รายการอ้างอิง

ภาษาอังกฤษ

1. http://Ourworld.compuserve.com/homepages/stereographics. Depth Cue, Receiving Stereoscopic Images, Composition, Stereo-Vision format

2. Mcalister, D.F. 3D Display. **Byte** 17 (May 1992) : 183-188

3. Hodges, LF. Time-Multiplexed Stereoscopic Computer Graphics. **IEEE Computer Graphic and Application** (March 1992) : 20-30.

4. James D.Murray and William vanRyper. JPEG File Interchange Format, File.FLI ,BMP **Encyclopedia of Graphics File Formats.** USA. O'Reilly & Associates, Inc. (July 1994) : 282-297, 377-383.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก

รูปแบบแฟ้มข้อมูลที่เกี่ยวข้องกับการรวมภาพ

1.JPEG File Interchnge Format

ลักษณะทั่วไป

JPEG (Joint Photographic Experts Group) ตามมาตรฐานทั่วไปจะหมายถึงวิธีในการลดขนาด file และบางครั้งคือรูปแบบของ file จริง ๆ แล้วลักษณะเฉพาะของ JPEG คือการลดขนาดข้อมูล ไม่ใช่การกำหนดการเปลี่ยน file ปกติไปเก็บข้อมูล และเคลื่อนย้ายข้อมูล JPEG ระหว่าง computer platforms และ operating systems. JPEG File Interchange Format หรือ JFIF คือการพัฒนาของ C-Cube Microsystems โดยมีวัตถุประสงค์เพื่อเก็บรักษาข้อมูลที่ได้เข้ารหัส JPEG. JFIF คือการยอมให้บรรจุข้อมูลที่เป็นรหัส JPEG เพื่อเปลี่ยนแปลงระหว่าง system ที่เข้ากันไม่ได้กับ applications

file JFIF คือบรรดาข้อมูลเบื้องต้นของ JPEG ซึ่งมีข้อจำกัดอยู่เล็กน้อย และมีเครื่องบ่งชี้อยู่บ้าง เพื่อที่จะให้เข้าใจรูปแบบของ JFIF เราจำเป็นจะต้องเข้าใจ JPEG โดยให้ดูที่ JPEG FAQ (Frequently Asked Questions) ซึ่งจะรวมถึง CD-ROM และสามารถใช้ได้กับ Internet อีกด้วย

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

File Organiaztion

ข้อมูลของ JPEG และ JFIF จะมี byte จำนวนมาก การเก็บรักษาค่าของ word จะใช้ 16 bit เสมอและในรูปแบบที่ใหญ่ ข้อมูล JPEG โดยทั่ว ๆ ไปจะเก็บรักษาเสมือนกับเป็น stream of blocks และแต่ละ block จะบ่งชี้ได้จาก marker value

ใน 2 byte แรกของทุก ๆ JPEG stream คือ Start Of Image (SOI) marker value FFh D8h ใน file ที่เป็น JFIF คือ JFIF APP0 (Application) marker และจะตามมาทันทีด้วย SOI ซึ่งบรรจุ

marker code value FFh E0h และ characters JFIF ใน marker data ซึ่งจะอธิบายในหัวข้อถัดไป นอกจากนี้ในส่วนของ JFIF marker มันอาจจะมีทางเลือกได้ 1 ทาง หรือมากกว่าที่จะขยาย JFIF marker segments นั่นคือการตามมาด้วย actual image data

File Details

ถึงแม้ว่า file JFIF จะไม่มี header ที่แน่ชัด แต่ SOI และ JFIF APP0 markers จะกระทำร่วมกันเสมือนกับเป็น header ใน marker segment structure ต่อไปนี้

```
typedef struct _JFIFHeader
{
  BYTE SOI [2] ;          /* 00h Start of Image Marker */
  BYTE APP0 [2] ;         /* 02h Application Use Marker */
  BYTE Length [2] ;       /* 04h Length of APP0 Field */
  BYTE Identifier [5] ;   /* 06h "JFIF" (Zero terminated) Id String */
  BYTE Version [2] ;      /* 07h JFIF Format Revision */
  BYTE Units ;            /* 09h Units used for Resolution */
  BYTE Xdensity [2] ;     /* 0Ah Horizontal Resolution */
  BYTE Ydensity [2] ;     /* 0Ch Vertical Resolution */
  BYTE XThumbnail ;       /* 0Eh Horizontal Pixel Count */
  BYTE YThumbnail ;       /* 0Fh Vertical Pixel Count */
} JFIFHEAD ;
```

SOI จะเริ่มด้วย image marker และมี marker code value ว่า FFh D8h เสมอ

APP0 คือ Application marker และจะมี marker code value ว่า FFh E0h เสมอ

Length คือ ขนาดของ JFIF (APP0) marker segment โดยจะรวมถึงขนาดของความยาวของ field ของมันเอง และ thumbnail data บรรจุใน APP0 segment เพราะว่าค่าของ Length จะเท่ากับ 16+3*XThumbnail*YThumbnail

Identifier จะบรรจุค่า 4Ah 46h 49h 46h 00h (JFIF) และจะใช้เพื่อระบุบรรดา code เสมือนกับเป็นลักษณะเฉพาะของ JFIF

Version จะบ่งชี้ลักษณะเฉพาะของ version ของ JFIF โดย byte แรกจะบรรจุตัวเลขหลักที่มีการแก้ไขใหม่ และใน byte ที่สองจะบรรจุตัวเลขรองที่มีการแก้ไขใหม่ สำหรับ version 1.02 ค่าของ Version field จะเท่ากับ 01h 02h: file เก่าจะบรรจุ 01h 00h หรือ 01h 01h

Units, Xdensity และ Ydensity จะบ่งชี้ถึงหน่วยของการวัด ซึ่งใช้อธิบาย image resolution. Units อาจจะเป็น 01h สำหรับ dot per inch, 02h สำหรับ dots per centimeter หรือ 00h for none (ใช้การวัดเสมือนกับเป็นเกณฑ์อัตราส่วนของ pixel) Xdensity และ Ydensity คือ resolution ทางด้านแนวนอนและแนวตั้งของ image data ตามลำดับ ถ้าค่าของ Units field คือ 00h, Xdensity และ Ydensity field จะบรรจุค่าอัตราส่วน pixel (Xdensity : Ydensity) โดยมันค่อนข้างจะมากกว่า image resolution เพราะว่า non-square pixels คือการกีดกันไม่ให้เคลื่อนย้ายได้ ค่าโดยปกติของ Xdensity และ Ydensity จะเท่ากับ 1 เมื่อ Units value เท่ากับ 0

XThumbnail และ YThumbnail จะให้มิติของ thumbnail image ซึ่งจะรวมถึง JFIF APP0 marker ถ้า thumbnail image ไม่ได้รวมอยู่ใน marker, field นี้จะเท่ากับ 0 thumbnail image จะเล็กกว่าการแสดงค่าของ image stored ใน main JPEG data stream (บางคนอาจจะเรียกมันว่าเป็น icon ของ preview image) ใน thumbnail data เองจะประกอบด้วยแถวของค่า pixel ของ XThumbnail*YThumbnail ซึ่งในแต่ละค่า pixel จะมีอยู่ 3 byte และบรรจุค่า RGB อีก 24 bit (เก็บอยู่ในคำสั่ง R,G,B) การไม่ลดขนาดคือการแสดงถึง thumbnail image

การเก็บ thumbnail image ใน JFIF APP0 marker จะมีการกีดกันเกิดขึ้น ทั้ง ๆ ที่มันยังคง support การย้อนกลับที่สอดคล้องกัน Version 1.2 ของ JFIF จะบ่งชี้ถึง extension marker ที่จะยอมให้ thumbnail image เข้าไปถูกเก็บรักษาแยกจาก identification marker วิธีนี้จะเป็นการยืดหยุ่นมาก เนื่องจากหลาย ๆ รูปแบบของ thumbnail จะยอมรับ และเพราะหลาย ๆ thumbnail images ของขนาดต่าง ๆ กัน จะสามารถรวมกันอยู่ใน file ได้ Version 1.02 ยอมให้ color-mapped thumbnails (1 byte ต่อ pixel บวกด้วย 256-entry colormap) และ JPEG จะลดขนาด thumbnails นอกจากนี้ 24-bit RGB thumbnail format ในบางกรณี thumbnail image จะจำกัดไว้ให้น้อยกว่า 64 K byte เพราะมันจะพอดีใน APP0 marker

JFIF marker segment ต่อไปนี้ อาจจะเป็นทางเลือก 1 หรือมากกว่าของการขยาย JFIF marker segment การขยาย segments จะใช้สำหรับเพิ่มเติมการเก็บข้อมูลและค้นหา JFIF version 1.02 และต่อจากนั้น โครงสร้างของการขยาย segment สามารถแสดงได้ดังนี้

```
typedef struct _JFIFExtension
{
    BYTE APP0 [2] ;          /* 00h Application Use Marker */
    BYTE Length [2] ;        /* 02h Length of APP0 Field */
    BYTE Identifier [5] ;    /* 04h "JFXX" (zero terminated) Id String */
    BYTE ExtensionCode ;     /* 09h Extension ID Code */

} JFIFEXTENSION
```

APP0 บรรจุค่าของ FFh E0h

Length จะเก็บความยาวใน bytes ของ extension segment

Identifier จะบรรจะค่าเท่ากับ 4Ah 46h 58h 58h 00h (JFXX)

Extension Code จะระบุชนิดของข้อมูลใน extension marker stores สำหรับ version 1.02, extension code เพียงอย่างเดียวก็สามารถบอกได้ว่าคือ 10h (JPEG ใช้ในการเข้ารหัส thumbnail), 11h (1 byte pixels และ palette ใช้ในการเก็บ thumbnail) และ 13h (3 byte RGB pixels ใช้ในการเก็บ thumbnail)

การขยายข้อมูลจะทำตาม extension segment information และจะเปลี่ยนแปลงขนาดจนเพียงพอ จะขึ้นอยู่กับ ExtensionCode value (อ้างถึงลักษณะเฉพาะในปัจจุบันของ JFIF สำหรับรูปแบบที่เป็นไปได้ของ extension marker segment)

ตัวถอดรหัส JFIF จะต้องตระเตรียมไว้เพื่อไม่ให้ลืม extension marker และ APPn segments. Application เฉพาะ APPn markers จะไม่ถูกยอมรับโดยการถอดรหัส JPEG ซึ่งสามารถจะข้ามไปได้ง่ายมากโดยการใช้ field ความยาวของข้อมูลของ marker

JFIF marker เป็นสิ่งจำเป็นที่จะรับประกันว่า file สอดคล้องกันกับแบบแผนของ JFIF การถอดรหัส JFIF โดยส่วนมาก โดยเฉพาะเกี่ยวกับ JFIF marker segment จะถือเสมือนกับเป็นทาง

เลือก และจะมีประสิทธิภาพในการอ่านบรรดาข้อมูลดิบของ JPEG ที่เดียว และจะยอมรับ color space และตัวอย่างของ JFIF (มันจะมี file ออกไปเป็นจำนวนมาก) เพราะรูปแบบปกติของ JFIF เท่านั้น ที่จะแสดงในพื้นที่นี้ได้) การถอดรหัสที่เข้มแข็งจะรักษา file JFIF เสมือนกับเป็น stream of block ยังไม่มีการสันนิษฐานเกี่ยวกับคำสั่งในการจัด block ถัดไป โดยมาตรฐานของ JPEG การกระทำนี้จะทำให้สามารถอ่านความแปรปรวนของ file JFIF ที่ไม่เป็นมาตรฐานและไม่ถูกต้องได้ เช่น COM marker ได้ inserted ระหว่าง soi และ JFIF APP0 markers เราจะแนะนำว่าการถอดรหัสควรจะยอมรับ file JFIF ด้วยการรู้ให้แน่ชัดเกี่ยวกับตัวเลขที่เป็น major version ถ้าตัวเลขที่เป็น minor version ใหม่กว่ามันจะรู้ถึงการถอดรหัส

ค่าจริงของข้อมูล JPEG ใน JFIF file ตาม APP0 ทั้งหมด และเกาะติดอยู่กับรูปแบบที่ถูกกำหนดในเอกสาร JPEG การประมวลผลเส้นหลักของ JPEG คือ แบบที่ชี้แนะของการเข้ารหัสรูปภาพ เพื่อใช้ใน JFIF files ซึ่งเป็นการทำให้มั่นในใจความสามารถสูงสุดของ JFIF file สำหรับ interchange ข้อมูล

เพื่อเป็นการบ่งชี้ JFIF file หรือ data stream ควรพิจารณาค่า FFh D8h FFh ซึ่งเป็นการบ่งชี้ถึง SOI marker ที่ตามด้วย marker อื่น ๆ ใน JIJF file ที่ถูกต้อง byte ถัดไปจะเป็น E0h แสดงเป็นค่า JFIF APP0 marker segment อย่างไรก็ตาม มันก็เป็นได้ว่า marker segment อันหนึ่งหรือมากกว่าเขียนไม่ถูกต้อง ระหว่าง SOI และ JFIF APP0 marker จากที่กล่าวถึงในบทก่อน การถอดรหัสควรจะยกเว้นการอ่าน file

2 byte ถัดไป (APP0 segment length) จะมีการเปลี่ยนแปลงค่าเป็น 00h 10h และค่าเหล่านี้ถูกตามด้วยค่าอีก 5 byte คือ 44h 46h 49h 46h 00h (JFIF) ถ้าค่าพวกนี้ถูกพบแล้ว SOI marker (FFh D8h) จะทำเครื่องหมายเริ่มต้นของ JFIF data stream ถ้าค่า FFh D8h FFh เท่านั้นที่ถูกพบแต่ไม่มีข้อมูลเหลืออยู่ ดังนั้น บรรดาข้อมูลดิบของ JPEG จะถูกค้นพบ Data stream ของ JFIF และ JPEG ทั้งหมดจะจบลงด้วย End Of Image (EOI) marker value คือ FFh D9h

Microsoft Window Bitmap (BMP)

BMP เป็นฟอร์แมตของไฟล์กราฟฟิคที่ใช้กันแพร่หลายบน Microsoft Windows และยังใช้บนระบบปฏิบัติการ OS/2 ด้วย สำหรับบน Microsoft Windows นั้น มีฟอร์แมต BMP ใช้กันอยู่ 2

เวอร์ชัน ขึ้นอยู่กับเวอร์ชันของ Microsoft Windows คือ Microsoft Windows เวอร์ชัน 1, X/2.X (BMP เวอร์ชัน 1) และ Microsoft Windows 3.X (BMP เวอร์ชัน 3) BMP เวอร์ชัน 1 เป็นฟอร์แมตที่ขึ้นกับอุปกรณ์หรือเรียกว่า device-dependent ซึ่งเวอร์ชันนี้เป็นฟอร์แมตง่าย ๆ ไม่มีการเก็บ palette และไม่มีการบีบอัดข้อมูล เหมาะสำหรับกราฟฟิคการ์ดรุ่นเก่า ๆ เช่น CGA, EGA และ Hercules ส่วน BMP เวอร์ชัน 3 เป็นเวอร์ชันที่ถูกพัฒนาขึ้นให้เป็นฟอร์แมตที่ไม่ขึ้นกับอุปกรณ์ หรือเรียกว่า device-independent ซึ่งทำให้มีความยืดหยุ่นในการใช้งานสูง ฟอร์แมตนี้จะมีส่วนที่เก็บรายละเอียดของรูปภาพมากขึ้น มีการเก็บ palette สี และสามารถบีบอัดข้อมูลรูปภาพโดยใช้รูปแบบ RLE ได้อีกด้วย BMP เวอร์ชัน 3 นี้ เป็นเวอร์ชันที่นิยมใช้กันอยู่ในปัจจุบัน ดังนั้น จึงขอกล่าวถึงรายละเอียดเฉพาะ BMP เวอร์ชัน 3 เท่านั้น

BMP เวอร์ชัน 3 เป็นฟอร์แมตรูปภาพกราฟฟิกที่มีการเก็บข้อมูลรูปภาพเป็นแบบ bitmap นั่นคือ แต่ละจุด (pixel) บนรูปภาพจะถูกกำหนดหรือถูกแทนด้วยข้อมูลเป็น bit จะเป็นกี่ bit นั้นจะขึ้นกับจำนวนสีที่ใช้ และแต่ละจุดนั้นจะมีค่า bit เป็นอะไรนั้น จะขึ้นกับว่าจุดนั้นถูกแทนด้วยสีอะไร สำหรับจำนวน bit ที่ใช้กำหนดจำนวนสี มีได้ตั้งแต่ 1 bit, 4 bit, 8 bit และ 24 bit แสดงว่ารูปภาพสามารถมีจำนวนสีได้ตั้งแต่ 2 สี, 16 สี, 256 สี ไปจนถึง 16 ล้านสี ยิ่งจำนวนสีมาก ยิ่งต้องใช้จำนวน bit มาก ทำให้ขนาดของไฟล์ BMP ค่อนข้างมีขนาดใหญ่

วิธีการเก็บข้อมูลของจุดในรูปภาพ จะมี 2 วิธี ขึ้นกับจำนวน bit ที่ใช้ในการเก็บแต่ละจุด คือ

ศูนย์วิทยทรัพยากร

<u>วิธีที่ 1</u> สำหรับ BMP 1 bit, 4 bit และ 8 bit จะเก็บข้อมูลแต่ละ bit ซึ่งแทนตำแหน่งของสีที่ใช้ ซึ่งชี้ไปยังตารางสี ซึ่งเก็บสีที่ใช้ทั้งหมดในรูปภาพ โดยข้อมูลตารางสีจะถูกเก็บไว้ในส่วนหัวของไฟล์ BMP ค่าของ bit ที่ถูกกำหนดให้แต่ละจุด คือ ตัวชี้บนตารางสี ตัวอย่างเช่น ถ้าเป็น BMP 4 bit จุดแต่ละจุดบนภาพจะถูกกำหนดค่าโดยใช้ 4 bit สมมติว่า จุดหนึ่งมีค่า 0101 จะหมายความว่า จุดนี้จะมีสีอะไรให้ดูว่าค่า 0101 หรือ 5 ในฐานสิบ นั้นชี้ไปที่สีอะไรในตารางสี

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิธีที่ 2 สำหรับ BMP ที่เป็น 24 bit จะเก็บข้อมูลแต่ละ bit เป็นค่าของสีนั้นเลย นั่นคือ จุด 1 จุด บนภาพจะใช้เนื้อที่ 3 byte หรือ 24 bit แต่ละ byte จะแทนสีน้ำเงิน เขียว แดง ตามลำดับ แต่ละ byte สามารถมีค่าได้ถึง 256 เฉดสี ทำให้ BMP 24 bit สามารถมีสีได้ถึงจำนวน 16 ล้านกว่าสี เนื่องจากว่าข้อมูลแต่ละ byte ของจุดแทนค่าสีเลย ทำให้ BMP 24 bit จึงไม่มีตารางสีเก็บอยู่ดังเช่นแบบ 1, 4, 8 bit

โครงสร้างของฟอร์แมต BMP เวอร์ชัน 3

แบ่งเป็น 3 ส่วน

- Bitmap Header
- Bitmap Information Header
- Image Data

Bitmap Header

- Image File Type มีขนาด 2 byte เป็นส่วนที่บอกว่าไฟล์นี้เป็นฟอร์แมตแบบ BMP โดยจะมีค่าเป็น 4D42h หรือ BM เสมอ
- File Size มีขนาด 4 byte เป็นที่เก็บขนาดของไฟล์ทั้งหมด
- Reserved 1 มีขนาด 2 byte จะมีค่าเป็น 0 เสมอ
- Reserved 2 มีขนาด 2 byte จะมีค่าเป็น 0 เสมอ
- Image Data Offset ขนาด 4 byte จะบอกว่าข้อมูลภาพจะเริ่มต้นที่ Offset ที่เท่าไหร่ของไฟล์

Bitmap Information Header

- Header Size ขนาด 2 byte บอกขนาดของส่วน Bitmap Information Header สำหรับ Microsoft Window 3.X จะมีค่าเท่ากับ 40
- Image Width ขนาด 4 byte ความกว้างของรูปภาพหน่วยเป็นจุด
- Image Height ขนาด 4 byte ความสูงของรูปภาพเป็นจุด
- Number Of Image Planes ขนาด 2 byte จำนวนของ plane ปกติเป็น 1
- Bits Per Pixel ขนาด 2 byte เป็นจำนวน bit ต่อจุด อาจเป็น 1, 4, 8 หรือ 24

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

- Compression Method ขนาด 4 byte วิธีที่ใช้ในการบีบอัดข้อมูล

ถ้าเป็น 0 ไม่มีการบีบอัดข้อมูล

ถ้าเป็น 1 มีการบีบอัดข้อมูลแบบ 8 bit RLE

ถ้าเป็น 2 มีการบีบอัดข้อมูลแบบ 4 bit RLE

- Size Of Bitmap ขนาด 4 byte บอกขนาดของข้อมูลรูปภาพที่ถูกบีบอัด ถ้ารูปภาพไม่ถูกบีบอัดข้อมูลจะมีค่าเป็น 00h

- Horz Resolution ขนาด 4 byte บอกความละเอียดของภาพในแนวนอนเป็น pixel per meter

- Vert Resolution ขนาด 4 byte บอกความละเอียดของภาพในแนวตั้งเป็น pixel per meter

- Num Colors Used ขนาด 4 byte บอกจำนวนสีทั้งหมดที่ใช้ในภาพ

- Num Significant Colors ขนาด 4 byte บอกจำนวนสีที่สำคัญใน palette

ตารางสี

ตารางสีจะมีอยู่ในไฟล์ BMP ที่เป็น 1, 4 หรือ 8 bit ต่อจุดเท่านั้น โดยตารางสีจะถูกเก็บอยู่ต่อจากส่วนที่เรียกว่า Bitmap Information Header และอยู่ก่อนส่วนที่เป็นข้อมูลภาพ หรือที่เรียกว่า Image Data

โครงสร้างของตารางสี ประกอบด้วย

- Blue ขนาด 1 byte เก็บ 8 bit เป็นส่วนประกอบของสีน้ำเงิน

- Green ขนาด 1 byte เก็บ 8 bit เป็นส่วนประกอบของสีเขียว

- Red ขนาด 1 byte เก็บ 8 bit เป็นส่วนประกอบของสีแดง

- Reserved ขนาด 1 byte ปกติจะเป็น 0

ขนาดของตารางสีสามารถดูได้จาก จำนวนสีทั้งหมดที่ใช้ในรูปภาพ ซึ่งจะบอกไว้ที่ NumColorsUsed ที่อยู่ในส่วน Bitmap Information Header

โครงสร้างทั้ง 4 ส่วน ของตารางสีนั้น จะแทนค่าเป็นสี 1 สี ที่เรามองเห็นกัน ข้อมูลรูปภาพที่เก็บในไฟล์ BMP แบบ 1, 4 หรือ 8 bit นั้น ก็คือตัวชี้ซึ่งที่มายังตำแหน่งที่เก็บค่าสีเหล่านี้นั่นเอง เมื่อเราอ่านค่าตำแหน่งสีได้ก็จะสามารถโยงมายังตารางสีได้ ทำให้เราทราบว่าจุดที่เราอ่านขึ้นมานั้นเป็นสีอะไร

Image Data

ไฟล์ BMP ไม่ว่าจะเป็นแบบ 1, 4, 8 หรือ 24 bit ต่อจุด ก็จะต้องมีส่วนที่เป็นข้อมูลรูปภาพ หรือ Image Data ด้วยกันทั้งนั้น แต่ความหมายของ Image Data ในไฟล์ BMP แบบ 1, 4, 8 bit และปบบ 24 bit ต่อจุดจะต่างกันคือ แบบแรกคือไฟล์ BMP ที่เก็บ Image Data เป็น 1, 4 หรือ 8 bit ต่อจุดนั้น ค่าที่เก็บคือค่าตัวชี้ซึ่งชี้ไปยังตารางสีที่ถูกเก็บอยู่ในไฟล์ไม่ได้หมายถึงค่าสีจริง ๆ แต่ไฟล์ BMP ที่เก็บ Image Data เป็น 24 bit ต่อจุดนั้นค่าที่เก็บคือค่าของสีของจุดนั้นจริง ๆ ไม่ใช่ตัวชี้ที่ชี้ไปที่ตารางสี

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียน

นายมงคล ภิญโญสโมสร เกิดวันที่ 5 พฤษภาคม 2508 เขตพระโขนง จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมอีเล็กทรอนิคส์ สถาบันเทคโนโลยีราชมงคล เมื่อปี พ.ศ.2532 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปี พ.ศ.2537 ปัจจุบันทำงานที่ บริษัท การบินไทย จำกัด (มหาชน) ตำแหน่ง วิศวกรคอมพิวเตอร์ ฝ่ายบริการข้อมูล

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย