TUGAS BESAR 2 IF 2123

ALJABAR LINIER DAN GEOMETRI

APLIKASI NILAI EIGEN DAN VEKTORI EIGEN DALAM KOMPRESI GAMBAR



DIBUAT OLEH

13520028 Timothy Stanley Setiawan

13520054 Farrel Farandieka Fibriyanto

13520082 Jeremy Rionaldo Pasaribu

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA INSTITUT TEKNOLOGI BANDUNG

Semester I Tahun 2021/2022

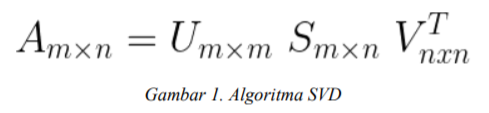
BAB 1: Deskripsi Masalah

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.

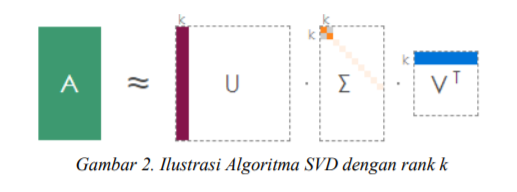
Kompresi gambar merupakan suatu tipe kompresi data yang dilakukan pada gambar digital. Dengan kompresi gambar, suatu file gambar digital dapat dikurangi ukuran filenya dengan baik tanpa mempengaruhi kualitas gambar secara signifikan. Terdapat berbagai metode dan algoritma yang digunakan untuk kompresi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



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.



Matriks U adalah matriks yang kolomnya terdiri dari vektor eigen ortonormal dari matriks AAT. 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 ATA. Matriks ini menyimpan informasi yang penting terkait kolom-kolom matriks awal, dengan informasi terpenting disimpan dalam baris pertama.



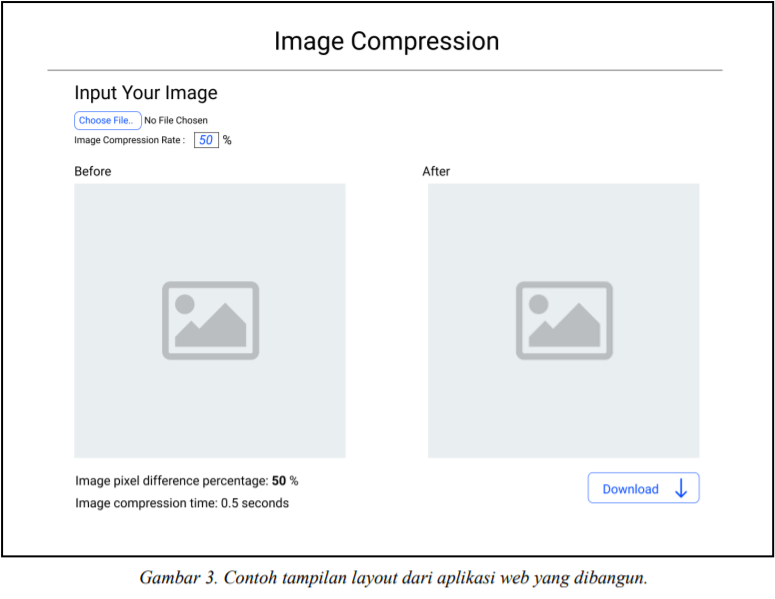
Dapat dilihat di gambar di atas bahwa dapat direkonstruksi gambar dengan banyak singular values k dengan mengambil kolom dan baris sebanyak k dari U dan V serta singular value sebanyak k dari S atau Σ terurut dari yang terbesar. Kita dapat mengaproksimasi suatu gambar yang mirip dengan gambar aslinya 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.

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. 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.

Membuatlah 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 2: Teori Dasar

* 1. Singular Value Decomposition (SVD)

Di dalam materi nilai eigen dan vektor eigen, pokok bahasan diagonalisasi, matriks bujursangkar A berukuran n x n dapat difaktorkan menjadi :

A = EDE-1

dalam hal ini,

E adalah matriks yang kolom-kolomnya adalah basis ruang eigen dari matriks A,

E = (e1| e2| … | en)

D adalah matriks diagonal sedemikian sehingga

D = E-1AE

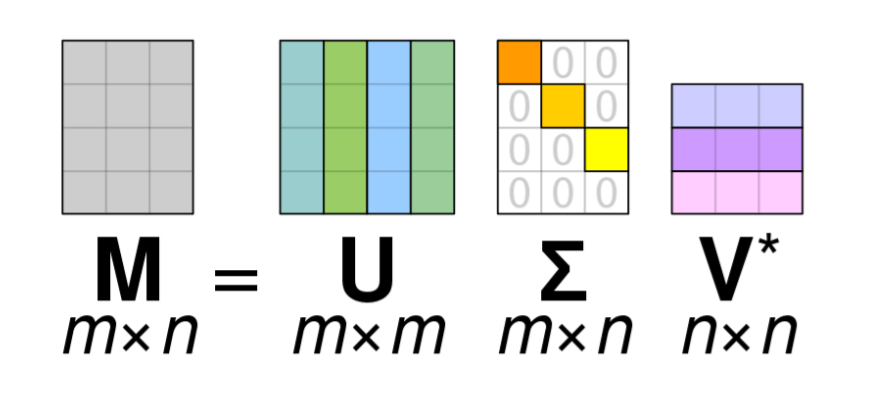
Namun, muncul permasalahan ketika memfaktorkan matrix non-bujursangkar berukuran m x n yang tidak memiliki nilai eigen. Oleh sebab itu untuk matriks non-bujursangkar, pemfaktorannya menggunakan metode singular decomposition value (SVD). SVD memfaktorkan matriks A berukuran m x n menjadi matriks U, ∑, dan V sedemikian sehingga

A = U∑VT

U = matriks ortogonal m x m,

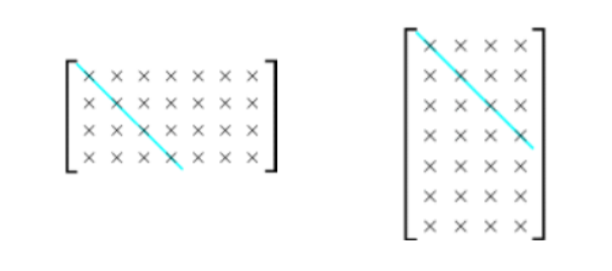
V = matriks orthogonal n x n

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



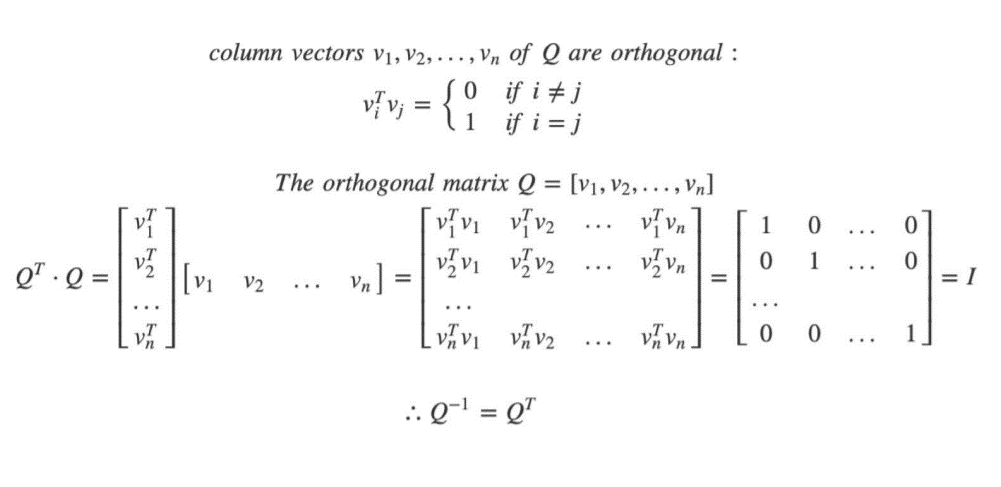
* + 1. Diagonal utama matriks m x n

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 di mulai dari sudut kiri atas terus ke bawah matriks sejauh mungkin.



* + 1. Matriks ortogonal

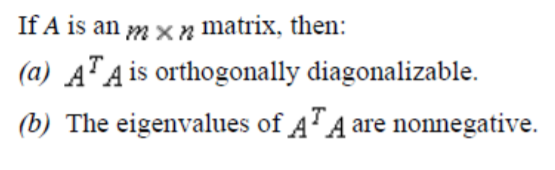
Matriks ortogonal adalah matriks yang kolom-kolomnya adalah vektor yang saling orthogonal satu sama lain (hasil kali titik sama dengan 0). Jika vektor-vektor kolom tersebut merupakan vektor satuan, maka matriks ortogonal tersebut dinaakan juga matriks ortonormal. Vektor satuan adalah vektor yang dinormalisasi 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 v1 , v2 , …, vm, maka vi ⋅ vj = 0 untuk i ≠ j. Atau, dapat juga dikatakan bahwa Q adalah matriks ortogonal jika QTQ= I, dalam hal ini I adalah matriks identitas.

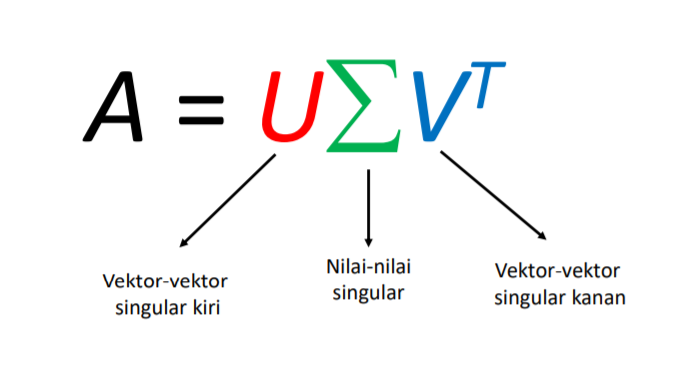


* + 1. Nilai-nilai singular matriks

Misalkan A adalah matriks m x n. Jika λ1 , λ2 , …, λn adalah nilai-nilai eigen dari ATA, maka σ1 = √ λ1 , σ2 = √ λ2 , …, σn = √ λn disebut nilai-nilai singular dari matriks A. Diasumsikan λ1 ≥ λ2 ≥ … ≥ λn ≥ 0 sehingga σ1 ≥ σ2 ≥ … ≥ σn ≥ 0

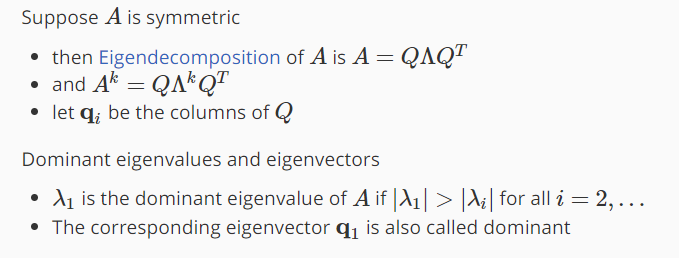
**Teorema**



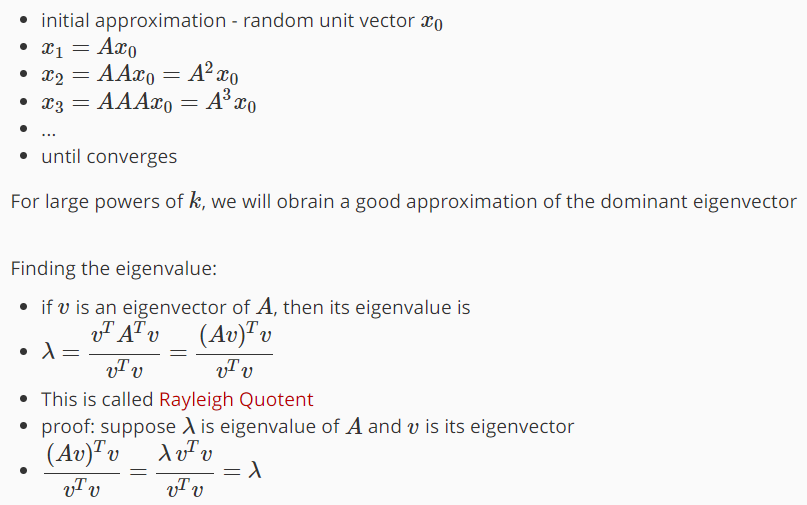


* + 1. Langkah-langkah SVD mendekomposisi Amxn menjadi U, ∑, dan V:
* Untuk vektor singular kiri, hitung nilai-nilai eigen dari AAT. Rank(A) = k = banyaknya nilai-nilai eigen tidak nol dari AAT.
* Tentukan vektor-vektor eigen u1 , u2 , …, um yang berkoresponden dengan nilainilai eigen dari AAT . Normalisasi u1 , u2 , …, um dengan cara setiap komponen vektornya dibagi dengan panjang vektor. Diperoleh matriks U.
* Untuk vektor singular kanan, hitung nilai-nilai eigen dari ATA lalu tentukan nilai-nilai singularnya.
* Tentukan vektor-vektor eigen v1 , v2 , …, vn yang berkoresponden dengan nilainilai eigen dari ATA. Normalisasi v1 , v2 , …, vn dengan cara setiap komponen vektornya dibagi dengan panjang vektor. Diperoleh matriks V. Transpose-kan matriks V sehingga menjadi VT.
* Bentuklah matriks ∑ 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 ∑ adalah akar pangkat dua dari nilai-nilai eigen yang tidak nol dari ATA.
* Maka, A = U∑VT
  1. Power Iteration

Untuk memdapatkan nilai dan vektor eigen kita dapat memanfaatkan metode Power iteration. Power iteration adalah metode aljabar liner untuk mendekati nilai eigen dan vektor eigen dari sebuah matrix.



Berikut ini implementasi untuk mencari nilai dan vektor eigen menggunkan metode power iteration



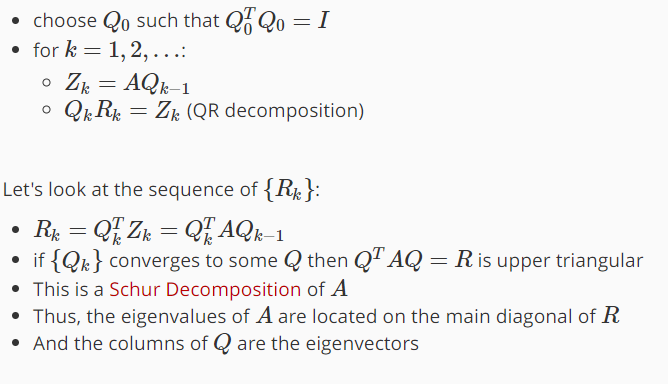
* + 1. Orthogonal Iteration

Seperti yang kita tahu, vektor eigen itu memiliki sifat yang saling orthogonal sehingga kita dapat menggunakan power method dan membuat vektor eigen kedua orthogonal dengan vektor eigen pertama. Dengan demikian, kita dapat membuat dua vektor eigen berbeda yang saling orthogonal. Hal ini tidak hanya berlaku untuk dua vektor, tetapi berlaku juga untuk banyak vektor eigen. Inilah yang disebut dengan orthogonal iteration.

Orthogonal iteration adalah versi blok dari power iteration yang biasa disebut juga sebagai “Stimultaneous Power”. Dalam versi ini, nilai A dikalikan tidak hanya pada satu vektor, tetapi pada banyak vektor sekaligus yang nantinya akan disimpan di suatu matrix. Setiap kali melakukan iterasi kita harus menormalisasi vektornya. Salah satu caranya dengan memanfaatkan algortima QR.

* + 1. Algortima QR

Berikut ini adalah implementasi algortima QR untuk menormalisasi vektor :



BAB 3: Implementasi Library dan Program Bahasa Python (Proses Compress) dan Html (Website)

BAB 4: Eksperimen

BAB 5: Kesimpulan, Saran, dan Refleksi

Referensi

Munir, Rinaldi. “Nilai Eigen dan Vektor Eigen (Bagian 1)”, diakses dari :

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

Munir, Rinaldi. “Nilai Eigen dan Vektor Eigen (Bagian 2)”, diakses dari :

https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-18-Nilai-Eigen-dan-Vektor-Eigen-Bagian2.pdf

Munir, Rinaldi. “Singular Value Decomposition (SVD)”, diakses dari :

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

Alexey. “Power iteration”, diakses dari :

http://mlwiki.org/index.php/Power\_Iteration