

# Aplikasi Deret Fourier: Transformasi Kosinus Diskrit (DCT) dalam Kompresi Citra Digital JPEG

Teosofi Hidayah Agung  
5002221132

21 Oktober 2025

## Ringkasan

Makalah ini menyajikan telaah mendalam mengenai aplikasi Deret Fourier, khususnya Transformasi Kosinus Diskrit (DCT), sebagai landasan matematis dari standar kompresi citra JPEG. Pembahasan dimulai dari evolusi konsep analisis frekuensi, mulai dari Deret Fourier untuk sinyal periodik, Transformasi Fourier untuk sinyal non-periodik, hingga adaptasi digital melalui Transformasi Fourier Diskrit (DFT). Analisis difokuskan pada keunggulan fundamental DCT, yaitu properti pemadatan energi (*energy compaction*) yang superior, yang memungkinkannya mengkonsentrasi informasi visual penting ke dalam sejumlah kecil koefisien. Selanjutnya, makalah ini membedah anatomi lengkap alur kerja kompresi JPEG, mulai dari pra-pemrosesan, kuantisasi, hingga pengkodean entropi. Analisis kinerja juga disajikan, menguraikan hubungan antara tingkat kompresi dan munculnya artefak visual seperti *blocking* dan *ringing*. Terakhir, JPEG ditempatkan dalam konteks evolusi teknologi dengan membandingkannya dengan standar yang lebih modern seperti JPEG 2000, WebP, dan AVIF untuk memberikan gambaran masa depan kompresi citra.

**Kata Kunci**—Deret Fourier, Transformasi Kosinus Diskrit, JPEG, Kompresi Citra, Artefak Kompresi.

## 1 Pendahuluan

Analisis Fourier adalah fondasi yang memungkinkan representasi sinyal dan citra dalam domain frekuensi; konsep ini menjelaskan mengapa transformasi seperti DCT mampu memusatkan energi dan mengurangi redundansi visual secara efektif. Sebagai pengantar singkat, pemahaman mengenai dekomposisi sinyal ke komponen frekuensi membantu menjelaskan perilaku *transform-domain compression*.

Standar JPEG, yang berlandaskan pada DCT, telah lama menjadi solusi praktis untuk kompresi citra *lossy* karena kombinasi efisiensi komputasi, pemadatan energi yang baik, dan dukungan perangkat keras/ekosistem yang luas.

Studi terhadap JPEG relevan secara praktis karena desainnya merepresentasikan trade-off nyata antara efisiensi dan kualitas visual: penggunaan blok  $8 \times 8$  dan kuantisasi memberikan penurunan bit-rate yang signifikan namun juga menimbulkan artefak seperti *blocking* dan *ringing* pada tingkat kompresi tinggi. Dengan menganalisis bagaimana DCT memusatkan energi, bagaimana kuantisasi memengaruhi distribusi koefisien frekuensi, serta bagaimana pilihan parameter (mis. ukuran blok, matriks kuantisasi, dan strategi kuantisasi) berdampak pada persepsi visual, kita memperoleh landasan untuk menilai dan menyesuaikan algoritme kompresi pada situasi

nyata — dari penyimpanan hingga transmisi dan pemrosesan citra. Oleh karena itu, memahami aspek-aspek praktis ini membantu jembatani teori transformasi frekuensi dengan keputusan implementasi yang menentukan kualitas akhir citra.

## 2 Formulasi Deret Fourier dan Transformasi Fourier

### 2.1 Esensi Deret Fourier

Setiap fungsi periodik dapat diuraikan sebagai deret tak hingga dari jumlahan fungsi sinus dan kosinus dengan berbagai frekuensi dan amplitudo [1]. Deret Fourier untuk fungsi periodik  $f(t)$  dengan periode  $2L$  dinyatakan sebagai:

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left( a_n \cos\left(\frac{n\pi t}{L}\right) + b_n \sin\left(\frac{n\pi t}{L}\right) \right) \quad (1)$$

di mana koefisien  $a_0$ ,  $a_n$ , dan  $b_n$  dihitung menggunakan integral berikut [2]:

$$a_0 = \frac{1}{L} \int_{-L}^{L} f(t) dt \quad (2)$$

$$a_n = \frac{1}{L} \int_{-L}^{L} f(t) \cos\left(\frac{n\pi t}{L}\right) dt \quad (3)$$

$$b_n = \frac{1}{L} \int_{-L}^{L} f(t) \sin\left(\frac{n\pi t}{L}\right) dt \quad (4)$$

Dekomposisi ini adalah transformasi dari domain waktu ke domain frekuensi. Sifat simetri fungsi asli berdampak langsung pada koefisiennya; fungsi genap hanya menghasilkan suku kosinus, sedangkan fungsi ganjil hanya menghasilkan suku sinus [1].

### 2.2 Transformasi Fourier untuk Sinyal Non-Periodik

Untuk sinyal non-periodik, konsep Deret Fourier diperluas dengan memperlakukan sinyal sebagai periodik dengan periode mendekati tak hingga ( $T \rightarrow \infty$ ) [1]. Dalam limit ini, spektrum frekuensi menjadi kontinu dan jumlahan berubah menjadi integral, melahirkan pasangan Transformasi Fourier:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt \quad (5)$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} d\omega \quad (6)$$

Untuk citra 2D, digunakan Transformasi Fourier 2D kontinu. Namun, bentuk ini tidak dapat dihitung secara langsung oleh komputer karena melibatkan operasi integral pada fungsi kontinu [2].

### 2.3 Transformasi Fourier Diskrit (DFT)

Untuk aplikasi digital, Transformasi Fourier Diskrit (DFT) dikembangkan. DFT diterapkan pada sinyal yang telah diambil sampelnya (diskrit) dan menghasilkan spektrum frekuensi yang

juga diskrit. Untuk sekuens data 1D  $f[n]$  dengan  $N$  sampel, DFT didefinisikan sebagai [2]:

$$F[k] = \sum_{n=0}^{N-1} f[n] e^{-j \frac{2\pi k n}{N}}, \quad k = 0, \dots, N-1 \quad (7)$$

Meskipun dapat diimplementasikan secara komputasi, DFT memiliki dua kelemahan fundamental untuk kompresi citra:

- Menghasilkan output bilangan kompleks, yang tidak efisien untuk disimpan.
- Asumsi periodisitas implisit menciptakan diskontinuitas artifisial pada batas blok citra, menghasilkan banyak koefisien frekuensi tinggi yang tidak perlu [3].

## 2.4 Hubungan DFT, DCT, dan DST

Dengan memisahkan komponen real dan imajiner dari (7), diperoleh:

$$F[k] = \sum_{n=0}^{N-1} x[n] \left[ \cos\left(\frac{2\pi k n}{N}\right) - j \sin\left(\frac{2\pi k n}{N}\right) \right]. \quad (8)$$

Jika sinyal diperluas secara *even symmetric*, maka komponen sinus akan hilang dan tersisa hanya kosinus. Hasilnya adalah:

$$F_c[k] = \sum_{n=0}^{N-1} x[n] \cos\left[\frac{\pi}{N} \left(n + \frac{1}{2}\right) k\right], \quad k = 0, 1, \dots, N-1 \quad (9)$$

yang dikenal sebagai **DCT-II**.

Jika sinyal diperluas secara *odd symmetric*, maka komponen kosinus akan hilang dan tersisa hanya sinus:

$$F_s[k] = \sum_{n=0}^{N-1} x[n] \sin\left[\frac{\pi}{N+1} (n+1)(k+1)\right], \quad k = 0, 1, \dots, N-1. \quad (10)$$

Dengan demikian, baik DCT maupun DST dapat dianggap sebagai versi *real-valued* dari DFT yang didesain untuk memenuhi kondisi batas tertentu.

## 3 Transformasi Kosinus Diskrit (DCT)

DCT muncul sebagai solusi yang dirancang untuk mengatasi kelemahan DFT. DCT tidak hanya menghasilkan koefisien riil, tetapi juga memiliki properti "pemadatan energi" yang jauh lebih unggul, yang menjadi kunci keberhasilan standar kompresi JPEG [4].

### 3.1 Definisi Matematis

DCT pertama kali diperkenalkan oleh Ahmed, Natarajan, dan Rao pada tahun 1974 [5]. Berbeda dengan DFT, DCT hanya menggunakan basis fungsi kosinus, sehingga input riil (nilai piksel) akan selalu menghasilkan output riil [3]. Varian yang paling umum digunakan dalam kompresi citra adalah DCT-II, yang untuk sekuens 1D  $f[n]$  didefinisikan sebagai:

$$F[k] = c[k] \sum_{n=0}^{N-1} f[n] \cos\left(\frac{(2n+1)k\pi}{2N}\right) \quad (11)$$

di mana  $c[k]$  adalah faktor normalisasi. Untuk aplikasi citra 2D pada blok  $N \times N$ , digunakan formula 2D-DCT yang dapat dipisahkan (*separable*). Secara konseptual, menerapkan DCT-II pada  $N$  titik setara dengan mengambil DFT dari sekuens yang telah diperpanjang secara simetris menjadi  $2N$  titik, yang secara efektif menghilangkan diskontinuitas pada batas blok [3].

### 3.2 DCT pada 2-Dimensi

DCT pada 2-dimensi diterapkan pada blok citra  $N \times N$  dengan cara yang dapat dipisahkan. Ini berarti bahwa DCT dapat diterapkan secara terpisah pada setiap baris dan kolom blok citra. Secara matematis, DCT 2D untuk blok citra  $f[x, y]$  didefinisikan sebagai:

$$F_{[u,v]}(x, y) = \frac{1}{4} \alpha[u] \alpha[v] \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right) \quad (12)$$

di mana  $\alpha[u]$  dan  $\alpha[v]$  adalah faktor normalisasi yang didefinisikan sebagai:

$$\alpha[k] = \begin{cases} \frac{1}{\sqrt{N}}, & k = 0, \\ \frac{1}{\sqrt{2N}}, & k \geq 1. \end{cases} \quad (13)$$

## 4 Anatomi Standar Kompresi JPEG

Standar kompresi JPEG adalah proses multi-tahap yang dirancang untuk mengurangi ukuran file citra secara signifikan, di mana setiap langkah bekerja secara sinergis [4].

### 4.1 Pra-pemrosesan

Langkah pertama untuk citra berwarna adalah transformasi ruang warna dari RGB ke YCbCr, yang memisahkan komponen kecerahan (Y) dari komponen warna (Cb, Cr). Karena mata manusia kurang sensitif terhadap perubahan warna, resolusi spasial komponen Cb dan Cr dikurangi melalui teknik *chroma subsampling*, yang merupakan langkah kompresi pertama yang sangat efisien [4].

Transformasi dari ruang warna RGB ke ruang Y'CbCr pada standar terbaru, yaitu *ITU-R BT.2020*, didefinisikan dengan mempertimbangkan sifat non-linear dari sistem tampilan modern [6]. Pada standar ini, komponen warna yang digunakan adalah  $R'$ ,  $G'$ , dan  $B'$ , yakni hasil dari penerapan fungsi *gamma correction* terhadap nilai RGB linear. Fungsi non-linear tersebut diberikan oleh:

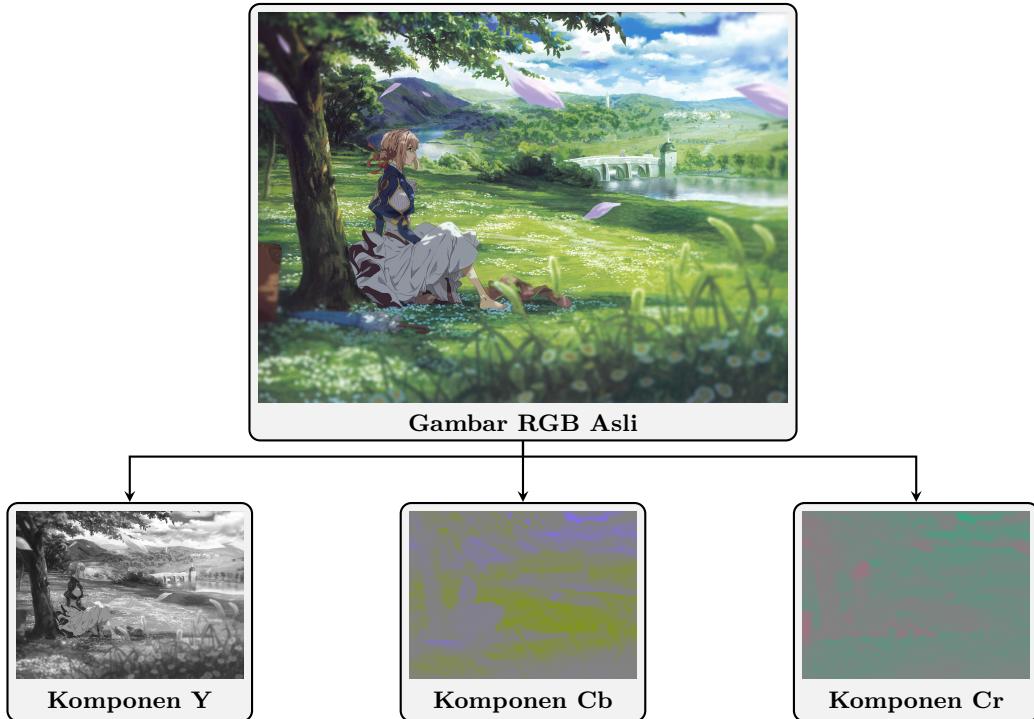
$$f(x) = \begin{cases} 4.5x, & x < 0.018, \\ 1.099x^{0.45} - 0.099, & x \geq 0.018, \end{cases} \quad (14)$$

sehingga diperoleh  $R' = f(R)$ ,  $G' = f(G)$ , dan  $B' = f(B)$ .

Selanjutnya, komponen luminansi ( $Y'$ ) serta dua komponen krominansi ( $Cb$ ,  $Cr$ ) dihitung menggunakan kombinasi linier dari  $R'$ ,  $G'$ , dan  $B'$  sebagai berikut:

$$\begin{bmatrix} Y' \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.2627 & 0.6780 & 0.0593 \\ -0.13963 & -0.36037 & 0.50000 \\ 0.50000 & -0.45979 & -0.04021 \end{bmatrix} \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix}. \quad (15)$$

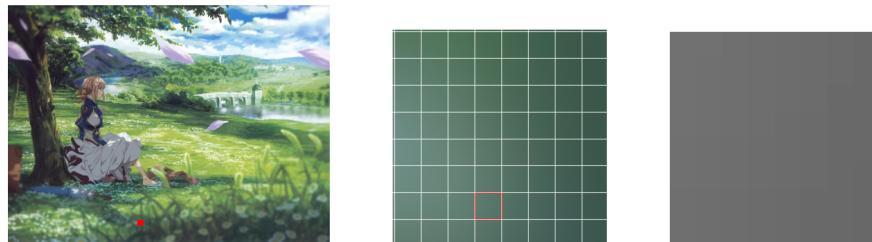
Koefisien pada matriks di atas diturunkan dari nilai konstanta primaries standar BT.2020, yaitu  $K_R = 0.2627$ ,  $K_B = 0.0593$ , dan  $K_G = 1 - K_R - K_B = 0.6780$ . Pendekatan ini memungkinkan representasi luminansi ( $Y'$ ) yang konsisten dengan persepsi visual manusia, serta menghasilkan rentang warna (*color gamut*) yang lebih luas pada sistem tampilan berdefinisi tinggi.



Gambar 1: Gambar RGB dipecah menjadi tiga komponen: Y, Cb, dan Cr.

## 4.2 Partisi Blok

Setiap kanal (Y, Cb, Cr) dibagi menjadi blok-blok  $8 \times 8$  piksel. Nilai piksel dalam setiap blok digeser levelnya menjadi rentang [-128, 127] dari yang semula [0, 255], kemudian 2D-DCT diterapkan pada setiap blok, mengubah 64 nilai piksel spasial menjadi 64 koefisien frekuensi.



Gambar 2: Partisi blok  $64 \times 64$  dilanjutkan dengan  $8 \times 8$  pada citra luminansi.

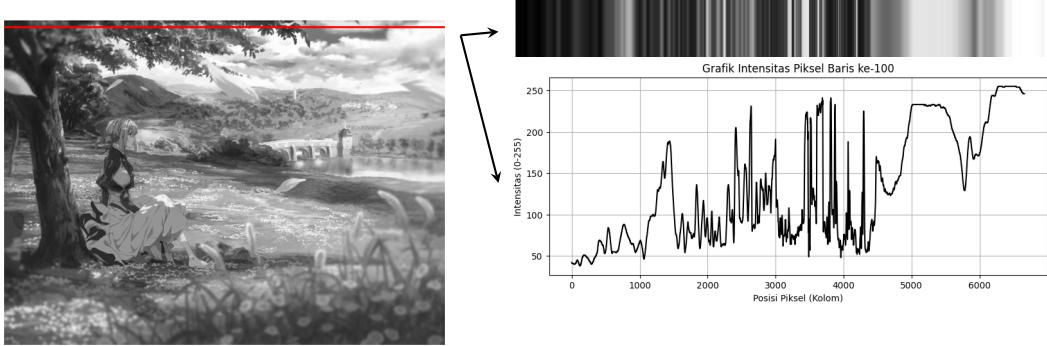
Selanjutnya, blok-blok ini dapat diproses dalam ukuran yang lebih besar, seperti  $16 \times 16$

atau  $32 \times 32$ , untuk mengurangi overhead pemrosesan. Namun, ukuran blok yang lebih besar cenderung menghasilkan artefak visual yang lebih jelas, terutama pada tingkat kompresi tinggi. Oleh karena itu, standar JPEG memilih ukuran blok  $8 \times 8$  sebagai kompromi optimal antara efisiensi kompresi dan kualitas visual [4]. Sebagai tambahan informasi, pada gambar 2 bisa diperoleh matriks  $8 \times 8$  dari blok yang dipilih sebagai berikut:

$$\begin{bmatrix} 114 & 113 & 111 & 110 & 109 & 108 & 107 & 106 \\ 113 & 112 & 111 & 110 & 109 & 108 & 107 & 105 \\ 113 & 112 & 111 & 110 & 109 & 108 & 106 & 105 \\ 113 & 112 & 111 & 110 & 108 & 107 & 106 & 105 \\ 112 & 111 & 110 & 109 & 108 & 107 & 106 & 105 \\ 112 & 111 & 110 & 109 & 108 & 107 & 106 & 104 \\ 111 & 110 & 109 & 108 & 107 & 106 & 105 & 104 \\ 111 & 110 & 109 & 108 & 107 & 106 & 105 & 104 \end{bmatrix}$$

### 4.3 Aplikasi DCT

Pertama jika kita tinjau salah satu baris piksel dari citra *grayscale* (Y), misalnya baris ke-100 dari citra  $6660 \times 4898$  piksel Gambar 1. Perhatikan pada Gambar 3 bahwa perbaris piksel dari



Gambar 3: Visualisasi baris piksel sebagai strip horizontal dan grafik intensitasnya.

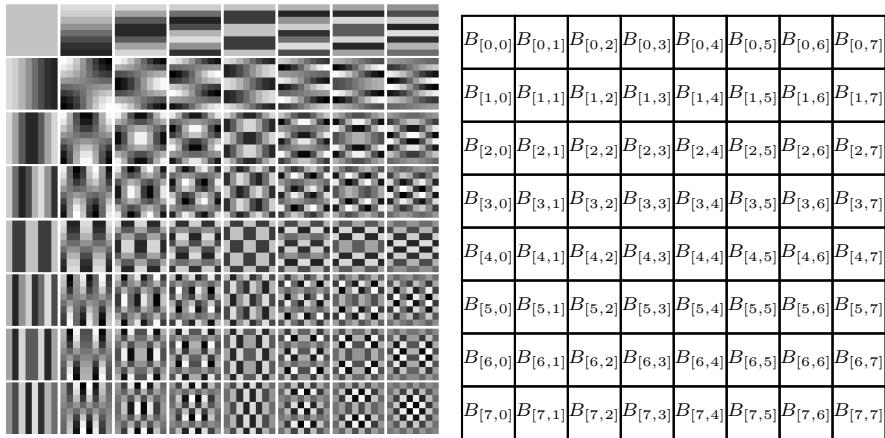
komponen Y (*luminance*) dapat kita pandang sebagai grafik intensitas piksel terhadap posisi horizontalnya. Grafik ini dapat kita anggap sebagai sinyal 1D yang diskrit dan terhingga, sehingga kita dapat menerapkan DCT 1D (11) untuk menganalisis komponen frekuensinya.

Pada kasus kompresi JPEG, DCT 2D diterapkan pada total 64 blok. Dengan menggunakan pengetahuan tentang Aljabar Linear, kita dapat mengekspresikan blok  $8 \times 8$  sebagai kombinasi linier dari basis DCT 2D. Basis ini diambil dari persamaan (12) dengan memasukkan nilai  $f(x, y) = 1$ . Sehingga disajikan dalam bentuk fungsi dua variabel  $B_{[u,v]}(x, y)$  sebagai berikut:

$$B_{[u,v]}(x, y) = \frac{1}{4} \alpha[u] \alpha[v] \cos\left(\frac{(2x+1)u\pi}{16}\right) \cos\left(\frac{(2y+1)v\pi}{16}\right) \quad (16)$$

dengan  $u, v, x, y = 0, 1, \dots, 7$  dan

$$\alpha[k] = \begin{cases} \frac{1}{\sqrt{8}}, & k = 0, \\ \frac{1}{2}, & k > 0. \end{cases} \quad (17)$$



Gambar 4: Basis DCT 2D untuk blok  $8 \times 8$ .

Gambar 4 menampilkan 64 basis DCT 2D yang digunakan untuk merepresentasikan blok  $8 \times 8$ . Basis ini diurutkan berdasarkan frekuensi spasialnya, dari frekuensi rendah di kiri atas hingga frekuensi tinggi di kanan bawah. Setiap basis dapat dianggap sebagai pola gelombang kosinus dengan frekuensi tertentu dalam arah horizontal dan vertikal.

#### 4.4 Kuantisasi: Kompresi *Lossy*

Kuantisasi adalah satu-satunya tahap di mana informasi hilang secara permanen. Prosesnya melibatkan pembagian setiap koefisien DCT dengan nilai dari matriks kuantisasi, lalu hasilnya dibulatkan ke bilangan bulat terdekat.

$$F_Q[u, v] = \text{round} \left( \frac{F[u, v]}{Q[u, v]} \right) \quad (18)$$

Matriks kuantisasi  $Q[u, v]$  dirancang berdasarkan sensitivitas visual manusia, dengan nilai kecil untuk frekuensi rendah dan besar untuk frekuensi tinggi. Ini berarti koefisien frekuensi tinggi yang kurang penting secara perceptual akan dikurangi secara drastis, sering kali menjadi nol [4].

#### 4.5 Serialisasi dan Pengkodean Entropi

Setelah kuantisasi, matriks  $8 \times 8$  yang "jarang" diubah menjadi vektor 1D menggunakan pemindaian zig-zag. Pola ini mengelompokkan koefisien non-nol di awal vektor dan menciptakan barisan panjang nilai nol di akhir. Vektor ini kemudian dikodekan secara *lossless* menggunakan DPCM untuk koefisien DC dan RLE untuk koefisien AC, diikuti oleh pengkodean Huffman untuk merepresentasikan data akhir seefisien mungkin [4].

#### 4.6 Proses Dekompresi

Untuk merekonstruksi citra, seluruh proses dilakukan secara terbalik: Dekode Entropi, De-serialisasi, De-kuantisasi, dan Transformasi Kosinus Diskrit Terbalik (IDCT). Pada IDCT, koefisien frekuensi dikonversi kembali ke domain spasial menggunakan sifat dari keorthonormalan

basis DCT [4].

$$f(x, y) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 \alpha[u] \alpha[v] F_{[u,v]}(x, y) \cos\left(\frac{(2x+1)u\pi}{16}\right) \cos\left(\frac{(2y+1)v\pi}{16}\right) \quad (19)$$

## 5 Analisis Kinerja dan Artefak Visual

Efektivitas JPEG diukur dari rasio kompresi dan kualitas visual, yang merupakan sebuah pertukaran (*trade-off*).



Gambar 5: Artefak visual pada citra JPEG berbanding lurus dengan tingkat kompresi

### 5.1 Keunggulan dan Kelemahan

Keunggulan utama JPEG adalah rasio kompresi yang sangat tinggi pada citra fotografi dengan degradasi perceptual minimal pada kualitas tinggi [4]. Namun, sifatnya yang *lossy* berarti kualitas citra menurun setiap kali disimpan ulang. Pada kompresi tinggi, artefak visual menjadi jelas terlihat.

### 5.2 Artefak Visual Kompresi

- **Artefak Pemblokiran (*Blocking Artifacts*):** Muncul sebagai pola kisi-kisi  $8 \times 8$ , terutama di area gradasi halus. Ini disebabkan oleh pemrosesan independen pada setiap blok dan kuantisasi kasar yang menciptakan batas artifisial antar blok [7], [8].
- **Ringing dan Blurring:** *Blurring* terjadi ketika koefisien frekuensi tinggi (detail halus) dihilangkan. *Ringing* muncul sebagai "gema" di sekitar tepi kontras tinggi, sebuah manifestasi dari Fenomena Gibbs akibat pembuangan komponen frekuensi tinggi yang diperlukan untuk merekonstruksi tepi yang tajam [9], [10].

## 6 Contoh Penerapan dan Perhitungan Kompresi JPEG

Untuk blok berukuran  $8 \times 8$ , transformasi DCT didefinisikan sebagai:

$$F_{[u,v]}(x, y) = \frac{1}{4} \alpha[u] \alpha[v] \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos\left[\frac{(2x+1)u\pi}{16}\right] \cos\left[\frac{(2y+1)v\pi}{16}\right], \quad (20)$$

Tabel 1: Perbandingan Kualitas vs. Tingkat Kompresi JPEG

Kualitas (%)	Rasio Kompresi	Deskripsi Artefak Visual
95-100	2:1 - 5:1	Artefak hampir tidak terlihat.
85-94	5:1 - 15:1	Kualitas sangat baik untuk sebagian besar penggunaan.
75-84	15:1 - 25:1	Keseimbangan baik, artefak mulai terlihat.
50-74	25:1 - 40:1	Artefak jelas, detail mulai hilang.
<50	> 40:1	Degradasi citra sangat parah.

dengan faktor normalisasi

$$\alpha[k] = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0, \\ 1, & k > 0. \end{cases}$$

Fungsi  $f(x, y)$  menyatakan nilai intensitas piksel pada posisi  $(x, y)$  setelah dikurangi dengan 128 untuk menormalkan rentang nilai menjadi sekitar nol.

Sebagai contoh, misalkan blok luminansi (komponen Y) dari citra berukuran  $8 \times 8$  diberikan oleh:

$$f(x, y) = \begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}.$$

Setelah dinormalisasi dengan mengurangkan 128 dari setiap elemen, diperoleh matriks  $f'(x, y) = f(x, y) - 128$ .

Koefisien pertama  $F_{[0,0]}$ , disebut *DC coefficient*, mewakili energi rata-rata dari blok dan dihitung dengan:

$$F_{[0,0]} = \frac{1}{8} \sum_{x=0}^7 \sum_{y=0}^7 f'(x, y).$$

Sementara itu, koefisien lainnya  $F_{[u,v]}$  untuk  $u, v > 0$  disebut *AC coefficients*, yang masing-masing menggambarkan variasi frekuensi spasial citra.

Hasil perhitungan DCT dari blok di atas (secara numerik) adalah:

$$C(u, v) = \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & 2.79 & -3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix}.$$

Koefisien di kiri atas menyimpan komponen frekuensi rendah (rata-rata dan struktur kasar), sedangkan koefisien di kanan bawah menyimpan frekuensi tinggi (tepi dan detail halus). Pada proses kompresi JPEG, nilai-nilai frekuensi tinggi ini akan dikurangi presisinya melalui proses *kuantisasi* untuk menghemat ruang penyimpanan tanpa mengorbankan kualitas visual secara signifikan.

Langkah selanjutnya adalah kuantisasi, di mana setiap elemen dari matriks DCT dibagi dengan elemen yang bersesuaian dari matriks kuantisasi standar, kemudian dibulatkan ke bilangan bulat terdekat. Proses ini adalah inti dari kompresi *lossy* pada JPEG. Matriks kuantisasi standar untuk komponen luminansi (kualitas sekitar 50%) adalah:

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

Dengan menerapkan rumus kuantisasi  $F_Q[u, v] = \text{round}(F[u, v]/Q[u, v])$ , matriks koefisien DCT yang telah dikuantisasi menjadi:

$$F_Q[u, v] = \begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Terlihat jelas bahwa sebagian besar koefisien frekuensi tinggi (di bagian kanan bawah) menjadi nol. Ini adalah sumber utama penghematan data.

Matriks yang telah dikuantisasi kemudian diubah menjadi urutan 1D menggunakan pemindai zig-zag. Pola ini mengelompokkan koefisien nol di akhir urutan, yang sangat efisien untuk kompresi.

$$\text{Urutan Zig-Zag} = [-26, -3, 0, -3, -2, -6, 2, -3, 1, 1, 1, 2, -4, \dots, 0, 0, 0].$$

Urutan ini kemudian dikodekan secara *lossless*. Koefisien DC (-26) dikodekan menggunakan DPCM (Differential Pulse Code Modulation), sedangkan koefisien AC (sisanya) dikodekan menggunakan RLE (Run-Length Encoding) untuk merepresentasikan urutan nol yang panjang, diikuti oleh pengkodean Huffman atau Aritmetika untuk menghasilkan aliran bit akhir yang terkompreesi.

## 7 Kesimpulan

Analisis mendalam terhadap alur kerja JPEG menunjukkan bagaimana DCT, dikombinasikan dengan kuantisasi yang mengeksplorasi keterbatasan persepsi visual manusia, menjadi inti dari kompresi *lossy*. Proses ini secara efektif mengurangi redundansi data, meskipun menimbulkan

artefak seperti *blocking* dan *ringing* pada tingkat kompresi tinggi. Dengan demikian, JPEG merepresentasikan sebuah kompromi cerdas antara efisiensi penyimpanan dan kualitas visual. Sebagai kesimpulan, keberhasilan JPEG membuktikan kekuatan penerapan teori transformasi frekuensi dalam rekayasa praktis. Meskipun standar yang lebih modern seperti AVIF kini menawarkan efisiensi yang lebih tinggi, pemahaman terhadap prinsip kerja JPEG dan DCT tetap relevan sebagai fondasi untuk memahami evolusi teknologi kompresi citra.

## Pustaka

- [1] R. N. Bracewell, *The Fourier Transform and Its Applications*, 3rd. McGraw-Hill, 1999.
- [2] A. V. Oppenheim **and** A. S. Willsky, *Signals and Systems*, 2nd. Prentice Hall, 1996.
- [3] Signal and Video Compression Lab, *Discrete Cosine Transform (DCT)*, University of California, San Diego, Course Notes for ECE161C, 2005.
- [4] G. K. Wallace, “The JPEG Still Picture Compression Standard,” *Communications of the ACM*, **jourvol** 34, **number** 4, **pages** 30–44, 1991. DOI: [10.1145/103085.103089](https://doi.org/10.1145/103085.103089).
- [5] N. Ahmed, T. Natarajan **and** K. R. Rao, “Discrete Cosine Transform,” *IEEE Transactions on Computers*, **jourvol** C-23, **number** 1, **pages** 90–93, 1974. DOI: [10.1109/T-C.1974.223784](https://doi.org/10.1109/T-C.1974.223784).
- [6] “Recommendation ITU-R BT.2020-2: Parameter values for ultra-high definition television systems for production and international programme exchange,” International Telecommunication Union, techreport, 2015.
- [7] S. Singh, “Reduction of Blocking Artifacts In JPEG Compressed Image,” **in** TECHNOVISION-10 arXiv:1210.1192, 2010.
- [8] J. Chou, M. Crouse **and** K. Ramchandran, “Implementation of a real-time software-only image smoothing filter for blocking artifact reduction,” **in** Proc. SPIE 5022, *Visual Communications and Image Processing* 1998. DOI: [10.1117/12.476438](https://doi.org/10.1117/12.476438).
- [9] D. Gottlieb **and** C.-W. Shu, “On the Gibbs Phenomenon and Its Resolution,” *SIAM Review*, **jourvol** 39, **number** 4, **pages** 644–668, 1997. DOI: [10.1137/S0036144596301390](https://doi.org/10.1137/S0036144596301390).
- [10] P. Marziliano, F. Dufaux, S. Winkler **and** T. Ebrahimi, “Perceptual blur and ringing metrics: application to JPEG2000,” **in** *Signal Processing: Image Communication* volume 19, 2004, **pages** 163–172.