

## ハッチングパターンを用いた多属性気象データの可視化

宮村 (中村) 浩子<sup>†</sup> 古谷 雅理<sup>†</sup> 萩原 洋一<sup>‡</sup> 斎藤 隆文<sup>†</sup>

<sup>†</sup> 東京農工大学 大学院生物システム応用科学府

<sup>‡</sup> 東京農工大学 総合情報メディアセンター

miyamura@cc.tuat.ac.jp

本研究では、多属性気象データの相関を、ハッチングパターンを用いて可視化する。本研究グループが独自に開発した気象データサーバシステムは、複数観測点の気温、湿度、気圧、風速、風向、雨量データを1分間隔で測定し、長期間収集している。これらのデータの解析作業には、統計処理や折線グラフ表示などが用いられるが、複数の属性値の変動を同時に捉えるのは困難である。そこで、複数属性間の相関をハッチングパターンによって示す可視化手法を提案する。

### Multi-attribute Visualization Using Hatching Patterns for Weather Data

Hiroko Nakamura Miyamura<sup>†</sup> Tadasuke Furuya<sup>†</sup> Yoichi Hagiwara<sup>‡</sup> Takafumi Saito<sup>†</sup>

<sup>†</sup> Graduate School of Bio-Applications and Systems Engineering,  
Tokyo University of Agriculture and Technology

<sup>‡</sup> Information Media Center, Tokyo University of Agriculture and Technology

We propose multi-attribute visualization by using hatching patterns for weather data. Our research group has developed a weather data server system that measures weather data, including temperature, humidity, wind speed, precipitation, and pressure. The system acquires weather data for several locations and stores these data. Statistical analysis and plotting graphs are generally used to analyze such data, however these methods are not conducive for finding correlations in multi-attribute data. To overcome this shortcoming, we propose a visualization method that uses hatching patterns for determining relationships between multi-attribute weather data.

#### 1 背景と目的

本研究では、多属性データの属性間相関を、ハッチングパターンを用いて可視化する手法を提案する。

東京農工大学気象データサーバシステム [1] では、気温、湿度、気圧、風速、風向、雨量など複数の項目を1分間隔で記録している。これらのデータは、天気予報や災害予測だけでなく、動植物の育成管理など幅広い利用が期待されている。しかし、記録した気象観測データを利用する際には複数項目を複合的に解析する必要がある。

多属性データの解析に関しては、多くの取り組みがなされてきた。統計処理を用いて解析する他

に、折れ線グラフに代表されるグラフを複数表示することにより、それらの変動を把握する方法がとられている。しかし、統計処理を用いると、個々の属性値情報が失われてしまう。一方、複数グラフの表示では、個々の属性値の変動を捉えることはできるが、多数の属性値の変動を比較したり、相関を捉えたりするのは困難である。

そこで本研究では個々の属性値の変動を示しつつ、大局的な多属性間の相関を把握できる可視化手法を提案する。本手法はハッチングを用いることで、テクスチャパターンから大局的な相関を、ハッチング線から詳細な情報を得ることができる。

## 2 多属性データの可視化

情報可視化の分野において、多属性データの可視化は注目されている課題のひとつである [2]。1次元多属性データの可視化に関しては、同一次元軸上に属性数だけ折れ線グラフのようなグラフを描画する方法がある。これに対し、積み上げ式のグラフを応用利用することで複数の線が乱雑に交差する問題改善した ThemeRiver [3] が提案された。この手法では、各属性値の変動だけでなく、多数の属性値の変動も捉えられる。しかし、表示スペースを必要とする、積み上げる属性の順によって把握できる情報が異なるなどの問題がある。

また、属性を次元として、多次元データの可視化手法である Parallel Coordinates [4], StarIcon を利用した可視化手法 [5] などを利用する方法もある。Parallel Coordinates を利用する方法では、複数属性の相関を捉えることはできるが、時系列データのように1次元多属性データに適用すると、時間軸方向の変化を捉えにくいという問題が生じる。さらに、ThemeRiver と同様に、軸の配置順に依存して把握できる情報が異なる。StarIcon を利用する方法では、軸を放射状にとるため、軸の配置順が可視化結果に与える影響を抑えることができる。しかし、個々のアイコンの情報は捉えられるが、全体的な傾向を捉えるには至らない。

その他、ユーザの対話的な選択によって、可視化する属性を選択しながら表示する Film Finder [6] も提案された。この手法ではユーザが提示する情報を選択し、切り替え表示しながら観察することで複数属性間の相関を認識する。

本研究では、大規模なデータを扱うことから、ユーザの対話的な操作を必要としない可視化手法を検討する。また、可視化対象となる属性数は無限にとれ、その表示順に極力依存しないように、StarIcon の軸の取り方を採用する。さらに、時間軸方向の変動を把握できるように、時間軸を特別なものとして扱う。

## 3 ハッチングによる属性値の提示

時系列多属性データを扱うため、まず時間軸は特別なものとして配置する。また、多数の属性情報を提示するために、無限数の軸を設定できる StarIcon の考え方に基づき軸を設定する。大局的な情報は複数の StarIcon が集まって作るハッチングパターンによって提示する。ハッチングパターンが示す情報を比較検討するために、本稿では2種類のハッチング線を用意する。

### 3.1 時間軸の設定

まず、扱うデータは時系列データであるため、時刻を表す次元を設定する。そのために、2次元ウィンドウの縦軸には時刻をとる。ここでは1日分のデータが収まるように縦軸のマスの間隔を設定する。次に横軸には日付を設定する。つまり、各日のデータを縦1列に並べ、それを日付ごとに横に並べた格子が生成できる。これにより、縦方向には時間変化を観察でき、横方向には同時刻の数値変動を、日を追って観察できる。

このように、時間軸を2次元空間に規則正しく設定することで、多属性データの変動をハッチングのテクスチャパターンとして捉えやすくなる。

### 3.2 属性軸の設定

前項によって作成した格子上に多属性データの情報を描画する。無限数の属性データに対応するため、時間軸格子の中心から放射状に軸を設ける。この軸同士が成す角は、表示したい属性数に応じて決定する。つまり、属性数が1つのときは図 1(a)、2つのときは図 1(b)、3つのときは図 1(c) のようになる。3つ以上のときも同様に成す角が等しくなるように軸の向きを設定する。

このようにして設けた軸に対し、StarIcon と同様に各属性値を軸方向の線の長さで表示する。

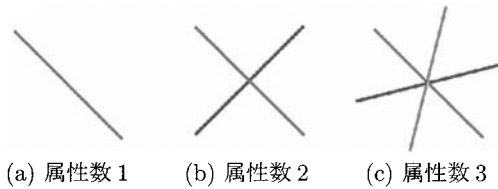


図 1: 多属性の軸の取り方

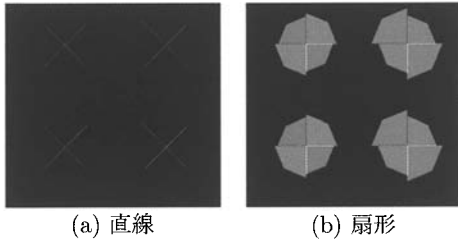


図 2: ハッチング線形状

### 3.3 ハッチング線形状

ハッチングパターンを生成する場合、向きをもった直線で表現する。ここでは、各属性の値の変動、相関をより直感的に示すために図 2 (a) に示す直線ハッチングだけでなく、幅をもったハッチング線の利用も検討する。

本手法では無限数の属性データが扱えることが条件である。しかし、単純に線を太くすると、値の小さい属性が、他の属性のハッチング線に隠れてしまう。そこで、線を扇形状に広げて幅をもたせることで、この問題を解決する (図 2 (b))。これは、表示する属性値が多い場合は直線に近く、表示する属性数が少ない場合は大きな中心角をもつ扇形になり、お互いの情報を隠すことはない。

ハッチング線の配置については、時間格子上に配置する。それぞれのハッチング線が交差しないように、属性値を正規化し間隔をあけて配置することも、間隔を狭めてハッチング線同士を交差させて特徴的なテクスチャパターンを発生させることもできる。

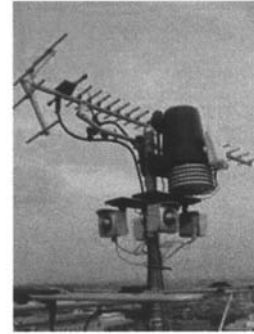


図 3: 東京農工大学気象データサーバシステム計測器 (校舎屋上)

## 4 実験

東京農工大学気象データサーバシステム [1] によって測定されたデータを用いる。ライブ気象台では、熱指数、屋外気温、体感温度、最高温度、最低温度、室外湿度、露点、風速、最大風速、風向、雨量、気圧、室内温度、室外温度を 1 分間隔で測定している。使用したデータは 2006 年 7 月 8 日 0 時 0 分～11 月 30 日 23 時 59 分の間、高さ約 30m の校舎の屋上に設置した計測器で測定したものである (図 3)。

図 4(a), (b) に 8 月の 1ヶ月間 (31 日間) 測定した屋外気温 (ピンク)、室外湿度 (緑) を直線ハッチング、扇形ハッチングそれぞれを用いて可視化した結果を示す。ここでは 1 時間ごとのデータを使用している。この可視化結果から、双方の値が昼間と夜間で異なるパターンを作り出していることがわかる。これは、昼は気温が高く湿度が低く、夜間はその逆になることを示している。また、数日昼に気温が上がっていない日がある。その日は昼でも湿度が高いことが結果画像から読み取れる。

次に、同じ期間のデータで、対象の属性数を増やして可視化した結果を示す。屋外気温 (ピンク)、室外湿度 (緑) に加えて、露点 (青)、風速 (黄)、雨量 (シアン)、気圧 (マゼンタ) を直線ハッチ

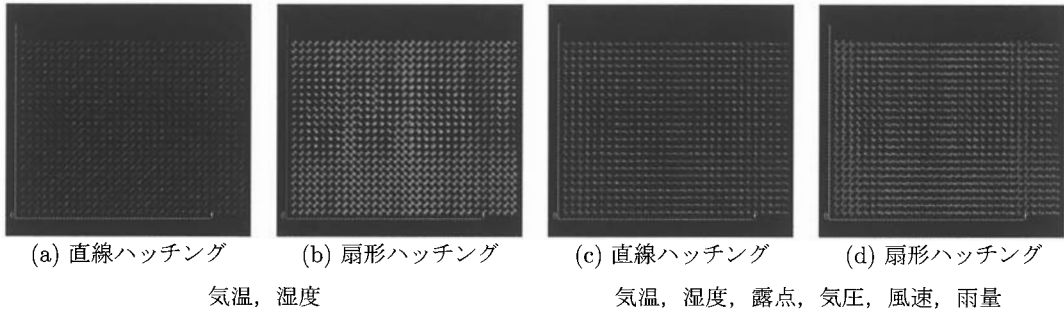


図 4: ハッチングパターンによる多属性データの可視化結果

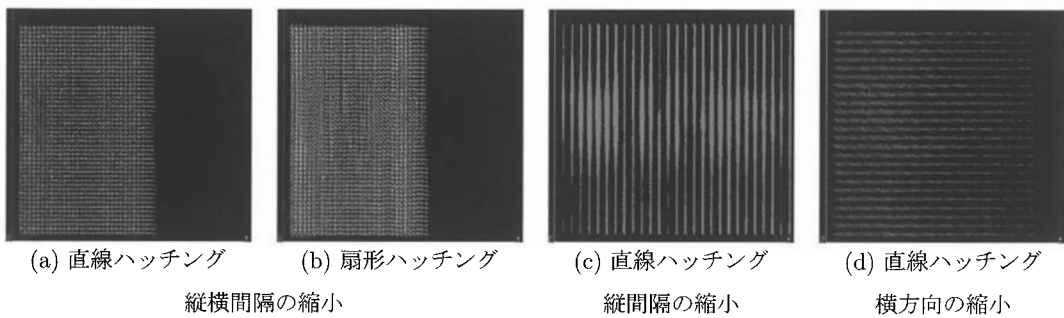


図 5: ハッチング線間隔を狭めた際の効果

ング, 扇形ハッチングそれぞれを用いて可視化した結果を図 4(c), (d) に示す. 図 4(a), (b) 同様に気温と湿度の関係を読み取ることができる. また, 風速や雨量が大きい日はほとんどなく, 気圧は高い日が数日あることがわかる. それぞれの属性値間の相関を詳細に読み取ることが難しいが, 可視化領域をパターン化し, 類似した領域に分けることが可能である.

よりハッチングパターンをテクスチャ化して傾向を読み取るために, ハッチング線の間隔を狭めた結果を図 5 に示す. 図 5(a), (b) では, ハッチング間隔を図 4 の  $1/2$  とした. 図 5(a) はハッチングに直線を用いている. ハッチング線の間隔狭めることによって, より複数属性値の変動をパターン化して捉えることができる. しかし, 扇型ハッチングを用いた結果 (図 5(b)) では, ハッチング間

隔が狭くなった分, 結果が混雑して認識しにくくなっている領域もある.

図 5(c) は, 時刻方向にハッチング間隔を狭めた結果である. 8 月の 1ヶ月間 1分ごとの屋外気温を可視化した結果である. 縦に伸びる線の太さに着目することで, 1日の気温変化を把握できる. また, 図 5(d) は, 全測定期間の 1時間ごとの屋外気温を, 横方向に狭めて可視化した結果である. これによって, 同時刻の気温がどのように変化したのかを折れ線グラフのように観察することができる.

このように提案手法では, 各属性値の変動を示すだけでなく, 全体的な傾向をパターン化して示すことができる. また, ハッチング線間隔を調整することで 1日単位, 1ヶ月単位などスケールを変えて観察することも可能である.

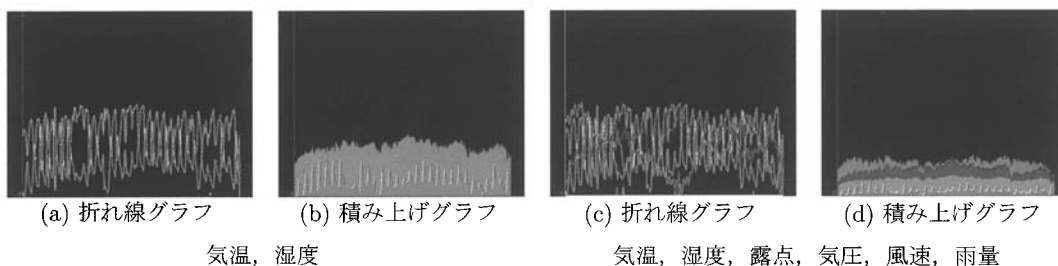


図 6: 折れ線グラフ, 積み上げグラフによる可視化

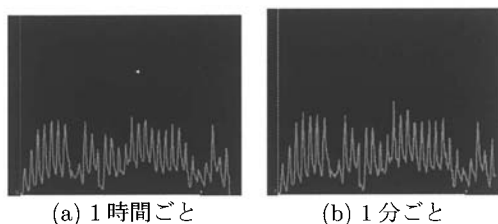


図 7: 詳細情報の提示精度

## 5 他の可視化手法との比較

図 6 に 8 月の測定データを折れ線グラフ, 積み上げグラフで表示した結果を示す. 色は図 4, 5 と同様に屋外気温 (ピンク), 室外湿度 (緑), 露点 (青), 風速 (黄), 雨量 (シアン), 気圧 (マゼンタ) を用いる. 図 6(a) は気温と湿度を折れ線グラフで可視化した結果である. 線の交差などから双方が逆の変動をしている様子がわかる. しかし, 線が混在するため, 直感的な理解は難しい. また, 1 日単位, 1 ヶ月単位などスケールを変えて観察することが難しい. ただし, 個々の属性値の変動を把握するには, 折れ線グラフの方が提案手法よりも優れていることがわかる.

図 6(b) は気温と湿度を積み上げグラフで可視化した結果である. 第一可視化属性として指定している気温の変動はわかりやすいが, 湿度の変動を読み取ることが困難である. これは, 湿度の帯の上辺が, 気温の値の影響を大きく受けるためである.

また, 帯の太さで値の変動を把握する必要があるため, 細かい変動の把握は難しい. ThemeRiver [3] のように, 滑らかな変動を見せるように調整することで, 双方の相関を捉えやすくなるが, そのためには表示に十分な領域が必要である.

図 6(c) は気温, 湿度, 露点, 風速, 雨量, 気圧の 6 属性を折れ線グラフで可視化した結果である. それぞれの線が複雑に絡み合い, 個々の属性値の変動を読み取ることも, 全体的な傾向を読み取ることも困難である. 縦軸方向に領域を分割し, それぞれの領域に各属性値のグラフを描画することも考えられるが, それでは複数属性間の相関を捉えることが難しいだけでなく, 多くの領域を必要とし, 多属性データの可視化には不向きである.

図 6(d) は気温, 湿度, 露点, 風速, 雨量, 気圧を積み上げグラフで可視化した結果である. 帯の幅からおおよそその変動を把握できるが, 属性値の積み上げ順によって得られる情報が大きく異なることが予測される.

これら従来手法による可視化結果と比較し, 本提案手法は個々の属性値の詳細な情報, 複数の属性値間の相関を捉えるのに有効である.

また, 図 7 に計測した気温データから 1 時間ごと, 1 分ごとのデータを用いて折れ線グラフ表示した結果を示す. 1 分ごとの計測データを用いても, 詳細な値の変動は限られた空間内には表現できないことがわかる. それと比較し, 提案手法では図 5(c) に示すように, 限られたスペースに詳細な情報を示すこともできる.

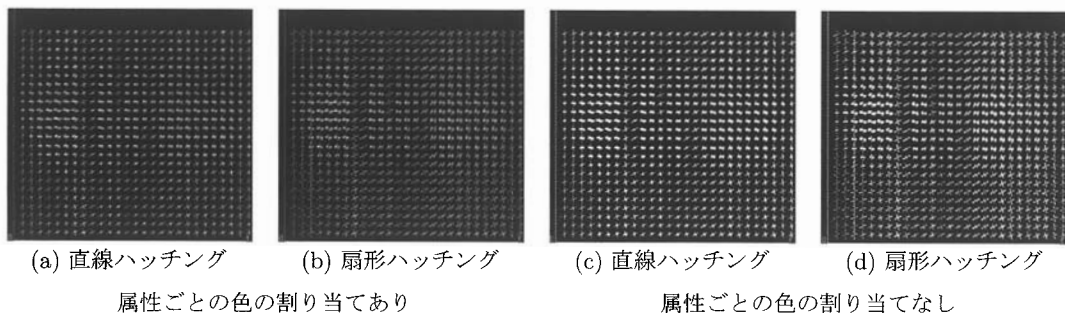


図 8: 属性数 13 のときの可視化結果

## 6 まとめ

本研究では、多属性データを可視化する手法を提案した。個々の属性値の変動を示すと共に、複数属性値間の相関を示すため、ハッチングを用いて相関をパターン化した。ハッチングパターンから相関傾向、ハッチング線の波形から値の変動を読み取ることができた。

また、多属性データを可視化する際に一般的に用いられる折れ線グラフ、積み上げグラフを実装し、それらの手法との比較を行なった。個々の属性値の変動を把握するのは折れ線グラフの方が提案手法よりも優れているが、多数の属性間の相関を捉えるには提案手法の方が有効であることが確認できた。提案手法において個々の属性値の変動を捉えたい場合には、ハッチング間隔を狭めることで認識しやすくなることも確認できた。

また、今回は属性数 6 までで実験を行なった。提案した手法では、軸を放射状に取るため、無限数の属性に対応できるが、ハッチングパターンを認識できる属性数には限界があることが予測される。図 8 に 13 属性値を可視化した結果を示す。ハッチングパターンから全体的な傾向を領域分けすることはできるが、個々の属性値を認識しながら観察することは困難である。そこで、今後はさらに多くの属性に適用し、提案手法の適用範囲、および可視化結果から読み取れる情報と精度の検証を行なう。

## 参考文献

- [1] 東京農工大学気象データサーバシステム, <http://www.tuat.ac.jp/~weather/>
- [2] D. A. Keim: "Information Visualization and Visual Data Mining," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 7, No. 1, pp. 1-8, 2002.
- [3] S. Havre, B. Hetzler, and L. Nowell: "ThemeRiver: Visualizing Theme Changes over Time," In *Proc. IEEE Information Visualization 2000*, pp. 115-124, 2000.
- [4] A. Inselberg: "The Plane with Parallel Coordinates," *Visual Computer*, No. 1, pp. 69-97, 1985.
- [5] J. Abello and J. Korn: "Mgv: A System for Visualizing Massive Multi-digraphs," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 8, No. 1, pp. 21-38, 2002.
- [6] C. Ahlberg and B. Shneiderman: "Visual Information Seeking Using the FilmFinder," In *Proc. ACM CHI 94 Conference Companion*, pp. 433-434, 1994.