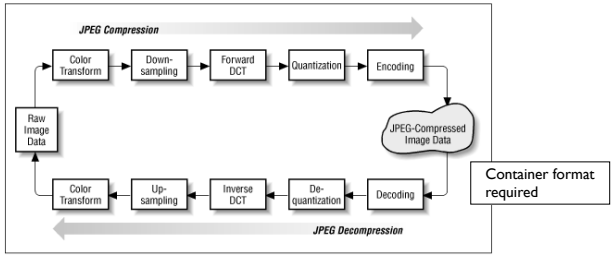
CS4405

JPEG File Format

JPEG Lifecycle



JFIF

- JPEG File Interchange Format is a minimal file format which enables JPEG bitstreams to be exchanged between a wide variety of platforms and applications
- A JFIF file consists of JPEG data with a header providing information missing from the JPEG stream
  - Version number, horizontal and vertical pixel density, pixel aspect ratio and an optional thumbnail

JFIF Header

Field	Size (bytes)	Description
APP0 marker	2	Always equals 0xFFE0
Length	2	Length of segment excluding APP0 marker
Identifier	5	Always equals "JFIF" (with zero following) (0x4A46494600)
Version	2	First byte is major version (currently 0x01), Second byte is minor version (currently 0x02)

test1.jpg

```
  0 FFD8FFE0 00104A46 49460001 02000001 00010000 FFDB0043 ˙ÿ˙‡  JFIF          ˙€ C
 24 00080606 07060508 07070709 09080A0C 140D0C0B 0B0C1912
```

```
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;
```

JFIF

- A JFIF file
  - Starts with the value oxFFD8 (SOI – start of image)
  - Ends with the value oxFFD9 (EOI – end of image)
- A value in the form oxFF*XX* is a marker
  - oxFF · Marker Number(1 byte) · Data size(2 bytes) · Data(n bytes)
  - Note: oxFFD8 has no data following it

JFIF Header

```
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;
```

```
[Record name]     [size]   [description]
-----------------------------------------
Identifier        5 bytes  ("JFIF\000" = 0x4a46494600)
MajorVersion      1 byte   major version (e.g. 0x01)
MinorVersion      1 byte   minor version (e.g. 0x01 or 0x02)
Units             1 byte   units (0: densities give aspect ratio
                                  1: density values are dots per inch
                                  2: density values are dots per cm)
Xdensity          2 bytes  horizontal pixel density
Ydensity          2 bytes  vertical pixel density
Xthumbnail        1 byte   thumbnail horizontal pixel count
Ythumbnail        1 byte   thumbnail vertical pixel count
ThumbnailData    3n bytes  thumbnail image
```

JFIF Extensions

```
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;
```

- JFIF can contain
  - A thumbnail image compressed using JPEG
  - A thumbnail image stored using one byte per pixel and a colour palette
  - A thumbnail stored using three bytes per pixel
- To specify a pixel aspect ratio
  - Set zero in the `units` field
  - `Xdensity` and `Ydensity` can then be programmed for the desired aspect ratio
  - Example: `Xdensity` = 1, `Ydensity` = 1 will program a 1:1 aspect ratio

JFIF Header Example

```
File Size                       : 3.0 kB
File Type                       : JPEG
MIME Type                       : image/jpeg

JFIF Version                    : 1.02
Resolution Unit                 : None
X Resolution                    : 1
Y Resolution                    : 1
Image Width                     : 96
Image Height                    : 160
Encoding Process                : Baseline DCT, Huffman coding
Bits Per Sample                 : 8
Color Components                : 1
Image Size                      : 96x160
```

```
JPEG code stream dump

marker 0xffd8 SOI at offset 0     (start of image)
marker 0xffe0 APP0 at offset 2    (application data section  0)
marker 0xffdb DQT at offset 20    (define quantization tables)
marker 0xffc0 SOF0 at offset 89   (start of frame [baseline jpeg])
marker 0xffc4 DHT at offset 102   (define huffman tables)
marker 0xffc4 DHT at offset 135   (define huffman tables)
marker 0xffda SOS at offset 318   (start of scan)
marker 0xffd9 EOC at offset 3052  (end of codestream)
```

http://svn.xiph.org/experimental/giles/jpegdump.c

JPEG Example

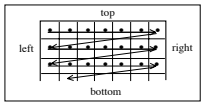
```
*** Marker: DQT (xFFDB) ***
  Define a Quantization Table.
  OFFSET: 0x0000AF5E
  Table length = 67
  ---
  Precision=8 bits
  Destination ID=0 (Luminance)
    DQT, Row #0:   3   2   2   3   5   8  10  12
    DQT, Row #1:   2   2   3   4   5  12  12  11
    DQT, Row #2:   3   3   3   5   8  11  14  11
    DQT, Row #3:   3   3   4   6  10  17  16  12
    DQT, Row #4:   4   4   7  11  14  22  21  15
    DQT, Row #5:   5   7  11  13  16  21  23  18
    DQT, Row #6:  10  13  16  17  21  24  24  20
    DQT, Row #7:  14  18  19  20  22  20  21  20
    Approx quality factor = 90.06
           (scaling=19.88 variance=1.14)
```

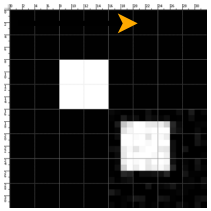
JFIF Requirements

- The color space to be used is YCbCr as defined by CCIR 601 (256 levels)
  - The RGB components calculated by linear conversion from YCbCr is not gamma corrected (gamma = 1.0)
  - If only one component is used that component is Y

JFIF Block Order

- In JFIF files the image orientation is always top-down
  - This means that the first image samples encoded in a JFIF file are located in the upper left hand corner of the image and encoding proceeds from left to right and top to bottom

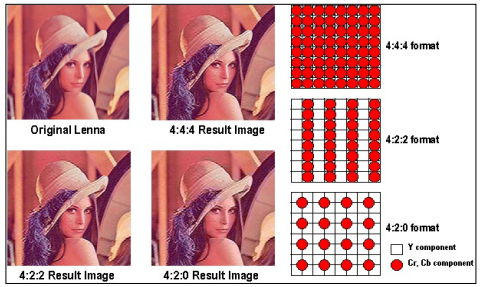




JFIF Sub-sampling

- The spatial positioning of pixel samples within components relative to the samples of other components is necessary for proper image presentation
- In JFIF files the position of the pixels in subsampled components are defined with respect to the highest resolution component
  - The Cb and Cr samples are sited interstitially halfway between alternate Y samples

Sub-sampling Example



Tables

- A JPEG file contains up to four Huffman tables
  - These define the mapping between these variable-length codes and the code values
  - Creating these tables generally involves counting how frequently DCT code word appears
  - Most JPEG encoders use the Huffman tables defined in the JPEG standard

JPEG Example

```
*** Marker: DHT (Define Huffman Table) (xFFC4) ***
  OFFSET: 0x0000AFFB
  Huffman table length = 31
  ----
  Destination ID = 0
  Class = 0 (DC / Lossless Table)
    Codes of length 01 bits (000 total):
    Codes of length 02 bits (001 total): 00
    Codes of length 03 bits (005 total): 01 02 03 04 05
    Codes of length 04 bits (001 total): 06
    Codes of length 05 bits (001 total): 07
    Codes of length 06 bits (001 total): 08
    Codes of length 07 bits (001 total): 09
    Codes of length 08 bits (001 total): 0A
    Codes of length 09 bits (001 total): 0B
    Codes of length 10 bits (000 total):
    Codes of length 11 bits (000 total):
    Codes of length 12 bits (000 total):
    Codes of length 13 bits (000 total):
    Codes of length 14 bits (000 total):
    Codes of length 15 bits (000 total):
    Codes of length 16 bits (000 total):
    Total number of codes: 012
```

JPEG Example

```
Huffman code histogram stats:
   Huffman Table: (Dest ID: 0, Class: DC)
      # codes of length 01 bits:       0 (  0%)
      # codes of length 02 bits:     427 (  6%)
      # codes of length 03 bits:    5650 ( 74%)
      # codes of length 04 bits:     879 ( 12%)
      # codes of length 05 bits:     433 (  6%)
      # codes of length 06 bits:     177 (  2%)
      # codes of length 07 bits:      34 (  0%)
      # codes of length 08 bits:       0 (  0%)
      # codes of length 09 bits:       0 (  0%)
      # codes of length 10 bits:       0 (  0%)
      # codes of length 11 bits:       0 (  0%)
      # codes of length 12 bits:       0 (  0%)
      # codes of length 13 bits:       0 (  0%)
      # codes of length 14 bits:       0 (  0%)
      # codes of length 15 bits:       0 (  0%)
      # codes of length 16 bits:       0 (  0%)
```

JPEG Example

```
Huffman Table: (Dest ID: 0, Class: AC)
      # codes of length 01 bits:       0 (  0%)
      # codes of length 02 bits:   56562 ( 46%)
      # codes of length 03 bits:   13583 ( 11%)
      # codes of length 04 bits:   25854 ( 21%)
      # codes of length 05 bits:   11856 ( 10%)
      # codes of length 06 bits:    4644 (  4%)
      # codes of length 07 bits:    4595 (  4%)
      # codes of length 08 bits:    1696 (  1%)
      # codes of length 09 bits:    1741 (  1%)
      # codes of length 10 bits:     636 (  1%)
      # codes of length 11 bits:     305 (  0%)
      # codes of length 12 bits:      72 (  0%)
      # codes of length 13 bits:       0 (  0%)
      # codes of length 14 bits:       0 (  0%)
      # codes of length 15 bits:       5 (  0%)
      # codes of length 16 bits:     107 (  0%)
```