Exploration of SVD for Image Compression and Time Series Processing

1st Jesse Redford
Department of Mechanical Engineering
University of North Carolina Charlotte
Charlotte, NC, USA
tredford@uncc.edu

2nd Xingjie Li
Department of Mathematics and Statistics
University of North Carolina Charlotte
Charlotte, NC, USA
xli47@uncc.edu

2020 IEEE MIT Undergraduate Research Technology Conference (URTC) | 978-1-7281-7571-3/20/$31.00 ©2020 IEEE | DOI: 10.1109/URTC51696.2020.9668891

***Abstract*—This paper explores the block-based singular-value-decomposition (SVD) with applications to image compression and processing in both static and dynamic cases. Results include a comprehensive performance study of rank-1 asymmetric block sizes in comparison to conventional uniform square blocks for static image compression. In addition, a method of movement detection for video streaming applications are presented and discussed in this paper. We show that by tracking the Frobenius norm of the derivatives of sequential images, represented as a time series matrix, allows one to infer change points in varying lighting conditions and foreground movement in video sequences. We show that by combining SVD background subtraction we can create clearer distinction between movements and changes in lighting intensity. It is also demonstrated that one can maintain the original derivative feature after re-scaling the images of the video stream to a fraction of their original resolution via local averaging.**

I. Introduction

Singular value decomposition (SVD) has numerous applications in image processing, including compression, face recognition, motion detection, etc. However, SVD is computationally demanding in the presences of large matrices. In addition, global implementations offer little flexibility for instances where regions or time varying features are of interest. In this study we investigate the application of block-based SVD in static image compression and dynamic background subtraction for enhanced movement detection in video streams. In section II, theoretical metrics are proposed to evaluate the compression performance of different block sizes in terms of peak-signal-to-noise-ratio (PSNR), time complexity (TC), and compression factor (CF). We empirically investigate various implementation settings to compress a gray scale image and compare the corresponding performances. We find that block based approaches with adaptive sizes offer more options to balance the quality and amount of compression, particularly for applications which require one to uniquely partition a matrix to accommodate select features of interests [1]. In section III, we study the derivatives of image sequences represented as time series matrices and propose a method to detect streaming movements by evaluating the Frobenius norms of its derivative. SVD background subtraction is employed to remove static artifacts and illumination changes in the time series data to improve detection estimates and change point characterization. We test several cases which included periodic and abrupt movement in the presences of varying illumination. We find that by combing SVD background subtraction we can create a clearer distinction of movements and changes in lighting intensity. It is also demonstrated that one can maintain the original derivative feature after re-scaling the images of the video stream to fraction of there original resolution via local averaging. [2] [3]

This project was funded by the NSF CAREER DMS-1847770

978-1-5386-5541-2/18/$31.00 ©2018 IEEE

II. Singular Value Decomposition

A. Block Based Compression

Singular value decomposition is a robust matrix decomposition method which can be preformed on any arbitrary shape of matrix [4], [5]. Common SVD compression schemes generally entail block based implementations, which often yield better compression performance and reduce processing time. Block based approaches also give rise to more adaptive implementations, particularly for applications which require one to uniquely partition a matrix to accommodate select features. For global SVD factors such as image reconstruction quality and compression rates depended on the estimated rank used to represent a given image matrix, hence in a block based implementation the compression performance is contingent upon the series of rank estimates used to describe the approximation of the image matrix as a whole. A global SVD implementation takes $O(M^2N)$, $M > N$ floating operations as the computation complexity [6]. For block-based implementations, evaluation of time cost (TC^b) and compression factor (CF^b) of a block-based approach are assessed using Equation (1) and Equation (2), respectively.

$$TC^b = \sum_{b_i} \left[\max\left(m_{b_i}, n_{b_i}\right)^2 \cdot \min\left(m_{b_i}, n_{b_i}\right) \right] \frac{M}{m_{b_i}} \times \frac{N}{n_{b_i}} \tag{1}$$

and

$$CF^b = \sum_{b_i} \frac{MN}{\left[k_s\left(1 + m_{b_s} + n_{b_s}\right) \times \left(\frac{M}{m_{b_i}} \times \frac{N}{n_{b_i}} \right) \right]} \tag{2}$$

where $M \times N$, is the original matrix dimensions and $m_{b_i} \times n_{b_i}$, is a collection of smaller matrices.

B. A Rank-1 Approach for block SVD Compression

We fix the truncated rank $k_{b_i} \equiv k = 1$ for each sub-block and divide the image matrix into a series of block matrices. In this setting, an emphasis is placed on the identification of optimal block dimensions m_{b_i}, n_{b_i}, rather than the rank of the image as a whole. Figure 1 represents examples of image reconstruction quality for a rank-1 update using commonly employed block sizes of 16×16 and 8×8, versus different divisible factors to compress the gray scale image.

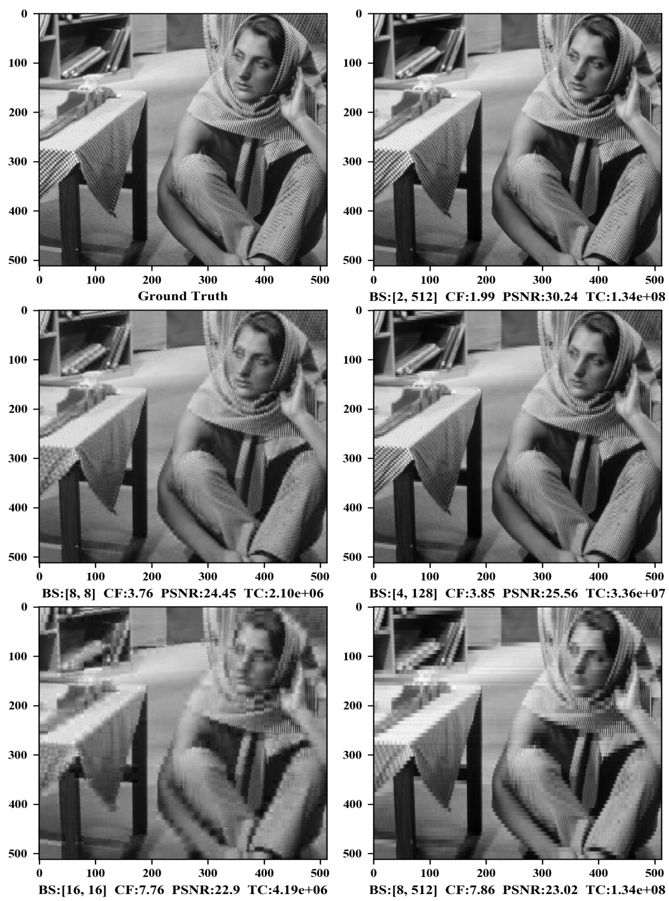

Fig. 1. Compressed images of Barbara using various rank-1 sub-matrix approximations. Block-size is denoted as (BS), compression Factor CF^b, peak-signal-to-noise-ratio (PSNR), and theoretical time complexity is TC^b. Source image adapted from [7].

Among different block size options, the second option, that is 2×512, is seen to reduce the image by a factor of 1.99, without an apparent loss in reconstruction quality. From Figure 1, it is observed that 16×16 is the worst from human's point of view, when compared to a block-size of 8×512. Figure 2 reports the recorded compression performance vs PSNR using all divisible factors to compress the image of Barbra given in Figure 1. We provide Figures 2 as one of our contributions to the block SVD analysis for image compression. Our results contribute a numerical experiment in which one can create a reference to find a suitable block size for an application interest, whether that interest be compression, reconstruction quality, real-time performance, or a combination of the three. PSNR is a positively correlated with image quality. Using Figure 2 and 3, a suitable block-size should correspond to a high PSNR value, Low TC, and high CF. Note that the intermediate block sizes on each graph follow the sequence $m \times 2, m \times 4, m \times 8, m \times 16, m \times 32, m \times 64, m \times 128, m \times 256, m \times 512$.

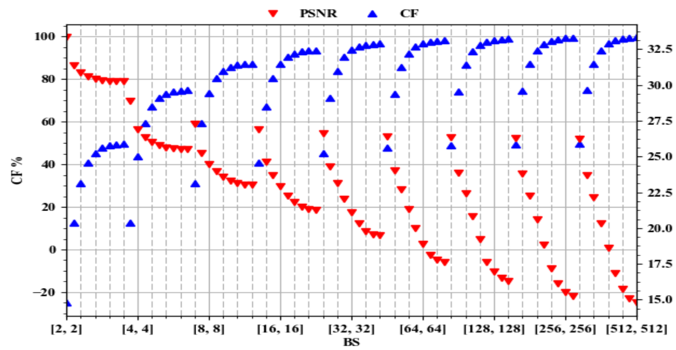

Fig. 2. Compression factor $CF^b\%$ vs $PSNR$ for compressed the image of Barbara using rank-1 update for evenly divisible block-sizes BS.

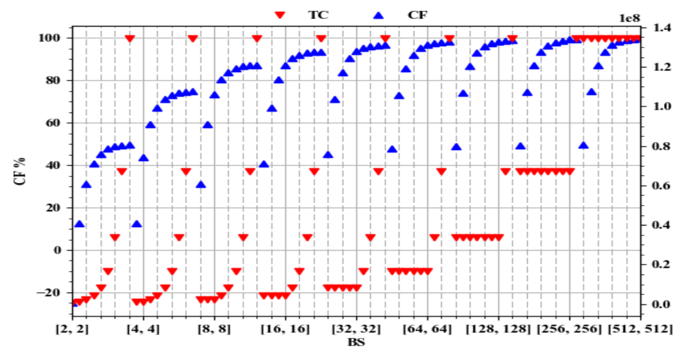

Fig. 3. Compression factor $CF^b\%$ vs $PSNR$ for compressed the image of Barbara using rank-1 update for evenly divisible block-sizes BS.

We find that block-based SVD offers more options to accommodate 'feature' partitions in static cases. However, in the next section, we also find it interesting that in dynamic cases where movement is considered to be a feature and the illumination changes are considered to be interruptions, SVD can repress data disruptions due to abrupt changes in lighting conditions, thus creating a more distinct and reliable signal for detecting movement.

III. Time Series Analysis

Time series data analysis is a topic of interest for a wide variety of industries, ranging from health sciences and weather forecasts to video streams, time dependent data is ubiquitous [8]. In many cases the decomposition of time series into a sequence of components, each having a meaningful interpretation, can be used to uncover trends, slowly varying component(s), in addition to periodicity, movement, activity and anomaly detection [9]. In the application of video processing, one could be less interested in every detail related to movement in a particular video, instead information related to sudden changes which have occurred in the video are more useful [10]. Although, in deriving these related details it is import that the applied method can differentiate between illumination changes and explicit movement in the observable scene, as the inability to make this distinction increases the likely hood of false-positive detection.

A. Representation of the Time Series Matrix

We consider the representation of a time series matrix $\widetilde{A}$ in a discrete-time setting with $t \in \mathbb{Z}$ as the time index of a sequential image matrix $A_{(t)}$, each with dimensions $M \times N$ captured by a static camera. To do so, we first convert $A_{(t)}$ into its column vector representation $\vec{A}_{(t)}$, as shown in (3).

$$A_{(t)} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix}_{M \times N} \rightarrow \vec{A}_{(t)} = \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ a_{12} \\ a_{22} \\ a_{32} \end{bmatrix}_{(M*N) \times 1} \tag{3}$$

Next, we can collect all these column vectors $\vec{A}_{(t)}$ to get the time series representation of $\widetilde{A}$, described by (4).

$$\widetilde{A} = \begin{bmatrix} \vec{A}_{t=0} & \vec{A}_{t=1} & \dots & \vec{A}_{t=T} \end{bmatrix}_{(M*N) \times T} \tag{4}$$

Suppose that a portion of the entries $\vec{A}_{(t)} \in T$ describe a particular feature which is time variant. Without having prior knowledge of the attributes associated with the feature of interest in $\widetilde{A}$, the question becomes how do we detect this feature or infer characteristics of how the feature varies with respect to time? Naturally, the first step of approaching this problem would to be to remove any static information in the times series, a process commonly referred to as background subtraction.

B. Background Subtraction

Background subtraction is a basic problem for movement detection in videos and also the first step of high-level computer vision applications [11]. If we know that $\widetilde{A}$ is likely to be of low-rank, this infers we can preform a background removal operation on the time series matrix $\widetilde{A}$, to obtain a sparser representation without loss of the primary signal of interest. Figure 4 compares three versions of the times series $\widetilde{A}, \widetilde{Ac}, \widetilde{Acm}$. Where $\widetilde{A}$ is the original time series matrix containing a non processed video stream of a cylindrical object moving periodically from the left to right for $t \in [30, 270]$, increasing in frequency and decreasing amplitude. $\widetilde{Ac}$ contains the same video sequence but block SVD background subtraction is applied, and $\widetilde{A_{cm}}$ is the sparse representation of $\widetilde{A}$ which is obtained using global SVD background subtraction given by (5).

$$\widetilde{A_{cm}} = \widetilde{A} - \widetilde{A_k}; (k \leq T \leq M \times N \in Z) \tag{5}$$

Where $\widetilde{A_k}$ is the rank-$k = 1$ approximation of the original time series matrix $\widetilde{A}$. For $\widetilde{Ac}$ a block size of 307200 x 4, is used to remove background information and obtain a sparser representation of a time series without loss of the primary signal of interest.

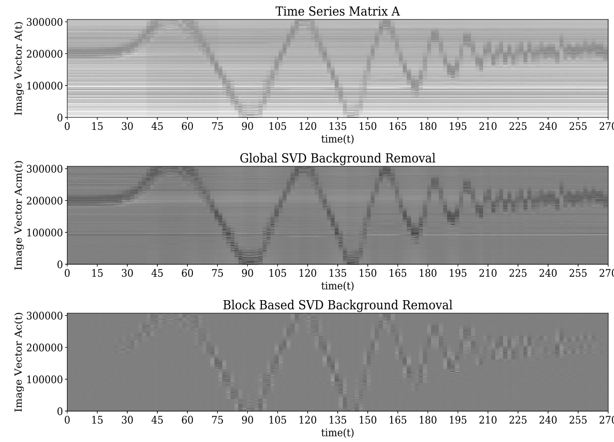

Fig. 4. Comparison of global and block based background subtraction processes for removing static redundancy in the ground space of a time series. Foreground motion occurring $t \in [30, 270]$. Processing time for global and block SVD 10.1 and 4.0 seconds, respectively.

From Figure 4, we see that by preforming background subtraction on the original times series matrix, we obtain a sparser representation without loss of the information describing the objects movement. Using a block based approach enables a higher removal rate of redundant data in comparison to global SVD. Specifically, on the interval $t \in [0, 30]$, where the foreground object is stationary, block SVD background subtraction is able to remove this data from the time series, where global SVD does not.

C. Derivative Norm Trajectories

Consequently, we compute the forward difference to evaluate the change in the time series $\vec{A}$ at some time t and the previous frame at $t-1$ by (6).

$$d\vec{A}(t) := \vec{A}_t - \vec{A}_{t-1} \tag{6}$$

Because $d\vec{A}(t)$ for each $t = 1, \dots, T$ is a column vector, we can compute its norm $S(t)$ and get a trajectory of the forward difference norm at each time t. Notice that, we can also compute the forward difference of a period by considering several time grids as a whole, in both cases, we employ the Frobenius norm for computational efficiency.

$$S(t) = \sqrt{d\vec{A}^T d\vec{A}} = \left\| d\vec{A}(t) \right\|_2 = \left\| d\vec{A}(t) \right\|_F \tag{7}$$

D. Step 1 - Identifying Matrix Scale

We downsize the original observations stored in $\widetilde{A}$, to convert its dimensions from $(M*N) \times T$ to $(M*N)/S_{factor} \times T$, where T is the total amount of samples, $M*N$ is the length of the image vector, and S_{factor} is the scaling factor we apply to each sample in the set T. The potential space savings can be evaluated as $SS = (1 - 1/S_{factor}^2) * 100\%$. In Figure 4, we shown that one can obtain the same norm trajectory of the times series, just on a different scale, regardless of the selected resizing factor. We believe this dissection is important, as downsizing the times series matrix aids computational efficiency, and may be required to implement in real-time settings on limited hardware.

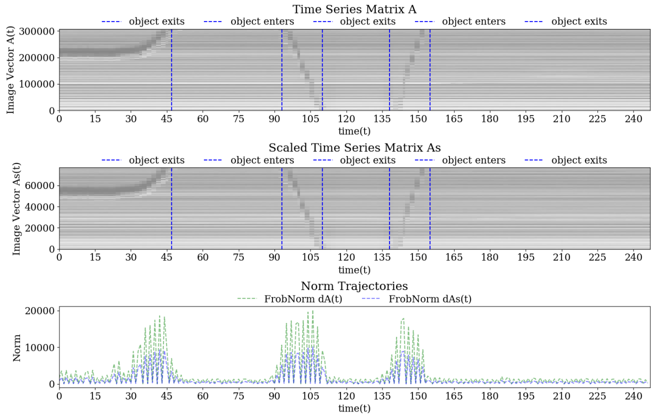

Fig. 5. Comparison of norm trajectories obtained from original times matrix and after scaling the times series matrix to 75% of original size.

From Figure 5, notice that the norm of each forward difference allows one to infer change points in lighting conditions, in addition to the periodicity of stochastic movements in the time series data.

E. Step 2 - Threshold Determination

In Figure 6 we compare a static background with illumination changes only, before and after global SVD background subtraction is applied. From Figure 6, we see that norm signal of $\widetilde{Acm}$ is less sensitive to sudden changes in illumination. Hence, we can decrease the chances of a false detection of motion in the scene by utilizing the norm trajectory of $\widetilde{Acm}$, rather than $\widetilde{A}$.

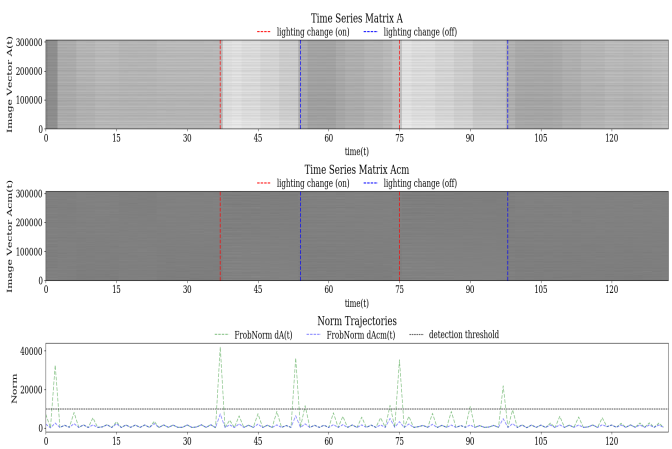

Fig. 6. Determination of a minimum detection threshold by comparison of time series matrix derivative norms under illumination changes before and after global SVD background subtraction is applied.

F. Step 3 - Benchmark of Movement Detection

Figure 7 demonstrates an example where one may encounter a false reading of motion due to periodic changes in luminous intensity. Based on the norm detection threshold of 1000, determined in step 2, 6 false readings are identified for the norm trajectory of $\widetilde{A}$. Whereas the time interval in which physical change has occurred is accurately captured by the norm signal of $\widetilde{Acm}$, the sparse foreground approximation of $\widetilde{A}$ obtained by (5).

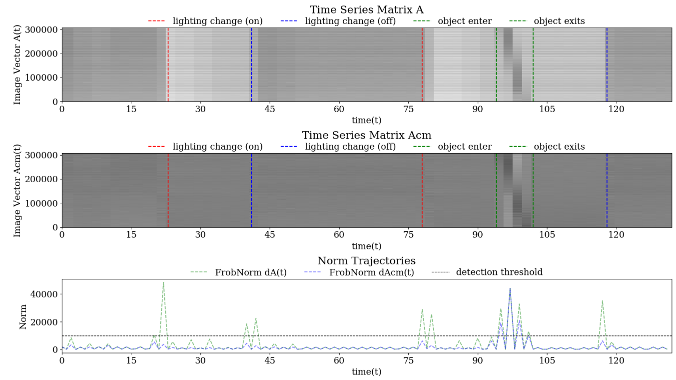

Fig. 7. Evaluation of movement detection performance given by norm trajectories of a time series matrix with and without global SVD background subtraction. Original times series consist of periodic illumination changes with a single instance of foreground movement between the interval $t \in [97, 103]$.

IV. Conclusion and Future Scope

A set of base-line metrics for SVD image block-truncation compression have been presented. We also introduced a movement detection scheme for video streams involving time series analysis. In such cases we used SVD to repress data disruptions and tracked the norm trajectory of a forward difference matrix to measure information changes at any particular point in time. A fast decomposition of a time series which results in a scalar component of instantaneous changes within the data can find use in many real time imaging applications. Periodicity and change detection could be used in surveillance, signal processing, and industrial applications involving imaging inspection and analysis. Future work aims to learn a mapping enabling concurrent localization and segmentation of selected features from the ground space. We believe this method could be useful as a dimensions reduction technique to effectively reduce the size of times series data and improve the learning efficiency of some high-level computer vision model.

References

[1] J. Redford and X. Li. A feature oriented framework and enhanced assessments for imaging compression. *(Accepted) Proceedings of the Fourth IEEE International Conference on Image Processing, Applications and Systems*, 2020.

[2] J Snyman. *Practical mathematical optimization*. Springer, 2005.

[3] K. Arrow and L. Hurwicz. *Constraint qualifications*, volume 9. 1970.

[4] NK El abbadi Et al. Image compression based on svd and mpq-btc. *JCS*, 10(10):2095–2104, 2014.

[5] H. Prasantha, H. Shashidhara, and B. Murthy. Image compression using svd. In *International Conference on Computational Intelligence and Multimedia Applications*, volume 3, pages 143–145. IEEE, 2007.

[6] L. Trefethen and D Bau. *Numerical linear algebra*, volume 50. Siam, 1997.

[7] Allen Gersho. *Barbara*. homepages.cae.wisc.edu/ ece533/images/.

[8] A Agarwal, J. Amjad, D. Shah, and D. Shen. Model agnostic time series analysis via matrix estimation. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2(3):1–39, 2018.

[9] S. Choudhary, G. Hiranandani, and S. Saini. Sparse decomposition for time series forecasting and anomaly detection. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 522–530.

[10] P. Sachan and P. Khanna. Foreground segmentation and change detection using singular value decomposition. pages 1–14, 2019.

[11] L. Guo, D. Xu, and Z. Qiang. Background subtraction using local svd binary pattern. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 86–94, 2016.