拡大図から数字2は不純なドット・ノイズでみにくくなっていることがはっきりするが、このような画像であっても、(図-7)のようにすっきりした数字の画像に変更することができる。(図-10)は、数字3を3分の1に縮小したものであるが、この場合は、当然ドット・ノイズは減少している。



図-6 原図になる数字



図-7 抽出した数字

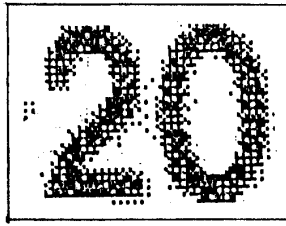


図-8 2倍拡大の数字 (1ドット)

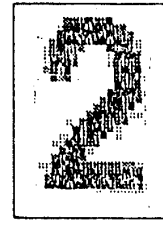


図-9 2倍拡大の数字 (2*2ドット)

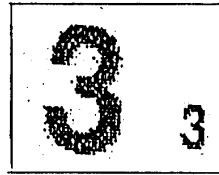


図-10 右は左の3を1/3に縮小したもの

5. まとめ

面積5階調表示の画像は、1画素が2*2の方形状の4ドットで構成されている為、これに影響されて生ずると考えられる不純なドット・ノイズが多数存在している。このノイズは文字の上、並びに文字の周囲を問わず至る所に散在している為、文字の抽出に先立ちドット・ノイズの除去処理をしなければならない。これに対処するソフトの作成上の一つの考え方を紹介し、実験によって一応の目安がついたことを示した。

参 考 文 献

- 1) 白崎智義：パレット番号を利用した画像輪郭の抽出法の研究, 福井工業大学研究紀要, 第20号(1990), 51~59
- 2) 山本良彦：卒業研究論文”画像処理の研究(1) - 濃度変換と幾何変換 - ”(1989)
- 3) 小柳信一：卒業研究論文”画像処理の研究(2) - 境界線検出および不良品摘出 - ”(1989)
- 4) NEC PC-9801VM2ユーザーズ・マニュアル P. 293
- 5) カラービデオ画像高速取り込みメモリーボード取扱説明書, 日本ハードコンピュータKK

謝 辞

本研究に協力してくれた卒研生山本良彦、小柳信一の両君に謝意を表します。

(平成2年12月20日 受理)