Discrete Cosine Transformation & Local Computation

M. Pouly

Department of Informatics
University of Fribourg, Switzerland

February 2007

JPEG Image Compression

ISO 10918-1 norm with 41 sub-file-formats. But only one is practically relevant.

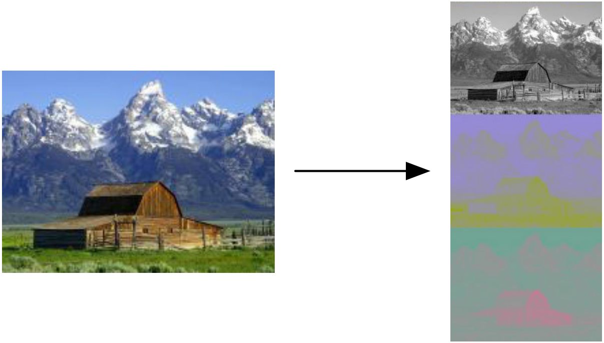
Procedure involves 5 steps:

1. Color Space Transformation: RGB → YCbCr model
2. Downsampling (lossy)
3. Block Splitting & Discrete Cosine Transformation (DCT)
4. Quantization (lossy)
5. Resorting & Entropy Encoding (Huffman)

top-down = JPEG encoding
bottom-up (with inverse DCT) = JPEG decoding

Color Space Transformation I

- Image exists in RGB (Red, Green, Blue).
- JPEG uses YCbCr (Y = Pixel brightness, Cb & Cr = Pixel chrominance split into blue and red).



Color Space Transformation II

This is a linear transformation (CCIR 601 norm):

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} \approx \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

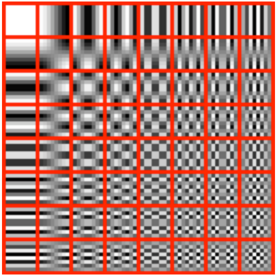
$Y, Cb, Cr \in \{0, \ldots, 255\}$.

Downsampling

- Humans can see considerably more fine detail in the brightness of an image (Y component) than in the color of an image (Cb and Cr components).
- Therefore, Cr and Cb are stored in a reduced resolution.
- Downsampling the Cb and Cr components saves 33% to 50% of the space taken by the image without drastically affecting perceptual image quality.

Block Splitting & DCT

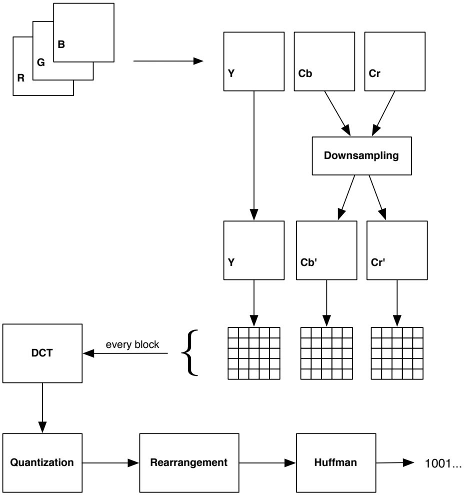
- The Y, Cr and Cb components are split into 8*8 blocks.
- We apply a 2-dimensional DCT on each block.
- Instead of 64 points, every block is represented by a linear combination of 64 blocks:



Quantization & Huffman Encoding

- Human eye is not very sensible to high frequency variations.
- We divide every component in the frequency domain (DCT coefficients) by a constant and round to the nearest integer:
  - many of the higher frequencies become zero
  - others become small positive and negative integers.
- Integers are rearranged and Huffman encoded.

Summary: The Big Picture



Discrete Cosine Transformation

$$\mathcal{F}(k) \quad = \quad 2\sum_{n=0}^{N-1} f(n) \cdot \cos\left[\frac{\pi k}{N} \cdot (n + \frac{1}{2})\right].$$

for $k = 0, \dots, N-1$.

- f: signal ($\mathbb{C}$-valued function of discrete time) $f : \mathbb{Z} \rightarrow \mathbb{C}$
- N: number of signal samples

For Image Compression:

- $f(n) \in \{0, \dots, 255\}$
  is the value of either Y, Cb or Cr at pixel n.

Why DCT and not DFT ?

$$\sum_{n=0}^{N-1} f(n) \cdot \cos\left[\frac{\pi k}{N} \cdot (n + \frac{1}{2})\right] \quad \text{versus} \quad \sum_{n=0}^{N-1} f(n) \cdot e^{-\frac{2\pi ink}{N}}$$

- Transforms images efficiently into a compressible form.
- Uses only cos, DFT uses sin and cos (hidden within $e^{-i\varphi}$).
- DCT coefficients are real, DFT coefficients are complex.

Towards a Projection Problem I

We use the definition

$$\cos(\varphi) = \frac{1}{2}\left(e^{i\varphi} + e^{-i\varphi}\right)$$

and obtain

$$\begin{aligned}
\mathcal{F}(k) &= 2\sum_{n=0}^{N-1} f(n)\cdot\frac{1}{2}\left(e^{\frac{i\pi k}{N}\cdot(n+\frac{1}{2})} + e^{-\frac{i\pi k}{N}\cdot(n+\frac{1}{2})}\right)\\
&= \sum_{n=0}^{N-1} f(n)\cdot\left(e^{\frac{i\pi kn}{N}}\cdot e^{\frac{i\pi k}{2N}} + e^{-\frac{i\pi kn}{N}}\cdot e^{-\frac{i\pi k}{2N}}\right)\\
&= \sum_{n=0}^{N-1} f(n)\cdot e^{\frac{i\pi kn}{N}}\cdot e^{\frac{i\pi k}{2N}} + \sum_{n=0}^{N-1} f(n)\cdot e^{-\frac{i\pi kn}{N}}\cdot e^{-\frac{i\pi k}{2N}}.
\end{aligned}$$

Towards a Projection Problem II

$$\sum_{n=0}^{N-1} f(n) \cdot e^{\frac{i\pi kn}{N}} \cdot e^{\frac{i\pi k}{2N}}$$

Where are the variables ?

Binary encoding: assume $N = 2^m$

$$n \;=\; \sum_{j=0}^{m-1} n_j \cdot 2^j \quad \text{and} \quad k \;=\; \sum_{l=0}^{m-1} k_l \cdot 2^l$$

Thus, $n = (n_0, \ldots, n_{m-1})$ and $k = (k_0, \ldots, k_{m-1})$

$$nk \;=\; \sum_{0 \leq j,l \leq m-1} n_j k_l \cdot 2^{j+l}$$

Towards a Projection Problem III

$$\sum_{n=0}^{N-1} f(n) \cdot e^{\frac{i\pi kn}{N}} \cdot e^{\frac{i\pi k}{2N}} =$$

$$\sum_{n_0,\ldots n_{m-1}} f(n_0, \ldots n_{m-1}) \cdot e^{\frac{i\pi \sum_{0 \leq j,l \leq m-1} n_j k_l \cdot 2^{j+l}}{2^m}} \cdot e^{\frac{i\pi \sum_{0 \leq l \leq m-1} k_l \cdot 2^l}{2^{m+1}}} =$$

$$\sum_{n_0,\ldots n_{m-1}} f(n_0, \ldots n_{m-1}) \prod_{0 \leq j,l \leq m-1} e^{\frac{i\pi n_j k_l \cdot 2^{j+l}}{2^m}} \prod_{0 \leq l \leq m-1} e^{\frac{i\pi k_l \cdot 2^l}{2^{m+1}}} =$$

$$\sum_{n_0,\ldots n_{m-1}} f(n_0, \ldots n_{m-1}) \prod_{0 \leq j,l \leq m-1} e^{\frac{i\pi n_j k_l}{2^{m-j-l}}} \prod_{0 \leq l \leq m-1} e^{\frac{i\pi k_l}{2^{m+1-l}}}.$$

DCT as Projection Problem

$$\sum_{n_0,\dots n_{m-1}} f(n_0,\dots n_{m-1}) \prod_{0\leq j,l\leq m-1} e^{\frac{i\pi n_j k_l}{2^{m-j-l}}} \prod_{0\leq l\leq m-1} e^{\frac{i\pi k_l}{2^{m+1-l}}} +$$

$$\sum_{n_0,\dots n_{m-1}} f(n_0,\dots n_{m-1}) \prod_{0\leq j,l\leq m-1} e^{-\frac{i\pi n_j k_l}{2^{m-j-l}}} \prod_{0\leq l\leq m-1} e^{-\frac{i\pi k_l}{2^{m+1-l}}}$$

This are two projection problems !

- Total domain: $\{k_0,\dots,k_{m-1},n_0,\dots,n_{m-1}\}$.
- All variables are binary.
- Query: $\{k_0,\dots,k_{m-1}\}$, factor domains: $\{n_0,\dots,n_{m-1}\}$, $\{n_j,k_l\}$.
- Factors: $f(n_0,\dots,n_{m-1})$ and $e^{\pm\frac{i\pi n_j k_l}{2^{m-j-l}}}$ and $e^{\pm\frac{i\pi k_l}{2^{m+1-l}}}$
- Factors are valuations induced by the Arithmetic Semiring.

Conclusion

- DCT can be written as two projection problem.
    - both queries are identical,
    - both knowledgebases have identical factor domains.
- Ergo: we can use the same covering join tree.