LAPORAN TUGAS BESAR 2

IF2123 Aljabar Linier dan Geometri

Aplikasi Nilai Eigen dan Vektor Eigen dalam Kompresi Gambar

Disusun oleh:

1. Afrizal Sebastian (13520120)
2. Johannes Winson Sukiatmodjo (13520123)
3. Frederik Imanuel Louis (13520163)

PROGRAM STUDI TEKNIK INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2021

BAB I

DESKRIPSI MASALAH

1.1 Abstraksi

Gambar adalah suatu hal yang sangat dibutuhkan pada dunia modern ini. Kita seringkali berinteraksi dengan gambar baik untuk mendapatkan informasi maupun sebagai hiburan. Gambar digital banyak sekali dipertukarkan di dunia digital melalui file-file yang mengandung gambar tersebut. Seringkali dalam transmisi dan penyimpanan gambar ditemukan masalah karena ukuran file gambar digital yang cenderung besar.

Kompresi gambar merupakan suatu tipe kompresi data yang dilakukan pada gambar digital. Dengan kompresi gambar, suatu file gambar digital dapat dikurangi ukuran filenya dengan baik tanpa mempengaruhi kualitas gambar secara signifikan. Terdapat berbagai metode dan algoritma yang digunakan untuk kompresi gambar pada zaman modern ini.

Three levels of JPG compression. The left-most image is the original. The middle image offers a medium compression, which may not be immediately obvious to the naked eye without closer inspection. The right-most image is maximally compressed.

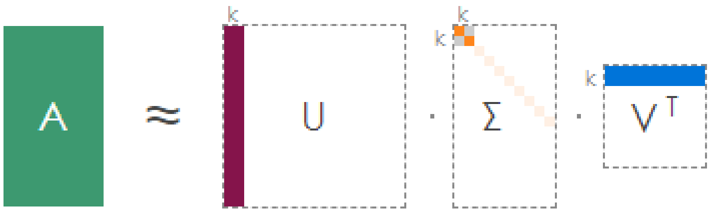
Gambar 1. Contoh kompresi gambar dengan berbagai tingkatan
Sumber : [Understanding Compression in Digital Photography (lifewire.com)]

Salah satu algoritma yang dapat digunakan untuk kompresi gambar adalah algoritma SVD (Singular Value Decomposition). Algoritma SVD didasarkan pada teorema dalam aljabar linier yang menyatakan bahwa sebuah matriks dua dimensi dapat dipecah menjadi hasil perkalian dari 3 sub-matriks yaitu matriks ortogonal U, matriks diagonal S, dan transpose dari matriks ortogonal V. Dekomposisi matriks ini dapat dinyatakan sesuai persamaan berikut.

$$A_{m\times n} = U_{m\times m}\ S_{m\times n}\ V^T_{nxn}$$

Gambar 1. Algoritma SVD

Matriks U adalah matriks yang kolomnya terdiri dari vektor eigen ortonormal dari matriks AA^T. Matriks ini menyimpan informasi yang penting terkait baris-baris matriks awal, dengan informasi terpenting disimpan di dalam kolom pertama. Matriks S adalah matriks diagonal yang berisi akar dari nilai eigen matriks U atau V yang terurut menurun. Matriks V adalah matriks yang kolomnya terdiri dari vektor eigen ortonormal dari matriks A^TA. Matriks ini menyimpan informasi yang penting terkait kolom-kolom matriks awal, dengan informasi terpenting disimpan dalam baris pertama.



Gambar 2. Ilustrasi Algoritma SVD dengan rank k

Dapat dilihat di gambar di atas bahwa dapat direkonstruksi gambar dengan banyak *singular values* **k** dengan mengambil kolom dan baris sebanyak k dari U dan V serta *singular value* sebanyak **k** dari S atau Σ terurut dari yang terbesar. Kita dapat mengaproksimasi suatu gambar yang mirip dengan gambar aslinya dengan mengambil **k** yang jauh lebih kecil dari jumlah total *singular value* karena kebanyakan informasi disimpan di *singular values* awal karena singular values terurut mengecil. Nilai **k** juga berkaitan dengan rank matriks karena banyaknya *singular value* yang diambil dalam matriks S adalah *rank* dari matriks hasil, jadi dalam kata lain **k** juga merupakan rank dari matriks hasil. Maka itu matriks hasil rekonstruksi dari SVD akan berupa informasi dari gambar yang terkompresi dengan ukuran yang lebih kecil dibanding gambar awal.

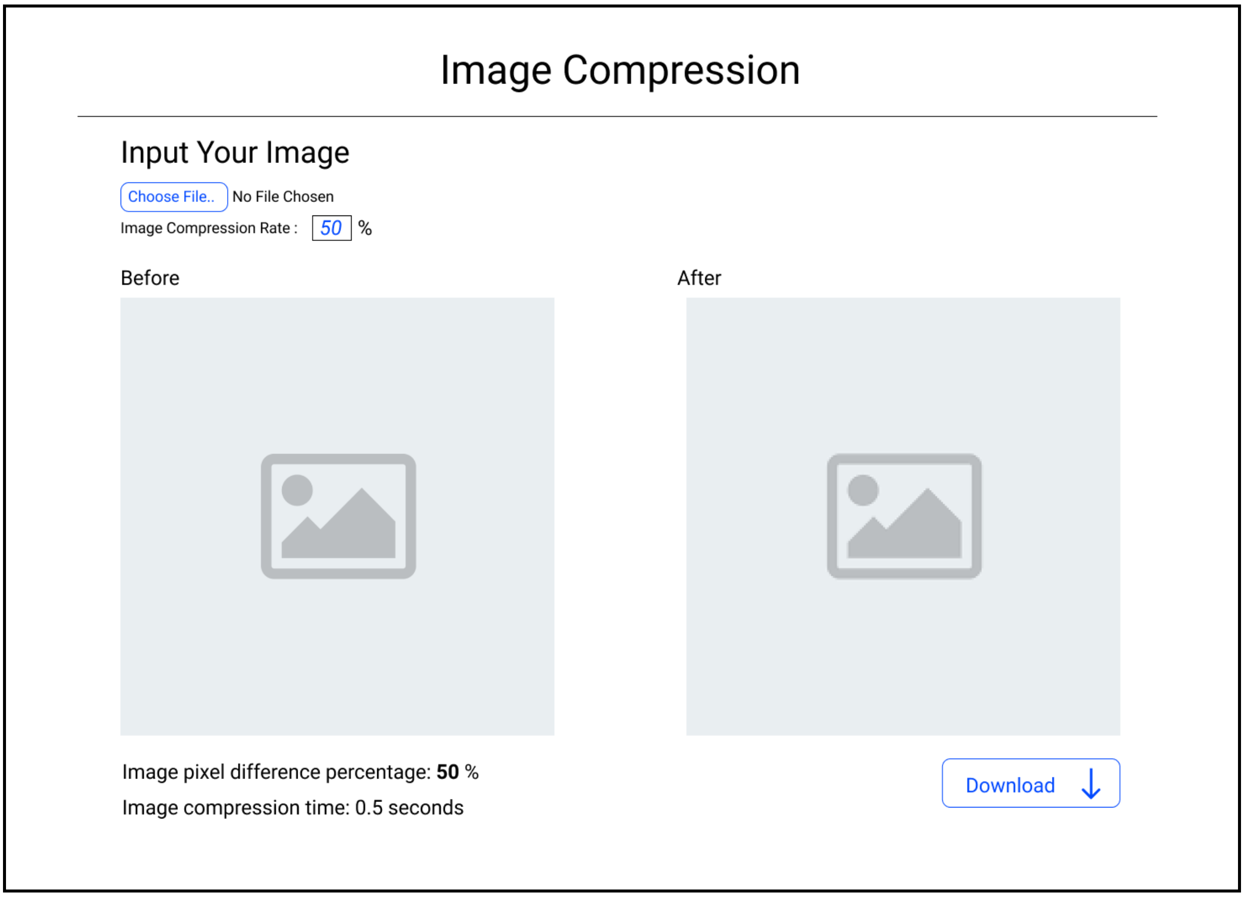
Pada kesempatan kali ini, kalian mendapatkan tantangan untuk membuat website kompresi gambar sederhana dengan menggunakan algoritma SVD.

1.2 Penggunaan Program

Berikut ini adalah input yang akan dimasukkan pengguna untuk eksekusi program.

1. **File gambar**, berisi *file* gambar input yang ingin dikompresi dengan format *file* yang bebas selama merupakan format untuk gambar.
2. **Tingkat kompresi**, berisi tingkat kompresi dari gambar (formatnya dibebaskan, cth: Jumlah *singular value* yang digunakan)

Tampilan *layout* dari aplikasi web yang akan dibangun kurang lebih adalah sebagai berikut. Anda dapat mengubah *layout* selama *layout* masih terdiri dari komponen yang sama.



Gambar 3. Contoh tampilan layout dari aplikasi web yang dibangun.

Catatan: Warna biru menunjukkan komponen yang dapat di klik.

Anda dapat menambahkan menu lainnya, gambar, logo, dan sebagainya. Tampilan *front end* dari *website* dibuat semenarik mungkin selama mencakup seluruh informasi pada layout yang diberikan di atas. Tampilan program merupakan bagian dari penilaian.

1.3 Spesifikasi Tugas

Buatlah program kompresi gambar dengan memanfaatkan algoritma SVD dalam bentuk website lokal sederhana. Spesifikasi website adalah sebagai berikut:

1. Website mampu menerima *file* gambar beserta *input* tingkat kompresi gambar (dibebaskan formatnya).
2. Website mampu menampilkan gambar *input*, *output*, *runtime* algoritma, dan persentase hasil kompresi gambar (perubahan jumlah pixel gambar).
3. File *output* hasil kompresi dapat diunduh melalui website.
4. Kompresi gambar tetap mempertahankan warna dari gambar asli.
5. **(Bonus)** Kompresi gambar tetap mempertahankan transparansi dari gambar asli, misal untuk gambar png dengan *background* transparan.
6. Bahasa pemrograman yang boleh digunakan adalah Python, Javascript, dan Go.
7. Penggunaan *framework* untuk *back end* dan *front end website* dibebaskan. Contoh

framework website yang bisa dipakai adalah Flask, Django, React, Vue, dan Svelte.

8. Kalian dapat menambahkan fitur fungsional lain yang menunjang program yang anda buat (unsur kreativitas diperbolehkan/dianjurkan).
9. Program harus modular dan mengandung komentar yang jelas.
10. Diperbolehkan menggunakan *library* pengolahan citra seperti OpenCV2, PIL, atau image dari Go.
11. **Dilarang** menggunakan *library* perhitungan SVD dan *library* pengolahan eigen yang sudah jadi.

BAB II

TEORI SINGKAT

2.1 Perkalian Matriks

Perkalian matriks adalah perkalian yang melibatkan suatu matriks atau susunan bilangan berupa kolom dan angka, serta memiliki sifat-sifat tertentu. Misalkan matriks A berukuran 2X2 dikalikan dengan matriks B berukuran 2X2, sehingga rumusnya akan menjadi:

$$A \times B$$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \times \begin{pmatrix} j & k \\ l & m \end{pmatrix}$$

$$A \times B = \begin{pmatrix} \boldsymbol{a}j + \boldsymbol{b}l & \boldsymbol{a}k + \boldsymbol{b}m \\ \boldsymbol{c}j + \boldsymbol{d}l & \boldsymbol{c}k + \boldsymbol{d}m \end{pmatrix}$$

Syarat dua matriks dapat dioperasikan perkalian yaitu banyak kolom matriks pertama harus sama dengan banyak baris matriks kedua, sebagai berikut:

$$A_{m\times\boldsymbol{n}} \times B_{\boldsymbol{n}\times t} = C_{m\times t}$$

Diberikan A, B, dan C adalah sembarang matriks yang elemennya bilangan riil, maka:

- Sifat perkalian dengan matriks nol

$$A \times O = O \times A = O$$

- Sifat perkalian asosiatif

$$(A \times B) \times C = A \times (B \times C)$$

- Sifat distributif kiri

$$A \times (B + C) = A \times B + A \times C$$

- Sifat distributif kanan

$$(A + B) \times C = A \times C + B \times C$$

- Sifat perkalian dengan konstanta c

$$c(A \times B) = (c \times A) \times B = A \times (c \times B)$$

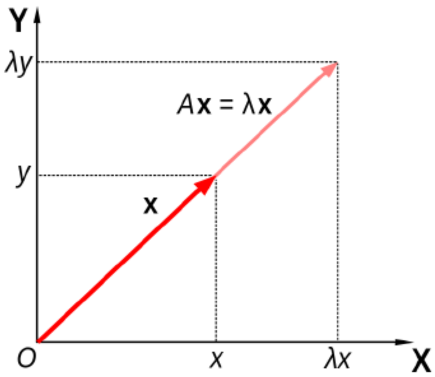
- Sifat perkalian dengan matriks identitas

$$A \times I = I \times A = A$$

2.2 Nilai Eigen dan Vektor Eigen

Kata “eigen” berasal dari Bahasa Jerman yang artinya “asli” atau “karakteristik”. Dengan kata lain, nilai eigen menyatakan nilai karakteristik dari sebuah matriks yang berukuran n x n.

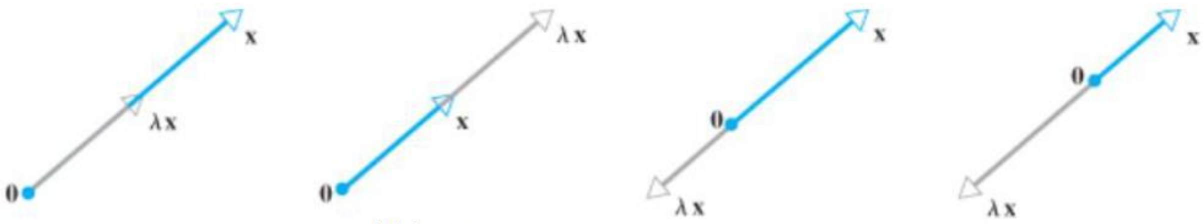
Vektor eigen x menyatakan vektor kolom yang apabila dikalikan dengan sebuah matriks n x n menghasilkan vektor lain yang merupakan kelipatan vektor itu sendiri.



Jika A adalah matriks n x n maka vektor tidak-nol x di R^n disebut vektor eigen dari A jika Ax sama dengan perkalian suatu skalar λ dengan x, yaitu

$$Ax = \lambda x$$

Skalar λ disebut nilai eigen dari A, dan x dinamakan vektor eigen yang berkoresponden dengan λ. Dengan kata lain, operasi Ax = λx menyebabkan vektor x menyusut atau memanjang dengan faktor λ dengan arah yang sama jika λ positif dan arah berkebalikan jika λ negatif.



(a) $0 \le \lambda \le 1$ (b) $\lambda \ge 1$ (c) $-1 \le \lambda \le 0$ (d) $\lambda \le -1$

Diberikan sebuah matriks A berukuran n x n. Vektor eigen dan nilai eigen dari matriks A dihitung sebagai berikut:

$$Ax = \lambda x$$

$$IAx = \lambda Ix$$

$$Ax = \lambda Ix$$

$$(\lambda I - A)x = 0$$

$x = 0$ adalah solusi trivial dari $(\lambda I - A)x = 0$. Agar $(\lambda I - A)x = 0$ memiliki solusi tidak-nol, maka haruslah

$$\det(\lambda I - A) = 0$$

Persamaan $\det(\lambda I - A) = 0$ disebut persamaan karakteristik dari matriks A, dan akar-akar persamaan tersebut, yaitu λ, dinamakan akar-akar karateristik atau nilai-nilai eigen. Namun, tidak semua matriks memiliki nilai-nilai eigen. Apabila akar-akar persamaan $\det(\lambda I - A) = 0$ adalah bilangan imajiner, matriks A tidak memiliki nilai-nilai eigen.

Nilai eigen dan vektor eigen dapat diaplikasikan dalam bidang grafika komputer, fisika (getaran mekanis, aliran panas, mekanika kuantum), biologi (dinamika populasi), sistem pendukung keputusan, ekonomi, dan lain-lain.

2.3 Matriks SVD

Mendekomposisi matriks artinya memfaktorkan sebuah matriks, misalnya A, menjadi hasil kali dari sejumlah matriks lain, $P_1, P_2, \ldots, P_k$

$$A = P_1 \times P_2 \times \ldots \times P_k$$

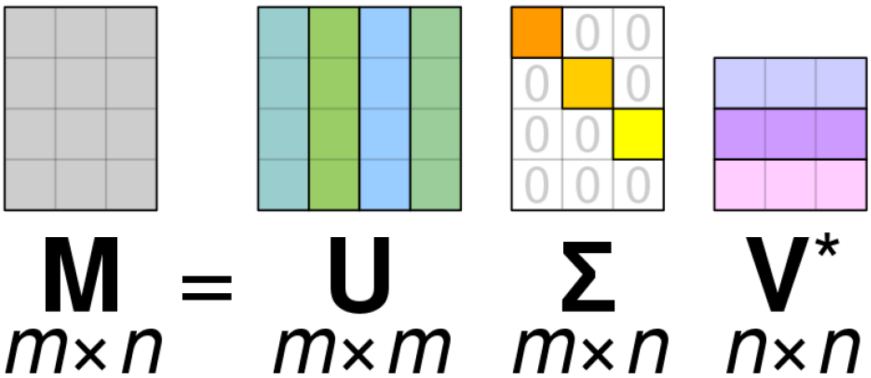
Terdapat beberapa metode mendekomposisi matriks, salah satunya adalah metode dekomposisi nilai singular (singular value decomposition – SVD). SVD memfaktorkan matriks A berukuran m x n menjadi matriks U, Σ, dan V sedemikian sehingga

$$A = U\Sigma V^T$$

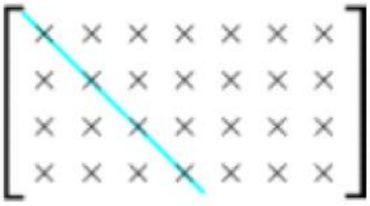
U = matriks ortogonal m x m

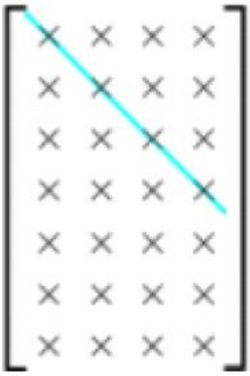
V = matriks ortogonal n x n

Σ = matriks berukuran m x n yang elemen-elemen diagonal utamanya adalah nilai-nilai singular dari matriks A dan elemen-elemen lainnya 0



Diagonal utama sebuah matriks biasanya didefinisikan pada matriks persegi (matriks bujursangkar) berukuran n x n. Untuk matriks bukan bujursangkar, yaitu matriks m x n, diagonal utama matriks didefinisikan pada garis yang dimulai dari sudut kiri atas terus ke bawah matriks sejauh mungkin.



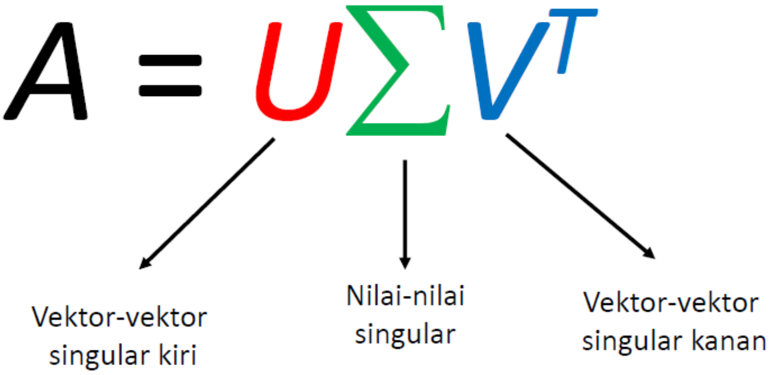


Matriks ortogonal adalah matriks yang kolom-kolomnya adalah vektor yang saling ortogonal satu sama lain (hasil kali titik sama dengan 0). Jika vektor-vektor kolom tersebut merupakan vektor satuan, maka matriks ortogonal tersebut dinamakan juga matriks ortonormal. Vektor satuan adalah vektor yang dinormalisasi dengan panjang atau magnitude-nya sehingga memiliki panjang atau magnitude = 1. Jika Q adalah matriks ortogonal m x n, dan kolom-kolom matriks Q adalah vektor-vektor satuan $v_1, v_2, \ldots, v_m$, maka $v_i \cdot v_j = 0$ untuk $i \neq j$. Atau, dapat juga dikatakan bahwa Q adalah matriks ortogonal jika $Q^TQ = I$, dalam hal ini I adalah matriks identitas.

Misalkan A adalah matriks m x n. Jika $\lambda_1, \lambda_2, \ldots, \lambda_n$ adalah nilai-nilai eigen dari A^TA, maka

$$\sigma_1 = \sqrt{\lambda_1},\ \sigma 2 = \sqrt{\lambda_2},\ \ldots,\ \sigma_n = \sqrt{\lambda_n}$$

disebut nilai-nilai singular dari matriks A. Diasumsikan $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$ sehingga $\sigma_1 \geq \sigma 2 \geq \ldots \geq \sigma_n \geq 0$



Langkah-langkah SVD mendekomposisi A_{mxn} menjadi U, Σ, dan V:

1. Untuk vektor singular kiri, hitung nilai-nilai eigen dari AA^T. Rank(A) = k = banyaknya nilai-nilai eigen tidak nol dari AA^T.

2. Tentukan vektor-vektor eigen $u_1, u_2, \ldots, u_m$ yang berkoresponden dengan nilai-nilai eigen dari AA^T. Normalisasi $u_1, u_2, \ldots, u_m$ dengan cara setiap komponen vektornya dibagi dengan panjang vektor. Diperoleh matriks U.

3. Untuk vektor singular kanan, hitung nilai-nilai eigen dari A^TA lalu tentukan nilai-nilai singularnya.

4. Tentukan vektor-vektor eigen $v_1, v_2, \ldots, v_n$ yang berkoresponden dengan nilai-nilai eigen dari A^TA. Normalisasi $v_1, v_2, \ldots, v_n$ dengan cara setiap komponen vektornya dibagi dengan panjang vektor. Diperoleh matriks V. Transpose-kan matriks V sehingga menjadi V^T.

5. Bentuklah matriks Σ berukuran m x n dengan elemen-elemen diagonalnya adalah nilai-nilai singular dari matriks A dengan susunan dari besar ke kecil. Nilai singular di dalam Σ adalah akar pangkat dua dari nilai-nilai eigen yang tidak nol dari A^TA.

6. Maka, $A = U\Sigma V^T$

SVD dapat diaplikasikan untuk kompresi gambar dan video (image and video compression), pengolahan citra (image processing), machine learning, computer vision, digital watermarking, dan lain-lain.

BAB III

IMPLEMENTASI PROGRAM

Program untuk menyelesaikan persoalan terbagi menjadi dua bagian, yaitu backend dan frontend, dimana bagian backend berfungsi untuk melakukan proses kompresi gambar dan bagian frontend berfungsi untuk menerima input gambar user, dan mengeluarkan gambar ke user.

1. Backend

Bagian backend terdiri dari file backend.py yang menangani seluruh proses kompresi data. File backend.py sendiri terdiri dari berbagai bagian untuk melaksanakan proses tersebut, antara lain:

a. imageToMatRGB
   Fungsi ini menggunakan library Pillow untuk mengubah file image menjadi numpy array. Array kemudian diproses menjadi tiga buah array, masing-masing merepresentasikan nilai Red, Green, dan Blue.
b. matToMatRGB
   Fungsi ini menggunakan library Pillow untuk mengubah numpy array menjadi image file. Image kemudian disimpan sebagai nama file yang diterima di argumen system.
c. sumPol
   Fungsi ini digunakan untuk menjumlahkan dua polinom yang direpresentasikan sebagai array satu dimensi.
d. subsPol
   Fungsi ini digunakan untuk mengurangkan dua polinom yang direpresentasikan sebagai array satu dimensi
e. subMatrix
   Fungsi ini digunakan untuk mengambil submatriks dari suatu matriks untuk keperluan pencarian determinan
f. detMatrixPol
   Fungsi ini digunakan untuk mencari nilai determinan matriks yang memiliki anggota variabel.
g. invMatDet
   Fungsi ini digunakan untuk membalik hasil determinan matriks dengan anggota variabel untuk keperluan penyamaan format dengan library roots dari numpy.
h. sqrt
   Fungsi ini digunakan untuk mencari panjang suatu vektor untuk keperluan normalisasi vektor.
i. gaussJordan
   Fungsi ini digunakan untuk mencari eigenvector dari nilai eigen yang diberikan. Fungsi akan mengeluarkan array of eigenvector sesuai dengan nullitas matriks.

j. U

   Fungsi ini menggunakan fungsi-fungsi diatas untuk menyusun matriks U dari singular value decomposition satu matriks.

k. sigma

   Fungsi ini menggunakan fungsi-fungsi diatas untuk menyusun matriks sigma dari singular value decomposition satu matriks.

l. VT

   Fungsi ini menggunakan fungsi-fungsi diatas untuk menyusun matriks V transpose dari singular value decomposition satu matriks.

m. SVD

   Fungsi ini menggunkan funsi U, sigma, dan VT untuk menyusun kembali matriks hasil kompresi.

n. main

   Fungsi ini memanggil fungsi imageToMatRGB, matToImageRGB dan SVD untuk menghubungkan seluruh proses kompresi gambar.

2. **Frontend**

   Bagian frontend terdiri dari file index.php dan file style.css. File index.php mengatur struktur website, menerima masukan gambar user, menampilkan gambar original beserti hasil kompresi, dan memungkinkan user untuk mendownload gambar hasil kompresi. Untuk melakukan hal ini, index.php memanggil backend.py untuk melakukan proses kompresi gambar. File style.css berfungsi untuk memperindah tampilan website.

BAB IV

EKSPERIMEN

1. **Test Case 1**

   Original Picture:

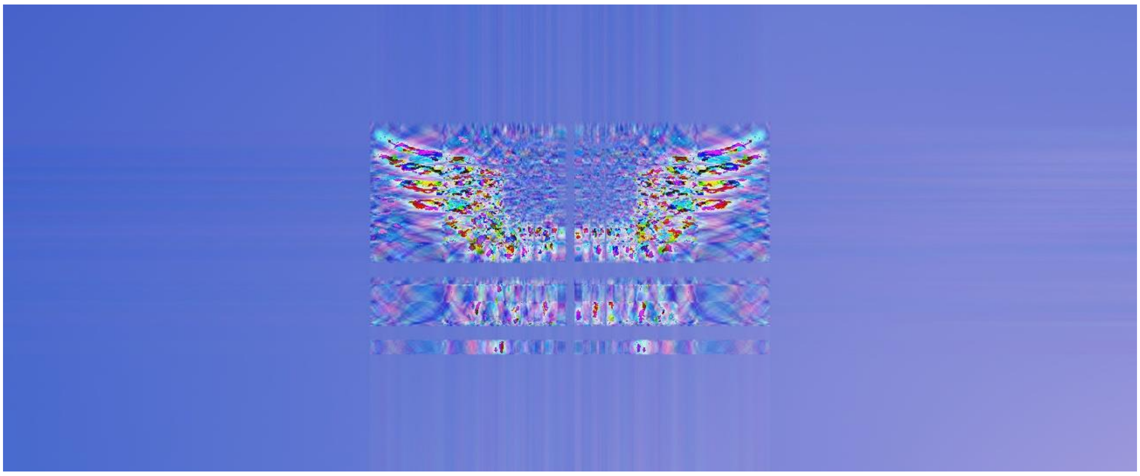
   Compressed Picture :

Original Picture memiliki dimensi 1000x1000 pixel serta gambar hasil kompresi menggunakan 80% dari Singular Value yang ada pada Original Picture.

2. **Test Case 2**

Original Picture :

Compressed Picture :



Original Picture memiliki dimensi 3500x1444 pixel serta gambar hasil kompresi menggunakan 70% dari Singular Value yang ada pada Original Picture.

Namun, dari kedua test case diatas, gambar hasil kompresi memiliki berat file yang lebih tinggi (lebih berat) dibandingkan dengan berat file aslinya.

BAB V

KESIMPULAN, SARAN, DAN REFLEKSI

5.1 Kesimpulan

Dari pengerjaan tugas besar ini, kesimpulan yang dapat diambil oleh penulis adalah sebagai berikut.

1. Bahasa pemrograman Python dan PHP sangat berguna dalam pembuatan website, khususnya website yang membutuhkan interaksi antara pengguna dengan program.

2. Aplikasi dari mata kuliah Aljabar Linier dan Geometri terutama pada materi SVD sangat bermanfaat bagi manusia, salah satu implementasinya adalah sebagai alat untuk kompresi gambar.

5.2 Saran

Untuk pengembangan tugas besar ini, tim penulis memiliki beberapa saran sebagai berikut.

1. Algoritma yang digunakan dalam tugas besar ini masih terdapat banyak kekurangan sehingga masih memungkinkan untuk melakukan efisiensi dengan beberapa library yang tersedia.

2. Spesifikasi tugas besar ini dapat diperjelas lagi untuk mencegah adanya multitafsir dan kesalahpahaman dalam proses pembuatan program.

3. Program ini dapat dikembangkan lebih lanjut dan sebaiknya dapat dipublikasikan agar memiliki kebermanfaatan yang lebih besar.

5.3 Refleksi

Dalam proses pengerjaan tugas besar ini, penulis dapat merefleksikan beberapa hal sebagai berikut.

1. Pembagian tugas sudah ditetapkan dengan rinci dan jelas sehingga pada pelaksanaannya sudah tidak ada anggota yang bingung atas tugasnya.

2. Komunikasi antar anggota berjalan dengan baik sehingga tidak ada miskomunikasi dalam pengerjaan tugas besar ini.

3. Ketika terdapat error pada program, tim penulis langsung saling berkoordinasi untuk mencari solusinya.

DAFTAR REFERENSI

https://saintif.com/perkalian-matriks/

https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-18-Nilai-Eigen-dan-Vektor-Eigen-Bagian1.pdf

https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-19b-Singular-value-decomposition.pdf