

TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ỨNG DỤNG CỦA SVD
ĐỀ TÀI :NÉN ẢNH SỬ DỤNG SVD
QUẢN LÝ CỦA HÀNG MÁY TÍNH

Người hướng dẫn: **Nguyễn Văn Hiệu**
Sinh viên thực hiện: **Nguyễn Thành Trung**
Mã SV: **102220044**

Đà Nẵng, 06/2024

Lời mở đầu:

Trong kỷ nguyên kỹ thuật số, lượng dữ liệu hình ảnh ngày càng tăng nhanh chóng đòi hỏi các phương pháp hiệu quả để lưu trữ và truyền tải thông tin. Nén ảnh là một kỹ thuật quan trọng nhằm giảm dung lượng lưu trữ mà vẫn giữ được chất lượng hình ảnh ở mức chấp nhận được. Trong số các phương pháp nén ảnh, Phân Tích Giá Trị Suy Biến (SVD - Singular Value Decomposition) đã nổi lên như một công cụ mạnh mẽ nhờ khả năng xấp xỉ ma trận ảnh bằng cách giữ lại các thành phần quan trọng nhất của dữ liệu.

Phương pháp SVD không chỉ mang lại hiệu quả trong việc giảm kích thước dữ liệu mà còn đảm bảo chất lượng ảnh sau khi nén vẫn được bảo toàn ở mức cao. Dựa trên nền tảng toán học vững chắc của đại số tuyến tính, SVD phân tách một ma trận ảnh thành ba ma trận con, trong đó các giá trị suy biến biểu thị tầm quan trọng của từng thành phần trong việc tái tạo lại ảnh. Bằng cách giữ lại các giá trị suy biến lớn nhất và loại bỏ những giá trị nhỏ, ta có thể nén ảnh mà không làm mất quá nhiều chi tiết.

Báo cáo này sẽ đi sâu vào nguyên lý của SVD, quy trình áp dụng nó trong nén ảnh, cũng như phân tích những ưu điểm và hạn chế của phương pháp này trong thực tế.

.....

Giới thiệu về SVD:

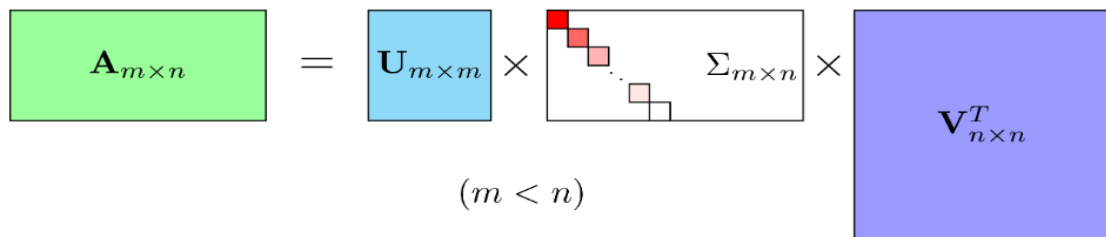
Phân Tích Giá Trị Suy Biến (SVD) là một kỹ thuật quan trọng trong đại số tuyến tính, được sử dụng để phân tách một ma trận bất kỳ thành ba ma trận con. SVD có rất nhiều ứng dụng, đặc biệt trong việc nén dữ liệu, giảm nhiễu và nhận diện mẫu.

Về mặt toán học, **SVD** cho phép phân tích một ma trận $A \in \mathbb{R}^{m \times n}$ (có thể là bất kỳ ma trận nào, không cần vuông) thành ba ma trận:

$$(1) \mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$$

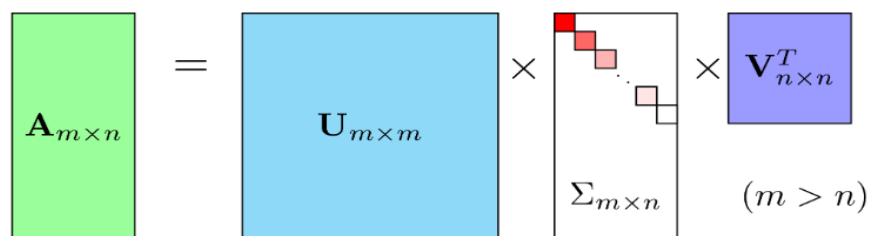
Trong đó:

- $\mathbf{U} \in \mathbb{R}^{m \times m}$ là ma trận trực giao chứa các vector riêng của ma trận $\mathbf{A}\mathbf{A}^T$
- $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$ là ma trận đường chéo chứa các giá trị suy biến (singular values) của \mathbf{A} .
- $\mathbf{V} \in \mathbb{R}^{n \times n}$ là ma trận trực giao chứa các vector riêng của ma trận $\mathbf{A}^T\mathbf{A}$.



$$\mathbf{A}_{m \times n} = \mathbf{U}_{m \times m} \times \mathbf{\Sigma}_{m \times n} \times \mathbf{V}_{n \times n}^T$$

$(m < n)$



$$\mathbf{A}_{m \times n} = \mathbf{U}_{m \times m} \times \mathbf{\Sigma}_{m \times n} \times \mathbf{V}_{n \times n}^T$$

$(m > n)$

Compact SVD:

Viết lại biểu thức (1) dưới dạng tổng của các ma trận rank 1:

$$A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \dots + \sigma_r u_r v_r^T$$

với chú ý rằng mỗi $u_i v_i^T, 1 \leq i \leq r$ là một ma trận có rank bằng 1.

Rõ ràng trong cách biểu diễn này, ma trận A chỉ phụ thuộc vào r cột đầu tiên của U, V và r giá trị khác 0 trên đường chéo của ma trận Σ . Vì vậy ta có một cách phân tích gọn hơn và gọi là *compact SVD*:

$$A = U_r \Sigma_r (V_r)^T$$

Với U_r, V_r lần lượt là ma trận được tạo bởi r cột đầu tiên của U và V . Σ_r là ma trận con được tạo bởi r hàng đầu tiên và r cột đầu tiên của Σ . Nếu ma trận A có rank nhỏ hơn rất nhiều so với số hàng và số cột $r \ll m, n$, ta sẽ được lợi nhiều về việc lưu trữ.

$$A = U_r \Sigma_r (V_r)^T$$

$$= \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \dots$$

Nén ảnh:

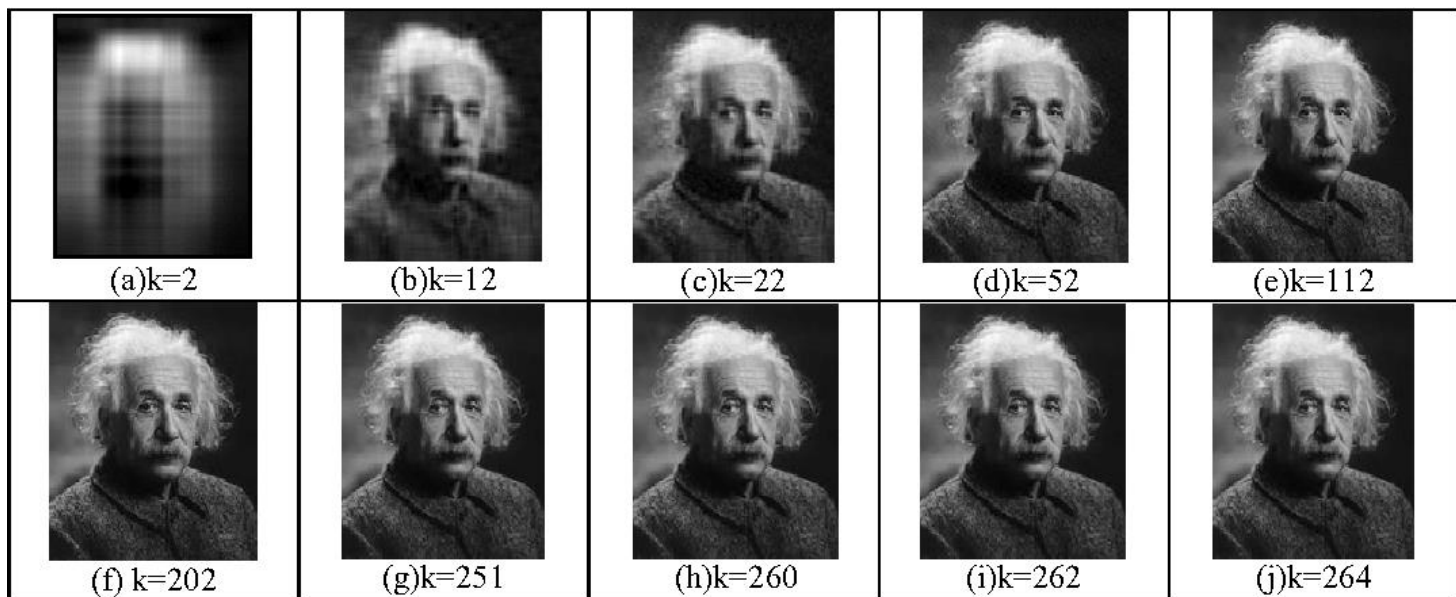
Trong ma trận Σ , các giá trị suy biến thường được sắp xếp giảm dần theo độ lớn. Điều này có nghĩa là các giá trị suy biến đầu tiên chứa nhiều thông tin nhất về ảnh, trong khi các giá trị nhỏ hơn ít ảnh hưởng đến chất lượng ảnh tổng thể.

Để nén ảnh, ta có thể giữ lại một số giá trị suy biến lớn nhất, và bỏ qua các giá trị nhỏ hơn trong ma trận Σ . Khi đó, ma trận A sẽ được xấp xỉ bằng một ma trận có hạng thấp hơn:

$$A_k = U_k \Sigma_k V_k^T$$

Ở đây, k là số giá trị suy biến lớn nhất mà bạn muốn giữ lại.

VÍ DỤ:



Giả sử bạn có một ảnh có kích thước 1000×1000 pixel, nếu bạn lưu toàn bộ ma trận A , sẽ cần 1 triệu giá trị. Nhưng nếu chỉ giữ lại 100 giá trị suy biến lớn nhất, thì bạn chỉ cần lưu các ma trận U , Σ , và V^T tương ứng với 100 cột, điều này giúp giảm kích thước dữ liệu đáng kể.

Khi k càng lớn thì khi nhìn bằng mắt ta sẽ không thấy gì khác biệt vì các giá trị trong ma trận Σ sẽ càng nhỏ dần đều này cho phép ta bỏ đi những giá trị nhỏ không cần thiết đó giúp ảnh lưu được giảm dung lượng đáng kể.

LỢI ÍCH KHI DÙNG SVD NÉN ẢNH:

- Giảm kích thước dữ liệu: Bằng cách giữ lại ít giá trị suy biến hơn, ta có thể giảm số lượng dữ liệu cần lưu trữ, giúp giảm dung lượng ảnh.
- Ảnh xấp xỉ tốt: Dù đã giảm kích thước, ảnh vẫn giữ được phần lớn đặc điểm trực quan nếu chọn đủ giá trị suy biến lớn.
- Điều chỉnh mức độ nén: Bạn có thể điều chỉnh số lượng giá trị suy biến để thay đổi mức độ nén và chất lượng ảnh.

HẠN CHẾ:

- Quá trình tính toán SVD có thể tốn kém thời gian cho các ma trận lớn.
- Nếu giữ lại quá ít giá trị suy biến, ảnh sẽ bị mất nhiều chi tiết và xuất hiện hiện tượng mờ.

Ứng dụng này của SVD rất hữu ích trong việc nén ảnh, giúp tiết kiệm dung lượng lưu trữ mà vẫn giữ được chất lượng ảnh chấp nhận được.

Kết luận:

Phương pháp Phân Tích Giá Trị Suy Biến (SVD) là một công cụ hiệu quả trong việc nén ảnh, cho phép xấp xỉ ma trận ảnh với hạng thấp hơn mà vẫn bảo toàn phần lớn thông tin quan trọng. Bằng cách giữ lại các giá trị suy biến lớn nhất, SVD giúp giảm kích thước lưu trữ và tốc độ truyền tải mà không làm giảm đáng kể chất lượng hình ảnh.

Lợi thế lớn của SVD là tính toán học mạnh mẽ và khả năng áp dụng rộng rãi trong các lĩnh vực khác nhau, không chỉ giới hạn trong xử lý ảnh mà còn có thể được sử dụng trong các ứng dụng như phân tích dữ liệu, học máy và xử lý tín hiệu. Tuy nhiên, hạn chế của phương pháp này là yêu cầu tính toán cao khi xử lý các ma trận lớn, và việc lựa chọn số lượng giá trị suy biến phù hợp đòi hỏi cân nhắc kỹ lưỡng giữa chất lượng ảnh và dung lượng nén.

SVD đã chứng tỏ là một giải pháp linh hoạt và hiệu quả, góp phần vào sự phát triển của các công nghệ nén dữ liệu hiện đại, và tiếp tục là một công cụ hữu ích trong tương lai của xử lý và phân tích hình ảnh số.

Tài liệu tham khảo:

<https://dmicz.github.io/machine-learning/svd-image-compression/>

https://people.math.wisc.edu/~chr/am205/g_act/svd_slides.pdf

<https://machinelearningcoban.com/2017/06/07/svd/>