A Study of Illumination Filtering and Movement Detection in Video Streaming Processing

1st Mai Phuong Nguyen
Department of Mathematics and Statistics
University of North Carolina Charlotte
Charlotte, NC, USA
pnguye45@uncc.edu

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

2021 IEEE MIT Undergraduate Research Technology Conference (URTC) | 978-1-6654-0595-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/URTC54388.2021.9701635

***Abstract*—We study a movement detection scheme for video streaming data using time series analysis. We employ simple scalar quantities such as (i) the Frobenius norm trajectory of a forward difference matrix and (ii) the Auto-correlation function to analyze the time series and detect movements in the video sequences. To remove the background interruptions, such as the change of lights, we study the performance of two filters, (1) SVD filter and (2) SVD combined with the Butterworth filter, to repress data disruptions. From the example, we identify the best filter and the best statistical quantity to detect movement in the disrupted data. By using the background subtraction filters as well as the Frobenius norm trajectory of data, we can distinguish the movements and light disruptions at various time points in the video sequences. The combo filter is found to be the best as it completely removes the light interruption while maintaining the movements data unaffected and smooths other static artifacts.**

keywords– *illumination filter, SVD, Butterworth, norm trajectory, movement detection, time series analysis*

I. Introduction

Singular value decomposition (SVD) has numerous applications in image processing, including compression, face recognition, motion detection, etc ([1], [2]). Time series analysis is another useful statistical tool for variety of time evolving applications ranging from health sciences, weather forecasts to video streams, when time dependent data is ubiquitous [3]. For translational purposes, information related to sudden changes which have occurred in the video is sometimes preferable to general trends or detailed information related to movement [4]. Moreover, it is important that the applied method can differentiate between background interruptions such as illumination changes and explicit movement in the observable scene, as the inability to remove the illumination effects increases the likelihood of false movement detection. It is also favorable to design simple, computationally inexpensive statistical quantities to measure sudden movement changes, as video sequence data are often large by nature.

In this study, we explore a movement detection scheme for video streaming data using time series analysis. In such cases, we investigate the performance of two filters, (1) SVD filter only and (2) SVD combined with the Butterworth of low pass filter, to repress data disruptions. We also study and compare two popular scalar statistical quantities, (i) the Auto-correlation function (ACF) and (ii) the Frobenius norm trajectory of a forward difference matrix, to measure and detect information changes at any particular time point. We find that using the combo filter as well as the simple Frobenious norm trajectory, the background illumination interruption is completely removed and the signals of object's movements are unaffected while other source of static artifacts is smoothed. In addition, by using the norm trajectory, which is a simple scalar quantity that can be computed very efficiently, we can detect changes of movements easily.

This paper is organized as follows. In section II, we study the time series matrix representation of video sequences data, and review the two popular statistical quantities, (i) Auto-correlation function (ACF) and (ii) the Frobenius norm trajectory of a forward difference matrix, to measure and detect sudden information changes at any particular time point. In section III, we review the two common filtering approaches to remove the interruption due to the sudden change of illuminations: (1) SVD filter only and (2) SVD combined with the Butterworth of low pass filter. In section IV, we test a benchmark example. We find that by combining SVD and Bultterworth low pass background subtraction, we can create a clearer distinction of movements and sudden changes in lighting intensity. In section V, we summarize this work and briefly discuss the future scope.

II. Time Series Analysis

Time series analysis is a useful tool for variety of time evolving applications ranging from health sciences and weather forecasts to video streams, when time dependent data is ubiquitous [3]. In many cases, the data is decomposed into a sequence of components with respect to time, each having a meaningful interpretation, and can be used to uncover trends, slowly varying component(s), in addition to periodicity, movement, activity and anomaly detection [5]. In the application of video data processing, one could be more interested in information related to sudden changes which have occurred in the video [4]. Moreover, it is important that the applied

This project was funded by the NSF CAREER DMS-1847770.

method can differentiate between illumination disruptions and explicit movement changes in the observable scene, as the inability to remove the illumination interruption increases the likelihood of false movement detection.

In this section, we first introduce the notations to represent the video streaming data in the matrix form in subsection II-A, we then introduce the statistical quantities to perform time series analysis for movement detection in subsection II-B.

A. Matrix Representation of Video Streaming Data

We consider the representation of a video streaming data in the format of 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 (1).

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

Next, we can collect all these column vectors $\vec{A}_{(t)}$ to get the representation of entire video, described by $\widetilde{A}$ in (2) and visualized by the top figure in Figure 2:

$$\widetilde{A} = \left[\vec{A}_{t=0} \quad \vec{A}_{t=1} \quad \dots \quad \vec{A}_{t=T}\right]_{(M \times N) \times T} . \quad (2)$$

Suppose that a portion of the entries $\vec{A}_{(t)} \in T$ describes 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 we detect this feature or infer characteristics of this feature that varies with respect to time. Naturally, the first step of approaching this problem is to remove any 'static' information in the times series, a process commonly referred to as background subtraction.

B. Statistics of Movement Detection

Two popular statistical quantities are employed to study and compare the movement detection in the streaming data.

- **Derivative Norm Trajectories**
  We compute the forward difference to evaluate the change in the time series $\widetilde{A}$ at some time t and its previous frame at $t-1$ by (3)

$$d\vec{A}(t) := \vec{A}_t - \vec{A}_{t-1}, \quad t = 1, \dots, T. \quad (3)$$

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

$$S(t) = \sqrt{d\vec{A}^T d\vec{A}} = \left\| d\vec{A}(t) \right\|_2 = \left\| d\vec{A}(t) \right\|_F . \quad (4)$$

- **Auto-Correlation of Time Series Data**
  Autocorrelation function (ACF) of data shows the degree of similarity between a given time series and a lagged version of itself over successive time interval [6]. To compute ACF, we transpose the time series $\widetilde{A}_{(M\times N)\times T}$ into $\widetilde{A}_{T\times(M\times N)}$. Now each row in $\widetilde{A}_{T\times(M\times N)}$ contains a single time series, and the entries of $\widetilde{A}_{T\times(M\times N)}$ can be visualized as follows

$$\widetilde{A}_{T\times(M\times N)} = \begin{pmatrix} \widetilde{A}_{1,1}, & \dots, & \widetilde{A}_{1,(M\times N)}: & \text{Time series } 1 \\ \widetilde{A}_{2,1}, & \dots, & \widetilde{A}_{2,(M\times N)}: & \text{Time series } 2 \\ \dots, & \dots, & \dots, & \dots \\ \widetilde{A}_{T,1}, & \dots, & \widetilde{A}_{T,(M\times N)}: & \text{Time series } T \end{pmatrix}.$$

  We assume that all T time series are nearly stationary with mean value $\overline{\widetilde{A}}$ and the autocorrelation with lag h is computed by [6]:

$$r_h = \frac{\sum_{t=h+1}^{n} (\widetilde{A}_t - \overline{\widetilde{A}})(\widetilde{A}_{t-h} - \overline{\widetilde{A}})}{\sum_{t}^{n} (\widetilde{A}_t - \overline{\widetilde{A}})^2}. \quad (5)$$

  Theoretically, ACF measures the similarity between a current frame at time t and the past frame(s) at time $t-h$. In our study, it estimates the short-time change of object's movements through a successive sequence of data. Meanwhile, ACF cannot fully distinguish the background light intensity or sudden changes happening in the video streaming. No matter the given data is unprocessed or filtered, the ACF at various lag h cannot separate the changes of illumination or movement, which is shown in Figure 1.

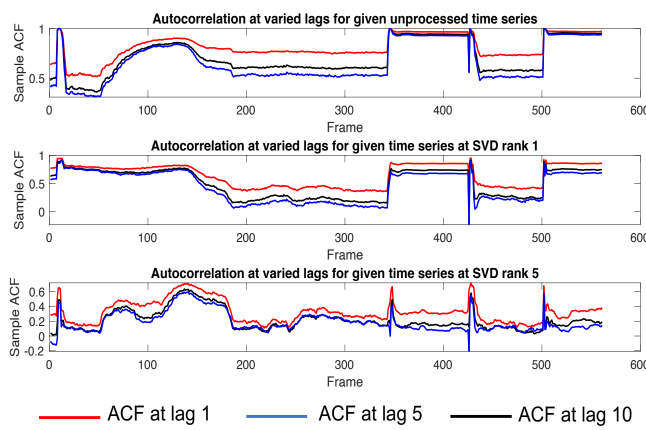

Fig. 1. ACF of unprocessed and processed data at various chosen lag h (h = 1,5,10) for the given time series (frames) data $\widetilde{A}$.

III. Background Illumination Filtering

In this section, we review the common filtering tools to remove the interruptions due to the sudden change of illuminations: (1) SVD filter only and (2) SVD combined with the Butterworth of low pass filter.

- **Singular Value Decomposition Filter**
Singular value decomposition (SVD) is a robust matrix decomposition method which can be performed on any arbitrary shape of matrix [1], [2]. Common SVD compression schemes generally entail block based implementations, which often yield better compression performance and reduce processing time [7].
Background subtraction is a basic problem for movement detection in videos and is also the first step of high-level computer vision applications [8]. If we know that $\widetilde{A}$ is likely to be of low-rank, this infers that 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 2 compares three versions of the time series matrices, $\widetilde{A}$, $\widetilde{A^1_{cm}}$,$\widetilde{A^5_{cm}}$. Where $\widetilde{A}$ on the top figure is the original time series matrix containing an unprocessed video streaming data of T frames ($T = 561$). This video shows an object's movements throughout the total time T (frames) but being interrupted by sudden changes of light intensity occurring at different time. The middle figure $\widetilde{A^1_{cm}}$ contains the same video sequence but the block based SVD background subtraction is applied with rank-1 subtraction, and the bottom figure $\widetilde{A^5_{cm}}$ is obtained by using the block SVD background subtraction with rank-5 subtraction. In general, $\widetilde{A^k_{cm}}$ is defined in (6):
$$\widetilde{A^k_{cm}} = \widetilde{A} - \widetilde{A_k}, \quad \text{for } k \leq \min\{T,\ M \times N\}, \tag{6}$$
where $\widetilde{A_k}$ is the rank-k block based SVD approximation of the original time series matrix $\widetilde{A}$.
- **Butterworth of Low Pass Filter**
SVD filter eliminates noise (light intensity) without distorting the shape of signal (that is the movements of object) [9]. However, its filter at optimal rank-k truncation is not smooth enough and may cause other artifacts, which is demonstrated clearly by the unprocessed norm trajectories (black) in the Figure 4. To have a better filtering tool that removes illumination smoothly while disaffecting the movements, the Butterworth of low pass filter combined with SVD is employed to resolve this issue. The magnitude response of a low pass Butterworth filter is given by [10]:
$$H_{j\omega} = \frac{1}{1 + (\frac{\omega}{\omega_c})^{2N}}, \tag{7}$$
where $j = \sqrt{-1}$, ω is the angular frequency, ω_c is the cut-off value and N equal to the number of reactive elements in a passive filter. Since Butterworth of low pass filter has a maximally flat response, there is no ripple in the pass band [10]. In this study, we use the Butterworth filter with order $= 1$ to minimize the delay and choose $\omega_c = 5$ Hz to filter unwanted noise such as unsteady camera holding, but mostly the noise of illumination. By combining the SVD and Butterworth low pass filter simultaneously, the light intensity can be totally removed while maintaining the pattern of norm trajectories for the movements. We will discuss this with more details in section IV and in Figure 4 with the benchmark example.

IV. Benchmark Examples

In this section, we consider a benchmark example which is composed of an unprocessed video streaming data of T frames ($T = 561$). This video displays an object's movements, as visualized in the top figure of Figure 2. We summarize the feature of data in subsection IV-A, and discuss the results processed by each filter in subsection IV-B and IV-C, respectively.

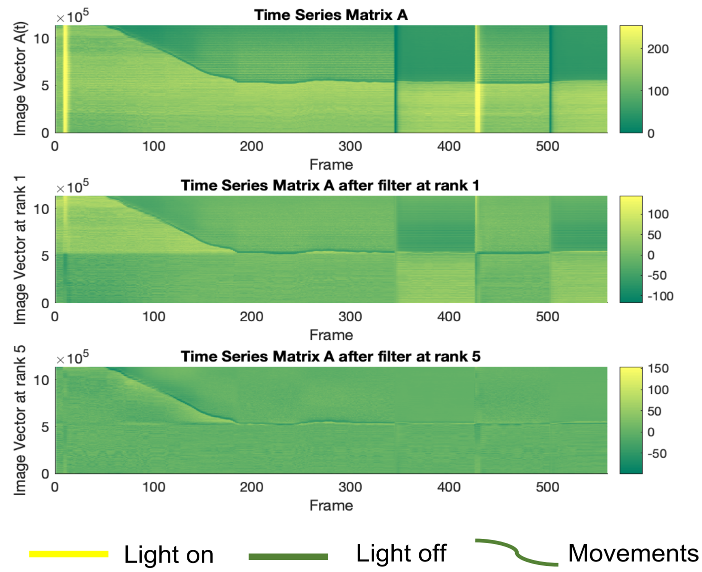

Fig. 2. Comparison of Original Time series matrix with no SVD filter (top), Time series matrix after using SVD filter at rank-1 (middle) and Time series matrix after using SVD filter at rank-5 (bottom) .

A. Features of the Unprocessed Data

This video is interrupted by sudden changes of room light intensity (background illumination) occurring at different time frame t. As displayed in the top figure of Figure 2, the room light suddenly turns on (highlighted by the yellow bar) at the frame number $t = 10$ and $t = 430$, and the room light suddenly turns off (highlighted by the dark green bar) at the frame number $t = 340$ and $t = 500$. The object keeps moving during the entire video and its trajectory is captured by the dark green curve.

Next, we apply the SVD filter to the original data and visualize the resulting representing matrices in the middle and bottom figures of Figure 2.

To find the optimal rank-k for SVD, we manually try $k \in \{1, \ldots, 10\}$. With rank-1 employed, the light-off signal is totally removed, but the light-on still exits. Hence, this lowest rank is not desired as the light interruption is not fully removed. Consequently, we test other values and finally find the optimal value being $k = 5$. From the middle and bottom figures of Figure 2 , we see that by preforming background subtraction on the original times series matrix, we can obtain a sparser representation without loss of the information describing the object's movements. The other higher ranks are not necessary as they are more computationally expensive and they will disrupt the movement signals. In general, the criteria for choosing optimal rank is to ensure the remove of light intensity while keep movements unaffected. Besides, computation cost is another important factor to be considered when processing data with SVD.

Moreover, we use a block based SVD instead of a global SVD on the given data $\widetilde{A}$. Using a block based approach enables a higher removal rate of redundant data in comparison to global SVD. For instance, on the frames interval $t \in [0, 180]$, where the foreground object is nearly stationary, the block SVD background subtraction is able to remove redundant data from the time series, whereas global SVD does not.

B. ACF Trajectory for Detecting Movements

We now apply the time series analysis to detect the movement signals. Following the definition in (5), we first compute the Auto-correlation function (ACF) trajectories of the data with lag $h = 1, 5, 10$, respectively.

From the top figure of Figure 3, we see that after applying the rank-5 SVD filter, the changes of background illumination still dominates the peak values of the ACF trajectories over the pattern of movements at $t = 10$, 340, 430, 500, no matter what lag h is used, though the appropriate choice of using rank-5 is confirmed by visualizing the matrix representation of filtered data. We also compute the ACF trajectories of data filtered by rank-5 SVD together with the Butterworth low pass filter in the bottom figure of Figure 3, the light interruption now completely disappears but the pattern of movement signals is homogenized and blurred at the same time. From these observations, we can conclude that it is not favorable to use ACF as a statistical quantity to detect the movement in the time series analysis of video data.

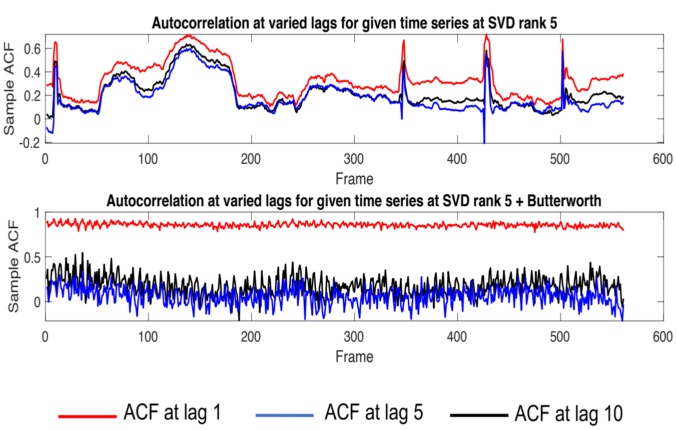

Fig. 3. ACF of filtered data at various chosen lag h (h = 1,5,10) for the given time series (frames) data $\widetilde{A}$.

C. Norm Trajectory for Detecting Movements

This time, we follow the definition in (4) and compute the norm trajectories of forward difference of the filtered data. For the SVD plus the Butterworth low pass filter, we employ the Butterworth filter (7) with order $= 1$ to minimize the delay and choose $\omega_c = 5$ Hz to filter unwanted noises such as unsteady camera holding, but mostly the noise of illumination.

The results of norm trajectories are summarized in Figure 4. Both norm trajectories for filtered data, (1) SVD rank-5 only (magenta) and (2) SVD rank-5 combined with Butterworth low pass filter (blue), remove the peak values due to the light interruption while preserving the pattern of movements very well. Regarding the movements signals, the difference among the unprocessed and filtered data is negligible. In addition, it is clear that the norm trajectory of this combo filter works better than that of the single use of SVD filter. By combining these two filters simultaneously, light intensity is not only totally removed, but also the pattern of movement is unchanged and other static artifacts are smoothed. This advantage will be more significant when we deal with high dimension data of the original time series.

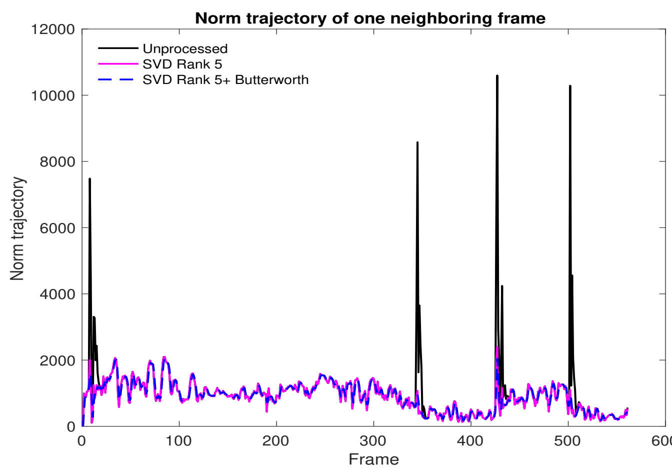

Fig. 4. Norm trajectory of the unprocessed time series data (black); norm trajectory of the filtered data with rank-5 SVD filter only (magenta); norm trajectory of the filtered data with rank-5 SVD combined with the Butterworth of low pass filter (blue) .

V. Conclusion and Future Scope

SVD is a simple but important tool in data compressing and processing. It can be used as a filter to eliminate unwanted noise (such as the background light illumination). In this study, we explore a movement detection scheme for video streaming data with time series analysis. In such cases, we study and compare the performance of two filters, (1) SVD rank-k only and (2) SVD combined with the Butterworth of low pass filter, to repress data disruptions. We also compare two popular statistical quantities, (i) Auto-correlation function (ACF) and (ii) the norm trajectory of a forward difference matrix, to measure and detect information sudden changes at any particular time point. We find that using the combo filter, the background illumination interruption is totally removed and the signals of object's movements are unaffected while other source of static artifacts is also smoothed. In addition, by using the Frobenius norm trajectory, which is a very simple scalar quantity, we can best track the pattern and detect movement changes easily.

A fast decomposition of a time series which results in a scalar component of instantaneous changes within the data can find useful 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. Our future work aims to learn a mapping enabling concurrent localization and segmentation of selected features from the ground space. Besides, we will also work on theoretical criteria to find the optimal value of threshold in filtering background noises and interruption. We will investigate more sophisticated cases when time series of movements become inhomogeneous at various time along with different sources of interruptions in the video data.

REFERENCES

[1] 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.

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

[3] 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.

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

[5] 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.

[6] Jonathan D Cryer and Kung-Sik Chan. *Time series analysis: with applications in R*, volume 2. Springer, 2008.

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

[8] 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.

[9] Woo-Hyuk Jung and Sang-Goog Lee. An r-peak detection method that uses an svd filter and a search back system. *Computer methods and programs in biomedicine*, 108(3):1121–1132, 2012.

[10] P Podder, M Hasan, M Islam, and M Sayeed. Design and implementation of Butterworth, Chebyshev-I and elliptic filter for speech signal analysis. *International Journal of Computer Applications*, 98, 2014.