

道路画像の環境状況を示す自己組織化マップの作成

鶴谷 泰文¹、田中 雅博²

¹ 甲南大学大学院自然科学研究科

² 甲南大学理工学部・知的情報通信研究所

(受理日 2005年4月1日)

概要

Our objective of this paper is to recognize the weather and time conditions based on the image analysis. The basic tool is Self-Organizing Map (SOM) by Kohonen. SOM creates a map compressed from high-dimensional data. The topological feature of the data can be preserved fairly well. The items of the feature vectors will be organized by using the knowledge of humans. Thus we can extract the compressed information that seem necessary to make judgement of weather conditions. Global and local features will be defined. Also, the color information of RGB and YCbCr will be applied to extract the feature vector. The organized map will be shown to be useful for real data that are not used for organizing the map.

1 はじめに

自己組織化マップ[4, 7, 8, 10]は、高次元データの可視化の技術として、近年広く使われるようになってきた。これは、位相的情報の欠落をなるべく小さく抑えながら、高次元データを2次元あるいは1次元程度の低次元データに写像する技術である。特に2次元にした場合には、平面状に表示できるため、近年は2次元へのマップを行うことが多い。一方、近年、ライブカメラを用いた、自動監視システムが広く用いられるようになってきた。そのうちの多くは、ビデオテープやハードディスクへの単純な保存を行っているだけで、あとは人間の監視に基づいている。しかし、そのシステムの目的に合わせ、コンピュータによる自動認識技術も急速に脚光を浴びつつある[1, 2]。本論文では、道路上に固定されているカメラから、ほぼ1秒おきに画像を取得している交通監視システムにおいて、画像を自動認識する際に、道路上の気象条件も同時に認識しようとするためのものである。本システムで具体的に対象としているのは、英国 Nottingham におけるものであり、当地は、にわか雨が頻繁に降る、不安定な気象条件である。こういうところでは、気象台が記録するデータだけでは、実際に雨が降っていたか曇っていたか、画像を見る以外に信頼できる情報は少ない。また、道路での様子は気象条件に強く影響を受けるため、本研究でのテーマは重要である。我々が興味のある情報を取得するために、画像上の全体の色情報や明度情報、あるいは、局所的情報を、我々の持っている常識により、たとえば、雨が降っていればこういう部分がこのように変化するか、晴れていたら影ができるなど、常識を持っている。本研究での手法も、そういう人間の常識を取り入れながら、特徴ベクトルを定義し、それを使って自動化するという手順を踏むものである。

本論文の構成は、以下のとおりである。第2章では、自己組織化マップと、クラスタマップについて述べる。第3章では、画像の特性と特徴ベクトルの構成について述べ、第4章で応用実験を行う。第5章はまとめである。

2 自己組織化マップとクラスタマップ

2.1 自己組織化マップの概要

自己組織化マップ (Self-Organizing Map; SOM) とは、ヘルシンキ工科大学のコホネン (Teuvo Kohonen) により提唱された教師なし学習ニューラルネットワークである[10]。

2.2 構造

SOMは入力空間と出力空間の2つの空間から構成されている。入力空間は高次元の入力データ空間 X であり、出力空間は主に2次元のユニット (格子点) で構成される離散空間 Ω である。各ユニット $(i, j) (\in \Omega)$ には入力空間と同次元の重み m_{ij} がかけられており、写像された入力データに対応するユニット m_{ij} が、その入力データに類似するように学習されてゆくものである。

2.3 アルゴリズム

SOMの学習アルゴリズムを以下に示す[9, 10]。

1. 各ユニット (格子点) に入力データ (N 次元) と同次元の重み $m_{ij}(0)$ をランダムに発生させ、初期化を行う。
2. 以下の(a), (b)を全学習回数 t_{max} 、データ数 n_{all} について繰り返す。
 - (a) 入力されたベクトル $\mathbf{x}(n)$ とのユークリッド距離が最小となるユニットを学習回数 t の時の、データ $\mathbf{x}(n)$ に対する勝者ユニット (i^*, j^*) とする。

$$(i^*, j^*) = \arg \min_{(i, j) \in D(i, j)} \|\mathbf{m}_{ij}(t) - \mathbf{x}(n)\| \quad (1)$$

- (b) 2aで求められた勝者ユニット (i^*, j^*) とその近傍ユニットを(2)式により更新する。

$$\mathbf{m}_{ij}(t+1) = \begin{cases} \mathbf{m}_{ij}(t) + \alpha(t) \{\mathbf{x}(n) - \mathbf{m}_{ij}(t)\}, & (i, j) \in (i^*, j^*) + N_c(t) \text{ のとき} \\ \mathbf{m}_{ij}(t), & (i, j) \notin (i^*, j^*) + N_c(t) \text{ のとき} \end{cases} \quad (2)$$

$$\alpha(t) = \alpha(0) \left(1 - \frac{t}{t_{max}} \right) \quad (3)$$

$$N_c(t) = N_c(0) e^{-2N_c(0) \frac{t}{t_{max}}} \quad (4)$$

2.4 クラスタマップ

クラスタマップとは、SOM上にある格子点の N 次元の重み $m_{ij}((i,j) \in \Omega)$ を、その類似性においてまとめることで類似したデータを一括りにしたマップである。クラスタマップを作成するに当たって、クラスタリング（クラスタ化）を行う必要がある。クラスタリングは、データ集合を“類似した”データの部分集合に分割すること目的としており、実際に重みの類似性をまとめることである[10]。ここでクラスタリングのアルゴリズムを次節に示す[9]。

2.5 クラスタリングのアルゴリズム

1. 第2.3節のアルゴリズムを用いてSOMを求める。
2. 各格子点 $(i,j) \in \Omega$ の N 次元の重み m_{ij} から(5)式で求められる値 $d(i,j)$ を昇順に並べ替える。 $D(i,j)$ は (i,j) における第1近傍で、 (i,j) が端点のときは2または3点で、その他は4点である。

$$d(i,j) = \frac{1}{|D(i,j)|} \sum_{(\mu,v) \in D(i,j)} (m_{i,j} - m_{i-\mu,j-v})^T (m_{i,j} - m_{i-\mu,j-v}) \quad (5)$$

3. (5)式で求められた $d(i,j)$ 、 $(i,j) \in \Omega$ を昇順に並べ替え、 $q=1,2,\dots,|\Omega|$ と番号を付けていく。このとき、並べ替えたものを $s(i,j)=q$ 、そのときの格子点の座標を $(i,j)=s^{-1}(q)$ とする。
4. $q=1,2,\dots,\theta$ まで以下の(a), (b)を繰り返す。ここで θ は $\theta \leq |\Omega|$ とする。これは、データ集合間の境界上に存在するようなまばらな部分のデータを、あえて隣接するクラスタに含ませない方が、良い結果が得られると考えられるためである。
 - (a) $d(s^{-1}(q))$ がそのすべての第1近傍よりも小さければ $L:=L+1$ とし、点 $s^{-1}(q)$ のラベルを $l(s^{-1}(q))=L$ とする。また、第1近傍よりも大きい点であるときは、第1近傍の中の点で最小値を持つ点のラベルを $l(s^{-1}(q))$ に割り当てる。
 - (b) 1.で求めたSOMと $l(i,j)$ 、 $(i,j) \in \Omega$ を使って再びデータを写像させる。このとき、 (i,j) がどのデータに対応するのかをわかるようにしておく。

ここで、第1近傍とは点 (i,j) に隣接する $(i-1,j)$ 、 $(i+1,j)$ 、 $(i,j-1)$ 、 $(i,j+1)$ である（ (i,j) が端点の場合を除く）。

3 画像の特性と特徴ベクトルの構成

我々は、英国Nottingham市内に設置された、道路監視カメラから約1秒ごとに送られてくる画像を用いて、その画像がどのような天候・時間帯状況(昼間晴天、昼間曇天、昼間雨天、夕刻、夜間)を示しているのかを、画像の色、明るさ等から判断し、SOM上に写像することで類似する特徴を持った画像をまとめてみようとしている。画像は天候状況、時間帯について比較的判断しやすいものを、夜間（非雨天）、夕刻（非雨天）、昼間晴天、昼間曇天、昼間雨天、各100枚ずつ用いる。図1-5は、それぞれの環境における例である。



図1: 夜間画像

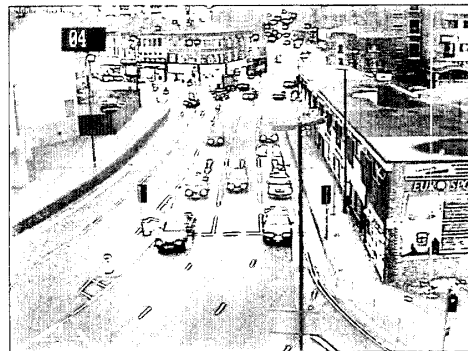


図2: 夕刻画像

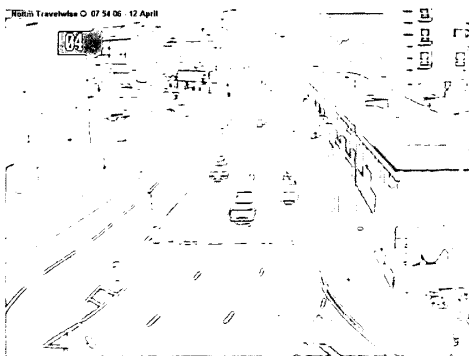


図3: 昼間晴天画像

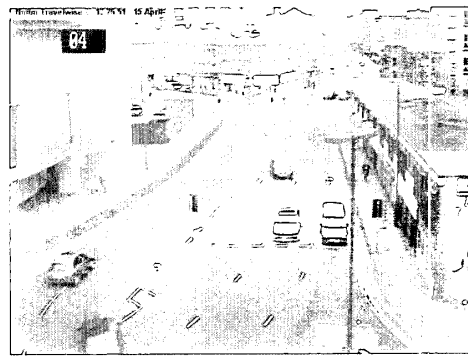


図4: 昼間曇天画像

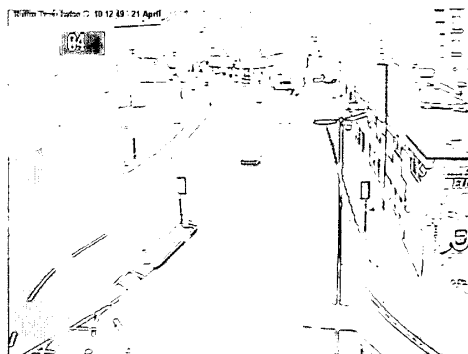


図5: 昼間雨天画像

表色系については、画像の性質上 YCbCr 表色系が有益である可能性があるが、RGB よりも有効であるかどうかを確認するため、クラス内変動とクラス間変動の比である、フィッシャーの線形判別法[11]を用いる。

3.1 RGB表色系からYCbCr表色系への変換

取り込む画像はサイズが $\Delta = \{(i,j) | 1 \leq i \leq 768, 1 \leq j \leq 576\}$ で、RGB3原色それぞれ256階調のデジタル画像である。画像から天候状況を読み取るためには、RGBそのままの値よりも輝度、色差から画像の‘白っぽさ’、‘明るさ’、‘暗さ’を読み取ることができるのではないかと考え、YCbCr表色系の利用を検討する。

RGBからYCbCrへの変換式は

$$\begin{aligned} Y &= 0.2990R + 0.5870G + 0.1140B \\ Cr &= 0.5000R - 0.4187G - 0.0813B + 128 \\ Cb &= -0.1687R - 0.3313G + 0.5000B + 128 \end{aligned} \quad (6)$$

である。

3.2 特徴抽出領域

遺伝的アルゴリズム[6]などを用いて、創発的、自動的に特徴ベクトルを決定する方法も考えられるが、ここでは、なるべく軽い計算で目的の特徴を的確に抽出するために、目的を明確にした特徴ベクトル要素を以下の要領で決定する。

3.2.1 画像全体の色調

まず、最初は1枚の画像全体から、平均RGB($\bar{R}, \bar{G}, \bar{B}$)を求める。次に、この平均RGBからYCbCr値を計算する。具体的に、主な天気状況、時間帯の画像を用いて値を計算したものを表1に示す。

表1: 画像全体における各状況での値の違い

状態	\bar{R}	\bar{G}	\bar{B}	Y	Cb	Cr
夜間(図1)	95.2	76.2	36.1	82.4	107.5	139.2
夕刻(図2)	104.3	108.3	106.2	107.8	127.6	126.3
昼間晴天(図3)	178.6	180.8	181.4	170.7	128.5	127.0
昼間曇天(図4)	111.1	115.5	111.8	113.7	127.0	126.3
昼間雨天(図5)	190.8	191.3	192.4	180.3	128.5	127.7

さらに、各クラス(夜間、夕刻、昼間晴天、昼間曇天、昼間雨天)間でフィッシャーの線形判別法を適用し、各クラス間の平均RGB値、YCbCr値から求められるクラス内分散・クラス間分散比最大規準の値を求めたものを表2に示す。

表2: 平均RGB値、YCbCr値から求められたフィッシャーの判別関数

(a) 平均 RGB

\	夕刻	昼間晴天	昼間曇天	昼間雨天
夜間	13.88	18.40	22.55	1.83
夕刻	—	9.37	0.84	0.57
昼間晴天	—	—	8.00	0.26
昼間曇天	—	—	—	0.40

(b) YCbCr

\	夕刻	昼間晴天	昼間曇天	昼間雨天
夜間	10.63	15.74	16.14	1.52
夕刻	—	9.56	0.93	0.57
昼間晴天	—	—	8.07	0.26
昼間曇天	—	—	—	0.39

表2からわかるように、夜間とその他の状況の識別には平均RGBを用いることが有効であると考えられるが、その他の状況間の識別には平均RGB、YCbCrのどちらを用いても大差は生じないように思われる。したがって、画像全体のデータとしては平均RGB、YCbCrの両方を用いることにする。

3.2.2 部分領域1

まず、道路に向かって右車線の停止線手前の長方形領域 $D_1 = \{(x, y) | 230 \leq x \leq 550, 400 \leq y \leq 500\}$ (図6) を設定する。

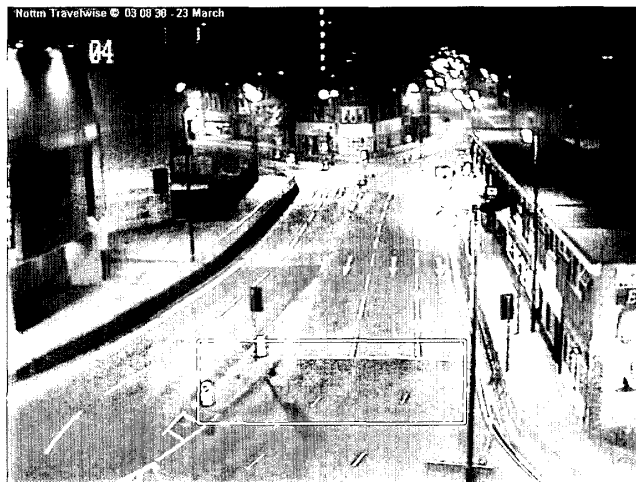


図6: 部分領域1 (中央下部白枠内)

この領域からは路面が乾いている、濡れている、光の照り返しがあるなどの情報が的確に得られることが期待できる。各状態の代表的であると思われる画像を3枚選び出し、(6)式を用いて各図における領域内の平均RGB $(\bar{R}_1, \bar{G}_1, \bar{B}_1)$ から YCbCr (Y_1, Cb_1, Cr_1) を求めた。ま

た、前小節と同様にこの領域の平均RGB値、YCbCr値から求められるクラス内分散・クラス間分散比最大規準フィッシャーの判別関数の値を求め、比較したところ、平均RGBのほうが識別に有効であったので、特徴ベクトルには平均RGBを用いることにする。

3.2.3 部分領域2

部分領域1と同様に、道路に向かって左車線の奥を捉えるように長方形領域

$$D_2 = \{(x, y) | 250 \leq x \leq 430, 130 \leq y \leq 270\}$$

を設定する。ただ、画像に向かって奥の部分はカメラが少し動作するだけで、捉える領域が大きくずれることになる。つまり捉える対象物体が異なるので、天候状況が類似する画像であっても、RGB値、YCbCr値が大きく異なることがあるので必ずしも参考に出来るわけではない。画像を2枚選び出し、(6)式を用いて各図における領域内の平均RGB($\bar{R}_2, \bar{G}_2, \bar{B}_2$)からYCbCr(Y_2, Cb_2, Cr_2)を求め、この領域の平均RGB、YCbCr値から求められるクラス内分散・クラス間分散比を比較したところ、部分領域2から各状況を分類するには平均RGBを用いることがよいことがわかった。

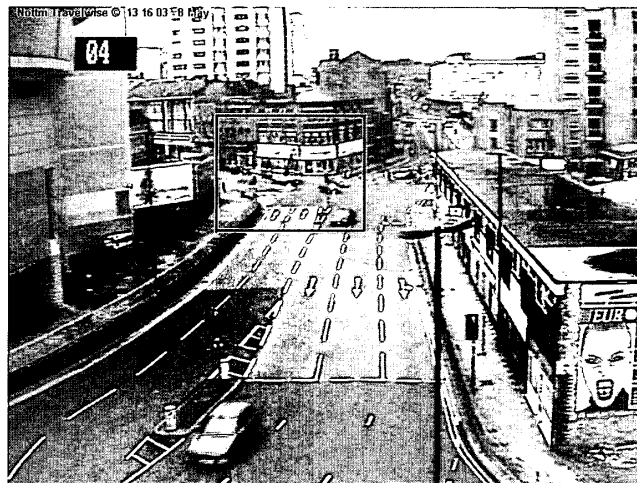


図7: 部分領域2 (中央部黒枠内)

3.2.4 部分領域3

画像の右最上部には、空が見えることが多い。天候の状態を知るのに適当な箇所であると思われるので、細長い長方形領域 $D_3 = \{(x, y) | 384 \leq x \leq 585, 1 \leq y \leq 20\}$ を設定した(図8)。

画像を4枚選び出し、(6)式を用いて各図における領域内の平均RGB($\bar{R}_3, \bar{G}_3, \bar{B}_3$)からYCbCr(Y_3, Cb_3, Cr_3)を求めた。さらに、この領域の平均RGB値、YCbCr値から求められるクラス内分散・クラス間分散比を比較したところ、部分領域3から各状況を分類するにはYCbCrを利用することがよいことがわかったので、YCbCrを利用する。

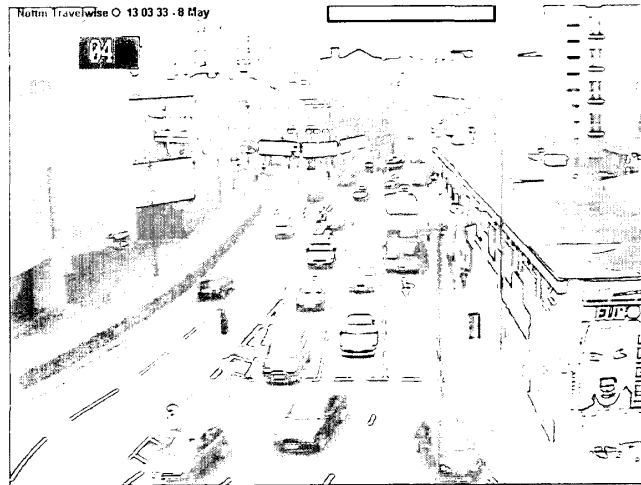


図8: 部分領域3 (右最上部黒枠内)

3.3 グレースケール画像のグレイの濃度

特徴ベクトルとの別の要素として、グレースケール画像における、特定の値以上・以下のピクセルの割合を用いる。濃度は値が大きいほど明るい画素、小さいものほど暗い画素である。

3.4 グレースケール画像の取り込み領域

グレースケール画像における特徴要素抽出のための定義領域を、3.2で述べたカラー画像の取り込み領域と同一とし、グレイ濃度の適当な値を決め、各クラス間でのクラス内分散・クラス間分散最大規準フィッシャーの判別関数を求め、比較した。その値の中から最良であると思われた値を用いて割合を求めた。

3.4.1 画像全体

1枚の画像全体からグレイ濃度180, 160, 130以上の割合の中で識別に最も有効な割合、80, 70, 60, 50以下の割合の中で識別に最も有効な割合をそれぞれ求めて利用することにする。

表4、5からわかるようにフィッシャーの判別関数が比較的大きい130以上と80以下の割合を利用することにする。この割合をそれぞれ $gray_{all}^{130}$, $gray_{all}^{80}$ とする。

表3: 画像全体における主要状況のグレイ濃度の割合

濃度	昼間晴天	昼間曇天	昼間雨天	夜間
180 以上	0.417	0.052	0.125	0.025
160 以上	0.593	0.081	0.215	0.036
130 以上	0.856	0.199	0.460	0.068
80 以下	0.018	0.075	0.106	0.529
70 以下	0.013	0.037	0.054	0.330
60 以下	0.011	0.017	0.027	0.241
50 以下	0.006	0.006	0.014	0.170

表4: 濃度 180, 160, 130 以上の割合から求められたフィッシャーの線形判別関数値

180 以上				
\	夕刻	晴天	曇天	雨天
夜間	3.047	4.107	8.359	0.493
夕刻	—	3.582	0.029	0.402
晴天	—	—	3.649	0.100
曇天	—	—	—	0.411

160 以上				
\	夕刻	晴天	曇天	雨天
夜間	3.941	5.014	15.658	0.628
夕刻	—	4.284	0.024	0.486
晴天	—	—	4.364	0.210
曇天	—	—	—	0.496

130 以上				
\	夕刻	晴天	曇天	雨天
夜間	4.295	43.111	14.378	1.247
夕刻	—	28.856	0.102	0.753
晴天	—	—	29.693	0.676
曇天	—	—	—	0.684

表5: 濃度 80, 70, 60, 50 以下の割合から求められたフィッシャーの線形判別関数値

80 以下					70 以下			
\	夕刻	晴天	曇天	雨天	夕刻	晴天	曇天	雨天
夜間	5.650	21.005	14.529	7.108	6.219	17.432	14.730	8.684
夕刻	—	54.658	1.811	0.478	—	4.978	2.765	0.631
晴天	—	—	5.1483	0.694	—	—	4.366	0.656
曇天	—	—	—	0.029	—	—	—	0.132

60 以下					50 以下			
\	夕刻	晴天	曇天	雨天	夕刻	晴天	曇天	雨天
夜間	10.206	19.987	18.658	14.086	8.361	12.650	12.457	11.294
夕刻	—	6.559	4.760	1.254	—	14.616	13.296	4.3245
晴天	—	—	2.158	0.626	—	—	0.306	0.871
曇天	—	—	—	0.222	—	—	—	0.589

3.4.2 部分領域1

3.2.2で述べた領域 D_1 におけるグレー濃度 180, 160, 130 以上の割合を求め、最良の割合をフィッシャーの判別関数の値の比較によって選ぶことにする。また、グレー濃度 80, 70, 60, 50 以下の割合の中で識別に最も有効な割合も同様に求めて利用することにする。以下の4枚の画像は3.2.2と同時刻の画像である。

表6: 部分領域1におけるグレー濃度

濃度	路面乾燥	照り返し	濡れている(1)	濡れている(2)
180 以上	0.020	0.962	0.122	0.006
160 以上	0.025	0.974	0.292	0.008
130 以上	0.056	0.986	0.946	0.275
80 以下	0.015	0.009	0.003	0.017
70 以下	0.006	0.006	0.001	0.011
60 以下	0.001	0.002	0.000	0.007
50 以下	0.000	0.000	0.000	0.004

多くのクラス間で識別が有効である濃度 130 以上の $gray_1^{130}$ と濃度 80 以下の $gray_1^{80}$ を部分領域 1 のデータとする。

3.4.3 部分領域 2

3.2.3 で説明した領域 D_2 におけるグレー濃度 180, 160, 130 以上の割合を求め、最良の割合をフィッシャーの判別関数の比較によって選ぶことにする。また、グレー濃度 80, 70, 60, 50 以下の割合の中で識別に最も有効な割合も同様に求めて利用することにする。

表7: 部分領域2におけるグレー濃度

濃度	状態 1	状態 2
180 以上	0.018	0.013
160 以上	0.042	0.029
130 以上	0.283	0.1300
80 以下	0.024	0.117
70 以下	0.009	0.030
60 以下	0.003	0.008
50 以下	0.001	0.004

多くのクラス間で識別が有効である濃度 130 以上の $gray_2^{130}$ と濃度 80 以下の $gray_2^{80}$ を部分領域 2 のデータとする。

3.4.4 部分領域 3

3.2.4 で説明した領域 D_3 におけるグレー濃度 180, 160, 130 以上の割合を求め、最良の割合をフィッシャーの線形判別関数の比較によって選ぶことにする。また、グレー濃度 80, 70, 60, 50 以下の割合の中で識別に最も有効な割合も同様に求めて利用することにする。

多くのクラス間で識別が有効である濃度 130 以上の $gray_3^{130}$ と濃度 80 以下の $gray_3^{80}$ を部分領域 3 のデータとする。

3.5 画像中のグレーの値の分散

最後にもう 1 種類の特徴ベクトルとして、グレースケール画像 1 枚の各画素の分散を利

用することにする。1枚の画像のサイズは Δ である。この $|\Delta|$ 個の画素のグレーの濃度値の分散 var をデータとして利用する。

4 実験結果

前章の領域を組み合わせることによって多次元ベクトルを設定し、画像データを次の条件の下でSOM上への写像、クラスタマップの作成を行った。各領域で使用するデータを次のようにする。

- 画像全体： $\bar{R}, \bar{G}, \bar{B}, Y, gray_{all}^{130}, gray_{all}^{50}, var$
- 部分領域1： $\bar{R}_1, \bar{G}_1, \bar{B}_1, Y_1, gray_1^{130}, gray_1^{80}$
- 部分領域2： $\bar{R}_2, \bar{G}_2, \bar{B}_2, gray_2^{130}, gray_2^{80}$
- 部分領域3： $Y_3, Cb_3, Cr_3, gray_3^{130}, gray_3^{80}$

データが正しく写像、分類されていると思われるSOM、クラスタマップについては、その特徴ベクトルを利用し、新しく画像データを写像し、その画像データがきちんと写像されているのかを検証する。

4.1 実験1

まず、画像全体のデータだけでどれほど分類できるかを見るために、実験1でのデータを画像全体のみの特徴ベクトル

$$data1(n) = [\bar{R}, \bar{G}, \bar{B}, Y]$$

として、表8の条件の下で写像実験を行った。SOMへのデータの写像9とクラスタマップ10を示す。

これらの図において、隣接するコードブックベクトルの内容が類似している場合、濃い色で表示し、差が大きいほど白い色で表示している。このマップを見ると、差が大きく、分離している部分が、白い帯になっていることが見て取れる。白くない部分においては、白っぽい部分で囲まれた領域がクラスタを成している。

図9内の \square が昼間晴天、 \square が昼間曇天、 \square が昼間雨天（路面が濡れている）、 \square が夜間、 \blacksquare が夕刻画像の集合を示している。結果は、夜間、夕刻、昼間晴天、昼間曇天、昼間雨天すべてがきれいに分類できている。ただ、図9の中央下部と左下部分、夕刻画像のデータ集合内に雨天画像のデータが点在する。また、右上部の昼間晴天画像の集合内に照り返しの多い雨天画像のデータが点在するなど全く異なる天気データが含まれている。クラスタマップには夜間とそれ以外の境界がきれいに出ている。その他の状況間、特に昼間曇天と夕刻画像の集合間の境界がはっきりしない。

表8: 実験1の条件

—	データ数	データ次元	格子点	繰り返し回数
実験1	500	4	30×30	100

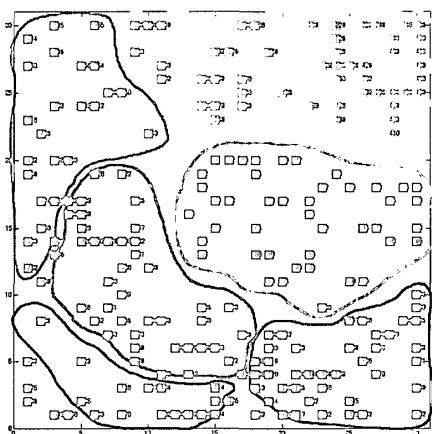


図9: 実験1 (SOMへの写像)

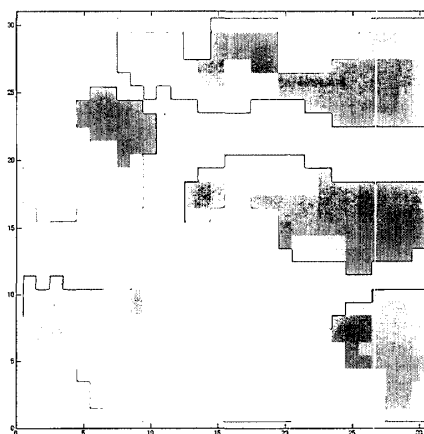


図10: 実験1 (クラスタマップ)

4.2 実験2

色情報を用いず、画像全体の輝度とグレー濃度の割合、部分領域1、2、3でのグレー濃度の割合を特徴ベクトルとし、

$$data2(n) = [Y, gray_{all}^{130}, gray_{all}^{50}, gray_1^{130}, gray_1^{80}, gray_2^{130}, gray_2^{80}, gray_3^{130}, gray_3^{80}]$$

として、表9の条件下で写像実験を行った。

表9: 実験2の条件

—	データ数	データ次元	格子点	繰り返し回数
実験2	500	9	30×30	100

図11内の \square が昼間晴天、 \square が昼間曇天、 \square が昼間雨天（路面が濡れている）、 \square が夜間、 \square が夕刻の画像の集合を示している。昼間雨天領域内に特徴ベクトルが類似した昼間晴天画像が紛れているが、データの分類としては成功である。ただ、環境状況の推定という点については精度が良くないと言える。クラスタマップは比較的明るい画像と暗い画像の境界がはっきりするものの、他のクラス間の境界ははっきりしない。

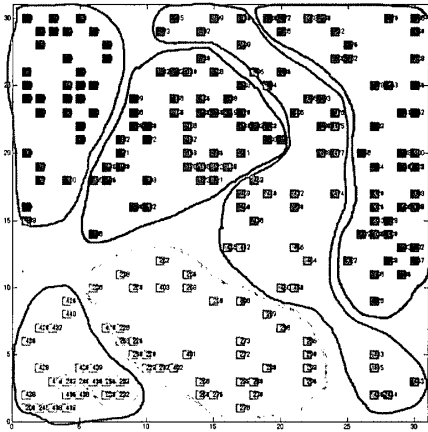


図 11: 実験 2 (SOM への写像)

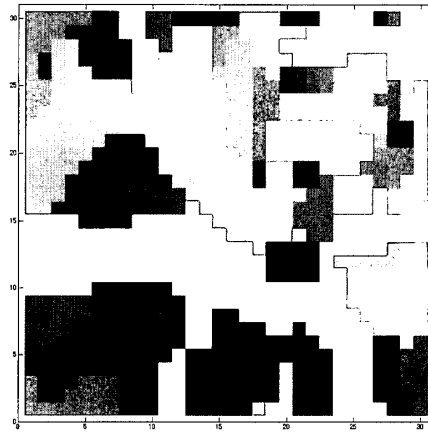


図 12: 実験 2 (クラスタマップ)

4.3 実験 3

画像全体、部分領域 1、2、3 の色情報とグレー濃度の割合を用いた特徴ベクトル

$$data3(n) = [Y, \bar{R}, \bar{G}, \bar{B}, \bar{R}_1, \bar{G}_1, \bar{B}_1, \bar{R}_2, \bar{G}_2, \bar{B}_2, Y_3, Cb_3, Cr_3, \\ gray_{all}^{130}, gray_{all}^{50}, gray_1^{130}, gray_1^{80}, gray_2^{130}, gray_2^{80}, gray_3^{130}, gray_3^{80}]$$

として、表 10 の条件下で写像実験を行った。

表 10: 実験 3 の条件

—	データ数	データ次元	格子点	繰り返し回数
実験 3	500	21	30 × 30	100

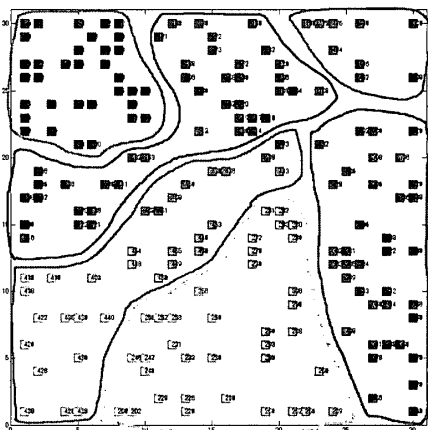


図 13: 実験 3 (SOM への写像)

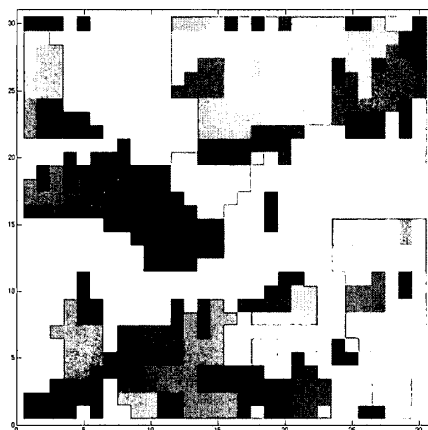


図 14: 実験 3 (クラスタマップ)

図13内の \square が昼間晴天、 \square が昼間曇天、 \square が昼間雨天（路面が濡れている）、 \square が夜間、 \blacksquare が夕刻の画像の集合を示している。昼間雨天画像の集合が2つに分かれてしまったが、クラスタマップはきれいに境界が見える。

このマップ上に新たに未確認の画像情報として100枚の画像のデータを写像させると以下ようになった。各○が新しいの画像データの写像結果であり、画像データは夜間：15枚、夕刻：20枚、昼間晴天：25枚、昼間曇天：20枚、昼間雨天（照り返し多い）：20枚である。この結果を図15に示す。この結果の画像番号を見ると新しい画像データが正しく写像されていることがわかる。

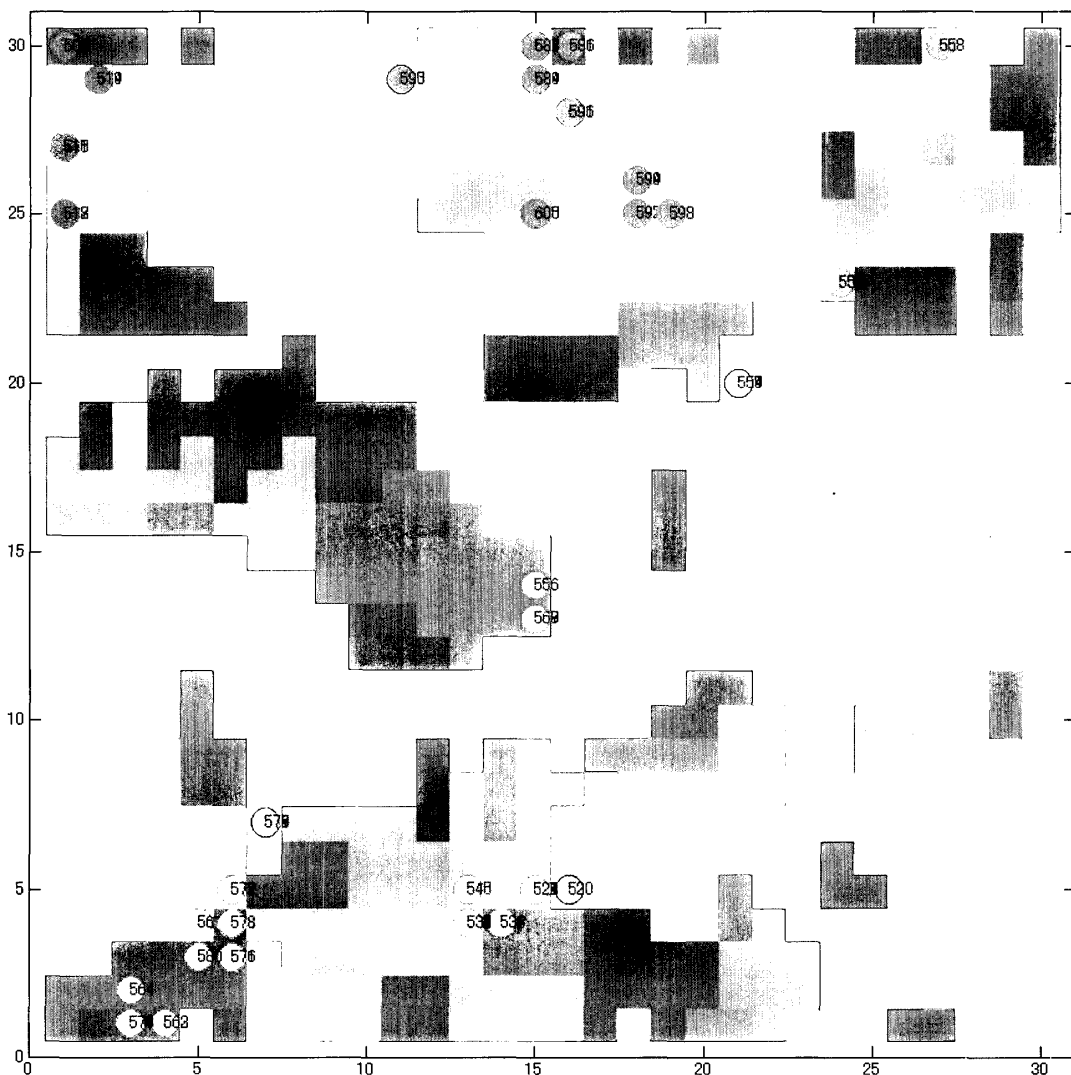


図15: 新しく画像を写像させたもの (2)

以上より、多次元データが2次元平面上に写像され、画像の状況ごとの分類が成功しているといえる。この中で実験3に関してはきれいに写像され、新しいデータもきちんと写像されていることがわかる。

4.4 計算時間

各実験に要した時間を表11にまとめた。なお、それらの時間は著者自身が作成したプログラムをMATLABで実行した際の経過時間である。ここに、Aは画像データ読込時間(秒)、BはSOM学習所要時間(秒)、Cはクラスタ化所要時間(秒)、Dは合計所要時間(秒)である。

表11: 各実験の所要時間 (秒)

実験番号	A	B	C	D
1	218	1204	75	1499
2	205	1223	76	1506
3	297	1223	94	1614

5 終わりに

画像を取得している環境条件を画像から読み取るためのシステムを考察した。画像の中に特徴が現れると考えられるものを、人間の経験的な情報をもとにして定義し、そこから得られる縮約された情報を特徴ベクトルに含めることにより、自己組織化マップを構成し、それを用いて教師なし学習を行うことにより、さまざまな気象条件や時間的な条件を、2次元のマップ上に自動的にマップ化することに成功した。ただ、昼間雨天の分類ができなかったことなど、当初予想した特徴が効果的に現れないものもあったが、特徴ベクトルの構成法を工夫し、たとえば、遺伝的アルゴリズムにより自動的に発見するような手法を取り入れることにより、さらに多様な情報を画像から取得することが可能になると思われる。これは今後の課題である。

謝辞

画像を提供して頂いたNottingham City Traffic Centreに感謝の意を表します。本研究の一部は文部科学省のオープン・リサーチ・センター整備事業による私学助成を得て行われた。

参考文献

- [1] Paolo Remagnino et al., *Video-based Surveillance Systems. Computer Vision and Distributed Processing*, Kluwer Academic, 2002.
- [2] Todd R. Reed (Ed), *Digital Image Sequence Processing, Compression, and Analysis*, CRC Press, 2005.
- [3] Guido Deboeck, Teuvo Kohonen (徳高平蔵, 田中雅博監訳), 金融・経済問題における可視化情報探索. 自己組織化マップの応用, シュプリンガー東京, 1999.
- [4] Teuvo Kohonen, *Self-Organizing Maps (2nd ed)*, Springer, 1997.
- [5] 徳高平蔵, 藤村喜久郎, 山川烈, 自己組織化マップ応用事例集—SOMによる可視化

情報処理一，海文堂出版，2002.

- [6] 坂和正敏，田中雅博，遺傳的アルゴリズム入門，朝倉書店，1994.
- [7] 坂和正敏，田中雅博，ニューロコンピューティング入門，森北出版，1997.
- [8] 田中雅博，ソフトコンピューティング入門，科学技術出版，1998.
- [9] 田中雅博，古河靖之，谷野哲三，“自己組織化マップを利用したクラスタリング,” 信学論，Vol.J79-D-II，No.2，pp.301-304，1996.
- [10] Marc M. Van Hulle (徳高平蔵，藤村喜久郎監訳)，自己組織化マップー理論・設計・応用一，海文堂出版，2001.
- [11] 石井健一郎他，わかりやすいパターン認識，オーム社，1998.