

# 「文化財情報処理システム」における画像データ

大野 康 男

## 1. はじめに

小稿は、「文化財情報処理システム」を構成するデータベースの一つである「既調査遺跡データベース」に含まれる画像データについて、クライアントマシンのディスプレイへの表示に関わる問題を紹介するもので、今後考えられる同システムの整備・拡張の参考に寄与することを目的とする。

## 2. 画像データの前提

### (1)画像データの概要

「文化財情報処理システム」における画像データは、同システムの性格から、紙もしくは印画紙に代わるメディアを目的としたものではない。画像データは、検索結果に対して視覚的なイメージを構成する一要素であり、詳細な画像については、利用者が自ら現物を確認するということを前提としている。したがって、ある程度のレベルで表示された画像の内容が理解できるという範囲の解像度しか要求していない。

現在登録してある画像データは、縦方向でおよそ300ドット程度の8 bitの256色カラー画像もしくは、256階調のモノクロ画像がほとんどであり、一部モノクロの2値画像が含まれる。

### (2)クライアントマシンの前提

本システムで使用するクライアントマシンはNECのPC98シリーズを選定している。ディスプレイは17インチのものを使用し、表示解像度は1125×750ドットのいわゆる98ハイレゾで、256色が表示できることを前提としている。クライアントマシンでフロントエンドツールとして使用するソフトウェアはWingzである。画像データは、Wingzのワークシートに貼り付けることで表示している。具体的にはWingzが備えるHyper ScriptによるReadPictureによる貼り付けである。したがって、最終的な画像データのフォーマットはWindows BitMap形式で標準的なBMPとなる。ただし、BMP

形式の画像をそのままデータとして保持することは、サーバーの記憶容量に問題があったため、データベースへの登録はjpeg圧縮を行っている。データはinformixにおいてgazouというテーブルを用意し、バイナリデータとして登録している。jpeg圧縮は、テレフォースのコンプレッションマスターを使ったハードウェア圧縮であり、イメージスキャナー等で読み取った画像をjpegに変換し、サーバ上のデータベースへ登録する。クライアントマシンへの表示は、検索によるサーバからの読み出しに際し、検索結果をクライアントマシンに返した後、クライアントマシン側で、コンプレッションマスターを使って再度BMPへ変換し、表示するというプロセスをとっている。

なお、コンプレッションマスターのハードウェアの制限から、扱える画像は縦方向で512ドットを限界としている。

## 3. 現在の問題点

現在登録してある画像は、縦方向で300ドット程度の比較的小きな画像である。これは、画像の表示速度があまりにも遅いため、必要最低限の大きさにしているためである。さて、画像データはモノクロ画像・カラー画像を問わず対応している。現在各調査事務所に導入してあるクライアントマシンは、標準で1677万色中256色が表示できるものである。ここで、開発当初予期できなかった問題があった。まず、コンプレッションマスターでjpeg圧縮した画像は24Bitの画像データとなる。このため、一旦圧縮した画像を再度クライアント側で伸張すると、当然のことながら24Bitのフルカラー画像に伸張されるのである。現在のところ、元となるデータは大部分が8Bit、すなわち256色のデータである。256色のBMP画像は、画像ファイルにパレット情報をもっており、256色という少ない色で比較的色彩バケのない表示をすることができる。また、Wingzのワークシートがもつカラーテーブルは、

外部から明示的にコントロールすることはできない。そのため、24Bitに伸張された画像データは、その時点でWingzのワークシートがもっているカラーテーブルの近似したパレットを使用して表示されることになる。その結果、本来の24Bitの色を再現することができず、非常に汚い画像となって表示されてしまう。汚いといっても想像しにくいですが、ワークシートのもつカラーテーブルはデフォルトで各色をバランスよくもっているため、中間色はほとんど再現できなくなる。場合によっては、ソラリゼーションを起こしたような画像が表示されるのである。具体的なイメージがわきにくいですが、極端に表現すると、赤・緑・青の3色からなる画像データを表示するとき、表示できる色が水色・青・紫の3色しかなかったら、表示される画像は何色となるだろう。

現在、この問題を回避するために、画像データをロードする前に、ダミーの画像ファイルをワークシートに貼り付け、カラーテーブルを明示的に調整している。ダミーの画像は1枚しか用意しておらず、なかなか、すべての画像をこのダミーの画像のカラーテーブルで再現することはむずかしい。カラー画像については、4Bit・8Bit・16Bit・24Bitと試みてみたが、現在のダミーの画像が8Bitであるため、8Bitより大きい16Bitや24Bitの画像は色バケが著しい。同様に4Bitの画像でも色バケが著しい。このため、イメージスキャナ等で作成した画像ファイルは、圧縮・伸張を行った後、用意してあるダミーの画像がもつカラーテーブルでの色の再現性を確認してから、データベースへ

の登録を行っている。

特に、Photo CDの使用は、一旦8Bitの画像にDepthするのであるが、この時のパレットの変換がむずかしい。

#### 4. おわりに

それほど精細な画像の提供を目的としたシステムではないにしろ、現在の画像の表示には問題が残っている。この問題の具体的な改善策となると難しいのであるが、一つはハードウェア、特にビデオメモリを多く載せることにより、マッピングできる色情報を増やすことが考えられる。98ハイレゾで24Bitを表示するには2.5MB近いメモリを必要とする。実際には24Bitまでの画像を扱う必要はないと考えられ、12Bitもしくは16Bit程度の画像表示で十分なはずである。

もう一つは、画像データの登録時にJPEG圧縮を行わず、8bitのBMP画像をそのままデータベースへ登録する方法がある。ちなみに、300×300ドットの画像データを8bitのBMPファイルとして保存したところ、約90kbであったのにたいし、JPEG圧縮(Photo Magicによりmore compressionを使用)したところ約9.5kbとなった。もともと、画像ファイルの大きさは一概に比較することはできないが、サーバーに大きな補助記憶装置を用意する必要が生じてしまう。現在のJPEG圧縮は1/10程度を目安としている。

いずれにしても、コスト的に問題が残り、あまり現実的な方法ではない。残念ながら現在のところ、積極的な改善策はないかもしれない。