FAST IMAGE COMPRESSION USING ENHANCED SINGULAR VALUE DECOMPOSITION

Harris KRISTANTO and Wannida SAE-TANG
The Sirindhorn International Thai-German Graduate School of Engineering,
King Mongkut's University of Technology North Bangkok, Bangkok, Thailand
E-mail: harris.h-sse2015@tggs-bangkok.org and wannida.s.sse@tggs-bangkok.org

***Abstract*—This paper proposes a new method called as expected K-dimension (EKD) for dealing with SVD image compression. The hardest thing is to find the best K-dimension or the redundancy level. The reason is that if the K-dimension that being chosen is too high, it means there is redundant data (more information can be cut), while if the K-dimension is set too low, it means some important information can be missing. The proposed method finds the minimum gap between values in S matrix and then utilizing the remaining values to determine the K-dimension. The results show that the proposed EKD method gives the result as good as the conventional SVD in terms of compression ratio and PSNR, but it reduces the processing time and the complexity drastically.**

Keywords. Expected K-dimension, Singular value decomposition, Discrete cosine transform, JPEG, Image compression

I. INTRODUCTION

The need of technology is undoubtedly increasing time by time. It is clearly seen from the number of Internet usage in the past 10 years shown in Fig. 1 [1]. Due to this increase, of course, it leads to human dependence towards technology. One of an example is when sending some data (could be file, image, music, ..., etc.), people tend to use technology rather than the conventional methods. Moreover, the higher number of bandwidth to handle data exchange process and the memory capacity to store data either temporarily or permanently are needed. From these problems, a lot of researchers have tried to maximize the usage of bandwidth and data storage by using some compression techniques. In image compression field, there are lots of compression techniques that exist today, for example, JPEG [2], JPEG 2000 [3], JPEG-LS [4], singular value decomposition (SVD) [5], ..., etc.

In SVD method, in order to compress an image, the number of eigen values that is required to reconstruct an image is reduced. Based on this research, it was concluded that SVD performs a better image compression in terms of having high standard deviation (higher pixel quality) than JPEG which uses discrete cosine transform (DCT) [5]. However, the hardest thing when dealing with SVD is to find the best K-dimension or the redundancy level. The reason is, if the K-dimension that being chosen is too high, it means the data still too redundant (there are still more information that can be cut), while if the K-dimension is set too low, it means some important information can be missing.

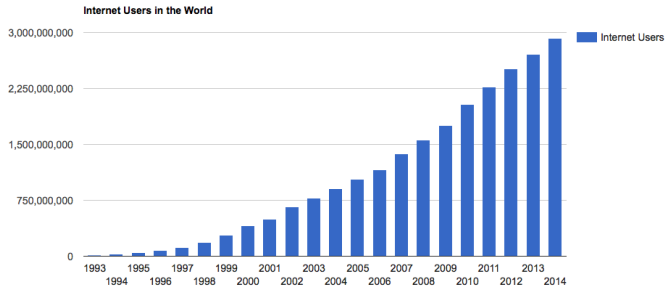


Fig. 1. Internet usage [1].

This paper proposes a new method called as EKD for dealing with SVD image compression. The proposed method finds the minimum gap between values in S matrix and then utilizing the remaining values to determine the K-dimension. The results show that the proposed EKD method gives the result as good as the conventional SVD in terms of compression ratio and PSNR, but it reduces the processing time and the complexity drastically.

The rest of this paper is organized as follows. Section II provides image compression background and literature reviews. Section III proposes a method of finding the EKD. Experimental results and discussions are given in Section IV. Finally, Section V concludes this paper.

II. PRELIMINARIES

Data compression is a process of converting an input data stream (the source stream or the original raw data) into another data stream (the output, the bit-stream, or the compressed stream) that has a smaller size, this means that compression data is done by changing its representation from inefficient to efficient form [6]. Image compression term is data compression technique that specifically converts an image in the form of matrix to code-streams. Image compression is divided into two types, which are:

- **Lossy**
  Lossy compression technique is any data compression that permanently delete/reduce some information of its

original due to compression processes, but it leads to a better compression.

- **Lossless**
  Lossless compression technique is a data compression that can be restored completely even after the compression process (no information losses).

It was concluded that SVD performs a better image compression in terms of having high standard deviation (higher pixel quality) than JPEG which uses discrete cosine transform (DCT) [5]. Even though SVD is better than DCT in terms of high standard deviation, DCT tends to give a really high-energy compaction while SVD tends to give more optimal energy compaction.

In SVD method [5], to compress an image, one of SVD property that is, the rank of matrix A is equal to the number of its nonzero singular values, was applied. The different K-term of singular value was used to eliminate the small singular values or the higher ranks to reduce the number of eigen values. In order to have an optimal energy compaction, it is required to choose the best number of eigen values. The lower the number of eigen values becomes, the smaller the compressed image size becomes. However, the image quality should also be considered. It is not an easy task to find the best number of eigen values that leads to the optimum compression ratio with the optimum image quality. Much processing time could be required for trial and error. To solve the problem, Section III proposes a suitable method of finding the EKD.

III. Proposed Method

This section describes the proposed EKD. This method works by comparing the differences (gaps) of each adjacent values in the singular matrix. The lowest gap, later, will be chosen to determine the K-dimension. Figure 2 shows the flowchart of the proposed EKD.

A. Input Image

In this first step, the system receives an image. The system then checks the color format of the input image. If it is not a gray-scale image, the system will divide the image into N_c other matrices, where N_c represents the number of channels in the image.

B. Singular Value Decomposition

After dividing the image into N_c channels, the next step is to optimize the intensity level of each channel by flattening the standard deviation of the corresponding channel. SVD is used to decompose a matrix to three other matrices as described by Eq. 1.

$$X = USV^t, \tag{1}$$

where X is the $m \times n$ original image as Eq. 2.

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix} \tag{2}$$

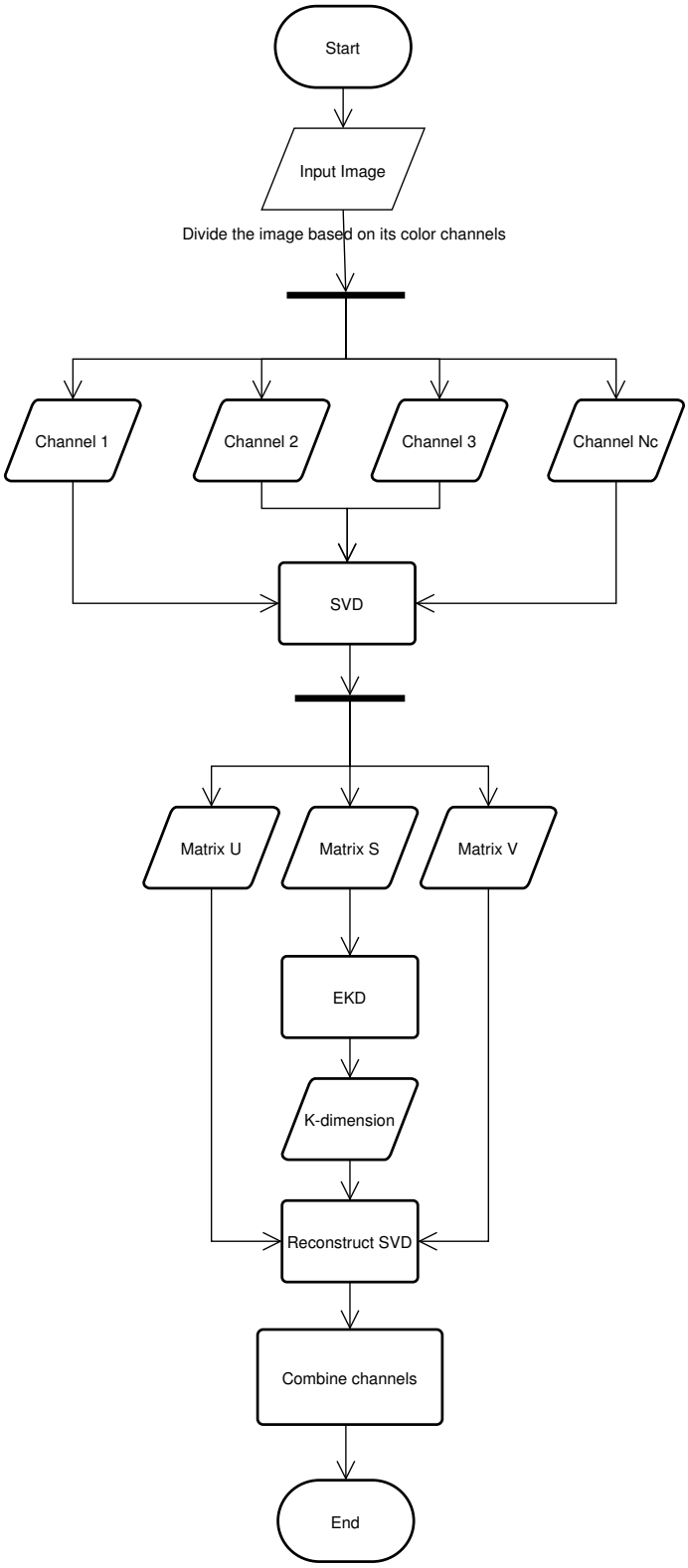


Fig. 2. Flowchart of proposed method.

U is $m \times m$ orthogonal matrix that shows the original row as Eq. 3.

$$U = \begin{bmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1m} \\ u_{21} & u_{22} & u_{23} & \dots & u_{2m} \\ u_{31} & u_{32} & u_{33} & \dots & u_{3m} \\ \dots & \dots & \dots & \dots & \dots \\ u_{m1} & u_{m2} & u_{m3} & \dots & u_{mm} \end{bmatrix} \tag{3}$$

S is the diagonal matrix that contains singular values as described in Eq. 4.

$$S = \begin{bmatrix} \sigma_{11} & 0 & 0 & \dots & 0 \\ 0 & \sigma_{22} & 0 & \dots & 0 \\ 0 & 0 & \sigma_{33} & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \sigma_{mm} \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{4}$$

where

$$\sigma_{11} \geq \sigma_{22} \geq \sigma_{33} \geq \dots \geq \sigma_{mm}. \tag{5}$$

V^t is the transpose of orthogonal $n \times n$ matrix that shows the original column as described in Eq. 6.

$$V^t = \begin{bmatrix} v_{11} & v_{12} & v_{13} & \dots & v_{1n} \\ v_{21} & v_{22} & v_{23} & \dots & v_{2n} \\ v_{31} & v_{32} & v_{33} & \dots & v_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ v_{n1} & v_{n2} & v_{n3} & \dots & v_{nn} \end{bmatrix} \tag{6}$$

C. Expected K-Dimension

Determination of redundancy level or K-dimension is a crucial thing to compress an image in the SVD method. The proposed EKD process is used to find this K-dimension. This step is divided into two other processes, which are:

1) Finding the lowest gap: The aim of this process is to find the minimum variance gap which is the difference of two adjacent σ values in S matrix. The reason of finding the lowest gap is that the lowest gap implies that increasing the K-dimension does not contribute much higher quality of the image. In other words, different information between two matrices are less. Therefore, the lower K-dimension should be chosen in order to achieve a better compression rate. The formula in Eq. 7 describes how to find the minimum variance gap.

$$temp_K = min[\sigma_{i,i} - \sigma_{i+1,i+1}] \forall_i \in \{1, 2, ..., n-1\}, \tag{7}$$

where $temp_K$ denotes the temporary K value.

2) Average of variance: After finding temporary K-dimension, the next step is to enhance the K-dimension by comparing the diagonal values $\sigma_{temp_K,temp_K}, \sigma_{temp_K+1,temp_K+1}, \dots, \sigma_{n,n}$ with the average value of them described by Eq. 8.

$$ave = \frac{\sum_{i=temp_K}^{n} \sigma i, i}{n - temp_K + 1} \tag{8}$$

The maximum diagonal value that is not greater than the average value is chosen as described by Eq. 9.

$$\sigma_{K,K} = \max_{\forall_i \in \{temp_K, temp_K+1, \dots, n | \sigma_{i,i} \leq ave\}} \sigma_{i,i}, \tag{9}$$

where $\sigma_{K,K}$ is the chosen diagonal value that corresponds to the chosen K-dimension. It is noted that $K \leq n$.

D. Reconstructing SVD

After K-dimension is found from the previous step, compression is done by eliminating the upper elements in U, S, and V matrices based on the chosen K-dimension. Then, the compressed single band image is reconstructed by using Eq. 10.

$$X' = U'S'(V^t)', \tag{10}$$

where X' is the $m \times n$ compressed single band image, U' is the $m \times K$ orthogonal matrix, S' is the $K \times K$ diagonal matrix, and $(V^t)'$ is the $K \times n$ orthogonal matrix as illustrated by Eqs. 11, 12 13, and 14, respectively.

$$X' = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{bmatrix} \tag{11}$$

$$U' = \begin{bmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1K} \\ u_{21} & u_{22} & u_{23} & \dots & u_{2K} \\ u_{31} & u_{32} & u_{33} & \dots & u_{3K} \\ \dots & \dots & \dots & \dots & \dots \\ u_{m1} & u_{m2} & u_{m3} & \dots & u_{mK} \end{bmatrix} \tag{12}$$

$$S' = \begin{bmatrix} \sigma_{11} & 0 & 0 & \dots & 0 \\ 0 & \sigma_{22} & 0 & \dots & 0 \\ 0 & 0 & \sigma_{33} & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \sigma_{KK} \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{13}$$

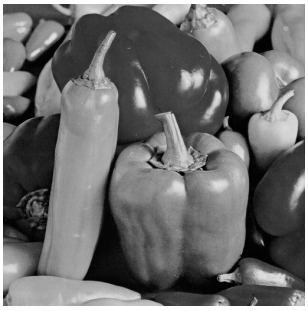
$$(V^t)' = \begin{bmatrix} v_{11} & v_{12} & v_{13} & \dots & v_{1n} \\ v_{21} & v_{22} & v_{23} & \dots & v_{2n} \\ v_{31} & v_{32} & v_{33} & \dots & v_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ v_{K1} & v_{K2} & v_{K3} & \dots & v_{Kn} \end{bmatrix} \tag{14}$$

E. Reconstructing the Image

The last step is reconstructing the image. In this step, if the input image is a color image, the results from the previous steps are combined into one image.

Image	Conventional SVD			Proposed EKD		
	Compression bit rate (bpp)	PSNR (dB)	Processing time (second)	Compression bit rate (bpp)	PSNR (dB)	Processing time (second)
Fruit	1.043	102.316	4.598	1.043	102.316	0.415
Lena	0.996	99.306	4.96	0.996	71.467	0.329
People	0.734	92.69	2.482	0.734	70.937	0.244
Road	1.881	68.255	34.976	1.88	55.12	1.054
View	1.552	102.24	12.634	1.552	105.251	0.665

TABLE I
COMPRESSION PERFORMANCE AND PROCESSING TIME OF CONVENTIONAL SVD AND PROPOSED EKD



(a) Fruit

(b) Lena

(c) People

(d) Road

(e) View

Fig. 3. Test images

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

The aim of this testing is to find out the comparison between the proposed method and SVD in terms of bit rate, peak signal-to-noise ratio (PSNR), and processing time. Five gray images: "Fruit," "Lena," "People," "Road," and "View" shown in Fig. 3 are used in the experiment. The specification of the computer that is being used to do the experiment are listed as:

- Processor: 2.6 GHz Intel Core i7
- Memory: 8 GB 1600 MHz DDR3
- Graphics: Intel HD Graphics 4000 1536 MB.

Table I shows the experimental results. From the results, the EKD method tends to have PSNR almost as high as the conventional SVD method and bit rate as low as the conventional SVD. But the main focus here is the processing time. The time needed by the proposed EKD method is in the range of 0.244 - 1.054 seconds, while for the conventional SVD, it needs 2.482 - 34.976 seconds. This means that the EKD method works faster than the conventional SVD, but it can give almost the same results as the conventional SVD.

Therefore, the proposed method seems to overcome the problem in finding the best K-dimension. To be precise, by finding the lowest gap between values in S matrix, it leads the researchers to proof their hypothesis, which is "the lowest gap between two information means that these two information are quite similar." Eventually, the researchers would like to emphasize that finding the average of remaining variances also helps improve determining K-dimension.

V. CONCLUSION

One of the hardest things to do when dealing with SVD is to find the best K-dimension or the redundancy level. The reason is that if the K-dimension that being chosen is too high, it means there is redundant data (more information can be cut), while if the K-dimension is set too low, it means some important information can be missing. In this paper the researchers have proposed a new method called as expected K-dimension (EKD). It works by finding the minimum gap between values in S matrix and then utilizing the remaining values to determine the K-dimension. The results show that the proposed EKD method gives the result as good as the conventional SVD in terms of compression ratio and PSNR, but it reduces the processing time and the complexity drastically.

REFERENCES

[1] Number of Internet Users (2015) - Internet Live Stats, Available from http://www.internetlivestats.com/internet-users.

[2] G. K. Wallace, "The JPEG Still Picture Compression Standard," *IEEE Trans. on Consumer Electronics*, vol. 38, no. 1, Feb. 1992.

[3] T. David and M. Michael, "JPEG2000 Image Compression Fundamentals, Standards and Practice," *The Springer International Series in Engineering and Computer Science*, vol. 642, 2002.

[4] Weinberger, M. J., G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Trans. on Image Processing*, pp. 1309–1324, 2000.

[5] S. Kahu, and R. Reena, "Image Compression Using Singular Value Decomposition," *International Journal of Advancements in Research & Technology*, 2015.

[6] D. Salomon, "Data Compression," *Springer*, 2015.