Compresión SVD

Rodrigo Andrés Muñoz Gomez
Pontificia Universidad Católica de Valparaíso Chile
rodrigo.munoz.g@mail.pucv.cl

***Abstract*—La descomposición de valores singulares (SVD) es una técnica matemática proveniente del álgebra lineal que describe la descomposición de cualquier matriz en tres matrices representativas. En este artículo, se aborda la aplicación de este algoritmo para la compresión y descompresión de imágenes a color, evaluando su efectividad y rendimiento en diferentes espacios de color.**

***Index Terms*—Compresión, Descompresión, Singular value decomposition, matriz.**

I. Introducción

SVD o *Singular value decomposition* describe la descomposición de una matriz entrante MxN en 3 matrices resultantes: U, Σ, y V'.

$$A = U\Sigma V' \tag{1}$$

La ecuación 1 descompone la matriz A entrante en:

- U : Esta matriz contiene vectores singulares izquierdos que representan las relaciones o características importantes entre las columnas.
- Σ : La matriz sigma contiene en su diagonal los valores singulares, los cuales están ordenados por relevancia descendente.
- V' : Similar a la matriz U, V' alberga los vectores singulares derechos que representan cómo se relacionan las filas de los datos proporcionados por la matriz A.

A partir de esto, cualquier matriz se puede descomponer en la suma de los productos externos (*outer products*) de los vectores singulares izquierdos y derechos. Dado que la matriz sigma se compone de valores diagonales descendentes, podemos comprimir una imagen truncando la suma explicada anteriormente. [1] [2]

II. SVD en Imágenes

Dado que SVD es aplicable a cualquier matriz sin importar sus dimensiones, podemos aplicar este concepto en imágenes a color. Sin embargo, como una imagen RGB se descompone en 3 matrices para cada canal de dimensiones iguales, podemos aplicar la descomposición SVD para cada canal, lo que resulta en 9 matrices diferentes correspondientes a cada canal RGB. A partir de esto, al truncar la suma de los productos externos (*outer products*), se obtiene una compresión de la imagen en función del porcentaje de valores singulares restantes después de la truncación.

Para tener una idea general del cuanto truncar, es necesario valorizar la importancia de la diagonal de Σ correspondiente a los valores singulares de una imagen. Para ello, se aplica la función *SVD* a una imagen en blanco y negro implicando la función a solo un canal.

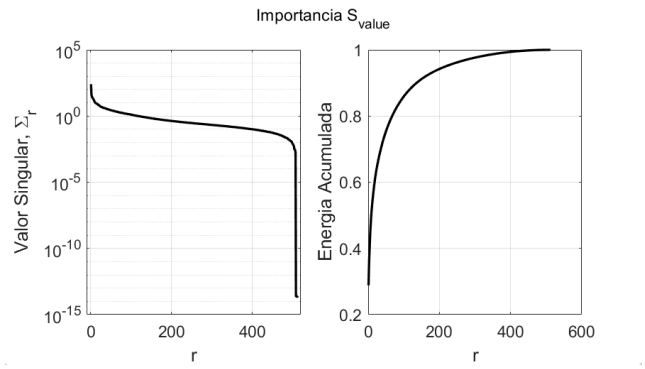


Fig. 1: Relevancia Σ (Lena)

Como reflejan las gráficas de la figura 1, el peso del valor singular disminuye notablemente conforme se avanza en la diagonal de Σ. Por otro lado, La energía acumulada en Σ Aumenta rápidamente, lo cual refleja como las características generales de la imagen se conservan en los primeros valores singulares.

(a) Lena.png (512 x 512)

(b) Perros.png (1024 x 902)

Fig. 2: Imágenes de Referencia

Las figuras 2 muestran las imágenes de referencia utilizadas para aplicar la compresión. La primera corresponde a ”Lena” con una dimensión de $512x512$, y la segunda, ”Perros”, fotografía real con dimensiones de $1024x902$.

III. Funciones Desarrolladas

A. Compresión

En primer lugar, se ha implementado una función de compresión que tiene como objetivo comprimir las matrices $U\Sigma V'$ truncadas para cada canal de una imagen de entrada, permitiendo la variación de los porcentajes de valores singulares

para cada canal y la elección del espacio de color para la compresión.

La función se llama de la siguiente manera:

$$SVD_COMP(Im, P_1, P_2, P_3, C_{Space}, F_{Name}); \quad (2)$$

- Donde los parámetros son los siguientes:
  - Im: La imagen a comprimir.
  - P_1, P_2, y P_3: Los porcentajes de valores singulares para cada canal.
  - C_{Space}: El espacio de color a utilizar, que puede ser RGB, YCbCr, LAB o NTSC.
  - F_{Name}: El nombre del archivo comprimido.
- La función realiza las siguientes operaciones:
  1) Realiza las transformaciones necesarias de RGB al espacio de color especificado.
  2) Utiliza la función *pagesvd* proporcionada por MATLAB para obtener las matrices U, Σ y V' para cada canal.
  3) Trunca cada matriz según el porcentaje proporcionado para cada canal.
  4) Cuantiza cada una de las 9 matrices resultantes en formatos de números enteros.
  5) Guarda las matrices comprimidas en un archivo binario junto con información importante como el tipo de dato por matriz, dimensiones de las matrices, espacio de color y datos de cuantización, para poder recuperar la información comprimida posteriormente.

B. Descompresión

La segunda función desarrollada se encarga de descomprimir la imagen comprimida. La función devuelve la imagen reconstruida a partir de la ruta y el nombre del archivo comprimido.

$$Imagen = SVD_DECOMP(File_{path}) \quad (3)$$

Esta función busca el archivo comprimido a partir de su dirección y nombre. Luego, obtiene los parámetros esenciales de las matrices, como el espacio de color, dimensiones de las matrices, valores mínimos y máximos de cada matriz, así como los bits de cuantización. A partir de esto, se recuperan los datos esenciales para obtener cada una de las 9 matrices, las cuales se descuantizan y redimensionan para obtener las matrices originales. Posteriormente, se multiplican para obtener los 3 canales de la imagen comprimida. Por último, la imagen se convierte nuevamente al espacio de color RGB y se devuelve en formato *UINT8*.

IV. Métricas Utilizadas

A. Peak Signal-to-Noise Ratio

El PSNR mide la calidad de la descompresión de una imagen y se calcula como la relación entre la energía máxima de la señal (en este caso, el valor máximo que puede tener un píxel) y el error cuadrático medio (MSE) entre la imagen original y la imagen comprimida. Un valor de PSNR más alto indica una mayor calidad de descompresión.

$$PSNR = 10 \cdot \log_{10}\left(\frac{V^2}{MSE}\right) \quad (4)$$

donde V es el valor máximo que puede tener un píxel en la imagen (por lo general, 255 para una imagen en escala de grises o 1 para una imagen en blanco y negro).

[3]

B. Structural Similarity Index

El SSIM (*Structural Similarity Index*) compara la similitud estructural de las imágenes, con valores cercanos a 1 indicando una mayor similitud. Se calcula de la siguiente manera:

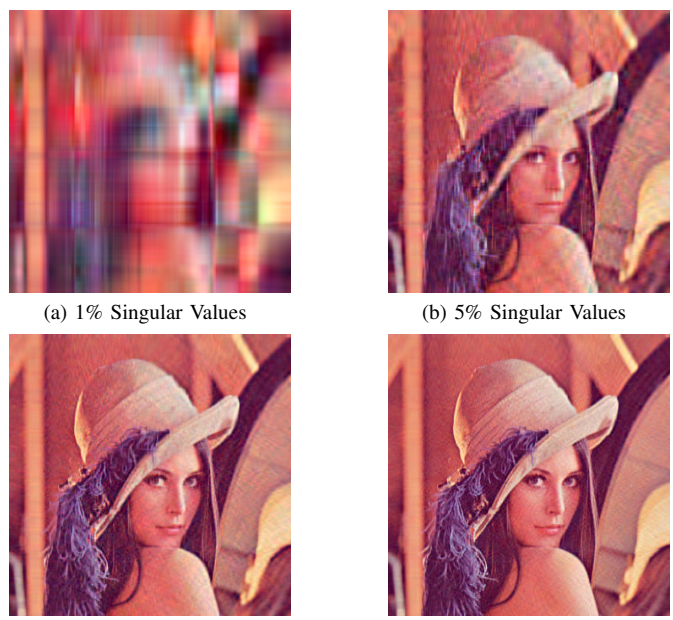
$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (5)$$

SIMM Emplea medidas estadísticas como promedios (brillo promedio de píxeles), varianzas (variación en intensidades) y covarianzas (cómo las intensidades varían juntas) para calcular la similitud perceptual. Esto permite evaluar cómo los cambios en estas propiedades influyen en la similitud visual entre las imágenes. [3]

V. Resultados Experimentales y Discusiones

A. Espacio RGB

El espacio de color RGB (Red, Green, Blue) es un modelo de representación de colores en el que se combinan diferentes intensidades de luz roja, verde y azul para crear una amplia gama de colores. La figura siguiente demuestra el funcionamiento de las funciones establecidas para este espacio de color.



(a) 1% Singular Values (b) 5% Singular Values

(c) 10% Singular Values (d) 13% Singular Values

Fig. 3: Lena Para distintos porcentajes de valores singulares

La figura 3 establece los resultados luego de la descompresión del archivo comprimido para distintos porcentajes de valores singulares equitativos por canal. Podemos notar que

las variaciones en porcentajes pequeños mejoran significativamente la calidad de la imagen. Por otro lado, al tratar con porcentajes mayores, los beneficios son marginales, lo que se explica por la gráfica de energía acumulada mencionada anteriormente.

Lena.tif - 769[Kb]			
$S_{\text{Values}}[\%]$	PSNR	SSIM	CT [%]
1	20.04	0.89	2.94
5	26.60	0.96	17.81
10	30.06	0.98	37.41
13	31.75	0.98	52.42
Perros.png - 2707[Kb]			
$S_{\text{Values}}[\%]$	PSNR	SSIM	CT [%]
1	20.57	0.73	3.18
5	26.63	0.84	18.80
10	29.85	0.90	39.91
13	31.34	0.92	51.9

TABLE I: Metricas RGB

Como se representa en la tabla I, podemos notar que a medida que el porcentaje de valores singulares aumenta, su error disminuye. Sin embargo, para porcentajes muy bajos de valores singulares el error sigue siendo bastante aceptable. Por otro lado, el porcentaje de compresión (CT) respecto al tamaño original, aumenta directamente proporcional con el porcentaje de valores singulares hasta el punto de sobrepasar el tamaño de la imagen original en donde la función de compresión pierde su efectividad completamente.

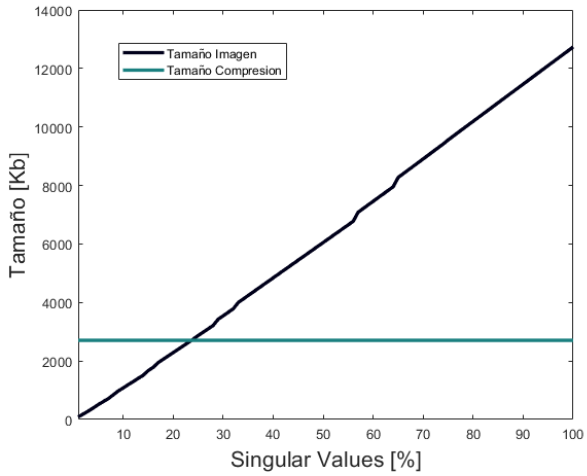
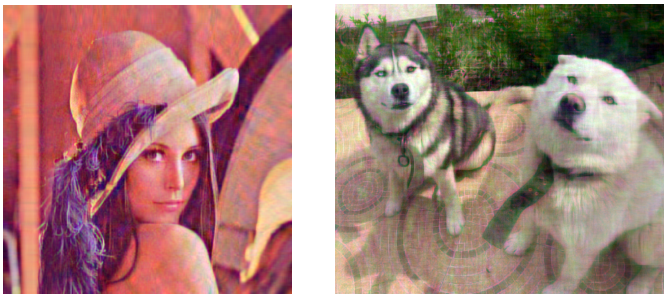


Fig. 4: Peso [Kb] V/S $S_V[\%]$ (Lena)

La gráfica representada en la figura 4, muestra el tamaño de la compresión para los porcentajes equitativos por canal desde el 1% al 100% de valores singulares,en contra del tamaño original de la imagen. Esta gráfica, demuestra el reducido espacio para efectuar una compresión eficiente para las funciones diseñadas.

Dado que se maneja el porcentaje de valores singulares de manera equitativa, puede ser beneficioso para la compresión utilizar distintos porcentajes por canal. Intentando disminuir el tamaño de la imagen y aumentando el intervalo de compresión establecido. La siguiente figura demuestra la igualdad de porcentajes para valores singulares por canal de las dos imágenes experimentales en RGB.



(a) Lena.tif (512 X 512) (b) Perros.tif (1024 X 902)

Fig. 5: $S_V\%$ distintos por canal.

Sin embargo, como podemos notar en la imagen 5, utilizar distintos porcentajes por canal aumenta el error de la imagen resultante. Esto se debe a que el espacio de color RGB presenta una mayor correlación entre canales.

B. Diferentes Espacios de Color

Los espacios de color YUV, CIELAB, NTSC y YCbCr separan la información de luminosidad y crominancia, lo que los hace aptos para la compresión SVD. Esto permite aplicar técnicas de compresión de manera más efectiva en cada componente por separado, reduciendo la redundancia en datos de crominancia y logrando una compresión eficiente sin pérdida significativa de calidad perceptual en imágenes.

Por otra parte, como el ojo humano es mayormente sensible a la luminancia que a la crominancia, en estos espacios de color la desigualdad entre los porcentajes de valores singulares por canal no representaran un error visual significativo en contraste a RGB.

Lena.tif - 768[Kb]						
C_s	$S_1\%$	$S_2\%$	$S_3\%$	$PSNR$	$SSIM$	$CT\%$
RGB	13	5	5	27.60	0.96	29.35
CIELAB	14.38	2.6	2.5	30	0.98	24.81
YCbCr	14.38	2.6	2.5	30.04	0.98	24.81
YUV	14.50	2.65	2.55	30.12	0.98	25.03
NTSC	14.8	2.6	2.6	30.08	0.98	25.34
Perros.tif - 2640[Kb]						
C_s	$S_1\%$	$S_2\%$	$S_3\%$	$PSNR$	$SSIM$	$CT\%$
RGB	13	1	1	22.31	0.72	19.42
CIELAB	13	1	1	30.19	0.90	19.42
YCbCr	12.20	1	1	30.01	0.90	18.38
YUV	12.5	1	1	30.16	0.90	18.82
NTSC	12	1	1	30.02	0.90	18.10

TABLE II: Métricas Espacios de Color

como podemos notar en II, comprimir una imagen en un espacio de color que no sea mayormente interdependiente entre sus canales, resulta ser beneficioso para una compresión óptima, disminuyendo el tamaño proporcional a la imagen, y por otro lado manteniendo su error.

VI. Conclusiones

En general, la compresión de imágenes utilizando *SVD* es sumamente efectiva, ya que puede reducir significativamente el tamaño de una imagen a color. Esto demuestra la relevancia de este algoritmo en la obtención de estructuras principales para ciertos tipos de datos. En el caso de las imágenes, aunque en el espacio de color RGB se obtiene un buen resultado de compresión, se ha demostrado que al explorar nuevos espacios de color y controlar la importancia de los valores singulares por canal, se logra una mejora en la compresión debido a cómo se organiza la representación del color en el espacio. Por último, es interesante plantear la idea de dejar de lado las pruebas empíricas y en su lugar implementar una función compresora capaz de encontrar cada parámetro en función de un objetivo deseado, lo que automatizaría el proceso de compresión por canal y haría que la función sea mucho más abstracta.

References

[1] Stephen Boyd, ”Introduction to Linear Algebra.”
[2] GILBERT STRANG, ”Linear Algebra and Learning from Data.”
[3] L. Brun, A. Tr´emeau, Digital Color Imaging Handbook, CRC Press, 2002, Ch. Color Quantization, pp. 589–638.