Seri bahan kuliah Algeo 22

Singular Value Decomposition (SVD)

(Bagian 2)

Update 2023

Bahan kuliah IF2123 Aljabar Linier dan Geometri

Oleh: Rinaldi Munir

Program Studi Teknik Informatika
STEI-ITB

Cara 2:

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 $\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_m}$ yang berkoresponden dengan nilai-nilai eigen dari AA^T. Normalisasi $\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_m}$ 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 $\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_n}$ yang berkoresponden dengan nilai-nilai eigen dari A^TA. Normalisasi $\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_n}$ dengan cara setiap komponen vektornya dibagi dengan panjang vektor. Diperoleh matriks V. Transpose-kan matriks V sehingga menjadi V^T.
5. Bentuklah matriks $\sum$ berukuran m x n dengan elemen-elemen diagonalnya adalah nilai-nilai singular tidak nol dari matriks A dengan susunan dari besar ke kecil. Nilai singular di dalam $\sum$ adalah akar pangkat dua dari nilai-nilai eigen yang tidak nol dari A^TA.
6. Maka, $A = U\sum V^T$

Contoh 4: Faktorkan kembali matriks $A = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}$ dengan metode SVD menggunakan cara kedua.

Penyelesaian:

(1) Singular kiri:

$$AA^T = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix}$$

Nilai-nilai eigen dari AA^T adalah $\lambda_1 = 12$ dan $\lambda_2 = 10$ (terurut dari besar ke kecil)
Jadi rank(A) = 2

(2) Menentukan matriks U

$$(\lambda I - AA^T)\mathbf{x} = 0 \rightarrow \begin{bmatrix} \lambda - 11 & -1 \\ -1 & \lambda - 11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Untuk λ= 12,diperoleh

$\begin{bmatrix} \lambda - 11 & -1 \\ -1 & \lambda - 11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

SPL: $x_1 - x_2 = 0$ dan $-x_1 + x_2 = 0$ $\rightarrow x_1 = x_2$, misal $x_1 = t$, maka $x_2 = t$, $t \in R$

Vektor eigen: $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = t \begin{bmatrix} 1 \\ 1 \end{bmatrix} \rightarrow \mathbf{u_1} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

Untuk λ= 10,diperoleh

$\begin{bmatrix} \lambda - 11 & -1 \\ -1 & \lambda - 11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \rightarrow \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

SPL: $-x_1 - x_2 = 0$ dan $-x_1 - x_2 = 0$ $\rightarrow x_1 = -x_2$, misal $x_1 = s$, maka $x_2 = -s$, $s \in R$

Vektor eigen: $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = s \begin{bmatrix} 1 \\ -1 \end{bmatrix} \rightarrow \mathbf{u_2} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

Normalisasi $\mathbf{u_1}$ dan $\mathbf{u_2}$: $\hat{\mathbf{u}}_1 = \frac{\mathbf{u}_1}{\|\mathbf{u}_1\|} = \frac{(1,1)}{\sqrt{2}} = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}$ dan $\hat{\mathbf{u}}_2 = \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} = \frac{(1,-1)}{\sqrt{2}} = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} \end{bmatrix}$

Diperoleh matriks U:

$$U = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}$$

(3) Singular kanan:

$$A^TA = \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 10 & 0 & 2 \\ 0 & 10 & 4 \\ 2 & 4 & 2 \end{bmatrix}$$

Nilai-nilai eigen dari A^TA adalah $\lambda_1 = 12$, $\lambda_2 = 10$ dan $\lambda_3 = 0$ (terurut dari besar ke kecil)
Nilai-nilai singular dari nilai eigen yang tidak nol adalah $\sigma_1 = \sqrt{12}$, $\sigma_2 = \sqrt{10}$

(4) Vektor-vektor eigen yang bersesuaian dengan setiap nilai eigen tersebut adalah:

$$\mathbf{v_1} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}, \quad \mathbf{v_2} = \begin{bmatrix} 2 \\ -1 \\ 0 \end{bmatrix}, \quad \mathbf{v_3} = \begin{bmatrix} 1 \\ 2 \\ -5 \end{bmatrix}$$

$$\begin{pmatrix} 2 & 0 & -2 \\ 0 & 2 & -4 \\ -2 & -4 & 10 \end{pmatrix} \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$x_1 - x_3 = 0$ $\qquad x_3 = t$

$x_2 - 2x_3 = 0$

$x_1 = t$
$x_2 = 2t$
$x_3 = t$

Normalisasi $\mathbf{v_1}$, $\mathbf{v_2}$, dan $\mathbf{v_3}$:

$$\hat{\mathbf{v}}_1 = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} = \frac{(1,2,1)}{\sqrt{6}} = \begin{bmatrix} \frac{\sqrt{6}}{6} \\ \frac{\sqrt{6}}{3} \\ \frac{\sqrt{6}}{6} \end{bmatrix}, \hat{\mathbf{v}}_2 = \frac{\mathbf{v}_2}{\|\mathbf{v}_2\|} = \frac{(2,-1,0)}{\sqrt{5}} = \begin{bmatrix} \frac{2\sqrt{5}}{5} \\ \frac{-\sqrt{5}}{5} \\ 0 \end{bmatrix}, \hat{\mathbf{v}}_3 = \frac{\mathbf{v}_3}{\|\mathbf{v}_3\|} = \frac{(1,2,-5)}{\sqrt{30}} = \begin{bmatrix} \frac{\sqrt{30}}{30} \\ \frac{\sqrt{30}}{15} \\ \frac{-\sqrt{30}}{6} \end{bmatrix}$$

Matriks *V* adalah:

$$V = \begin{bmatrix} \frac{\sqrt{6}}{6} & \frac{2\sqrt{5}}{5} & \frac{\sqrt{30}}{30} \\ \frac{\sqrt{6}}{3} & \frac{-\sqrt{5}}{5} & \frac{\sqrt{30}}{15} \\ \frac{\sqrt{6}}{6} & 0 & \frac{-\sqrt{30}}{6} \end{bmatrix} \text{ dan } V^T = \begin{bmatrix} \frac{\sqrt{6}}{6} & \frac{\sqrt{6}}{3} & \frac{\sqrt{6}}{6} \\ \frac{2\sqrt{5}}{5} & \frac{-\sqrt{5}}{5} & 0 \\ \frac{\sqrt{30}}{30} & \frac{\sqrt{30}}{15} & \frac{-\sqrt{30}}{6} \end{bmatrix}$$

(5) Matriks Σ yang berukuran 2 x 3 adalah $\Sigma = \begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \end{bmatrix}$

(6) Jadi, $A = U\Sigma V^T$

$$\underbrace{\begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}}_{\substack{A \\ 2 \times 3}} = \underbrace{\begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}}_{\substack{U \\ 2 \times 2}} \underbrace{\begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \end{bmatrix}}_{\substack{\Sigma \\ 2 \times 3}} \underbrace{\begin{bmatrix} \frac{\sqrt{6}}{6} & \frac{\sqrt{6}}{3} & \frac{\sqrt{6}}{6} \\ \frac{2\sqrt{5}}{5} & \frac{-\sqrt{5}}{5} & 0 \\ \frac{\sqrt{30}}{30} & \frac{\sqrt{30}}{15} & \frac{-\sqrt{30}}{6} \end{bmatrix}}_{\substack{V^T \\ 3 \times 3}}$$

Hasilnya sama seperti pada cara pertama sebelumnya

$$\begin{bmatrix} -2 & -1 & 2 \\ 2 & 1 & -2 \end{bmatrix}$$

① cari u

$$AA^T = \begin{bmatrix} -2 & -1 & 2 \\ 2 & 1 & -2 \end{bmatrix} \begin{bmatrix} -2 & 2 \\ -1 & 1 \\ 2 & -2 \end{bmatrix}$$

$$= \begin{bmatrix} 9 & -9 \\ -9 & 9 \end{bmatrix}$$

$\det(\lambda I - AA^T)x = 0$

$$\left| \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 9 & -9 \\ -9 & 9 \end{bmatrix} \right|$$

$$\begin{vmatrix} \lambda - 9 & 9 \\ 9 & \lambda - 9 \end{vmatrix}$$

$(\lambda - 9)^2 - 81 = 0$

$\lambda^2 - 18\lambda = 0$

$\lambda(\lambda - 18) = 0$

$\lambda_1 = 0$ dan $\lambda_2 = 18$

untuk $\lambda = 18$: $\begin{bmatrix} 9 & 9 \\ 9 & 9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

$$\begin{bmatrix} 9 & 9 & 0 \\ 9 & 9 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$x_1 + x_2 = 0$

$x_2 = t$

$x_1 = -t$

$u_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$

$\hat{u}_1 = \frac{u}{\|u\|} = \frac{[-1, 1]}{\sqrt{2}} = \begin{bmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$

② cari v

$$A^T A = \begin{bmatrix} -2 & 2 \\ -1 & 1 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} -2 & -1 & 2 \\ 2 & 1 & -2 \end{bmatrix}$$

$$= \begin{bmatrix} 8 & 4 & -8 \\ 4 & 2 & -4 \\ -8 & -4 & 8 \end{bmatrix}$$

$\det(\lambda I - A^T A)x = 0$

$$\left| \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 8 & 4 & -8 \\ 4 & 2 & -4 \\ -8 & -4 & 8 \end{bmatrix} \right|$$

$$\begin{vmatrix} \lambda - 8 & -4 & 8 \\ -4 & \lambda - 2 & 4 \\ 8 & 4 & \lambda - 8 \end{vmatrix} = 0$$

$$\lambda - 8 \begin{vmatrix} \lambda - 2 & 4 \\ 4 & \lambda - 8 \end{vmatrix} - (-4) \begin{vmatrix} -4 & 4 \\ 8 & \lambda - 8 \end{vmatrix} + 8 \begin{vmatrix} -4 & \lambda - 2 \\ 8 & 4 \end{vmatrix}$$

$\lambda - 8((\lambda - 2)(\lambda - 8) - 16) - (-4)(-4\lambda + 32 - 32) + 8(-16 - 8(\lambda - 2))$

$\lambda - 8(\lambda^2 - 10\lambda) - (-4)(-4\lambda) + 8(-16 - 8\lambda + 16)$

$\lambda - 8(\lambda^2 - 10\lambda) - 16\lambda - 64\lambda = 0$

$\lambda - 8(\lambda^2 - 10\lambda) - 80\lambda = 0$

$\lambda^3 - 18\lambda^2 + 80\lambda - 80\lambda = 0$

$\lambda^3 - 18\lambda^2 = 0$

$\lambda^2(\lambda - 18) = 0$

$\lambda_1 = 0$ dan $\lambda_2 = 18$

untuk $\lambda = 18$

$$\left(18 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 8 & 4 \\ 4 & 2 \\ -8 & -4 \end{bmatrix} \right.$$

$$\begin{bmatrix} 10 & -4 & 8 & 0 \\ -4 & 16 & 4 & 0 \\ 8 & 4 & 10 & 0 \end{bmatrix} = \begin{bmatrix} 1 & -4 & -1 & 0 \\ 10 & -4 & 8 & 0 \\ 8 & 4 & 10 & 0 \end{bmatrix} = \begin{bmatrix} 1 & -4 & -1 & 0 \\ 0 & 36 & 18 & 0 \\ 0 & 36 & 18 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -4 & -1 & 0 \\ 0 & 1 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$x_2 + \frac{1}{2}x_3 = 0$

$x_3 = t$, $x_2 = -\frac{1}{2}t$

$x_1 = 4(-\frac{1}{2}t) + t$

$= -2t + t = -t$

$v_1 = \begin{bmatrix} -1 \\ -\frac{1}{2} \\ 1 \end{bmatrix}$

$$\left(0\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}-\begin{bmatrix}8&4&-8\\4&2&-4\\-8&-4&8\end{bmatrix}\right)\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix}=\begin{pmatrix}0\\0\\0\end{pmatrix}$$

$$\begin{bmatrix}-8&-4&8&0\\-4&-2&4&0\\8&4&-8&0\end{bmatrix}$$

$$=\begin{bmatrix}1&\frac{1}{2}&-1&0\\-8&-4&8&0\\8&4&-8&0\end{bmatrix}$$

$$=\begin{bmatrix}1&\frac{1}{2}&-1&0\\0&0&0&0\\0&0&0&0\end{bmatrix}$$

$x_1+\frac{1}{2}x_2-x_3=0$

$x_3=t$ $\quad x_1=-\frac{1}{2}s+t$

$x_2=s$

$V_2=\begin{bmatrix}1\\0\\1\end{bmatrix}$

$V_3=\begin{bmatrix}-\frac{1}{2}\\1\\0\end{bmatrix}$

normalisasi v_1, v_2, v_3

SVD Tereduksi (*Reduced Singular Value Decomposition*)

- Baris-baris dan kolom bernilai nol pada matriks Σ dapat dihilangkan sehingga menjadi:

$$A = \underbrace{[\mathbf{u}_1 \quad \mathbf{u}_2 \quad \cdots \mathbf{u}_k]}_{U_1} \underbrace{\begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_k \end{bmatrix}}_{\Sigma_1} \underbrace{\begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_k^T \end{bmatrix}}_{V_1} \qquad (4)$$

dan disebut **SVD tereduksi**. Matriks sebelah kiri kita tulis U_1, matriks di tengah kita tulis Σ_1, dan matriks di kanan V_1 sehingga persamaan di atas ditulis sebagai

$$A = U_1 \Sigma_1 V_1^T$$

U_1 berukuran m x k, Σ_1 berukuran k x k, dan V^T berukuran k x n

Matriks Σ_1 memiliki balikan karena semua elemen diagonal utamanya positif

$$A = [\mathrm{u}_1 \quad \mathrm{u}_2 \quad \cdots \mathrm{u}_k] \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_k \end{bmatrix} \begin{bmatrix} \mathrm{v}_1^T \\ \mathrm{v}_2^T \\ \vdots \\ \mathrm{v}_k^T \end{bmatrix}$$

- Jika kita mengalikan matriks U_1, $\sum_1$ dan V_1^T, maka diperoleh:

$$A = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \ldots + \sigma_k \mathbf{u}_k \mathbf{v}_k^T \qquad (5)$$

yang dinamakan **bentuk ekspansi SVD** matriks A

- Dapat ditunjukkan bahwa matriks $M = \mathbf{u}\mathbf{v}^T$ memiliki rank = 1, sehingga bentuk SVD tereduksi mengekspresikan matriks A sebagai kombinasi linier dari k buah matriks yang memiliki rank 1.

Contoh 5. Tentukan bentuk SVD tereduksi dari matriks $A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$

Penyelesaian:

Dari Contoh 3 (lihat Bagian 1), sudah diperoleh

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{6}}{3} & 0 & \frac{1}{\sqrt{3}} \\ \frac{\sqrt{6}}{6} & -\frac{\sqrt{2}}{2} & \frac{1}{\sqrt{3}} \\ \frac{\sqrt{6}}{6} & \frac{\sqrt{2}}{2} & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} \sqrt{3} & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}$$

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

Karena A memiliki rank k = 2, maka bentuk SVD tereduksi dari matriks A adalah

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{6}}{3} & 0 \\ \frac{\sqrt{6}}{6} & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{6}}{6} & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \sqrt{3} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}$$

Bentuk ekspansi SVD matriks A adalah:

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T = \sqrt{3} \begin{bmatrix} \frac{\sqrt{6}}{3} \\ \frac{\sqrt{6}}{6} \\ \frac{\sqrt{6}}{6} \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} + (1) \begin{bmatrix} 0 \\ -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}$$

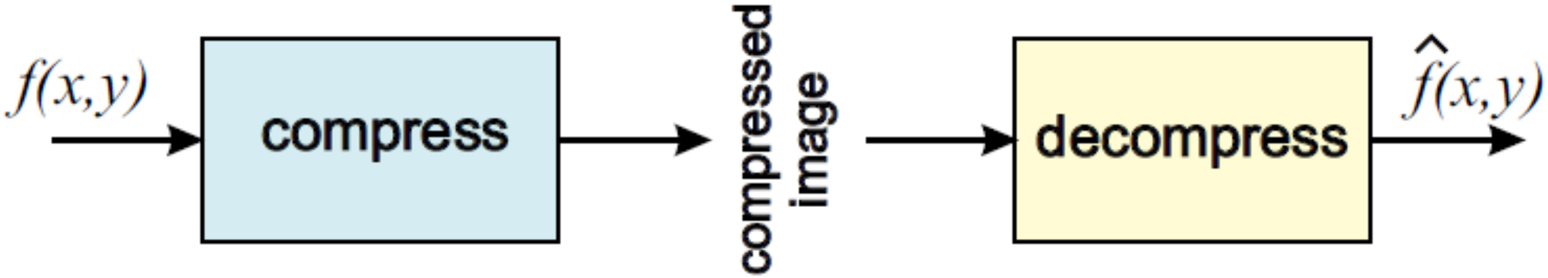
$$= \sqrt{3} \begin{bmatrix} \frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3} \\ \frac{\sqrt{3}}{6} & \frac{\sqrt{3}}{6} \\ \frac{\sqrt{3}}{6} & \frac{\sqrt{3}}{6} \end{bmatrix} + (1) \begin{bmatrix} 0 & 0 \\ -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}$$

Matriks-matriks pada persamaan ekspansi di atas memiliki rank 1

Aplikasi SVD

- Kompresi (pemampatan) gambar dan video (*image and video compression*)
- Pengolahan citra (*image processing*)
- *Machine Learning*
- *Computer vision*
- *Digital watermarking*
- Dll

Pemampatan Citra



- *Image compression* = pemampatan citra, kompresi citra
- *Image decompression* = penirmampatan citra, dekompresi citra

- Citra dimampatkan ketika ia disimpan ke dalam *storage* atau ditransmisikan.

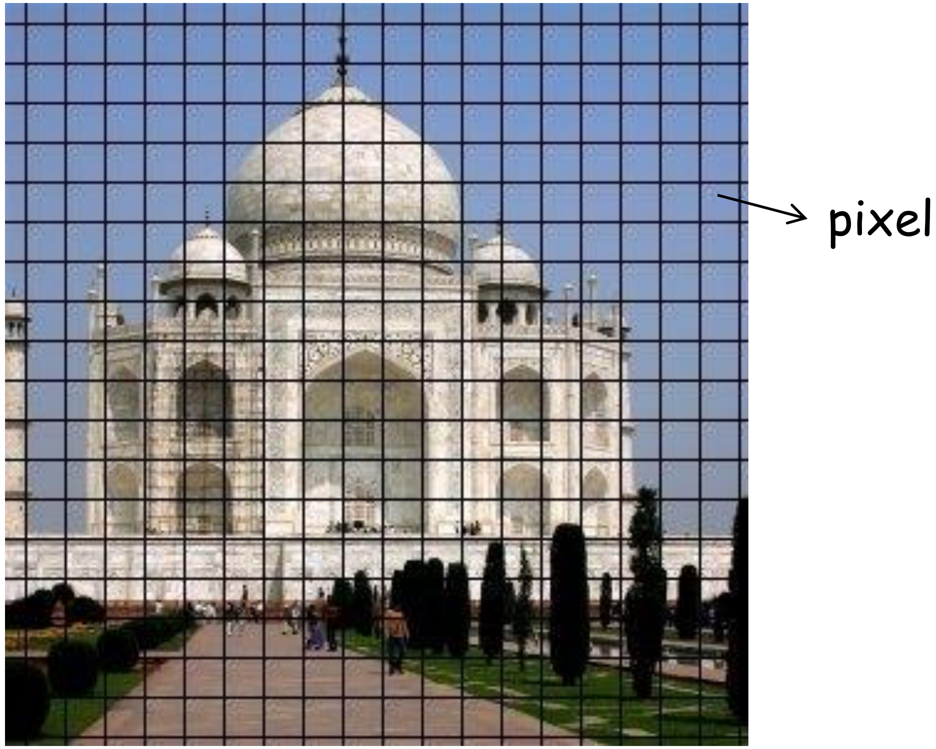
- Citra dinirmampatkan ketika ia ditampilkan ke layar, dicetak ke *printer*, atau disimpan ke dalam dokumen dengan format tidak mampat

- Citra digital direpresentasikan sebagai matriks berukuran M x N

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

- M x N menyatakan ukuran citra
- Setiap elemen matriks menyatakan sebuah *pixel* (*picture element*).
- Setiap pixel dinyatakan dalam sejumlah bit (atau byte)
- Pada citra biner, satu pixel = 1 bit (0 atau 1 saja, hitam atau putuh)
- Pada citra *grayscale*, satu pixel = 8 bit (1 byte)
- Pada citra berwarna RGB, satu pixel = 24 bit (3 byte)

- Citra dengan ukuran 1200 x 1500 berarti memiliki 1200 x 1500 pixel = 1.800.000 pixel



Jadi, secara umum dikenal 3 jenis citra digital:

1. Citra biner (1 *pixel* = 1 bit)

   Graylevel hanya 0 dan 1 (hitam dan putih)

2. Citra *grayscale* (1 *pixel* umumnya 8 bit)

   Graylevel dari 0 sampai 255 (hitam ke putih)

3. Citra berwarna (24-bit RGB)

   Terdiri dari tiga kanal warna: *red* (R), *green* (G), dan b*lue*(B)

   Graylevel pada setiap kanal warna panjangnya 8 bit

Color image (24-bit RGB)

Grayscale image (8-bit)

Binary image (1-bit)

Mengapa citra perlu dimampatkan?

- Representasi citra digital membutuhkan memori yang besar.
- Pemampatan citra adalah metode untuk mereduksi redundansi pada representasi citra sehingga dapat mengurangi kebutuhan memori untuk ruang penyimpanan.
- Citra dimampatkan tanpa mengurangi kualitas citra secara visual
- Tujuan:
  1. Mengurangi kebutuhan ruang penyimpanan sembari tetap mempertahankan kualitas citra secara visual. (Gonzalez, Woods and Eddins, 2017).
  2. Merepresentasikan citra dengan kualitas yang hampir sama dengan citra aslinya namun dalam bentuk yang lebih kompak.

Dapatkah anda melihat perbedaan kualitas hasil pemampatan?

Original image
(not compressed)

Compressed image

peppers.bmp, 256 x 256
(193 KB)

peppers.jpg, 256 x 256
(31 KB), JPEG Quality = 5

peppers2.jpg, 256 x 256
(24 KB), JPEG Quality = 1

- Misalkan sebuah citra berwarna (RGB) berukuran 1200x1600

  Kebutuhan ruang penyimpanan:

  1200 x 1600 x 3 byte = 5760000 byte

  = 5,760 *Kbyte*

  = 5.76 *Mbyte*

- Misalkan sebuah film digital dengan resolusi 760x480, 30 frame/sec, selama 2 jam.

  Kebutuhan ruang penyimpanan:

$$30\frac{frame}{\sec}\times(760\times480)\frac{pixels}{frame}\times3\frac{bytes}{pixel}=31{,}104{,}000 bytes/\sec$$

$$31{,}104{,}000\times\frac{bytes}{\sec}\times(60\times60)\frac{\sec}{hour}\times2hours=2.24\times10^{11}bytes$$

$$=224GByte.$$

Sangat besar!!

Aplikasi pemampatan citra

1. Penyimpanan data di dalam media sekunder (*storage*)

   Citra mampat membutuhkan memori di dalam *storage* yang lebih sedikit dibandingkan dengan citra yang tidak mampat.

   Contoh: file citra berformat JPEG/JPG versus citra berformat BMP

2. Pengiriman data (*data transmission*) pada saluran komunikasi data

   Citra mampat membutuhkan waktu pengiriman yang lebih singkat dibandingkan dengan citra tidak mampat.

   Contoh: pengiriman gambar via email, *videoconference*, via satelit luar angkasa, mengunduh gambar dari internet, dan sebagainya.

Aplikasi SVD pada Pemampatan Citra

- SVD dapat digunakan untuk pemampatan citra.
- Prinsip dasarnya adalah menggunakan SVD tereduksi (*reduced singular value decomposition*).
- Sebuah citra A (yang representasinya adalah matriks) dinyatakan sebagai bentuk SVD tereduksi:

$$A = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \ldots + \sigma_k \mathbf{u}_k \mathbf{v}_k^T \qquad (6)$$

  Simpan nilai-nilai σ_j, $\mathbf{u}_j$ dan $\mathbf{v}_j$ ini ke dalam storage (memori).

- Matriks A dapat direkonstruksi kembali dari persamaan 6 di atas. Karena $\mathbf{u}_j$ memiliki *m* buah bilangan dan $\mathbf{v}_j$ memiliki *n* buah bilangan, maka metode kompresi ini membutuhkan penyimpanan $km + kn + k = k(m + n + 1)$ bilangan di dalam memori (storage).

- Misalkan nilai-nilai singular $\sigma_{r+1}, \sigma_{r+2}, \ldots, \sigma_k$ sangat kecil sehingga dapat dibuang dari persamaan (6), sehingga citra A diaproksimasi menjadi

$$A_r = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \cdots + \sigma_r \mathbf{u}_r \mathbf{v}_r^T \quad (7)$$

- Persamaan (7) ini dinamakan aproksimasi A dengan rank *r*.
- Matriks yang dibutuhkan sekarang hanya perlu menyimpan $rm + rn + r = r(m + n + 1)$ buah bilangan .

- Jadi, citra hasil pemampatan dengan SVD adalah berupa aproksimasi dengan berbagai nilai rank r.

- Sebagai contoh, untuk citra berukuran 1000 x 1000 dengan rank r = 100, kita hanya perlu menyimpan 100(1000 + 1000 + 1) = 200.100 buah bilangan. Bandingkan jika tanpa pemampatan maka kita perlu menyimpan 1000 x 1000 = 1.000.000 buah bilangan (nilai-nilai pixel di dalam citra).

- Gambar di bawah ini memperlihatkan hasil pemampatan dengan SVD untuk berbagai nilai rank.

Rank 4 Rank 10 Rank 20 Rank 50 Rank 128

- Semakin besar nilai rank r, kualitas citra hasil aproksimasi (hasil pemampatan) semakin bagus, namun semakin banyak memori yang diperlukan untuk menyimpan nilai-nilai σ_j, $\mathbf{u}_j$ dan $\mathbf{v}_j$ ke dalam storage.

Tugas Besar Tahun 2021

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Tugas Besar 2 IF 2123 Aljabar Linier dan Geometri
Aplikasi Nilai Eigen dan Vektor Eigen dalam Kompresi Gambar
Semester I Tahun 2021/2022

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

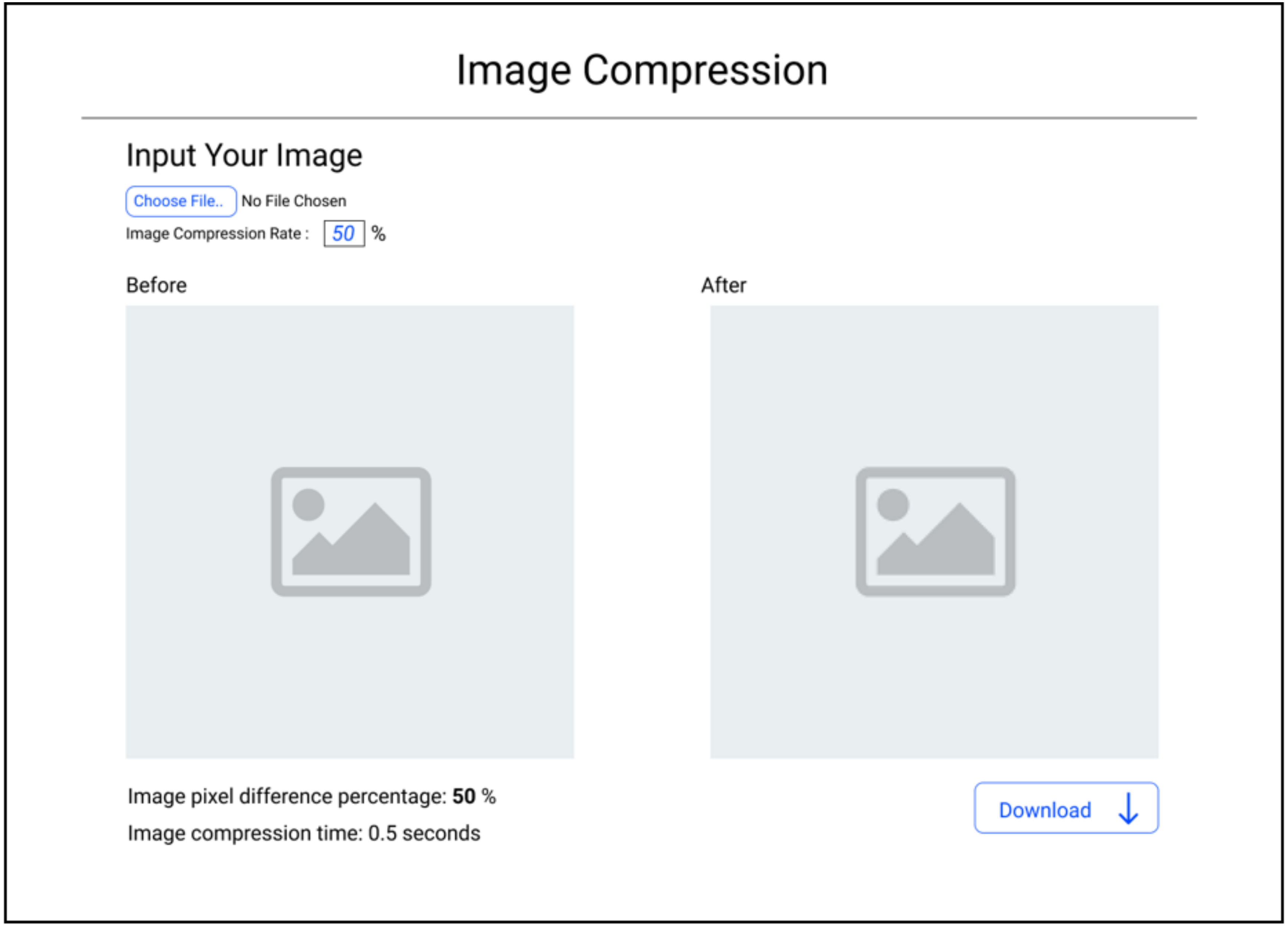
Sumber : [Understanding Compression in Digital Photography (lifewire.com)]

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.

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.



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

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.

- Contoh luaran program dari kelompok Ng Kyle (asisten Algeo 2022):

IF2123 Aljabar Geometri

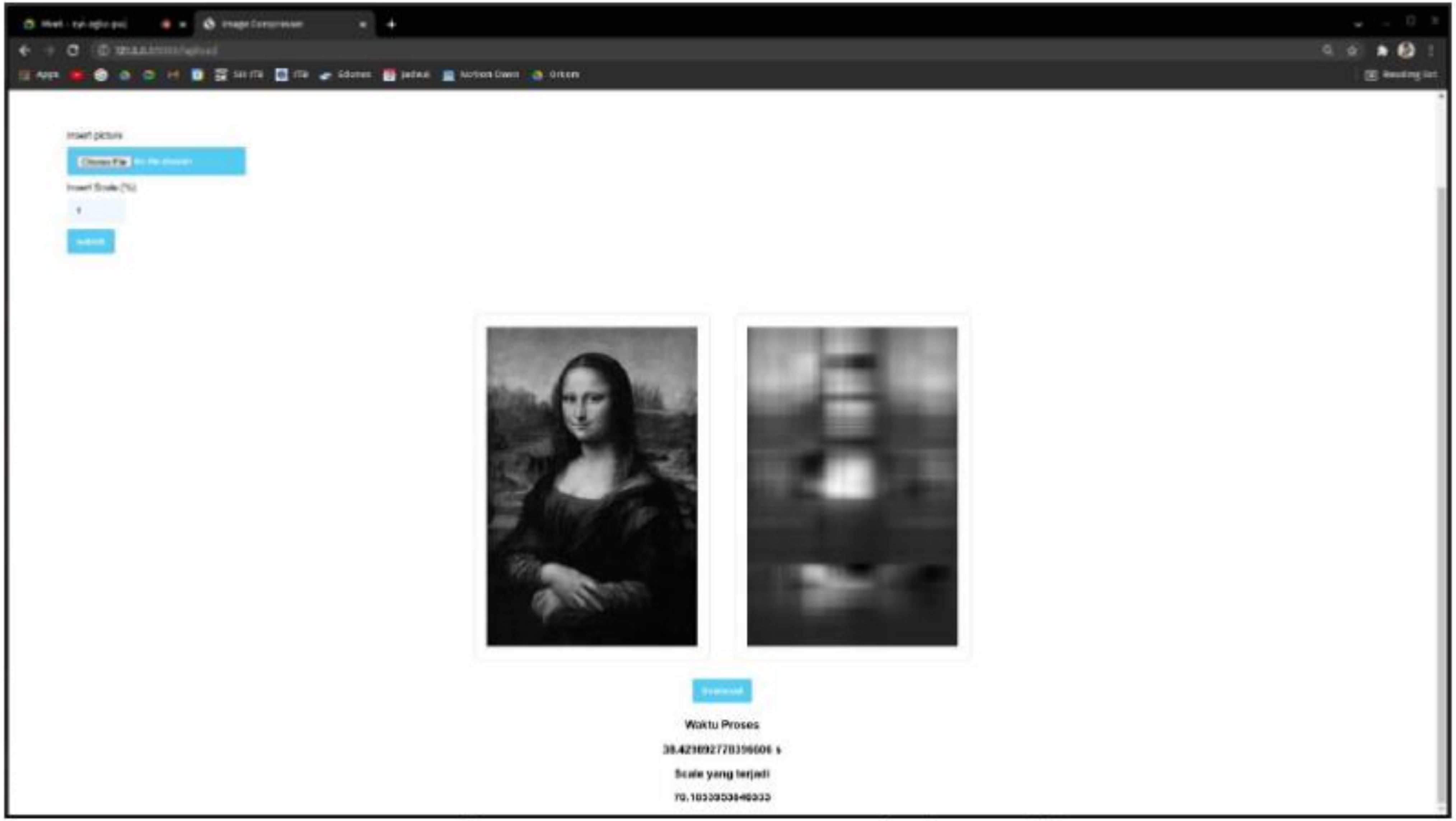
Web-App Kompresi Gambar

metode SVD Matrix Decomposition

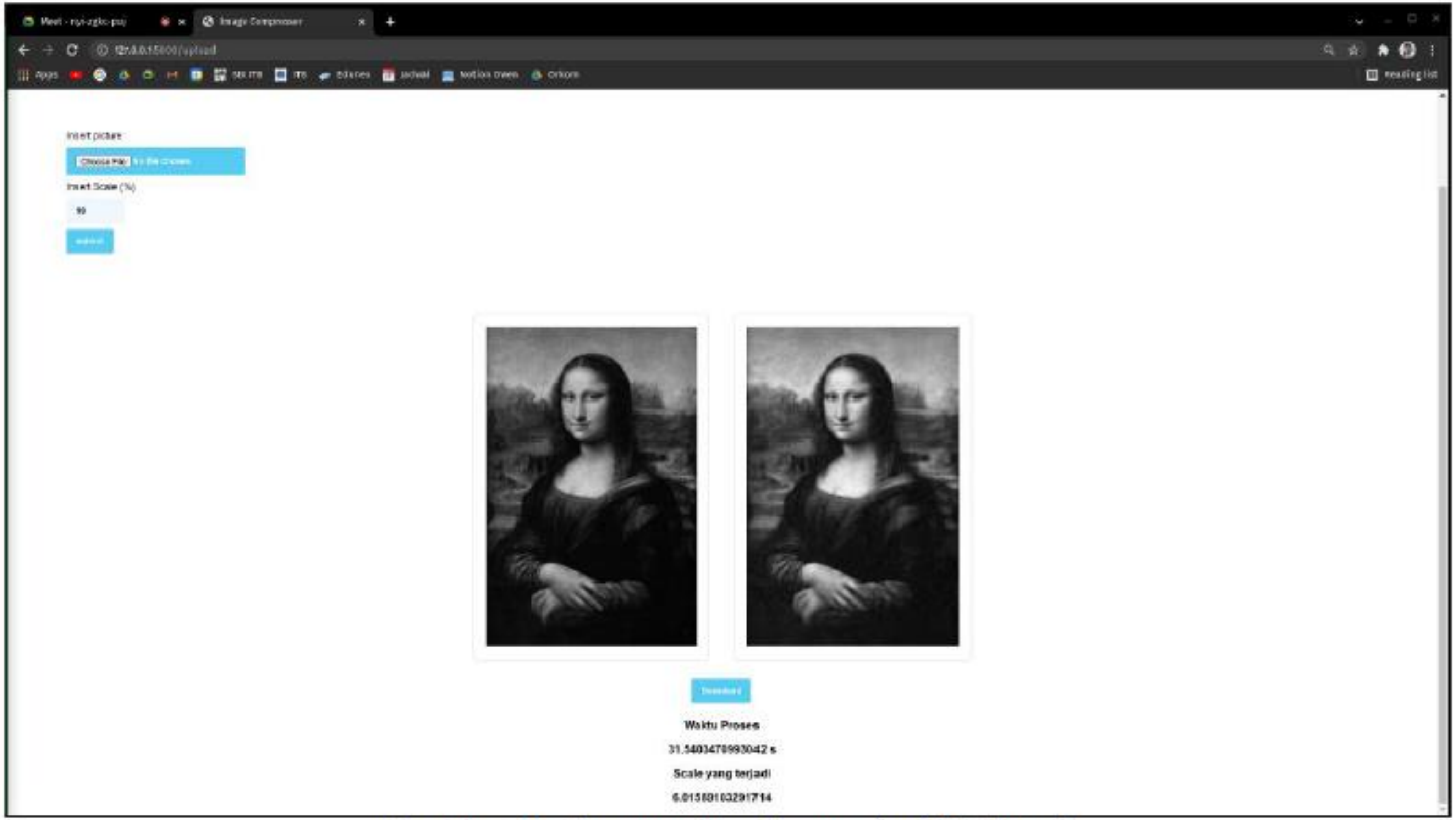
Bryan Bernigen
13520034

Ng Kyle
13520040

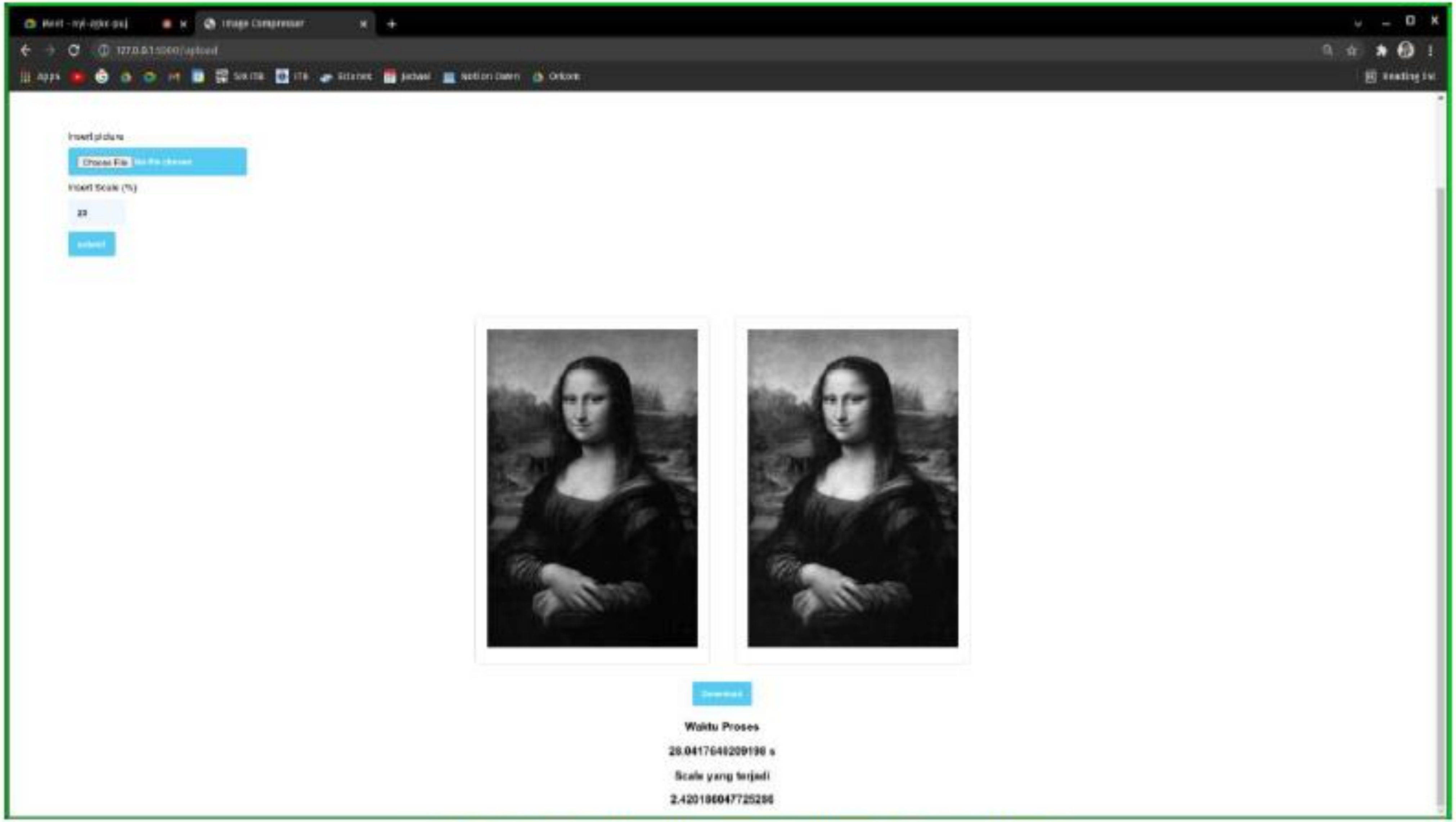
Muhammad Risqi Firdaus
13520043



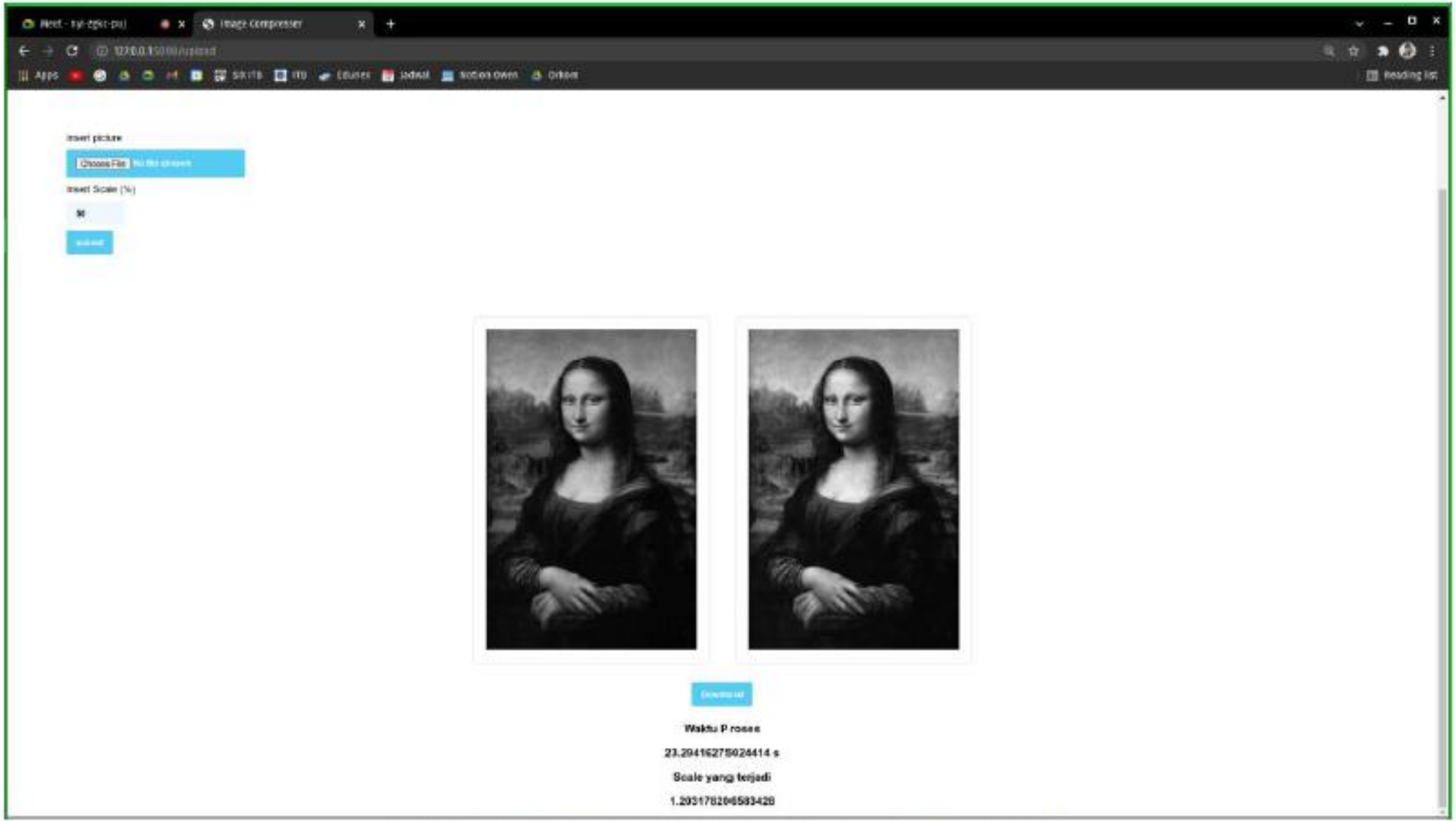
Gambar 10. Compression Grayscale (1% Rank)



Gambar 11. Compression Grayscale (10% Rank)



Gambar 12. Compression Greyscale (25% Rank).

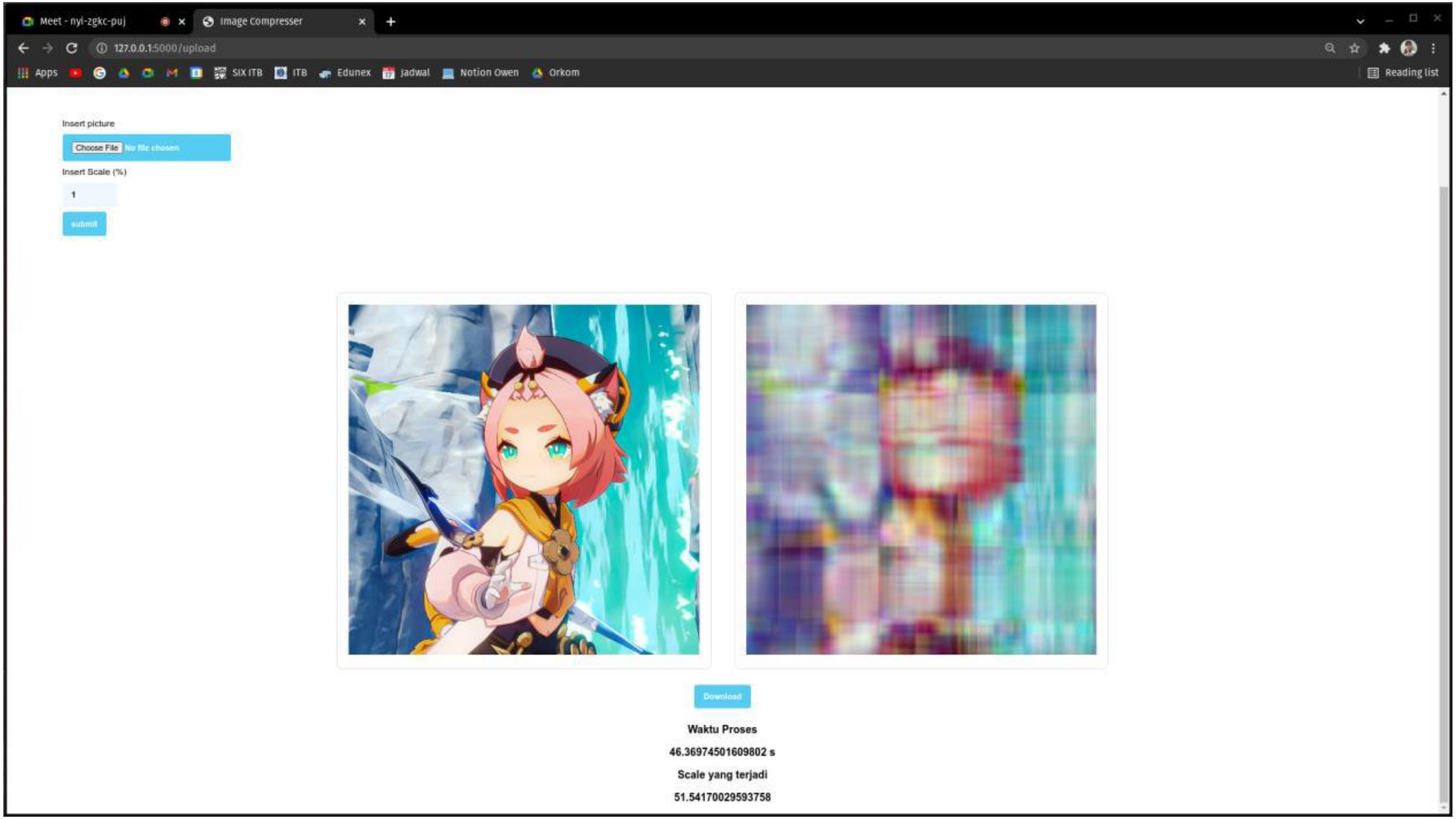


Gambar 13. Compression Greyscale (50% Rank)

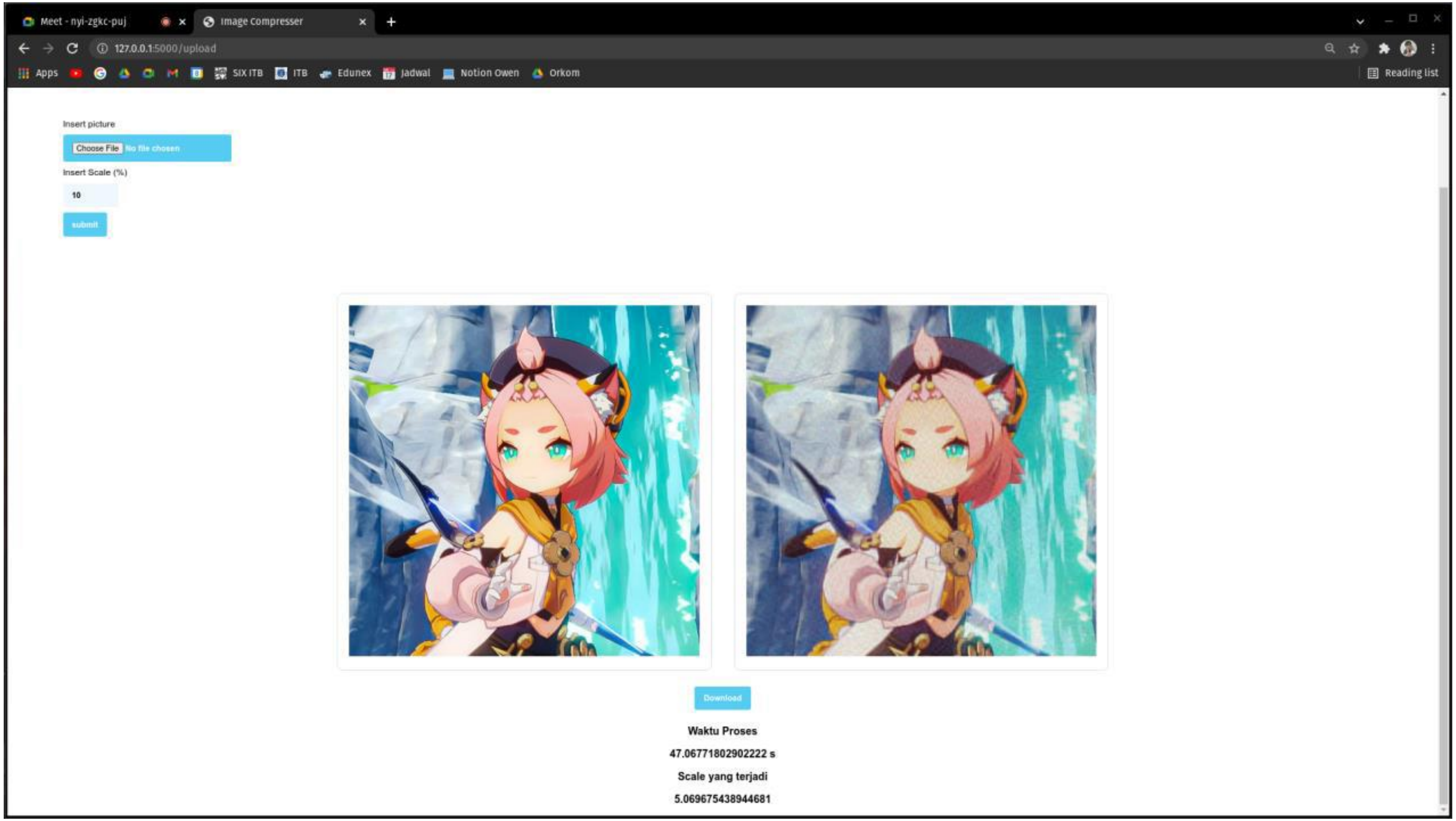
Berikut merupakan data hasil compression gambar tiap ranknya :

Image Rank Used (%)	Size After Compression (KB)	Processing Time (seconds)	Compression Ratio
1	9.65	38.42	70
10	19.18	31.54	6
25	22.76	28.04	2.4
50	24.96	23.29	1.2

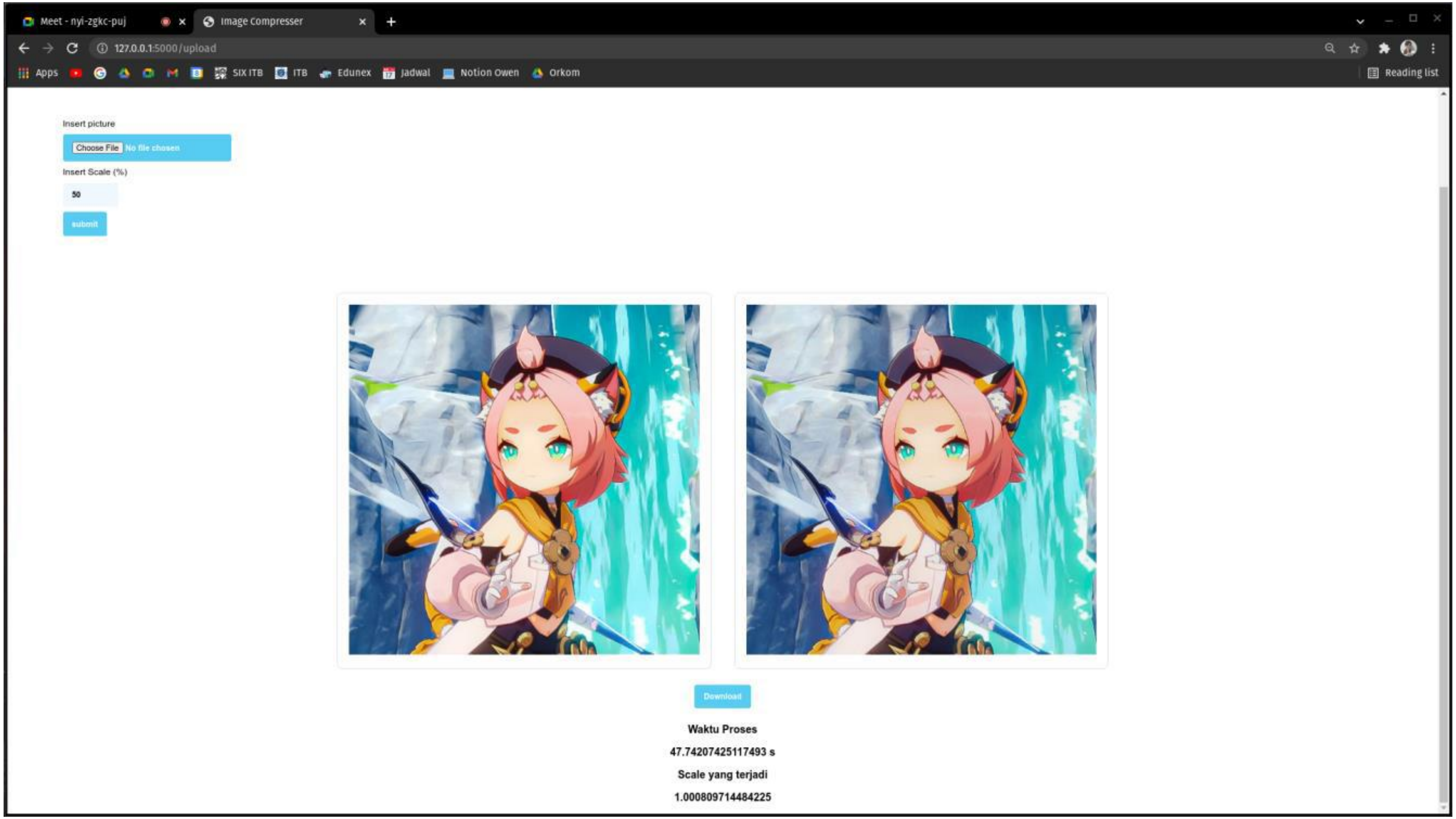
Catatan: ukuran file gambar semula, diona.jpg, adalah 26,6 KB



Gambar 14. Compression RGB (1% Rank)



Gambar 15. Compression RGB (5% Rank)



Gambar 17. Compression RGB (50% Rank)

Berikut merupakan data gambar hasil compression:

Image Rank Used (%)	Size After Compression (KB)	Processing Time (seconds)	Compression Ratio
1	31.67	46.37	51.54
10	56.92	47.07	5.1
25	66.47	47.23	2
50	69.08	47.74	1

Catatan: ukuran file gambar semula, diona.jpg , adalah 98.62 KB

Berikut merupakan data hasil compression gambar tiap ranknya :

Image Rank Used (%)	Size After Compression (KB)	Processing Time (seconds)	Compression Ratio
1	9.65	38.42	70
10	19.18	31.54	6
25	22.76	28.04	2.4
50	24.96	23.29	1.2

Catatan: Ukuran gambar semula, mona.jpg, adalah 26.6 KB.

Sumber:

1. Howard Anton & Chris Rores, *Elementary Linear Algebra, 10th Edition*
2. Gregoria Ariyanti, *Dekomposisi Nilai Singular dan Aplikasinya*, Program Studi Pendidikan Matematika Universitas Widya Mandala Madiun
3. Laporan Tubes 2 Algeo tahun 2022,Kelompok Ng Kyle dkk

TAMAT