

IBM PC VGA用 畫像處理 소프트웨어 (IMAPRO)

徐在榮·智光薰

韓國資源研究所

(1992년 5월 20일 받음; 1992년 5월 30일 수리)

Image Processing Software Package(IMAPRO) for IBM PC VGA

Jae Young Suh, Kwang Hoon Chi

Korea Institute of Geology, Mining and Resources(KIGAM)

(Received May 20, 1992 ; Accepted May 30, 1992)

Abstract

The IMAPRO software package was mainly focused to provide an algorithm which is capable of displaying various color composite images on IBM PC, VGA(Video Graphic Array) card with no special hardware. It displays the false color images using a low-cost eight-bit plane refresh buffer. This produces similar quality to the one obtained from image board with three eight-bit plane. Also, it provides user friendly menu driven method for the user who are not familiar with technical knowledge of image processing. It may prove useful for universities, institute and private company where expensive hardware is not available.

1. 序論

畫像處理(image processing)는 人工衛星이나 航空機에 의해 收集된 資料를 處理, 分析하여 資源探查, 污染監視, 病蟲害監視, 作況豫報, 收穫量調查, 水質分析 및 監視, 水溫測定 等 地質, 海洋, 林業, 農業, 環境 等の 各 分野에서 폭넓게 使用되고 있으며, 最近에는 廣告, 醫學分野 等에서도 利用되고 있으나, 高價의 image board 및 特定한 모니터와 소프트웨어를 購入하여야 하는 經濟的인 問題와 畫像處理에 있어서 高度의 技術이 要求되는 技術的인 問題 때문에

一般人の使用이 극히 制限的이다.

本 研究는 専門知識이 없는 一般人에게도 쉽게 使用될 수 있고, 高價의 裝備購入이 必要 없이 現在 業務用 等으로 많이 使用하고 있는 VGA(Video Graphic Array) 搭載 PC를 利用할 수 있는 畫像處理 소프트웨어를 開發함으로써, 先進國에 비해 落後되어 있는 畫像處理 技術의 發展과 擴散에 寄與하고자 하였다.

2. 畫像데이터의 構造

畫像處理에서 使用되는 畫像資料(本 論文에서는 주로 LANDSAT 資料를 意味한다)는 header와 data 部分으로 構成되어 있다. Header에는 畫像의 크기(column數 等), 샘플別 비트數, 壓縮 및 貯藏方法 등이 記錄되어져 있고, data 部分에는 人工衛星이나 航空機에 搭載되어진 sensor에 의해 地表面으로부터 反射 혹은 放射되는 電磁波를 收集하여 된 資料는 파일로 貯藏되는데, 이 때 어떠한 파일 포맷으로 貯藏하는냐에 따라 header에 수록되는 情報를 달리 한다. 本 IMAPRO에서는 人工衛星 資料의 header를 分析하여 IMAPRO의 固有 format으로 있는 KGM(KIGAM)으로 變換하여 使用한다. 만약 KGM format이 아닌 다른 畫像 파일 format을 處理할 때에는 header 및 畫像의 길이, 그리고 幅 等を 入力하여야 한다.

KGM 畫像데이터의 構造는 Fig. 1과 같이 header 部分에는 畫像데이터의 位置(Path-Row): 30 바이트, 走査한 날짜를 나타내는 Date: 30 바이트, 畫像의 길이와 폭을 나타내는 Size: 30 바이트, 그리고 NXLEN, NYLEN 各各 2 바이트를 包含하여 모두 94 바이트를 차지하고, Data 部分에는 畫像의 左上端부터 NXLEN의 길이 만큼 1 line씩 차례로 NYLEN line이 貯藏된다. 이 때 데이터의 1 바이트는 畫像의 1 pixel을 表示한다.

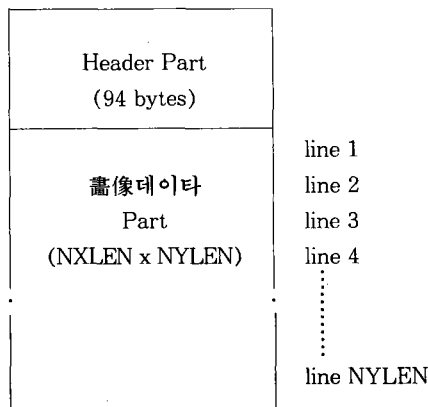


Fig. 1. KIGAM format.

3. Color Composite Image 作成

3.1 Full Color 디스플레이 방법

高價의 畫像處理 專用裝置에서 畫像을 디스플레이하기 위하여 Fig. 2에 表示한 것과 같이 one pixel을 RGB(Red, Green and Blue) 各 8 bits씩 總 24 bit-plane을 使用한다. RGB는 各各 256個의 다른 明暗을 區分할 수 있으므로 使用可能한 色相의 數는 253^3 個이다. 24 bit-plane 에서는 이처럼 많은 色相을 使用하여 畫像을 모니터에 出力할 수 있으나, 色에 관한 專門家 들도 10-12 bit 程度의 識別力이 있으면 充分하므로 一般人들이 畫像處理 專用裝置를 構築하는 것은 經濟的인 側面에서 不合理하다.

24 bit-plane에서는 3個의 畫像화일을 RGB로 分類하여 各各 8 bit(one pixel)의 畫像을 만든 후, 그 畫像에 該當하는 色相을 Red, Green, 그리고 Blue別로 'Pixel Value Mapping'을 통하여 만든다. RGB의 色相值를 DAC(Digital to Analog Converter)로 보내주면 RGB 各各의 色相을 24 bit-plane으로 hardware的인 方法으로 組合하여 RGB gun을 통하여 모니터(CRT)로 出力시킨다. 이러한 方式을 利用하여 畫像을 디스플레이하는 方法을 'Full Color Display Scheme'라고 한다. 최근에는 이러한 裝備는 거의 찾아볼 수 없고 one pixel 當 8 bits를 使用하여 畫像을 表現하는 비디오 카드들을 使用하고 있다.

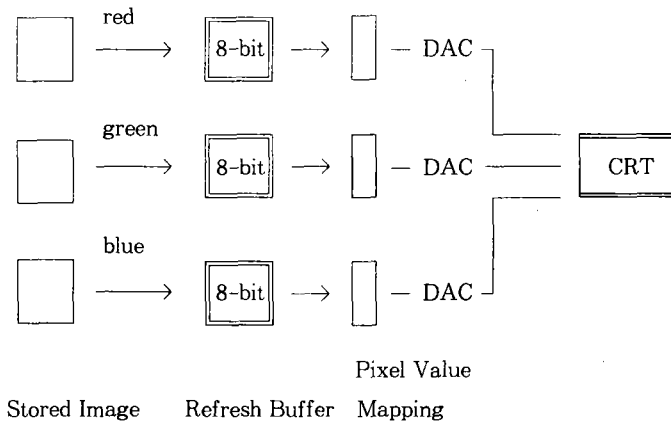


Fig. 2. Color Composite Display for 24 bit-plane.

3.2 VGA 카드를 이용한 디스플레이 방법

一般的으로 널리 사용되는 IBM PC 또는 互換機種의 VGA 카드와 Workstation에서는 보통 8-bit refreshing buffer를 사용하며, RGB는 각각 64가지의 각기 다른 명암을 표현할 수 있으므로 RGB의 명암을 모두 조합하여 색상을 표시할 때에는 6 bit씩 모두 18 bit가 필요하나, VGA 카드에서는 hardware의으로 8 bit만 사용한다. 그래서 본 IMAPRO에서는 각 64개의 명암 중 Red에서 6가지, Green에서 7가지, 그리고 Blue에서 6가지를 조합하여 모두 252개의 각기 다른 색상을 추출하여 畫像을 디스플레이하였다. 8 bit plane에서 동시에 사용할 수 있는 256개의 색상 중 252개를 사용한 이유는 限定된 색상 중에서 RGB의 색상을 골고루 표현하기 위하여 6, 7, 6의 패턴을 사용하였다. 256개의 색상을 출력시키기 위하여는 8, 8, 4의 패턴을 조합하여 나타낼 수 있으나 Blue 색상이 약하게 디스플레이되는 短點이 있다.

畫像의 디스플레이시 考慮하여야 할 重要的 事項은 入力되는 畫像의 畫素值에 대한 分布圖, 즉 各 畫素의 最低值와 最高值, 平均值, 그리고 分布 갯수 等を 分析한 값을 파라미터로 入力하여 디스플레이 할 時 선명한 畫像이 出力된다.

畫像을 모니터에 디스플레이 하려면 畫像의 畫素值 分布를 토대로 作成된 CLUT에 Mapping시켜 주어야 하는데, CLUT의 index 값은 아래 式과 같이 RGB별로 畫像의 最小值와 最高值를 基準으로 하며, Red와 Blue 畫像은 10, 그리고 Green은 8로 나누어서 얻은 定數를 代入하여 얻은 結果 z의 값이 된다.

$$z = 42r + 6g + b$$

이 때 z의 값은 0부터 251 사이의 값이며, 모니터에 出力되는 畫像의 Composite Color 이다. 畫素值의 $\Delta D(DN_{max} - DN_{min})$ 값의 差가 클수록 畫像의 Composite Color는 鮮明하지 않기 때문에 그 隔差를 좁히는 것이 有利하다. 主要 알고리즘은 다음과 같다.

$$i = TT_{red} [red]$$

$$k = TT_{green} [green]$$

$$k = TT_{blue} [blue]$$

$$dm[i, j, k] = dm[i, j, k] + 1$$

$$s = T[r] = \left(\frac{N P[r]}{Total - 1} \right) \quad \begin{array}{l} P[r] : \text{number of pixels at gray level } r \\ T[r] : \text{approximately Total/N pixels} \end{array}$$

$$P_{\text{blue, red-part}}[a], \text{green-part}[b] = \sum_{i=i_{\text{ired}}[a]}^{s_{\text{ired}}[a]} \sum_{j=j_{\text{igreen}}[b]}^{s_{\text{igreen}}[b]} dm[i, j, k]$$

Fig. 3은 VGA 카드 위에서 Composite Color를 作成하는 方法을 diagram으로 表示하였다.

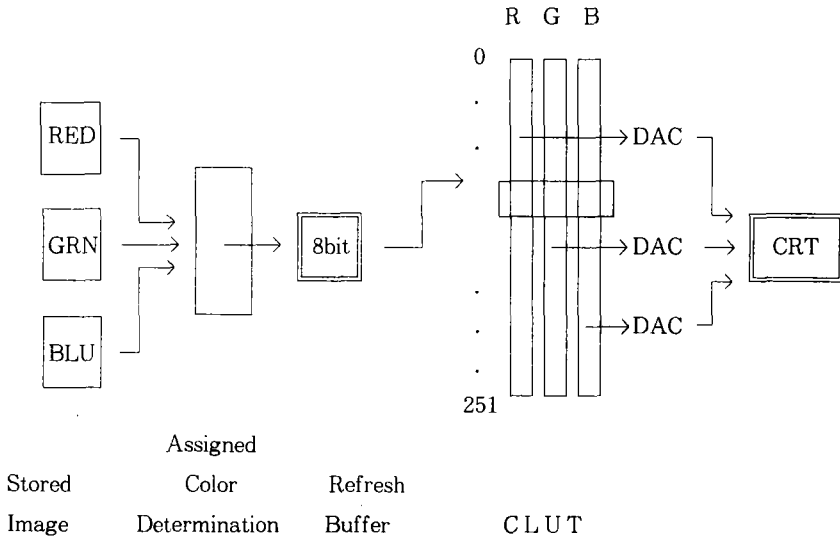


Fig. 3. Composite Color Display for VGA.

4. IMAPRO의 主要機能

IMAPRO는 Multi-Channel data의 合成, 統計處理, 濃度變換에 의한 強調處理, 濃度分割, 比演算處理, Filtering, 線構造線(Lineament)抽出 等を 實行시킬 수 있는데, 그 主要機能을 아래에 表示하였다.

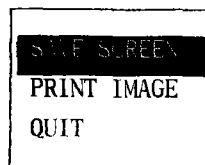
Main Menu

Here are available Modules :

- { 1} Display Image(one or three band)
 - { 2} Histogram in Graphic or Numeric
 - { 3} Filtering Image Module
 - { 4} Ratioing Image Module
 - { 5} Lineament Extraction Module
 - { 6} Density Slice Image Module
 - { 7} Display Saved Image File(xxxxx. SIF)
 - { 8} Displays Available Color Table
 - { 9} Edit Parameter
 - {10} DOS command
 - {11} Help
 - {0, Esc} Quit to DOS
- >>> Enter Desired Option(0~11) :

4.1 Display Image(one or three 밴드)

入力되는 畫像 화일(band)의 數에 따라 Fig. 4에 表示된 것과 같이 黑白 또는 Composite Color로 화면에 디스플레이 되는 모듈로, 畫像의 크기에 따라 縮小 또는 擴大가 可能하고, 원하는 畫像의 一部分만을 抽出하는 機能을 갖고 있다. 또한, cursor에 의해 畫素의 價値와 그 값을 x, y, z로 表示할 수 있으며, 마우스가 設置된 시스템에서는 아래와 같은 메뉴를 불러내어 디스플레이된 畫像을 貯藏, 프린트 또는 DOS로 빠질 수 있는 것이 特徵이다.



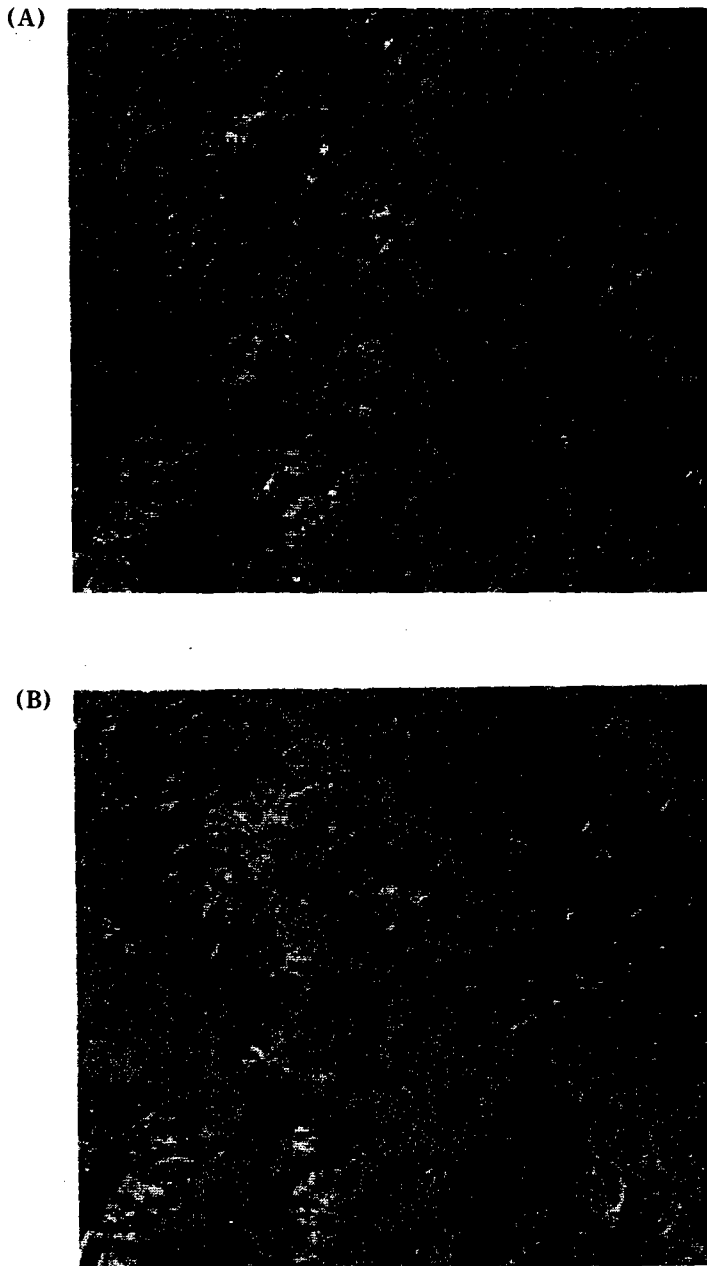


Fig. 4. Output of B/W and Composite Color Image.

4.2 Histogram in Graphic or Numeric

畫像 데이터가 가지고 있는 density value의 分布와 最小, 最大, 그리고 平均值를 그래픽 또는 數値의 形態로 보여주는 모듈이다. 특히 數値로 表示된 값들은 텍스트 화일 'density. rgb'로 貯藏되며, 畫像 데이터에 대한 여러 가지 情報들도 같이 貯藏되는 것이 特徵이다.

4.3 Filtering Image Module

畫像의 特徵을 抽出하기 위하여 아래와 같은 4가지의 필터링 方法을 使用하였다.

1. Smoothing
2. Edge Enhancement
3. Differential
4. Smooth with Weighting

各 필터링은 Fig. 5에 表示된 것과 같은 3×3 mask를 使用하여 다음과 같은 式에 의해 計算된다.

Smoothing :

$$a' = (a+b+c+d+e+f+g+h+i)/9$$

Edge Enhancement :

$$a' = e^2 - (b+d+f+h)/4$$

Differential :

$$a' = ((b-h)^2 + (d-f)^2)^{1/2}$$

Smooth with Weighting :

$$a' = (a+2b+c+2d+4e+2f+g+2h+i)/16$$

a	b	c
d	e	f
g	h	i

Fig. 5. 3×3 mask.

4.4 Ratioing Image Module

두 개의 畫像 사이를 演算한 後, 그 結果를 새로운 畫像으로 貯藏하는 모듈로써, IMAPRO에서는 4가지의 연산자를 提供한다. Ratio 方法은 두 개의 畫像 a, b 화일에 대해 아래와 같은 演算式을 각 畫像에 對應되는 pixel 값에 適用하여 새로운 畫像 c를 만드는 方法이다.

1. $C = \tan^{-1}(A/B) * (2/\pi * 255)$
2. $C = (A/(A+B)) * 255$
3. $C = A - B$
4. $C = A + B$

4.5 Lineament Extraction Module

어느 地形의 條件에서도 專門家の 專門 知識이 없이도 線形 構造를 自動 抽出할 수 있는 技法을 페케이지화한 모듈로써, 3×3의 filtering을 使用한 후, 臨界值 以下の 것을 제거하므로써 雜音이 적은 線構造線(Lineament) 畫像을 作成하여 그 結果를 새로운 화일 이름으로 貯藏하는 것이 特徵이다.

4.6 Displays Saved Image File(XXXX. SIF)

디스플레이된 畫像을 實際 모니터에 出力되는 畫素值로 變換하여 貯藏할 수 있다. 이 때 貯藏되는 포맷은 IMAPRO에서 自體的으로 開發한 畫像화일 포맷을 使用하고 있다. 디스플레이 모듈에 의해서 畫像을 出力할 경우, 畫像을 1라인씩 읽어서 RGB別로 明暗 分布值를 抽出하여 畫像處理 可能한 8-bit로 만든 다음, CLUT의 色相을 모니터에 出力하는 여러 스텝을 거쳐야 하나, SIF 화일 포맷은 모니터에 出力되는 畫素值가 貯藏되어 있어 上記에 說明된 스

템들을 거치지 않고 바로 出力하므로써 速度가 디스플레이 모듈보다 현저하게 빠르다. SIF 화일의 構造는 다른 畫像 화일과 같이 header와 data 部分으로 區分되는데, header 部分에는 아래와 같은 情報가 收錄된다.

- VGA 카드의 種類
- Graphics의 解像度
- Printer Attributes 화일 이름
- 화일 길이
- 화일 폭
- 디스플레이 모드(B/W or Color Composite)

5. 印刷方法

畫面에 디스플레이된 畫像을 프린트하는 方法은, 畫像이 디스플레이된 後, mouse의 왼쪽 버튼 또는 'F1'키에 의해 메뉴를 불러낸다. 그리고 'PRINT IMAGE' 位置로 box를 移動하여 出力시킨다. 이 때 良質의 出力을 얻기 위하여는 프린트時에 必要한 파라미터들을 補充하여야 한다. 이들 파라미터들은 화일 'printer. atr'로 貯藏되어 있다.

6. 實行環境

本 IMAPRO 패키지의 實行을 위해 갖추어야 할 시스템의 構成은 아래와 같이 一般的으로 사용하고 있는 PC based system이다. IBM PC/386 以下の 機種에서는 處理速度가 떨어지므로 計算을 빠르게 하여 주는 Co-processor의 搭載가 實用的이며, 畫像出力時 高解像度의 畫像을 必要로 할 경우에는 1MB(1024×768×256 color)容량의 VGA 카드 使用을 권유한다.

- IBM PC/AT/386/486 또는 호환기종
- 칼라 모니터
- VGA 카드
- 칼라 프린터(畫像出力時)
- 2 또는 3 버튼 마우스

7. 結 論

IMAPRO는 畫像處理에 대한 專門知識이 없는 一般人에게도 쉽게 使用할 수 있고, 高價의 image board 및 特定한 모니터나 소프트웨어의 購入이 必要 없이 現在 業務用 등으로 많이 使用하고 있는 VGA(Video Graphic Array) 搭載 PC를 利用할 수 있는 소프트웨어의 開發을 目的으로 하였다. 追後 幾何學的 補正, 主成分分析 等の 모듈 開發과 機能의 補強은 물론 使用者가 보다 使用하기 편하기 위하여 다음과 같은 機能을 追加함으로써, 畫像處理 技術의 發展과 擴散에 寄與하고자 한다.

- 가. 한글화
- 나. Pop UP 메뉴 方式 採擇
- 다. 포인팅 裝置 使用의 極大化

參 考 文 獻

- 1) Microsoft Corporation, 1990, Microsoft C 6.0 Run-Time Library Reference.
- 2) Microsoft Corporation, 1990, Microsoft C 6.0 Language Reference.
- 3) Media Cybernetics, 1990, Halo Professional 2.0 Programmer's Guide, Library Reference, Device Reference.
- 4) IBM Corporation, 1984, IBM Reference Macro Assembler and IBM Macro Assembler 2.0, Personal Computer Software.
- 5) IBM Corporation, 1983, IBM Technical Reference Personal Computer XT/AT.