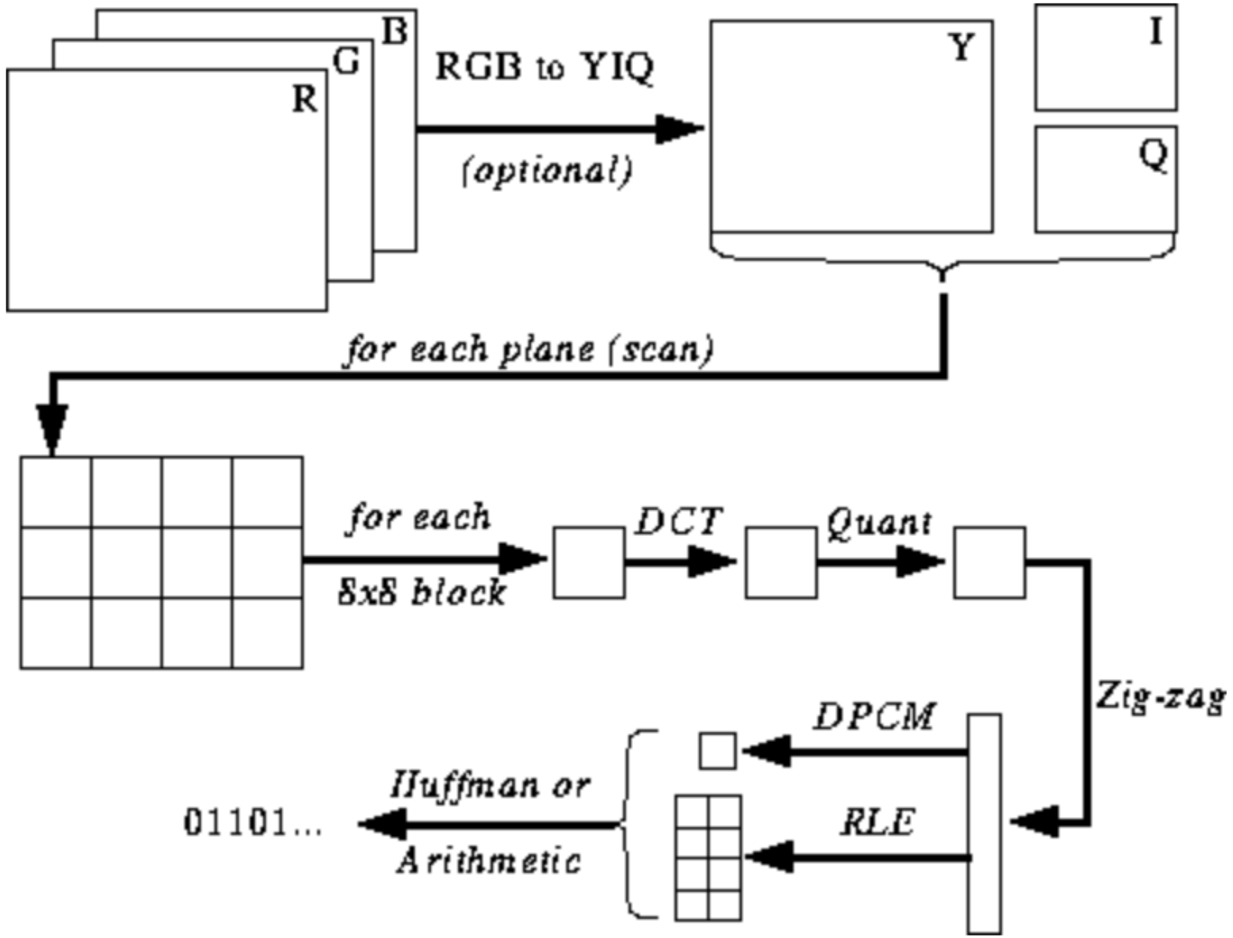
JPEG image format

Markers

- Markers can be seen as bookmarks of various information about a particular segment.
- **FF D8 :** tells us about the start of the image.
- **FF D9 :** tells us about the end of the image.
- Both ff d8 and ff d9 are two bytes long.
- Otherthen FFD0 to FFD9 and FF01 is followed by a length specifier that represents the length of the marker segment.

- Each markers in jpeg are 4 hex digits. For example ffd8.
- One hex digit equals to 4 bits or 1/2 byte.
- The whole 4 hex digits will equal to 2 bytes

- The start of scan section is immediately scaned for image data.
- The length of the this image data is different for very image.
- This scan continues till the end of scan.



https://yasoob.me/posts/understanding-and-writing-jpeg-decoder-in-python/#further-reading

Jpeg compression

- Jpeg stands for “Joint Photographic Expert Group”
- The major steps of encoding jpeg image are:
  - DCT
  - Quantization
  - Zigzag scan
  - DPCM on DC component
  - RLE on AC component
  - Entropy Coding

JPEG color space

- Images with three components rgb are encoded in YCbCr unless APP14 marker is present.
- In case of APP14 the color encoded is either rgb or YcbCr depending upon the application of data specified in the APP14.
- Other color encoding available in jpeg are greyscale and CMYK.

- Most jpeg algorithm uses luminance or chrominance encoding.

Discrete Cosine Transform & Quantization

- Jpeg converts an image into chunks of 8x8 blocks pixels.
- These chunks are called MCU’s or Minimum Coding Units.
- DCT transform converts discrete data points in combination of cosine waves.
- The dct is applied each component of pixel seperately.

- Till now there is no loss of information.
- The lossy part comes when quantization is performed.
- Quantization process takes couple of values falling in a specific range and turns them into a discrete value.
- By using quantization we tends to change high frequency value obtained in DCT turn to zero.
- This change is un-noticable to the human eye.

Example: 8x8 dct matrix

$$\begin{bmatrix} -415 & -33 & -58 & 35 & 58 & -51 & -15 & -12 \\ 5 & -34 & 49 & 18 & 27 & 1 & -5 & 3 \\ -46 & 14 & 80 & -35 & -50 & 19 & 7 & -18 \\ -53 & 21 & 34 & -20 & 2 & 34 & 36 & 12 \\ 9 & -2 & 9 & -5 & -32 & -15 & 45 & 37 \\ -8 & 15 & -16 & 7 & -8 & 11 & 4 & 7 \\ 19 & -28 & -2 & -26 & -2 & 7 & -44 & -21 \\ 18 & 25 & -12 & -44 & 35 & 48 & -37 & -3 \end{bmatrix}$$

8x8 quantization matrix

$$\begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

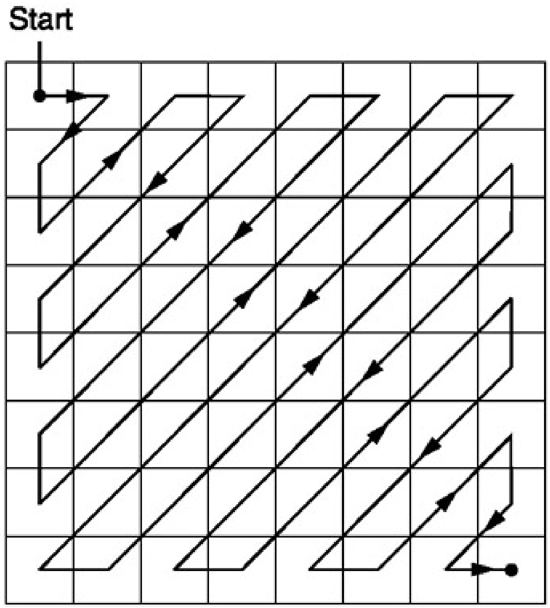
8x8 quantized matrix

$$\begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -3 & 4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -4 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- The first value is called the DC value and rest are called the AC value,
- If only take DC value and regenerate the image then we end up 1/8 resolution of original image.

Zig-zag

- After quantization the zig-zag converts the matrix into a 1-D.



Quantised matrix

$$\begin{bmatrix} 15 & 14 & 10 & 9 \\ 13 & 11 & 8 & 0 \\ 12 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The output of zig-zag encoding

[15 14 13 12 11 10 9 8 0 ... 0]

This zig-zag encoding is done the most significant the low frequency information is stored in the beginning.

Run length encoding

- It is used to compress repeated data.
- 10 10 10 10 10 10 10 changes to 7 10
- We are able to change 7 byte data to 2 byte data.

Delta encoding

- Delta encoding technique is used to represent a byte relative to the byte before it.
- “10 11 12 13 10 9” changes to “10 1 2 3 0 -1”

Huffman encoding

- It is a method for lossless compression of information.
- It allows us a sort of variable-length mapping.
- It takes some input data, maps the most frequent characters to the smaller bit patterns and least frequent characters to larger bit patterns, and finally organises the mapping into a binary tree.
- In jpeg the dct information using Huffman encoding.

- A jpeg contains up to 4 huffman tables and these are stored in “define Huffman table” section (starting with 0xffc4).
- The DCT coefficients are stored in 2 different Huffman tables.
- One table contains DC value and other stores the AC values.
- The DCT information for luminance and chrominance channel is stored 4 tables. 2 hoffman tables for DC values and 2 hoffman tables for AC values.

JPEG decoding

We can break down the decoding into a bunch of steps:

1. Extract the Huffman tables and decode the bits
2. Extract DCT coefficients by undoing the run-length and delta encodings
3. Use DCT coefficients to combine cosine waves and regenerate pixel values for each 8x8 block
4. Convert YCbCr to RGB for each pixel
5. Display the resulting RGB image

JPEG standard supports 4 compression formats:

- Baseline
- Extended Sequential
- Progressive
- Lossless